CYBER SECURITY AND ETHICAL HACKING

WEEK -2 ASSIGNMENT

BASH SHELL BASICS

SUBMITED BY,

NAME : NITHIYASRI M

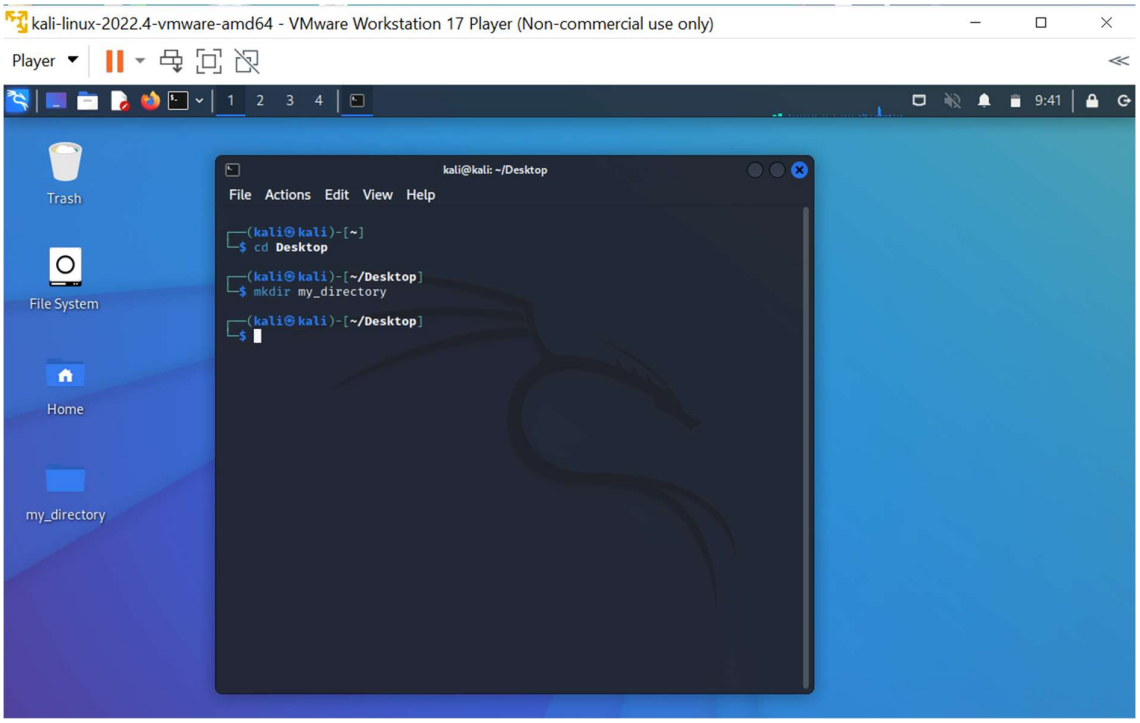
REG_NO: 20BCI0230

DATE : 28-05-2023

Task 1: File and Directory Manipulation

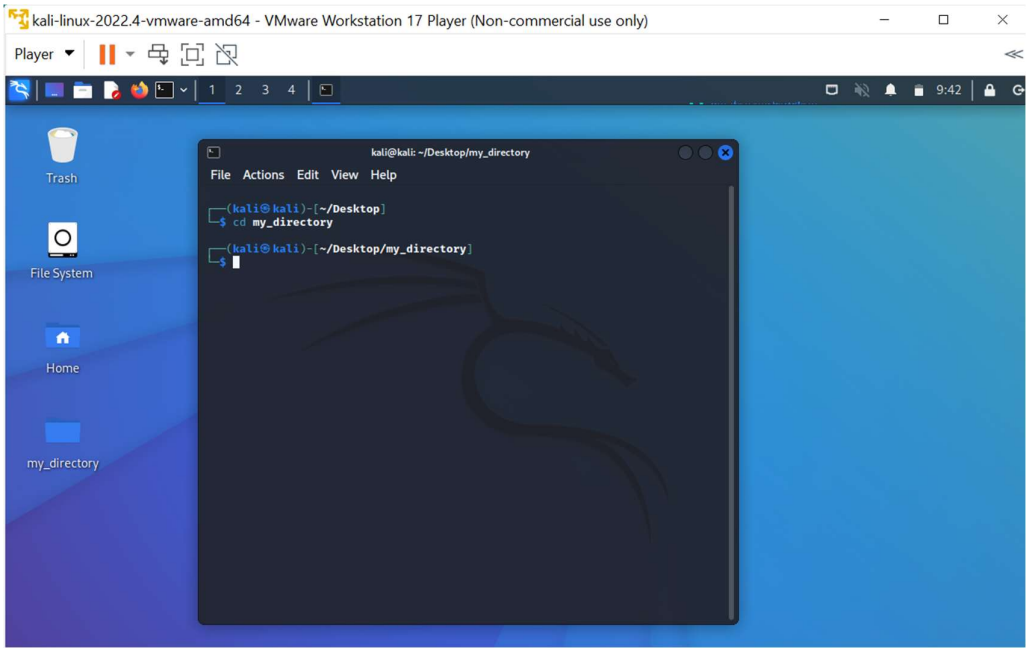
1. Create a directory called "my_directory".

