

## FONDAMENTI DI SHELL LINUX

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

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

top - 13:26:27 up 5 min, 2 users, load average: 0.00, 0.01, 0.00
Tasks: 174 total, 3 running, 171 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.3 us, 0.2 sy, 0.0 ni, 99.4 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 1976.2 total, 1106.2 free, 480.3 used, 389.7 buff/cache
MiB Swap: 976.0 total, 976.0 free, 0.0 used, 1404.0 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S  %CPU  %MEM    TIME+  COMMAND
 633 root        20   0 1423372 106204 49860 S   2.7   5.2   0:05.35 Xorg
1394 kali       20   0 412008 94696 75744 S   1.3   4.7   0:02.10 qterminal
 934 kali       20   0 343996 27432 19684 R   0.7   1.4   0:01.39 panel-15-genmon
    7 root        20   0      0      0      0 I   0.3   0.0   0:00.67 kworker/u16:0-events_unbound
 778 kali       9  -11  51240 29408 9260 S   0.3   1.5   0:00.03 pipewire
 779 kali       20   0 147996 25440 19000 S   0.3   1.3   0:00.17 pipewire-media-
 872 kali       20   0 1796428 92548 66820 R   0.3   4.6   0:00.70 xfwm4
 935 kali       20   0 660904 42612 33152 S   0.3   2.1   0:00.54 panel-16-pulsea
    1 root        20   0 165556 12460 7612 S   0.0   0.6   0:00.36 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_gp
    6 root        0 -20      0      0      0 I   0.0   0.0   0:00.00 kworker/0:0H-events_highpri
    8 root        0 -20      0      0      0 I   0.0   0.0   0:00.00 mm_percpu_wq
    9 root        20   0      0      0      0 S   0.0   0.0   0:00.00 rcu_tasks_rude_
   10 root        20   0      0      0      0 S   0.0   0.0   0:00.00 rcu_tasks_trace
   11 root        20   0      0      0      0 S   0.0   0.0   0:00.00 ksoftirqd/0
   12 root        20   0      0      0      0 I   0.0   0.0   0:00.21 rcu_sched
   13 root        rt   0      0      0      0 S   0.0   0.0   0:00.00 migration/0
   14 root        20   0      0      0      0 I   0.0   0.0   0:00.00 kworker/0:1-events
   15 root        20   0      0      0      0 S   0.0   0.0   0:00.00 cpuhp/0
   16 root        20   0      0      0      0 S   0.0   0.0   0:00.00 cpuhp/1
   17 root        rt   0      0      0      0 S   0.0   0.0   0:00.00 migration/1
   18 root        20   0      0      0      0 S   0.0   0.0   0:00.00 ksoftirqd/1
   19 root        20   0      0      0      0 I   0.0   0.0   0:00.00 kworker/1:0-events_power_efficient
   20 root        0 -20      0      0      0 I   0.0   0.0   0:00.00 kworker/1:0H-events_highpri
   21 root        20   0      0      0      0 S   0.0   0.0   0:00.00 cpuhp/2
   22 root        rt   0      0      0      0 S   0.0   0.0   0:00.00 migration/2
   23 root        20   0      0      0      0 S   0.0   0.0   0:00.00 ksoftirqd/2
   25 root        0 -20      0      0      0 I   0.0   0.0   0:00.00 kworker/2:0H-events_highpri
   26 root        20   0      0      0      0 S   0.0   0.0   0:00.00 cpuhp/3
   27 root        rt   0      0      0      0 S   0.0   0.0   0:00.00 migration/3
   28 root        20   0      0      0      0 S   0.0   0.0   0:00.00 ksoftirqd/3
   30 root        0 -20      0      0      0 I   0.0   0.0   0:00.00 kworker/3:0H-events_highpri
   31 root        20   0      0      0      0 S   0.0   0.0   0:00.00 cpuhp/4
   32 root        rt   0      0      0      0 S   0.0   0.0   0:00.00 migration/4
```

PID: numero identificativo del processo. Ordinati, penso, in base alla % di CPU che usano.

USER: utente che esegue quel processo. In questo caso sono presenti solo 2 utenti: root e kali

COMMAND: rappresenta il comando da usare per attivare il 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»). Suggerimento: **top | grep «utente»**

