

# LINUX

Linux è un sistema operativo di tipo Unix. La principale differenza rispetto ai "veri" sistemi Unix consiste nel fatto che Linux può essere copiato liberamente e gratuitamente, compreso l'intero codice sorgente.

Il termine Linux, per essere precisi, si riferisce solo al kernel. Il kernel è il nucleo essenziale di un sistema operativo che si occupa della gestione della memoria, dei processi e del controllo dell'hardware e che essenzialmente permette a tutti gli altri programmi di girare. La maggior parte delle persone, comunque, quando parlano di LINUX, intendono un sistema operativo completo comprensivo delle applicazioni che ci girano sopra, alternativo a Microsoft Windows® o Apple MacOS®.

Cio' che rende Linux così diverso è il fatto di essere cresciuto grazie all'aiuto di moltissimi ed eterogenei gruppi di programmatori UNIX ed esperti di Sistemi Operativi che hanno messo a disposizione il codice da loro prodotto, gratuitamente. Questa eterogeneità si riferisce sia alle competenze tecniche e al settore di provenienza che alla dislocazione geografica. Affinché queste comunità di programmatori potessero lavorare insieme era necessario un efficiente strumento di comunicazione. Lo strumento fu Internet ed essendo Linux il sistema operativo scelto da queste persone, significa che gli strumenti e le utility necessarie ad utilizzare la rete furono tra le prime a comparire. Oltre ad essere sviluppate appositamente per Linux, molte delle applicazioni furono portate da quello che era il miglior software disponibile in quel momento sul mercato.

In 7 anni Linux è diventato il sistema operativo più evoluto tecnicamente dell'intera categoria e - a causa della disponibilità dei sorgenti - è diventato il punto di riferimento per lo sviluppo del kernel da parte di Microsoft .

## **Caratteristiche principali di Linux**

Attualmente Linux offre:

- Multitasking: possibilità di eseguire più programmi contemporaneamente
- Funzionamento multiutente: uso contemporaneo della macchina da parte di più utenti

- Multipiattaforma: Linux è praticamente compatibile con tutti i comuni hardware di PC, e può girare anche su processori non INTEL (Digital Alpha, Sun Sparc, Mips, Motorola, PowerPC)
- funzioni di protezione della memoria tra processi
- gestione della memoria virtuale attraverso la paginazione
- un pool di memoria unificato per i programmi e la memoria cache
- librerie statiche e dinamiche
- utilizza un proprio tipo di file system in cui i file possono avere nomi composti da non più di 255 caratteri e dimensioni fino a 2 Gbyte, con file system fino a 4 Tbyte
- Linux permette di accedere a file system diversi: MS-DOS, VFAT (Windows 9x con nomi lunghi), HPFS (OS/2 2.x), NTFS (Windows NT - sola lettura), HFS (Apple Mac), FFS (Amiga), Minix, NFS (file system di rete), SMB (file system delle reti Windows) e così via
- supporto per diverse tastiere internazionali
- supporto per un'ampia gamma di protocolli di rete (TCP/IP, SLIP, PPP, etc...)

## **Un pò di storia**

La prima versione del kernel Linux (la numero 0.0.1) è stata messa a disposizione su Internet da Linus Torvalds (Helsinki) nel settembre del 1991. Da allora Linux è stato

sviluppato a un ritmo impressionante da migliaia di programmatori in tutto il mondo . Ciò che all'inizio era una prerogativa degli hacker, si stima che nel 2002 abbia superato i 10 milioni di utenti (anche se fare una stima corretta è molto difficile).

Un nuovo sistema operativo come Linux non sarebbe mai stato prodotto senza una comunicazione tra programmatori a livello internazionale via Internet.

Una delle principali ragioni per cui Linux ha potuto essere sviluppato così rapidamente sta sia nelle nuove possibilità di comunicazione che Internet ha reso disponibile a tutti, sia nel software che era già reperibile gratuitamente a quell'epoca. Linux non è nato dal nulla, ma è basato su un'ampia piattaforma di software libero.

