

# Lavorare con File di Testo nella CLI

## profilo studente

- Gabriel Giustinelli
- Eicode Cyber Security
- classe CS0525
- data 12/02/2026

## Progetto

In questo laboratorio lavoreremo con i file di testo da riga di comando Linux e i file di configurazione. per rispondere alle seguenti domande:

## Domande

### 1: Editor di Testo Grafici

**Sei riuscito a trovare subito space.txt?**

No, inizialmente il file non viene visualizzato perché SciTE cerca estensioni note e **.txt** non è tra queste.

**Perché il prompt non viene mostrato nel terminale (dopo aver lanciato SciTE)?**

Il prompt non viene mostrato perché il terminale è occupato a eseguire il processo di SciTE in primo piano. Finché SciTE non viene chiuso o il processo non viene interrotto il terminale non può accettare nuovi comandi.

### 2: Editor di Testo da Riga di Comando

**Quale carattere usa nano per rappresentare che una linea continua oltre i bordi dello schermo?**

Nano utilizza il carattere **\$** o **< , >** all'estremità destra o sinistra dello schermo per indicare che la riga di testo prosegue oltre il margine visibile.

### 3: Lavorare con i File di Configurazione

**Perché i file di configurazione delle applicazioni utente sono salvati nella directory home dell'utente e non sotto /etc con tutti gli altri file di configurazione a livello di sistema?**

Perché gli utenti regolari hanno permessi di scrittura solo nella propria directory home.

Permettendo di personalizzare le proprie applicazioni senza compromettere l'intero sistema. La directory `/etc` è riservata alle configurazioni a livello di sistema e può essere modificata solo dall'utente `root`.

### **Modificare e Salvare i File di Configurazione**

**Anche la finestra del terminale che era già aperta ha cambiato colore da verde a rosso? Spiega.**

No, la finestra già aperta non cambia colore.

Questo accade perché il file `.bashrc` viene letto ed eseguito dalla shell solo nel momento in cui viene avviata una nuova sessione.

Per vedere le modifiche in una sessione esistente, bisognerebbe ricaricare il file manualmente

### **Modificare i File di Configurazione per i Servizi**

**A cosa si riferisce il messaggio di errore?**

Il messaggio di errore si riferisce al fatto che `nginx` non riesce a trovare un file chiamato `favicon.ico`. Questo accade perché il browser cerca automaticamente l'icona del sito, ma poiché abbiamo cambiato la directory radice in `/usr/share/nginx/html/text_ed_lab/`, il server non trova quel file specifico in quella nuova posizione.

**Appare la pagina web?**

No, la pagina web non appare più. Al suo posto, il browser mostrerà un errore di "Connessione rifiutata" o "Impossibile connettersi", perché il comando `sudo pkill nginx` ha arrestato il servizio server web.

**Domanda Sfida: Puoi modificare il file `/etc/nginx/custom_server.conf` con SciTE?  
Descrivi il processo di seguito.**

Sì, è possibile farlo, ma è necessario utilizzare privilegi elevati perché il file si trova nella directory `/etc`, che è protetta.

**Processo:**

1. Apri il terminale.
2. Lancia SciTE come amministratore usando il comando: `sudo scite /etc/nginx/custom_server.conf`.
3. Inserisci la password dell'utente analyst `cyberops` quando richiesto.
4. Apporta le modifiche necessarie nell'interfaccia grafica di SciTE e salva il file.

# BONUS Prendere Familiarità con la Shell Linux

## Parte 1

### Passo 2

Elenca alcune sezioni incluse in una pagina man.

1. **NAME (NOME)**: Il nome del comando o della funzione, seguito da una brevissima descrizione di una riga (utile per capire subito cosa fa).
2. **SYNOPSIS (SINTASSI)**: Mostra come deve essere scritto il comando. Ti elenca tutti gli argomenti, le opzioni e i parametri che puoi usare (ad esempio: `cp [OPZIONE]... SORGENTE DEST`).
3. **DESCRIPTION (DESCRIZIONE)**: Spiega nel dettaglio cosa fa il comando e come funzionano le sue diverse opzioni.
4. **OPTIONS (OPZIONI)**: Un elenco dettagliato di tutti i "flag" (come `-r`, `-v`, `-l`) che puoi aggiungere al comando per modificarne il comportamento.
5. **EXAMPLES (ESEMPI)**: (Non sempre presente, ma molto utile) Mostra casi pratici di utilizzo del comando.
6. **EXIT STATUS (STATO DI USCITA)**: Indica i codici che il comando restituisce al sistema per dire se ha avuto successo o se c'è stato un errore.
7. **SEE ALSO (VEDI ANCHE)**: Suggerisce altri comandi correlati o file di configurazione pertinenti.

