
S11-L4

Editor di testo Linux

Emanuele Benedetti | 20 febbraio 2025

Consegna

Laboratorio incentrato sul lavorare con i file di testo nella CLI

In questo laboratorio si acquisirà familiarità con gli editor di testo da riga di comando in Linux e con la modifica dei 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

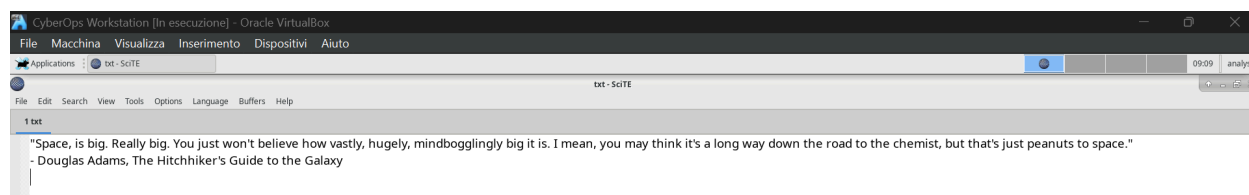
Svolgimento

L'esercizio è stato svolto su macchina virtuale *CyberOps*

Parte 1 - Editor di testo grafici

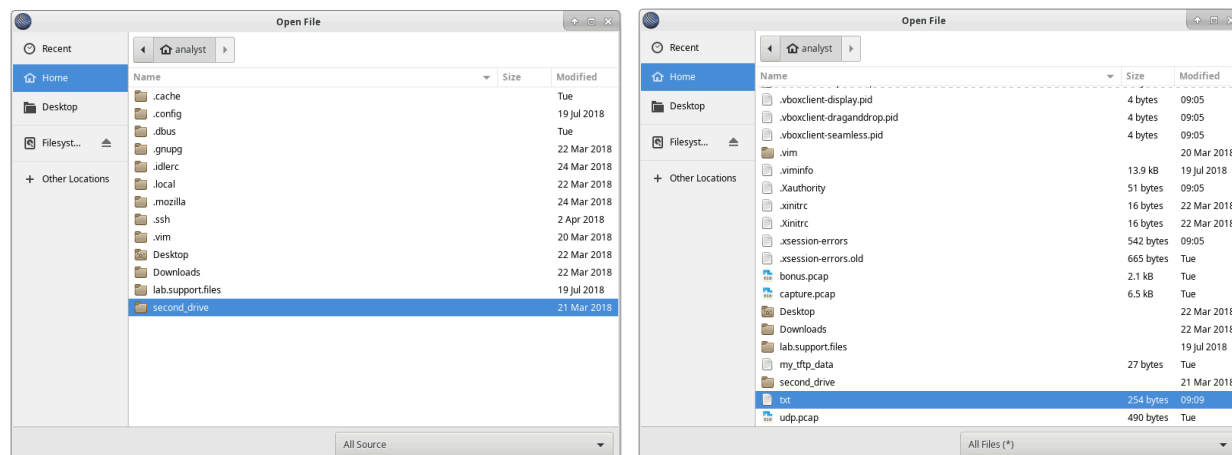
Dopo aver eseguito l'accesso alla macchina, apriamo l'editor di testo SciTE con *Applications > CyberOps > SciTE*.

Creiamo un nuovo file e inseriamo del testo all'interno, come mostrato in figura:



Salviamo il file con *File > Save* nella directory */home/analyst*.

Chiudiamo il file e andiamo su *File > Open*. Come mostrato di seguito non siamo in grado di trovare il file nella directory, dobbiamo modificare la in basso *All Source* con *All Files (*)*.



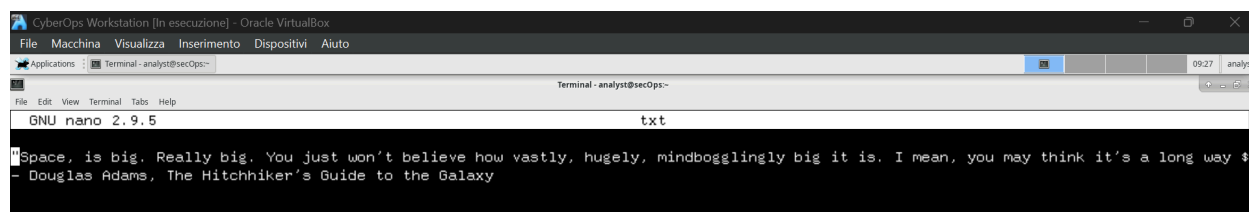
In generale in Linux ogni cosa è considerata un file, tuttavia alcune app (come SciTE) tentano di utilizzare le estensioni dei file per identificare i tipi.

