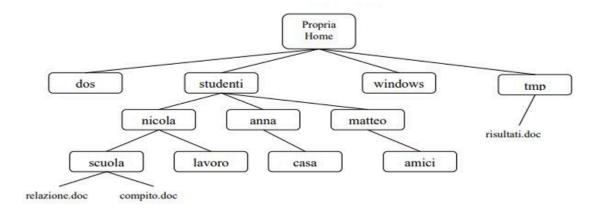
Shell Linux

In questo compito ci viene richiesta una serie di task per iniziare a prendere confidenza con la shell di linux ed i suoi comandi. La prima cosa da fare è creare una struttura di cartelle all'interno del file system come nell'immagine di seguito.



Per prima cosa ci posizioniamo nella nostra directory **/home** e da li possiamo iniziare a creare il primo livello di cartelle.

Dopodiché ci spostiamo nella directory **studenti** e creiamo le cartelle che devono stare al suo interno.

```
(kali⊕ kali)-[~]
$ cd studenti & mkdir nicola anna matteo

(kali⊕ kali)-[~/studenti]
$ ls
anna matteo nicola

(kali⊕ kali)-[~/studenti]

$ ¶
```

Successivamente ci spostiamo nella cartella **nicola** e creiamo le cartelle **lavoro** e **scuola**.

```
(kali@ kali)-[~/studenti]
$ cd nicola &6 mkdir scuola lavoro

(kali@ kali)-[~/studenti/nicola]
$ ls
lavoro scuola

(kali@ kali)-[~/studenti/nicola]

$ "
```

Adesso dobbiamo spostarci all'interno della directory **scuola** e creare i due file vuoti **relazione.doc** e **compito.doc**. Per questo scopo ho utilizzato il comando **touch** ma ci sono altri modi per creare dei file in Linux. Un'alternativa può essere quella di utilizzare un editor di testo come **Vi**, **Vim** o **Nano**.

Completato questo "lato" dell'albero torniamo indietro di due livelli (.../../), entriamo nella cartella anna e creiamo la directory casa.

```
(kali⊕ kali)-[~/studenti/nicola/scuola]
$ cd ../../anna & mkdir casa

[kali⊕ kali)-[~/studenti/anna]
$ ls
casa

[kali⊕ kali)-[~/studenti/anna]
```

Dalla cartella **anna** spostiamoci indietro di un livello. Entriamo quindi nella cartella **matteo** e creiamo la cartella **amici**.

```
(kali@ kali)-[~/studenti/anna]
$ cd ../matteo & mkdir amici

(kali@ kali)-[~/studenti/matteo]
$ ls
amici

(kali@ kali)-[~/studenti/matteo]

$ [kali@ kali]-[~/studenti/matteo]
```

Da qui saliamo di due livelli e creiamo il file vuoto **risultati.doc** all'interno della cartella **tmp**. Adesso il nostro sistema di cartelle è pronto per essere utilizzato per l'esercizio successivo.

Esercizio - Lavorare nel File System via CLI

Ti trovi nella directory **lavoro** (sotto **nicola**). Devi passare alla directory **casa** (sotto **anna**) con percorso relativo e percorso assoluto.

Path relativo.

Path assoluto.

```
(kali@ kali)-[~/studenti/nicola/lavoro]
$ cd /home/kali/studenti/anna/casa

(kali@ kali)-[~/studenti/anna/casa]
/home/kali/studenti/anna/casa

(kali@ kali)-[~/studenti/anna/casa]

$ [ kali@ kali)-[~/studenti/anna/casa]
```

A. Copia il file **compito.doc** (dalla directory **scuola**) nella directory corrente (**casa**).

B. Sposta il file **relazione.doc** nella directory corrente (**casa**).

```
(kali@ kali)-[~/studenti/anna/casa]
$ mv ../../nicola/scuola/relazione.doc .

(kali@ kali)-[~/studenti/anna/casa]
$ ls
compito.doc relazione.doc

(kali@ kali)-[~/studenti/anna/casa]
$ ls ../../nicola/scuola
compito.doc

(kali@ kali)-[~/studenti/anna/casa]
$ "
```

C. Cancella la cartella **tmp**.

```
(kali kali) - [~/studenti/anna/casa]
$ rm -rf ../../mp

(kali kali) - [~/studenti/anna/casa]
$ ls ~

Desktop Documents dos Downloads Music perl5 Pictures Public studenti Templates venv Videos windows

(kali kali kali) - [~/studenti/anna/casa]

(kali kali kali) - [~/studenti/anna/casa]
```

