

# **Passare da Windows a Linux**

*La guida di Etnablog*

<http://etnablog.altervista.org>

*link diretto alla guida:*

*<http://etnablog.altervista.org/linux/passare-da-windows-a-linux-prima-parte/>*

*aggiornata al 22/08/2007*

## Passare da Windows a Linux (Prima parte)

In questa guida c'è l'indispensabile per usare la distribuzione **Ubuntu Feisty 7.04** e anche qualcosina in più.

Per la maggior parte delle cose può essere usata anche per altre distribuzioni di linux. In tal caso considerate che il percorso di alcuni file potrebbe essere diverso da distribuzione a distribuzione.

E' sicuramente da affiancare ad altre guide più complesse e dedicate ma in sostanza si tratta dei miei appunti personali su molti problemi che ho avuto durante il mio passaggio da Windows a Linux, scritti in forma decente per essere proposti ad altri.

Se avete appena installato Ubuntu e non avete mai avuto Linux prima, vi consiglio di dare una lettura a questa guida. Vi è la soluzione a molti piccoli problemi in cui vi potreste trovare in futuro, e che vi farebbero perdere molto tempo.

Considerate che molte operazioni sui file potreste non trovarle in questa guida in quanto facilmente eseguibili in punta di mouse attraverso **Nautilus** (il file manager di Ubuntu) e gli strumenti di **GNOME** (l'interfaccia grafica di ubuntu).

In caso di errore di tipo "**permesso negato**" oppure "**are you root?**" fare precedere il comando che vi dà l'errore da **sudo** (cerca nella guida a cosa serve).

Questo perchè su linux per tutte le operazioni più pericolose per il vostro sistema, è richiesta la password di "root", cioè dell'amministratore del sistema.

Un'ultima cosa: se questa guida vi sembra più lunga delle altre che trovate in giro per la rete, è perchè **deve essere così**. Linux è un sistema operativo nuovo e **passare da Windows a Linux non è esattamente semplice come cambiare un paio di scarpe** nonostante questo sia quello che molti vogliono farvi credere.

### **Menu:**

- [Introduzione a Linux e ad Ubuntu](#)
- [Cartelle utilizzate da linux](#)
- [Gestire i processi in esecuzione esicurezza web](#)
- [Scompattare e installare pacchetti](#)
- [Disinstallazione di pacchetti](#)
- [Gestione files, cartelle e permessi](#)
- [Apache2 E Vsftpd \(server ftp\)](#)
- [Stampe](#)
- [Qemu, Vmware e Wine](#)
- [Montare-smontare partizioni](#)
- [Schede grafiche e Linux](#)
- [Audio, midi](#)
- [Boot del sistema e avvio automatico di applicazioni](#)
- [Cominciare a programmare C con GTK e JAVA](#)
- [Gli script di shell, i file .sh](#)
- [Passare da Windows a Linux](#)

## Introduzione a Linux e ad Ubuntu

### Guida: Passare da Windows a Linux (seconda parte)

**Linux** non è come Windows. Sono due sistemi operativi diversi con due filosofie completamente diverse.

**Ubuntu** è una distribuzione di Linux, come lo sono anche Debian, Red Hat, Suse, Slackware, Mandriva ed altre.

**Ubuntu** è una parola africana che vuol dire "**Umanità verso gli altri**", ma c'è chi la traduce anche "**Io sono perché noi siamo**", ma soprattutto, Ubuntu, è la distribuzione che uso io, **Ubuntu** è la "**mia distribuzione**".

**Linux** è un sistema sviluppato dalla collaborazione di migliaia e migliaia di persone in tutto il mondo che ogni giorno si incontra nei forum, comunica con newsletter, si aggiorna, si aggrega e porta avanti insieme lo sviluppo di questo sistema operativo e del software che vi gira.

E' gente che gratuitamente crea e contribuisce a mandare avanti quello che a tutti gli effetti è **il più grande progetto comunitario che è stato portato avanti nella storia dell'uomo**.

La gente che usa linux e vi scopre un difetto lo rende noto alla comunità e questo difetto viene corretto.

Se siete esperti informatici voi stessi potete modificare il difetto e ridistribuire il software corretto.

Linux è così da quando è nato.

I passi avanti compiuti dalla **comunità di Linux** sono tali che potreste anche in questo momento sostituire il vostro Windows per la maggior parte delle cose per cui adesso lo usate.

Tutto questo è gratis, va avanti gratuitamente ed è disponibile a tutti.

Se questo non vi interessa è perchè probabilmente siete così abituati ad usare software piratato su Windows che tutto quello che avete vi sembra libero, e vi sembra gratis.

No, non solo Windows non è gratis (chiedetelo a chi il software lo compra veramente..), ma detta legge sul mercato, impone delle regole e fa di tutto per fare in modo di continuare ad essere lei l'unica padrona del mercato.

Come biasimarla? La grande intuizione che permise a Bill Gates di creare il suo vasto impero fu che "i soldi si fanno con il software e non con l'hardware".

Qui però arriva l'**opensource**, cioè il codice aperto e libero a tutti, portato avanti da gente che non vuole denaro, ma soddisfazione personale, e che crede in uno spirito di comunità.

Diciamo le cose come stanno: in Windows c'è del software veramente molto bello, a cui noi ci siamo così tanto abituati che pensare di cambiare ci sembra una follia.

In linux c'è del buon software, a volte superiore a quello in Windows, di certo spessissimo più sicuro di quello presente in Windows...

Non fate l'errore di sottovalutare Linux e l'Opensource però. Il browser più usato al mondo, **Internet Explorer**, è **Microsoft** ed è in assoluto uno dei più pericolosi software per la vostra sicurezza in internet, e **in assoluto il browser che crea più problemi ad una persona che si impegna a creare un sito web**.

