A drawing of a face

Description automatically generated

**FINTECH**

**Unità Formativa (UF): Linux Server**

**Docente: Wolfgang Cecchin**

**Titolo argomento: Comandi essenziali**

**Programma odierno:**

* **Leggere e scrivere files;**
* **Muoversi nelle directories;**
* **Creare utenti;**
* **Substitute User e Sudo;**
* **Installare pacchetti**

**La shell**

<https://it.wikipedia.org/wiki/Shell_Unix>

<https://en.wikipedia.org/wiki/Unix_shell>

Esistono vari tipi di shell:

sh, zsh, bash…

Muoversi agevolmente con la shell:

Il tab fornisce l’autocomplete dei comandi noti;

la freccia in su serve per scorrere la history dei comandi (vedi anche il comando “history”);

CTRL + D effettua il logout;

CTRL + C interrompe un comando;

q serve per uscire dalla modalità lettura di alcuni programmi (man);

~ : è la home directory.

Il comando:

$ clear # “pulisce” il terminale

**Struttura dei comandi in Linux**

La struttura di un comando in Linux è:

**Comando + opzioni + parametri**

Es:

**ls** = list

**ls -lah** =

list

- l = long listening form

- a = all

- h = human readable

**ls -lah /home =**

list

- l = long listening form

- a = all

- h = human readable

/home = della directory home

**ls -lah /home** = fai l’elenco del contenuto **(ls)** della directory home **(/home)**, usando la modalità estesa **(long listening form)**, mostrando tutti i file **(all**), in modo che siano comprensibile da un utente umano

Opzioni brevi

**$ ls -lah /home**

Opzioni in forma estesa

**$ ls -l --all --human-readable /home**

Manuali ed help dei comandi

**$ man ls**

**$ ls --help**

**Creare file e reindirizzare l’input verso un file**

$ touch MioFile

$ echo “ciao mondo\n” > mioFile

$ echo “ciao mondo\n” >> mioFile

$ ls > output\_ls

**Leggere un file**

$ cat MioFile

**Scrivere un file**

$ nano MioFile

Oppure (cuoriosità):

$ vi MioFile

(:i per scrivere

:w per salvare

:q per uscire

:wq per scrivere e salvare

:q! per forzare l’uscita)

**Creare directories, Copiare, rinominare, cancellare un file**

$ mkdir miaDir #creo una directory

$ cp source destination # copia

$ cp -R source destination # copia una directory (copia ricorsiva)

$ mv oldName newName # rinomino un file

$ rm file # cancella il file

$ rm -r directory # rimuove una directory e il suo contenuto

L’opzione -f forza (per lo più evitare di usarla).

**Muoversi nel file system**

$ cd # change directory

$ pwd # quale è la directory corrente?

Simboli “speciali”

. = è la directory stessa;

.. = è la directory superior;

/ = è la directory principale del sistema (la root);

~ = è la home dell’utente (~ è la tilde; su linux usare i tasti: alt gr + ì )

$ cd / # vado nella directory root del sistema

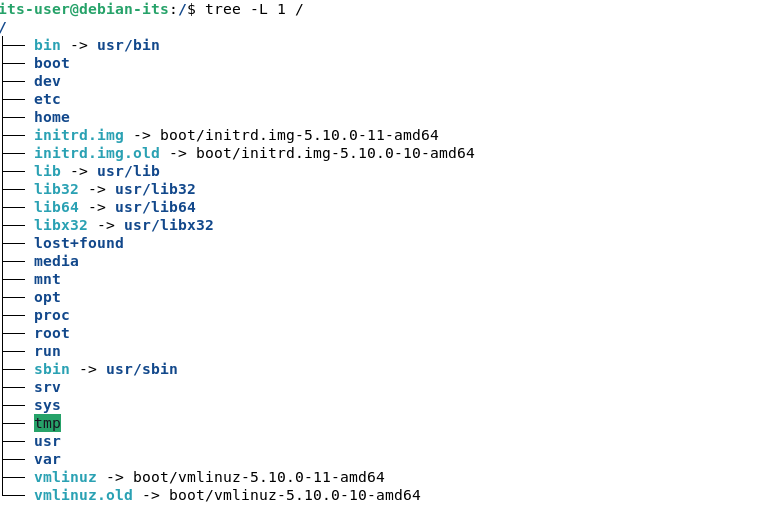
$cd - # torno alla directory precedente

$ tree -L 1 / # vedo l’alberatura del sistema

**FHS Filesystem Hierarchy Standard**

**Riferimenti:**

<https://en.wikipedia.org/wiki/Filesystem_Hierarchy_Standard>



Attenzione nelle moderne distribuzioni alcuni directories “puntano” ad altre.

**Gli utenti**

**root** (individuato con #):è l’amministratore del sistema

**su = substitute user**

**sudo =** “substitute user do” (“super user do”); esegue il comando per un altro utente-

E’ necessario aggiungere l’utente in /etc/sudoers.d

Creare un file con il nome utente e aggiungere all’’interno:

user   ALL=(ALL:ALL) ALL

* The first ALL is the users allowed to use the sudo command.
* The second ALL defines the hosts (servers) on which sudo can be employed.
* The third ALL is the user you are running the command as.
* The last ALL defines the commands allowed.

**Nota:** la home di root è /root

Attenzione non confondere:

l’utente root;

la home di root che è /root;

la root directory che è /.

Aggiungere utenti:

**# useradd**

**# useradd -D alberto**

**# useradd -d /var/marco marco**

**# groupadd**

File che gestiscono gli utenti

Il file /etc/passwd

Il file /etc/group

Il file /etc/shadow

Il file /etc/shadow

**Cambiare password di un utente**

Cambiare la propria password:

$ passwd

Cambiare la password di un utente:

# passwd marco

**Link simbolici**

$ ls -l file nomeLink

**Comando tar (“tape archive”)**

Creazione di un archivio:

$ tar **cvf** nomeMioArchivio.tar directory1 directory2 dirctory3 file1 file2…

Estrazione di un archivio:

$ tar **xvf** nomeMioArchivio.tar directory1 directory2 dirctory3 file1 file2…

**Creazione di un archivio compresso - compressore gzip – opzione z :**

$ tar c**z**vf nomeMioArchivio**.tar.gz** directory1 directory2 dirctory3 file1 file2…

Estrazione di un archivio:

$ tar x**z**vf nomeMioArchivio.tar.gz directory1 directory2 dirctory3 file1 file2…

**Creazione di un archivio compresso(compressore bz2 – opzione j:**

$ tar c**j**vf nomeMioArchivio.**tar.bz2** directory1 directory2 dirctory3 file1…

Estrazione di un archivio:

$ tar x**j**vf nomeMioArchivio.tar.bz2 directory1 directory2 dirctory3 file1…

**Installazione pacchetti in Debian**

Directory di configurazione per i repositories (lo vedremo per il PHP)

/etc/apt.d

Per aggiornare i repositories:

# apt update

Per installare i pacchetti:

# apt install

Attenzione!

Usare con molta cautela:

# apt upgrade

Effettua l’upgrade dei pacchetti disponibili