

---

## Report

Filip S.  
Cybersecurity, Week 2, Day 1

# Comandi shell Linux

Con l'esercizio di oggi andremo a usare il Linux Kali del laboratorio virtuale per imparare a usare il Shell tramite Terminal.

I comandi che andremo a usare sono, top (table of processes) è il comando che ci mostra nel tempo reale i processi eseguiti nel Linux, tra altro ci mostra anche le risorse usate dal CPU e la memoria.

Spieghiamo quali funzioni hanno alcune colonne del comando Top

Dimostriamo come creare le directory e i file ".txt" come dare i permessi ad altri utenti o come rimuoverli.

Dimostriamo come spostare i file e cancellarli.

Dimostriamo come creare un utente senza permesso di leggere il file creato dal unaltro utente, per poi dimostrare come darli il permesso di sola lettura.

Alla fine cancelliamo tutti i cambiamenti fatti durante l'esercizio.

---

## Report week2,day2

Dopo aver acceso il Kali Linux, apriamo Terminal e scriviamo <<Top>> e inviamo il comando.

```
top - 09:49:39 up 52 min, 1 user, load average: 0.07, 0.09, 0.03
Tasks: 189 total, 1 running, 188 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.5 us, 0.3 sy, 0.0 ni, 99.2 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 7954.5 total, 6527.0 free, 804.6 used, 623.0 buff/cache
MiB Swap: 975.0 total, 975.0 free, 0.0 used, 6912.5 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM    TIME+  COMMAND
  654 root        20   0 407552 123328 56288  S   1.7   1.5    0:15.01 Xorg
 1400 filip       20   0 499728 110996 88952  S   1.0   1.4    0:03.83 qterminal
   828 filip    20   0 285000  43936 17636  S   0.3   0.5    0:01.03 xfce4-session
   933 filip    20   0 153000  2688  2208  S   0.3   0.0    0:03.06 VBoxClient
  1040 filip    20   0 231952  32148 19808  S   0.3   0.4    0:01.01 xfsettingsd
  1053 filip    20   0 496964  76908 37904  S   0.3   0.9    0:02.06 xfdesktop
  1057 filip    20   0 217252  40280 18848  S   0.3   0.5    0:07.80 panel-13-cpugra
  1061 filip    20   0 416880  30780 20828  S   0.3   0.4    0:04.45 panel-15-genmon
  1062 filip    20   0 661140  48000 36528  S   0.3   0.6    0:01.53 panel-16-pulsea
  1064 filip    20   0 393416  48924 33836  S   0.3   0.6    0:00.71 panel-18-power-
  1101 filip    20   0 188248  19780 15304  S   0.3   0.2    0:00.60 xfce4-power-man
  1115 filip    20   0 377048  54740 33712  S   0.3   0.7    0:01.01 blueman-applet
  1152 filip    20   0 562108  53400 39672  S   0.3   0.7    0:00.78 nm-applet
    1 root        20   0 168044  12016  8924  S   0.0   0.1    0:00.73 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
    5 root        0 -20      0      0      0  I   0.0   0.0    0:00.00 netns
    7 root        0 -20      0      0      0  I   0.0   0.0    0:00.00 kworker/0:0H-events_highpri
    9 root        0 -20      0      0      0  I   0.0   0.0    0:00.05 kworker/0:1H-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    0:00.00 ksoftirqd/0
   15 root        20   0      0      0      0  I   0.0   0.0    0:00.58 rcu_preempt
   16 root        rt   0      0      0      0  S   0.0   0.0    0:00.00 migration/0
   17 root        20   0      0      0      0  I   0.0   0.0    0:00.00 kworker/0:1-events
```

Come possiamo notare abbiamo più colonne con diverse informazioni, quelle che ci interessano sono **PID**, **USER** e **COMMAND**

**PID**: Process **ID**entifier è il numero usato da maggior parte dei OS kernel, come nel nostro caso Linux, per identificare un processo in attivo/esecuzione.

