**Assignment: Bash Shell Basics**

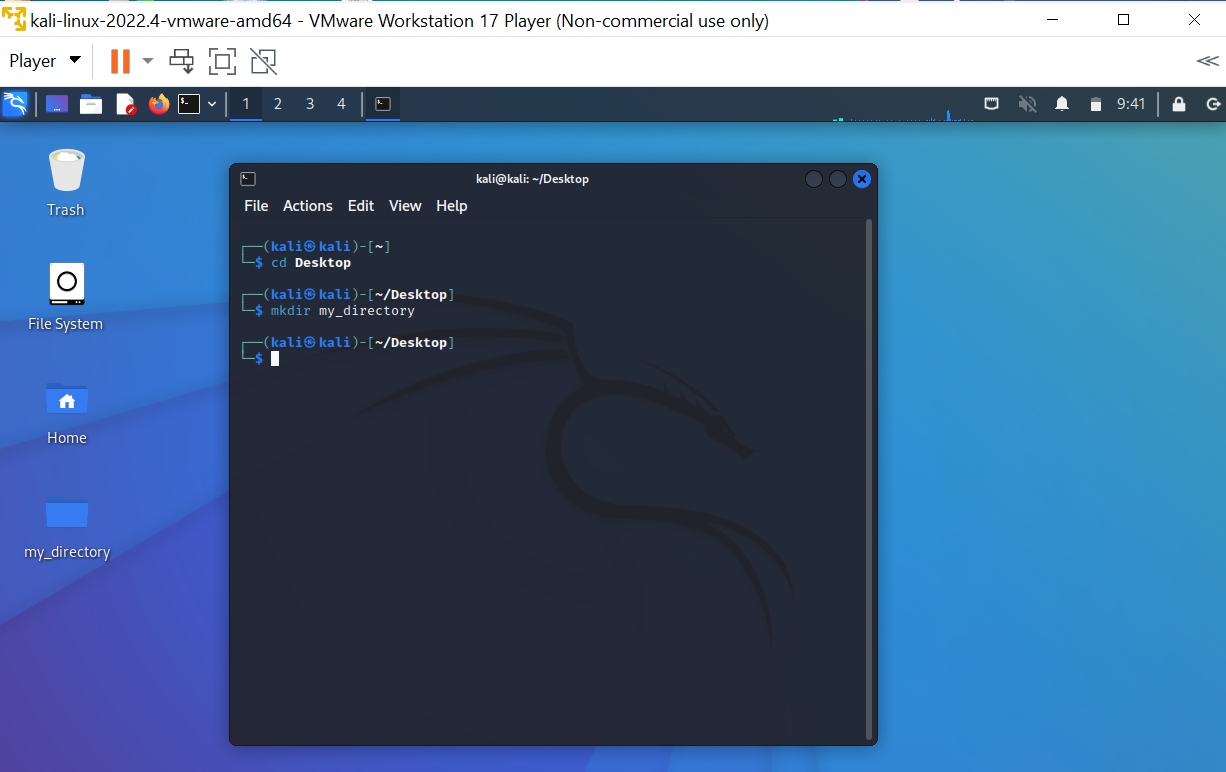
***NAME:POOVARASAN.A***

***REG.NO:20BCI0194***

**Task 1: File and Directory Manipulation**

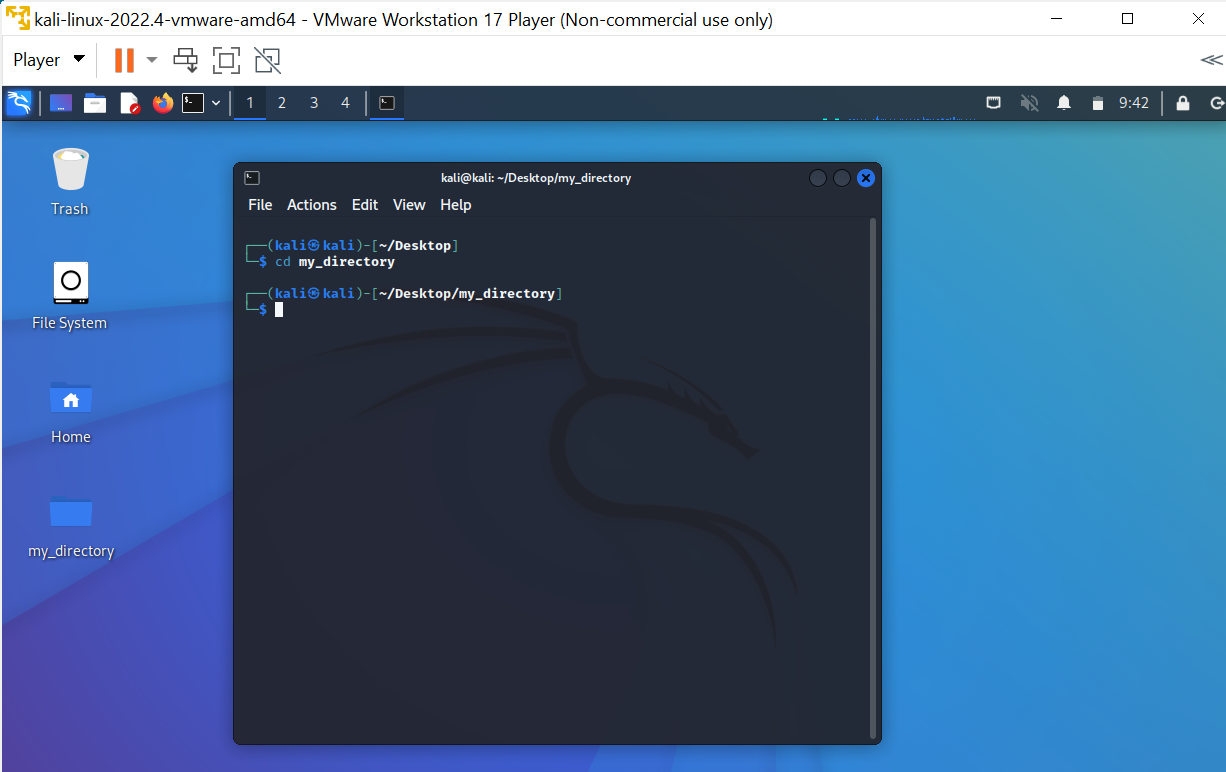
1. Create a directory called "my\_directory".

