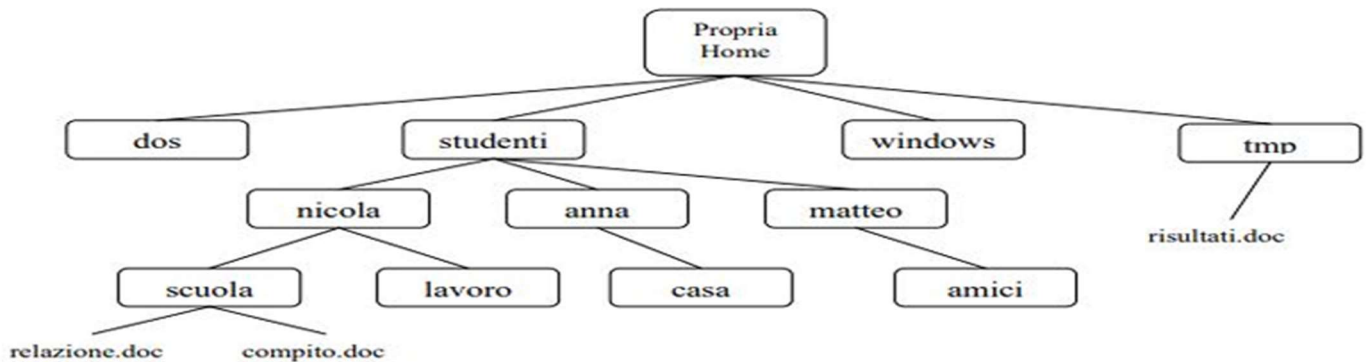


DATA
Cybersecurity Analyst

Studente:
Andrea Scarmagnani
Docente:
Giuseppe Placanica

Esercizio 1

Come prima cosa creare le seguenti cartelle e sottocartelle (usando i comandi “terminale” mkdir cd rmdir ... a partire dalla propria HOME e visualizzarle a video:
(Per “Propria home” si intende il posto dove vi posiziona quando aprite il terminale!)



Ti trovi nella directory **lavoro** (sotto nicola), scrivere il comando per passare alla directory **casa** (sotto anna) con percorso relativo e percorso assoluto.

- Copia il file compito.doc (dalla directory scuola) nella directory corrente (casa).
- Sposta il file relazione.doc nella directory corrente (casa).
- Cancella la cartella **tmp**
- Creare il file pippo.txt nella cartella lavoro
- Cambiare gli attributi del file pippo.txt e renderlo scrivibile e leggibile solo per il proprietario, mentre per tutti gli altri solo leggibile...
- Nascondere il contenuto della cartella anna
- Spostarsi nella cartella lavoro e visualizzare il contenuto del file pippo.txt
- Rimuovere la cartella amici
- Rimuovere tutte le cartelle precedentemente create

Qui sotto l'elenco dei comandi utilizzati per l'esecuzione dell'esercizio.

239 mkdir dos	323 mv cd ..
240 ls	327 cd scuola/
241 mkdir studenti	328 ls
242 mkdir windows	329 cd ..
243 mkdir tmp	330 cd ..
244 cd /studenti	331 cd scuola
245 cd studenti	332 ls
246 mkdir nicola mkdir anna mkdri matteo	333 mv relazione.doc /home/kali
247 mkdir nicola mkdir anna mkdir matteo	334 ls
248 cd nicola	335 cd
249 mkdir scuola mkdir lavoro	336 cd /home/kali
253 cd studenti/nicola	337 ls
254 cd scuola	339 rmdir tmp/
255 touch relazione.doc	340 cd tmp
256 touch compito.doc	341 ls
257 cd	342 cd
258 cd studenti/nicola/scuola	343 cd /home/kali
261 cd	344 man rm
262 cd tmp	345 cd tmp
263 touch risultati.doc	346 ls
265 cd	347 cd ..
266 cd studenti/anna	348 ls
267 ls	349 rm -r /tmp
269 mkdir casa	350 ls
273 cd studenti/matte	351 rm tmp
274 cd studenti/matteo	352 rm -r tmp
275 ls	353 ls
276 cd studenti	354 touch
278 cd home/kali	/home/kali/studenti/nicola/lavoro/pippo.txt

279 ls 280 cd studenti/matteo 281 ls 282 mkdir amici 283 ls 286 cd studenti/nicola/lavoro 288 ls 300 cd home/kali 301 ls 307 cd studenti 308 ls 309 cd nicola 310 ls 311 cd scuola 314 cd .. 315 cd tmp 316 ls 317 cd .. 318 dir 319 cp studenti/nicola/scuola/compito.doc studenti/nicola/lavoro 320 ls 321 cd studenti/nicola/lavoro 322 ls	357 ls -lr 358 ls -l 359 cd studenti/nicola/lavoro 360 ls 361 ls -l 364 chmod u+w pippo.txt 365 ls -l 366 chmod u+x pippo.txt 367 ls -l 368 cd .. 370 cd .. 371 cd studenti/anna 372 cd studenti/anna/ 373 cd /studenti/anna/ 374 cd /studenti/anna 375 cd anna 376 cd.. 377 ll 378 ls 379 mkdir .anna 380 ls 382 cd .. 383 ls
385 mv anna .anna 386 ls 387 ls -l 388 ls -a 389 ls -la 390 cd studenti/nicola/lavoro 391 ls 392 cd studenti/nicola/lavoro 393 cd /nicola/lavoro 394 cd nicola 395 ls 396 cd lavoro 397 ls 398 nano pippo.txt 399 cd .. 400 ls 401 rm -r studenti/matteo/amici 402 cd studenti/matteo 403 ls 404 cd .. 405 ls 406 rm -r dos rm -r studenti rm -r windows 407 ls 408 rm relazione.doc 409 ls	

