

Prendere Familiarità con la Shell Linux

- Elenca alcune sezioni incluse in una pagina man

NAME – Il nome del comando e una breve descrizione

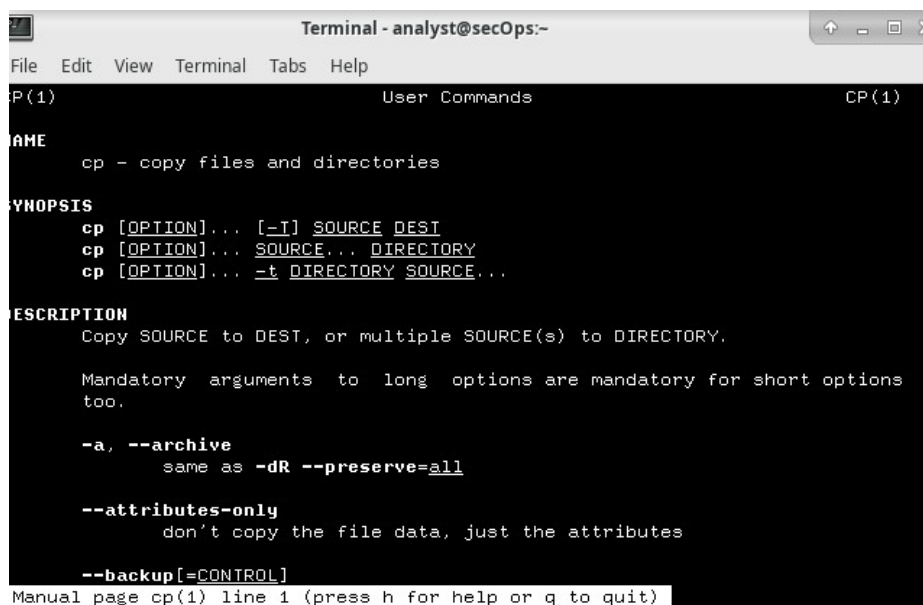
SYNOPSIS – La sintassi del comando, cioè come scriverlo correttamente

DESCRIPTION – Una descrizione dettagliata del comando e del suo comportamento

OPTIONS – Tutte le opzioni e flag disponibili (es. -r, -i, --help, ecc.)

- Quale è la funzione del comando cp?

Il comando CP serve per copiare file o directory in Linux.



```
Terminal - analyst@secOps:~
File Edit View Terminal Tabs Help
CP(1) User Commands CP(1)
NAME
cp - copy files and directories
SYNOPSIS
cp [OPTION]... [-I] SOURCE DEST
cp [OPTION]... SOURCE... DIRECTORY
cp [OPTION]... -t DIRECTORY SOURCE...
DESCRIPTION
Copy SOURCE to DEST, or multiple SOURCE(s) to DIRECTORY.

Mandatory arguments to long options are mandatory for short options
too.

-a, --archive
    same as -dR --preserve=all

--attributes-only
    don't copy the file data, just the attributes

--backup[=CONTROL]
Manual page cp(1) line 1 (press h for help or q to quit)
```

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

man pwd

- Quale è la funzione del comando pwd

La funzione del comando pwd serve per mostrare il percorso assoluto della directory corrente in cui ti trovi nel terminale.

- Quale è la directory corrente?

/home/analyst

```
analyst@sec0ps ~]$ pwd
/home/analyst
analyst@sec0ps ~]$
```

- In quale cartella ti trovi ora?

Mi trovo nella cartella cyops_folder3

```
analyst@sec0ps cyops_folder3]$
```

- Perché è successo?

Digitando il comando `cd ~` vieni riportato alla **home directory del tuo utente**, che in questo caso è: /home/analyst

Il simbolo ~ (tilde) è una scorciatoia che rappresenta la directory personale dell'utente attualmente loggato.

- Cosa succede?

```
analyst@sec0ps cyops_folder3]$ cd .
analyst@sec0ps cyops_folder3]$
```

Digitando `cd .` Non succede nulla di visibile perché `.` rappresenta la directory corrente.