```
(kali@kali)-[~]
$ top | grep root

  1 root        20   0 165560 12500 7612 S   0.0   0.6   0:00.52 systemd
  2 root        20   0      0      0      0 S   0.0   0.0   0:00.01 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
  6 root        0 -20      0      0      0 I   0.0   0.0   0:00.00 kworker/0:0H-events_highpri
  7 root        20   0      0      0      0 I   0.0   0.0   0:00.80 kworker/u16:0-flush-254:0
  8 root        0 -20      0      0      0 I   0.0   0.0   0:00.00 mm_percpu_wq
  9 root        20   0      0      0      0 S   0.0   0.0   0:00.00 rcu_tasks_rude_
 10 root        20   0      0      0      0 S   0.0   0.0   0:00.00 rcu_tasks_trace
 11 root        20   0      0      0      0 S   0.0   0.0   0:00.04 ksoftirqd/0
 12 root        20   0      0      0      0 I   0.0   0.0   0:01.22 rcu_sched
 13 root        rt   0      0      0      0 S   0.0   0.0   0:00.03 migration/0
 15 root        20   0      0      0      0 S   0.0   0.0   0:00.00 cpuhp/0
 16 root        20   0      0      0      0 S   0.0   0.0   0:00.00 cpuhp/1
 17 root        rt   0      0      0      0 S   0.0   0.0   0:00.03 migration/1
 18 root        20   0      0      0      0 S   0.0   0.0   0:00.00 ksoftirqd/1
 19 root        20   0      0      0      0 I   0.0   0.0   0:00.00 kworker/1:0-events
 20 root        0 -20      0      0      0 I   0.0   0.0   0:00.00 kworker/1:0H-events_highpri
 21 root        20   0      0      0      0 S   0.0   0.0   0:00.00 cpuhp/2
 22 root        rt   0      0      0      0 S   0.0   0.0   0:00.03 migration/2
 23 root        20   0      0      0      0 S   0.0   0.0   0:00.00 ksoftirqd/2
 25 root        0 -20      0      0      0 I   0.0   0.0   0:00.00 kworker/2:0H-events_highpri
 26 root        20   0      0      0      0 S   0.0   0.0   0:00.00 cpuhp/3
 27 root        rt   0      0      0      0 S   0.0   0.0   0:00.03 migration/3
 28 root        20   0      0      0      0 S   0.0   0.0   0:00.00 ksoftirqd/3
 30 root        0 -20      0      0      0 I   0.0   0.0   0:00.00 kworker/3:0H-events_highpri
 31 root        20   0      0      0      0 S   0.0   0.0   0:00.00 cpuhp/4
 32 root        rt   0      0      0      0 S   0.0   0.0   0:00.03 migration/4
 33 root        20   0      0      0      0 S   0.0   0.0   0:00.00 ksoftirqd/4
 35 root        0 -20      0      0      0 I   0.0   0.0   0:00.00 kworker/4:0H-events_highpri
 36 root        20   0      0      0      0 S   0.0   0.0   0:00.00 cpuhp/5
 37 root        rt   0      0      0      0 S   0.0   0.0   0:00.03 migration/5
 38 root        20   0      0      0      0 S   0.0   0.0   0:00.00 ksoftirqd/5
 40 root        0 -20      0      0      0 I   0.0   0.0   0:00.00 kworker/5:0H-events_highpri
```

3. Ripetere il punto 2, filtrando i risultati per mostrare solamente i processi in esecuzione dall'utente kali

