

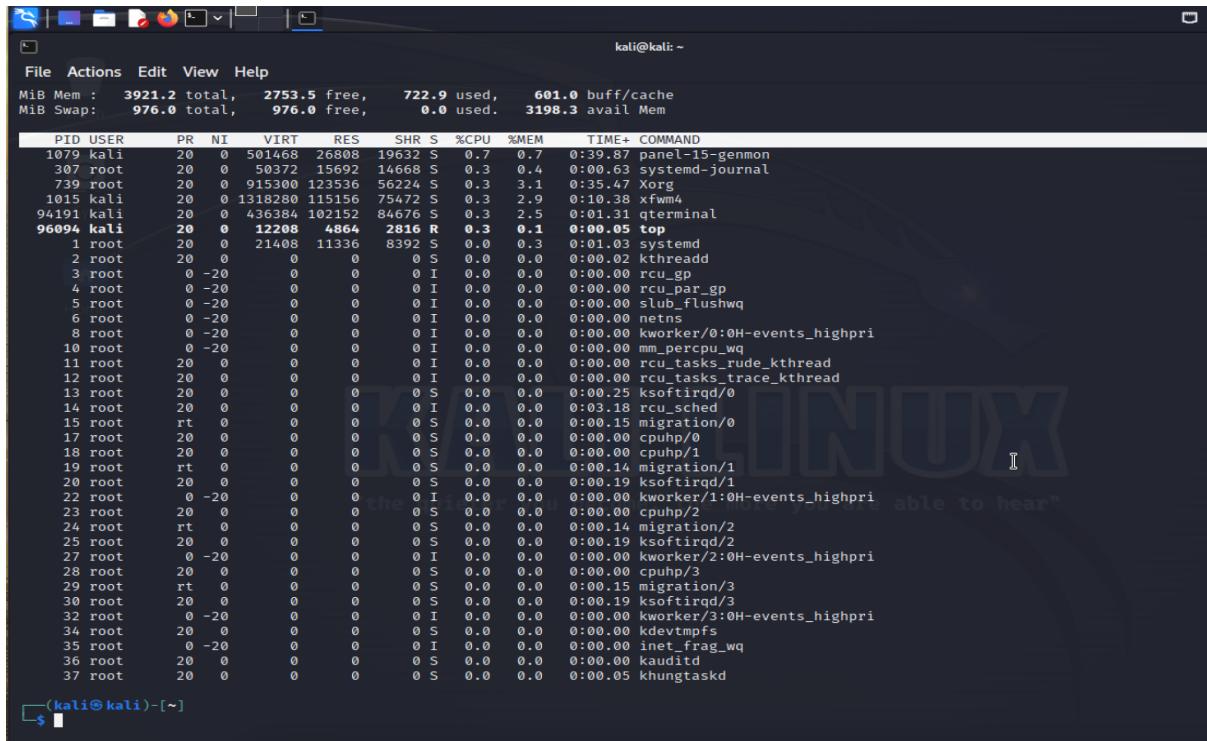
Esercizio S2/L2

Prima parte:

Traccia:

Nell'esercizio di oggi familiarizzeremo con i comandi da shell Linux. Pertanto, si richiede allo studente di:

- Controllare i processi attivi sulla macchina Linux con il comando «top» e descrivere il significato delle colonne: I) PID, USER, COMMAND;
- Filtrare i risultati del comando top inviando l'output al comando grep (utilizzare la pipe «|» per mostrare solo i programmi in esecuzione per l'utente «root»)
- Ripetere il punto 2, filtrando i risultati per mostrare solamente i processi in esecuzione dall'utente kali
- Creare una nuova directory chiamata «Episode_Lab» nella seguente directory /home/kali/Desktop Spostarsi nella directory appena creata e creare il file «Esercizio.txt»
- Modificare il file con l'editor di testo «nano», e salvatelo. Per salvare il file utilizzate la sequenza «ctrl+x» e successivamente «y», come mostrato in figura sotto.



PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1079	kali	20	0	501468	26808	19632	S	0.7	0.7	0:39.87	panel-15-genmon
307	root	20	0	50372	15692	14668	S	0.3	0.4	0:00.63	systemd-journal
739	root	20	0	915300	123536	56224	S	0.3	3.1	0:35.47	Xorg
1015	kali	20	0	1318280	115156	75472	S	0.3	2.9	0:10.38	xwmw4
94191	kali	20	0	436384	102152	84676	S	0.3	2.5	0:01.31	qterminal
96094	kali	20	0	12208	4864	2816	R	0.3	0.1	0:00.05	top
1	root	20	0	21408	11336	8392	S	0.0	0.3	0:01.03	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.02	kthreadd
3	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_gp
4	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	rcu_par_gp
5	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	slub_flushwq
6	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	netns
8	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/0:0H-events_highpri
10	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	mm_percpu_wq
11	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tasks_rude_kthread
12	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tasks_trace_kthread
13	root	20	0	0	0	0	S	0.0	0.0	0:00.25	ksoftirqd/0
14	root	20	0	0	0	0	I	0.0	0.0	0:03.18	rcu_sched
15	root	rt	0	0	0	0	S	0.0	0.0	0:00.15	migration/0
17	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
18	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/1
19	root	rt	0	0	0	0	S	0.0	0.0	0:00.14	migration/1
20	root	20	0	0	0	0	S	0.0	0.0	0:00.19	ksoftirqd/1
22	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/1:0H-events_highpri
23	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/2
24	root	rt	0	0	0	0	S	0.0	0.0	0:00.14	migration/2
25	root	20	0	0	0	0	S	0.0	0.0	0:00.19	ksoftirqd/2
27	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/2:0H-events_highpri
28	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/3
29	root	rt	0	0	0	0	S	0.0	0.0	0:00.15	migration/3
30	root	20	0	0	0	0	S	0.0	0.0	0:00.19	ksoftirqd/3
32	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/3:0H-events_highpri
34	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmpfs
35	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	inet_frag_wq
36	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kauditd
37	root	20	0	0	0	0	S	0.0	0.0	0:00.05	khungtaskd

tab.1

Nella prima schermata ho avviato PowerShell su Kali Linux e ho inserito il comando “**top**” per controllare i processi attivi.

A partire da dx troviamo:

PID serve per identificare il processo

USER è l'utente che ha avviato il processo

COMMAND è associata al comando o al programma associato al processo.