**Commands used:**mkdirmy\_directory



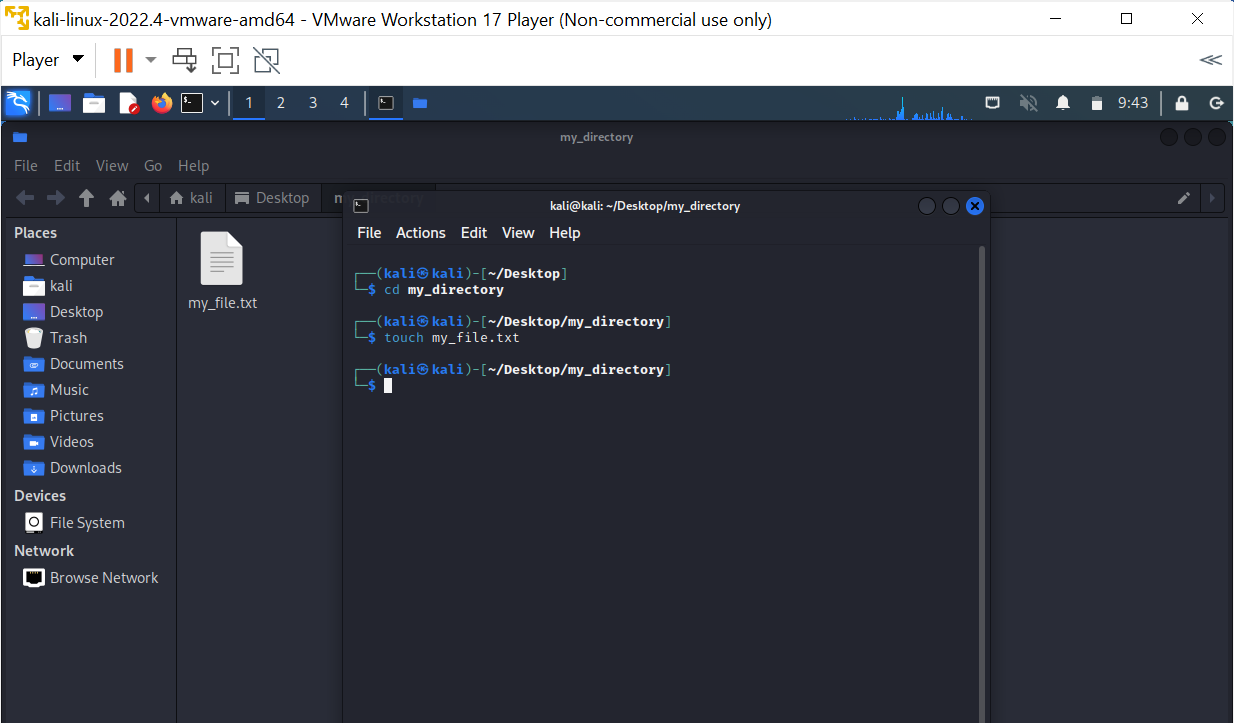
2. Navigate into the "my\_directory".

**Commands used:**cd my\_directory



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

**Commands used:**touch my\_file.txt



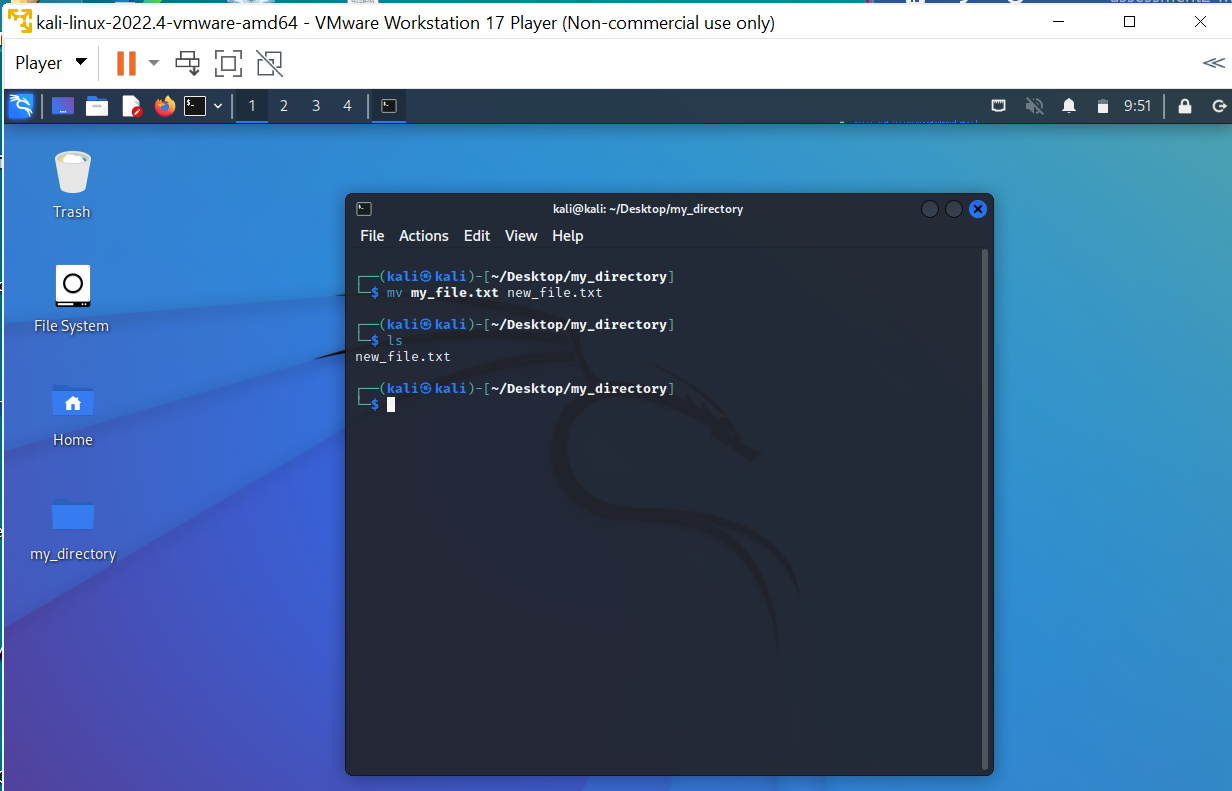
4. List all the files and directories in the current directory.