- mkdir my_directory



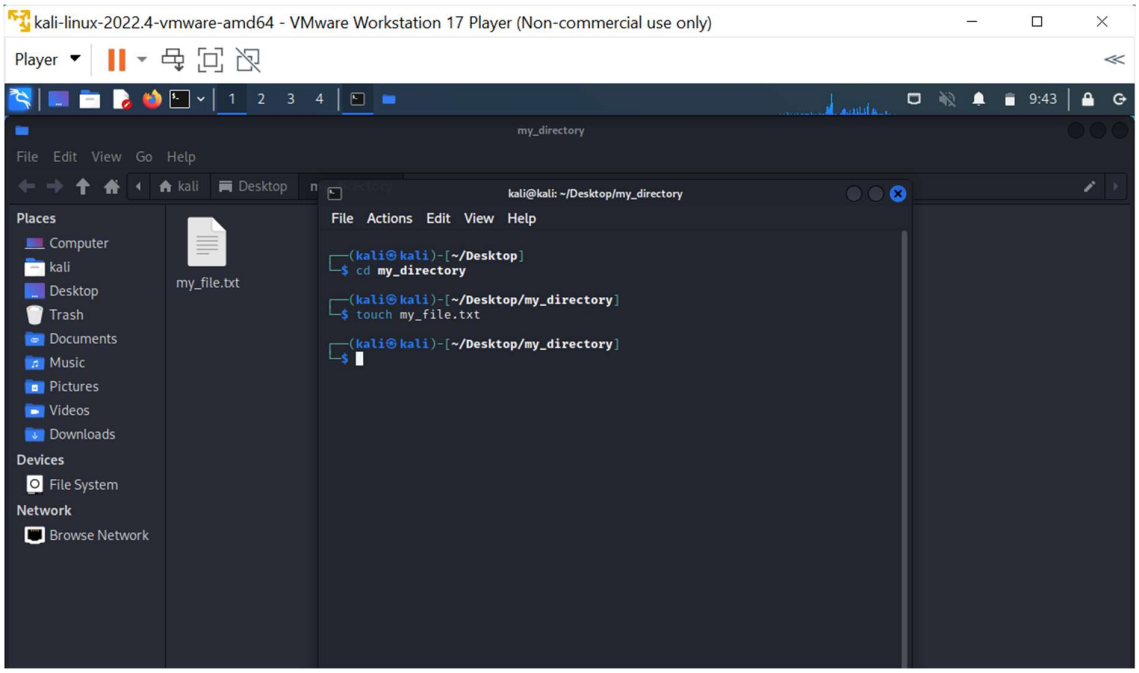
2. Navigate into the "my_directory".

- cd my_directory



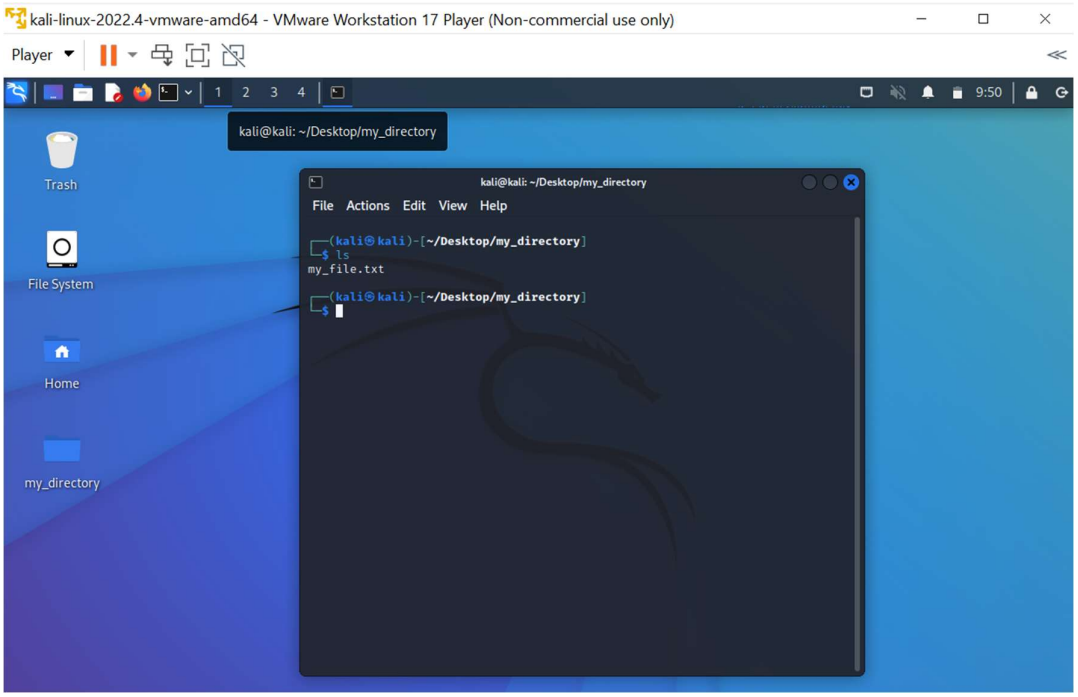
3. Create an empty file called "my_file.txt".

- touch my_file.txt



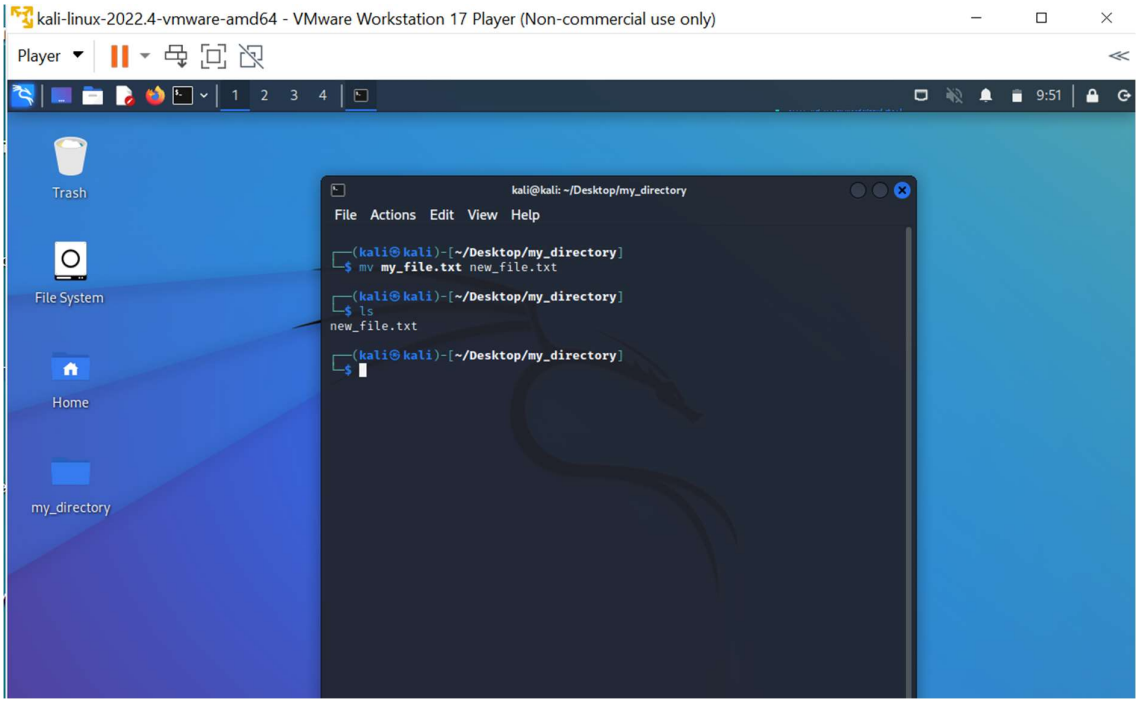
4. List all the files and directories in the current directory.