**USER**: **USER**name del proprietario del processo.

**COMMAND**: Il nome del **COMANDO** quale ha iniziato il processo.

Per fermare il Top bisogna premere “**CTRL+C**”

Per filtrare i risultati del comando <<Top>> per vedere solo i user “root” useremo il pipe seguito dal grep:

<<top | grep root>>

**GREP**: Global Regular Expression Print è il comando per cercare solo le “string” dei caratteri nella linea specifica.

```
top - 10:20:45 up 10 min, 1 user, load average: 0.10, 0.08, 0.02
Tasks: 658 total, 1 running, 657 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.5 us, 0.3 sy, 0.0 ni, 99.2 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 7954.5 total, 6527.0 free, 804.6 used, 623.0 buff/cache
MiB Swap: 975.0 total, 975.0 free, 0.0 used, 6912.5 avail Mem

  PID USER      PR  NI   VIRT   RES   SHR  S  %CPU  %MEM    TIME+  COMMAND
 658 root        20   0 405992 124212 56176  S   1.0   1.5    0:05.73 Xorg
    1 root        20   0 167912  11996  8916  S   0.0   0.1    0:00.64 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
    5 root        0 -20      0      0      0  I   0.0   0.0    0:00.00 netns
    7 root        0 -20      0      0      0  I   0.0   0.0    0:00.00 kworker/0:0H-events_highpri
    9 root        0 -20      0      0      0  I   0.0   0.0    0:00.01 kworker/0:1H-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    0:00.00 ksoftirqd/0
   15 root        20   0      0      0      0  I   0.0   0.0    0:00.12 rcu_preempt
   16 root        rt   0      0      0      0  S   0.0   0.0    0:00.00 migration/0
   18 root        20   0      0      0      0  S   0.0   0.0    0:00.00 cpuhp/0
   19 root        20   0      0      0      0  S   0.0   0.0    0:00.00 cpuhp/1
   20 root        rt   0      0      0      0  S   0.0   0.0    0:00.25 migration/1
   21 root        20   0      0      0      0  S   0.0   0.0    0:00.00 ksoftirqd/1
   23 root        0 -20      0      0      0  I   0.0   0.0    0:00.00 kworker/1:0H-kblockd
   24 root        20   0      0      0      0  S   0.0   0.0    0:00.00 cpuhp/2
   25 root        rt   0      0      0      0  S   0.0   0.0    0:00.25 migration/2
   26 root        20   0      0      0      0  S   0.0   0.0    0:00.00 ksoftirqd/2
   28 root        0 -20      0      0      0  I   0.0   0.0    0:00.00 kworker/2:0H-events_highpri
   29 root        20   0      0      0      0  S   0.0   0.0    0:00.00 cpuhp/3
   30 root        rt   0      0      0      0  S   0.0   0.0    0:00.25 migration/3
   31 root        20   0      0      0      0  S   0.0   0.0    0:00.00 ksoftirqd/3
   32 root        0 -20      0      0      0  I   0.0   0.0    0:00.00 kworker/3:0H-events_highpri
   33 root        20   0      0      0      0  S   0.0   0.0    0:00.00 cpuhp/4
```