**Commands used:**ls



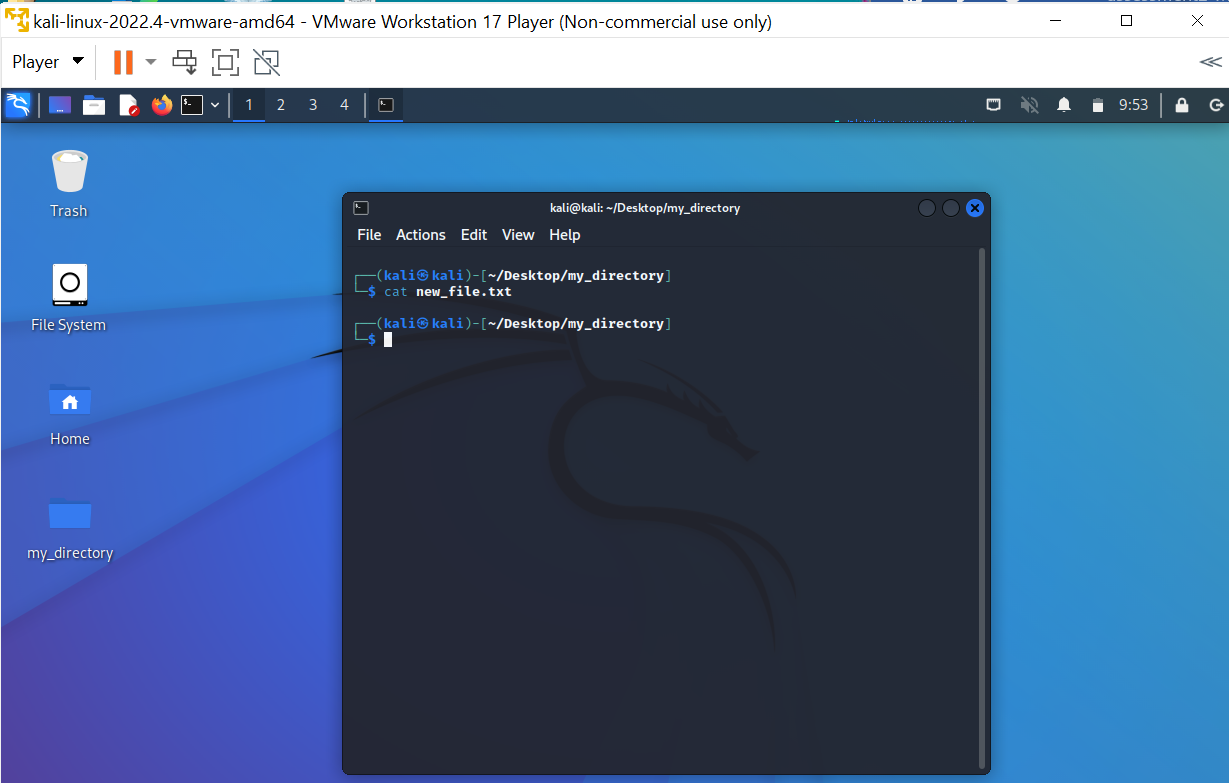
5. Rename "my\_file.txt" to "new\_file.txt".

**Commands used:**mv my\_file.txt new\_file.txt

****

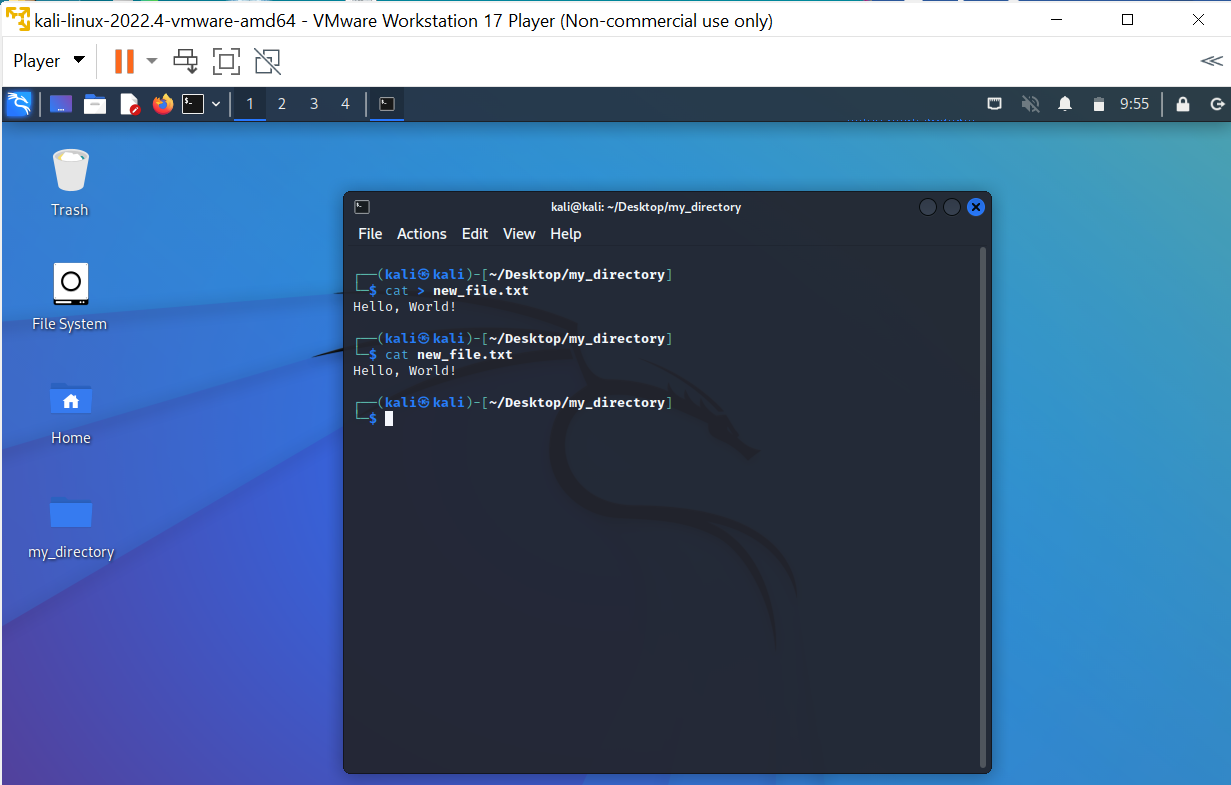
6. Display the content of "new\_file.txt" using a pager tool of your choice.