Es: (la linea in grassetto) vedi tab.1

PID 96094; USER kali; COMMAND top

```

kali@kali: ~
File Actions Edit View Help
top - 17:27:01 up 3:12, 2 users, load average: 0.01, 0.04, 0.00
Tasks: 159 total, 1 running, 158 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.2 us, 0.4 sy, 0.0 ni, 99.4 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 3921.2 total, 2756.3 free, 720.2 used, 600.7 buff/cache
MiB Swap: 976.0 total, 976.0 free, 0.0 used. 3200.9 avail Mem

PID USER PR NI VIRT RES SHR %CPU %MEM TIME+ COMMAND
739 root 20 0 915108 123384 56072 S 0.7 3.1 0:34.44 Xorg
1 root 20 0 21408 11336 8392 S 0.3 0.3 0:01.03 systemd
469 root 20 0 8100 7364 1536 S 0.3 0.2 0:00.88 haveged
1059 kali 20 0 504980 64852 33956 S 0.3 1.6 0:02.42 xfce4-panel
1081 kali 20 0 549192 40900 30696 S 0.3 1.0 0:00.33 panel-17-notifi
91020 root 20 0 0 0 0 I 0.3 0.0 0:00.02 kworker/2:0-events
94191 kali 20 0 436052 101872 84524 S 0.3 2.5 0:00.50 qterminal
2 root 20 0 0 0 0 S 0.0 0.0 0:00.02 kthreadd
3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_gp
4 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_par_gp
5 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 slab_flushwq
6 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 netns
8 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/0:0H-events_highpri
10 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 mm_percpu_wq
11 root 20 0 0 0 0 I 0.0 0.0 0:00.00 rcu_tasks_rude_kthread
12 root 20 0 0 0 0 I 0.0 0.0 0:00.00 rcu_tasks_trace_kthread
13 root 20 0 0 0 0 S 0.0 0.0 0:00.23 ksoftirqd/0
14 root 20 0 0 0 0 I 0.0 0.0 0:03.14 rcu_sched
15 root rt 0 0 0 0 S 0.0 0.0 0:00.15 migration
17 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/0
18 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/1
19 root rt 0 0 0 0 S 0.0 0.0 0:00.14 migration/1
20 root 20 0 0 0 0 S 0.0 0.0 0:00.19 ksoftirqd/1
22 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/1:0H-events_highpri
23 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/2
24 root rt 0 0 0 0 S 0.0 0.0 0:00.14 migration/2
25 root 20 0 0 0 0 S 0.0 0.0 0:00.19 ksoftirqd/2
27 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/2:0H-events_highpri
28 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/3
29 root rt 0 0 0 0 S 0.0 0.0 0:00.15 migration/3
30 root 20 0 0 0 0 S 0.0 0.0 0:00.18 ksoftirqd/3
32 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/3:0H-events_highpri
34 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kdevtmpfs
35 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 inet_frag_wq
36 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kauditd
37 root 20 0 0 0 0 S 0.0 0.0 0:00.05 khungtaskd

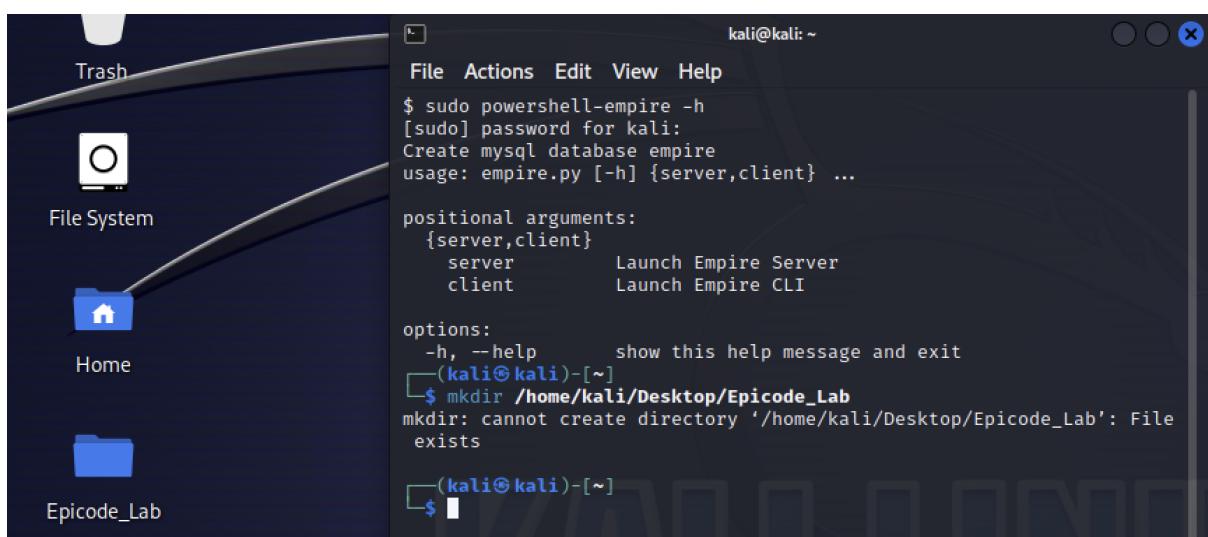
```

tab.2

Nella seconda parte dobbiamo filtrare i risultati solo per l'utente *root*, come comando ho fatto “**top grep | root**” in una nuova finestra di PowerShell, infatti possiamo vedere nella tab.2 che gli USER sono *root*.

