

S11- L4

Lavorare con File di Testo nella CLI

Obiettivi:

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

Risorse: VM CyberOps Workstation

PARTE 1 — Editor di Testo Grafici

Passo 1 – Aprire SciTE dalla GUI

Azione

- Accedo alla VM come utente **analyst**
- Applications → CyberOPS → SciTE
- Inserisco il testo richiesto
- Salvo come **space.txt** nella directory **/home/analyst**
- Chiudo e riapro SciTE
- File → Open ...

Domanda:

“Sei riuscito a trovare subito space.txt?”

Risposta:

No. Perché SciTE applica un filtro sulle estensioni e il file **.txt** non viene mostrato finché non si seleziona “All Files (*)”.

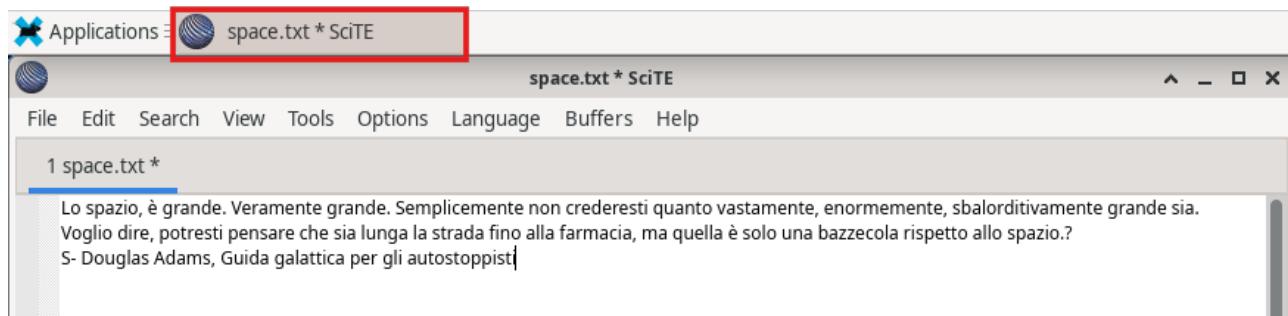
Passo 2 – Aprire SciTE dal Terminale

Azione

Nel terminale eseguo:

```
ls
```

```
scite space.txt
```



Output: SciTE avviato dal terminale con file space . txt aperto in primo piano.

Domanda:

“Perché il prompt non viene mostrato nel terminale?”

Risposta:

Perché SciTE viene eseguito in primo piano. Il terminale resta occupato dal processo finché non viene chiuso o interrotto.

PARTE 2 — Editor da Riga di Comando

Apertura con nano

Nel terminale:

```
nano space.txt
```

Osservo il contenuto del file e la barra dei comandi in basso.

Domanda:

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

Risposta:

Il carattere > viene utilizzato per indicare che la riga continua oltre i bordi dello schermo.

```
GNU nano 8.5 space.txt
Lo spazio, ♦ grande. Veramente grande. Semplicemente non crederesti quanto vastamente, enormemente, >
Voglio dire, potresti pensare che sia lunga la strada fino alla farmacia, ma quella ♦ solo una bazz< >
S- Douglas Adams, Guida galattica per gli autostoppisti
```

Output: Nano aperto con visualizzazione della riga che supera il bordo dello schermo indicata dal simbolo >.