```
(kali@kali)-[~]
$ top | grep kali
 9425 kali      20    0   10312   3388   2812 R   6.2    0.2   0:00.01 top
 8079 kali      20    0  412080  95636  76756 S   1.0    4.7   0:01.17 qterminal
   934 kali      20    0  343996  27520  19684 S   0.7    1.4   0:08.96 panel-15-genmon
   785 kali      20    0    9676   4804   3732 S   0.3    0.2   0:01.08 dbus-daemon
   934 kali      20    0  343996  27520  19684 S   0.3    1.4   0:08.97 panel-15-genmon
 8079 kali      20    0  412080  95636  76756 S   0.3    4.7   0:01.18 qterminal
 9425 kali      20    0   10312   3388   2812 R   0.3    0.2   0:00.02 top
   934 kali      20    0  343996  27520  19684 S   0.3    1.4   0:08.98 panel-15-genmon
   935 kali      20    0  660904  42612  33152 S   0.3    2.1   0:02.86 panel-16-pulsea
 9425 kali      20    0   10312   3388   2812 R   0.3    0.2   0:00.03 top
   934 kali      20    0  343996  27520  19684 S   0.3    1.4   0:08.99 panel-15-genmon
 8079 kali      20    0  412080  95636  76756 S   0.3    4.7   0:01.19 qterminal
```

4. Creare una nuova directory chiamata «Epicode\_Lab» nella seguente directory /home/kali/Desktop
5. Spostarsi nella directory appena creata e creare il file «Esercizio.txt»

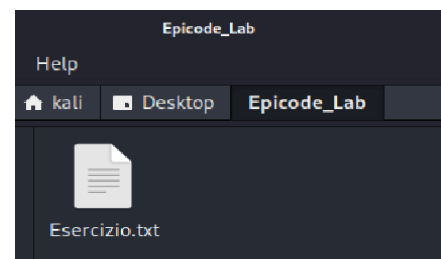
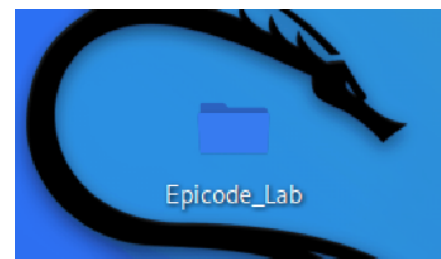
```
(kali@kali)-[~]
$ /home/kali/Desktop

(kali@kali)-[~/Desktop]
$ mkdir Epicode_Lab

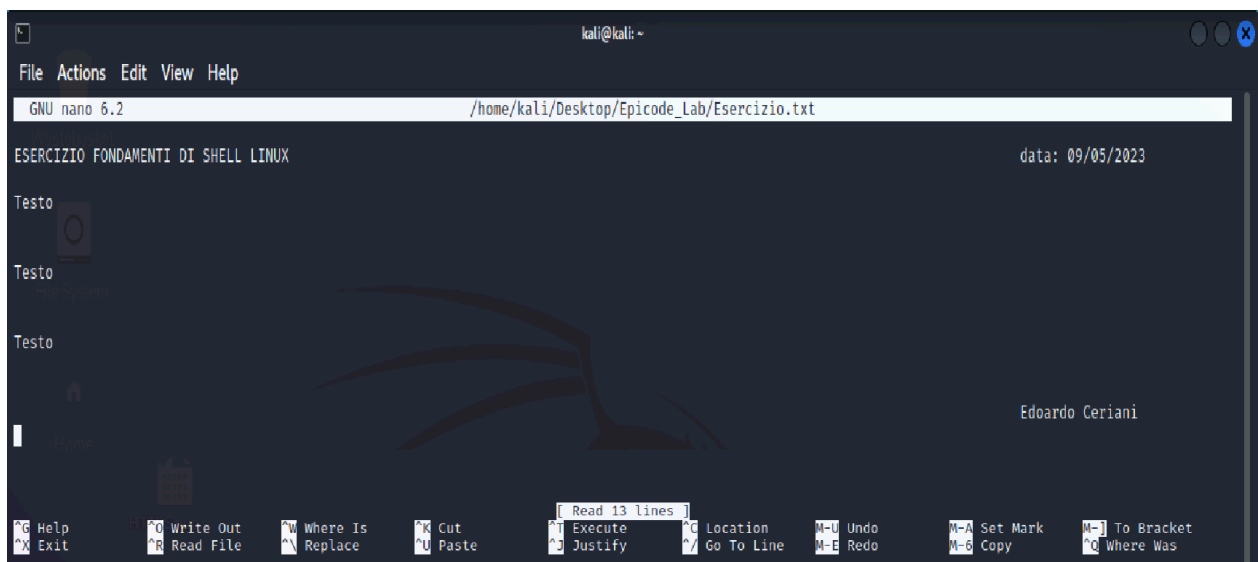
(kali@kali)-[~/Desktop]
$ cd Epicode_Lab

