

LABORATORIO 11 FEBBRAIO 2025 S10-L2

Permessi di Linux

Esercizio di oggi:

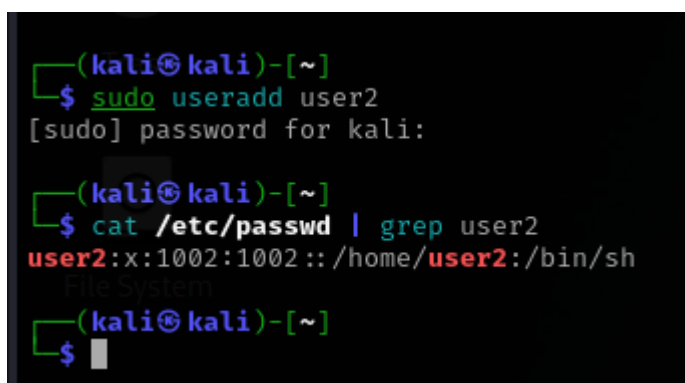
Gestione dei permessi di lettura, scrittura ed esecuzione in Linux.

Abbiamo visto come si gestiscono i permessi in Linux.

Obiettivo: Configurare e gestire i permessi di lettura, scrittura ed esecuzione per file o directory in un sistema Linux. La scelta dei file o delle directory da configurare spetta allo studente.

Infine, lo studente dovrà creare degli screenshot che mostrino i passaggi effettuati e scrivere una relazione spiegando le scelte fatte riguardo ai permessi.

Inizialmente procedo con la creazione di un nuovo utente con il comando **sudo useradd user2**.



```
(kali㉿kali)-[~]  
$ sudo useradd user2  
[sudo] password for kali:  
  
(kali㉿kali)-[~]  
$ cat /etc/passwd | grep user2  
user2:x:1002:1002::/home/user2:/bin/sh  
File System  
(kali㉿kali)-[~]  
$
```

Consegna:

1. Screenshot della Creazione del File o della Directory:

Fornire uno screenshot che mostri i comandi utilizzati per creare il file o la directory.

Procediamo dunque con la creazione di una nuova directory che chiamerò **archivio_lavoro** e al cui interno inserirò due nuovi file chiamati rispettivamente **dati_archivio1** e **dati_archivio2**.

```
(kali㉿kali)-[~]
$ mkdir archivio_lavoro

(kali㉿kali)-[~]
$ ls
1234.py      '#2912c.c#'
2912         2912c.c
2912b       3012b.py
2912b.c     3012c.py
2912.c      a.out
2912c       archivio_lavoro
```

```
(kali㉿kali)-[~]
$ cd archivio_lavoro
Home

(kali㉿kali)-[~/archivio_lavoro]
$ touch dati_archivio1

(kali㉿kali)-[~/archivio_lavoro]
$ touch dati_archivio2

(kali㉿kali)-[~/archivio_lavoro]
$ ls
dati_archivio1  dati_archivio2

(kali㉿kali)-[~/archivio_lavoro]
$
```

2. Screenshot della Verifica dei Permessi Attuali:

Fornisci uno screenshot che mostri i comandi `ls -l` e l'output prima della modifica dei permessi.

Controlliamo i permessi relativi alla directory e ai file creati.

```

(kali㉿kali)-[~]
$ ls -l
total 145456
-rw-rw-r-- 1 kali kali      1119 Dec 17 04:21 1234.py
-rwxrwxr-x 1 kali kali    17208 Dec 29 06:50 2912
-rwxrwxr-x 1 kali kali    17208 Dec 29 06:50 2912b
-rw-rw-r-- 1 kali kali      114 Dec 29 06:50 2912b.c
-rw-rw-r-- 1 kali kali      401 Dec 29 05:57 2912.c
-rwxrwxr-x 1 kali kali    17336 Dec 29 09:37 2912c
-rw-rw-r-- 1 kali kali      182 Dec 29 07:10 '#2912c.c#'
-rw-rw-r-- 1 kali kali      325 Dec 29 09:37 2912c.c
-rw-rw-r-- 1 kali kali      451 Dec 30 10:50 3012b.py
-rw-rw-r-- 1 kali kali      591 Dec 30 11:14 3012c.py
-rwxrwxr-x 1 kali kali    15960 Dec  3 09:51 a.out
drwxrwxr-x 2 kali kali     4096 Feb 11 09:18 archivio_lavoro
-rw-rw-r-- 1 kali kali     1603 Dec 19 04:08 bonus1234.py
-rw-rw-r-- 1 root root    58155 Dec 19 05:58 bonus_1.pcap