PARTE 3 — Lavorare con i File di Configurazione

Visualizzare file nascosti

Nel terminale:

```
ls -la
```

Verifico la presenza del file **.bashrc**.

```
[analyst@secOps ~]$ ls -la
total 196
drwx----- 16 analyst analyst 4096 Feb 19 08:57 .
drwxr-xr-x  3 root    root    4096 Mar 20  2018 ..
-rw-------  1 analyst analyst   195 Feb 17 10:12 .bash_history
-rw-r--r--  1 analyst analyst    21 Feb  7 2018 .bash_logout
-rw-r--r--  1 analyst analyst    57 Feb  7 2018 .bash_profile
-rw-r--r--  1 analyst analyst   97 Mar 20  2018 .bashrc
-rw-r--r--  1 analyst analyst   141 Feb  7 2018 .bashrc_stock
drwxr-xr-x  8 analyst analyst 4096 Feb 17 09:23 .cache
-rw-r--r--  1 root    root     24 Feb 17 10:11 capture2.pcap
-rw-r--r--  1 root    root      0 Feb 17 10:33 capture3.pcap
-rw-r--r--  1 root    root    5113 Feb 17 10:27 capture.pcap
drwxr-xr-x 12 analyst analyst 4096 Feb 17 10:28 .config
drwxr-xr-x  3 analyst analyst 4096 Feb 17 09:25 Desktop
-rw-r--r--  1 analyst analyst   23 Mar 23  2018 .dmrc
drwxr-xr-x  3 analyst analyst 4096 Jun 18  2025 Downloads
drwx-----  4 analyst analyst 4096 Jun 19  2025 .gnupg
-rw-------  1 analyst analyst 2520 Jun 17  2025 .ICEauthority
drwxr-xr-x  2 analyst analyst 4096 Mar 24  2018 .idlerc
drwxr-xr-x  9 analyst analyst 4096 Jun 18  2025 lab.support.files
-rw-------  1 analyst analyst   51 Jun 18  2025 .lessshst
drwxr-xr-x  3 analyst analyst 4096 Mar 22  2018 .local
drwx-----  5 analyst analyst 4096 Mar 24  2018 .mozilla
```

Output: Visualizzazione dei file nascosti con **ls -la**, evidenziando la presenza del file di configurazione **.bashrc** nella home dell'utente.

Visualizzare il file .bashrc

Nel terminale:

cat .bashrc

Individuo la variabile PS1 con codice colore.

```
[analyst@secOps ~]$ cat .bashrc
export EDITOR=vi

PS1='[\e[1;32m]\u@\h \W]\$[\e[0m] '
alias ls="ls --color"
alias vi="vim"
```

Output: Visualizzazione del contenuto del file .bashrc tramite comando cat, con evidenza della variabile PS1 che definisce il colore del prompt.

Domanda:

“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?”

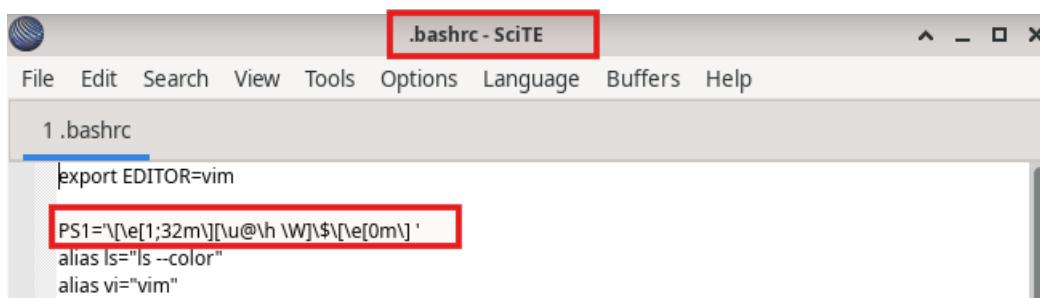
Risposta:

Perché sono configurazioni personali modificabili senza privilegi root. I file sotto /etc sono globali e richiedono permessi amministrativi.

Modifica del file .bashrc con editor grafico

Azione

- Apro .bashrc
- Modifico il valore colore da 32 a 31
- Salvo il file
- Apro un nuovo terminale per verificare il cambio colore



The screenshot shows the SciTE editor window with the title bar ".bashrc - SciTE". The menu bar includes File, Edit, Search, View, Tools, Options, Language, Buffers, and Help. A single buffer titled "1 .bashrc" is open, containing the following code:

```
export EDITOR=vim
PS1='\[\e[1;31m\]\u@\h \W\$\[\e[0m\] '
alias ls="ls --color"
alias vi="vim"
```

Output 1 e 2: Modifica del file .bashrc in SciTE con variazione del parametro PS1 da colore verde (32) a rosso (31) per personalizzare il prompt del terminale.

Domanda:

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

Risposta:

No. Perché la shell già aperta ha caricato la configurazione precedente. Il cambiamento si applica solo alle nuove sessioni o dopo il reload della shell.

Modifica del file .bashrc con nano

Nel terminale:

sudo nano .bashrc

Modifico il valore da 31 a 33, salvo con CTRL+X → Y → Invio.