Adesso faremo la stessa procedura per utente in questo caso “filip”, quindi scriveremo:

```
<<top | grep filip>>
```

```

filip@KaLinux: ~
File Actions Edit View Help
16 root rt 0 0 0 0 S 0.0 0.0 0:00.00 migration/0
18 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/0
19 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/1
20 root rt 0 0 0 0 S 0.0 0.0 0:00.25 migration/1
21 root 20 0 0 0 0 S 0.0 0.0 0:00.00 ksoftirqd/1
23 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/1:0H-kblockd
24 root 20 0 0 0 0 S 0.0 0.0 0:00.00 cpuhp/2
25 root rt 0 0 0 0 S 0.0 0.0 0:00.25 migration/2
26 root 20 0 0 0 0 S 0.0 0.0 0:00.00 ksoftirqd/2
28 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 kworker/2:0H-events_highpri

(filip@KaLinux)-[~]
$ top | grep filip
3132 filip 20 0 499780 110760 88772 S 0.7 1.4 0:00.89 qterminal
833 filip 20 0 269020 27572 17760 S 0.3 0.3 0:00.48 xfce4-session
937 filip 20 0 153000 2696 2216 S 0.3 0.0 0:00.88 VBoxClient
1034 filip 20 0 580416 101108 78340 S 0.3 1.2 0:01.25 xfwm4
1060 filip 20 0 217264 39872 18524 S 0.3 0.5 0:02.15 panel-13-cpugra
1064 filip 20 0 416864 30512 20712 S 0.3 0.4 0:01.36 panel-15-genmon
1065 filip 20 0 661136 49516 35972 S 0.3 0.6 0:00.59 panel-16-pulsea
1067 filip 20 0 327884 46576 33564 S 0.3 0.6 0:00.36 panel-18-power-
1101 filip 20 0 188244 19548 15216 S 0.3 0.2 0:00.28 xfce4-power-man
1164 filip 20 0 266960 25584 16584 S 0.3 0.3 0:00.28 light-locker
3132 filip 20 0 499780 110760 88772 S 1.3 1.4 0:00.93 qterminal
1056 filip 20 0 481616 62848 37432 S 0.7 0.8 0:01.02 xfdesktop
833 filip 20 0 269020 27572 17760 S 0.3 0.3 0:00.49 xfce4-session
1043 filip 20 0 231956 29060 18892 S 0.3 0.4 0:00.49 xfsettingsd
1051 filip 20 0 342136 26512 17196 S 0.3 0.3 0:00.30 Thunar
1060 filip 20 0 217264 39872 18524 S 0.3 0.5 0:02.16 panel-13-cpugra
1064 filip 20 0 416864 30512 20712 S 0.3 0.4 0:01.37 panel-15-genmon
1107 filip 20 0 262008 19384 15208 S 0.3 0.2 0:00.25 xfce4-notifd
1116 filip 20 0 377032 54304 33504 S 0.3 0.7 0:00.64 blueman-applet
1164 filip 20 0 266960 25584 16584 S 0.3 0.3 0:00.29 light-locker
1175 filip 20 0 788504 62584 46820 S 0.3 0.8 0:00.43 evolution-alarm

```

Per il prossimo esercizio andremo a creare una directory chiamata <<Epicode\_Lab>> nella directory del “/home/filip/Desktop”

La directory possiamo crearla se noi spostiamo la nostra “working directory” dove vogliamo che sia creata la nuova directory, come nel mio caso facendo “cd /home/filip/Desktop” e poi usiamo il comando “mkdir” per creare la nuova directory dandole il nome. Per essere più veloci possiamo scrivere “mkdir” e specificare dove crearla, alla fine dandole il nome, nel mio caso “Epicode\_Lab2”

```

(filip@KaLinux)-[~]
$ cd /home/filip/Desktop
(filip@KaLinux)-[~/Desktop]
$ mkdir Epicode_Lab
(filip@KaLinux)-[~/Desktop]
$ cd ..
(filip@KaLinux)-[~]
$ mkdir /home/filip/Desktop/Epicode_Lab2
(filip@KaLinux)-[~]
$ 

```

Adesso andremo a spostare la WD nella cartella creata

```

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

```

e creiamo il file <<Esercizio.txt>> con il comando “nano Esercizio.txt” il terminal ci aprirà il file dove possiamo scrivere per esempio “Hello World!” e per salvarlo faremo “Ctrl+O”>Invio per non modificare il nome del file>”Ctrl+X”

```

(filip@KaLinux)-[~/Desktop/Epicode_Lab]
$ nano Esercizio.txt
(filip@KaLinux)-[~/Desktop/Epicode_Lab]
$ 