Per programmare il web c'è uno standard a cui tutti i browser hanno scelto di attenersi. Chi si avvantaggia di più da questa collaborazione è, oltre ai webmaster, anche i **disabili**, dato che molte regole sono pensate apposta per rendere loro la vita sul web molto meno dura.

Ovviamente il browser che si attiene meno di tutti a queste regole è proprio **Internet Explorer, il più comune browser al mondo.**

Progettare siti web su browser come **Firefox, Opera, Safari** è un gioco, un divertimento. Quando però si prova il proprio sito su **Internet Explorer** sono dolori e si è costretti a rifare tutto e ricercare il modo di far piacere il nostro sito a IE spesso uscendo fuori dagli standard della buona programmazione.

Questo mio piccolo sfogo consideratelo la mia piccola vendetta personale verso questo browser che tanti problemi mi ha creato in quanto webmaster.

**Firefox**, browser opensource, ha strappato a Internet Explorer una quota di mercato mondiale intorno al **20%**, una enormità se considerate che Internet Explorer la gente se lo trova già installato nel suo pc al momento in cui lo compra

Questo significa che **Internet Explorer anche se la gente ce l'ha (e l'ha pagato!), preferisce non usarlo.**

Fatelo anche voi, scaricate ed installate **gratuitamente** Firefox, non permettetegli ancora di fare del mondo ciò che vogliono.

Tornando a Linux.

Il **passaggio da Windows a Linux** può essere per qualcuno praticamente indolore, per altri un gran tribolare.

Dipende dalle vostre esigenze ma alla fine sarete soddisfatti del VOSTRO Linux, scoprirete quanto è un sistema malleabile, flessibile. **Lo adatterete a voi.**

I problemi in Linux sapete spesso da dove derivano? Dal fatto che le case che producono hardware spesso non dedicano grande cura ai driver per linux, rispetto a quanta ne dedichino invece a quelli per Windows.

Questo ci costringe spesso a creare dei giochetti per superare i problemi di configurazione del sistema..

Non si ci può far nulla... Tanto più però Linux continuerà a crescere ed ad essere scelto dalla gente, tanto più ne guadagnerà in considerazione da parte delle case produttrici di hardware ( e anche di software)..

Perché non fare in modo che il miracolo succeda?...

Un'ultima cosa:

Anziché lamentarci di Linux, dire che può essere più o meno problematico cominciare ad usarlo, configurarlo, dire altre sciocchezze, fermiamoci, e **cominciamo ad impegnarci ed a collaborare.**

---

## Le cartelle utilizzate da Linux

### Guida: Passare da Windows a Linux (Terza parte):

Nella tabella che segue sono indicate le cartelle più importanti di un sistema **linux**, con alcune delle sottocartelle e dei files più importanti.

Tenete presente che se avete un sistema linux diverso da **Ubuntu**, è probabile che qualche file possa trovarsi in una sottocartella diversa, ma in linea di massima, fra una distribuzione e l'altra la differenza non è mai troppa:

/	Directory generale del sistema, viene chiamata <b>root</b> (che è diverso da "utente root", cioè l'amministratore del sistema).
bin/	In questa cartella sono contenuti comandi e programmi, e anche "script di shell".
boot/	Kernel e altri file indispensabili per avviare Linux. Qualche sottocartella: <ul style="list-style-type: none"><li>• <b>/boot/grub/menu.lst</b> Questo è il file di configurazione di GRUB (è il boot loader)</li></ul>
dev/	Questa cartella contiene degli speciali file, a cui sono collegate le periferiche hardware del sistema. Qualche sottocartella: <ul style="list-style-type: none"><li>• <b>/dev/sndstat</b> Questo file di testo contiene informazioni sulla scheda audio. Vedi anche <b>/proc/asound/cards</b>.</li></ul>
etc/	File di configurazione dei programmi. Alcune sottocartelle e files: <ul style="list-style-type: none"><li>• <b>/etc/apt/sources.list</b> : contiene la lista delle sorgenti da cui attingere i pacchetti ("repository list").</li><li>• <b>/etc/apt/sources.list.d/</b> : In questa cartella sono contenute altre liste repository.</li><li>• <b>/etc/event.d/</b> : è la cartella usata da <b>upstart</b> (vedi la sezione relativa al <a href="#">boot del sistema e l'avvio in automatico di applicazioni</a>).</li><li>• <b>/etc/esound/esd.conf</b> : E' un file di testo contenente la configurazione audio di GNOME.</li><li>• <b>/etc/firefox/firefoxrc</b> : File di configurazione del browser firefox. Se avete problemi con l'audio di firefox, magari con i siti in flash, sostituite la riga <b>FIREFOX_DSP=""</b> con <b>FIREFOX_DSP="aoss"</b>, in modo da far usare a firefox i driver "alsa".</li><li>• <b>/etc/fonts/fonts.conf</b>: In questo file di testo vi è la configurazione dei "fonts", cioè caratteri che usate.</li></ul> <p>I "fonts" si trovano, sotto forma di files, dentro alcune cartelle che si trovano nel vostro Hard disk. Quali sono queste cartelle? Sono appunto indicate in questo file di configurazione "fonts.conf".</p> <p>Se intendete aggiungere un nuovo font, questo va posizionato in una di queste cartelle, poi bisogna lanciare da shell il comando <b>fc-cache</b> .</p> <ul style="list-style-type: none"><li>• <b>/etc/fstab</b> : In questo file di testo sono elencati i filesystem e i loro punti di mount, in modo da renderli disponibili già all'avvio del sistema.</li><li>• <b>/etc/init.d/</b> : In questa cartella sono ubicati tutti gli script per fare partire i programmi all'avvio del sistema.</li><li>• <b>/etc/login.defs</b> : In questo file di configurazione risiede la "configurazione generale del sistema di autenticazione". Si può ad esempio impostare un tempo massimo di validità per le</li></ul>

password degli utenti, in modo che le cambino spesso, oppure impostare la variabile d'ambiente "\$PATH", che permette di richiamare i comandi residenti in /bin/, o in /usr/X11R6/bin/ (i comandi per la shell grafica), senza necessità di specificarne la posizione.