Parte 2 - Editor di testo da riga di comando

Gli editor di testo da riga di comando sono comodi ed utili in situazioni in cui non è possibile utilizzare l'interfaccia grafica per modificare un file ma solamente la riga di comando (ad esempio connessione da una shell remota).

Utilizziamo l'editor *nano* per aprire il file di testo creato nella parte 1.

Eseguiamo sul terminale il comando *nano txt* per aprire il file. Poiché non è stato inserito nessun carattere di ritorno, nano visualizza il testo su un'unica riga, mostrando che il testo non è finito tramite il simbolo *\$*



Parte 3 - Lavorare con file di configurazione

I file di configurazione di Linux sono contenuti nella directory `/etc` del file system. Alcuni file di configurazione delle applicazioni dell'utente tuttavia sono file nascosti nelle rispettive directory home.

Ad esempio eseguendo il comando `ls -la` in `/home/analyst` possiamo vedere i file nascosti che non vediamo con `ls -l`.

```
[analyst@sec0ps ~]$ ls -l
total 40
-rw-r--r-- 1 root    root      2061 Feb 18 09:26 bonus.pcap
-rw-r--r-- 1 root    root      6527 Feb 18 08:31 capture.pcap
drwxr-xr-x 2 analyst analyst 4096 Mar 22 2018 Desktop
drwxr-xr-x 3 analyst analyst 4096 Mar 22 2018 Downloads
drwxr-xr-x 9 analyst analyst 4096 Jul 19 2018 lab.support.files
-rw-r--r-- 1 root    root        27 Feb 18 10:43 my_tftp_data
drwxr-xr-x 2 analyst analyst 4096 Mar 21 2018 second_drive
-rw-r--r-- 1 analyst analyst   254 Feb 20 09:09 txt
-rw-r--r-- 1 root    root        490 Feb 18 10:43 udp.pcap

[analyst@sec0ps ~]$ ls -la
total 168
drwx----- 15 analyst analyst 4096 Feb 20 09:09 .
drwxr-xr-x  3 root    root      4096 Mar 20 2018 ..
-rw-----  1 analyst analyst   956 Feb 18 10:48 .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
-rw-r--r--  1 root    root      2061 Feb 18 09:26 bonus.pcap
drwxr-xr-x  8 analyst analyst 4096 Feb 18 08:25 .cache
-rw-r--r--  1 root    root      6527 Feb 18 08:31 capture.pcap
drwxr-xr-x 10 analyst analyst 4096 Jul 19 2018 .config
drwx-----  3 analyst analyst 4096 Feb 18 08:25 .dbus
drwxr-xr-x  2 analyst analyst 4096 Mar 22 2018 Desktop
```

Possiamo utilizzare `ls /etc` per listare il contenuto della directory `etc`

```
[analyst@sec0ps ~]$ ls /etc
adjtime          ethertypes       initcpio          logrotate.conf   ntab              pcmcia            security          tmpfiles.d
apparmor.d       fonts             inputrc           logrotate.d       nanorc            pkcs11            sensors3.conf    trusted-key.key
arch-release     fstab             iproute2          lvm               netconfig         polkit-1          sensors.d         ts.conf
avahi             gai.conf          iptables          machine-id        netctl            profile           services         udev
bash.bash_logout group             issue             mailcap           ncd.conf          protocols         shadow           UPower
bash.bashrc      group            kernel            mail.rc           nssd.conf         rc.d              shadow-          vbox
binfmt.d         grub.d            krb5.conf         makepkg.conf      nsswitch.conf     rc_keymaps        shells           vdpau-wrapper.cfg
ca-certificates gshadow           ld.so.cache       man_db.conf       ntp.conf          rc_keymaps        skel             vimrc
conf.d            gshadow-         ld.so.conf        mdadm.conf        openldap          request-key.conf  snort            vsftpd.conf
crypttab         gtk-2.0           ld.so.conf.d      mime.types        openvswitch       request-key.conf  ssh              vsftpd.conf_stock
dbus-1           gtk-3.0           libnl             mkinitcpio.conf  os-release        request-key.conf  ssl              X11
default          healthd.conf     lightdm           mkinitcpio.d      pacman.conf       sudoers           sudoers         xdg
depmod.d         hostname         locale.conf       modprobe.d        pacman.d          sysctl.d          sysctl.d         xinetd.d
dhcpcd.conf      hosts            localtime         modules-load.d    pam.d             syslog-ng         syslog-ng        yaourt.rc
drirc            ifplugd          login.defs        motd              passwd            securetty         systemd
```

