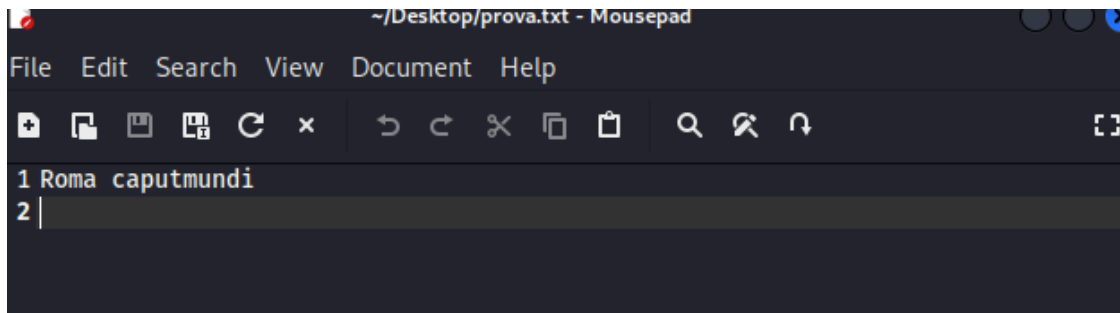


# Gestione dei Permessi di Lettura, Scrittura ed Esecuzione in Linux

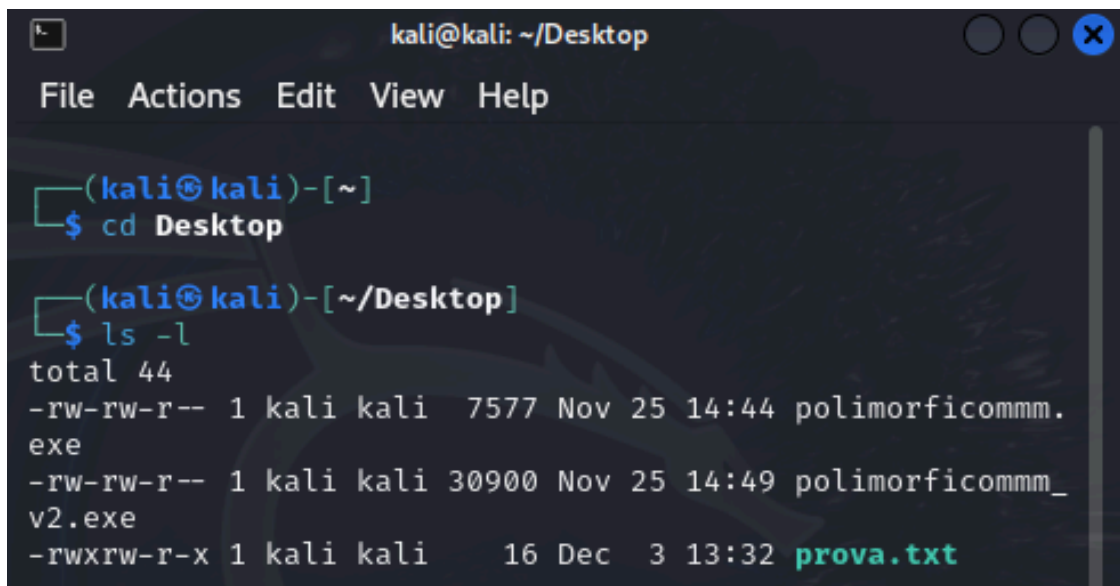
**Obiettivo:** Configurare e gestire i permessi di lettura, scrittura ed esecuzione per file o directory in un sistema Linux.

1° Creazione del file con Mousepad e relativo salvataggio come .txt .



```
~/Desktop/prova.txt - Mousepad
File Edit Search View Document Help
1 Roma caputmundi
2 |
```