- **/etc/modules** : Questo file contiene la lista dei moduli che devono essere caricati all'avvio del sistema (per esempio il modulo per la scheda grafica e audio, per la riproduzione dei file midi, modulo per la stampante ecc.). Ogni linea deve contenere il nome di un modulo; le linee vuote o che iniziano per # vengono ignorate.
- **/etc/profile** : in questo file e anche in /home/nome\_utente/.profile sono contenute le "variabili d'ambiente". In uno dei due file trovate ad esempio la variabile "PATH", che deve contenere il percorso di quei file che volete vengano trovati senza bisogno di specificarne il percorso, ad esempio i file eseguibili contenuti in /usr/bin/.

Altra variabile che potreste un giorno avere bisogno di aggiungere è "LD\_LIBRARY\_PATH", che deve contenere i percorsi in cui si trovano i file di libreria. Ci sono infatti dei programmi che collocano le loro librerie in percorsi inconsueti, ad esempio /usr/local/lib/ o altro.

La variabile "PS1" serve invece a controllare l'aspetto del prompt (cioè bash, shell, chiamatela come volete).

Potreste anche trovarvi il comando "**umask**", di solito settato a 022. Vi basterà sapere che questo comando (settato a 022) serve a fare in modo che quando un utente crea un file (ad esempio un file di testo), lui solo potrà modificarlo, mentre tutti gli altri utenti NON possono modificarlo (ovviamente root, il padrone assoluto potrà sempre fare quello che vuole, senza mai limitazioni).

- **/etc/rc0.d.....rc6.d/** : cartelle dei runlevels (vedi sezione [boot e avvio](#)).
- **/etc/rc.local** : E' un file di testo eseguito al termine di ogni runlevel. Potete inserire qui delle operazioni che voi eseguite spesso da Shell (o "terminale"), in modo che il sistema le esegua in automatico.
- **/etc/sudoers** : E' il file di configurazione del comando **sudo** è **/etc/sudoers** che è modificabile tramite l'utility **visudo**, che è un editor di testo (come può essere **gedit**, **vi**, **nano** o altri) ma con in più il check della sintassi del file (e quindi evita di salvare il file con degli errori).
- **/etc/shadow** : E' il file in cui vengono salvate, in forma criptata, le password degli utenti ed altre informazioni relative alle password, tipo ultima modifica, durata minima, durata massima, ed altri parametri.

Nei sistemi Unix in passato l'elenco degli utenti con le loro password si trovava nel file **"/etc/passwd"**. Oggi in quel file la password viene sostituita da una lettera "x" mentre le password sono contenute nel file **"/etc/shadow"**, leggibile solo da root ed in cui le password si trovano criptate in maniera robusta.

- **/etc/timidity/timidity.cfg** : File di configurazione di Timidity. In questo file potete impostare il file .sf2 (soundfont) che dovrà essere usato da timidity per riprodurre i file midi.
- **/etc/X11/gdm/gdm.conf** e **/etc/X11/gdm/gdm.conf-custom** : "gdm" (Gnome Display Manager) permette ad un utente di effettuare il login in un sistema GNU/Linux tramite una piacevole interfaccia grafica dove inserire username, password ed eventualmente scegliere il gestore di finestre da avviare o scegliere di riavviare o arrestare il sistema.

GDM viene distribuito solitamente fra i pacchetti di GNOME (uno dei Desktop Environment più famosi), ed è facilmente configurabile tramite l'utility **gdmsetup** o manualmente modificando i file **/etc/X11/gdm/gdm.conf** e **/etc/X11/gdm/gdm.conf-custom** (operazioni

	<p>che richiedono i privilegi di root).</p> <p>GDM viene fatto partire all'avvio del sistema, è quindi fra i programmi avviati nei runlevels.</p> <ul style="list-style-type: none"> <li>• <b>/etc/X11/default-display-manager</b>: se avete installato sia GNOME che KDE come display manager, è da questo file che scegliete quale dei due deve essere quello usato di default.</li> <li>• <b>/etc/X11/xorg.conf</b>: file di configurazione di xorg. Vedi la sezione "<a href="#">scheda grafica</a>" della guida.</li> </ul>
home/	<p>Contiene le cartelle assegnate all'utente/i (tutte tranne quella di root).</p> <p>Ogni utente è padrone assoluto nella <b>sua</b> cartella, in cui potrà creare o cancellare liberamente file o cartelle.</p> <p>Ogni cartella "nome_utente" è di solito ricca di file nascosti (quindi preceduti da un puntino ".", che li rende invisibili).</p> <p>Molti di questi file nascosti sono file di configurazione, modificabili dall'utente.</p> <ul style="list-style-type: none"> <li>• <b>/home/nome_utente/.local/share/applications/defaults.list</b>: questo file defaults.list svolge la stessa funzione di quello presente in /usr/share/applications/defaults.list ma con la importante differenza che questo è contenuto nella home di un utente e di conseguenza eventuali modifiche saranno effettive solo per questo utente. A dire il vero questo defaults.list presente nella home può mancarvi.</li> </ul> <p>Potrebbe allora essere utile crearlo da voi questo file (usando un qualsiasi editor di testo), nel caso che la comoda funzione "apri con" per assegnare l'esecuzione di un tipo di file ad un programma specifico vi dia problemi.</p> <p>Nella stessa cartella "applications" vi è anche il file mimeinfo.cache, che serve al sistema a tenere traccia di tutti i programmi che possono usare quel determinato tipo di file (invece defaults.list indica il programma da usare di default!).</p> <p>Cercate su google il modo di come impostare il file defaults.list, mimeinfo.cache e i file con estensione .desktop presenti nella cartella.</p> <ul style="list-style-type: none"> <li>• <b>/home/nome_utente/.profile</b>: vedi il file /etc/profile</li> </ul>
lost+found/	File persi causa errori del disco e recuperati dal sistema
mnt/	<p>Directory di mount delle partizioni dei dischi, dei CD-rom, del floppy, delle cartelle dischi Windows condivise.</p> <p>Oggi si preferisce usare la directory /media per fare queste cose.</p>
opt/	<p>nella directory /opt dovrebbe finire dentro tutto il software che viene installato non usando l'"installer" ufficiale.</p> <p>Nel caso Debian tutto quello che è installato non usando i .deb.</p> <p>Serve a tenere staccato da tutto il resto quello che non viene gestito direttamente dal pacchettizzatore ufficiale.</p>
proc/	<p>File virtuali di informazioni sul sistema:</p> <ul style="list-style-type: none"> <li>• <b>/proc/asound/cards</b> Questo file di testo contiene informazioni sulla scheda audio. Vedi</li> </ul>

