

Cyber Security and Ethical Hacking

ASSESSMENT – 2 (27,28 MAY 2023)

ANNGELA ROY - 20MIS0186

TASK1. File and directory manipulation

1. Create a directory called "my_directory".

```
(anngela@Anngela)-[~]
$ mkdir my_directory

(anngela@Anngela)-[~]
$ ls
angela      Desktop    latest.zip  passwords  Templates
anngel      Documents  Music       passwords.txt.gz user.txt
archive1.tar Downloads  my_directory Pictures     Videos
csec        hydra.restore passwd      Public      wordpress
```

2. Navigate into the "my_directory".

```
(anngela@Anngela)-[~]
$ cd my_directory

(anngela@Anngela)-[~/my_directory]
$
```

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

```
(anngela@Anngela)-[~]
$ touch my_file.txt

(anngela@Anngela)-[~]
$ ls
angela      Desktop    latest.zip  passwd      Public      wordpress
anngel      Documents  Music       passwords   Templates
archive1.tar Downloads  my_directory passwords.txt.gz user.txt
csec        hydra.restore my_file.txt Pictures     Videos
```

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

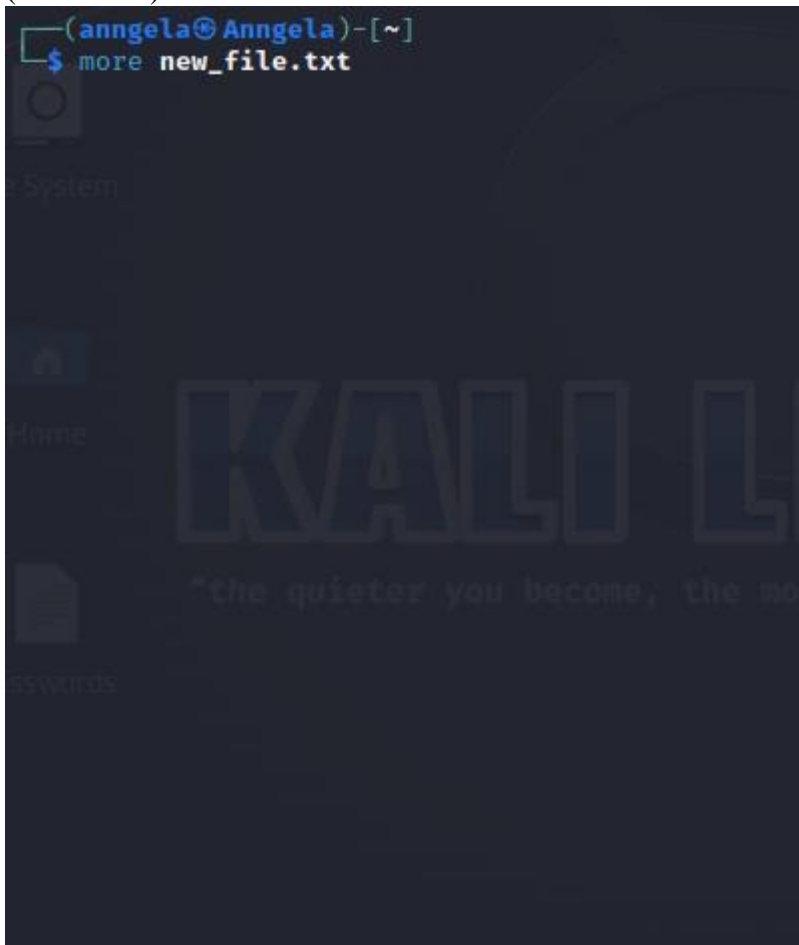
```
(annabela@Anngela)-[~]
$ ls -la
total 24236
drwx----- 21 annabela annabela      4096 Jun 28 22:08 .
drwxr-xr-x  3 root      root          4096 Jun 25 22:51 ..
drwxr-xr-x  3 annabela annabela      4096 Jun 28 22:01 annela
drwxr-xr-x  2 annabela annabela      4096 Jun 28 19:57 annel
-rw-r--r--  1 annabela annabela    20480 Jun 28 21:04 archive1.tar
-rw-r--r--  1 annabela annabela       220 Jun 25 22:51 .bash_logout
-rw-r--r--  1 annabela annabela     5551 Jun 25 22:51 .bashrc
-rw-r--r--  1 annabela annabela     3526 Jun 25 22:51 .bashrc.original
drwxr-xr-x  9 annabela annabela      4096 Jun 25 23:30 .cache
drwxr-xr-x 17 annabela annabela      4096 Jun 28 20:32 .config
drwxr-xr-x  2 annabela annabela      4096 Jun 28 19:57 csec
drwxr-xr-x  2 annabela annabela      4096 Jun 26 01:05 Desktop
-rw-r--r--  1 annabela annabela        35 Jun 26 01:50 .dmrc
drwxr-xr-x  2 annabela annabela      4096 Jun 26 01:06 Documents
drwxr-xr-x  2 annabela annabela      4096 Jun 25 23:14 Downloads
-rw-r--r--  1 annabela annabela    11759 Jun 25 22:51 .face
lrwxrwxrwx  1 annabela annabela         5 Jun 25 22:51 .face.icon → .face
drwx----- 3 annabela annabela      4096 Jun 25 23:10 .gnupg
-rw-r--r--  1 annabela annabela    26843 Jun 26 01:29 hydra.restore
-rw-----  1 annabela annabela         0 Jun 25 23:10 .ICEauthority
drwxr-xr-x  3 annabela annabela      4096 Jun 25 22:51 .java
-rw-r--r--  1 annabela annabela  24498824 May 20 10:01 latest.zip
-rw-----  1 annabela annabela         20 Jun 28 20:55 .lessht
drwxr-xr-x  4 annabela annabela      4096 Jun 25 23:10 .local
drwx----- 4 annabela annabela      4096 Jun 25 23:13 .mozilla
drwxr-xr-x  2 annabela annabela      4096 Jun 25 23:10 Music
drwxr-xr-x  2 annabela annabela      4096 Jun 28 22:08 my_directory
-rw-r--r--  1 annabela annabela         0 Jun 28 22:08 my_file.txt
-rw-r--r--  1 annabela annabela     3140 Jun 28 20:28 passwd
-rw-r--r--  1 annabela annabela    14519 Jun 26 01:04 passwords
```