2° Con il comando `cd` ci muoviamo all'interno delle directory fino a trovare il nostro file di nome **prova.txt** . Ora digitiamo **ls -l** per aprire i nostri file con i relativi permessi.



```
kali@kali: ~/Desktop
File Actions Edit View Help

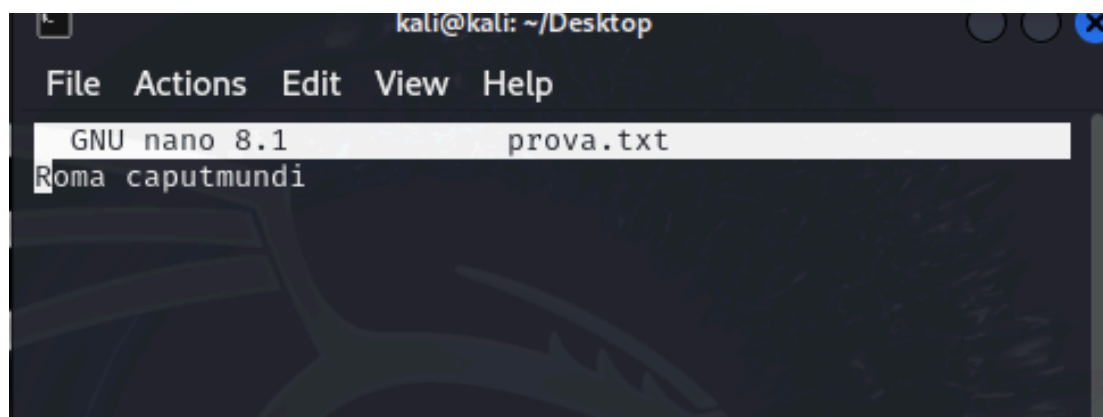
(kali@kali)-[~]
$ cd Desktop

(kali@kali)-[~/Desktop]
$ ls -l
total 44
-rw-rw-r-- 1 kali kali 7577 Nov 25 14:44 polimorficomm.exe
-rw-rw-r-- 1 kali kali 30900 Nov 25 14:49 polimorficomm_v2.exe
-rwxrw-r-x 1 kali kali 16 Dec 3 13:32 prova.txt
```

Come possiamo vedere l'user possiede tutti i comandi, ovvero read, write e execute

3° Digitando **nano prova.txt** potremo entrare all'interno del file creato e modificarlo.

```
(kali㉿kali)-[~]  
$ cd Desktop  
  
(kali㉿kali)-[~/Desktop]  
$ ls -l  
total 44  
-rw-rw-r-- 1 kali kali 7577 Nov 25 14:44 polimorficomm.exe  
-rw-rw-r-- 1 kali kali 30900 Nov 25 14:49 polimorficomm_v2.exe  
-rwxrw-r-x 1 kali kali 16 Dec 3 13:32 prova.txt  
  
(kali㉿kali)-[~/Desktop]  
$ nano prova.txt  
  
(kali㉿kali)-[~/Desktop]  
$  
  
(kali㉿kali)-[~/Desktop]  
$ nano prova.txt
```



