

## Esercizio week 2 – 2. Pre-requisiti Sistemi operativi

1. Il comando «top» fornisce diverse informazioni sui processi attivi (task), ma anche data, ora, e dettagli sull'uso della memoria.

```
(kali㉿kali)-[~]  
$ top
```

Con il termine **PID** si indica l'ID, l'identificativo, del processo; **USER** è l'utente che ha lanciato il processo e il **COMMAND** è il comando usato per creare il progetto.

```
top - 14:39:37 up 0 min, 1 user, load average: 1.57, 0.44, 0.15  
Tasks: 158 total, 1 running, 157 sleeping, 0 stopped, 0 zombie  
%Cpu(s): 1.5 us, 1.5 sy, 0.0 ni, 96.6 id, 0.2 wa, 0.0 hi, 0.2 si, 0.0 st  
MiB Mem : 1972.4 total, 973.0 free, 751.5 used, 397.5 buff/cache  
MiB Swap: 1024.0 total, 1024.0 free, 0.0 used, 1220.9 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
612	root	20	0	382736	99164	57748	S	1.3	4.9	0:01.49	Xorg
893	kali	20	0	875008	103252	77288	S	0.7	5.1	0:00.61	xfwm4
952	kali	20	0	204020	27872	18672	S	0.7	1.4	0:00.23	panel-13-cpugra
954	kali	20	0	358544	30592	20788	S	0.7	1.5	0:00.22	panel-15-genmon
1	root	20	0	102240	12084	9000	S	0.3	0.6	0:01.12	systemd
159	root	20	0	0	0	0	I	0.3	0.0	0:00.02	kworker/u4:3-events_unbound
316	root	20	0	41352	18060	14816	S	0.3	0.9	0:00.24	systemd-journal
414	message+	20	0	10032	5624	4252	S	0.3	0.3	0:00.36	dbus-daemon

2. usando il comando **top | grep "root"** filtro solo i processi dell'USER root

```
(kali㉿kali)-[~]  
$ top | grep "root"  
1 root 20 0 102240 12084 9000 S 0.0 0.6 0:01.17 systemd  
2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 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+  
5 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 slub_fl+  
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:
```

3. Ripeto il filtro per l'USER kali

```
(kali㉿kali)-[~]  
$ top | grep "kali"  
952 kali 20 0 352548 37508 21572 S 6.2 1.9 0:08.61 panel-1+  
19127 kali 20 0 11580 4968 3068 R 6.2 0.2 0:00.02 top
```

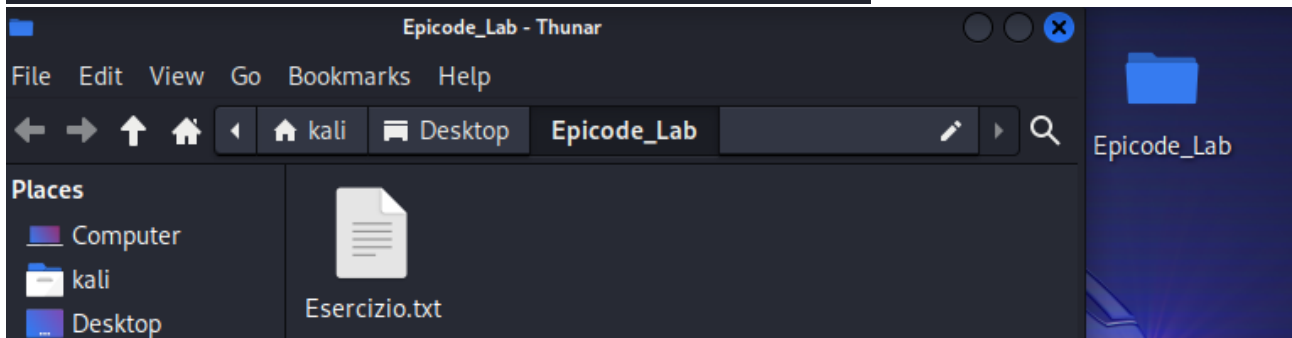
4. Con il comando **mkdir** creo la directory **Epicode\_Lab**, e con **mv** lo sposto sul **/home/kali/Desktop**

```
(kali㉿kali)-[~]  
$ mkdir "Epicode_Lab"  
  