D. Creare il file **pippo.txt** nella cartella **lavoro**.

```
(kali@ kali)-[~/studenti/anna/casa]
$ touch ../../nicola/lavoro/pippo.txt

(kali@ kali)-[~/studenti/anna/casa]
$ ls ../../nicola/lavoro/
pippo.txt

(kali@ kali)-[~/studenti/anna/casa]
$ [
```

E. Cambiare gli attributi del file **pippo.txt** e renderlo scrivibile e leggibile solo per il proprietario, mentre per tutti gli altri solo leggibile.

```
(kali@ kali)-[~/studenti/anna/casa]
$ chmod 644 ../../nicola/lavoro/pippo.txt

(kali@ kali)-[~/studenti/anna/casa]
$ ls -l ../../nicola/lavoro/
total 0
-rw-r--r-- 1 kali kali 0 Oct 20 15:07 pippo.txt

(kali@ kali)-[~/studenti/anna/casa]

$ [
```

F. Nascondere il contenuto della cartella anna.

```
(kali@ kali)-[~/studenti/anna]

state casa

(kali@ kali)-[~/studenti/anna]

state mv casa .casa

(kali@ kali)-[~/studenti/anna]

state ls

(kali@ kali)-[~/studenti/anna]

state ls

drwxrwxr-x 3 kali kali 4096 Oct 20 15:13 .
drwxrwxr-x 5 kali kali 4096 Oct 20 14:54 ...
drwxrwxr-x 2 kali kali 4096 Oct 20 15:05 .casa

(kali@ kali)-[~/studenti/anna]
```

G. Spostarsi nella cartella lavoro e visualizzare il contenuto del file pippo.txt.

```
(kali@ kali)-[~/studenti/anna]
$ cd ../nicola/lavoro

(kali@ kali)-[~/studenti/nicola/lavoro]
$ cat pippo.txt

(kali@ kali)-[~/studenti/nicola/lavoro]
$ [
```

H. Rimuovere la cartella amici.

I. Rimuovere tutte le cartelle precedentemente create.

```
Ckali@kali)-[~]
Desktop Documents dos Downloads Music perl5 Pictures Public studenti Templates venv Videos windows

(kali@kali)-[~]
$ rm -rf dos studenti windows

(kali@kali)-[~]
Besktop Documents Downloads Music perl5 Pictures Public Templates venv Videos

(kali@kali)-[~]

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

Esercizio Facoltativo - Processi

Provare i comandi w, who e whoami.

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

15:18:40 up 34 min, 1 user, load average: 0.00, 0.00, 0.00
USER TTY FROM LOGIND IDLE JCPU PCPU WHAT
kali - 14:44 0.00s ? lightdm -- session-child 13 24

[(kali@ kali)-[~]
who
kali seat0 2025-10-20 14:44 (:0)

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

Processi

- 1. Aprire un terminale.
- 2. Leggere il manuale del comando **jobs** (non disponibile) , **ps** e **kill**.

```
(kali⊗ kali)-[~]
$ man job
No manual entry for job

(kali⊗ kali)-[~]
$ man jobs
No manual entry for jobs

(kali⊗ kali)-[~]
$ [
```

```
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 BSO options, which may be grouped and must be used with a dash.
3 GAU long options, which may be grouped and must not be used with a dash.
4 GAU long options, which may be grouped and must not be used with a dash.
5 When the process 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.

By default, ps selects all processes with the same effective user ID (euid=EUID) as the current user and associated with the same terminal as the invoker. It displays the process ID (pid=PID), the terminal associated with the process (tname=TTY), the cumulated CPU time in [DD-]bhrsm:ss format (time=TIME), and the executable name (ucund=CMD) output is unsorted by default.

The use of BSD-style options will add process state (stat=STAT) to the default display and show the command args (args=COMMAND) instead of the executable name. You can override this with the Ps_FORMAT environment variable. The use of BSD-style options will also change the process selection to include processes on other terminals (TrYs) that are owned by you; alternately, this may be described as setting the selection to be the set of all processes filtered to exclude processes owned by you alternately, this may be described as setting the selection to be the set of all processes filtered to exclude processes below, and other users of not on a terminal. These effects are not considered when options are described as being 'identical' below, so -M will be considered identical, or and on on.

Except as described below, pro
```

```
NAME

kill - send a signal to a process

SYNOPSIS

kill [options] 
pto featult 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, -$IGKILL or -KILL. Negative PID values may be used to choose whole process groups; see the PGID column in ps command output. A PID of -l is special; it indicates all processes except the kill process it self and init.

oPTIONS

cpid> [...]
Send signal to every <pid> listed.

-s<signal>
-s<signal>
-s<signal>
-signal (7) manual page.