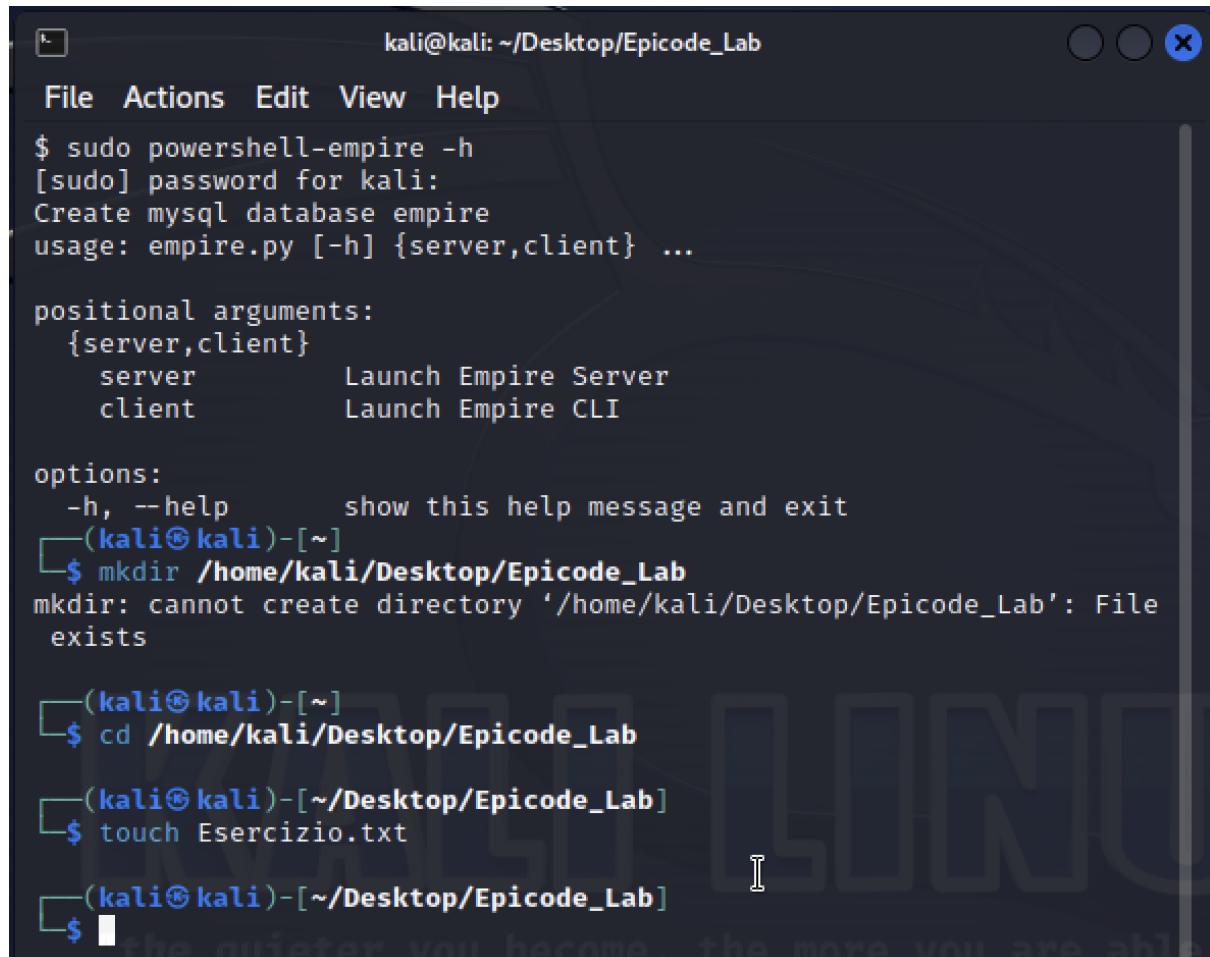
Ripetiamo lo stesso processo per filtrare l'utente *kali* aprendo un'altra finestra di PowerShell mettendo però questa volta “**top grep | kali**”.

Ora dobbiamo creare una nuova **Directory** chiamata **Epicode_Lab** sul desktop di *kali*, per farlo ho usato il comando “**mkdir /home/kali/Desktop/Epicode_Lab**” in questo modo ci troviamo la cartella nel Desktop come possiamo vedere nella tab.3.