(kali㉿kali)-[~]  
$ mv "Epicode_Lab" /home/kali/Desktop
```

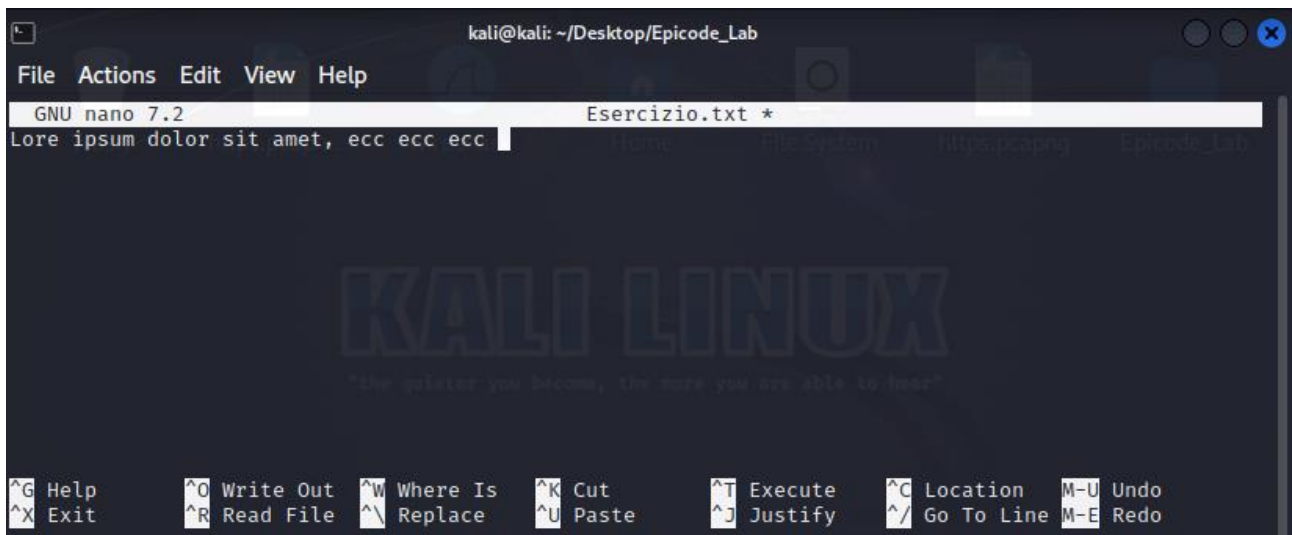
5. Con il comando **cd /home/kali/Desktop/Epicode\_Lab** mi sposto nella cartella appena creata, e con il comando **cat> Esercizio.txt** creo file di testo.

```
(kali@kali)-[~]
$ cd /home/kali/Desktop/Epicode_Lab

(kali@kali)-[~/Desktop/Epicode_Lab]
$ cat> Esercizio.txt
```



6. Apro il file .txt con il comando **nano Esercizio.txt** e scrivo una frase, salvo con Ctrl+o, Invio, e chiudo con Ctrl+X



7. Visualizzo i permessi del file, con il comando **ls -l** in questo caso abbiamo per **u** (utente) i permessi **r** e **w**; a **g** (gruppo) abbiamo **r** e a **o** (other, altri) è presente il solo permesso **r**.

```
(kali@kali)-[~/Desktop/Epicod]
$ ls -l Esercizio.txt
-rw-r--r-- 1 kali kali 35 May  9
```

8. Con il comando **chmod** diamo a **u** (utente) tutti i permessi, **r+w+x**; a **g** (gruppo) diamo **r+w**; a **o** (other, altri) diamo **r** con il comando **chmod 764 Esercizio.txt** sfruttando la sintassi ottale- I permessi saranno ora **u:rwX / g:rw / o:r**

```
(kali@kali)-[~/Desktop/Epicode_Lab]
$ chmod 764 Esercizio.txt
```



```
(kali@kali)-[~/Desktop/Epicod]
$ ls -l Esercizio.txt
-rwxrw-r-- 1 kali kali 35 May  9
```

9. Esco dalla directory e creo un nuovo utente chiamato **nolan** a cui assegno una **password**

```

(kali㉿kali)-[~/Desktop/Epicode_Lab]
$ sudo useradd nolan
[sudo] password for kali:

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

```

10. Tolgo i permessi di lettura al other e controllo che sia avvenuto

```

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

(kali㉿kali)-[~/Desktop/Epicode_Lab]
$ ls -l Esercizio.txt
-rwxrwx--- 1 kali kali 35 May  9 16:32 Esercizio.txt

```

11. Sposto il file Esercizio.txt nella directory /root, controllo e non è più presente nella vecchia cartella.