- ls



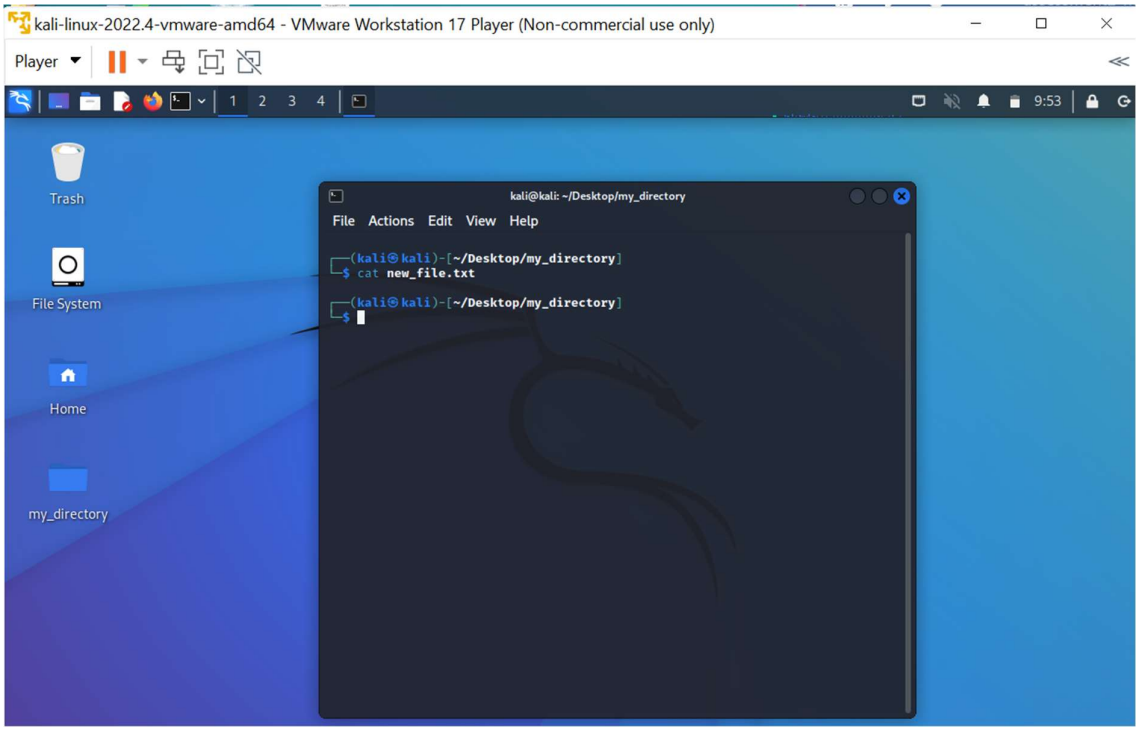
5. Rename "my_file.txt" to "new_file.txt".

- mv my_file.txt new_file.txt



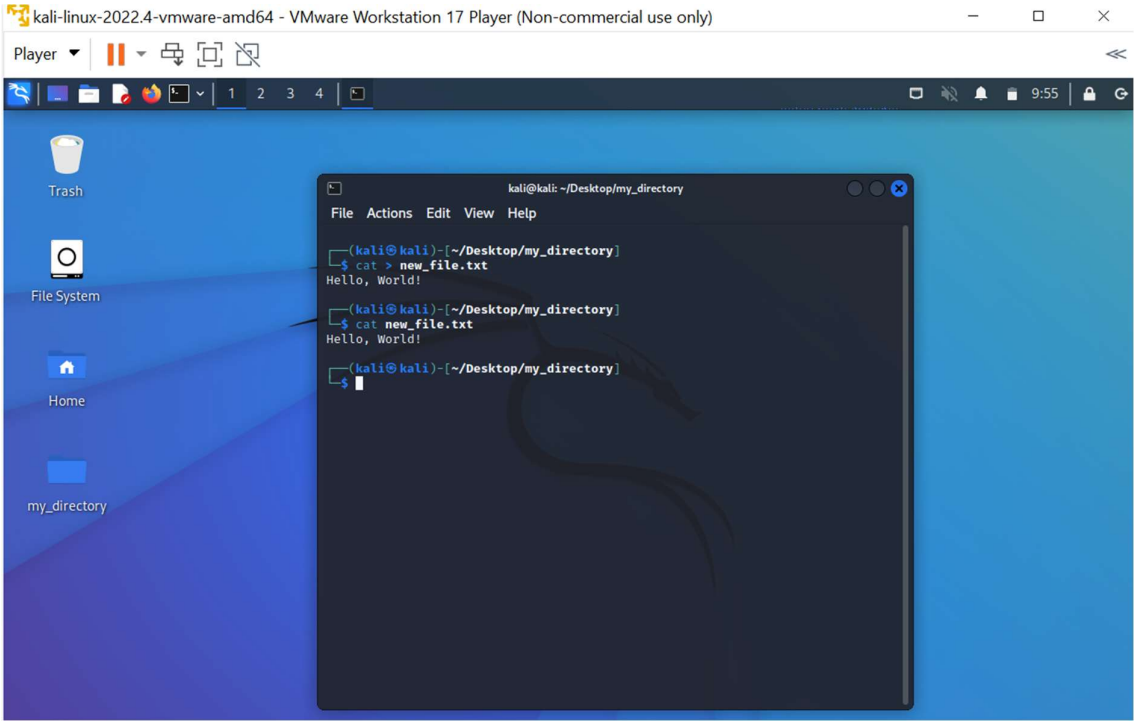
6. Display the content of "new_file.txt" using a pager tool of your choice.

