

Una breve guida su Linux/Ubuntu

Guida ad Ubuntu/Linux

Basato su Ubuntu 20.04.1 LTS
(Focal Fossa)

Andrea Aralla

Sommario

1. Informazioni sulla distribuzione	2
2. Un primo sguardo a Ubuntu.....	4
3. Installazione di Ubuntu.....	4
4. Il primo approccio	8
Aggiornamento della lingua di sistema.....	8
5. Personalizzare il desktop	10
Cambiare lo sfondo	10
Cambiare il tema grafico.....	10
6. File, directory e volumi	11
7. Uso del terminale.....	13
8. i comandi più utilizzati da terminale.....	15

1. INFORMAZIONI SULLA DISTRIBUZIONE

Ubuntu Linux è una distribuzione GNU/Linux che negli ultimi anni è entrata a far parte della schiera delle “grandi” distribuzioni. Ubuntu è infatti particolarmente indicata per l’utilizzo in ambiente desktop e ha degli automatismi mai invasivi che la rendono sobria e funzionale allo stesso tempo.

Ubuntu deve la sua fortuna a Debian. Debian è la roccia sulla quale Ubuntu è costruita, come recita il manifesto di Ubuntu. Ed infatti da Debian, Ubuntu acquisisce moltissimo, innanzitutto la struttura dei pacchetti ed il gestore `apt-get`, semplice e potente.



Figura 1 Logo Ubuntu

Ubuntu è un’antica parola africana che vuol dire, più o meno, **Universo di persone**, inteso come concetto di sostegno e valorizzazione tra esseri umani. Una parola di questo tipo ricalca molto bene l’etica dell’open source e del progetto GNU, ed infatti la distribuzione si pone degli obiettivi molto “umani”, come:

- Rendere il software disponibile gratuitamente;
- Rendere disponibili gli strumenti nella lingua madre degli utenti;
- Dare agli utenti la libertà di modificare e personalizzare il software.

Ma le analogie con Debian non sono solo etiche, ma sono soprattutto tecniche; Ubuntu, infatti:

- viene costruita sulla versione “sid” (non stabile) di Debian ed ogni sei mesi viene rilasciata una nuova versione, con programmi aggiornati (e relativi aggiornamenti di sicurezza), tra i quali l’ultima versione di GNOME (il rilascio semestrale coincide con questo progetto);
- è formata da una comunità di sviluppatori (alcuni anche stipendiati) che lavora sia su Ubuntu che su Debian, o che comunque segnala prontamente al progetto Debian correzioni e migliorie;

Per gli utenti che non l’hanno mai provata, Ubuntu può essere considerata come una Debian Desktop, tutti i vantaggi di Debian (filosofia, gestore di pacchetti, impostazione) ma con molti accorgimenti per il desktop (programmi aggiornati, ambiente desktop di ultima generazione, utili automatismi).

Esistono delle varianti di Ubuntu che mantengono la struttura base ma hanno impostazione o target differenti:

- **Kubuntu:** identica ad Ubuntu solo che utilizza l'ambiente desktop KDE, invece di GNOME;
- **Xubuntu:** altra variante che utilizza Xfce come ambiente desktop. Indicata per computer poco potenti;
- **Edubuntu:** variante "educativa" pensata per i bambini e gli adolescenti;
- **Ubuntu lite:** pensata per PC "obsoleti", utilizza IceWM come Window Manager;

Ora che sapete cosa è Ubuntu, non resta altro che scaricarla e passare all'installazione ed iniziare così ad usare questa eccezionale distribuzione.

2. UN PRIMO SGUARDO A UBUNTU

Per poter avere un sistema Ubuntu completamente funzionante, la prima cosa da fare è ovviamente scaricare la distribuzione. Ubuntu è disponibile sul [sito ufficiale](#) in due versioni, una desktop ed una server.

Per gli scopi di questa guida tratteremo ovviamente l'installazione e l'uso della versione desktop, mentre, per dovere di cronaca, avvertiamo che il sistema server non differisce poi molto (come installazione).

L'immagine del CD di installazione della versione desktop deve essere, quindi, scaricato dal sito di Ubuntu.

3. INSTALLAZIONE DI UBUNTU

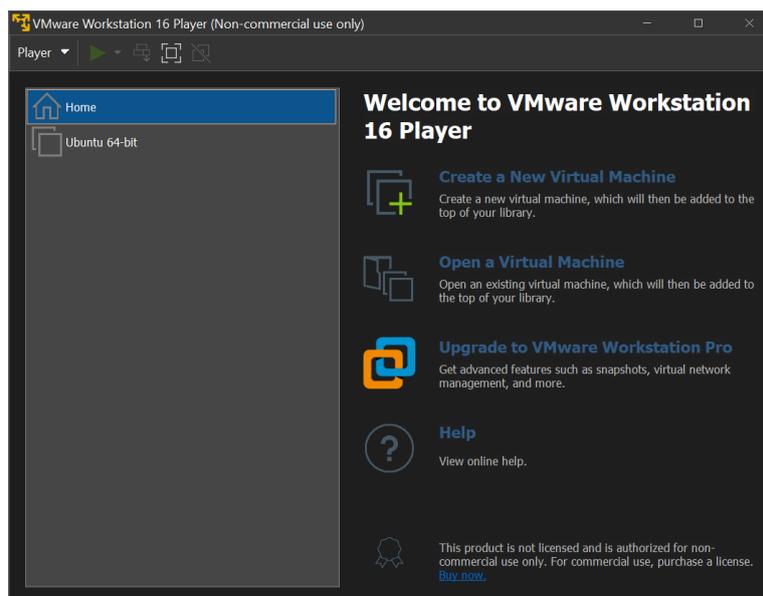
Terminato il download del file ISO di Ubuntu dovete **masterizzarlo** su un CD o creare un boot da pen drive per poter installare la distribuzione.

Esiste una terza via ed è quella di utilizzare una macchina virtuale su cui "installare" questo sistema operativo. In questo caso eviteremo di utilizzare il file ISO ma ci serviremo di un file immagine dell'hard disk contenete il sistema operativo già installato.

