

**Cyber  
Security  
Analyst**

# PROJECT REPORT

## W5D1 PRATICA 1

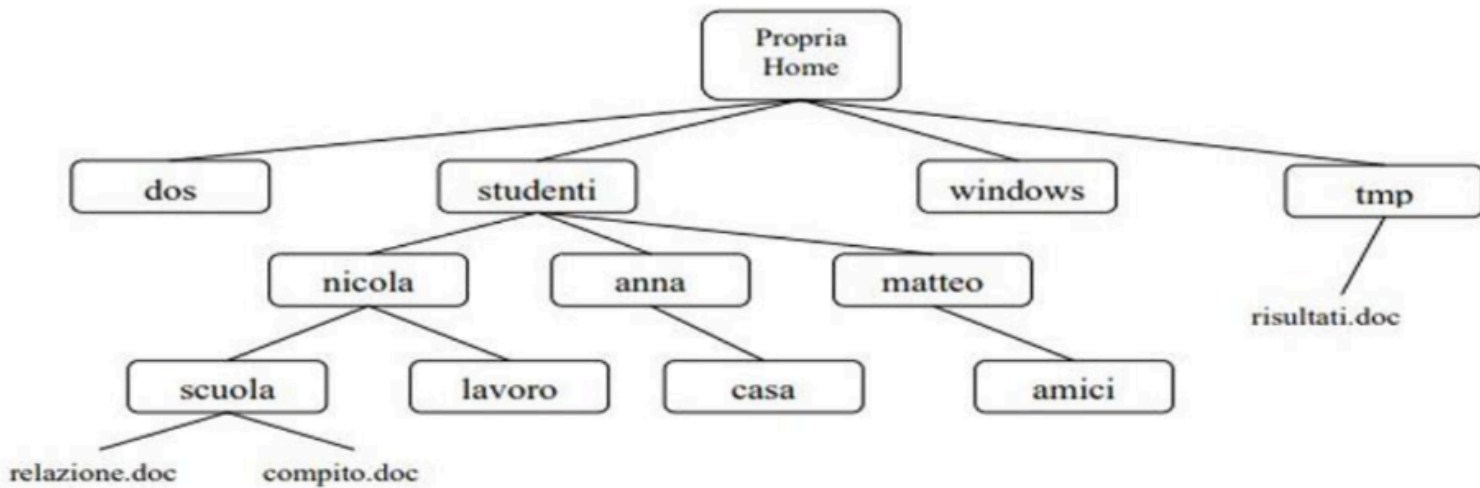
In risposta all'esercizio sulla  
Shell di linux parte 2 : vedi  
schema in pagina seguente.

**Prepared by**

Fulvio Zalateu

**Security  
Rookies**

# Esercitazione



Ti trovi nella directory **nicola** (sotto studenti),

- j) Cambiare gli attributi della directory lavoro facendo in modo che il suo contenuto non sia leggibile ma consentendo solo a te di entrarvi
- k) Entra nella cartella lavoro e crea un file (emacs – ricordati di salvarlo)
- l) Prova a visualizzare il contenuto della directory dove ti trovi dopo di che modifica gli attributi della directory '.'. Facendo in modo che siano nuovamente leggibile il contenuto e prova nuovamente a leggere il contenuto
- m) Spostati nella cartella scuola usando il percorso relativo (a dove ti trovi)
- n) Nella directory scuola crea una directory .mia (punto mia) e fai in modo che sia leggibile scrivibile ed eseguibile solo da te
- o) Prova a eseguire il comando ls e successivamente ls -al

## Esercizi:

10. lancia il comando emacs &
11. esegui il comando jobs
12. lancia il comando firefox e successivamente sul terminale premi ^z
13. manda il processo firefox in background (bg)
14. lancia il comando jobs
15. manda in foreground (fg) il programma emacs
16. termina emacs con il tasti ^c
17. verificare quanto spazio si sta occupando su disco

# Shell di Linux: parte 2

**Cambiamo gli attributi della directory “lavoro” facendo in modo che il suo contenuto non sia leggibile ma consentendo solo a te di entrare all’interno.**

Rechiamoci al livello appena superiore alla cartella lavoro con il comando cd.

Lanciamo il comando:

`ls -al`

per visualizzare gli attuali attributi della cartella. Utilizziamo il comando:

`chmod u=rwx,go -r-w-x lavoro`

che attribuisce i permessi di lettura (read), scrittura (write) e esecuzione (execute), mentre group (gruppo) e others (altri) toglie i permessi di rwx (lettura, scrittura ed esecuzione).