```



Adesso possiamo usare il comando “cat” (concatenate) per vedere i contenuti del file sul Terminal.

```
(filip@KaLinux)-[~/Desktop/Epicode_Lab]
$ cat Esercizio.txt
Hello World!
```

Il prossimo passo ci indica di controllare i permessi del file con il comando “ls -la” e di cambiare i permessi del file per

filip:rwX  
gruppo:rw  
altri utenti:r

```
(filip@KaLinux)-[~/Desktop/Epicode_Lab]
$ ls -la
total 12
drwxr-xr-x 2 filip filip 4096 Nov  2 10:42 .
drwxr-xr-x 4 filip filip 4096 Nov  2 10:33 ..
-rw-r--r-- 1 filip filip  13 Nov  2 10:38 Esercizio.txt
```

Il nome del nostro file

La data e tempo del ultima modifica del file

I permessi del file  
r=read  
w=write  
x=execute

Owner del file, ovvero il proprietario

La grandezza del file (nei Bytes)

```
(filip@KaLinux)-[~/Desktop/Epicode_Lab]
$ chmod u+x Esercizio.txt

(filip@KaLinux)-[~/Desktop/Epicode_Lab]
$ ls -la
total 12
drwxr-xr-x 2 filip filip 4096 Nov  2 10:42 .
drwxr-xr-x 4 filip filip 4096 Nov  2 10:33 ..
-rwxr--r-- 1 filip filip  13 Nov  2 10:38 Esercizio.txt

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

(filip@KaLinux)-[~/Desktop/Epicode_Lab]
$ ls -la
total 12
drwxr-xr-x 2 filip filip 4096 Nov  2 10:42 .
drwxr-xr-x 4 filip filip 4096 Nov  2 10:33 ..
-rwxrw-r-- 1 filip filip  13 Nov  2 10:38 Esercizio.txt

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

Andremo a creare un nuovo utente chiamato “Mario”

```
(filip@KaLinux)-[~/Desktop/Epicode_Lab]
$ sudo su
[sudo] password for filip:
(filip@KaLinux)-[~/Desktop/Epicode_Lab]
# useradd Mario
```

li daremo una password debole come “123456”

```
(root@KaLinux)-[/home/filip/Desktop/Epicode_Lab]
# passwd Mario
New password:
Retype new password:
passwd: password updated successfully

(root@KaLinux)-[/home/filip/Desktop/Epicode_Lab]
# exit
```

---

## Report week2,day2

Cambiamo i permessi per il nostro file Esercizio.txt per altri (Other) nel - - - (ovvero che non possano neanche leggerlo).

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

(filip@KaLinux)-[~/Desktop/Epicode_Lab]
$ ls -la
total 12
drwxr-xr-x 2 filip filip 4096 Nov  2 10:42 .
drwxr-xr-x 4 filip filip 4096 Nov  2 10:33 ..
-rwxrw---- 1 filip filip  13 Nov  2 10:38 Esercizio.txt
```

Adesso sposteremo il file nella directory del root (/)

```
(filip@KaLinux)-[~/Desktop/Epicode_Lab]
$ sudo su
(root@KaLinux)-[/home/filip/Desktop/Epicode_Lab]
# mv Esercizio.txt /

(root@KaLinux)-[/home/filip/Desktop/Epicode_Lab]
# ls
```

cambiamo l'utente in quello del Mario e mettiamo la password 123456

```
(filip@KaLinux)-[~/Desktop/Epicode_Lab]
$ su Mario
Password:
```