	anche <b>/dev/sndstat</b> , il comando <b>alsamixer</b> e la sezione di questa guida " <a href="#">Audio, midi</a> "
root/	Directory privata dell'amministratore di sistema. Questa directory non si trova nella cartella /home come quelle degli altri utenti del sistema.
sbin/	Contiene comandi e programmi riservati al root.
tmp/	Directory per i files temporanei
usr/	<p>Contiene tutti i programmi installati. Praticamente il sistema risiede per lo più qui dentro. Alcune sottocartelle:</p> <ul style="list-style-type: none"> <li>• <b>/usr/bin/ e /usr/sbin/</b> : contengono file eseguibili usabili dagli utenti (bin) e dall'amministratore (bin e/sbin), che non sono strettamente legati al funzionamento base.</li> </ul> <p>Se installate un programma qualsiasi è molto facile che il suo eseguibile finirà proprio in /usr/bin/.</p> <p>In generale gli eseguibili contenuti in /usr/bin/ e in alcune altre cartelle (per vedere quali sono queste cartelle dove vengono cercati i binari scrivete su shell "<b>\$PATH</b>". Questo vi mostrerà una lista di cartelle, tra loro separate da ":". Se vi interessasse modificare questa lista, modificate il file "/etc/login.defs".) sono eseguibili senza bisogno che ne dobbiate specificare il percorso completo.</p> <p>Se non ricordate esattamente come si chiama un comando ma ne ricordate le prime lettere, scrivetele su shell e premete due volte il tasto <b>[tab]</b>, è un trucchetto che si usa spesso e fa risparmiare molto tempo.</p> <ul style="list-style-type: none"> <li>• <b>/usr/doc/ e /usr/man/</b> : Pagine e pagine di documentazione e manuale.</li> <li>• <b>/usr/games/</b> : i file binari (ovvero gli "eseguibili") di videogiochi e programmi educativi vanno spesso in questa cartella o in "/usr/local/games/", e non in "/usr/bin/".</li> <li>• <b>/usr/include/</b> : per gli sviluppatori. Contiene moltissimi file .h .</li> <li>• <b>/usr/local/</b> dovrebbero essere installati (da parte dell'amministratore) tutti quei programmi locali che non appartengono alla distribuzione.</li> <li>• <b>/usr/local/games/</b> : vedi "/usr/games/".</li> <li>• <b>/usr/man/</b> : vedi /usr/doc/.</li> <li>• <b>/usr/share/applications/defaults.list</b> : In base al tipo di file, in questo file di testo è indicato quale è l'applicazione che deve essere eseguita. Ad esempio qui sarà indicato quale è il lettore di default per leggere gli mp3, quale il lettore per i file midi ecc... ecc...</li> </ul> <p>Sempre nella cartella "applications" sono contenuti dei file con estensione .desktop che sono modificabili da comuni editor di testo (tipo gedit, nano o altro). Questi file .desktop permettono di associare per ciascun tipo di file una icona e un programma da eseguire (vedi anche /home/nome_utente/.local/share/applications/defaults.list).</p> <ul style="list-style-type: none"> <li>• <b>/usr/share/xsessions/</b> : In questa cartella stanno i file con estensione .desktop, che sono le "sessioni" con cui potete scegliere di avviare il vostro linux. Le sessioni vi servono quando volete che all'avvio di linux vengano avviate in automatico anche alcuni programmi.</li> </ul> <p>Per fare questo, create una "Sessione" con le applicazioni che vi interessano.</p> <p>Potete creare tante sessioni se volete, e potete scegliere da quale sessione avviare all'avvio del sistema, dallo stesso pannello che in cui vi si chiede nome utente e password (<b>GDM</b>).</p> <p>Ad esempio, si può creare una sessione <b>Xgl.desktop</b> in cui venga avviato il file</p>