- cat new_file.txt (the file is empty)



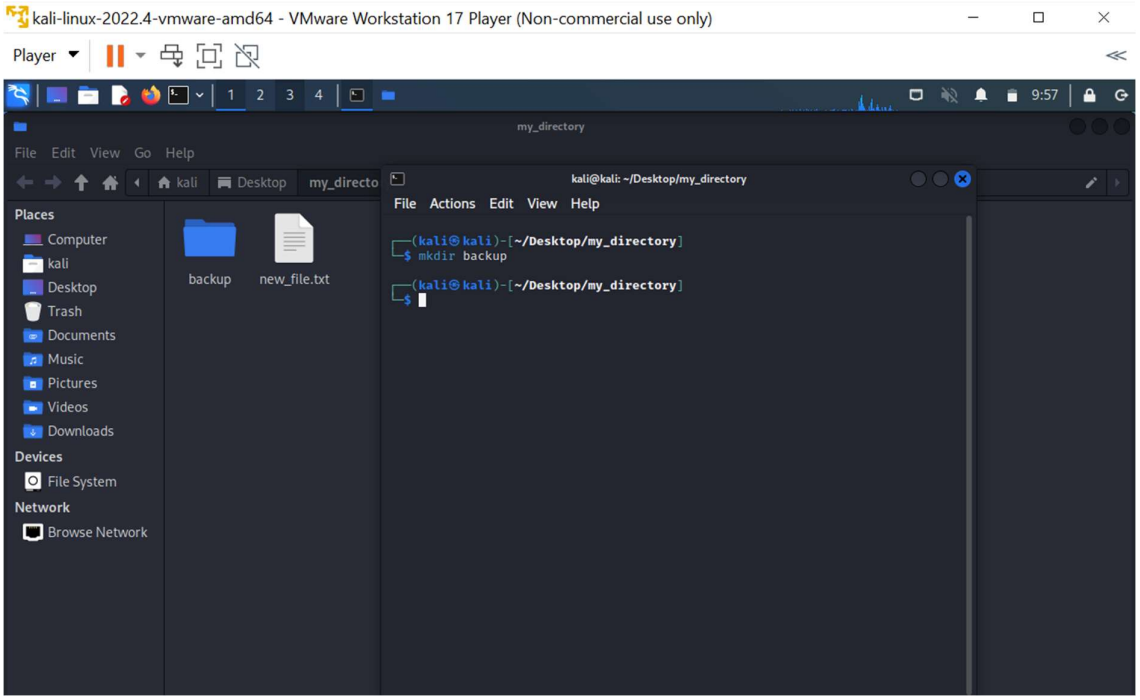
7. Append the text "Hello, World!" to "new_file.txt".

- cat > new_file.txt



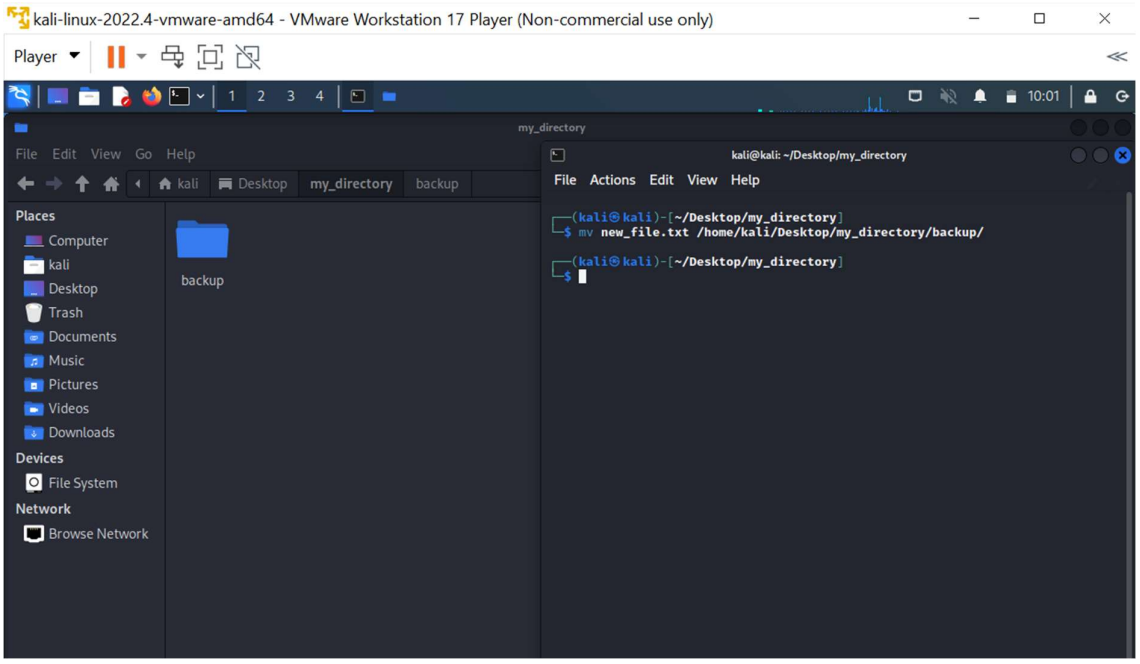
8. Create a new directory called "backup" within "my_directory".