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

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



6. Modificare il file scrivendo qualcosa con l'editor di testo «nano», e salvarlo.



7. Utilizzare il comando «cat» per leggere a schermo il file.txt appena modificato

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

ESERCIZIO FONDAMENTI DI SHELL LINUX                                     data: 09/05/2023

Testo
Testo
Testo

Edoardo Ceriani
```

8. Controllare i permessi del file con il comando ls -la

```
(kali@kali)-[~]
$ ls -la /home/kali/Desktop/Epicode_Lab/Esercizio.txt
-rw-r--r-- 1 kali kali 122 May  9 14:19 /home/kali/Desktop/Epicode_Lab/Esercizio.txt
```

9. 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)-[~]
$ chmod u+x /home/kali/Desktop/Epicode_Lab/Esercizio.txt

(kali@kali)-[~]
$ chmod g+w /home/kali/Desktop/Epicode_Lab/Esercizio.txt

(kali@kali)-[~]
$ ls -la /home/kali/Desktop/Epicode_Lab/Esercizio.txt
-rwxrw-r-- 1 kali kali 122 May  9 14:19 /home/kali/Desktop/Epicode_Lab/Esercizio.txt
```

10. 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)-[~]
$ sudo useradd edo
[sudo] password for kali:
File system

(kali@kali)-[~]
$ sudo passwd edo
New password:
Retype new password:
passwd: password updated successfully
```

11. 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)-[~]  
$ chmod o-r /home/kali/Desktop/Epicode_Lab/Esercizio.txt  
  
(kali㉿kali)-[~]  
$ ls -la /home/kali/Desktop/Epicode_Lab/Esercizio.txt  
-rwxrwx--- 1 kali kali 122 May  9 14:19 /home/kali/Desktop/Epicode_Lab/Esercizio.txt
```

12. Spostate il file nella directory di root

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

13. Cambiate utente con il comando «su» seguito dal nome dell'utente che volete utilizzare

```
(kali㉿kali)-[~]  
$ su edo  
Password:  
$
```

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

```
GNU nano 6.2 /root/Esercizio.txt  
  
[ Path '/root' is not accessible ] ...  
^G Help      ^O Write Out  ^W Where Is   ^K Cut        ^J Execute    ^Q Location   ^U Undo       ^-A Set Mark  ^-] To Bracket  
^X Exit      ^R Read File  ^_ Replace    ^P Paste      ^_ Justify    ^_ Go To Line ^-R Redo     ^-C Copy      ^_ Where Was
```

Errore: [ Path '/root' is not accessible ]

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

```
(kali@kali)-[~]
$ sudo chmod o+r /root/Esercizio.txt

(kali@kali)-[~]
$ sudo ls -la /root/Esercizio.txt
-rwxr--r-- 1 kali kali 122 May  9 14:19 /root/Esercizio.txt
```



16. Rimuovete il file, le cartelle e l'utente che avete creato, riportando lo scenario allo stato iniziale.

FATTO MA SENZA SCREEN