- Digita `cd ..` – Cosa succede?

```
analyst@sec0ps cyops_folder3]$ cd ..
analyst@sec0ps ~]$
```

Ci spostiamo nella directory genitore, cioè un livello sopra rispetto alla posizione attuale.

- Qual è la directory corrente se esegui `cd ..` da `[analyst@secOps ~]$`?

/home

```
analyst@secOps ~]$ cd ..
analyst@secOps home]$ ls
```

- Qual è la directory corrente se esegui `cd ..` da `[analyst@secOps home]$`?

/

- Qual è la directory corrente se esegui `cd ..` da `[analyst@secOps /]$`?

Rimane la stessa

```
analyst@secOps home]$ cd ..
analyst@secOps /]$ cd ..
analyst@secOps /]$
```

- Nessun output è stato mostrato. È previsto? Spiega.

```
[analyst@secOps ~]$ echo Questo è un messaggio inviato al terminale da echo. > some_text.txt
[analyst@secOps ~]$
```

Sì, è previsto! Quando si usa `>` stiamo reindirizzando l'output del comando (in questo caso `echo`) in un file, quindi il messaggio non viene mostrato a schermo.

- Utilizzando l'operatore `>` Cosa è successo al file di testo? Spiega.

```
[analyst@secOps ~]$ cat some_text_file.txt
Questo è un messaggio DIVERSO, ancora una volta inviato al terminale da echo.
```

Il contenuto del file `some_text_file.txt` è stato sovrascritto.

- Utilizzando l'operatore `>>` Cosa è successo al file di testo? Spiega.

Il nuovo messaggio è stato **accodato**, cioè **aggiunto alla fine del file**, mantenendo il contenuto già esistente

- Quanti file vengono visualizzati?

```
analyst@sec0ps ~]$ ls -l
total 48
-rw-r--r-- 1 root    root      7301 Jun 10 08:38 capture.pcap
drwxr-xr-x 2 analyst analyst  4096 Jun 12 10:10 cyops_folder1
drwxr-xr-x 2 analyst analyst  4096 Jun 12 10:10 cyops_folder2
drwxr-xr-x 3 analyst analyst  4096 Jun 12 10:21 cyops_folder3
drwxr-xr-x 3 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
drwxr-xr-x 2 analyst analyst  4096 Mar 21  2018 second_drive
-rw-r--r-- 1 analyst analyst    79 Jun 12 10:43 some_text_file.txt
-rw-r--r-- 1 analyst analyst    51 Jun 12 10:35 some_text.txt
-rw-r--r-- 1 analyst analyst   312 Jun 12 08:24 space.txt
analyst@sec0ps ~]$
```

- Quanti file in più vengono visualizzati con ls -la? Spiega.

Con ls -la vengono mostrati:

Tutti i file visibili, In più i file nascosti (quelli che iniziano con .) ,Anche le directory speciali . (corrente) e .. (genitore)

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

Sì, In Linux, qualsiasi file o directory il cui nome inizia con . viene considerato nascosto.

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

Sì, sempre, come: . → directory corrente.. → directory genitore E poi eventuali file o cartelle come .bashrc, .profile, ecc.

- Fornisci tre esempi di file nascosti non mostrati nell'output di ls -l ma visibili in ls -la

.bashrc

.bash_profile

.bash_history

Parte 2

- Quali sono i file sorgente e destinazione?

```
[analyst@secOps ~]$ cp some_text_file.txt cyops_folder2/
```

File sorgente:

/home/analyst/some_text_file.txt

Directory di destinazione:

/home/analyst/cyops_folder2/

- Quale comando hai usato per completare l'attività?

Per completare l'attività ho utilizzato il comando mv, che serve a spostare un file da una directory all'altra. In questo caso, ho spostato il file some_text_file.txt dalla cartella cyops_folder2 alla mia directory personale (/home/analyst)

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

La riga di comando in Linux è uno strumento efficiente e potente, che consente di eseguire operazioni complesse con rapidità, spesso con un solo comando, offrendo un controllo completo sul sistema, permette di automatizzare attività e gestire file o processi in modo dettagliato.

È indispensabile per lavorare su server remoti o ambienti senza interfaccia grafica ed è uno strumento fondamentale per chi opera in ambito sistemistico, sviluppo o cybersecurity