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

```
(annabela@Anngela)-[~]
$ mv my_file.txt new_file.txt

(annabela@Anngela)-[~]
$ ls
annela      Desktop      latest.zip   passwd       Public       wordpress
annel       Documents    Music        passwords    Templates
archive1.tar Downloads     my_directory passwords.txt.gz user.txt
csec        hydra.restore new_file.txt Pictures      Videos
```

6. Display the content of "new_file.txt" using a pager tool of your choice.
(no content)



```
(anngela@Anngela)-[~]  
$ more new_file.txt
```

The image shows a terminal window with a dark background. The prompt is `(anngela@Anngela)-[~]`. The command `$ more new_file.txt` has been entered. The terminal output is empty, indicating the file has no content. In the background, there is a faint Kali Linux logo and the text "the quieter you become, the more you are able to hear".

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

```
(anngela@Anngela)-[~]  
$ echo 'Hello, World!' >> new_file.txt  
  
(anngela@Anngela)-[~]  
$ more new_file.txt  
Hello, World!
```

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

```
(anngela@Anngela)-[~/my_directory]  
$ mkdir backup  
  
(anngela@Anngela)-[~/my_directory]  
$ ls  
backup my_file.txt
```

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

```
(anngela@Anngela)-[~]  
$ mv new_file.txt backup  
  
(anngela@Anngela)-[~]  
$
```

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

```
(anngela@Anngela)-[~]  
$ cd my_directory  
  
(anngela@Anngela)-[~/my_directory]  
$ cd backup  
  
(anngela@Anngela)-[~/my_directory/backup]  
$ ls  
new_file.txt
```

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

```
(anngela@Anngela)-[~/my_directory]  
$ ls  
backup  
  
(anngela@Anngela)-[~/my_directory]  
$ rm -rf backup  
  
(anngela@Anngela)-[~/my_directory]  
$ ls
```

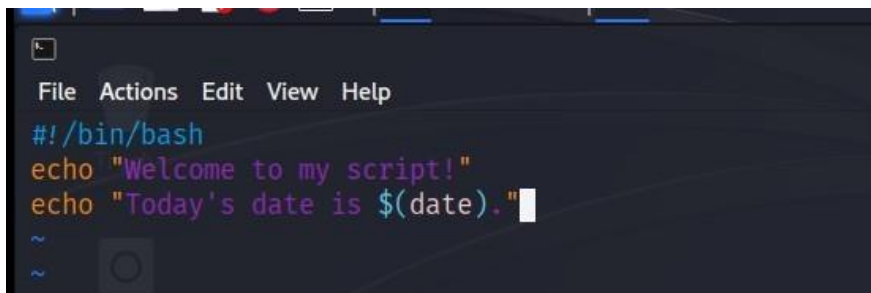
TASK 2: PERMISSIONS AND SCRIPTING

1. Create a new file called “my_script.sh”

```
(anngela@Anngela)-[~/my_directory]
$ touch my_script.sh

(anngela@Anngela)-[~/my_directory]
$ ls
my_script.sh
```