**Commands used:**cat new\_file.txt (the file is empty)



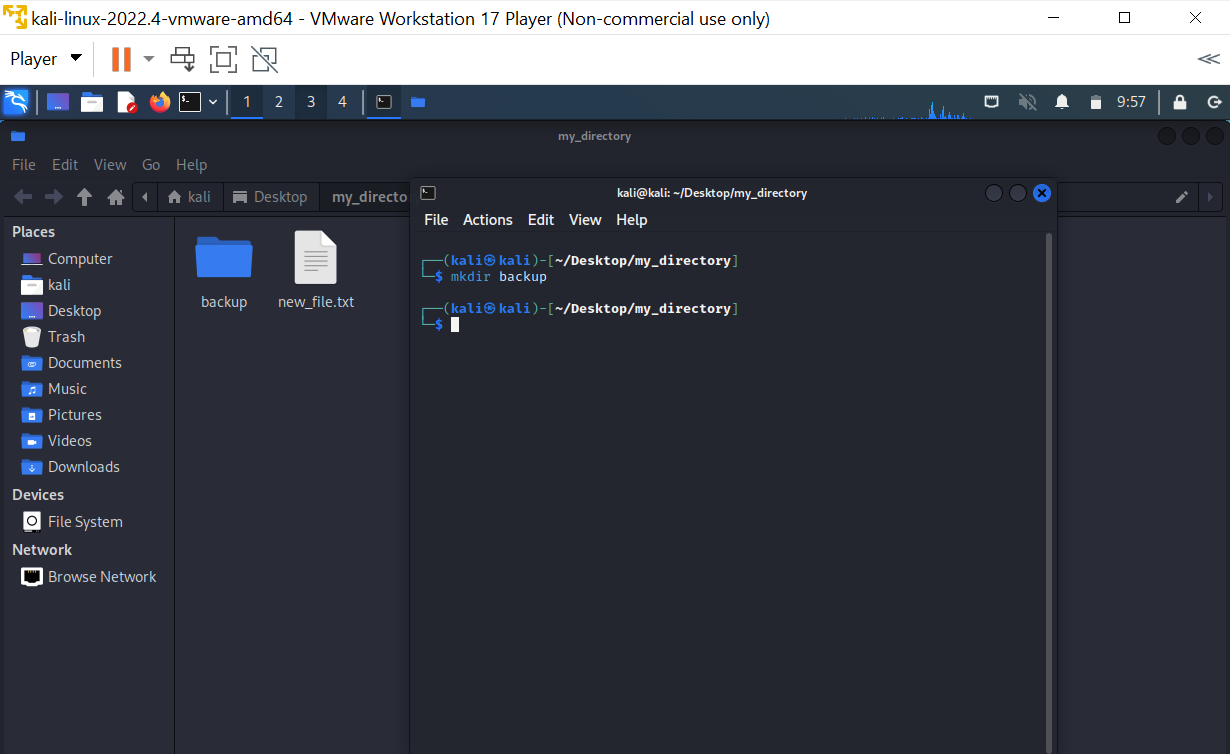
7. Append the text "Hello, World!" to "new\_file.txt".

**Commands used:**cat > new\_file.txt



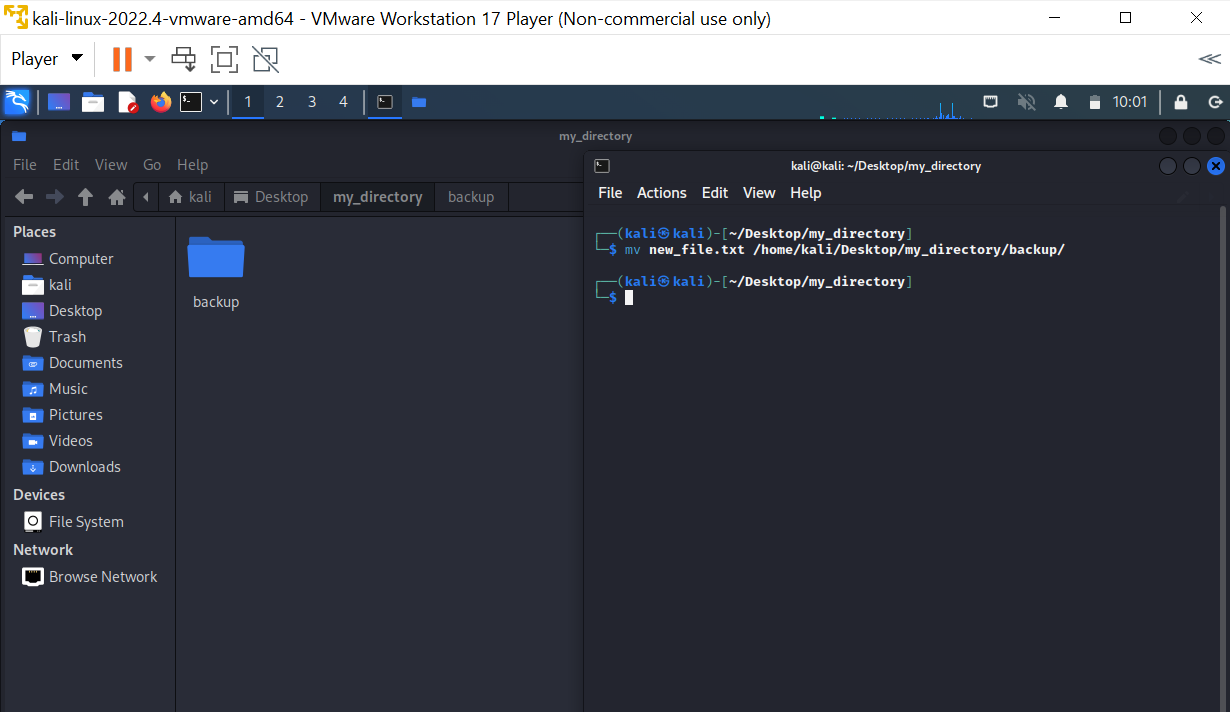
8. Create a new directory called "backup" within "my\_directory".