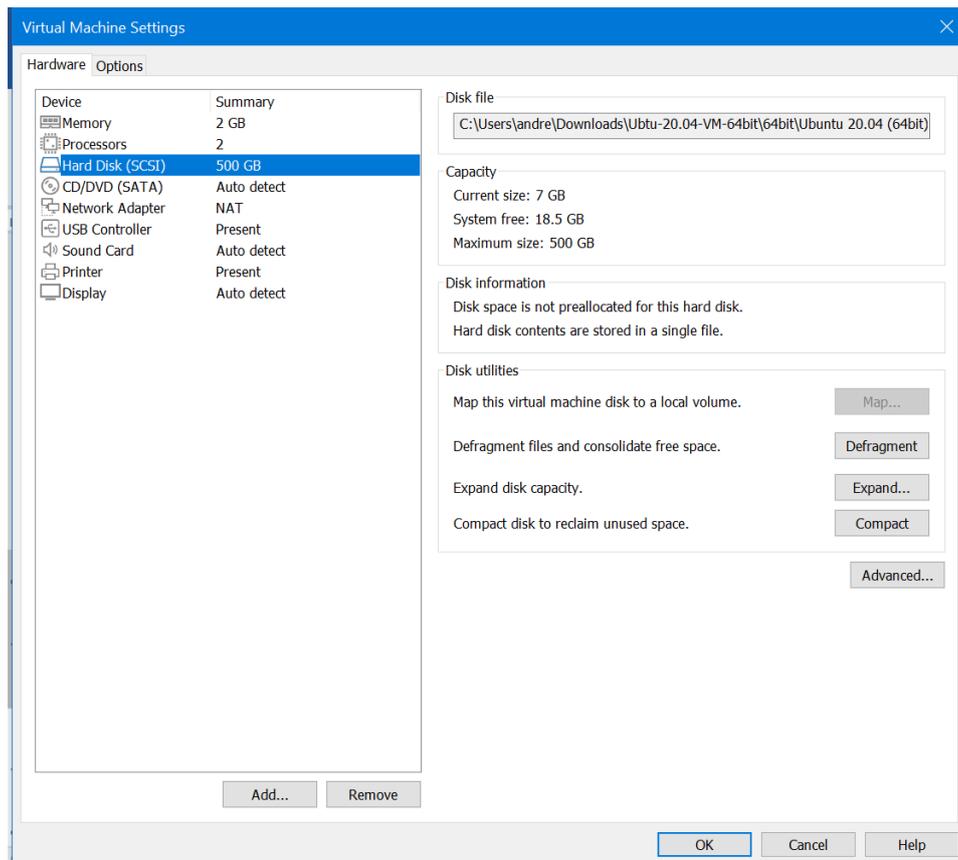
Per motivi di semplicità d'uso utilizzeremo il software di virtualizzazione VM Player scaricabile dal seguente sito:

<https://www.vmware.com/it/products/workstation-player.html>

- L'installazione di questo software è assolutamente intuitiva.
- Creiamo una macchina virtuale UBUNTU a 64 bit



- Modifichiamo i settaggi della macchina sostituendo l'hard disk.



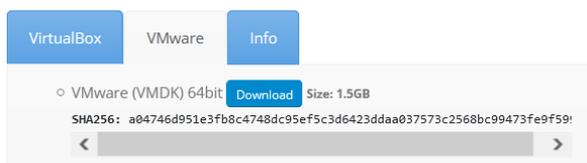
- A questo punto sar  necessario dotarla di un Sistema Operativo funzionante. Anche in questo caso ce la caveremo a buon mercato tramite un sito web, www.osboxes.org, che distribuisce sistemi operativi pronti all'uso.

www.osboxes.org

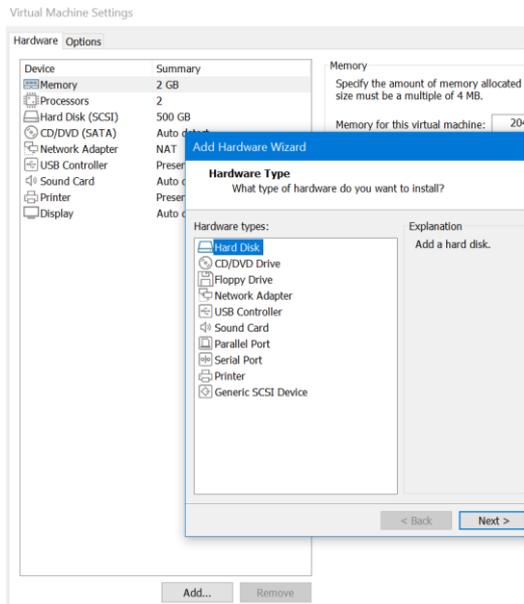


- Scarichiamo l'ultima versione. Si tratta di un hard disk virtuale contenente il Sistema Operativo Linux/Ubuntu.

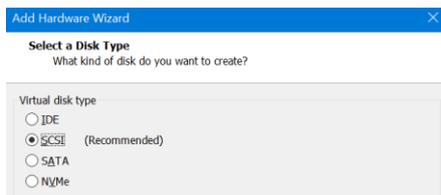
Ubuntu 20.04 Focal Fossa



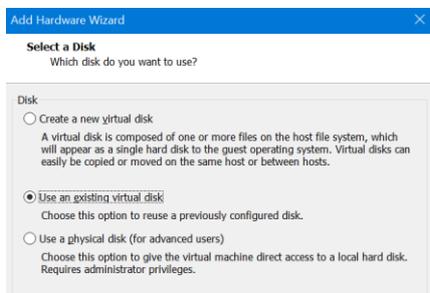
- Infine sostituiamo l'hard disk della macchina virtuale appena creata con il nuovo hard disk



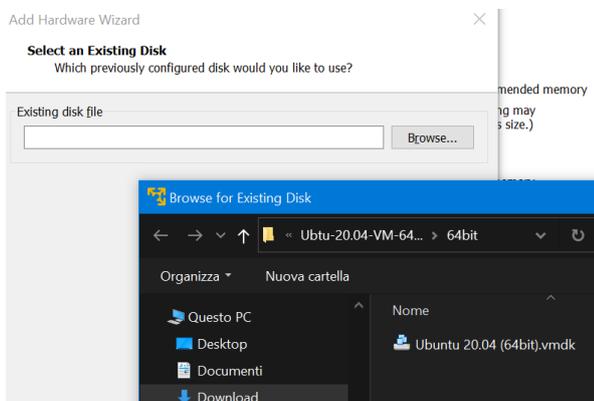
- Tipo di hard disk SCSI



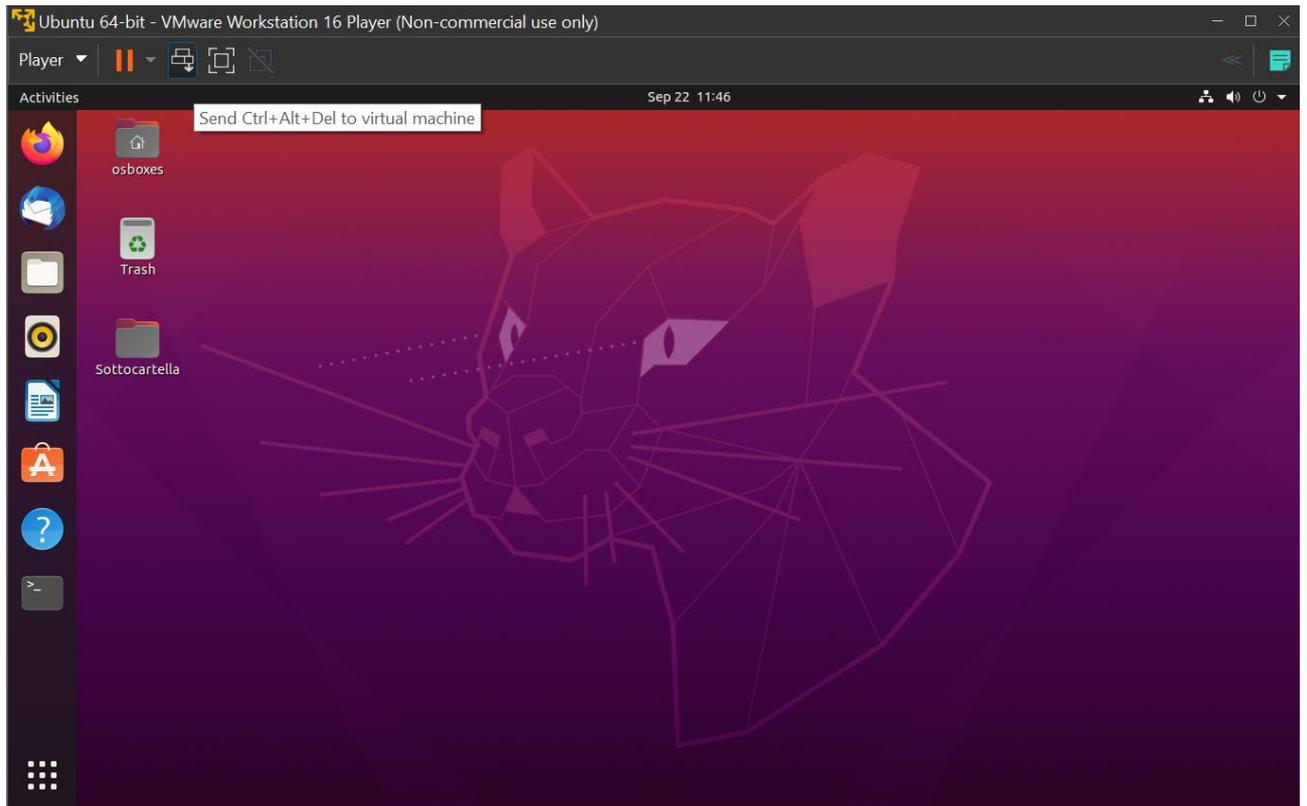
- Selezioniamo un hard disk già esistente (quello che abbiamo appena scaricato da osboxes)