2. Edit my_script.sh using any text editor , add the given lines, make it executable , and run.



A screenshot of a text editor window with a dark theme. The menu bar at the top includes 'File', 'Actions', 'Edit', 'View', and 'Help'. The text area contains the following lines: `#!/bin/bash`, `echo "Welcome to my script!"`, and `echo "Today's date is $(date)."` followed by a cursor. At the bottom left, there are two tabs, each labeled with a tilde (~).

```
(anngela@Anngela)-[~]
$ chmod +x my_script.sh

(anngela@Anngela)-[~]
$ ./my_script.sh
Welcome to my script!
Today's date is Wednesday 28 June 2023 10:59:58 PM IST.
```

TASK 3: COMMAND EXECUTION AND PIPELINES

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

```
(annabela@Anngela)-[~]
$ ps aux
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.0	0.1	102268	12344	?	Ss	19:52	0:01	/sbin/init splash
root	2	0.0	0.0	0	0	?	S	19:52	0:00	[kthreadd]
root	3	0.0	0.0	0	0	?	I<	19:52	0:00	[rcu_gp]
root	4	0.0	0.0	0	0	?	I<	19:52	0:00	[rcu_par_gp]
root	5	0.0	0.0	0	0	?	I<	19:52	0:00	[slub_flushwq]
root	6	0.0	0.0	0	0	?	I<	19:52	0:00	[netns]
root	10	0.0	0.0	0	0	?	I<	19:52	0:00	[mm_percpu_wq]
root	11	0.0	0.0	0	0	?	I	19:52	0:00	[rcu_tasks_kthread]
root	12	0.0	0.0	0	0	?	I	19:52	0:00	[rcu_tasks_rude_kth
root	13	0.0	0.0	0	0	?	I	19:52	0:00	[rcu_tasks_trace_kt
root	14	0.0	0.0	0	0	?	S	19:52	0:01	[ksoftirqd/0]
root	15	0.0	0.0	0	0	?	I	19:52	0:08	[rcu_preempt]
root	16	0.0	0.0	0	0	?	S	19:52	0:00	[migration/0]
root	18	0.0	0.0	0	0	?	S	19:52	0:00	[cpuhp/0]
root	19	0.0	0.0	0	0	?	S	19:52	0:00	[cpuhp/1]
root	20	0.0	0.0	0	0	?	S	19:52	0:00	[migration/1]
root	21	0.0	0.0	0	0	?	S	19:52	0:02	[ksoftirqd/1]
root	23	0.0	0.0	0	0	?	I<	19:52	0:00	[kworker/1:0H-event
root	26	0.0	0.0	0	0	?	S	19:52	0:00	[kdevtmpfs]
root	27	0.0	0.0	0	0	?	I<	19:52	0:00	[inet_frag_wq]
root	28	0.0	0.0	0	0	?	S	19:52	0:00	[kauditd]
root	29	0.0	0.0	0	0	?	S	19:52	0:00	[khungtaskd]
root	31	0.0	0.0	0	0	?	S	19:52	0:00	[oom_reaper]
root	32	0.0	0.0	0	0	?	I<	19:52	0:00	[writeback]
root	33	0.0	0.0	0	0	?	S	19:52	0:00	[kcompactd0]
root	34	0.0	0.0	0	0	?	SN	19:52	0:00	[ksmd]
root	35	0.0	0.0	0	0	?	SN	19:52	0:00	[khugepaged]
root	36	0.0	0.0	0	0	?	I<	19:52	0:00	[kintegrityd]
root	37	0.0	0.0	0	0	?	I<	19:52	0:00	[kblockd]
root	38	0.0	0.0	0	0	?	I<	19:52	0:00	[blkcg_punt_bio]
root	39	0.0	0.0	0	0	?	I<	19:52	0:00	[tpm_dev_wq]
root	40	0.0	0.0	0	0	?	I<	19:52	0:00	[edac-poller]

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

```
(annabela@Anngela)-[~]
$ ps aux | grep bash
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
annabela	96299	0.0	0.0	6332	2112	pts/0	S+	23:03	0:00	grep --color=auto bash

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

```
(annabela@Anngela)-[~]
$ ps aux | grep bash | wc -l
```

1