**Commands used:**mkdir backup



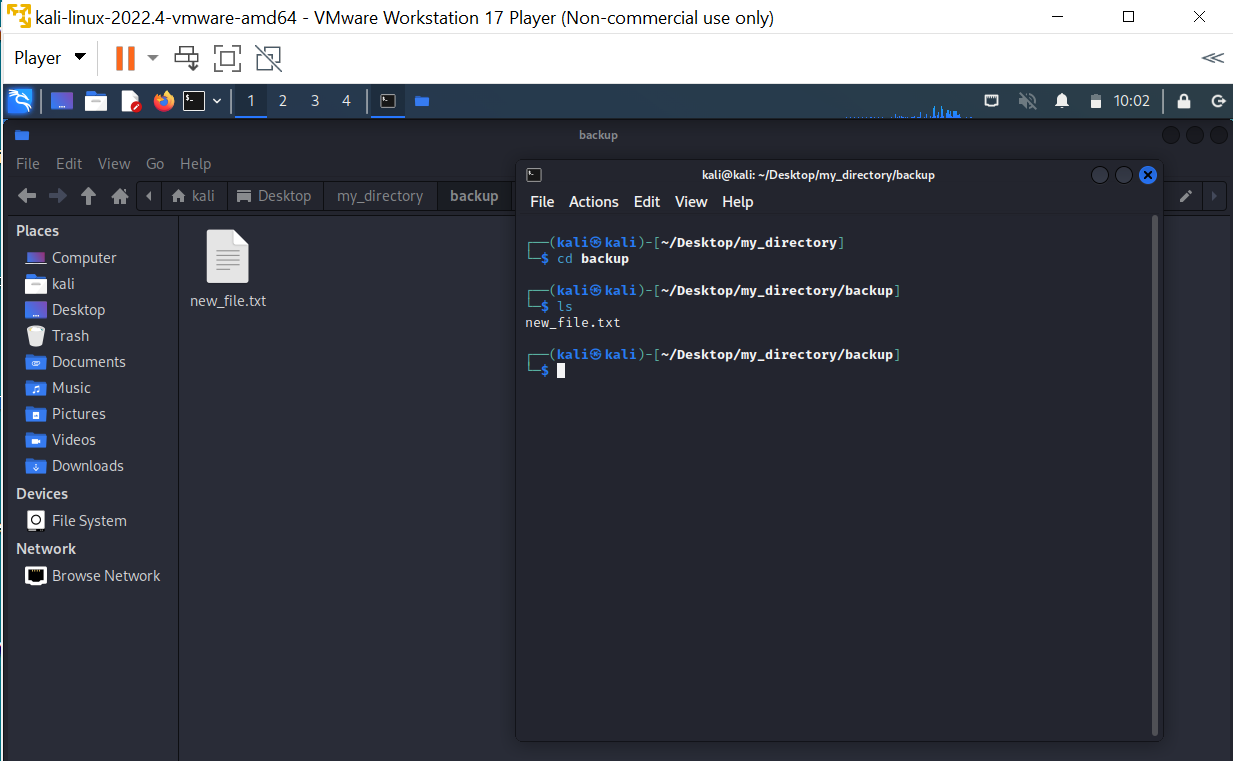
9. Move "new\_file.txt" to the "backup" directory.

**Commands used:**mv new\_file.txt /home/kali/Desktop/my\_directory/backup



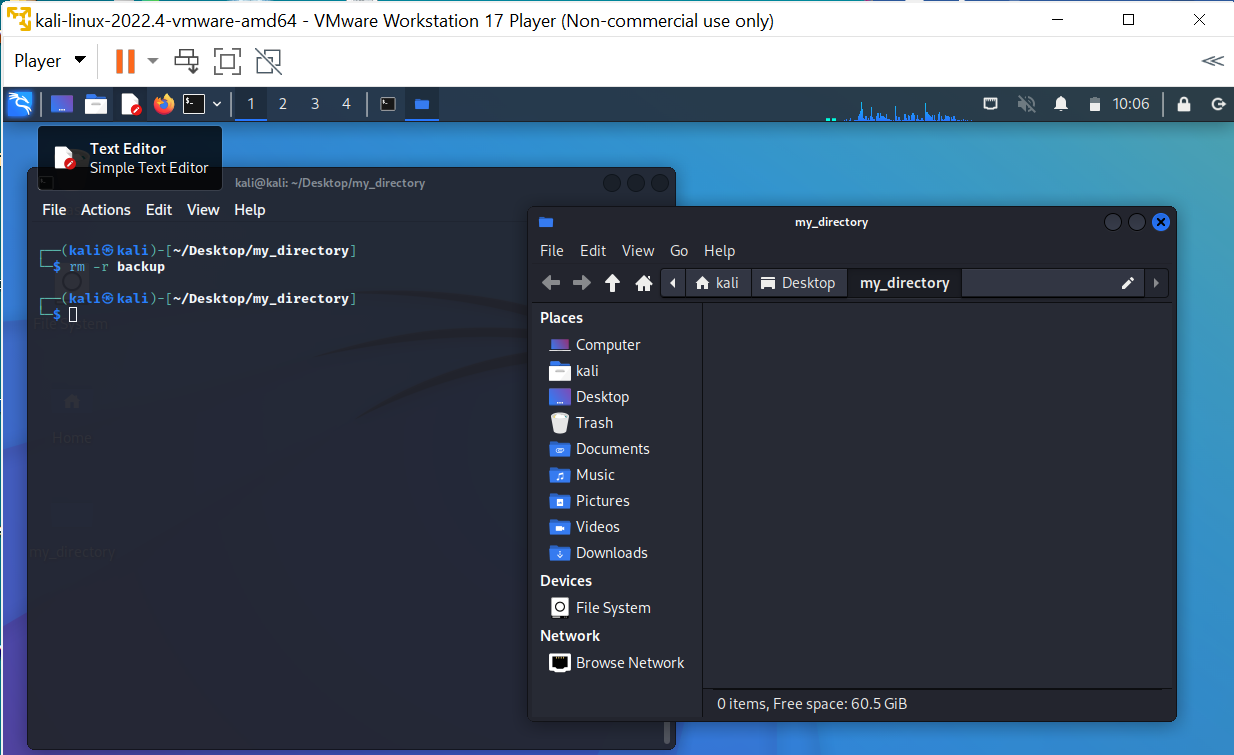
10. Verify that "new\_file.txt" is now located in the "backup" directory.