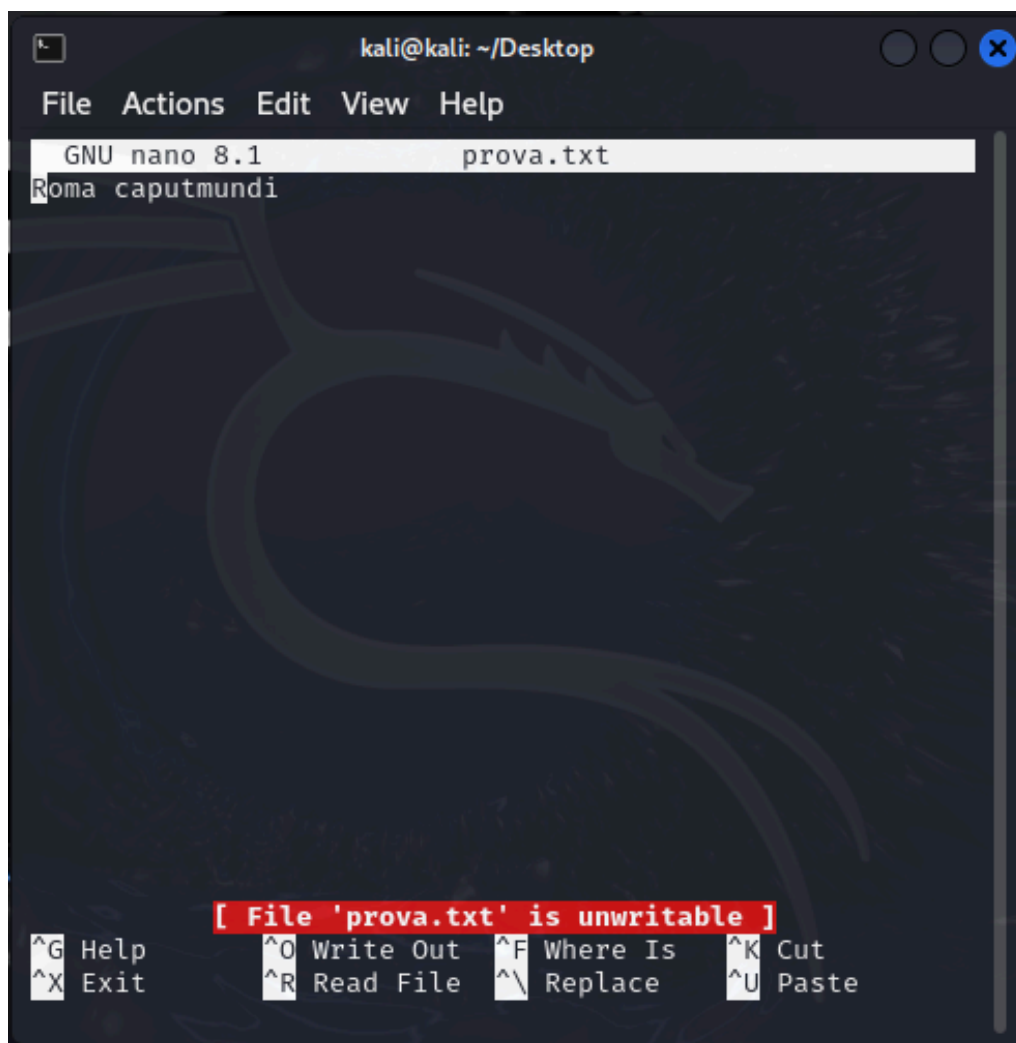
4° Ora modificheremo i parametri visti in precedenza e toglieremo all'user la possibilità di modificare il file con il comando **chmod u-w prova.tx**

```
(kali㉿kali)-[~/Desktop]
$ 
(kali㉿kali)-[~/Desktop]
$ nano prova.txt
(kali㉿kali)-[~/Desktop]
$ ls -l
total 44
-rw-rw-r-- 1 kali kali 7577 Nov 25 14:44 polimorficomm.exe
-rw-rw-r-- 1 kali kali 30900 Nov 25 14:49 polimorficomm_v2.exe
-rwxrw-r-x 1 kali kali 16 Dec 3 13:32 prova.txt
(kali㉿kali)-[~/Desktop]
$ chmod u-w prova.txt
```

Come possiamo veder dall'immagine la modifica è andata a buon fine

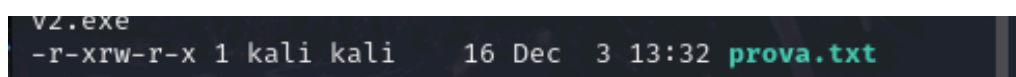
```
(kali㉿kali)-[~/Desktop]
$ chmod u-w prova.txt
(kali㉿kali)-[~/Desktop]
$ ls -l
total 44
-rw-rw-r-- 1 kali kali 7577 Nov 25 14:44 polimorficomm.exe
-rw-rw-r-- 1 kali kali 30900 Nov 25 14:49 polimorficomm_v2.exe
-r-xrw-r-x 1 kali kali 16 Dec 3 13:32 prova.txt
(kali㉿kali)-[~/Desktop]
$
```

5° Apriamo il file e controlliamo se il comando non è più disponibile



The screenshot shows a terminal window with the nano text editor open. The title bar indicates the user is 'kali' at 'kali' in the directory '~/Desktop'. The menu bar includes 'File', 'Actions', 'Edit', 'View', and 'Help'. The status bar at the top shows 'GNU nano 8.1' and the filename 'prova.txt'. The main text area contains the text 'Roma caputmundi'. A red error message is displayed in the center: '[ File 'prova.txt' is unwritable ]'. At the bottom, a help menu lists various shortcuts: ^G for Help, ^O for Write Out, ^F for Where Is, ^K for Cut, ^X for Exit, ^R for Read File, ^\ for Replace, and ^U for Paste. The background of the editor features a faint, stylized dragon logo.

