

COMANDI SHELL LINUX

1- Controllare i processi attivi sulla macchina Linux con il comando «top» e descrivere il significato delle colonne: PID, USER, COMMAND

```
(kali@kali)-[~]
$ sudo top
[sudo] password for kali:
```

```
top - 18:40:11 up 14 min, 1 user, load average: 0.34, 0.37, 0.32
Tasks: 220 total, 1 running, 219 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 25689.2 total, 24441.5 free, 1079.9 used, 520.4 buff/cache
MiB Swap: 975.0 total, 975.0 free, 0.0 used, 24609.3 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1	root	20	0	20824	12160	9216	S	0.0	0.0	0:03.76	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.17	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_kthread
12	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tasks_rude_kthread
13	root	20	0	0	0	0	I	0.0	0.0	0:00.00	rcu_tasks_trace_kthread
14	root	20	0	0	0	0	S	0.0	0.0	0:01.17	ksoftirqd/0
15	root	20	0	0	0	0	I	0.0	0.0	0:01.24	rcu_preempt
16	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migration/0
17	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/0
18	root	20	0	0	0	0	I	0.0	0.0	0:07.47	kworker/0:1-ata_sff
19	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/0
20	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/1
21	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/1
22	root	rt	0	0	0	0	S	0.0	0.0	0:00.19	migration/1
23	root	20	0	0	0	0	S	0.0	0.0	0:02.00	ksoftirqd/1
25	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/1:0H-events_highpri
26	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/2
27	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/2
28	root	rt	0	0	0	0	S	0.0	0.0	0:00.20	migration/2
29	root	20	0	0	0	0	S	0.0	0.0	0:01.90	ksoftirqd/2
31	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/2:0H-events_highpri
32	root	20	0	0	0	0	S	0.0	0.0	0:00.00	cpuhp/3
33	root	-51	0	0	0	0	S	0.0	0.0	0:00.00	idle_inject/3
34	root	rt	0	0	0	0	S	0.0	0.0	0:00.20	migration/3
35	root	20	0	0	0	0	S	0.0	0.0	0:08.16	ksoftirqd/3
37	root	0	-20	0	0	0	I	0.0	0.0	0:00.00	kworker/3:0H-events_highpri

- 1- PID (process id): è il numero univoco che identifica il processo e viene assegnato in modo sequenziale
- 2- USER: indica l'utente che ha avviato il processo
- 3- COMMAND: indica il nome del comando/programma associato al processo

2- 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:

```
(kali@kali)-[~]
$ top | grep root
 1 root    20   0   20824   12160   9216 S   0.0   0.0   0:03.76 systemd
 2 root    20   0     0     0     0 S   0.0   0.0   0:00.17 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_fli+
 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+
10 root    0  -20    0     0     0 I   0.0   0.0   0:00.00 mm_perc+
11 root    20   0     0     0     0 I   0.0   0.0   0:00.00 rcu_tas+
12 root    20   0     0     0     0 I   0.0   0.0   0:00.00 rcu_tas+
13 root    20   0     0     0     0 I   0.0   0.0   0:00.00 rcu_tas+
14 root    20   0     0     0     0 I   0.0   0.0   0:01.44 ksoftir+
15 root    20   0     0     0     0 I   0.0   0.0   0:01.36 rcu_pre+
16 root    rt   0     0     0     0 S   0.0   0.0   0:00.00 migrati+
17 root    -51  0     0     0     0 S   0.0   0.0   0:00.00 idle_in+
19 root    20   0     0     0     0 S   0.0   0.0   0:00.00 cpuhp/0
20 root    20   0     0     0     0 S   0.0   0.0   0:00.00 cpuhp/1
```

```
(kali@kali)-[~]
$ top | grep kali
1131 kali    20   0   560000   50532   37672 S   0.3   0.2   0:07.44 nm-appl+
1057 kali    20   0   470432   43452   33476 S   0.6   0.2   0:09.89 xfwm4
1131 kali    20   0   560000   50532   37672 S   0.6   0.2   0:07.46 nm-appl+
1082 kali    20   0   305540   31808   20032 S   0.3   0.1   0:02.08 xfsetti+
1107 kali    20   0   660056   109596   43132 S   0.3   0.4   0:02.32 xfdesk+
1248 kali    20   0   302608   45536   18688 S   0.3   0.2   0:09.20 panel-1+
1249 kali    20   0   413916   26352   17664 S   0.3   0.1   0:00.86 panel-1+
1250 kali    20   0   358556   27436   20428 S   0.3   0.1   0:06.51 panel-1+
2721 kali    20   0   452452   108008   88264 S   0.3   0.4   0:03.88 qtermin+
```

con questi due comandi andiamo a filtrare i risultati prima user root e poi user kali

3- Creare una nuova directory chiamata «Epicode_Lab» nella seguente directory /home/kali/Desktop:

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

ci spostiamo con il comando “cd” in /home/kali/Desktop, dopodichè con il comando “mkdir” creiamo la nuova directory. Controlliamo con “ls”

4- Spostarsi nella directory appena creata e creare il file «Esercizio.txt»:

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

creiamo il nuovo file con il comando “touch”

5- Modificare il file con l’editor di testo «nano»:

```
(kali@kali)-[~/Desktop/Epicode_Lab]
$ sudo nano Esercizio.txt
[sudo] password for kali:
```

6- Utilizzare il comando «cat» per leggere a schermo il file.txt appena modificato:

```
(kali@kali)-[~/Desktop/Epicode_Lab]
$ cat Esercizio.txt
Provo a modificare questo esercizio
```

7- Controllare i permessi del file con il comando ls -la:

```
(kali@kali)-[~/Desktop/Epicode_Lab]
$ ls -la
total 12
drwxr-xr-x 2 kali kali 4096 Nov 28 18:30 .
drwxr-xr-x 3 kali kali 4096 Nov 28 18:29 ..
-rw-r--r-- 1 kali kali 36 Nov 28 18:30 Esercizio.txt
```

8- 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) :

```
(kali@kali)-[~/Desktop/Epicode_Lab]
$ chmod u+x Esercizio.txt
(kali@kali)-[~/Desktop/Epicode_Lab]
$ chmod g+w Esercizio.txt
(kali@kali)-[~/Desktop/Epicode_Lab]
$ ls -la
total 12
drwxr-xr-x 2 kali kali 4096 Nov 28 18:30 .
drwxr-xr-x 3 kali kali 4096 Nov 28 18:29 ..
-rwxrw-r-- 1 kali kali 36 Nov 28 18:30 Esercizio.txt
```

9- 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:

```
(kali@kali)-[~/Desktop/Epicode_Lab]
$ sudo useradd Prova
(kali@kali)-[~/Desktop/Epicode_Lab]
$ passwd Prova
passwd: You may not view or modify password information for Prova.
(kali@kali)-[~/Desktop/Epicode_Lab]
$ sudo passwd Prova
New password:
Retype new password:
passwd: password updated successfully
```

10- Con l’utente attuale cambiate i privilegi del file .txt creato in precedenza in modo tale che «altri utenti» non siano abilitati alla lettura:

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

11- Spostate il file nella directory di root (/)

```
(kali@kali)-[~/Desktop/Epicode_Lab]
└─$ sudo mv Esercizio.txt /

(kali@kali)-[~/Desktop/Epicode_Lab]
└─$ cd /

(kali@kali)-[/]
└─$ ls
Esercizio.txt  boot  etc  initrd.img  lib  lib64  lost+found  mnt  proc  run  srv  tmp  var  vmlinuz  vmlinuz.old
bin            dev  home  initrd.img.old  lib32  libx32  media  opt  root  sbin  sys  usr  vmlinuz
```

12-Cambiate utente con il comando «su» seguito dal nome dell'utente che volete utilizzare:

13- Provate ad aprire in lettura il file.txt creato in precedenza con il comando nano, che errore ricevete

```
(kali@kali)-[/]
└─$ sudo su Prova
$ ls
Esercizio.txt  boot  etc  initrd.img  lib  lib64  lost+found  mnt  proc  run  srv  tmp  var  vmlinuz  vmlinuz.old
bin            dev  home  initrd.img.old  lib32  libx32  media  opt  root  sbin  sys  usr  vmlinuz
$ cat Esercizio.txt
cat: Esercizio.txt: Permission denied
$ exit
```

i permessi sono negati perché sono stati cambiati in precedenza

14- Modificate i permessi del file per far in modo che il vostro nuovo utente possa leggerlo e ripetete gli ultimi 2 step:

```
(kali@kali)-[/]
└─$ chmod o+r Esercizio.txt

(kali@kali)-[/]
└─$ su Prova
Password:
$ cat Esercizio.txt
Provo a modificare questo esercizio
$ exit
```

con l'utente appena creato il file è leggibile perché sono stati cambiati i permessi

15- Riportiamo tutto allo stato iniziale:

```
(kali@kali)-[/]
└─$ sudo deluser Prova
info: Removing crontab ...
info: Removing user 'Prova' ...

(kali@kali)-[/]
└─$ sudo deluser Prova
fatal: The user 'Prova' does not exist.

(kali@kali)-[/]
└─$ ls
Esercizio.txt  boot  etc  initrd.img  lib  lib64  lost+found  mnt  proc  run  srv  tmp  var  vmlinuz  vmlinuz.old
bin            dev  home  initrd.img.old  lib32  libx32  media  opt  root  sbin  sys  usr  vmlinuz

(kali@kali)-[/]
└─$ sudo rm Esercizio.txt

(kali@kali)-[/]
└─$ ls
bin  dev  home  initrd.img  lib  lib64  lost+found  mnt  proc  run  srv  tmp  var  vmlinuz  vmlinuz.old
boot  etc  initrd.img  lib  lib32  libx32  media  opt  root  sbin  sys  usr  vmlinuz

(kali@kali)-[/]
└─$ cd

(kali@kali)-[~]
└─$ ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  Videos

(kali@kali)-[~/Desktop]
└─$ cd Desktop

(kali@kali)-[~/Desktop]
└─$ ls
Epicode_Lab

(kali@kali)-[~/Desktop]
└─$ rmdir Epicode_Lab

(kali@kali)-[~/Desktop]
└─$ ls

(kali@kali)-[~/Desktop]
└─$
```