Verifichiamo di aver apportato le modifiche utilizzando nuovamente il comando:

`ls -al.`

**Entra nella cartella lavoro e crea un file (emacs - ricordati di salvarlo)**

Spostiamoci nella cartella lavoro con il comando:

`cd`

e usiamo l'editor nano (comando):  
`sudo nano emacs` per scrivere il contenuto del programma emacs

**Prova a visualizzare il contenuto della directory dove ti trovi dopo di che modifica gli attributi della directory '.' facendo in modo che sia nuovamente leggibile il contenuto e prova nuovamente a leggere il contenuto**

Utilizzando il comando:

`cd`

rechiamoci alla cartella studenti.

Cambiamo i permessi con il comando

`chmod ugo+r .anna`

La cartella anna viene così nascosta per tutti (user+group+others) e può essere letta +(read)

Verifichiamo il contenuto con i comandi  
ls e/o ls -a

Rendiamo nuovamente non nascosta la  
cartella anna con il comando:  
mv .anna anna

**Spostati nella cartella scuola  
usando il percorso relativo (a dove  
ti trovi)**

Utilizziamo il comando:  
cd

seguito dal percorso relativo in base a  
dove ci troviamo in quel momento.

Ricordiamo che “cd-” riporta alla directory  
da cui siamo arrivati immediatamente  
prima; “cd ..” porta alla directory padre di  
quella corrente ;”cd ../..” porta a due livelli  
più in alto nella struttura delle directory.

**Nella directory scuola crea una  
directory .mia (punto mia) e fai in  
modo che sia leggibile ed  
eseguibile solo da te**

Rechiamoci con il comando:

`cd`

nella cartella scuola. Creiamo la cartella

“.mia” quindi:

`mkdir .mia`

Otteniamo la lista del contenuto, anche  
nascosto della cartella e i permessi attuali

comando:

`ls -al`

Modifichiamo i permessi:

`chmod u=rwx, go-rwx .mia` e cioè

utente = read write execute, mentre group

& others toglie i permessi read write

excecute alla cartella .mia



Verifichiamo nuovamente le modifiche apportate con:

`ls -al`

Le modifiche sono state apportate con successo: non ci sono permessi nella colonna group e others

**Prova a eseguire il comando `ls` e successivamente `ls -al`**

La differenza principale tra i due comandi sta nella mole di dati che si ottengono con `ls -al`

Quest'ultimo mostra molti più dati, soprattutto file nascosti e attributi.

**Esercizi sulla gestione processi**



## **Lancia il comando emacs &**

Emacs è un editor di testo con una grande varietà di compiti, grazie a flessibilità e potenza.

Rechiamoci ora alla cartella “Lavoro” con il comando:

`cd`

Avviamo emacs in background emacs &  
Si evince che il PID (Process ID utilizzato per monitorare il processo in questione) numero 169380 è stato chiuso.

## **Esegui il comando jobs**

Usiamo il comando:

`jobs`

Al momento non funziona in quanto non ci sono processi di competenza del comando attivi.

# **Lancia il comando firefox e successivamente sul terminale premi CTRL+Z**

Quando digitiamo firefox e premiamo Invio, il sistema avvia il browser Firefox. Questo processo occupa il terminale fino a quando Firefox è in esecuzione. Premendo “ctrl Z” l’esecuzione di firefox si stoppa liberando di fatto il terminale per altri comandi.

## **Manda il processo firefox in background (bg) e lancia il comando jobs**

Digitando “bg” mettiamo il processo sospeso in background. Questo comando riprende l'esecuzione del processo Firefox, ma non blocca più il terminale. Il terminale mostra qualcosa come:

Digitando “jobs” possiamo vedere l'elenco dei processi che sono stati lanciati dal terminale e che sono attualmente sospesi o in background.

## **Manda in foreground (fg) il programma emacs**

Non è stato possibile perché il programma emacs è stato già terminato dal sistema precedentemente.

## **Termina emacs con i tasti CTRL+C**

Anche in questo caso, l'operazione non è stata tecnicamente fattibile essendo il programma emacs già terminato dal sistema precedentemente.

## **Verificare quanto spazio si sta occupando sul disco**

Utilizzando il comando:

`du -h`

riporta che la directory occupa 8 kilobyte sul disco.

**GRAZIE**