- Selezioniamo il file Ubuntu 20.04 (64bit).vmdk

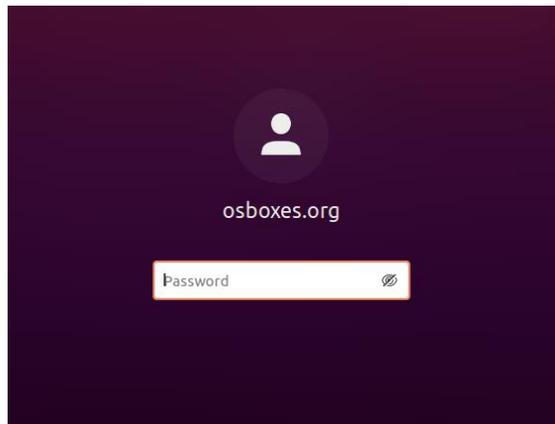


- Avviamo la macchina virtuale e dopo pochi istanti potremo usare il sistema operativo.



4. IL PRIMO APPROCCIO

Il primo avvio di Ubuntu non è velocissimo, in quanto vengono impostati i parametri iniziali di molti programmi, tuttavia dopo pochi minuti dovrete vedere apparire il **gestore di login** di Ubuntu (il cui nome è GDM). Da questo gestore potete decidere di arrestare o riavviare il sistema, oppure, come è presumibile, di entrare nell'ambiente desktop. Per farlo basterà inserire il nome utente e la password scelta in fase di avvio.



Se la login è corretta, si avvia **GNOME**, che è l'ambiente desktop predefinito di Ubuntu. Come ogni installazione di GNOME, è presente il menu che consente di accedere ai programmi installati, alle risorse e alle funzionalità di sistema (amministrazione, pannello di controllo ed help). In alto a destra, invece, è presente l'orologio, il gestore dell'audio e la tray-bar (quella che contiene alcuni programmi in esecuzione, proprio come accade in Windows).

Avete notato sicuramente che la scheda audio, la scheda video e praticamente **tutto il vostro hardware** non solo è stato correttamente riconosciuto da Ubuntu, ma è stato anche configurato in maniera ottimale. La dimostrazione la sia ha proprio quando si accede per la prima volta a GNOME e (generalmente) si può udire la intro sonora e si può verificare che la risoluzione è già impostata su quella massima per la scheda video (e relativo monitor).

AGGIORNAMENTO DELLA LINGUA DI SISTEMA

Una delle prime operazioni da compiere in un sistema Ubuntu appena installato è l'**aggiornamento della lingua di sistema**. Se in fase di installazione è stata scelta la lingua Italiana, ma per vari motivi non sono stati scaricati i pacchetti di localizzazione, è necessario farlo. In pratica tutti i programmi al momento installati (GNOME compreso) sono in lingua inglese, dovrete quindi installare i file di localizzazione in Italiano di GNOME e di altri programmi (quali Firefox e OpenOffice.org).

Cliccate sulla barra del menu, quindi scegliete *System (Sistema)*, poi *Administration (Amministrazione)* ed infine *Language Support (Supporto Lingue)*.

Il desktop si oscurerà e vi verrà chiesta la **password** per ottenere i privilegi di amministratore. La password che dovete inserire è la vostra password utente e non quella di root, in quanto l'utente root su Ubuntu non esiste e non esiste, quindi, nemmeno una password associata. Comunque dopo

aver inserito la password ed aver premuto invio, si aprirà la finestra di gestione delle lingue del sistema.

Scorrete la lista fino a quando non trovate la lingua "Italiano", mettete il segno di spunta cliccando nell'apposito checkbox ed impostate la lingua come predefinita. Il programma, in automatico, dovrebbe chiedervi se desiderate installare i file da Internet e dopo un vostro assenso, provvederà a scaricare ed installare i programmi necessari.

Dopo una decina di minuti dovrete aver portato a termine correttamente questa fase, cliccate quindi su *Sistema > Esci > Termina Sessione*, per tornare al gestore di login, rientrate e verificate che il sistema sia in lingua italiana.

5. PERSONALIZZARE IL DESKTOP

Una delle prime esigenze, cui si deve imparare a sopperire, quando si ha un ambiente desktop nuovo è di poter **personalizzare l'aspetto e l'impostazione grafica** dell'ambiente stesso, per renderlo più piacevole e più funzionale alle proprie abitudini. Ubuntu, come ogni distribuzione GNU/Linux, permette un elevato grado di personalizzazione, sia estetica che funzionale.

CAMBIARE LO SFONDO

Per cambiare lo sfondo, cliccate con il tasto destro sul desktop, quindi selezionate IMPOSTA SFONDO SCRIVANIA. Si aprirà una finestra in cui viene mostrata la lista degli sfondi disponibili, per poterne aggiungere uno potete seguire diverse strade. Potete aprire una finestra di Nautilus e trascinare un file immagine all'interno della finestra, e poi scegliere se volete l'immagine CENTRATA, SCALATA, INGRANDITA, AFFIANCATA oppure che RIEMPIE LO SCHERMO.

Il **gestore degli sfondi** di GNOME ha due particolarità: la prima è che consente di utilizzare immagini vettoriali SVG, formato supportato da molto tempo in Linux, il quale consente un notevole "alleggerimento" delle immagini; la seconda particolarità è che un'immagine ha comunque un "colore base" e se questa immagine contiene delle trasparenze, queste mostreranno un colore in funzione del colore base impostato (in pratica potete avere sfondi basati su trasparenze cui cambiare colore con un click). Ovviamente potete decidere di utilizzare come sfondo una tinta unita o un gradiente (orizzontale o verticale) di due colori.

CAMBIARE IL TEMA GRAFICO

Linux, e soprattutto Ubuntu, non ha niente da invidiare ad altri sistemi operativi per quanto riguarda la personalizzazione grafica. Si possono cambiare, infatti, lo stile dei controlli, il bordo delle finestre ed il set di icone. Per accedere alle PREFERENZE DEL TEMA, basta cliccare su SISTEMA > PREFERENZE > TEMA.