- mkdir backup



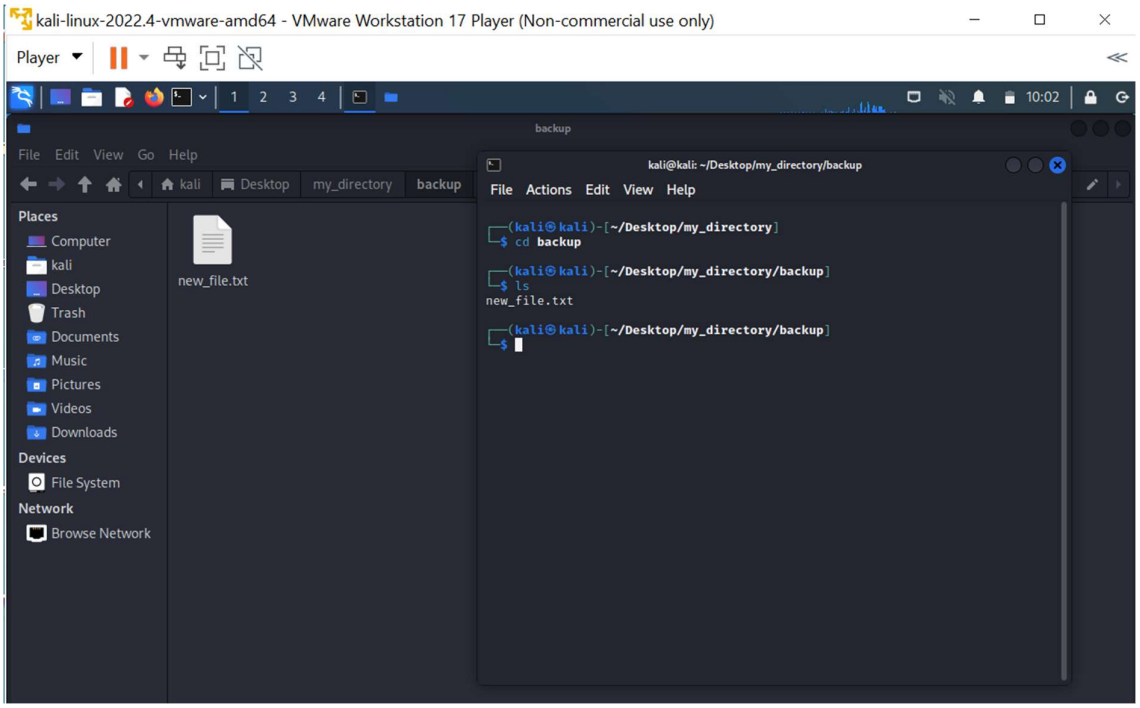
9. Move "new_file.txt" to the "backup" directory.

- mv new_file.txt /home/kali/Desktop/my_directory/backup



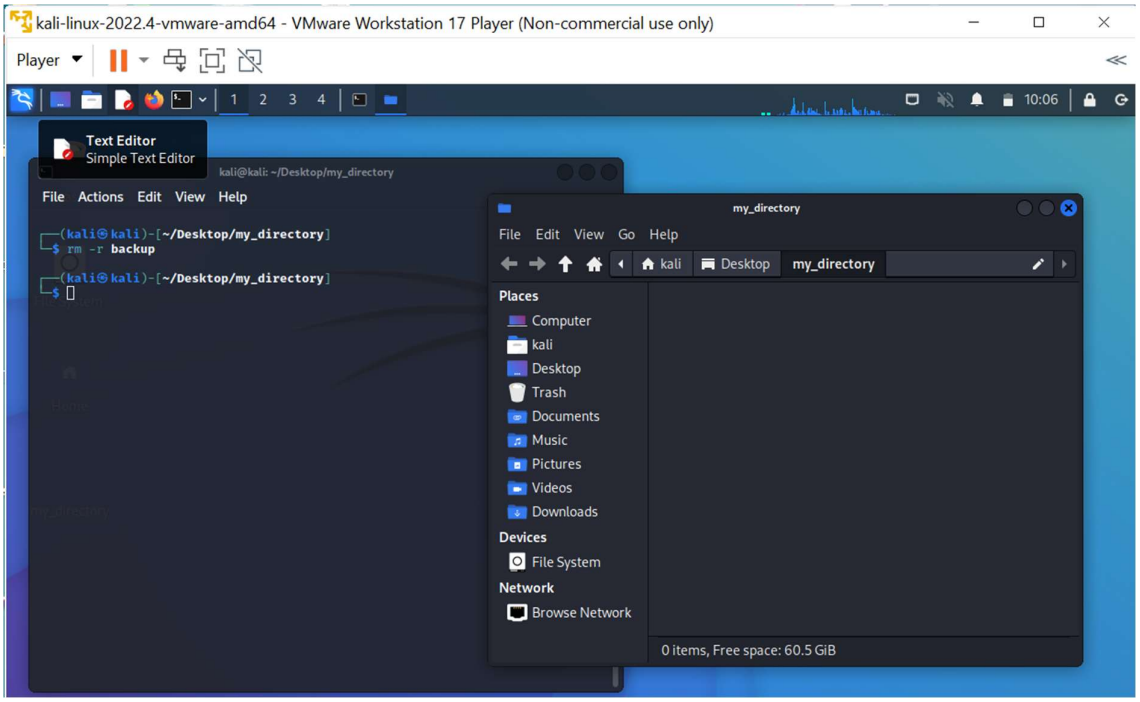
10. Verify that "new_file.txt" is now located in the "backup" directory.