```

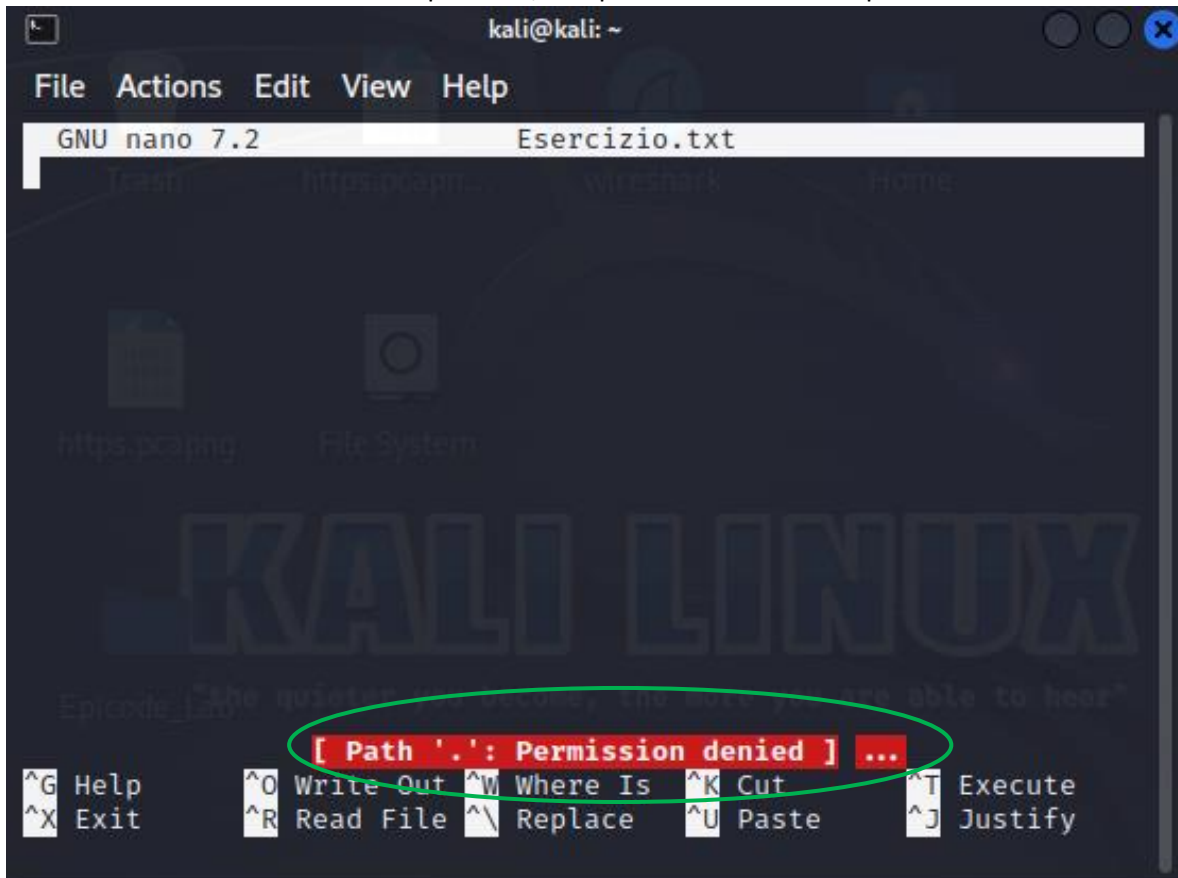
(kali㉿kali)-[~/Desktop/Epicode_Lab]
$ mv Esercizio.txt ~/

(kali㉿kali)-[~/Desktop/Epicode_Lab]
$ ls -l Esercizio.txt
ls: cannot access 'Esercizio.txt': No such file or directory

(kali㉿kali)-[~]
$ ls -l Esercizio.txt
-rwxrwx--- 1 kali kali 35 May  9 16:32 Esercizio.txt

```

12. Passando al nuovo utente apro il file, ma questo utente non ha i permessi necessari alla lettura



13. Torno su kali e rimetto i permessi di lettura agli others

```

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

(kali㉿kali)-[~]
$ ls -l Esercizio.txt
-rwxrw-r-- 1 kali kali 35 May  9 16:32 Esercizio.txt

```

14. Provo a scrivere e salvare il file ma non sono presenti i permessi di scrittura quindi non è possibile

```

[ Error writing Esercizio.txt\ : Permission denied ]
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^T Execute    ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^_ Go To Line

```

15. Cancello il file Esercizio.txt con il comando `rm -r`

```

(kali㉿kali)-[~]
$ rm -r Esercizio.txt

(kali㉿kali)-[~]
$ rm -r Esercizio.txt
rm: cannot remove 'Esercizio.txt': No such file or directory

```

16. Apro il percorso per il Desktop e cancello la directory Epicode\_Lab

```

(kali㉿kali)-[~]
$ cd /home/kali/Desktop/

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

(kali㉿kali)-[~/Desktop]
$ rm -r Epicode_Lab
rm: cannot remove 'Epicode_Lab': No such file or directory

```

17. Infine, elimino l'utente **nolan** con il comando `userdel nolan`

```

(kali㉿kali)-[~/Desktop]
$ sudo userdel nolan
[sudo] password for kali:

(kali㉿kali)-[~/Desktop]
$ su nolan
su: user nolan does not exist or the user entry does not contain all the require
d fields

```