-q, -queue value
Use signaled of the signal to the sent. The signal can be specified by using name or number. The behavior of signals is explained in signal(7) manual page.

-l, -list [signal]
List signal names. This option has optional argument, which will convert signal number to signal name, or other way round.

-l, -table
Nanual page kill(1) line 1 (press h for help or q to quit)
```

3. Lanciare il comando vi pippo.

```
The Action Est View Help

Trippo" [New]
```

4. Aprire un nuovo terminale e visualizzare tutti i processi.

```
      (kali⊕ kali)-[~]

      $ ps -a

      PID TTY
      TIME CMD

      21739 pts/0
      00:00:00 vi

      21907 pts/1
      00:00:00 ps

      (kali⊕ kali)-[~]
```

5. Killare il processo vi per sbloccare il terminale precedente.

```
      (kali⊕ kali)-[~]

      $ ps -a
      PID TTY
      TIME CMD

      21739 pts/0
      00:00:00 vi
      22390 pts/1

      21739 pts/0
      00:00:00 ps

      (kali⊕ kali)-[~]
      *

      $ ps -a
      PID TTY
      TIME CMD

      22481 pts/1
      00:00:00 ps

      (kali⊕ kali)-[~]
      *

      * (kali⊕ kali)-[~]
      *
```

6. Lanciare il comando **firefox** in background.



7. Cercare di terminare il processo **firefox**.

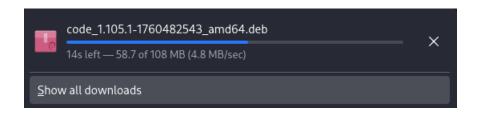
```
-(kali⊕kali)-[~]
  PID TTY
                    TIME CMD
22943 pts/1
               00:00:04 firefox-esr
               00:00:00 Socket Process
23000 pts/1
23028 pts/1
               00:00:00 Privileged Cont
23069 pts/1
               00:00:00 file:// Content
23097 pts/1
               00:00:00 WebExtensions
23167 pts/1
               00:00:00 Utility Process
23175 pts/1
               00:00:00 Web Content
23179 pts/1
               00:00:00 Web Content
23185 pts/1
               00:00:00 Web Content
23562 pts/1
               00:00:00 ps
 -(kali⊕kali)-[~]
-$ kill 22943
```

8. Verificare quanto spazio si sta occupando su disco.

```
-(kali⊛kali)-[~]
Filesystem
                Size
                      Used Avail Use% Mounted on
udev
                1.9G
                         0 1.9G
                                   0% /dev
tmpfs
                392M
                     992K 391M
                                   1% /run
/dev/sda1
                47G
                     15G 30G 33% /
                                   1% /dev/shm
tmpfs
                2.0G
                     4.0K
                            2.0G
tmpfs
                5.0M
                         0
                            5.0M
                                   0% /run/lock
tmpfs
                1.0M
                         0
                           1.0M
                                   0% /run/credentials/systemd-journald.service
tmpfs
                2.0G
                      556K 2.0G
                                   1% /tmp
tmpfs
                1.0M
                         0 1.0M
                                   0% /run/credentials/getty@tty1.service
                                   1% /run/user/1000
tmpfs
                392M
                     116K 392M
  -(kali% kali)-[~]
```

Esercizio - Installare Visual Studio Code su Kali

Per prima cosa scarichiamo il file relativo alla nostra distribuzione Linux dal sito ufficiale.



Successivamente installiamo il pacchetto di installazione con il comando mostrato nella schermata seguente:

```
(kali kali) - [~/Downloads]
$ sudo dpkg -i code_1.105.1-1760482543_amd64.deb
[sudo] password for kali:
Selecting previously unselected package code.
(Reading database ... 413800 files and directories currently installed.)
Preparing to unpack code_1.105.1-1760482543_amd64.deb ...
Unpacking code (1.105.1-1760482543) ...
```

Installiamo anche il debugger GDB.

```
(kali® kali)-[~/Downloads]
$ sudo apt update & sudo apt install gdb
Get:1 https://packages.microsoft.com/repos/code stable InRelease [3,590 B]
Get:2 https://packages.microsoft.com/repos/code stable/main amd64 Packages [19.9 kB]
Get:3 http://kali.download/kali kali-rolling InRelease [34.0 kB]
Get:4 http://kali.download/kali kali-rolling/main amd64 Packages [20.9 MB]
30% [4 Packages 12.4 MB/20.9 MB 59%]
```

Questo comando aggiornerà il repository e installerà la versione più recente di **GDB**. Fatto questo possiamo avviare Visual Studio Code per verificare che tutto sia correttamente installato.