Nel 1985 Richard Stallman, ex ricercatore del MIT, fondò la Free Software Foundation (FSF), finanziata per donazioni, alla cui base c'era il progetto GNU (GNU's Not Unix), che ambiva alla realizzazione di un sistema operativo free, nel senso più esteso del termine.

Il software free, rilasciato con la GNU General Public Licence (GPL), in questo contesto, implica:

- Ridistribuzione libera del software e del codice
- Codice sorgente aperto e disponibile per lettura, modifiche, analisi, controllo
- Prodotti derivati da prodotti GPL devono avere licenza GPL
- Copyright dell'autore

Il gran numero di programmi GNU che erano in circolazione da molto più tempo di Linux fu un fattore che risultò (e risulta tuttora) fondamentale per lo sviluppo di Linux.

## **La shell**

Se il kernel è il nucleo, la shell è il guscio. Per shell si intende l'interfaccia (testuale) tramite la quale l'utente può operare sul sistema.

La shell è un programma che gestisce la comunicazione fra utente e sistema operativo interpretando ed eseguendo i comandi dell'utente. Può avere diversi utilizzi:

- Uso interattivo: il sistema attende i comandi digitati dall'utente;

- Configurazione della propria sessione: si possono definire variabili e parametri che vengono utilizzati in ogni interazione dell'utente con la macchina;
- Programmazione: utilizzando comandi di sistema e funzionalità della shell è possibile realizzare piccoli programmi (script shell) in grado di automatizzare operazioni e reagire ad eventi.

La shell è qualcosa di simile all'interprete dei comandi del DOS, ma è molto più potente e versatile. In Linux sono disponibili diverse shell, ma la più diffusa è la bash.

## Gestione dei file

Per quanto riguarda la gestione dei file, alcune differenze fondamentali di Linux rispetto al DOS (e a Windows), sono le seguenti:

- I nomi dei file possono raggiungere la lunghezza massima di 255 caratteri
- Il carattere di separazione delle directory è il carattere "/" anziché il carattere "\"
- Linux distingue tra lettere minuscole e maiuscole (un file che ad esempio si chiama pippo.txt è diverso da un file denominato Pippo.txt nella stessa directory)
- I file (e directory) i cui nomi cominciano con un punto sono considerati file nascosti (generalmente sono utilizzati per contenere i parametri di configurazione del software personalizzati per ciascun utente)
- L'estensione dei file è spesso utile per determinare il tipo del file, ma non ha significato vincolante per il sistema operativo: i file eseguibili, generalmente, non hanno estensione

- I file e le directory hanno un proprietario (l'utente che ha creato il file) e un gruppo di appartenenza
- Come in Unix, i file e le directory hanno dei permessi: permessi di lettura, di scrittura, e di esecuzione. Tali permessi sono suddivisi in tre tipi (per ciascun file): i permessi concessi al proprietario del file, i permessi concessi al gruppo, i permessi concessi a tutti gli altri utenti
- Solo il proprietario del file (e l'amministratore di sistema) può modificare i permessi di un file
- Per poter lanciare un programma, occorre avere permessi di esecuzione sul file che contiene il programma
- Per poter accedere al contenuto di una directory, occorre avere permessi di lettura ed esecuzione sulla directory

Linux inoltre presenta vaste possibilità nell'uso dei caratteri jolly, utili quando si lavora su gruppi di file.

## Comandi Linux

### ***cd***

*cd [DIRECTORY]*

*cd* entra nella directory specificata.

Se non è fornita alcuna directory allora entra nella home directory dell'utente.

**Esempio** *cd dir1* entra nella directory *dir1*

### ***cp***

*cp [OPZIONE...] FILE\_ORIGINE... DIRECTORY*

*cp [OPZIONE...] FILE\_ORIGINE FILE\_DESTINAZIONE*

Copia i file specificati con *[FILE\_ORIGINE...]*

nella directory *DIRECTORY*.

Copia il `FILE_ORIGINE` nel `FILE_DESTINAZIONE`.