```

In output per quanto riguarda la directory **archivio_lavoro** vediamo attivi tutti i permessi.

d ci indica che è una directory

rwxrwxr-x sono i permessi assegnati e nello specifico:

rw (proprietario kali): lettura (r), scrittura (w) ed esecuzione (x).

rw (gruppo kali): lettura (r), scrittura (w) ed esecuzione (x).

r-x (altri utenti): lettura (r), nessuna scrittura (-), esecuzione (x).

```

(kali㉿kali)-[~]
$ cd archivio_lavoro

(kali㉿kali)-[~/archivio_lavoro]
$ ls -l
total 0
-rw-rw-r-- 1 kali kali 0 Feb 11 09:18 dati_archivio1
-rw-rw-r-- 1 kali kali 0 Feb 11 09:18 dati_archivio2

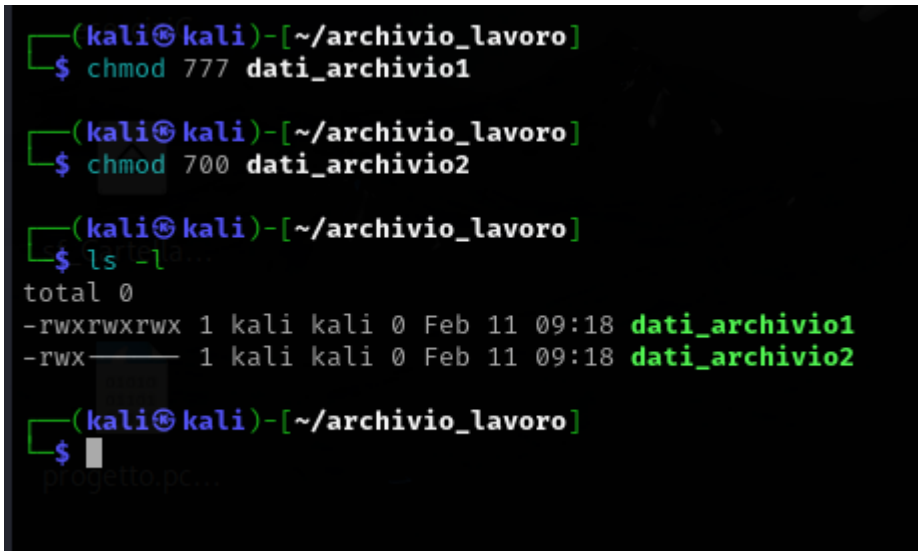
(kali㉿kali)-[~/archivio_lavoro]
$ █

```

I file invece hanno solo i permessi di lettura e scrittura per il proprietario kali e il gruppo kali e solo il permesso di lettura per gli altri utenti.

3. Screenshot della Modifica dei Permessi:

Fornisci uno screenshot che mostri i comandi `chmod` utilizzati e l'output successivo con `ls -l`.

A screenshot of a terminal window with a black background and green text. The prompt is (kali@kali)-[~/archivio_lavoro]. The first command is \$ chmod 777 dati_archivio1. The second command is \$ chmod 700 dati_archivio2. The third command is \$ ls -l, which produces the following output:
total 0
-rwxrwxrwx 1 kali kali 0 Feb 11 09:18 dati_archivio1
-rwx----- 1 kali kali 0 Feb 11 09:18 dati_archivio2
The prompt is then \$ and the cursor is on a new line.

```
(kali@kali)-[~/archivio_lavoro]
$ chmod 777 dati_archivio1

(kali@kali)-[~/archivio_lavoro]
$ chmod 700 dati_archivio2

(kali@kali)-[~/archivio_lavoro]
$ ls -l
total 0
-rwxrwxrwx 1 kali kali 0 Feb 11 09:18 dati_archivio1
-rwx----- 1 kali kali 0 Feb 11 09:18 dati_archivio2

(kali@kali)-[~/archivio_lavoro]
$
```

Con il comando **`chmod 777 dati_archivio1`** assegniamo tutti i permessi al proprietario, tutti i permessi al gruppo e tutti i permessi agli altri utenti.

Con il comando **`chmod 700 dati_archivio2`** invece assegniamo per questo file i permessi solo al proprietario mentre non avranno nessun permesso ne' il gruppo ne' gli altri utenti.