	<p><b>/usr/bin/startxgl.sh</b> (che va creato) che fa partire "Xgl" (con anche GNOME).</p> <p>Altro esempio è <b>GNOME.desktop</b> che fa partire GNOME con il file <b>/usr/bin/gnome-session</b></p> <p>Ma cosa sono questi file con estensione .sh? C'è un'apposita <a href="#">sezione</a> in questa guida.</p> <ul style="list-style-type: none"> <li>• <b>/usr/X11R6/</b> contiene gli eseguibili (<b>/usr/X11R6/bin/</b>), le librerie (<b>/usr/X11R6/lib/</b>) e le pagine di manuale (<b>/usr/X11R6/man/</b>) relative all'ambiente grafico più comunemente noto come "X".</li> </ul>
var/	<p>Contiene i dati che vengono modificati quando il sistema lavora normalmente, i file di log, le directory per le mail di sistema, le code di stampa, i file temporanei e le cache dei programmi. Alcune sottocartelle:</p> <ul style="list-style-type: none"> <li>• <b>/var/cache/apt/archives/</b> : in particolare i sistemi derivati da Debian, come Ubuntu, scaricano in <b>/var/cache/apt/archives</b> tutti i pacchetti scaricati con apt-get (osynaptic - la sua gui). Per una installazione desktop, con pochi utenti, il peso di /var dipende quasi esclusivamente dalla cartella dei pacchetti (che periodicamente puo' essere ripulita in automatico).</li> <li>• <b>/var/lib/dpkg/info/</b> : In questa cartella sono contenuti i file <b>.list</b>, che contengono le liste dei file di ogni pacchetto installato. Digitate da shell "<b>less /var/lib/dpkg/info/nomepacchetto.list</b>" se conoscete il nome del pacchetto del quale vi serve ottenere la lista di tutti i file (in alternativa potete anche vederne la lista da Synaptic).</li> <li>• <b>/var/mail/</b> : contiene messaggi di posta elettronica scaricati da internet o inviati dalla rete locale e non ancora letti.</li> <li>• <b>/var/run/</b>: contiene il PID (numero di identificazione) dei servizi in esecuzione.</li> <li>• <b>/var/spool/</b> : contiene lo spool di stampa, delle email e simili.</li> <li>• <b>/var/www/</b> : se avete installato apache è in questa cartella che va posto il vostro sito.</li> </ul>

## Gestire i processi in esecuzione e sicurezza web

### Guida: Passare da Windows a Linux (Quarta parte)

kill -9 "PID"	in linux ogni processo è identificato in modo univoco da un numero chiamato process ID, o PID. Conoscendo il PID di un processo, con questo comando possiamo chiuderlo. Per ottenere il PID di un processo vedi <b>ps aux</b>
ps aux	Mostra tutti i processi in esecuzione, con il PID, il proprietario e altre informazioni. Preferisco usare "ps aux" rispetto a "top"
top	<p>In Linux lo strumento più usato per visualizzare lo stato dei processi è il comando 'ps'.</p> <p>'top', a differenza di 'ps', è dinamico (cioè mostra l'utilizzo delle risorse da parte dei processi a intervalli regolari) e offre funzionalità aggiuntive che possono essere di aiuto in caso di eventuali problemi.</p> <p>Lanciato da riga di comando, top produce un output a monitor diviso in una parte alta, che contiene informazioni generali sul sistema, e in una parte sottostante, che mostra i processi che usano più Cpu.</p> <p>Nella prima vengono visualizzati l'ora, il tempo di attività, il numero di utenti loggati e il carico medio di sistema ogni minuto, ogni 5 minuti e ogni 15 minuti (se questi valori sono molto alti c'è qualcosa che non funziona correttamente).</p> <p>Inoltre sono presenti informazioni sui processi, sul processore, sulla memoria e lo swap.</p> <p>Accertatevi che non ci siano processi zombie (che riducono le prestazioni della macchina) e che lo swap usato sia a 0 (zero) oppure molto basso (altrimenti abbiamo problemi di memoria).</p> <p>Nella seconda ci sono le informazioni relative ai processi, ad esempio: il numero del processo (pid), il nome dell'utente proprietario del processo (user), la priorità del processo (pr), il valore nice (ni), l'utilizzo del processore(%cpu), l'utilizzo della memoria (%mem), lo stato del processo (s), il comando utilizzato per avviare il processo (command).</p> <p>'Top' accetta una serie di comandi interattivi, lanciati premendo un determinato tasto della tastiera, ad esempio:</p> <ul style="list-style-type: none"><li>- k + numero processo, termina il processo in questione.</li><li>- r + numero processo, modifica il valore nice di un processo; cioè la sua priorità, da -20 (massima priorità) a 19 (minima priorità).</li></ul> <p>Quindi un valore nice negativo migliora le prestazioni di esecuzione di un processo rispetto ad uno positivo.</p> <ul style="list-style-type: none"><li>- f, potete aggiungere o togliere alcuni campi nella tabella processi.</li><li>- s, potete cambiare il tempo che intercorre tra un aggiornamento e l'altro dei valori visualizzati.</li></ul> <p>Uscite da 'top' premendo il tasto q.</p>
locate hpiod	"locate" serve per cercare un file. In questo caso "hpiod". Locate è un comando velocissimo perché sfrutta una "database dei file", che ovviamente deve essere

	aggiornato con il comando " <b>updatedb</b> ". Se volete cercare un file senza usare nessun database usate "find" al posto di "locate" (attenzione però. find è molto potente ma può essere anche molto lento).
fuser -vn tcp 2208	Se sappiamo di una porta occupata (nell'esempio è la 2208), con questo comando possiamo risalire all'applicazione che la usa.
netstat -atu	<p>Permette di monitorare l'attività delle porte contro eventuali aggressori. La colonna "state" dichiara lo stato della connessione.</p> <p>Questa può essere ESTABLISHED (stabilita) se si è connessi o LISTENING se si è in attesa di una connessione.</p> <p>L'opzione -t mostra i socket TCP, -u mostra gli UDP.</p>

---

## Scompattare ed installare pacchetti su Linux Ubuntu

### Guida: Passare da Windows a Linux (Quinta parte)

I file che possono essere eseguiti, qui su linux, non sono quelli con estensione **.exe** (come in windows) ma i **.bin**.

Quindi non aspettatevi di trovare file **.exe** qui su linux (a dire il vero, esiste un modo per fare partire software Windows, quindi **.exe**, qui su Linux, vedi la sezione di questa guida "[Qemu, vmware e wine](#)").

Quello che però realmente cambia da Windows a Linux è il modo in cui si installa il software (programmi, giochi, ecc..).

Qui su linux i programmi da installare possono essere di due tipi: "**pacchetti**", oppure "**sorgenti**".

I pacchetti hanno estensione **.deb** (per Debian e Ubuntu) o **.rpm** (Red Hat, Suse e altre distribuzioni).