andremo nella directory del root (/) dove sta il nostro file Esercizio.txt

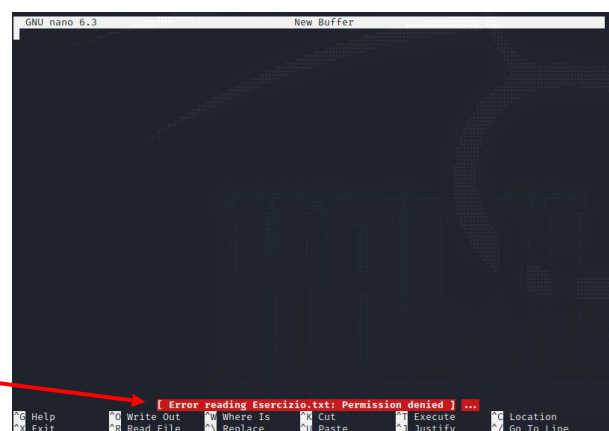
```
$ pwd
/home/filip/Desktop/Epicode_Lab
$ cd /
$ pwd
/
$ ls
0      dev      home      lib      libx32    mnt      root     srv      usr      vmlinuz.old
bin    Esercizio.txt  initrd.img  lib32    lost+found  opt      run      sys      var
boot  etc          initrd.img.old  lib64    media     proc     sbin     tmp      vmlinuz
```

e andiamo a provare a leggere con "cat" il file. Come possiamo vedere non abbiamo il permesso.

```
$ cat Esercizio.txt
cat: Esercizio.txt: Permission denied
```

Possiamo provare a modificarlo con "nano Esercizio.txt"

ancora una volta ci da "Permission denied"



```
GNU nano 6.3      New Buffer
[Error reading Esercizio.txt: Permission denied] ...
Help Exit Write Out Read File Where Is Replace Cut Paste Execute Justify Location Go To Line
```

Torniamo al utente filip per modificare il permesso per Altri (Other) che possano leggere il file.

```
$ su filip
Password:
(filip@KaLinux)-[/]
$ ls -la
drwxr-xr-x 17 root root 3240 Nov 2 10:59 dev
-rwxrw- 1 filip filip 13 Nov 2 10:38 Esercizio.txt
drwxr-xr-x 170 root root 12288 Nov 2 11:10 etc

(filip@KaLinux)-[/]
$ chmod o+r Esercizio.txt

(filip@KaLinux)-[/]
$ ls -la
drwxr-xr-x 17 root root 3240 Nov 2 10:59 dev
-rwxrw-r- 1 filip filip 13 Nov 2 10:38 Esercizio.txt
drwxr-xr-x 170 root root 12288 Nov 2 11:10 etc
```

Qui possiamo vedere che sotto la categoria Other abbiamo r

torniamo al utente Mario

```
(filip@KaLinux)-[/]
$ su Mario
Password:
$ nano Esercizio.txt

GNU nano 6.3 Esercizio.txt
Hello World!
|

[File 'Esercizio.txt' is unwritable] ...
```

Questa volta possiamo leggere il file.

però non possiamo modificarlo

Adesso andremo a riportare lo scenario allo stato iniziale.

Torniamo al utente filip e rimuoviamo il file Esercizio.txt

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

(filip@KaLinux)-[/]
$ sudo rm Esercizio.txt
[sudo] password for filip:

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

rimuoviamo la directory Epicode\_Lab

```
(filip@KaLinux)-[/]
$ rmdir home/filip/Desktop/Epicode_Lab

(filip@KaLinux)-[~]
$ cd /home/filip/Desktop/Epicode_Lab
cd: no such file or directory: /home/filip/Desktop/Epicode_Lab

(filip@KaLinux)-[~]
$
```

per cancellare user Mario, basta dare il comando con sudo “userdel” e nome utente

```
filip@KaLinux: ~
File Actions Edit View Help

(filip@KaLinux)-[~]
$ sudo userdel Mario
[sudo] password for filip:
(filip@KaLinux)-[~]
$ su Mario
su: user Mario does not exist or the user entry does not contain all the required fields
```