Il comando inserito in precedenza ha disabilitato all 'user la scrittura del file



The screenshot shows a terminal window displaying the command 'v2.exe' and its output. The output shows the permissions for the file 'prova.txt': '-r-xrw-r-x 1 kali kali 16 Dec 3 13:32 prova.txt'. The permissions are displayed in green text.

6° Creiamo un nuovo user con privilegi di root. Digitando **sudo adduser nome\_utente** avremo il nostro nuovo user

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ sudo adduser Aluandr  
[sudo] password for kali:  
err: Please enter a username matching the regular expression  
configured via the NAME_REGEX configuration variable. Use the  
`--allow-bad-names' option to relax this check or reconfigure  
NAME_REGEX in configuration.
```

7° Inseriamo il comando `sudo usermod -aG sudo nome_utente` per aggiungere il nostro utente appena creato al gruppo sudo, conferendogli così i permessi necessari per eseguire comandi con privilegi di root. Questo non trasforma direttamente l'utente in "root", ma gli consente di acquisire privilegi elevati quando necessario, utilizzando il comando sudo.

```
Aluandr@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ sudo usermod -aG sudo Aluandr  
[sudo] password for kali:  
(kali@kali)-[~]  
$ su - Aluandr  
Password:  
(Aluandr@kali)-[~]  
$ sudo whoami  
[sudo] password for Aluandr:  
root  
(Aluandr@kali)-[~]  
$
```

## Motivazioni delle Scelte dei Permessi

La configurazione dei permessi di lettura, scrittura ed esecuzione sui file e directory è stata fatta seguendo i principi fondamentali della sicurezza informatica, con l'obiettivo di garantire:

- Protezione dei dati sensibili: Limitare l'accesso solo a utenti o gruppi autorizzati.
- Funzionalità minima necessaria: Fornire a ciascun utente i permessi strettamente necessari per eseguire le operazioni previste.
- Riduzione dei rischi di compromissione: Prevenire la possibilità di esecuzione non autorizzata o modifica di file critici.

- Permessi di lettura (r):

La lettura è stata concessa solo agli utenti che necessitano di accedere alle informazioni. Per i file di configurazione critici, ad esempio, è stato limitato il permesso al solo proprietario, escludendo il gruppo e altri utenti. Ciò riduce il rischio di divulgazione non autorizzata.

- Permessi di scrittura (w):

La scrittura è stata riservata esclusivamente al proprietario del file o al gruppo autorizzato. Questo impedisce modifiche accidentali o malevole ai file. Per i file pubblici o condivisi, la scrittura è stata limitata a specifici utenti del gruppo.

- Permessi di esecuzione (x):

I file eseguibili sono stati resi tali solo per gli utenti che necessitano di eseguirli. Per esempio, script utilizzati da servizi di sistema hanno il permesso di esecuzione solo per il proprietario e il gruppo associato, mentre gli altri utenti sono esclusi.

## **Analisi dei Risultati Durante i Test dei Permessi**

Durante la fase di test, sono stati eseguiti vari scenari per verificare la corretta implementazione dei permessi:

- Tentativi di accesso non autorizzato:

Gli utenti senza permessi di lettura o esecuzione non sono riusciti ad accedere o eseguire i file, dimostrando che le impostazioni di protezione sono efficaci.

- Operazioni con utenti autorizzati:

Gli utenti con permessi corretti hanno potuto leggere, modificare o eseguire i file senza problemi. Questo conferma che la configurazione non limita l'operatività.

- Test con utenti del gruppo:

I membri del gruppo autorizzato hanno potuto accedere alle risorse comuni, ma non è stato possibile per altri utenti esterni al gruppo.

- Controllo su directory condivise:

Le directory condivise configurate con permessi 770 hanno garantito che solo il proprietario e il gruppo autorizzato potessero accedere e modificare i contenuti, evitando accessi da parte di utenti esterni.

## **Conclusioni**

I test confermano che le scelte effettuate hanno rispettato i requisiti di sicurezza, garantendo l'accesso controllato e la protezione delle risorse. Le configurazioni attuali offrono un buon equilibrio tra sicurezza e funzionalità operativa. Tuttavia, si consiglia di:

- Effettuare regolarmente una revisione dei permessi per verificare la conformità con le esigenze in evoluzione.
- Implementare un sistema di monitoraggio che registri i tentativi di accesso non autorizzato per identificare eventuali minacce.