**Commands used:**ls



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

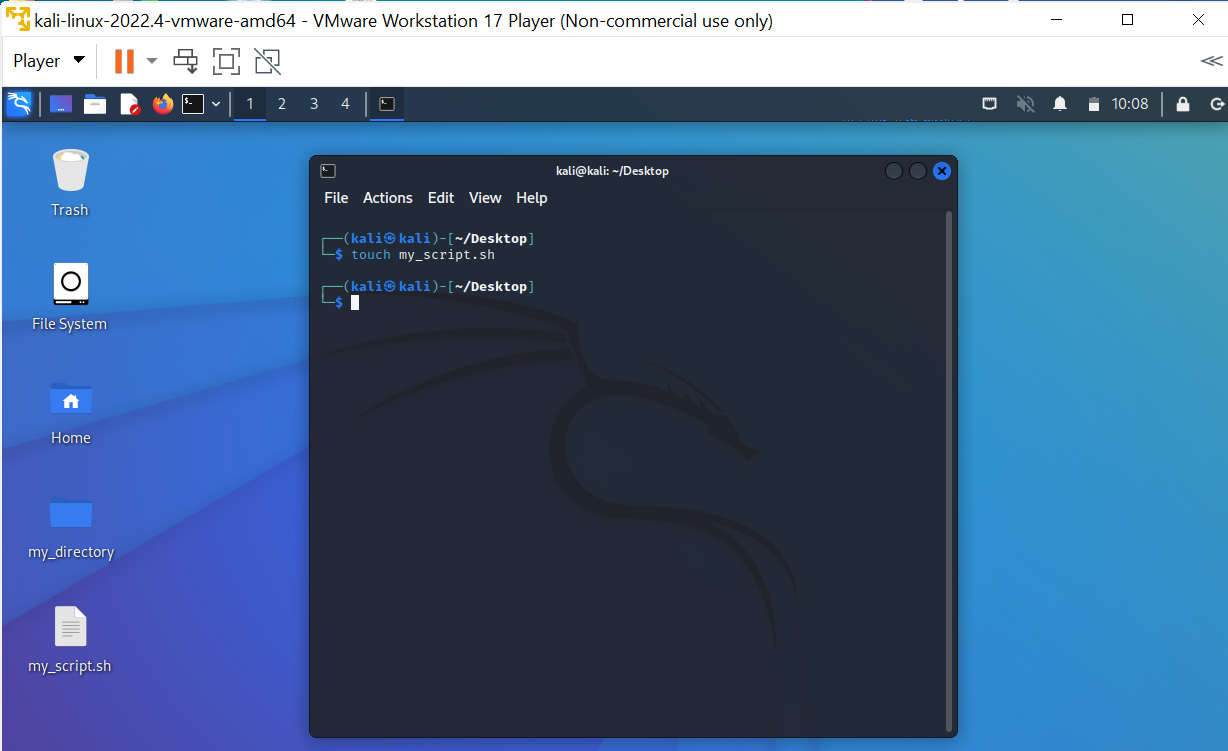
**Commands used:**rm -r backup



**Task 2: Permissions and Scripting**

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

**Commands used:**touch my\_script.sh



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