### **opzioni**

`-R, --recursive` copia ricorsivamente directory e sottodirectory

`-v, --verbose` mentre copia i file visualizza in uscita i loro nomi

### **esempi**

```
cp bianca.txt neve.txt /home/skywalker
```

```
cp claudia.old claudia.new
```

## ***ls***

***ls*** [*OPZIONE...*] [*NOME\_DIRECTORY...*] [*NOME\_FILE...*]

*ls* lista il contenuto di ogni directory indicata

in [*NOME\_DIRECTORY...*] e i file individuati da [*NOME\_FILE...*].

### **opzioni**

`-a, --all` elenca tutti i file compresi quelli che iniziano con `.` (i file nascosti).

`-l, --format=long`

per ogni voce della lista fornisce anche le seguenti informazioni:

tipo di file, permessi, numero di hard link, nome del possessore, nome del gruppo, dimensione, momento di ultima modifica.

`-R, --recursive` discende ricorsivamente anche tutte le sottodirectory

`-t, --sort=time` elenca gli elementi della lista vengono elencati in ordine temporale, per primi quelli modificati più di recente

## ***mkdir***

***mkdir*** [*OPZIONE...*] *NOME\_DIRECTORY...*

*mkdir* crea le directory specificate con (*NOME\_DIRECTORY...*).

**esempio** `mkdir dir1`

## ***mv***

***mv*** [*OPZIONE...*] *FILE\_ORIGINE...* *DIRECTORY*

***mv*** [*OPZIONE...*] *FILE\_ORIGINE* *FILE\_DESTINAZIONE*

Sposta i file specificati con [*FILE\_ORIGINE...*] nella directory *DIRECTORY*.

Rinomina il file FILE\_ORIGINE: il nuovo nome sarà FILE\_DESTINAZIONE.

**esempi** mv jane.png giungla.png /home/tarzan  
mv libro.txt libro.old

## **rm**

*rm [OPZIONE...] NOME\_FILE...*

rm rimuove i file specificati, non però le directory.

**esempio** rm pippo.txt

## **rmdir**

*rmdir [OPZIONE...] NOME\_DIRECTORY...*

rmdir rimuove le directory specificate, a patto che siano vuote.

**esempi** rmdir dir1

## **less**

*less [OPZIONE...] [NOME\_FILE...]*

less riporta in uscita gli ingressi specificati una schermata alla volta. è simile al comando `more`, ma ha più possibilità; ad esempio oltre a poter avanzare, si può anche tornare indietro. Viene generalmente utilizzato quando si vuole velocemente prendere visione di un lungo file di testo oppure in abbinamento con altri comandi, quando essi forniscono in uscita informazioni che non hanno spazio su di una singola schermata.

Sono possibili le seguenti operazioni attivate dalla pressione del tasto indicato all'inizio della riga:

ENTER -> avanza di una riga

BARRA SPAZIATRICE -> avanza di una schermata

B -> torna indietro di una schermata

Q -> esci

**esempio** less ulisse.txt

## **more**

*more [OPZIONE...] [NOME\_FILE...]*

more riporta in uscita gli ingressi specificati una schermata alla volta.

La pressione del tasto ENTER permette di avanzare una riga alla volta, la pressione della barra spaziatrice invece fa avanzare di una schermata alla volta. Per uscire prima della fine degli ingressi è sufficiente la pressione del tasto Q o di CTRL-C.

**esempio** `more divina_commedia.txt`

### ***lpr***

`lpr [OPZIONE...] [NOME_FILE]`

Il file di ingresso viene mandato nella coda di stampa.

### ***pwd***

`pwd` visualizza il nome della directory corrente con il percorso completo.

### ***man***

`man NOME...`

`man` visualizza le eventuali pagine del manuale che riguardano `NOME...`. `NOME` può essere il nome di un comando, il nome di una funzione, il nome di un file di configurazione (solo pochi sono documentati) o altro.

**esempi** `man ls` visualizza la pagina di manuale di `ls`