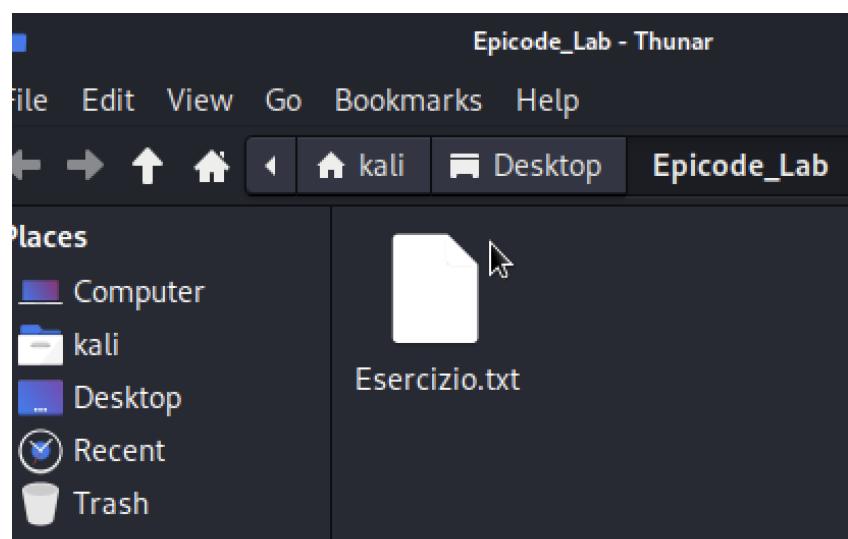
tab.3

A questo punto creiamo il file “Esercizio.txt” nella Directory *Epicode_Lab* usando il comando “`cd /home/kali/Desktop/Epicode_Lab`” e in seguito creo il file *Esercizio.txt* con il comando “`touch Esercizio.txt`”. (Vedi tab.4 e tab.5)



```
kali@kali: ~/Desktop/Epicode_Lab
File Actions Edit View Help
$ sudo powershell-empire -h
[sudo] password for kali:
Create mysql database empire
usage: empire.py [-h] {server,client} ...
positional arguments:
{server,client}
    server      Launch Empire Server
    client      Launch Empire CLI
options:
-h, --help      show this help message and exit
[(kali㉿kali)-[~]] $ mkdir /home/kali/Desktop/Epicode_Lab
mkdir: cannot create directory '/home/kali/Desktop/Epicode_Lab': File exists
[(kali㉿kali)-[~]] $ cd /home/kali/Desktop/Epicode_Lab
[(kali㉿kali)-[~/Desktop/Epicode_Lab]] $ touch Esercizio.txt
[(kali㉿kali)-[~/Desktop/Epicode_Lab]] $
```

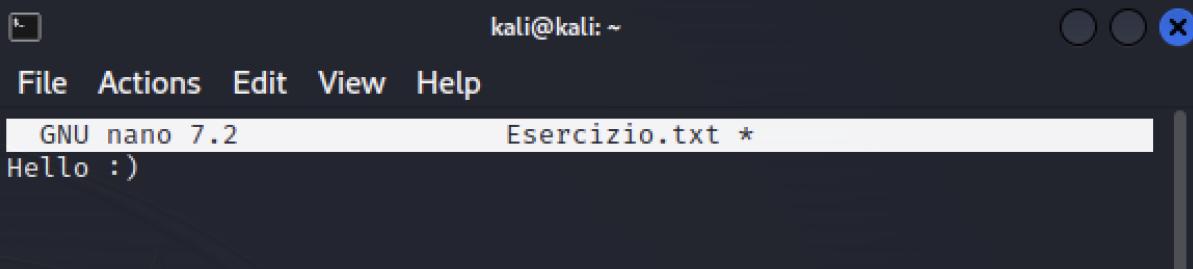
tab.4



tab.5

Ora bisogna modificare il file “Esercizio.txt” con l’editor di testo “nano” e salvarlo con i comandi “**CONTROL+X**” e accettare le modifiche con Y.

Dunque apro il file con “nano” e eseguo il comando “**nano Esercizio.txt**”. (Vedi tab.6)



```
File Actions Edit View Help
GNU nano 7.2          Esercizio.txt *
Hello :)
```

tab.6

Seconda parte:

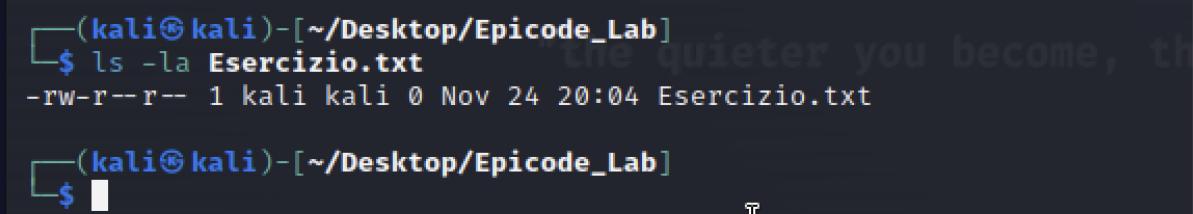
Traccia:

Nell’esercizio di oggi familiarizzeremo con i comandi da shell Linux. Pertanto, si richiede allo studente di:

1. Utilizzare il comando «cat» per leggere a schermo il file.txt appena modificato
2. Controllare i permessi del file con il comando ls –la
3. Modificare i privilegi del file in modo tale che l’utente corrente abbia tutti i privilegi (r,w,x), il gruppo (r,w), gli altri utenti solo lettura (r)
4. Creare un nuovo utente, chiamatelo pure come volete. Utilizzate il comando «useradd» per creare un utente e «passwd» seguita dal nome dell’utente per assegnare una password.
5. Con l’utente attuale cambiate i privilegi del file .txt creato in precedenza in modo tale che «altri utenti» non siano abilitati alla lettura
6. Spostate il file nella directory di root (/)
7. Cambiate utente con il comando «su» seguito dal nome dell’utente che volete utilizzare
8. Provate ad aprire in lettura il file.txt creato in precedenza con il comando nano, che errore ricevete?
9. Modificate i permessi del file per far in modo che il vostro nuovo utente possa leggerlo e ripetete gli ultimi 2 step.
10. Rimuovete il file, la cartelle e l’utente che avete creato, riportando lo scenario allo stato iniziale.

Utilizziamo il comando “**cat**” per visualizzare il contenuto del file creato “Esercizio.txt”.

Dunque utilizzo il comando “**cat Esercizio.txt**” nel portale PowerShell e troveremo la parola che avevo inserito “Hello :)”. Dopodiché controlliamo i permessi del file usando il comando “**ls -la Esercizio.txt**” (vedi tab.7)



```
(kali㉿kali)-[~/Desktop/Epicode_Lab]
$ ls -la Esercizio.txt
-rw-r--r-- 1 kali kali 0 Nov 24 20:04 Esercizio.txt

(kali㉿kali)-[~/Desktop/Epicode_Lab]
```

tab.7

Come possiamo vedere con il comando “**ls -la**” vengono visualizzati i permessi del file che sono -rw-r – r – 1 kali kali 0 Nov 24 20:04 Esercizio.txt.

Nello specifico si riferisce ad un file su un sistema UNIX e i caratteri indicano che:

(-) file regolare

(rw-) il proprietario ha il permesso di read cioè leggere e write cioè di scrivere sopra il file creato o esistente ma (-) non di eseguire.

(r-) indicano i permessi del gruppo e cioè il gruppo può solo read leggerlo ma non scriverlo o eseguirlo.

(r-) finale indica che altri utenti ugualmente possono solo leggerlo ma non scriverlo o eseguirlo.

Modifichiamo i privilegi sia per il gruppo (r,w) che per l’utente (r,w,x) attuale e per altri utenti solo (r) del file con il comando seguente “**chmod u=rwx,g=rw,o=r Esercizio.txt**” (Vedi tab.8)

```
(kali㉿kali)-[~/Desktop/Epicode_Lab]
$ chmod u=rwx,g=rw,o=r Esercizio.txt

(kali㉿kali)-[~/Desktop/Epicode_Lab]
$
```

tab.8

Creiamo un nuovo utente con il comando “**sudo useradd ket93**” creiamo la password con il comando “**sudo passwd ket93**” (Vedi tab.9)

```
(kali㉿kali)-[~/Desktop/Epicode_Lab]
$ sudo useradd ket93
[sudo] password for kali:
"the quieter you become, the more you are heard"

(kali㉿kali)-[~/Desktop/Epicode_Lab]
$ sudo passwd ket93
New password:
Retype new password:
passwd: password updated successfully

(kali㉿kali)-[~/Desktop/Epicode_Lab]
$
```

