

# Esercizio\_S10\_L4\_Exploring\_Linux

## Consegna

### Obiettivi

In questo laboratorio, prenderai familiarità con gli editor di testo da riga di comando Linux e i file di configurazione.

- Parte 1 Editor di Testo Grafici
- Parte 2 Editor di Testo da Riga di Comando
- Parte 3 Lavorare con i File di Configurazione

### Risorse Richieste

- Macchina virtuale CyberOps Workstation

## Svolgimento

### Parte 1

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

Dopo aver premuto su "Open" ed essermi diretto sulla directory dove è stato salvato il file, è stato necessario cambiare tipo di visualizzazione da **All Souce** a **all Files (\*)**.

#### Perché il prompt non viene mostrato nel terminale?

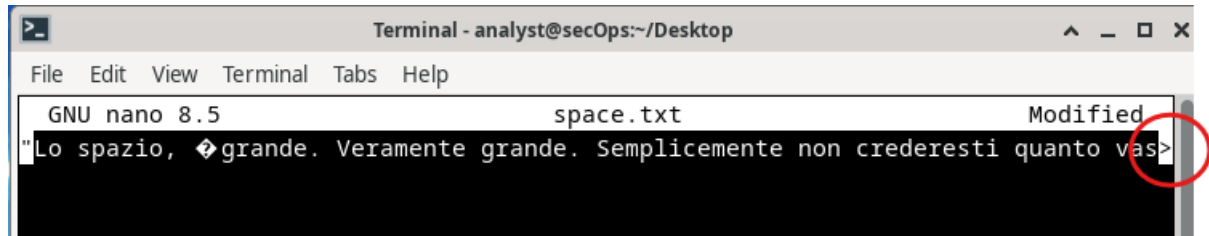
Perché il prompt ha dato origine ad un processo figlio, per l'appunto SciTe, e rimane impegnato su tale processo.

In sostanza, il prompt non viene mostrato perché la shell è ancora associata al processo figlio (SciTE) avviato in **foreground**.

## Parte 2

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

Il carattere utilizzato è >:



## Parte 3

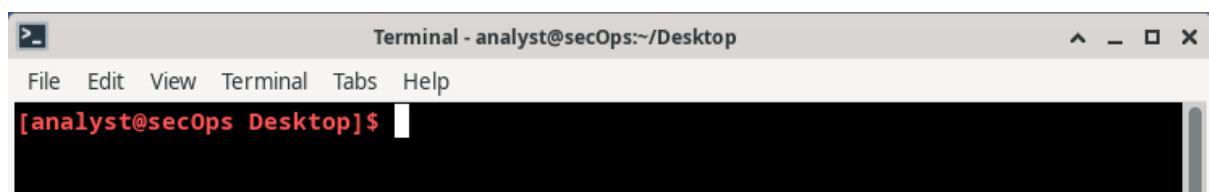
Perché i file di configurazione delle applicazioni utente sono salvati nella directory home dell'utente e non sotto /etc con tutti gli altri file di configurazione a livello di sistema?

I file di configurazione delle applicazioni utente risiedono nella home directory per motivi di personalizzazione, sicurezza e separazione dei ruoli; /etc rimane sotto il controllo dell'amministratore per impedire importanti modifiche di sistema, mentre ogni utente mantiene libertà sul proprio ambiente di lavoro.

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

La modifica non viene applicata automaticamente alle shell già aperte. Le shell esistenti hanno già caricato la loro configurazione al momento dell'avvio, quindi continueranno a mostrare il prompt con il vecchio colore (verde).

Aprendone invece una nuova possiamo appurare che la modifica è andata buon fine:



Ricaricando invece la CLI tramite comando `bash` potremmo notare eventuali modifiche avvenute nel file di configurazione del terminale:

```
[analyst@secOps ~]$ sudo nano .bashrc
[sudo] password for analyst:
[analyst@secOps ~]$ bash
[analyst@secOps ~]$
```

Dopo aver aperto con successo la homepage di nginx, guarda il messaggio di connessione nella finestra del terminale. A cosa si riferisce il messaggio di errore?

Personalmente non ho riscontrato alcun errore:

The screenshot shows a web browser window with the address bar set to `localhost:8080/`. The page displays a large "Congratulations!" heading, followed by the message: "As part of the Working with Text Files lab, you have successfully configured NGINX!". This message is circled in red. Below the browser window, a terminal window titled "Terminal - analyst@secOps:~" is open, showing a list of files and directories with their permissions, owner, group, size, and modification date. The terminal also shows the command history, including the commands used to edit the `.bashrc` file and start the nginx server.

