

ESERCIZIO W5D1 PARTE 2

Da qui in poi parte la seconda fase dell'esercizio. Abbiamo un'altra sfilza di incarichi da portare a termine.

j) Cambia gli attributi della cartella lavoro

Assegniamo privilegi a lavoro, in questo modo solo noi possiamo entrare al suo interno. La traccia chiedeva anche che si negasse il diritto di lettura.

```
(kali㉿kali)-[~/studenti/nicola]
$ sudo chmod 100 /home/kali/studenti/nicola/lavoro

(kali㉿kali)-[~/studenti/nicola]
$ ls -l
total 16
-rw-r--r-- 1 root root  8 Nov 22 13:21 compito.doc
d--x----- 2 kali kali 4096 Nov 22 13:18 lavoro
-rw-r--r-- 1 root root 10 Nov 22 13:21 relazione.doc
drwxr-xr-x 2 kali kali 4096 Nov 22 13:18 scuola

(kali㉿kali)-[~/studenti/nicola]
$ cd lavoro

(kali㉿kali)-[~/studenti/nicola/lavoro]
$
```

assegnazione privilegi lavoro

k) Entra nella cartella lavoro e crea un file emacs

Entriamo dentro la cartella a cui solo l'utente può accedere per creare un file con emacs; per crearlo utilizzeremo `sudo emacs somefile.txt`. Nel caso vogliamo vedere questo file, solo `sudo ls` ce lo permetterà.

```
File Edit Options Buffers Tools Text Help
+ - x Save Undo ✕
Un bellissimo testo dentro emacs
```

emacs running

```
(kali㉿kali)-[~/studenti/nicola/lavoro]
$ sudo ls
somefile.txt
```

Creazione file in lavoro

l) cambia i diritti della cartella lavoro

Se vogliamo entrare nuovamente nella cartella dobbiamo aggiungere i diritti di esecuzione e scrittura con il comando `sudo chmod 500 lavoro`.

```
(kali㉿kali)-[~/studenti/nicola]
$ sudo chmod 500 lavoro

(kali㉿kali)-[~/studenti/nicola]
$ cd lavoro

(kali㉿kali)-[~/studenti/nicola/lavoro]
$ ls
somefile.txt
```

assegnazione diritti

m) Spostati nella cartella scuola usando il percorso relativo

Come da istruzioni, usiamo un percorso relativo per muoverci dentro la cartella scuola.

```
(kali㉿kali)-[~/studenti/nicola]
$ cd lavoro

(kali㉿kali)-[~/studenti/nicola/lavoro]
$ cd ../scuola

(kali㉿kali)-[~/studenti/nicola/scuola]
$
```

percorso relativo scuola

n) Crea una directory `.mia` all'interno della cartella scuola. Fornisci tutti i privilegi all'utente.

Creiamo la cartella nascosta .mia e le conferiamo tutti i privilegi tramite sudo chmod 700 .mia.

```
(kali㉿kali)-[~/studenti/nicola/scuola]
$ sudo chmod 700 .mia
[sudo] password for kali: 
```

Cambio privilegi mia

o) Esegui il comando ls e successivamente ls -a

Testiamo i risultati dei due comandi ls e ls-a. il primo comando non fornisce nulla poiché la cartella di scuola è nascosta, mentre il secondo ci mostra il contenuto di scuola ed i privilegi annessi.

```
(kali㉿kali)-[~/studenti/nicola/scuola]
$ ls
he quieter you become, the mo

(kali㉿kali)-[~/studenti/nicola/scuola]
$ ls -al
total 12
drwxr-xr-x 3 kali kali 4096 Nov 24 07:02 .
drwxr-xr-x 4 kali kali 4096 Nov 22 13:21 ..
drwx----- 2 kali kali 4096 Nov 24 07:02 .mia
```

ls e ls -al

ESERCIZIO PROCESSI

10) Esegui il comando emacs &

Inizia la fase riguardante i processi e la loro manutenzione. Mettiamo un file di testo in background con l'aiuto di emacs & ,

```
(kali㉿kali)-[~]
$ setxkbmap -layout it

(kali㉿kali)-[~]
$ emacs &
[1] 3278


(kali㉿kali)-[~]
$ 
```

Mandare emacs in bg

11) Esegui il comando jobs

Su terminale inseriamo il comando jobs per vedere i processi in bg attivi. In questo visualizziamo emacs in esecuzione.


```
(kali@kali)-[~]  
$ jobs  
[1] + running      emacs
```



12) Esegui Firefox ed interrompilo con ^Z prima che finisca

Prima che finisca l'esecuzione possiamo premere Ctrl + Z per interrompere un processo prima che finisca.


```
(kali@kali)-[~]  
$ firefox  
^Z  
zsh: suspended  firefox  
^Z
```



13) Manda il processo firefox in bg

Riprendiamo il processo bloccato di prima e lo concludiamo scrivendo bg firefox.

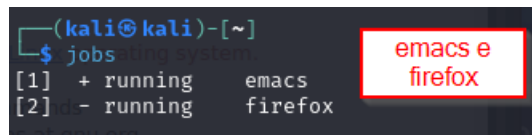
```
(kali@kali)-[~]  
$ bg firefox  
[2] - continued  firefox
```



14) Esegui nuovamente jobs

Come da titolo, controlliamo di nuovo la lista dei processi in esecuzione. Questa troviamo anche firefox in aggiunta, poiché ancora non è stato chiuso.

```
(kali㉿kali)-[~]  
$ jobs  
[1] + running  emacs  
[2] - running  firefox
```



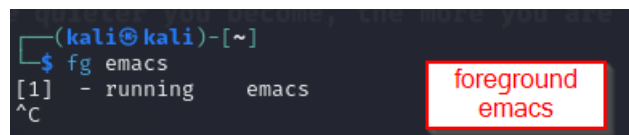
15) Manda in foreground il programma emacs

Mandiamo il processo emacs in foreground, ponendolo in attesa di un input da parte nostra per continuare.

16) Concludi processo in foreground con ^C

Essendo in attesa di un nostro input, decidiamo di porre fine alla vita di emacs con il comando Ctrl+C.

```
(kali㉿kali)-[~]  
$ fg emacs  
[1] - running  emacs  
^C
```



17) Verifica quanto spazio si sta occupando su disco

Controlliamo la memoria occupato tramite il comando df -h.

```
(kali@kali)-[~]  
$ df -h  
Filesystem      Size  Used Avail Use% Mounted on  
udev            2.4G   0    2.4G   0% /dev  
tmpfs           499M  960K  498M   1% /run  
/dev/sda1       79G   17G   58G  23% /  
tmpfs           2.5G   0    2.5G   0% /dev/shm  
tmpfs           5.0M   0    5.0M   0% /run/lock  
tmpfs           499M  76K  499M   1% /run/user/1000
```