Dicevamo poco fa che il livello di personalizzazione è molto elevato. Lo **stile dei controlli** consente di cambiare l'aspetto delle forme e dei colori, sia delle barre che dell'interno delle finestre. Per capirsi, lo sfondo "grigio" delle finestre può essere cambiato con altre tonalità o addirittura altri colori, così come per le voci di menu evidenziate. Cambiare i controlli ha effetto soprattutto sui colori "dominanti" dell'ambiente grafico, il **bordo delle finestre** eredita, infatti, la sua tonalità di colore proprio dal colore dominante per i controlli.

Quando si **cambia il bordo della finestre**, questo, a meno di alcune eccezioni, modifica solo la forma e lo stile del bordo, ad esempio se utilizzare una X per chiudere, piuttosto che un altro simbolo. Infine il cambio delle icone, invece, condiziona la grafica di tutto il sistema, in quanto le icone vengono usate sia per gli oggetti sul desktop, quelli del menu di sistema e quelli delle eventuali icone presenti nei programmi (come Apri, Salva, ecc.).

Per scaricare temi, icone e sfondi, potete visitare Gnome-Look.org e Kde-Look.org in cui sono disponibili risorse per le più disparate esigenze.

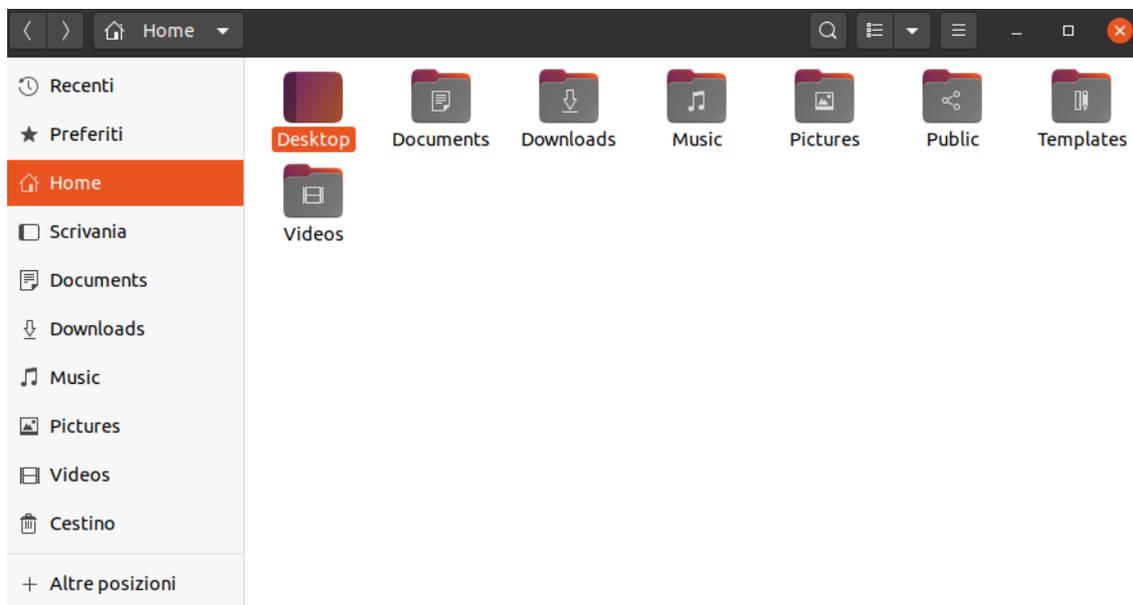
6. FILE, DIRECTORY E VOLUMI

Linux si differenzia da Windows non solo per la filosofia ed i programmi, ma anche per l'**impostazione del sistema**. In Linux tutti gli elementi del sistema, compresi i dispositivi (dischi, cd) e le periferiche, vengono trattati come dei file. Questa impostazione può sembrare strana ai più, tuttavia Linux eredita questa caratteristica da Unix, di cui ne ricalca la struttura.

Il problema principale è che **il sistema non viene identificato da "C:"**, tantomeno i dischi esterni avranno lettere come "D:" e simili.

In Linux esiste la directory di root ("/") che contiene al suo interno tutti i file e tutti i collegamenti non solo verso i dati fisici ma anche verso tutti i dispositivi e le periferiche. In questo modo il lettore CD, ad esempio, sarà accessibile tramite il percorso `/media/cdrom` o `/mnt/cdrom` e potrà essere letto solo se appositamente "montato", cioè solo se viene fatta richiesta esplicita di lettura (e quindi di associazione dispositivo-file) dello stesso.

In Ubuntu, ma soprattutto in Gnome, il **comportamento di Linux è fortunatamente automatizzato** ed in parte mascherato così da renderlo più amichevole per l'utente poco esperto. Cliccando su COMPUTER viene mostrata, infatti, una lista dei dispositivi presenti, di quelli montati (es. altre partizioni) e di quelli da montare (es. il lettore CD). Se si inserisce un CD, ad esempio, e si clicca due volte sull'icona corrispondente, questo verrà montato e sarà quindi possibile leggere i dati al suo interno.



Per capire quali sono i dischi e come questi vengono associati ad eventuali percorsi nel sistema, dovete entrare in SISTEMA > AMMINISTRAZIONE > DISCHI e, dopo aver inserito la vostra password cliccare sui vari dispositivi presenti. Vi verrà mostrata una prima parte informativa che contiene, oltre alle informazioni sul modello, la periferica associata (generalmente `/dev/hda` per dischi Eide e `/dev/sda` per dischi Scsi o S-Ata). Cliccando su partizioni potrete, invece, verificare in quante

partizioni (e quindi relativi file di dispositivo come `/dev/hda1`, `/dev/hda2`, e così via) è suddiviso il disco e gli eventuali mount-point associati (es. `/media/cdrom`).

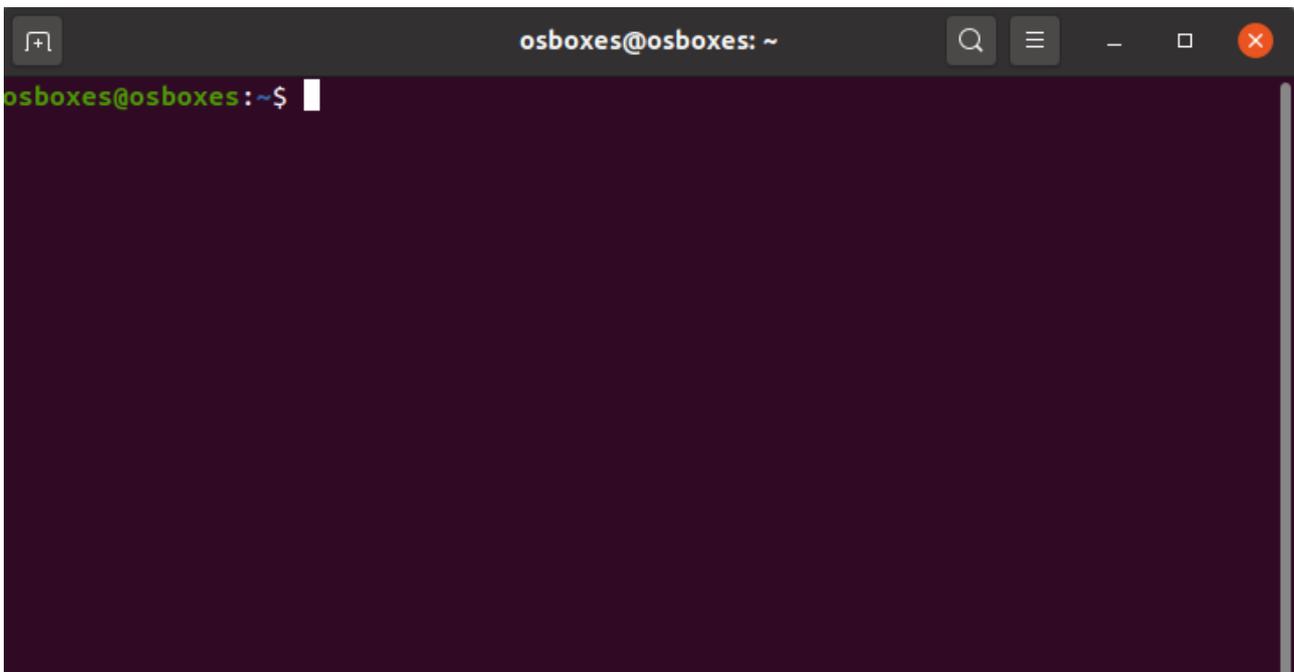
La **gestione dei dischi** di Ubuntu è talmente avanzata che se viene inserito un CD o una memoria esterna (es. USB o una Memory Card), questa viene automaticamente montata e viene resa disponibile un'icona sul desktop. Inoltre si possono associare diverse azioni a seconda del supporto inserito; se si inserisce un CD vuoto o una macchina fotografica, si può far partire il programma di masterizzazione piuttosto che il gestore di immagini.