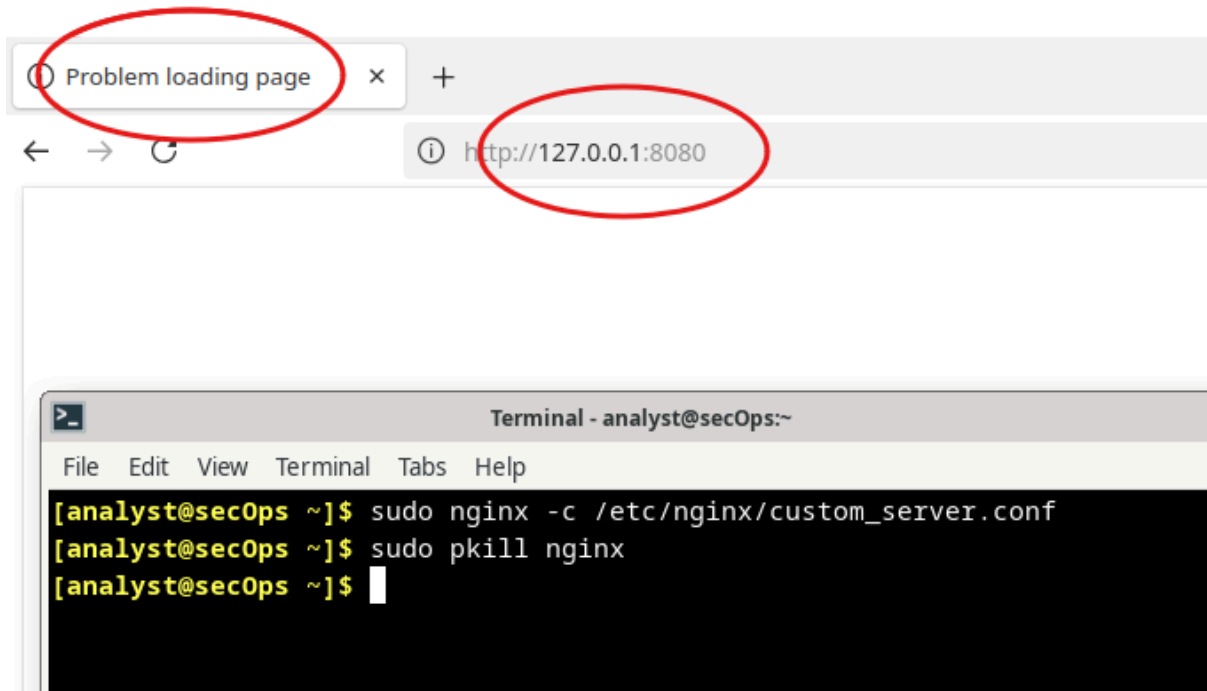
```
File Edit View Terminal Tabs Help
-rw-r----- 1 analyst analyst 4 Sep 25 07:49 .vboxclient-hostversion-tty7-co
ntrol.pid
-rw-r----- 1 analyst analyst 4 Sep 25 07:49 .vboxclient-seamless-tty7-contr
ol.pid
-rw-r----- 1 analyst analyst 4 Sep 25 07:49 .vboxclient-seamless-tty7-servi
ce.pid
-rw-r----- 1 analyst analyst 4 Sep 25 07:49 .vboxclient-vmtoolsd-session-tty7
-control.pid
drwxr-xr-x 3 analyst analyst 4096 Mar 20 2018 .vim
-rw-r----- 1 analyst analyst 13912 Jul 19 2018 .viminfo
-rw-r----- 1 analyst analyst 51 Sep 25 07:49 .Xauthority
-rw-r--r-- 1 analyst analyst 16 Mar 22 2018 .xinitrc
-rw-r--r-- 1 analyst analyst 16 Mar 22 2018 .Xinitrc
-rw-r----- 1 analyst analyst 635 Sep 25 07:49 .xsession-errors
-rw-r----- 1 analyst analyst 855 Sep 23 07:35 .xsession-errors.old
-rwxr-xr-x 1 analyst analyst 90 Jun 17 17:05 .xsessionrc
drwxr-xr-x 5 analyst analyst 4096 Jun 18 19:27 yay
[analyst@secOps ~]$ sudo nano .bashrc
[sudo] password for analyst:
[analyst@secOps ~]$ bash
[analyst@secOps ~]$ sudo nano -l /etc/nginx/custom_server.conf
[analyst@secOps ~]$ sudo nginx -c /etc/nginx/custom_server.conf
[analyst@secOps ~]$
```

Per arrestare il webserver nginx si può procedere digitando il seguente comando nella finestra del terminale:

```
sudo pkill nginx
```

Appare la pagina web?