Provare i comandi:

```
w  
who  
who am i
```

Esercizi - processi:

1. Aprire un terminale
2. leggere il manuale del comando job, ps e kill
3. lanciare il comando vi pippo
4. aprire un nuovo terminale e visualizzare tutti i propri processi...
5. cercare di terminare (killare) il processo vi per sbloccare il terminale precedente
6. lanciare il comando firefox in background
7. portarlo in background
8. cercare di terminare il processo firefox
9. verificare quanto spazio si sta occupando su disco

Qui sotto l'esecuzione dell'esercizio

Manuale di jobs si scrive nel terminale: *man builtins*

```
jobs [-npsr] [ jobspec ... ]  
jobs -x command [ args ... ]  
The first form lists the active jobs. The options have the following meanings:  
-n Display information only about jobs that have changed status since the user was last notified of their status.  
-r Display only running jobs.  
-s Display only stopped jobs.  
If jobspec is given, output is restricted to information about that job. The return status is 0 unless an invalid option is encountered or an invalid jobspec is supplied.  
If the -x option is supplied, jobs replaces any jobspec found in command or args with the corresponding process group ID, and executes command passing it args, returning its exit status.
```

Manuale di PS: nel terminale si scrive: *man ps*

```
ps(1)  
User Commands  
ps(1)  
NAME  
ps - report a snapshot of the current processes.  
SYNOPSIS  
ps [options]  
DESCRIPTION  
ps displays information about a selection of the active processes. If you want a repetitive update of the selection and the displayed information, use top instead.  
This version of ps accepts several kinds of options:  
1 UNIX options, which may be grouped and must be preceded by a dash.  
2 BSD options, which may be grouped and must not be used with a dash.  
3 GNU long options, which are preceded by two dashes.  
Options of different types may be freely mixed, but conflicts can appear. There are some synonymous options, which are functionally identical, due to the many standards and ps implementations that this ps is compatible with.  
Note that ps -aux is distinct from ps aux. The POSIX and UNIX standards require that ps -aux print all processes owned by a user named x, as well as printing all processes that would be selected by the -a option. If the user named x does not exist, this ps may interpret the command as ps aux instead and print a warning. This behavior is intended to aid in transitioning old scripts and habits. It is fragile, subject to Manual page ps(1) line 1 (press h for help or q to quit)
```