e *cat bash.bashrc* per leggere il contenuto del file *bash.bashrc*.

```
[analyst@sec0ps ~]$ cat /etc/bash.bashrc
#
# /etc/bash.bashrc
#
# If not running interactively, don't do anything
[[ $- != *i* ]] && return

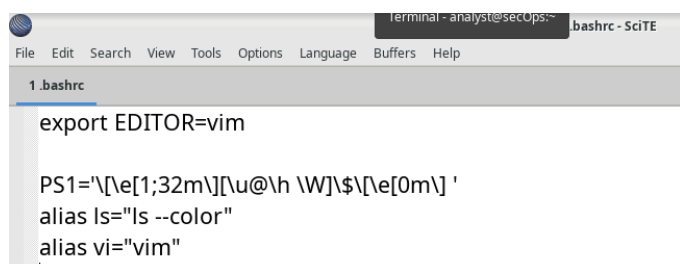
[[ $DISPLAY ]] && shopt -s checkwinsize

PS1='${\u@\h \w}\$ '

case ${TERM} in
  xterm*|rxvt*|Eterm|aterm|kterm|gnome*)
    PROMPT_COMMAND=${PROMPT_COMMAND:+$PROMPT_COMMAND; }'printf "\033]0; %s%s: %s\007" "${USER}" "${HOSTNAME%%.*}" "${PWD/#$HOME/\~}"'
    ;;
  screen*)
    PROMPT_COMMAND=${PROMPT_COMMAND:+$PROMPT_COMMAND; }'printf "\033_%s0%s: %s\033\\" "${USER}" "${HOSTNAME%%.*}" "${PWD/#$HOME/\~}"'
    ;;
  esac

[ -r /usr/share/bash-completion/bash_completion ] && . /usr/share/bash-completion/bash_completion
```

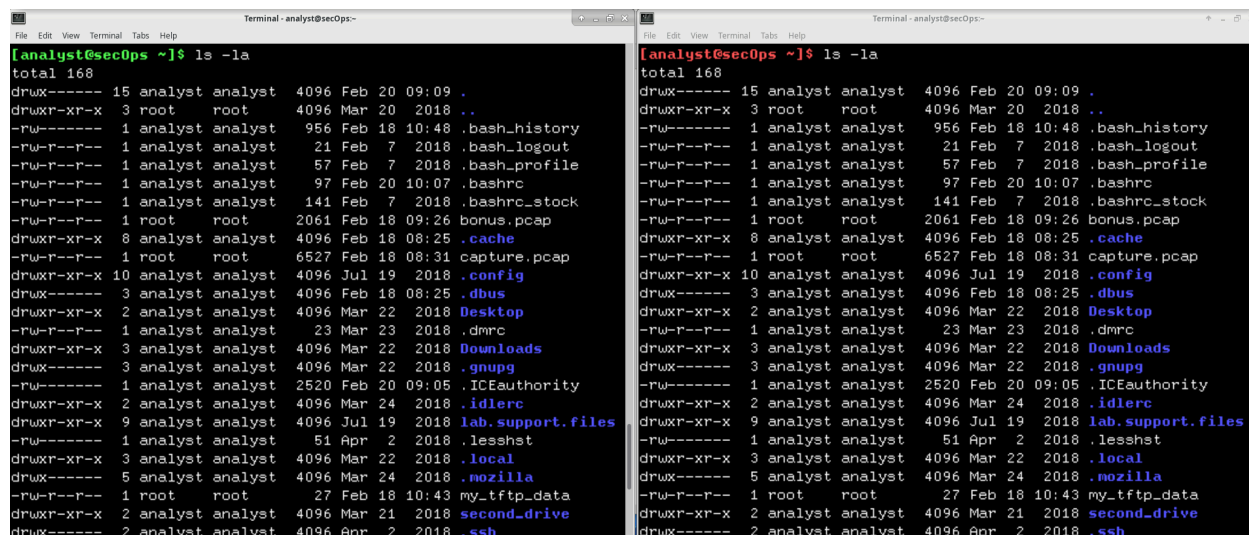
Apriamo il file di configurazione *.bashrc* tramite *SciTE*



```
1 .bashrc
export EDITOR=vim

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

e modifichiamo il valore 32 presente all'interno del file con 31. 32 indica il colore verde del prompt del terminale, 31 il colore rosso. Così facendo stiamo cambiando il colore d'aspetto del terminale. come mostrato in figura.