Per espellere, o comunque staccare, eventuali dispositivi, si deve necessariamente "smontarlo". In Gnome basta cliccare con il tasto destro sull'icona corrispondente, quindi scegliere ESPELLI. Se non si può effettuare l'operazione, generalmente la risorsa è ancora occupata (es. se vi si stanno copiando dei file), oppure è stata montata da un altro utente (es. viene montata automaticamente in fase di avvio).

7. USO DEL TERMINALE

Sebbene Ubuntu Linux permetta di modificare le impostazioni del sistema e di operare senza dover mai passare dal terminale, è opportuno fornire qui un piccolo “manuale” sull’**utilizzo del terminale in Ubuntu**. Questo perché può sempre tornarvi utile se in futuro dovrete impartire comandi dal terminale per installare, ad esempio, un modem ADSL USB o qualcosa del genere.

Una delle prime cose da sapere in Ubuntu è come **impartire comandi di root** dal terminale. In Ubuntu, come abbiamo già detto, l’utente root è disabilitato, quindi per acquisire i poteri di amministratore potete seguire altre strade. Innanzitutto aprite il terminale andando su *Applicazioni > Accessori > Terminale*.



```
utente@ubuntu:~$
```

La “stringa” che vi verrà mostrata è molto significativa, in quanto utente rappresenta il nome vostro nome utente e, ovviamente, è quello con il quale state al momento lavorando, ubuntu, invece, è l’**hostname del vostro computer**, l’identificativo univoco con il quale il computer viene visto in rete; la tilde (~) sta ad indicare che vi trovate nella vostra cartella home, quella analogamente rappresentata sul desktop, mentre il simbolo del dollaro (\$) significa che state operando come un utente normale, con privilegi “limitati” che non consentono di modificare impostazioni o file di sistema.

Supponiamo che vogliate **cambiare la data e l’ora** del vostro computer ed impostarla, ad esempio al 1° Ottobre 2006, ore 12.00, potete farlo con il comando `date` e le cifre che identificano il mese, il giorno, l’ora e l’anno.

```
utente@ubuntu:~$ date 10011200
```

```
date: impossibile impostare la data: Funzione non permessa
```

Il messaggio di errore è dovuto al fatto che l'ora di sistema può essere cambiata **solo dall'utente root** (l'amministratore del computer), ed è per questo che si ricorre al comando `sudo`:

```
utente@ubuntu:~$ sudo date 10011200
Password: inserite la vostra password utente
dom ott 1 12:00:00 CEST 2006
```

Il comando `sudo`, anteposto ad un qualsiasi comando, consente di eseguire quel comando con i **privilegi di root**; per acquisire temporaneamente questi privilegi si deve inserire la propria password utente e premere invio. Ovviamente se si lascia il terminale aperto e dopo un paio di minuti si vuole eseguire un altro comando utilizzando `sudo`, non sarà necessario fornire di nuovo la password, in quanto la password rimane valida per circa cinque minuti dall'ultimo utilizzo del comando.

Se però, come spesso accade, si sta seguendo una guida e si danno dei comandi come utente root con poca frequenza, si dovrà inserire la **password ad ogni utilizzo**, oppure nel caso si debbano utilizzare molti comandi, può risultare scomodo dover anteporre sempre `sudo` ad ogni digitazione. È per questo che è possibile passare ad una shell di root, sempre utilizzando il comando `sudo`:

```
utente@ubuntu:~$ sudo -s
root@ubuntu:~#
```

Come potete notare, la **sintassi della shell** è cambiata un po', nel senso che il vostro nome utente è stato sostituito da root, ad indicare che la shell in cui siete in questo momento è quella con i privilegi di amministratore, inoltre l'ultimo carattere non è il simbolo del dollaro, ma bensì lo "sharp" (#), o cancelletto che dir si voglia. Il simbolo del dollaro ed il cancelletto rappresentano, spesso, nelle guide con quali privilegi deve essere fornito quel comando, se come utente normale o come utente root (utilizzando uno dei due metodi che `sudo` mette a disposizione).

A dire la verità Ubuntu **non segue** quella che è la regola di altre distribuzioni, cioè abilitare fin dall'inizio l'utente root. Se lo ritenete necessario, ad esempio perché lavorate molto con questo utente, o semplicemente perché ritenete che ciò consenta una gestione più pulita, potete abilitare l'utente root, con i seguenti comandi:

```
utente@ubuntu:~$ sudo passwd root
Password: inserite la vostra password utente
Enter new UNIX password: inserite la nuova password di root
Retype new UNIX password: ripetete la nuova password di root
```

A questo punto potrete "entrare" nell'utente root semplicemente digitando `su` e inserendo la password di root (e non la vostra password utente come con il comando `sudo`)

```
utente@ubuntu:~$ su
Password: inserite la password di root
root@ubuntu:~#
```

8. I COMANDI PIÙ UTILIZZATI DA TERMINALE

FILES E DIRECTORIES

pwd

visualizza il nome della directory corrente

```
$ pwd
/home/anubis/anuload
```

ls [opzioni] [file]

visualizza il contenuto della directory corrente o della directory specificata. Il parametro -l consente di avere una visione estesa della directory.

```
$ ls
ANU01266.sql ANU01269.sql temp
$ ls -lrt
total 32
drwxrwxrwx 3 anuload anubis 96 Mar 19 2004 temp
-rw-r----- 1 dsadm dsadm 3697 Oct 15 2007 ANU01269.sql
-rw-r----- 1 dsadm dsadm 2348 Oct 15 2007 ANU01266.sql
```

parametri

-l: directory estesa

-r: rovescia l'ordine di ordinamento in modo da visualizzare prima i file più vecchi.