Quando di un software da installare avete la possibilità di scegliere fra diversi tipi di pacchetti, (.deb, .rpm), se avete Ubuntu scegliete quelli con estensione **.deb**.

Se disponete solo del pacchetto **.rpm** (e non avete .deb), questo potete convertirlo in **.deb** con un programma che si chiama "**alien**".

Dal momento che la conversione **da .rpm a .deb** può non essere perfetta, è preferibile quando si può scegliere, il pacchetto **.deb** oppure, installare da "**sorgenti**".

La maniera più semplice di **installare pacchetti** In Ubuntu è installarli dalla funzione "**Aggiungi/Rimuovi**" (che in realtà è un pacchetto che si chiama "**gnome-app-install**") che si trova nel menu "Applicazioni" di GNOME.

Vi sono altri programmi per gestire i pacchetti, fra questi c'è "**Aptitude**" (che non si appoggia nè su GNOME nè su KDE), Synaptic (in GNOME) e Adept (in KDE).

**Synaptic** si trova nel menù di GNOME "Sistema --> Amministrazione", ed è l'interfaccia grafica per gestire i pacchetti più importante in Debian e Ubuntu.

**Adept** è usata da chi possiede, al posto di GNOME, l'interfaccia grafica "KDE" (come ad esempio chi usa la distribuzione "Kubuntu").

Synaptic può comunque essere installato in KDE e Adept può essere installato in GNOME infatti tutto il software di KDE e GNOME è interscambiabile.

Considerate **Synaptic** come il vostro punto di riferimento in fatto di installazione di software: una enorme lista di programmi e giochi che potete facilmente consultare e in cui rapidamente trovate il software che vi interessa, lo selezionate e lo installate.

Da dove proviene tutto questo software? Proviene da delle "liste" (abitatevi al termine "**Repository**") che si trovano in rete e che vengono tenute aggiornate per voi dalla comunità di Debian e Ubuntu (e non solo, potete voi stessi aggiungere altre **repository**..).

Queste liste si trovano nel file **/etc/apt/sources.list** e nella cartella **/etc/apt/sources.list.d/**, altre liste possono essere liberate se abilitate da "Sistema --> Amministrazione --> Sorgenti software" anche i sorgenti con

restrizione e proprietari (abilitateli tutti).

Una cosina da sapere sui repository: siate più o meno certi della provenienza delle liste dei repository da cui attingete pacchetti... Può essere pericoloso per la sicurezza del vostro sistema.

I repository teoricamente più sicuri sono quelli rilasciati da Ubuntu stessa. Ubuntu rilascia quattro tipi di repository, separati sulla base del tipo di supporto che è fornito e l'aderenza alla filosofia di Ubuntu. Questi sono:

- **Main** (software supportato ufficialmente)
- **Restricted** (software supportato ma non disponibile nei termini del software libero)
- **Universe** (mantenuto dalla comunità, non ufficialmente supportato)
- **Multiverse** (software non libero)

Il CD di installazione di Ubuntu contiene software preso dai componenti **Main e Restricted**.

Per aumentare la sicurezza dei repository, esiste un sistema di "**chiavi**" che vengono fornite dal manutentore dei repository, e che sono necessarie per recuperare il software da questi repository.

Non mi dilungo oltre sull'argomento chiavi visto che coloro che rilasciano i repository vi spiegano facilmente cosa scrivere per recuperare questa chiave, vi dico solo che è un'azione svolta dal comando "**gpg**".

La prima cosa da fare, quando c'è un programma o un gioco (un "pacchetto") che vi interessa è cercare in Synaptic se il pacchetto è presente.

Se il pacchetto che vi interessa è nella lista di Synaptic, installatelo da lì perchè facendo così è veramente probabile che funzioni senza storie.

Se non vi funziona, vi consiglio di provare a cercare su internet e nel forum di Ubuntu (o della vostra distribuzione) se qualcuno ha avuto il vostro stesso problema ed insieme lo si risolve.

Un'altra cosa da sapere è che esiste un software che si chiama **gdebi**, che vi permette di **scaricare ed installare** da internet i pacchetti .deb **con un semplice click**.

I pacchetti installati con **gdebi** ve li ritrovate già pronti ed installati ed, ovviamente, aggiunti alla lista di Synaptic. Comodo no?

Se non avete installato **gdebi**, cercatelo in Synaptic ed installatelo.

Abbiamo parlato dell'installazione da pacchetti, adesso consideriamo l'installazione da "**sorgenti**".

**Installare da sorgenti** significa in sostanza installare direttamente dal codice scritto dal programmatore.

I sorgenti sono normalmente compattati (o "compressi") in archivi denominati "**tarball**" ed hanno più comunemente estensione **.tar.gz**, **.tar**, "**tar.bz2**", ma possono anche essere **.rar**, **.zip** o anche altra estensione.

In Ubuntu premendo con il pulsante destro sulle icone di questi archivi, potete scompattare i file con un'apposita opzione ("**Estrai qui**", più facile di così? E' la maniera più rapida di estrarre un archivio).

Altrimenti gli archivi possono essere "scompattati" (o "decompressi") usando la "riga di comando" (detta anche "**shell**" o "**terminale**", o "**bash**") mediante il comando "**tar**".

Giocoforza, in linux, se si vuole fare qualcosa di un minimo serio, è necessario cominciarsi ad abituare che bisogna imparare a svolgere le comuni operazioni, non soltanto da **GNOME** o **KDE** (cioè attraverso le più comuni interfacce grafiche, che permettono di effettuare molte operazioni usando solo il mouse, come in Windows), ma anche da **Shell**.

Fra l'altro, considerate che vi potrebbe capitare un giorno di avere problemi con l'interfaccia grafica, che non si avvierà. In tal caso l'unico modo che avrete per venire fuori dall'inghippo sarà soltanto avviare il vostro sistema in "recovery mode" e sistemare il tutto con i soli strumenti offerti dalla riga di comando.