**#!/bin/bash**

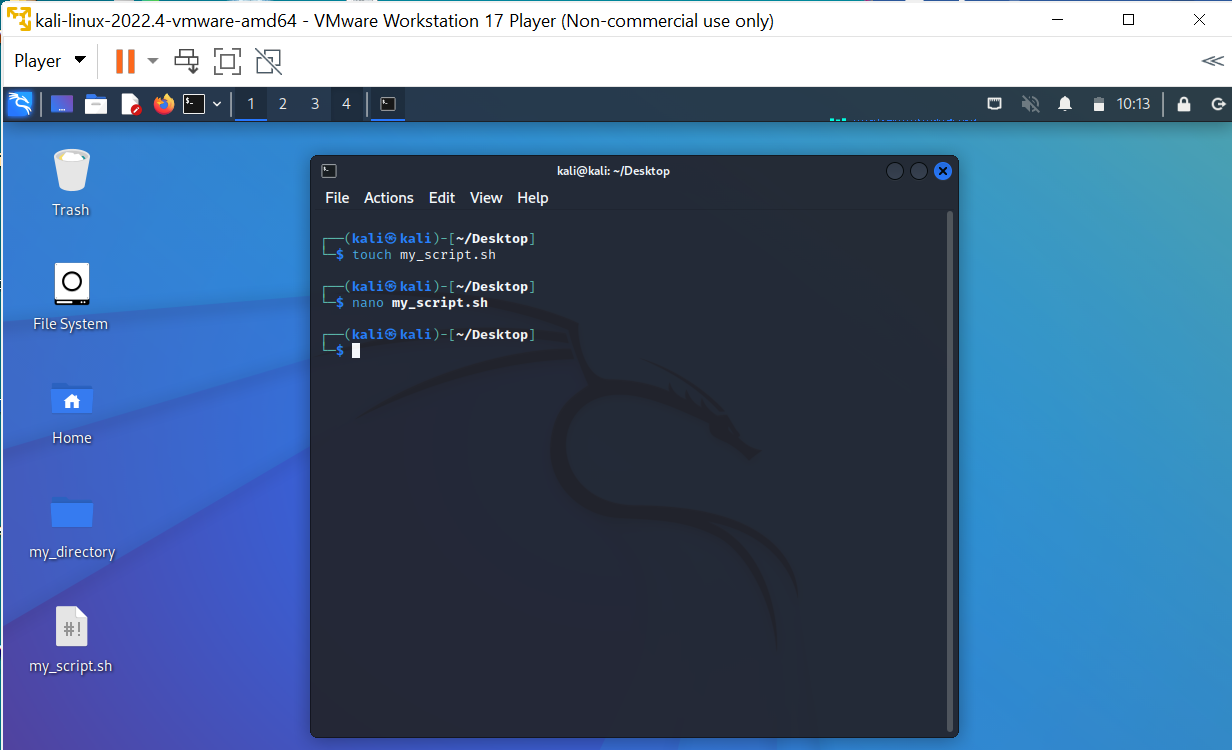
**echo "Welcome to my script!"**

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

**Save and exit the file.**

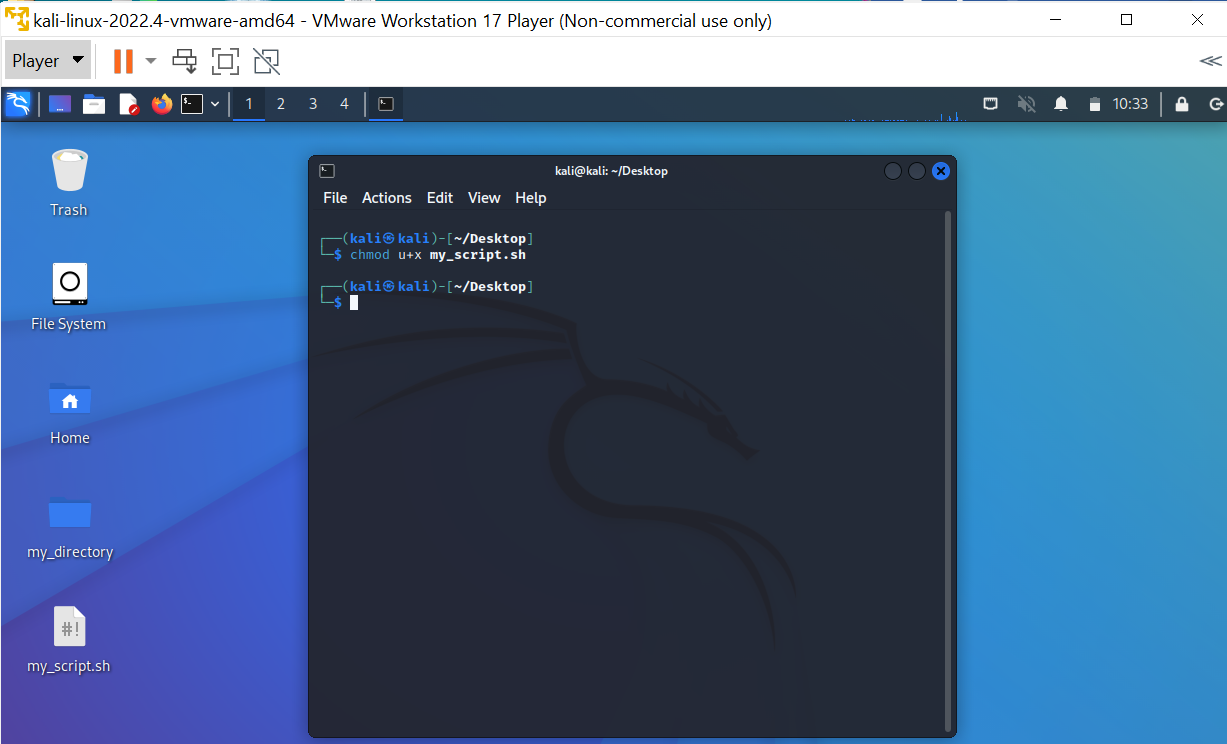
**Commands used:**nano my\_script.sh





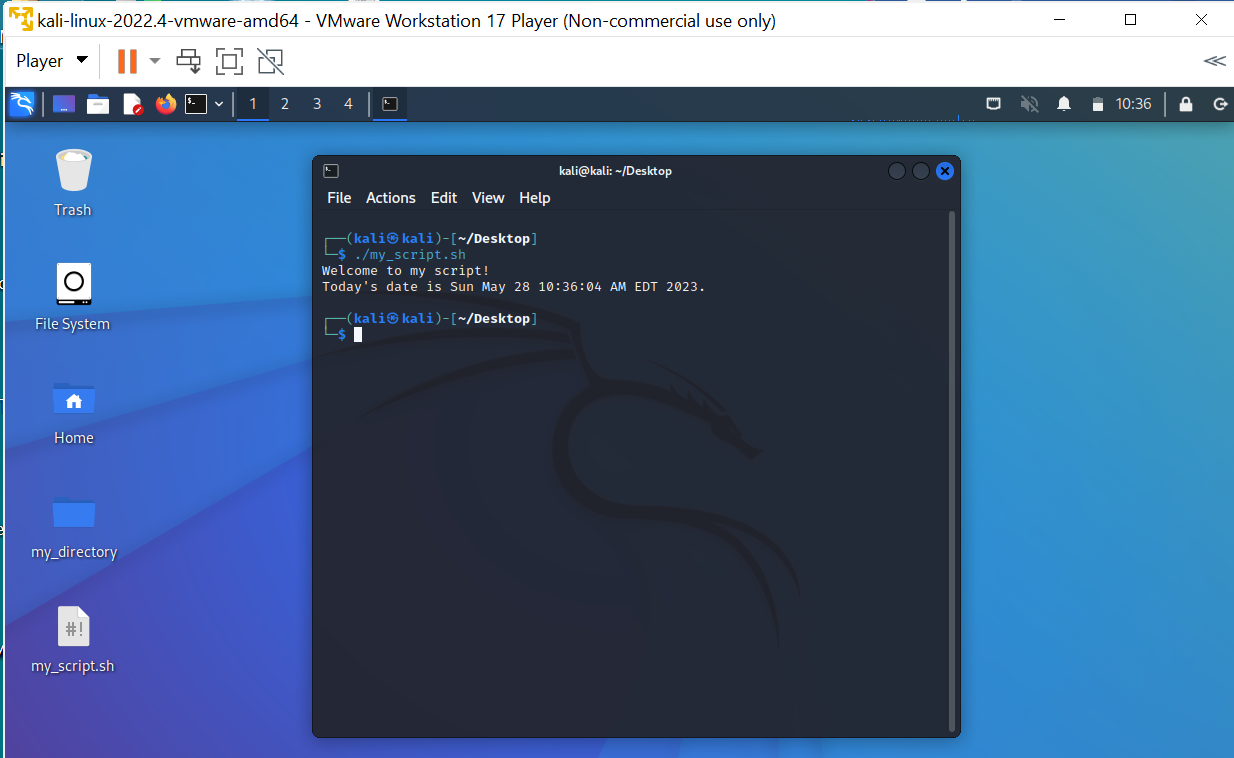
● Make "my\_script.sh" executable.