-t: visualizza la directory corrente con ordinamento per data.

cd [path]

consente di cambiare la directory corrente.

cd .. si sposta nella directory superiore

```
$ pwd
/home/anubis/anuload
$ cd ..
$ pwd
/home/anubis
```

cd [nome_dir]

si sposta nella directory [nome_dir] se presente

```
$ cd anuload
$ pwd
/home/anubis/anuload
```

mkdir [opzioni] [nome dir]

crea una directory

rmdir [opzioni] [nome dir]

elimina una directory. La directory deve essere vuota

mv [opzioni] [file vecchio] [file nuovo]

\$ mv pippo.sh /home/pippo/pippo.sh

muove il file pippo.sh dalla directory corrente a quella specificata

\$ mv pippo.sh pluto.sh

rinomina il file pippo.sh in pluto.sh, se pluto.sh esiste verrà sovrascritto perdendone il contenuto

cp [opzioni] [file vecchio] [file nuovo]

copia file vecchio in file nuovo; file nuovo non dovrebbe già esistere. Questo comando crea o sovrascrive file nuovo

rm [opzioni] file

rimuove (elimina) un file o una directory

-r elimina ricorsivamente la directory ed il suo contenuto

-i chiede conferma prima di eliminare i file

cat [opzioni] file/files

mostra e/o concatena files

cat pippo.sh

mostra a video il contenuto del file pippo

cat pippo.sh > topolino.sh

trasferisce il contenuto di pippo.sh su topolino.sh

cat pippo.sh pluto.sh paperino.sh > topolino.sh

concatena i tre files e reindirige l'output verso il file topolino.sh

more [opzioni] [file]

mostra files di grandi dimensioni suddividendoli per pagine

file [opzioni] [file]

consente di conoscere il tipo di file

\$ file pippo.sh

pippo.sh: Bourne shell script text executable

PERMESSI

chmod [opzioni] [permessi] [files/directories]

cambia i permessi su un file o una directory

chmod 777 ANU01266.sql cambia i permessi con full control (777) per owner e gruppo e lettura/esecuzione per gli altri per il file ANU01266.sql

```
$ ls -lrt ANU01266.sql
```

```
-rw-r----- 1 dsadm dsadm 2348 Oct 15 2007 ANU01266.sql
```

```
$ chmod 777 ANU01266.sql
```

```
$ ls -lrt ANU01266.sql
```

```
-rwxrwxrwx 1 dsadm dsadm 2348 Oct 15 2007 ANU01266.sql
```

0755 – imposta lettura, scrittura ed esecuzione per il proprietario, e lettura ed esecuzione per gruppo ed altri (comune per le directory e per i file eseguibili);

0700 – imposta lettura, scrittura ed esecuzione per il proprietario, nessun permesso per il gruppo e per gli altri (comune per le directory private);

0600 – imposta lettura e scrittura per il proprietario, e nessun permesso per il gruppo e per gli altri (comune per file di dati personali);

0644 – imposta lettura e scrittura per il proprietario, solo lettura per il gruppo e nessun permesso per gli altri (comune per file di dati da condividere);

+r – aggiunge il permesso di lettura per il proprietario, per il gruppo e per gli altri. Non rimuove altri permessi già presenti;

-x – rimuove il permesso di esecuzione a proprietario, gruppo e altri. Non rimuove altri permessi già presenti;

ug=rx – imposta il permesso di lettura ed esecuzione al proprietario, rimuovendo eventuali permessi di scrittura già presenti; non altera gli i permessi per gli altri;

u=rx,go= – imposta il permesso di lettura ed esecuzione al proprietario, rimuovendo un eventuale permesso di scrittura già presente; rimuove tutti i permessi esistenti per il gruppo e per gli altri.

RICERCHE E ORDINAMENTI

grep [opzioni] [stringa] [files/directories]

consente di trovare all'interno di un file o in una directory i files che contengono una determinata stringa

find [espressioni]

```
find ./ -name "*.sh"
```

cerca a partire dalla directory corrente tutti i file che hanno come estensione .sh

sort [opzioni] [file/files]

ordina l'output del file, ordinato per righe.

```
sort -f topolinia.txt paperopoli.txt > disney_ordinato.txt
```

ordina le linee dei files topolinia.txt e paperopoli.txt indirizzandole su un file chiamato disney_ordinato.txt

OPERAZIONI SUL SISTEMA

finger

fornisce l'elenco degli utenti collegati al server nel momento in cui lo si lancia

```
> finger
```

```
Login Name TTY Idle When Bldg. Phone
```

```
anu_load ??? *0 Tue 15:17
```

```
anu_load ??? *1 4:09 Tue 11:18
```

```
anu_load ??? *2 6 Tue 11:23
```

```
dsadm ??? *4 22 Tue 14:46
```

who

fornisce l'elenco degli utenti collegati al server nel momento in cui lo si lancia

```
> who
```

```
anu_load pts/0 Aug 12 15:17
```

```
anu_load pts/1 Aug 12 11:18
```

```
anu_load pts/2 Aug 12 11:23
```

```
dsadm pts/4 Aug 12 14:46
```

w

elenco degli utenti collegati al sistema

```
> w
```

```
3:24pm up 98 days, 2:43, 4 users, load average: 1.80, 1.67, 1.61
```

```
User tty login@ idle JCPU PCPU what
```

```
anu_load pts/0 3:17pm w
```

```
anu_load pts/1 11:18am 4:05 -ksh
```

```
anu_load pts/2 11:23am 1 -ksh
```

```
dsadm pts/4 2:46pm 17 -ksh
```

ps

elenco dei processi attivi in esecuzione sul server

```
> ps
```

```
PID TTY TIME COMMAND
```

```
7544 pts/0 0:00 ksh
```

```
7578 pts/0 0:00 ps
```

top

elenco dei processi che consumano più risorse.

kill [opzioni] [idprocesso]

è usato per interrompere forzatamente un processo;

vediamo come funziona:

```
> kill -9 7544
```

in questo caso verrà forzata la chiusura del processo con PID = 7544 cioè ksh (vedere comando ps).

Verrà chiusa immediatamente la console!

date

restituisce la data del sistema

```
> date
```

```
Tue Aug 12 15:16:03 METDST 2008
```

hostname

visualizza l'indirizzo esteso della macchina

```
> hostname
```

```
h3mih193
```

host [opzioni][ip/nome macchina]

converte il nome della macchina nel corrispondente indirizzo numerico e viceversa.

ping [opzioni][ip/nome macchina]

consente di verificare se una macchina è raggiungibile.

history [opzioni]

restituisce la lista degli ultimi comandi utilizzati dall'utente collegato.

```
history
```

```
1 finger
```

```
2 sudo
```

```
3 sudo chmod 777 *.*
```

```
4 last
```

```
5 history
```

last

mostra l'elenco degli ultimi utenti che hanno effettuato la login al sistema

```
itdevadm ftp Wed Mar 5 16:16 - 16:19 (00:02)
```

```
itdevadm ftp Wed Mar 5 16:16 - 16:16 (00:00)
```

```
dsadm ftp Wed Mar 5 16:16 - 17:06 (1+00:49)
```

```
dsadm ftp Wed Mar 5 16:16 - 16:16 (00:00)
```

```
dsadm ftp Wed Mar 5 16:16 - 16:19 (00:02)
```

```
dsadm ftp Wed Mar 5 16:16 - 17:06 (1+00:50)
```

```
dsadm ftp Wed Mar 5 16:16 - 17:06 (1+00:50)
```

```
dsadm pts/1 Wed Mar 5 16:13 - 16:20 (00:07)
```

```
ravellr ftp Wed Mar 5 16:13 - 16:18 (00:05)
```

script [nome file]

scrive su file tutte le attività che vengono effettuate da console.

nohup [comando] [argomenti]

lancia un applicativo in background che non cadrà al logout dell'utente.

su [opzioni] [nome utente]

serve per cambiare utente se usato con un parametro; senza parametro serve per assumere l'identità di root.

sudo [opzioni] [nome utente] [comando]

permette di eseguire comandi con i permessi di un altro utente, tipicamente usato per eseguire comandi che soltanto root potrebbe lanciare; al momento del lancio dei comandi viene richiesta la password di root.

per esempio vediamo cosa succede se vogliamo forzare e assegnare dei permessi dove non potremo:

```
sudo chmod 777 *.*
```

We trust you have received the usual lecture from the local System Administrator. It usually boils down to these three things:

#1) Respect the privacy of others.