Avendo interrotto il servizio del webserver, ovviamente la pagina web non verrà raggiunta:



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

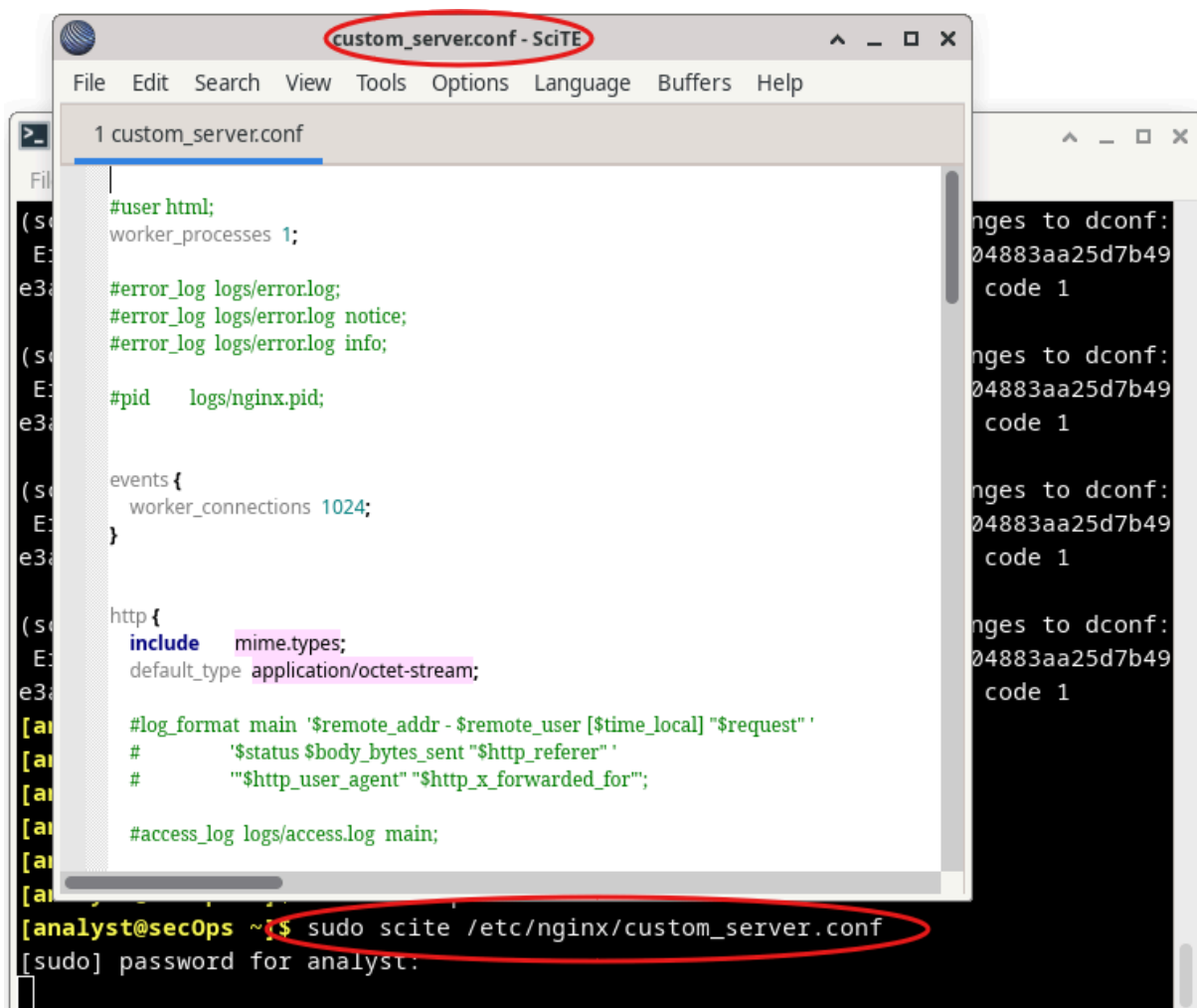
Dovendo accedere ad una cartella di sistema ho subito intuito di dover avviare SciTe tramite `sudo`; ciò però non è stato sufficiente in quanto, anche se lanciato con privilegi root, il processo non aveva comunque la possibilità di esplorare liberamente le directories:

```
(scite:1680): dconf-WARNING **: 09:10:02.789: failed to commit changes to dconf:  
Error spawning command line ?dbus-launch --autolaunch=ce0656a717e04883aa25d7b49  
e3a68e1 --binary-syntax --close-stderr?: Child process exited with code 1  
  
(scite:1680): dconf-WARNING **: 09:10:02.806: failed to commit changes to dconf:  
Error spawning command line ?dbus-launch --autolaunch=ce0656a717e04883aa25d7b49  
e3a68e1 --binary-syntax --close-stderr?: Child process exited with code 1  
  
(scite:1680): dconf-WARNING **: 09:10:03.506: failed to commit changes to dconf:  
Error spawning command line ?dbus-launch --autolaunch=ce0656a717e04883aa25d7b49  
e3a68e1 --binary-syntax --close-stderr?: Child process exited with code 1
```