**Commands used:**chmodu+x my\_script.sh



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

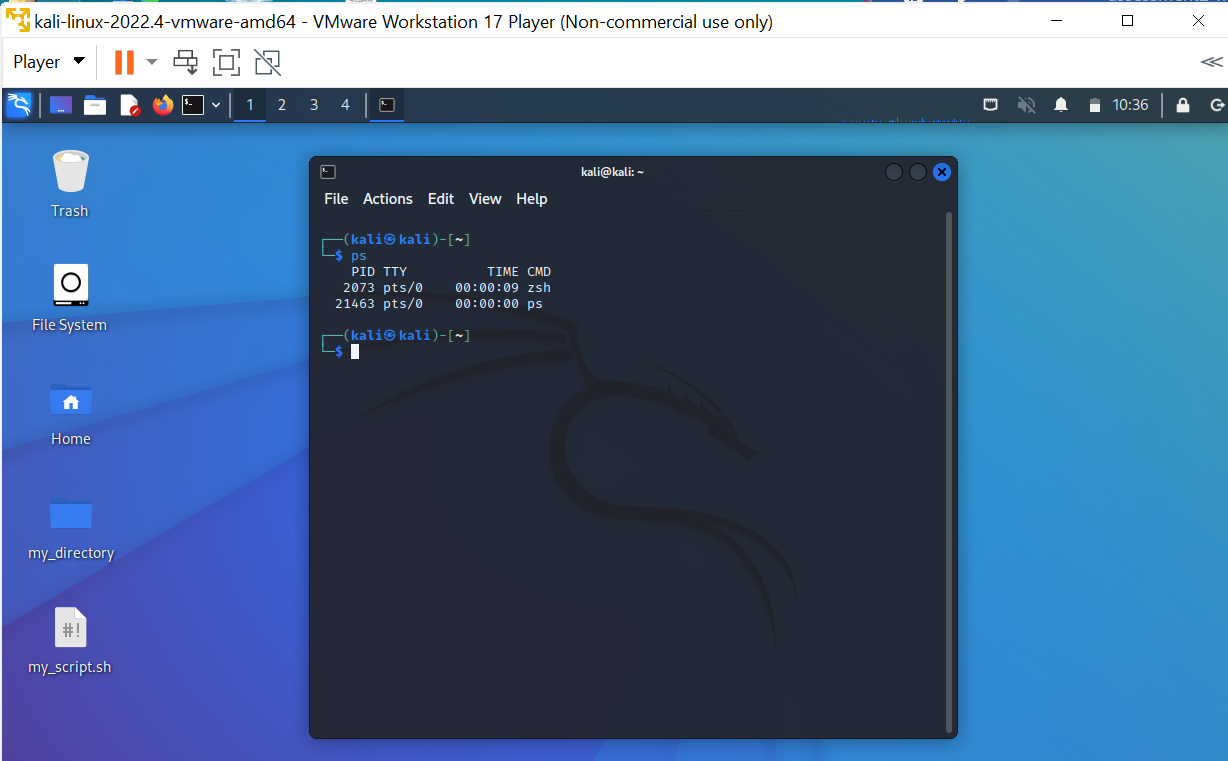
**Commands used:**./my\_script.sh



**Task 3: Command Execution and Pipelines**

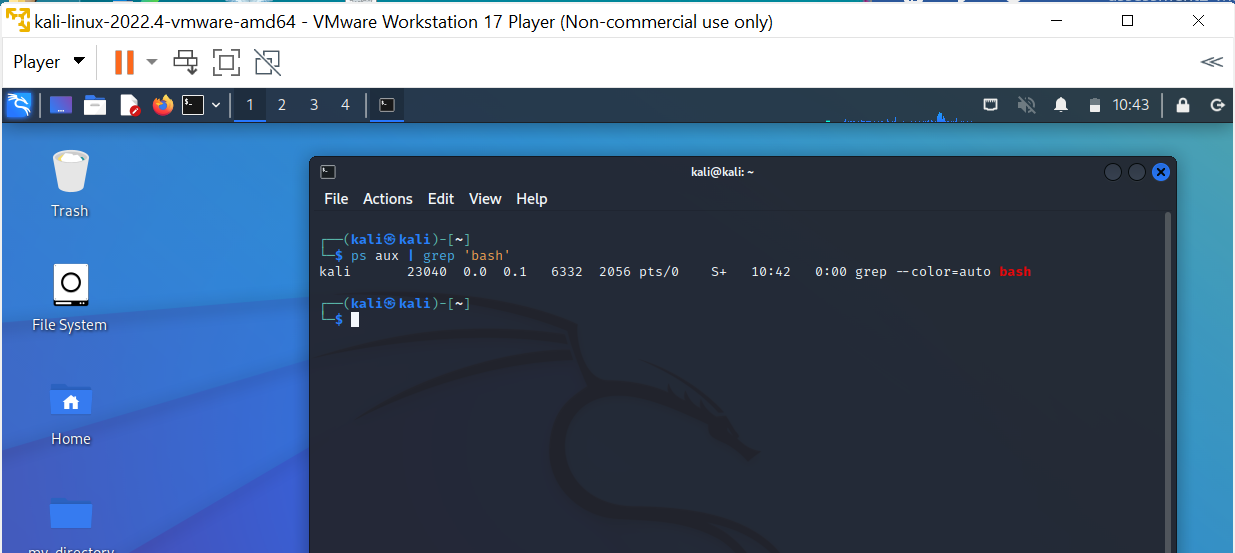
● List all the processes running on your system using the "ps" command.

**Commands used:**ps – this command lists the active processes and their PIDs



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

**Commands used:**ps aux | grep ‘bash’



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

**Commands used:**ps aux | grep ‘bash’ | wc