4. Screenshot del Test dei Permessi:

Fornisci uno screenshot che mostri i tentativi di scrivere nel file o di creare un nuovo file nella directory, insieme ai comandi e agli output.

```
(kali㉿kali)-[~/archivio_lavoro]
$ cd ..

(kali㉿kali)-[~]
$ whoami
kali

(kali㉿kali)-[~]
$ ls
1234.py      '#2912c.c#'      bonus1234.py      ciao.c      eserciziC
2912         2912c.c          bonus_1.pcap      Desktop     esercizio1.
2912b       3012b.py         bonus2.pcap      Documents   esercizio1.
2912b.c     3012c.py         bonus.pcap       Downloads   esercizioPa
2912.c      a.out            bonustcp.pcap    encdec.py   esercizioPr
2912c     ME  archivio_lavoro  bonus_tcp_udp.pcap encdec.py~   esercizioPr

(kali㉿kali)-[~]
$ cd archivio_lavoro

(kali㉿kali)-[~/archivio_lavoro]
$ ls
dati_archivio1  dati_archivio2

(kali㉿kali)-[~/archivio_lavoro]
$ cat dati_archivio1
DATI CHE POSSONO LEGGERE TUTTI I DIPENDENTI!

(kali㉿kali)-[~/archivio_lavoro]
$ cat dati_archivio2
DATI IMPORTANTISSIMI DEL CEO!!!

(kali㉿kali)-[~/archivio_lavoro]
$
```

```
(kali㉿kali)-[~/archivio_lavoro]
$ nano dati_archivio1

(kali㉿kali)-[~/archivio_lavoro]
$ nano dati_archivio2

(kali㉿kali)-[~/archivio_lavoro]
$ cat dati_archivio1 dati_archivio2
DATI CHE POSSONO LEGGERE TUTTI I DIPENDENTI!

sono il capo e aggiugno ciò che voglio!
DATI IMPORTANTISSIMI DEL CEO!!!

sono il capo e scrivo ciò che voglio!

(kali㉿kali)-[~/archivio_lavoro]
$
```

Come possiamo vedere in quanto utente principale (kali) abbiamo tutti i permessi e possiamo leggere, scrivere, eseguire e modificare i file a nostro piacimento.

Adesso cambiamo utente e dimostriamo che non possiamo fare lo stesso con **user2**.

Quando andiamo ad effettuare il cambio utente ci chiede di inserire la password. Non essendo stata inserita probabilmente ne fornisce una di default. Procediamo quindi con la creazione di una nuova password.

```
(kali㉿kali)-[~]  
$ sudo passwd user2  
[sudo] password for kali:  
New password:  
Retype new password:  
passwd: password updated successfully  
  
(kali㉿kali)-[~]  
$
```

```
(kali㉿kali)-[~]  
$ su - user2  
Password:  
su: warning: cannot change directory to /t  
$ whoami  
user2  
$
```

```
$ ls -l  
total 16  
drwxrwxrwx  2 kali      kali      4096 Feb 11 10:13 archivio_lavoro  
drwx----- 26 kali      kali      4096 Feb 11 11:35 kali  
drwx-----  5 test_user test_user 4096 Jan 17 06:26 test_user  
drwx-----  5 user2     user2     4096 Feb 11 11:32 user2  
$ cd archivio_lavoro  
$ pwd  
/home/archivio_lavoro  
$ ls  
dati_archivio1  dati_archivio2  
$ ls -l  
total 8  
-rwxrwxrwx 1 kali kali 87 Feb 11 10:13 dati_archivio1  
-rwx----- 1 kali kali 73 Feb 11 10:13 dati_archivio2  
$ cat dati_archivio2  
cat: dati_archivio2: Permission denied  
$ cat dati_archivio1  
DATI CHE POSSONO LEGGERE TUTTI I DIPENDENTI!  
  
sono il capo e aggiugno ciò che voglio!  
$
```

Possiamo vedere che l' **user2** ha i permessi per vedere e modificare il file **dati_archivio1** e invece non può leggere scrivere o modificare il file **dati_archivio**.

Relazione:

Scrivi una relazione spiegando le scelte fatte riguardo ai permessi configurati.

La relazione deve includere:

La motivazione delle scelte fatte per i permessi di lettura, scrittura ed esecuzione.

Un'analisi dei risultati ottenuti durante i test dei permessi.

Ho effettuato questa scelta sui permessi perché ho immaginato uno scenario in cui il CEO di un'azienda ha bisogno di scrivere, modificare o salvare dei dati all'interno di un file system condiviso con altri dipendenti ma allo stesso tempo i suoi file non devono essere visti o modificabili da questi ultimi.