- ls



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

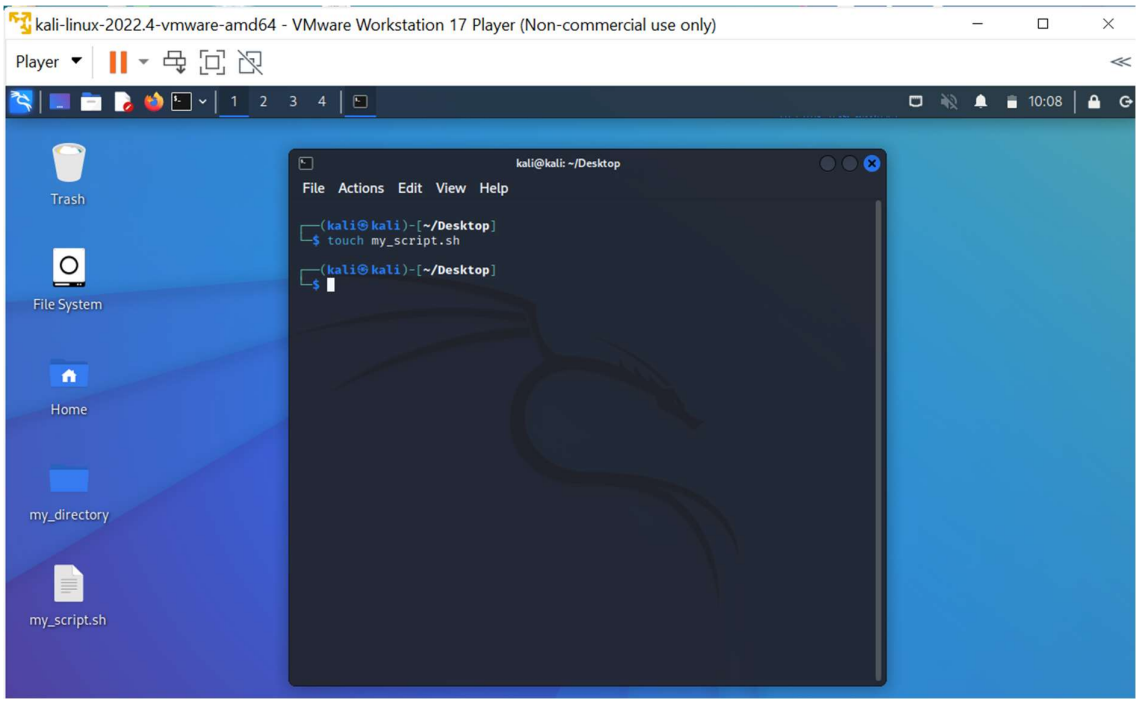
- rm -r backup



Task 2: Permissions and Scripting

1. Create a new file called "my_script.sh".

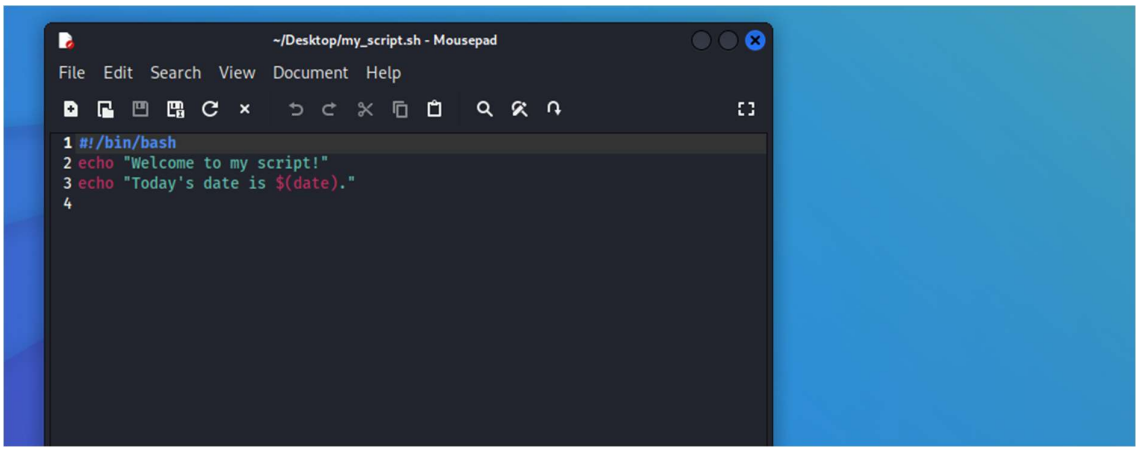
- touch my_script.sh