In una ipotesi del genere, o conoscerete i comuni comandi da riga di comando e ve ne uscite fuori, o vi ricorderete di avere stampato questa guida su carta, leggerete i comandi per muovervi su Shell e ne verrete fuori, o non ne verrete fuori affatto..

La **Shell** può essere avviata dal menù di GNOME facendo "**Applicazioni --> Accessori --> Terminale**", oppure con la combinazione di tasti **CTRL+ALT+F1** (fate **CTRL+ALT+F7** per ritornare al Desktop).

Vi consiglio di NON cancellare i sorgenti del software che installate: createvi una apposita cartella chiamata "sorgenti" nella vostra /home/nome\_utente/ e metteteli tutti lì dentro. I sorgenti infatti potrebbero tornarvi utili nel caso vogliate disinstallare quell'applicazione (vedi la sezione "[Disinstallare pacchetti su Ubuntu](#)").

Nella tabella che segue vi sono le più comuni operazioni da eseguire da shell.

Notate in particolar modo il comando **apt-get**, col quale potete installare, aggiornare e rimuovere (disinstallare) pacchetti **.deb**.

alien -d nomefile.rpm	per convertire pacchetti rpm in .deb. Se non avete <b>alien</b> , scaricatelo da <b>Synaptic</b> .
apt-get clean	E' molto più radicale di " <b>apt-get autoclean</b> ", poichè rimuove dalla cache di apt ogni file .deb, anche quelli relativi ai pacchetti correntemente installati. Generalmente non avete bisogno dei file .deb per i pacchetti installati, quindi potrebbe essere la cosa migliore se avete problemi di spazio.
apt-get autoremove	Pulizia pacchetti
apt-get install nomepacchetto	Installa il pacchetto. Puoi usare anche il gestore di pacchetti "Synaptic" che si trova nel menu per installare nuovi pacchetti
	Potrete avere bisogno di eseguire questo comando qualora, cercando di installare un pacchetto, vi venga restituito un errore di dipendenze irrisolvibili, nonostante questo pacchetto si trovi in Synaptic.
sudo apt-get -f install	Può succedere ad esempio, se voi aggiungete nella vostra source.list un repository nuovo che contenga delle versioni aggiornate di pacchetti che avete già in Synaptic. E' complicato spiegare come e quando questo può andarvi a dare problemi, ma vi basti sapere che con questo comando potrete risolvere i vostri problemi.

apt-get remove nomepacchetto	Rimuove il pacchetto (ma non i suoi file di configurazione). Si può usare anche Synaptic per la stessa funzione.
apt-get --purge remove nomepacchetto	Eseguendo apt-get remove verranno rimossi i pacchetti, ma non i loro file di configurazione, per eliminare anch'essi è necessario anteporre <b>--purge</b> a <b>remove</b> .  Tenete bene a mente questa istruzione perchè se avete dei problemi irrisolvibili a <b>reinstallare</b> un pacchetto, grazie a questo comando potreste risolvere tutto.
apt-get update	Aggiorna la lista dei pacchetti disponibili dai repositories.  Da eseguire in special modo ogni volta che avete cambiato le "liste dei repositoryes", tipo <b>/etc/apt/sources.list</b> . Dopo avere eseguito questo comando è probabile che nella lista di Synaptic troviate del software nuovo da scaricare, che prima non avevate.  E' consigliabile eseguire l'update periodicamente per essere sicuri che le proprie liste rimangano sempre aggiornate.
apt-get upgrade	Aggiorna tutti i pacchetti installati.  È anche possibile aggiornare un singolo pacchetto attraverso l'esecuzione del comando: apt-get upgrade nomepacchetto
apt-get install build-essential	Build-essential è un pacchetto che si trova nei repository. In questo pacchetto vi stanno i compilatori di base, <b>gcc</b> , <b>make</b> , <b>g++</b> e altro
apt-get install linux-headers-`uname -r`	Versione kernel in uso
dpkg --get-selections	Elenca tutti i pacchetti installati
dpkg --get-selections   grep gnome	Elenca tutti i pacchetti "gnome" installati.  <b>dpkg</b> è un gestore di pacchetti Debian. <b>APT</b> è considerato un gestore di pacchetti più avanzato rispetto a <b>dpkg</b>
dpkg-reconfigure nome_pacchetto	Problemi con un pacchetto, dovrebbe risolverlo. Tipo reinstallazione
tar -xvzf file.tar.gz	Scompattare tar.gz o tar.gz2. "x" è per estrarre, "v" sta per "verbose" cioè visualizza i file che vengono estratti, "z" è per filtrare l'archivio con gzip, "f" è perchè è un file archivio.
find . -name "*.tar" -exec tar xvf {} ;	Questo comando è utile se dovete decomprimere molti archivi tutti in una volta. Cerca (find) tutti files che finiscono per .tar (-name "*.tar") e per ognuno di questi ({} ) esegue il tar xvf (-exec)  La direttiva -exec deve terminare per ;

Se dovete scompattare un archivio **.rar** ed avete difficoltà, potrebbe essere perchè dovete installare l'apposito pacchetto "rar" da **Synaptic**. Controllate da Synaptic se l'avete installato, altrimenti installatelo.