#2) Think before you type.

#3) With great power comes great responsibility.

Password: _

ovviamente una chiamata al nostro amico amministratore di sistema potrebbe essere utile in questo caso...

3.5 AIUTI

man [opzioni] [comando]

visualizza la pagina di aiuto per il comando specificato

```
man grep
```

visualizza la descrizione del comando grep

9. I PERMESSI DEI FILE SU LINUX

Una caratteristica del file system sul sistema operativo Linux sono i permessi sui files.

TIPI DI PERMESSI SU LINUX

Ogni file ha **tre tipi di permessi**

- Permesso di **lettura** (r). Dove la r sta per read. È il permesso che consente di vedere il contenuto.
- Permesso di **scrittura** (w). Dove la w sta per write. È il permesso di scrivere o modificare il contenuto del file.
- Permesso di **esecuzione** (x). Dove la x sta per eXecute. È il permesso di eseguire il file. In genere, si tratta di un programma eseguibile.

A questi si aggiunge il simbolo del trattino (-) che indica la mancanza del permesso.

COME VEDERE I PERMESSI SU LINUX

Per vedere la lista dei permessi dei files, entro in una directory qualsiasi di Linux.

Poi digito il comando `ls -l` sulla linea di comando.

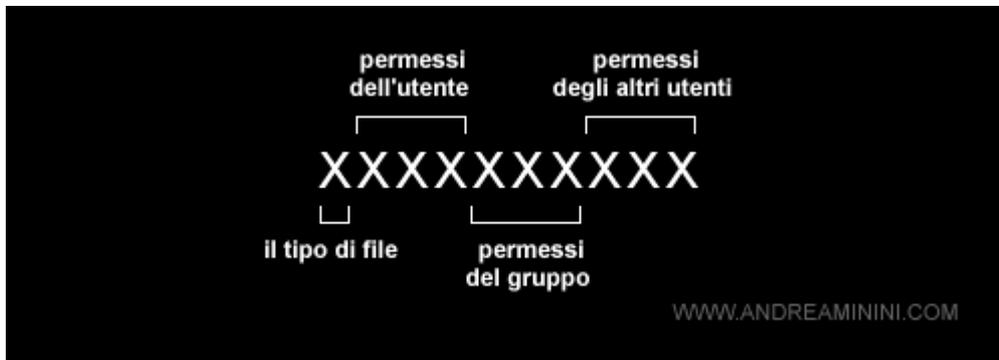
```
$ ls -l
totale 124
drwxr-xr-x 10 root root 4096 nov 24 13:36 local
drwxr-xr-x  2 root root 4096 nov 24 14:00 games
drwxr-xr-x  5 root root 4096 nov 24 14:01 src
drwxr-xr-x 23 root root 4096 nov 24 14:02 include
drwxr-xr-x 152 root root 12288 dic 27 19:22 lib
drwxr-xr-x  2 root root 69632 dic 27 19:22 bin
drwxr-xr-x 314 root root 12288 dic 27 19:22 share
drwxr-xr-x  2 root root 12288 dic 27 19:22 sbin
```

i permessi sui file WWW.ANDREAMININI.COM

COME FUNZIONANO I PERMESSI SU LINUX

La stringa dei permessi di un file è composta da 10 caratteri.

Ogni posizione della stringa è associata a una specifica informazione o permesso a seconda del carattere che compare (drw-).

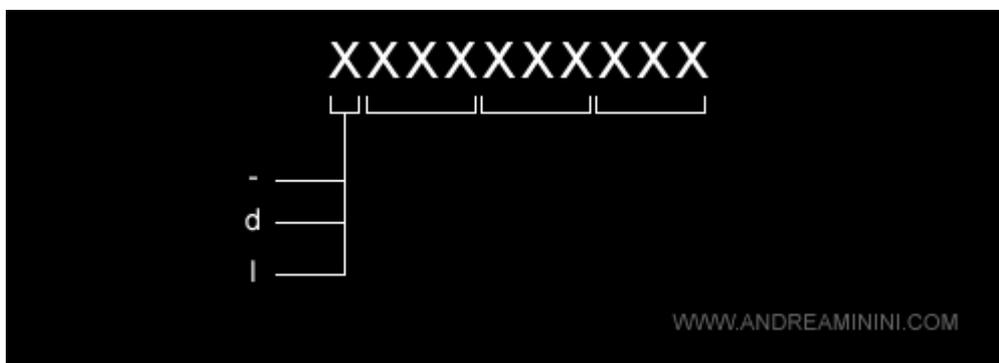


I permessi possono essere associati a tre categorie di utenti:

- L'utente (u). Dove la lettera u significa user.
- Il gruppo (g). Dove la lettera g significa group.
- Gli altri utenti (o). Dove la lettera o significa others.

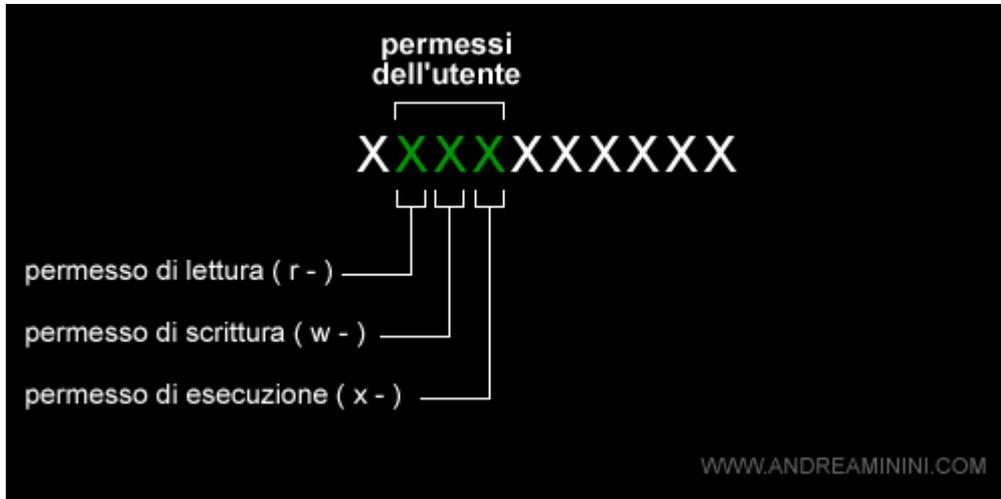
Il tipo di file

La prima posizione indica la tipologia del file (file normale, directory, link).