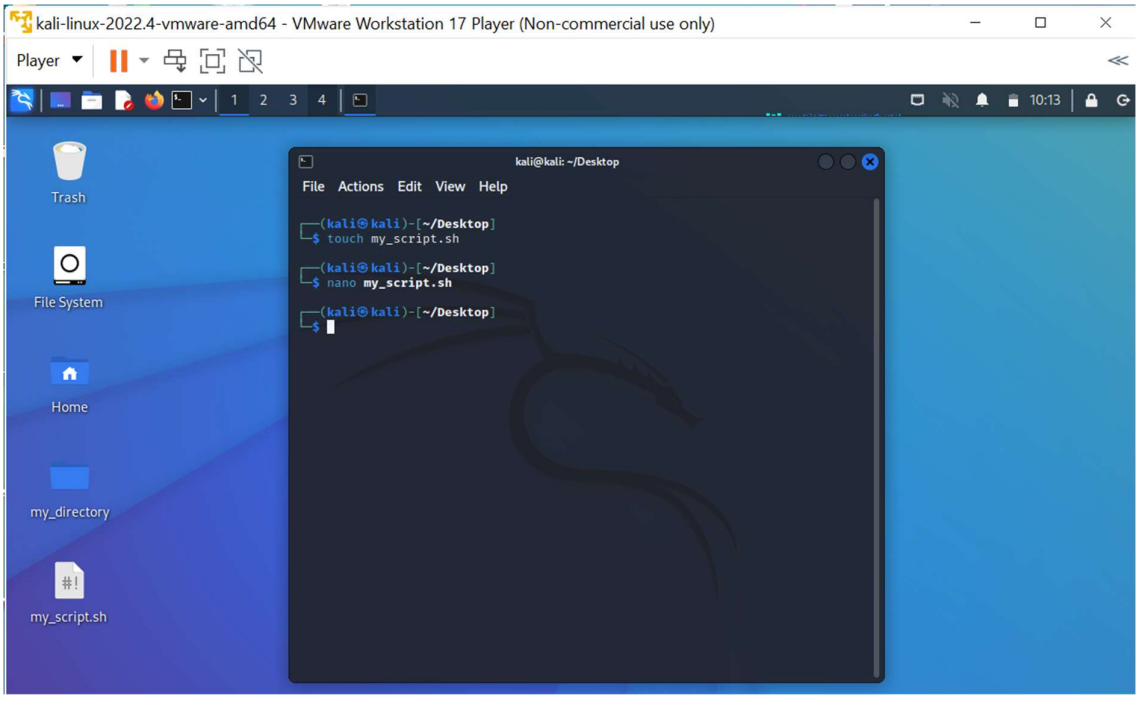


2. Edit "my_script.sh" using a text editor of your choice and add the following lines: bash

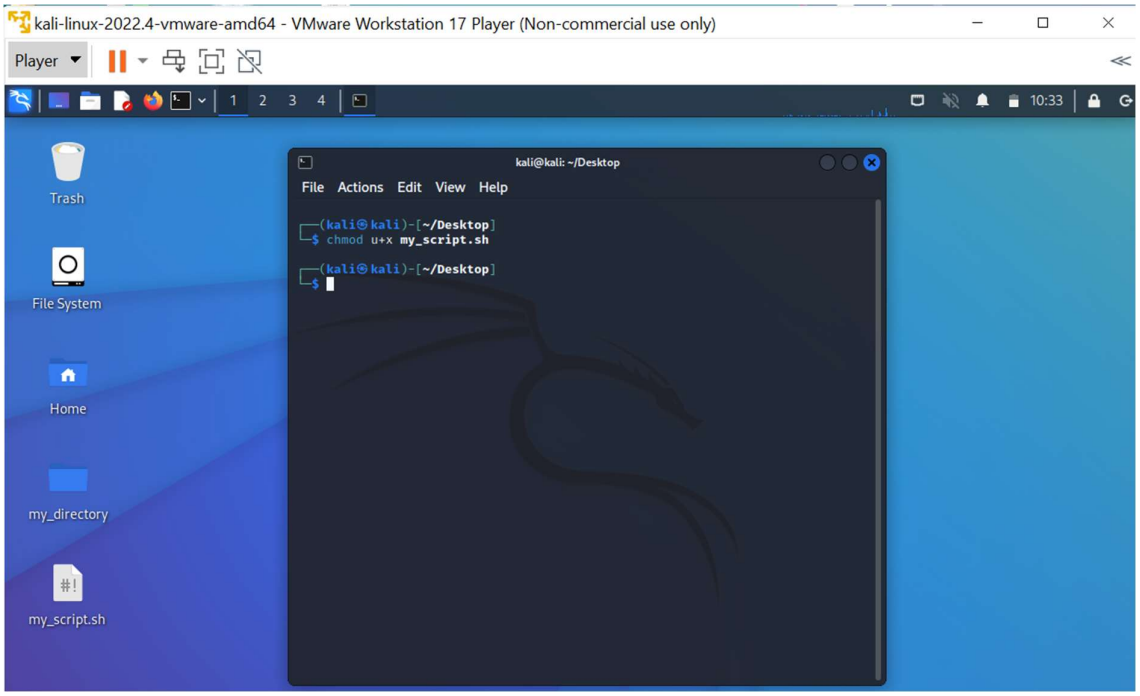
#!/bin/bash
echo "Welcome to my script!"
echo "Today's date is $(date)."
Save and exit the file.

