

PERMESSI LINUX

Introduzione

Capita spesso che un computer sia utilizzato da più utenti e quindi il sistema operativo deve mettere a disposizione un meccanismo per mantenere separati i dati di più utenti.

Linux identifica gli utenti come «numeri»: ciascun utente ha un proprio numero univoco detto «uid» e associa ad ogni uid una home directory. Configura poi i permessi (le autorizzazioni) sulle home directory relative allo specifico uid in modo tale che uid diversi da quello specificato non siano in grado di accedere.

Una caratteristica fondamentale del file system Linux sono i **permessi**, le autorizzazioni che vengono assegnate ai file. I file in Linux riportano le autorizzazioni per l'utente corrente «u», il gruppo «g» e gli altri utenti «o» come nella figura a fianco.

Abbiamo tre tipi di permesso: *Lettura, Scrittura, Esecuzione*

Con il comando «ls -l» è possibile controllare i permessi sui file sia per l'utente attuale, sia per il gruppo, sia per gli altri utenti,

Svolgimento

L'obiettivo odierno era di configurare e gestire i permessi di lettura, scrittura ed esecuzione per file o directory in un sistema Linux.

```
(kali@kali)-[~]
└─$ mkdir /home/kali/OnePiece

(kali@kali)-[~]
└─$ touch /home/kali/Poneglyph.txt

(kali@kali)-[~]
└─$ mv /home/kali/Poneglyph.txt /home/kali/OnePiece/

(kali@kali)-[~]
└─$ ls -l
total 752
-rw-rw-r-- 1 kali kali 148465 Nov 19 06:22 40839
-rw-rw-r-- 1 kali kali 207 Nov 14 10:12 backdoor.elf
drwxr-xr-x 4 kali kali 4096 Nov 29 10:39 Desktop
drwxr-xr-x 2 kali kali 4096 Sep 29 16:14 Documents
drwxr-xr-x 2 kali kali 4096 Nov 29 08:31 Downloads
-rw-rw-r-- 1 kali kali 321 Oct 16 08:46 encdec.py
d-x-x-x-x 13 kali kali 4096 Oct 15 06:47 gameshell
-rwxrwxr-x 1 kali kali 230872 Oct 15 12:07 gameshell-save.sh
-rw-rw-r-- 1 kali kali 210298 Aug 29 08:44 gameshell.sh
-rw-rw-r-- 1 kali kali 96001 Nov 14 06:41 GxHCTEFj.jpeg
-rw-rw-r-- 1 kali kali 244 Nov 20 06:58 'hash bcrypt.txt'
-rw-rw-r-- 1 kali kali 244 Nov 20 06:59 hashbcrypt.txt
drwxr-xr-x 2 kali kali 4096 Sep 29 16:14 Music
drwxrwxr-x 2 kali kali 4096 Dec 3 09:58 OnePiece ←
drwxr-xr-x 2 kali kali 4096 Sep 29 16:14 Pictures ←
-rw-rw-r-- 1 kali kali 1704 Oct 16 08:27 private_key.pem
drwxr-xr-x 2 kali kali 4096 Sep 29 16:14 Public
-rw-rw-r-- 1 kali kali 451 Oct 16 08:28 public_key.pem
-rw-rw-r-- 1 kali kali 3204 Nov 19 05:13 PwnKit.c
drwxr-xr-x 2 kali kali 4096 Sep 29 16:14 Templates
-rw-rw-r-- 1 kali kali 33 Jun 10 2021 user.txt
drwxr-xr-x 2 kali kali 4096 Sep 29 16:14 Videos

(kali@kali)-[~]
└─$ cd OnePiece

(kali@kali)-[~/OnePiece]
└─$ ls -l
total 0
-rw-rw-r-- 1 kali kali 0 Dec 3 09:57 Poneglyph.txt
```

Ho inizialmente creato una directory con il comando “mkdir” e un file di testo (.txt) con il comando “touch”, che successivamente è stato spostato nella directory con “mv”.

Come si può notare appena creata la directory con il comando "ls -l" abbiamo potuto visionare i permessi attivi << drwxrwxr-x >> la D iniziale indica che si tratta di una directory mentre con lo "score", come possiamo notare nei permessi del file.txt, si tratta appunto di file.

r, w, x, sono i tre tipi di permessi citati precedentemente quando non è possibile eseguirne uno il campo viene sostituito dallo "score".

```
(kali㉿kali)-[~]
└─$ chmod u=w,g=r,o=r OnePiece
(kali㉿kali)-[~]
└─$ cat OnePiece
cat: OnePiece: Permission denied
(kali㉿kali)-[~]
└─$ cd OnePiece
cd: permission denied: OnePiece
(kali㉿kali)-[~]
└─$ chmod u=rwx,g=r,o=r OnePiece
(kali㉿kali)-[~]
└─$ cd OnePiece
(kali㉿kali)-[~/OnePiece]
└─$ ls -l
total 0
-rw-r--r-- 1 kali kali 0 Dec  3 09:57 Poneglyph.txt
```

Con il comando "chmod" sono andato a modificare i permessi iniziali, e come notiamo nell'esempio ho tolto il permesso di lettura ed esecuzione, così che la directory non fosse raggiungibile infatti mostrava in output la voce "permission denied".

Ristabiliti i permessi ho esplorato la directory e sono andato a modificare il file.txt e sono andato a inserire un testo come contenuto del file avendo il permesso (w).

```
(kali㉿kali)-[~/OnePiece]
└─$ chmod u=rwx,g=rw,o=r Poneglyph.txt
(kali㉿kali)-[~/OnePiece]
└─$ nano Poneglyph.txt
(kali㉿kali)-[~/OnePiece]
└─$ cat Poneglyph.txt
Diventerò re dei pirati!
(kali㉿kali)-[~/OnePiece]
└─$ chmod u=rX,g=r,o=r Poneglyph.txt
(kali㉿kali)-[~/OnePiece]
└─$ nano Poneglyph.txt
```

Sempre con lo stesso procedimento utilizzato per la directory sono andato a modificare i permessi per far sì che il contenuto del file fosse solo leggibile e non modificabile.

```
GNU nano 8.2 Poneglyph.txt *
Diventerò re dei pirati!
Black hat
[ Error writing Poneglyph.txt: Permission denied ]
```

Svolgimento

La gestione dei permessi è una componente cruciale per mantenere sicurezza e controllo in un sistema come Kali Linux. Comprendere e manipolare i permessi aiuta a garantire che solo utenti o gruppi autorizzati possano accedere, modificare o eseguire file e directory.