```
[analyst@sec0ps ~]$ ls -la
total 168
drwxr-xr-x 3 root root 4096 Mar 20 2018 ..
-rw-r--r-- 1 analyst analyst 956 Feb 18 10:48 .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 Feb 20 10:07 .bashrc
-rw-r--r-- 1 analyst analyst 141 Feb 7 2018 .bashrc_stock
-rw-r--r-- 1 root root 2061 Feb 18 09:26 bonus.pcap
drwxr-xr-x 8 analyst analyst 4096 Feb 18 08:25 .cache
-rw-r--r-- 1 root root 6527 Feb 18 08:31 capture.pcap
drwxr-xr-x 10 analyst analyst 4096 Jul 19 2018 .config
drwxr-xr-x 3 analyst analyst 4096 Feb 18 08:25 .dbus
drwxr-xr-x 2 analyst analyst 4096 Mar 22 2018 .Desktop
-rw-r--r-- 1 analyst analyst 23 Mar 23 2018 .dmrc
drwxr-xr-x 3 analyst analyst 4096 Mar 22 2018 .Downloads
drwxr-xr-x 3 analyst analyst 4096 Mar 22 2018 .gnupg
-rw-r--r-- 1 analyst analyst 2520 Feb 20 09:05 .ICEauthority
drwxr-xr-x 2 analyst analyst 4096 Mar 24 2018 .idlerc
drwxr-xr-x 9 analyst analyst 4096 Jul 19 2018 .lab.support.files
-rw-r--r-- 1 analyst analyst 51 Apr 2 2018 .lessht
drwxr-xr-x 3 analyst analyst 4096 Mar 22 2018 .local
drwxr-xr-x 5 analyst analyst 4096 Mar 24 2018 .mozilla
-rw-r--r-- 1 root root 27 Feb 18 10:43 my.tftp_data
drwxr-xr-x 2 analyst analyst 4096 Mar 21 2018 .second_drive
drwxr-xr-x 2 analyst analyst 4096 Apr 2 2018 .ssh
```

Modifichiamo ora il file di configurazione del servizio *nginx* (web server preinstallato nella VM) situato in */etc/nginx* tramite *nano sudo -l /etc/nginx/custom_server.conf*.

Alla riga 39 modifichiamo la porta di ascolto del server HTTP da *81* a *8080*.

```
32
33     #gzip on;
34
35     types_hash_max_size 4096;
36     server_names_hash_bucket_size 128;
37
38     server {
39         listen      81;
40         server_name localhost;
```

```
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;
```

E alla linea 47 modifichiamo */usr/share/nginx/html/* in */usr/share/nginx/html/text_ed_lab/*. Salviamo le modifiche con *CTRL+x > y > ENTER*.

Applichiamo le modifiche con *sudo nginx -c custom_server.conf* e riavviamo il server *nginx* con *sudo kill nginx*.

Infine verifichiamo l'aggiornamento collegandoci tramite browser web all'indirizzo *127.0.0.1:8080* ed otteniamo il messaggio mostrato nello screenshot:



Congratulations!

As part of the Working with Text Files lab, you have successfully configured NGINX!