Ricarico la configurazione con:

source .bashrc

The screenshot shows a terminal window titled "Terminal - analyst@secOps:~". The menu bar includes File, Edit, View, Terminal, Tabs, and Help. The terminal prompt is "[analyst@secOps ~]\$". The user enters the command "nano .bashrc" and then "source .bashrc". The output shows the command being run and the prompt returning to the user.

Output: Il prompt diventa giallo come a seguire

Modifica file di configurazione nginx

Apertura del file

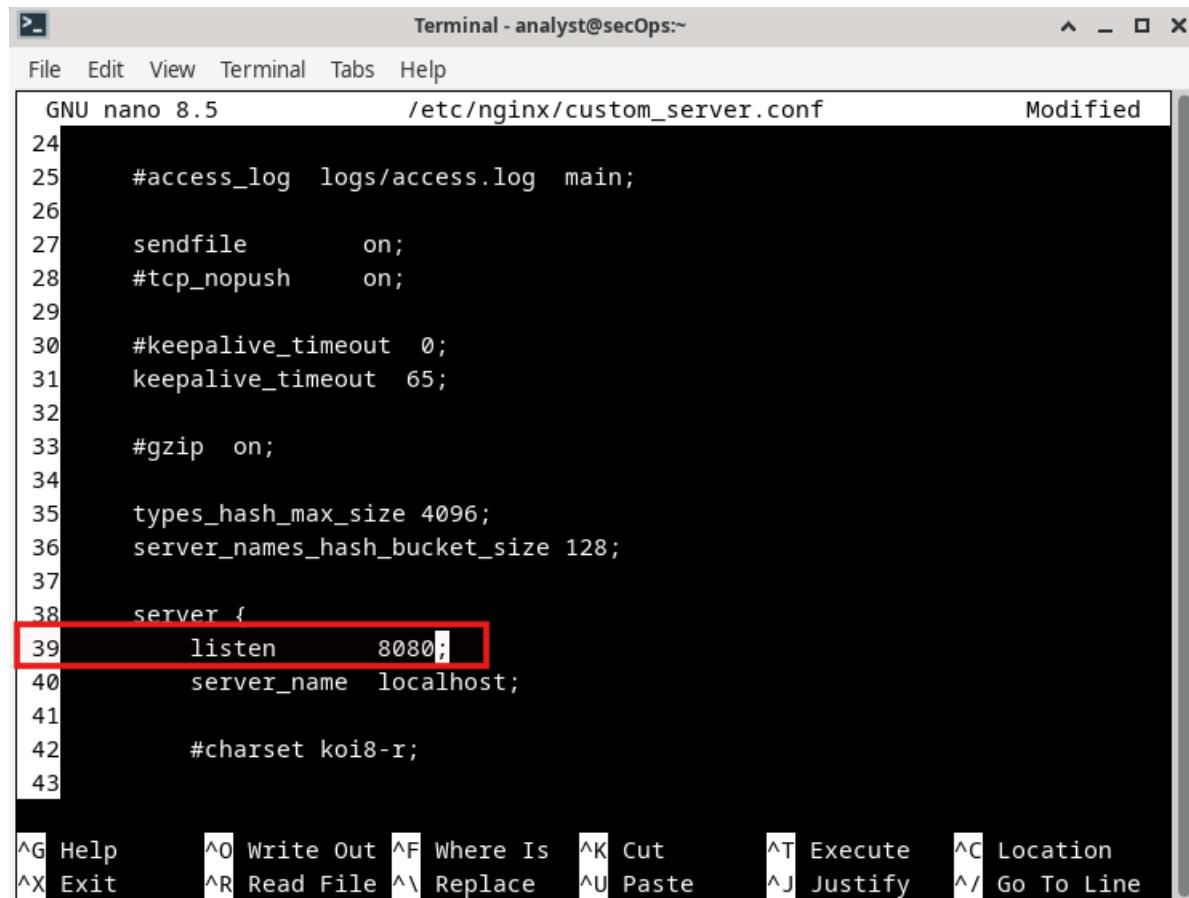
Nel terminale:

```
sudo nano -l /etc/nginx/custom_server.conf
```

Inserisco la password quando richiesta.