A quel punto ho deciso di tentare di avviare direttamente SciTe aprendo il file interessato:

```
sudo scite /etc/nginx/custom_server.conf
```

A quel punto è stato possibile aprire il file senza problemi:



# BONUS Prendere Familiarità con la Shell Linux

Elenca alcune sezioni incluse in una pagina man.

## NAME

Contiene il nome del comando o della funzione e una breve descrizione.

*Esempio:* `pwd - print name of current/working directory`.

## SYNOPSIS

Mostra la sintassi d'uso del comando, indicando parametri e opzioni disponibili.

*Esempio:* `pwd [OPTION]....`

## DESCRIPTION

Spiega in dettaglio cosa fa il comando e come funziona.

## OPTIONS

Elenca le opzioni (con i relativi significati) che possono essere usate con il comando.

Qual è la funzione del comando `cp`?

Il comando `cp` serve a copiare file o directory in un'altra posizione. È utile per duplicare un file mantenendo l'originale intatto, oppure fare backup rapidi di directory; se non specificato diversamente, `cp` sovrascrive senza avviso i file di destinazione.

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

Per ottenere documentazione dettagliata sul comando `pwd`, i metodi principali sono:

`man pwd` → apre il manuale del comando con descrizione, sintassi e opzioni.

`info pwd` → mostra una guida in formato ipertestuale più estesa, se disponibile.

Qual è la funzione del comando `pwd`?

Il comando `pwd` (*print working directory*) stampa sul terminale il percorso della directory corrente. È utile per orientarsi nel filesystem e per verificare in quale cartella si sta lavorando.

Qual è la directory corrente?

La directory corrente è `/home/analyst`.

Digita `cd /home/analyst/cyops_folder3`; In quale cartella ti trovi ora?

Mi trovo ovviamente all'interno di `/home/analyst/cyops_folder3`.

Sfida: Digita il comando `cd ~` e descrivi cosa succede. Perché è successo?

`cd` è il comando usato per cambiare directory. L'argomento `~` (tilde) ha come significato il rappresentare la **home directory dell'utente corrente**.

Mi ritroverò dunque nuovamente all'interno di `/home/analyst`.

Digita `cd .` Cosa succede?

Il punto `.` è un **alias simbolico** che indica la cartella in cui ci si trova già.

Il comando è dunque valido ma non accadrà nulla; rimarremo nella cartella corrente.

Digita ora `cd ..` cosa succede?

Il doppio punto `..` indica la directory superiore rispetto a quella corrente. Usare `cd ..` mi sposterà quindi di **un livello verso l'alto** nell'albero delle directory. Tornerò quindi alla directory nella quale è contenuta la cartella corrente.

Quale sarebbe la directory corrente se eseguiessi il comando `cd ..` da `[analyst@secOps $]`?

`/home/analyst` → `/home`

Quale sarebbe la directory corrente se eseguiessi il comando `cd ..` da `[analyst@secOps home]$`?

`/home` → `/`

Quale sarebbe la directory corrente se eseguiessi il comando `cd ..` da `[analyst@secOps /]$`?