### Qual è la funzione del comando `cp`?

Il comando `cp` (abbreviazione di *copy*) ha la funzione di **copiare file e directory** da una posizione a un'altra all'interno del filesystem di Linux. crea un duplicato esatto mentre il file originale rimane dove si trova.

### Quale comando useresti per trovare maggiori informazioni sul comando `pwd`?

Il comando principale da utilizzare è: `man pwd` apre la guida ufficiale del sistema per qualsiasi comando specificato. In alternativa, puoi usare `pwd --help` per una descrizione più rapida e sintetica direttamente nel terminale.

### Qual è la funzione del comando `pwd`?

La funzione del comando `pwd` è **mostrare il percorso completo della directory (cartella) in cui ti trovi attualmente**.

### **Passo 3**

**Qual è la directory corrente?**

La directory corrente è **/home/analyst**

**In quale cartella ti trovi ora?**

La directory corrente è **/home/analyst/cyops\_folder3**

**Perché è successo?**

successo perché in Linux il carattere tilde (~) è una scorciatoia che rappresenta la **directory home** dell'utente corrente, digitando cd ~, il sistema riporta istantaneamente l'utente alla sua cartella personale , indipendentemente da dove si trovi nel filesystem.

**K. Digita cd . Cosa succede?**

il punto singolo (.) rappresenta la **directory corrente**. Pertanto, il comando cd .. dice al sistema di "cambiare directory verso la cartella in cui siamo già.

**I. Digita `cd ..` Cosa succede?**

Il sistema si sposta nella **directory genitore**, ovvero sale di un livello nell'albero delle cartelle. da **/home/analyst/cyops\_folder3**, ora siamo in **/home/analyst**

**Quale sarebbe la directory corrente se eseguissi il comando cd .. da [analyst@secOps \$?**

La directory corrente sarebbe **/home**. (Poiché ~ corrisponde a /home/analyst, salire di un livello ti porta in /home).

**Quale sarebbe la directory corrente se eseguissi il comando cd .. da [analyst@secOps home]\$?**

La directory corrente sarebbe la radice del sistema, indicata dal simbolo **/** (root).

**Quale sarebbe la directory corrente se eseguissi il comando cd .. da [analyst@secOps /\$?**

Rimane nella directory ( / ). La radice è il livello più alto del filesystem e non ha una directory "padre" sopra di sé.

**Passo 4:**  
**È previsto? Spiega.**

Sì, è previsto che non venga mostrato alcun output nel terminale. Quando si utilizza l'operatore di redirezione >, l'output del comando viene deviato dalla finestra del terminale e scritto direttamente all'interno del file specificato. Il terminale rimane vuoto proprio perché l'output ha una nuova destinazione.

**Cosa è successo al file di testo? Spiega.**

Il contenuto originale del file di testo è stato **sovrascritto**. Questo è successo perché è stato utilizzato l'operatore di redirezione singolo >. Questo comando cancella tutto il contenuto precedentemente esistente all'interno del file e lo sostituisce completamente con il nuovo output del comando

**Passo 5:**

**Cosa è successo al file di testo? spiega.**

Il nuovo messaggio è stato **accodato (aggiunto in coda)** al file senza cancellare quello che c'era prima. l'operatore doppio >> dice al sistema di mantenere intatto il contenuto esistente di **some\_text\_file.txt** e di inserire le nuove informazioni partendo dalla fine del file.

**Passo 6:**

**Quanti file vengono visualizzati?**

Viene visualizzato **solo 1 file**, il comando **ls -l** mostra solo i file "regolari".

**Quanti file in più vengono visualizzati rispetto a prima? Spiega.**

Vengono visualizzati **molti più file**. Il comando **ls -la** include l'opzione -a (*all*), che istruisce il sistema a mostrare tutti i file, compresi quelli che iniziano con un punto (.)

**È possibile nascondere intere directory aggiungendo un punto prima del loro nome?**

Sì. In Linux, qualsiasi elemento (file o cartella) il cui nome inizia con un punto viene trattato dal sistema come "nascosto"

**Ci sono directory nascoste nell'output di ls -la sopra?**

Sì. Oltre alle directory speciali `.` (corrente) e `..` (genitore), sono spesso presenti directory di configurazione come ad esempio `.cache`, `.local` o `.config`.

**Fornisci tre esempi di file nascosti mostrati nell'output di `ls -la` sopra.**

1. `.ssh`
2. `.bash_profile`
3. `.bashrc`