Modifiche richieste

- Riga 39: cambio porta da **81** a **8080**



```
GNU nano 8.5          /etc/nginx/custom_server.conf      Modified
24
25     #access_log    logs/access.log  main;
26
27     sendfile        on;
28     #tcp_nopush     on;
29
30     #keepalive_timeout  0;
31     keepalive_timeout  65;
32
33     #gzip  on;
34
35     types_hash_max_size 4096;
36     server_names_hash_bucket_size 128;
37
38     server {
39         listen      8080;
40         server_name localhost;
41
42         #charset koi8-r;
43     }
```

^G Help ^O Write Out ^F Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line

- Riga 47: modifco la directory root in
/usr/share/nginx/html/text_ed_lab/;

```

Terminal - analyst@secOps:~          ^ _ X
File Edit View Terminal Tabs Help
GNU nano 8.5           /etc/nginx/custom_server.conf      Modified
32
33     #gzip  on;
34
35     types_hash_max_size 4096;
36     server_names_hash_bucket_size 128;
37
38     server {
39         listen      8080;
40         server_name localhost;
41
42         #charset koi8-r;
43
44         #access_log  logs/host.access.log  main;
45
46         location / {
47             root      /usr/share/nginx/html/text_ed_lab/;  █
48             index   index.html index.htm;
49         }
50
51         #error_page  404          /404.html;

```

^G Help ^O Write Out ^F Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^/ Go To Line

Salvo ed esco.

Avvio del servizio nginx

Nel terminale:

sudo nginx -c /etc/nginx/custom_server.conf

```
[analyst@secOps ~]$ sudo nginx -c /etc/nginx/custom_server.conf
[sudo] password for analyst:
2026/02/19 10:01:44 [emerg] 1200#1200: invalid number of arguments in "root" directive
in /etc/nginx/custom_server.conf:48
[analyst@secOps ~]$
```

Domanda:

“A cosa si riferisce il messaggio di errore?”

Risposta:

Il messaggio di errore indica un problema di configurazione nel file `custom_server.conf`, nello specifico un numero non valido di argomenti nella direttiva `root` alla riga 48, che impedisce l'avvio corretto del servizio nginx.

Test dal Browser

Digito:

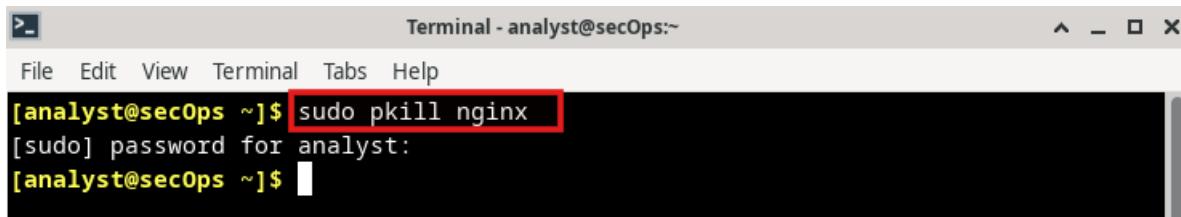
127.0.0.1:8080

Verifico il caricamento della pagina.

Arresto del servizio nginx

Nel terminale:

sudo pkill nginx



```
Terminal - analyst@secOps:~  
File Edit View Terminal Tabs Help  
[analyst@secOps ~]$ sudo pkill nginx  
[sudo] password for analyst:  
[analyst@secOps ~]$
```

Ricarico il browser.



Unable to connect

Firefox can't establish a connection to the server at 127.0.0.1:8080.

- The site could be temporarily unavailable or too busy. Try again in a few moments.
- If you are unable to load any pages, check your computer's network connection.
- If your computer or network is protected by a firewall or proxy, make sure that Firefox is permitted to access the web.

[Try Again](#)

Output: Pagina web non raggiungibile su 127.0.0.1:8080 dopo l'arresto del servizio nginx, con messaggio "Unable to connect".

Domanda:

“Appare la pagina web?”

Risposta:

No. Dopo l’arresto del servizio nginx, il browser non riesce a stabilire la connessione e mostra il messaggio “Unable to connect”, indicando che il server non è in esecuzione.

