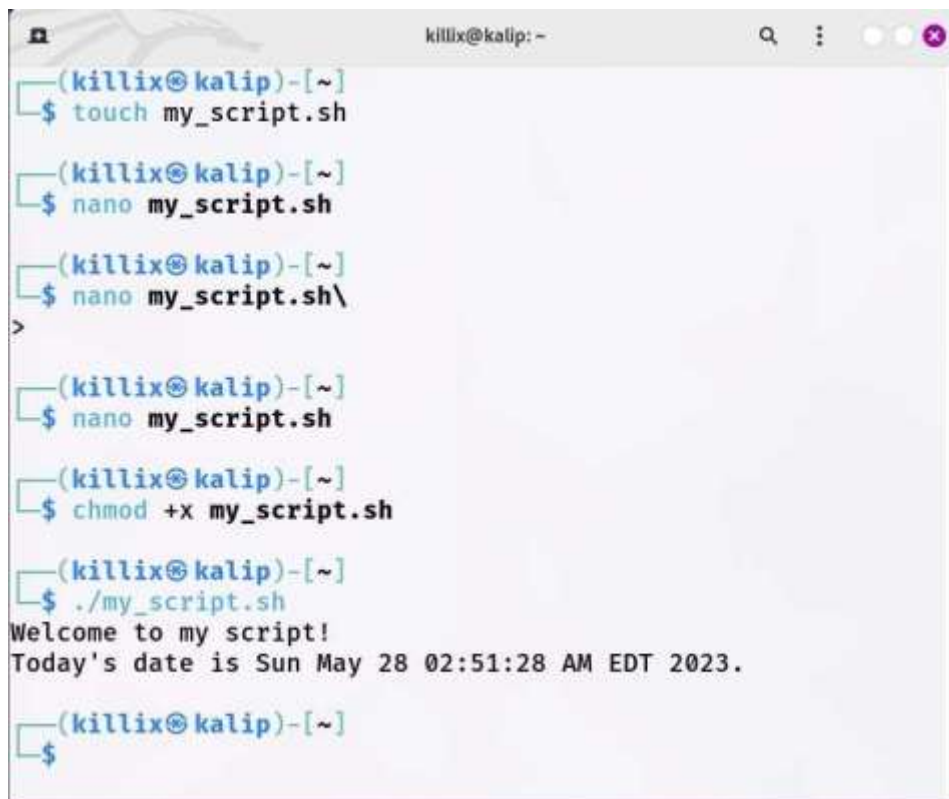Task 2: Permissions and Scripting

● Create a new file called "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.

● Make "my_script.sh" executable.
● Run "my_script.sh" and verify that the output matches the expected result.

```
┌──(killix㉿kalip)-[~]
└─$ touch my_script.sh

┌──(killix㉿kalip)-[~]
└─$ nano my_script.sh

┌──(killix㉿kalip)-[~]
└─$ nano my_script.sh\
>

┌──(killix㉿kalip)-[~]
└─$ nano my_script.sh

┌──(killix㉿kalip)-[~]
└─$ chmod +x my_script.sh

┌──(killix㉿kalip)-[~]
└─$ ./my_script.sh
Welcome to my script!
Today's date is Sun May 28 02:51:28 AM EDT 2023.

┌──(killix㉿kalip)-[~]
└─$
```

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

Task 3: Command Execution and Pipelines

- List all the processes running on your system using the "ps" command.
- Use the "grep" command to filter the processes list and display only the processes with "bash" in their name.
- Use the "wc" command to count the number of lines in the filtered output.


```
┌──(killix⊛kalip)-[~]
└─$ ps aux | grep bash
killix      18348  0.0  0.0   6332  2144 pts/1     S+   03:00   0:00 grep --color=auto bash

┌──(killix⊛kalip)-[~]
└─$ ps aux | grep bash | wc -l
1
```