tab.9

Con lo stesso utente **ket93** cambiamo i privilegi del file in modo che gli altri utenti non possano leggerlo ed uso il comando “**chmod o-r Esercizio.txt**” (Vedi tab.10)

```
(kali㉿kali)-[~/Desktop/Epicode_Lab]
$ chmod o-r Esercizio.txt

(kali㉿kali)-[~/Desktop/Epicode_Lab]
$
```

tab.10

Dobbiamo spostare il file “Esercizio.txt” nella directory di root con il comando “**sudo mv Esercizio.txt**”. (Vedi tab.11)

```
[kali㉿kali)-[~/Desktop/Epicode_Lab]
$ sudo mv Esercizio.txt /
```

```
[kali㉿kali)-[~/Desktop/Epicode_Lab]
$ [REDACTED]
```

tab. 11

Cambiamo l’utente con il comando “**su**” dando al nuovo utente “**ket93**” i privilegi di super utente, infatti, il termine del comando “**su**” sta per substituteuser or switchuser. Dunque utilizzerò il comando “**su ket93**” (Vedi tab.12)

```
[kali㉿kali)-[~/Desktop/Epicode_Lab]
$ su ket93
```

```
Password:
```

```
$ [REDACTED]
```

tab. 12

Proviamo ad aprire il file in modalità lettura con il comando “**nano /Esercizio.txt**” Ora come errore mi dirà che la lettura è denied (Vedi tab.13)

```
[ Error reading /Esercizio.txt: Permission denied ] ...
```

```
^G Help      ^O Write Out ^W Where Is   ^K Cut        ^T Execute
^X Exit      ^R Read File ^\ Replace    ^U Paste     ^J Justify
```

tab. 13

Per consentire la lettura l’esercizio ci richiede di cambiare i permessi per consentire al nuovo utente di leggerlo.

Utilizzeremo come comando “**chmod u+r /Esercizio.txt**”

Grazie a questo comando adesso sono in grado di farlo leggere al nuovo utente ma non di scriverlo poiché ho inserito il comando **chmod** ma solo con **u** che sta ad indicare utente **+r** che sta ad indicare solo lettura.

Ora rimuoviamo il file, cartella, directory e utente “**ket93**” per tornare allo stato iniziale, usiamo i comandi “**sudo rm /Esercizio.txt**” e **sudo rmdir /home/ket93**

come risultato mi dice director non trovata dunque per ovviare al problema ho constatato che con il comando “**sudo userdel -r ket93**” ho cancellato definitivamente il nuovo utente e come risultato mi dice user does not exist, dunque è stato già eliminato in precedenza.**sudo userdel ket93**. (Vedi tab. 14)

```
kali@kali: ~
File Actions Edit View Help

[(kali㉿kali)-[~]
$ chmod u+r /Esercizio.txt

[(kali㉿kali)-[~]
$ sudo rm /Esercizio.txt
[sudo] password for kali:

[(kali㉿kali)-[~]
$ sudo rmdir /home/ket93
rmdir: failed to remove '/home/ket93': No such file or directory

[(kali㉿kali)-[~]
$ sudo userdel ket93
userdel: user ket93 is currently used by process 184826

[(kali㉿kali)-[~]
$ sudo userdel ket93
[

[(kali㉿kali)-[~]
$ ]
```

tab.14

CONCLUSIONE:

Creazione e modifica del file con i comandi **mkdir**, **cd**, **touch** per creare una nuova directory e un file.

Modifica del file con l'editor di test nano

Controllo dei permessi con **ls -la** e modifica dei privilegi con **chmod**

Creazione di un nuovo utente con **useradd** e assegnazione di una password con **passwd**

Cambio di permessi per i diversi utenti e spostamento del file in una posizione diversa.

Utilizzo del comando **su** per cambiare utente e tentativo di aprire l'editor con nano

Regolazione dei permessi per un utente specifico

Rimozione utente della directory e del file in maniera sicura

L'esercizio ha fornito una panoramica pratica di vari comandi e concetti di gestione dei file e degli utenti su sistemi Kali Linux e in particolare su PowerShell per la gestione dei permessi e la creazione di un nuovo file, directory e nuovo utente.

Questo dimostra in maniera efficiente come questi processi debbano essere condotti in maniera sicura e soprattutto questo esercizio ha dimostrato la cautela nell'utilizzare il comando **su** (super utente).