I PERMESSI DELL'UTENTE

I tre caratteri dalla seconda alla quarta posizione indicano i permessi dell'utente a cui appartiene il file.

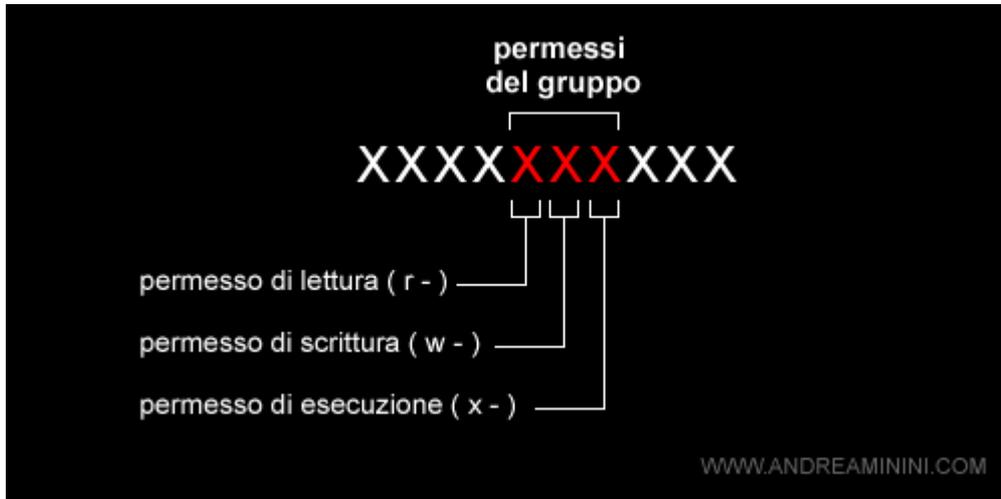


Ogni posizione è associata a un particolare permesso:

- Il permesso di lettura (r). La seconda posizione indica il permesso di lettura (r) o meno (-) sul file da parte dell'utente proprietario.
- Il permesso di scrittura (w). La terza posizione indica il permesso di scrittura (r) o meno (-) sul file da parte dell'utente.
- Il permesso di esecuzione (x). La quarta posizione indica il permesso di scrittura (r) o meno (-) sul file da parte dell'utente.

I PERMESSI DEL GRUPPO

I tre caratteri dalla quinta alla settima posizione sono i permessi del gruppo associato al file.



Nota. Ogni utente deve essere associato almeno a un gruppo principale. E' comunque molto frequente associare un utente a più gruppi, a seconda delle funzioni che deve svolgere sul sistema.

I PERMESSI DEGLI ALTRI UTENTI

I tre caratteri dall'ottava alla decima posizione sono i permessi degli altri utenti (others) sul file.



UN ESEMPIO PRATICO

Nella cartella /usr viene visualizzato il seguente elenco di files.

```
$ ls -l
totale 124
drwxr-xr-x 10 root root 4096 nov 24 13:36 local
drwxr-xr-x  2 root root 4096 nov 24 14:00 games
drwxr-xr-x  5 root root 4096 nov 24 14:01 src
drwxr-xr-x 23 root root 4096 nov 24 14:02 include
drwxr-xr-x 152 root root 12288 dic 27 19:22 lib
drwxr-xr-x  2 root root 69632 dic 27 19:22 bin
drwxr-xr-x 314 root root 12288 dic 27 19:22 share
drwxr-xr-x  2 root root 12288 dic 27 19:22 sbin
```

i permessi sui file WWW.ANDREAMININI.COM

- Il primo file della lista, quello chiamato "local", ha la seguente stringa dei permessi:

```
drwxr-xr-x
```

- Ora cerchiamo di capire il significato dei permessi. La lettera "d" nella posizione iniziale mi fa capire che si tratta di una directory.

```
drwxr-xr-x
```

- La combinazione "rwx" nei permessi degli utenti indica che l'utente può leggere, scrivere ed eseguire il files.

```
drwxr-xr-x
```

- La combinazione "r-x" nei permessi del gruppo vuol dire che gli altri utenti del gruppo associato al file possono leggere ed eseguire il files ma non modificarlo.

```
drwxr-xr-x
```

- Lo stesso vale per tutti gli altri utenti in quanto anche ai permessi degli others è associata la combinazione "r-x". Possono leggere ed eseguire il file ma non modificarlo.

```
drwxr-xr-x
```

COME CAMBIARE I PERMESSI SU LINUX

Per modificare i permessi di un file o di una directory sul sistema operativo Linux si utilizza il comando `chmod`.

```
chmod [ugoa] [+ -=] [rwx] nomefile
```

Il comando è l'abbreviazione di `change modality`. Si chiama così perché i permessi sono anche detti modalità.

COME FUNZIONA CHMOD

Il primo parametro `[ugoa]` di `chmod` indica a chi devo cambiare i permessi. Posso scegliere tra i seguenti caratteri:

- u (user)
- g (group)
- (others)
- a (all)

Nel secondo parametro devo scegliere se aggiungere (+), rimuovere (-) o impostare (=) i permessi.

- + (aggiungi permesso)
- - (rimuovi permesso)
- = (imposta permesso)

A questo punto, nel terzo parametro devo indicare il tipo di permesso:

- r (lettura)
- w (scrittura)
- x (esecuzione)

Vedremo un sistema molto più semplice per cambiare i permessi, il sistema numerico.

I PERMESSI NUMERICI

Sul sistema operativo Linux, e in generale su Unix, ogni file è associato a dei permessi di lettura (r), scrittura (w) ed esecuzione (x).

COME CALCOLARE IL PERMESSO NUMERICO SU LINUX

La stessa informazione può essere rappresentata in forma numerica, sostituendo per ogni posizione il carattere (rwx) con 1 e il carattere (-) con 0.

Ad esempio, tralasciando la prima posizione, la stringa

```
rwx rwx rwx
```

Posso riscrivere la stringa in modo numerico nel seguente modo:

```
111 111 111
```

Ogni componente della sequenza è un numero binario.

decimale	binario	r	w	x	stringa dei permessi	significato
7	111	1	1	1	rwx	lettura, scrittura, esecuzione
6	110	1	1	0	rw-	lettura, scrittura
5	101	1	0	1	r-x	lettura, esecuzione
4	100	1	0	0	r--	lettura
3	011	0	1	1	-wx	scrittura, esecuzione
2	010	0	1	0	-w-	scrittura
1	001	0	0	1	--x	esecuzione
0	000	0	0	0	---	nessun permesso

Quindi, scrivere 777 oppure rwxrwxrwx è la stessa cosa. Ho così ottenuto la rappresentazione numerica del permesso.

Al posto della stringa dei caratteri "rwxrwxrwx" scrivo il numero 777.

```
chmod 777 nomefile
```

ESEMPI PRATICI

Ecco alcuni esempi pratici di utilizzo dei permessi numerici con chmod

Chmod 700

Se volessi autorizzare soltanto l'utente alla lettura, scrittura ed esecuzione, dovrei scrivere nel seguente modo:

```
chmod 700 nomefile
```

Cmod 760

Per autorizzare anche il gruppo alla lettura e scrittura (110) devo scrivere il numero 6 nella seconda cifra.

```
chmod 760 nomefile
```

Chmod 644

Per autorizzare l'utente alla lettura e scrittura (110), il gruppo e gli altri utenti soltanto alla lettura (100) devo scrivere la modalità 644.

```
chmod 644 nomefile
```