Manuale di Kill: nel terminale si scrive: *man kill*

```
kill(1)  
User Commands  
kill(1)  
NAME  
kill - send a signal to a process  
SYNOPSIS  
kill [options] <pid> [...]  
DESCRIPTION  
The default signal for kill is TERM. Use -l or -L to list available signals. Particularly useful signals include HUP, INT, KILL, STOP, CONT, and 0. Alternate signals may be specified in three ways: -9, -SIGKILL or -KILL. Negative PID values may be used to choose whole process groups; see the PGID column in ps command output. A PID of -1 is special; it indicates all processes except the kill process itself and init.  
OPTIONS  
<pid> [...]  
Send signal to every <pid> listed.  
-s <signal>  
-s <signal>  
--signal <signal>  
Specify the signal to be sent. The signal can be specified by using name or number. The behavior of signals is explained in signal(7) manual page.  
-q, --queue value  
Use sigqueue(3) rather than kill(2) and the value argument is used to specify an integer to be sent with the signal. If the receiving process has installed a handler for this signal using the SA_SIGINFO flag to sigaction(2), then it can obtain this data via the si_value field of the siginfo_t structure.  
Manual page kill(1) line 1 (press h for help or q to quit)
```

```
kali@kali: ~
File Actions Edit View Help
Prova scrittura
sad
sda
asd
sds
asdasassdsasd
d
asssdas
$
10,0-1 All

(kali@kali)-[~]
$ ps -aux | grep pippo
kali 6055 0.0 0.2 14900 9600 pts/0 SL+ 12:50 0:00 vi pippo
kali 7060 0.0 0.0 6340 2304 pts/1 S+ 12:52 0:00 grep --col

(kali@kali)-[~]
$ kill 6055

(kali@kali)-[~]
$

File Actions Edit View Help
-s Display only stopped jobs.
If jobspec is given, output is restricted to information about tha
If the -x option is supplied, jobs replaces any jobspec found in c
kill [-s sigspec | -n signum | -sigspec] [pid | jobspec] ...
Send the signal named by sigspec or signum to the processes named
-b Report the status of terminated background jobs immediatel
checkjobs
If set, bash lists the status of any stopped and running j
tervening command (see JOB CONTROL in bash(1)). The shell
If set, bash will send SIGHUP to all jobs when an interact
given, wait waits for all running background jobs and the last-exe

Vim: Caught deadly signal TERM
Vim: Finished.
zsh: terminated vi pippo

(kali@kali)-[~]
$

File Actions Edit View Help
File Actions Edit View Help
kali 1562 0.1 0.2 14676 8144 pts/0 Ss 12:41 0:00 /usr/bin/z
root 3394 0.0 0.0 0 0 ? I 12:45 0:00 [kworker/1
root 5923 0.0 0.0 0 0 ? I 12:50 0:00 [kworker/1
kali 6055 0.0 0.2 14900 9600 pts/0 SL+ 12:50 0:00 vi pippo
kali 6092 0.5 2.5 446140 102460 ? SL 12:50 0:00 /usr/bin/q
kali 6095 0.3 0.1 10224 6632 pts/1 Ss 12:50 0:00 /usr/bin/z
kali 6426 0.0 0.1 307036 7912 ? SL 12:51 0:00 /usr/lib/x
root 6677 0.0 0.0 0 0 ? I 12:51 0:00 [kworker/u
kali 6980 0.0 0.1 12536 5504 pts/1 R+ 12:52 0:00 ps -aux

(kali@kali)-[~]
$ ps -aux | grep pippo
kali 6055 0.0 0.2 14900 9600 pts/0 SL+ 12:50 0:00 vi pippo
kali 7060 0.0 0.0 6340 2304 pts/1 S+ 12:52 0:00 grep --col

(kali@kali)-[~]
$ kill 6055

(kali@kali)-[~]
$

File Actions Edit View Help
File Actions Edit View Help
kali 7529 0.2 2.5 445280 101176 ? SL 12:53 0:00 /usr/bin/q
kali 7540 0.1 0.1 10220 6504 pts/0 Ss+ 12:53 0:00 /usr/bin/z
root 7904 0.0 0.1 9972 4736 pts/0 TN 12:54 0:00 sudo firef
root 8629 0.0 0.0 0 0 ? I 12:55 0:00 [kworker/1
kali 8803 0.0 0.1 12536 5504 pts/1 R+ 12:55 0:00 ps -aux

(kali@kali)-[~]
$ ps -aux | grep firefox
root 7904 0.0 0.1 9972 4736 pts/0 TN 12:54 0:00 sudo firef
ox
kali 9297 0.0 0.0 6340 2304 pts/1 S+ 12:56 0:00 grep --col
or=auto firefox

(kali@kali)-[~]
$ ps -aux | grep firefox
root 7904 0.0 0.1 9972 4736 pts/0 TN 12:54 0:00 sudo firefox
kali 9340 0.0 0.0 6340 2304 pts/1 S+ 12:56 0:00 grep --color=auto firefox

(kali@kali)-[~]
$
```


1. Con il comando *firefox &* metto in background l'applicazione
2. Con il comando *jobs* verifico i lavori attivi nel terminale
3. Con *fg %1* metto in foreground l'applicazione che era in background
Qui il terminale si blocca e non è utilizzabile si esci con *ctrl + z*
4. Con il comando *bg %1*
5. Possiamo terminare il processo di firefox con *kill* e *pid* del processo
Oppure da terminale con *ctrl + c*.

```
(kali㉿kali)-[~]
└─$ firefox &
[1] 5860

(kali㉿kali)-[~]
└─$ ps
  PID TTY          TIME CMD
  4354 pts/0        00:00:00 zsh
  5860 pts/0        00:00:33 firefox-esr
  6013 pts/0        00:00:00 Socket Process
  6061 pts/0        00:00:00 WebExtensions
  6098 pts/0        00:00:00 Privileged Cont
  6320 pts/0        00:00:02 Isolated Web Co
  6465 pts/0        00:00:00 Web Content
  6489 pts/0        00:00:00 Web Content
  6539 pts/0        00:00:00 Web Content
  6694 pts/0        00:00:00 ps

(kali㉿kali)-[~]
└─$ jobs
[1]  + running      firefox

(kali㉿kali)-[~]
└─$ bg %1
bg: job already in background

(kali㉿kali)-[~]
└─$ bg 1
bg: job not found: 1

(kali㉿kali)-[~]
└─$ fg %1
[1]  + running      firefox
^Z
zsh: suspended  firefox

(kali㉿kali)-[~]
└─$ bg %1
[1]  + continued    firefox

(kali㉿kali)-[~]
└─$ jobs
[1]  + running      firefox

(kali㉿kali)-[~]
└─$ df -h
Filesystem      Size  Used Avail Use% Mounted on
udev            1.9G   0    1.9G   0% /dev
tmpfs           392M 1020K  391M   1% /run
/dev/sda1       79G   16G   59G  21% /
tmpfs           2.0G   0    2.0G   0% /dev/shm
tmpfs           5.0M   0    5.0M   0% /run/lock
tmpfs           392M 120K  392M   1% /run/user/1000
```

Cybersecurity Analyst 2023