

PRATICA - S10/L2

Nell'esercizio di oggi dobbiamo configurare e gestire i permessi di lettura, scrittura ed esecuzione per file o directory in un sistema Linux.

- Avvio Kali Linux e apro un terminale per andare a creare un file con **touch topsecret.txt** e con l'editor **nano** ci scrivo dentro del testo:

```
(kali㉿kali)-[~/Desktop]
$ touch topsecret.txt

(kali㉿kali)-[~/Desktop]
$ ls
DirBusterReport-192.168.56.101-80.txt  shell2.php  shell3.php  shell.php  topsecret.txt
```

- Ora, con **ls -l** vado a vedere quali sono i permessi attuali:

```
(kali㉿kali)-[~/Desktop]
$ ls -l
total 16
-rw-rw-r-- 1 kali kali 568 Jan 28 07:17 DirBusterReport-192.168.56.101-80.txt
-rw-rw-r-- 1 kali kali 1883 Jan 13 10:41 shell2.php
-rw-rw-r-- 1 kali kali 2292 Jan 13 10:53 shell3.php
-rw-rw-r-- 1 kali kali 35 Jan 13 08:32 shell.php
-rw-rw-r-- 1 kali kali 0 Feb 11 09:15 topsecret.txt
```

- Decido di concedere permessi di lettura, scrittura ed esecuzione al proprietario e solo lettura agli altri, quindi mando **chmod 744 topsecret.txt**, poi con **ls -l** vado di nuovo a verificare i permessi:

```
(kali㉿kali)-[~/Desktop]
└─$ ls -l
total 16
-rw-rw-r-- 1 kali kali 568 Jan 28 07:17 DirBusterReport-192.168.56.101-80.txt
-rw-rw-r-- 1 kali kali 1883 Jan 13 10:41 shell2.php
-rw-rw-r-- 1 kali kali 2292 Jan 13 10:53 shell3.php
-rw-rw-r-- 1 kali kali 35 Jan 13 08:32 shell.php
-rwxr--r-- 1 kali kali 0 Feb 11 09:15 topsecret.txt
```

- Notare che il comando **chmod 744 topsecret.txt** è un'abbreviazione del comando **chmod u=rwx,g=r,o=r topsecret.txt** dove:
 - **u=rwx** indica che il proprietario può leggere scrivere ed eseguire
 - **g=r** indica che il gruppo può solo leggere
 - **o=r** indica che tutti gli altri possono solo leggere.
- Con **mv** sposto il file creato in una cartella condivisa accessibile anche agli altri utenti (**tmp**):

```
(kali㉿kali)-[~/Desktop]
└─$ mv /home/kali/Desktop/topsecret.txt /tmp/

(kali㉿kali)-[~/Desktop]
└─$ ls
DirBusterReport-192.168.56.101-80.txt  shell2.php  shell3.php  shell.php
```

- Per dimostrare la funzionalità dei permessi inseriti mi sposto su un utente diverso da Kali (o da root ovviamente) e scelgo **test_user**, che avevo creato in un test precedente.

- Per spostarmi uso **su - test_user** e inserisco la password per effettuare l'accesso:

```
(kali@kali)-[~/Desktop]
$ su - test_user
Password:
(test_user@kali)-[~]
$
```

- Ora che ho un'altro utente provo a leggere con **cat topsecret.txt** il file che abbiamo creato prima:

```
(test_user@kali)-[/tmp]
$ cat topsecret.txt
Questi sono documenti riservati di sola visualizzazione
```

- Ora invece vado a scrivere all'interno del file e mi aspetto che l'azione non riesca a causa dei permessi che ho impostato precedentemente:

```
(test_user@kali)-[/tmp]
$ echo "prova" > topsecret.txt
-bash: topsecret.txt: Permission denied
```

- Come volevo dimostrare il permesso viene negato in quanto solo l'utente creatore della cartella ovvero Kali e l'utente root possono modificare il contenuto del file.
- Un'applicazione interessante di questo permesso può essere ad esempio rendere accessibile il file alla lettura da parte di tutti i dipendenti di un reparto ma non rendere possibile la modifica, o almeno dover richiedere personalmente l'attuazione di una modifica a un superiore che ha accesso alla scrittura del file.

