

Relazione Tecnica: Gestione dei Permessi in Linux

Profilo studente

Gabriel Giustinelli

epicode Cyber Security

classe CS0525

data 10/02/2026

1. Obiettivo

L'obiettivo dell'attività è configurare e gestire i permessi di lettura, scrittura ed esecuzione su un file o una directory, verificando l'efficacia delle restrizioni tramite test pratici.

2. Configurazione dell'Ambiente

Per questo esercizio ho deciso di operare su un file di testo contenente informazioni ipoteticamente sensibili.

- **Creazione della Directory e del File:** Ho creato una cartella denominata **laboratorio_permessi** e, al suo interno, un file chiamato **segreto.txt**.

```
Session Actions Edit View Help
zsh: corrupt history file /home/kali/.zsh_history
(kali@kali)-[~]
$ mkdir laboratorio_permessi
touch laboratorio_permessi/segreto.txt
```

- **Verifica Iniziale:** Ho utilizzato il comando **ls -l** per controllare i permessi assegnati automaticamente dal sistema al momento della creazione.

```
(kali@kali)-[~]
$ ls -l laboratorio_permessi/segreto.txt
-rw-rw-r-- 1 kali kali 0 Feb 10 16:51 laboratorio_permessi/segreto.txt
```

- **Di default:** notiamo che qualsiasi utente del sistema avrebbe potuto leggere il contenuto del file.

3. Modifica dei Permessi

Ho proceduto alla modifica dei permessi utilizzando il comando **chmod**.

- **Comando utilizzato:** **chmod 600 laboratorio_permessi/segreto.txt**.
- **scelta:** Nel comando **600** al proprietario ho assegnato permessi di scrittura e lettura (**6**) per permettere la gestione del file. Mentre ai gruppi e altri (**0**) ho rimosso ogni permesso(**0**) così da garantire la riservatezza del file, rispettando il principio del minimo privilegio.

- non ho assegnato il permesso di esecuzione dato che si tratta di un file di testo e non di uno script eseguibile

```
(kali㉿kali)-[~]  
$ chmod 600 laboratorio_permessi/segreto.txt  
ls -l laboratorio_permessi/segreto.txt  
-rw----- 1 kali kali 0 Feb 10 16:51 laboratorio_permessi/segreto.txt
```

4. Analisi dei Test effettuati

Per verificare che la configurazione fosse corretta, ho effettuato dei test di accesso.

- **Test di Accesso Negato:** Per simulare un tentativo di accesso non autorizzato, ho eseguito il comando `sudo -u nobody cat laboratorio_permessi/segreto.txt`.
- **Risultato:** Il sistema ha restituito l'errore "**Permission denied**". Con il comando `sudo -u nobody` eseguo l'operazione come utente **nobody**, che non ha alcun tipo di permesso sul file appena creato.

```
(kali㉿kali)-[~]  
$ sudo -u nobody cat laboratorio_permessi/segreto.txt  
[sudo] password for kali:  
cat: laboratorio_permessi/segreto.txt: Permission denied
```

5. Conclusioni

L'analisi dimostra come i permessi di linux sono molto importanti, soprattutto in aziende nella quale più persone usano lo stesso computer, aiutando la difesa di un sistema, non condividendo informazioni sensibili verso utenti non autorizzati, mantenendo il controllo completo nelle mani del proprietario del file.