Non essendoci altro oltre alla cartella `root`, rimarrei all'interno di `/`

echo Questo è un messaggio inviato al terminale da echo. > some\_text\_file.txt  
Nessun output è stato mostrato. È previsto? Spiega.

Sì, tutto è normale in quanto la stringa che segue echo viene memorizzata all'interno di un file denominato some\_text\_file.txt tramite >

Usa di nuovo l'operatore > per redirigere un diverso output di echo al file di testo some\_text\_file.txt. Cosa è successo al file di testo? Spiega.

E' stato generato un nuovo file di testo con lo stesso nome del precedente che, esistendo già, viene sovrascritto.

echo Questa è un'altra riga di testo. Sarà ACCODATA al file di output. >>  
some\_text\_file.txt Cosa è successo al file di testo? Spiega.

In questo caso, tramite >> non viene generato un nuovo file ma la stringa viene aggiunta in seguito a quanto già pre-esistente all'interno del file target.  
(append)

Usa ls -l per visualizzare i file memorizzati nella directory home dell'utente analyst. Quanti file vengono visualizzati?

I file visualizzati saranno gli stessi visualizzati con un semplice ls; la differenza sta nel fatto che verranno mostrate, per ciascun file presente, ulteriori informazioni circa permessi owner, gruppi ecc..

Usa ora ls -la; quanti file in più vengono visualizzati rispetto a prima? Spiega.

Utilizzando ls -la verranno visualizzati in aggiunta anche i file e le cartelle nascosti presenti all'interno della directory.

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

Sì. Anche intere directory diventano nascoste se precedute da un punto.

Ci sono directory nascoste nell'output di ls -la sopra? Fornisci tre esempi di file nascosti mostrati nell'output di `ls -la` sopra.

.gnupg  
.local  
.ssh

```
drwxr-xr-x 1 analyst analyst 4096 Feb  7 2018 .bashrc_stock
drwxr-xr-x 8 analyst analyst 4096 Sep 23 07:45 .cache
-rw-r--r-- 1 root root 6637 Sep 23 07:55 capture.pcap
drwxr-xr-x 12 analyst analyst 4096 Sep 23 07:58 .config
drwxr-xr-x 2 analyst analyst 4096 Sep 25 08:49 Desktop
-rw-r--r-- 1 analyst analyst 23 Mar 23 2018 .dmrc
drwxr-xr-x 3 analyst analyst 4096 Jun 18 20:17 Downloads
drwx----- 4 analyst analyst 4096 Jun 19 03:00 .gnupg
-rw----- 1 analyst analyst 2520 Jun 17 07:35 .ICEauthority
drwxr-xr-x 2 analyst analyst 4096 Mar 24 2018 .idlcrc
drwxr-xr-x 9 analyst analyst 4096 Jun 18 20:17 lab.support.files
-rw----- 1 analyst analyst 51 Jun 18 19:58 .lessht
drwxr-xr-x 3 analyst analyst 4096 Mar 22 2018 .local
drwx----- 5 analyst analyst 4096 Mar 24 2018 .mozilla
drwxr-xr-x 3 analyst analyst 4096 Jun 18 19:55 scripts
drwxr-xr-x 2 analyst analyst 4096 Mar 21 2018 second_drive
drwx----- 2 analyst analyst 4096 Apr  2 2018 .ssh
-rw-r----- 1 analyst analyst 4 Sep 25 07:49 .vboxclient-clipboard-t
```



```
cp some_text_file.txt cyops_folder2/
```

Identifica i parametri nel comando cp sopra. Quali sono i file sorgente e destinazione? (usa percorsi completi per rappresentare i parametri)

File sorgente: `/home/analyst/some_text_file.txt`

Directory di destinazione: `/home/analyst/cyops_folder2/`

Spostare `some_text_file.txt` da `/home/analyst/cyops_folder2` di nuovo alla directory home. Quale comando hai usato per completare l'attività?

```
mv some_text_file.txt cyops_folder2/some_text_file.txt .
```

Riflessione: Quali sono i vantaggi dell'utilizzo della riga di comando Linux?

La riga di comando Linux offre un controllo molto più preciso e potente rispetto alla GUI: permette di automatizzare compiti con script, lavorare in modo rapido ed efficiente, combinare comandi per operazioni complesse e gestire sistemi da remoto tramite SSH. È leggera sulle risorse e dà accesso a funzioni avanzate spesso non disponibili in interfaccia grafica. Per questo è uno strumento fondamentale per amministratori e professionisti della sicurezza.