- nano my_script.sh

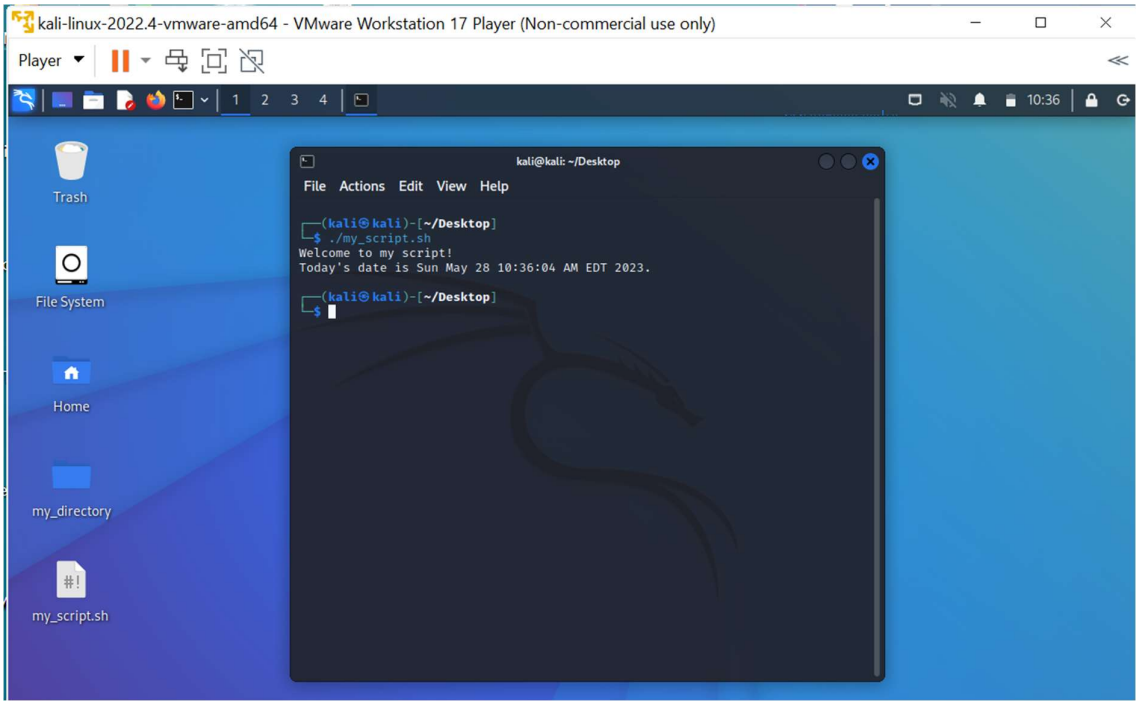




3. Make "my_script.sh" executable.
    - chmod u+x my_script.sh

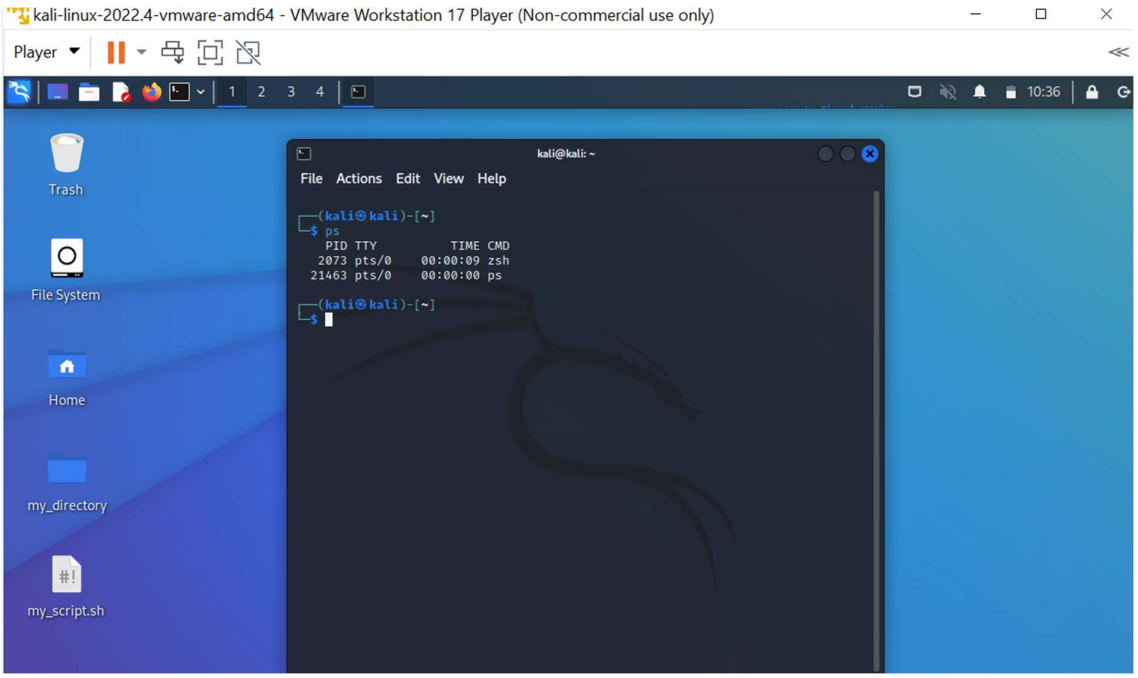


4. Run "my_script.sh" and verify that the output matches the expected result.
    - ./my_script.sh

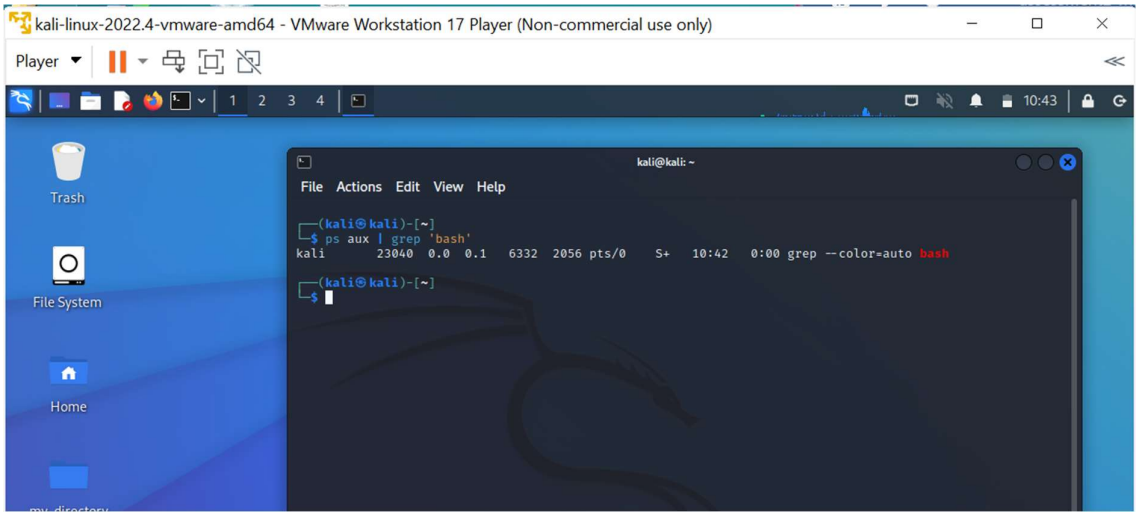


Task 3: Command Execution and Pipelines

1. List all the processes running on your system using the "ps" command.
    - ps – this command lists the active processes and their PIDs



2. Use the "grep" command to filter the processes list and display only the processes with "bash" in their name.
    - ps aux | grep ‘bash’



3. Use the "wc" command to count the number of lines in the filtered output.
    - ps aux | grep ‘bash’ | wc