Domanda Sfida:

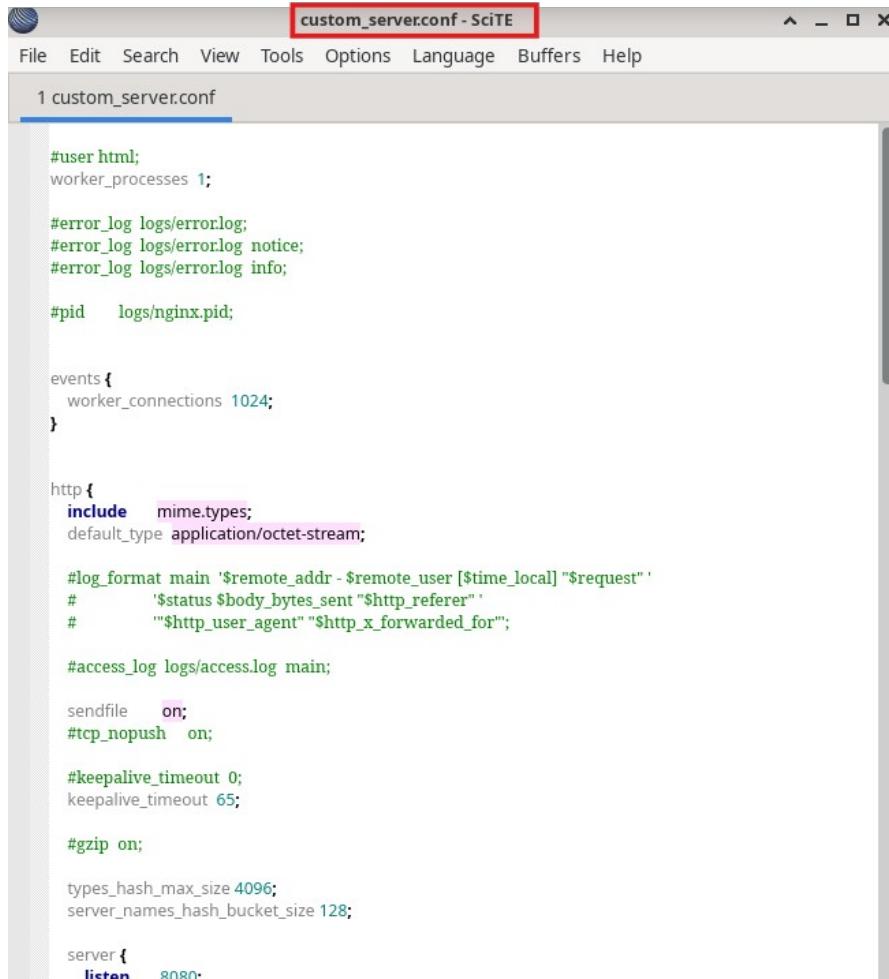
“Puoi modificare il file /etc/nginx/custom_server.conf con SciTE? Descrivi il processo di seguito.”

Risposta:

Sì. Dal terminale eseguo:

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

Inserisco la password, modifco il file e salvo. Sono necessari privilegi root perché il file si trova nella directory /etc.



The screenshot shows the SciTE editor window with the title bar "custom_server.conf - SciTE". The menu bar includes File, Edit, Search, View, Tools, Options, Language, Buffers, and Help. A single buffer tab labeled "1 custom_server.conf" is open. The code in the editor is a valid Nginx configuration file:

```
#user html;
worker_processes 1;

#error_log logs/error.log;
#error_log logs/error.log notice;
#error_log logs/error.log info;

#pid    logs/nginx.pid;

events {
    worker_connections 1024;
}

http {
    include mime.types;
    default_type application/octet-stream;

    #log_format main '$remote_addr - $remote_user [$time_local] "$request" '
    #          '$status $body_bytes_sent "'.$http_referer" '
    #          '"$http_user_agent" "$http_x_forwarded_for"';

    #access_log logs/access.log main;

    sendfile on;
    #tcp_nopush on;

    #keepalive_timeout 0;
    keepalive_timeout 65;

    #gzip on;

    types_hash_max_size 4096;
    server_names_hash_bucket_size 128;

    server {
        listen 8080;
    }
}
```

Output: Modifica del file custom_server.conf in SciTE con configurazione della porta 8080 per il servizio nginx.