Quando del pacchetto che vi interessa non avete il .deb ma avete i "sorgenti", vi è una tipica procedura in

linux per installarli, che è composta da tre comandi, da eseguire uno dopo l'altro:

```
./configure  
make  
sudo make install
```

in cui:

./configure	<p>Con <b>./configure</b> viene lanciato un script che si occupa di controllare se il sistema risponde ai requisiti richiesti come librerie e programmi (per esempio dei compilatori). Si occupa anche di creare il <b>Makefile</b>, un file contenente delle variabili settate da <b>./configure</b> (come la directory dove verrà installato il programma e il percorso delle librerie necessarie alla compilazione).</p> <p>E' in questa fase che avvengono i maggiori problemi di installazione da sorgente. Infatti una libreria mancante causa quasi sempre un'errore e quindi l'interruzione del <b>./configure</b>.</p> <p>Anzichè scrivere semplicemente <b>./configure</b>, potete anche scrivere:</p> <pre>./configure --prefix=/usr/local/</pre> <p>--prefix farà in modo che <b>./configure</b> crei un makefile tale che il programma verrà installato dentro la cartella da noi indicata, in questo caso <b>/usr/local/</b>.</p>
make	<p><b>Make</b> è il programma che si occupa di analizzare il file <b>Makefile</b> creato da <b>./configure</b> e di compilare il programma.</p> <p>Ma cosa ha di diverso il comando <b>make</b> dal comando <b>gcc</b>?</p> <p>Make è concepito per compilare interi programmi, e ogni buon <b>grosso</b> programmatore che si rispetti, è costituito da tanti, anche tantissimi file, magari di centinaia di mega, e compilare tutta questa roba può richiedere molto tempo.</p> <p>Make fa questo lavoro per voi: spulcia le relazioni dei file che compongono il programmatore nel "makefile" creato da <b>./configure</b> e compila per voi tutta quella gran mole, generando dei file ".o" ("oggetto"), da ogni file sorgente e da questi, alla fine, ricava il da voi tanto desiderato <b>file eseguibile</b>.</p> <p>Se anzichè "installare" state "reinstallando" sopra una precedente installazione del programmatore, magari perchè volete semplicemente <b>aggiornarlo</b>, può capitare che "make" vi pianti in asso con un errore che non riuscite a capire..</p> <p>Questo perchè make si ricontrolla ogni file sorgente e capisce, se dall'epoca ultima vostra compilazione, il file è cambiato oppure no e solo i file modificati saranno ricompilati, facendovi risparmiare moltissimo tempo. Ma allora perchè questo errore? Capita MOLTO di rado, tipo quando vi ricompilate il kernel o cercate di riaggiornare qualcosa con il metodo "svn".</p> <p>E' complicato da spiegare il perchè questo errore si verifichi in queste situazioni, ma vi basterà sapere che la situazione si risolve facilmente: prima di eseguire il comando "make", scrivete:</p> <pre>make clean</pre> <p>Con questo comando vengono eliminati i precedenti file ".o" e potete fare una</p>



	reinstallazione bella pulita del programmone..
sudo make install	<b>Make install</b> si occupa di andare ad installare (tramite il comando <b>make</b> ) il programma sul nostro sistema.

Quella appena descritta è la procedura "classica" per installare un file a partire dai sorgenti.

In generale un programma in Linux, installato a partire da pacchetto o da sorgenti, sostanzialmente si sparpaglia fra le diverse cartelle di sistema di Linux.

I file di configurazione di questo programma, ad esempio, finiranno nella cartella **"/etc/"**, che per convenzione è fatta apposta per contenere i file di configurazione del software che viene installato.

I file che non saranno soggetti ad ulteriori modifiche (quindi il programma vero e proprio) finiscono in genere in una qualche sottocartella di **"/usr/share/"** o **"/usr/local/"**, mentre i file che subiscono modifica vanno in genere sotto **"/var/"**.

Il file per lanciare il programma, ovvero "l'eseguibile", va in genere in **"/bin/"**, le librerie vanno in genere sotto **"/usr/lib/"** o (più raramente), sotto **"/usr/libexec/"**.

Come potete vedere il programma si sparpaglia, ma in modo "razionale". Segue una logica che è regolata da uno standard chiamato **"Filesystem Hierarchy Standard (FHS)"**.

Tuttavia a volte può capitare di avere qualche problema nel disinstallare il software, o perchè il programmatore del software non ha pensato a un sistema per la disinstallazione, o perchè comunque sia non riusciamo a rintracciare tutti i file installati.

Proprio per questo motivo è consigliata in Ubuntu una procedura di installazione da sorgenti che è diversa dalla solita triade di comandi **"./configure && make && make install"**, e questa procedura è:

```
sudo auto-apt run ./configure
sudo make
sudo checkinstall
```

auto-apt run	<p>"auto-apt run" serve ad eseguire un comando ma sotto il controllo di auto-apt.</p> <p><b>Auto-apt</b> serve a fare in modo che se il programma che state installando manca di qualche file (in gergo si dice "ha problemi di dipendenze"), lui prova a cercare nel database dei pacchetti se esista un pacchetto che abbia il file che gli serve e ci chiederà se può installarlo. Il database può essere aggiornato con <b>" auto-apt update "</b>.</p> <p>Attenzione però di <b>non fare troppo affidamento</b> alle capacità di ricerca di auto-apt per la risoluzione delle dipendenze in quanto in realtà piuttosto spesso lui fallisce nella sua ricerca e dobbiamo essere noi stessi a cercare il pacchetto che serve mediante Synaptic</p>
--------------	---

	(cioè dobbiamo essere noi stessi, manualmente, a "risolvere le dipendenze"). Nonostante non sia infallibile vale sempre la pena di provare ad installare con <b>auto-apt</b> perchè male che vada può solo farci risparmiare del tempo.  <b>auto-apt</b> potrebbe non essere installato. In tal caso aprite Synaptic ed installatelo.
checkinstall	<b>checkinstall</b> tiene traccia di tutti i file installati da <b>make install</b> o equivalenti, crea pacchetti Slackware, rpm o deb con tutti i file, e li aggiunge nel database dei pacchetti installati, permettendo di essere facilmente rimossi con il gestore dei pacchetti della distribuzione.

Dopo avere digitato "**sudo auto-apt run ./configure**" vedrete che il sistema opererà un controllo sui file che possiede. Se li possiede tutti, alla fine verrà generato il **makefile** che verrà usato dal successivo comando che digiterete, "**sudo make**" (in realtà make potrebbe essere eseguito anche senza "sudo", a meno che i nostri sorgenti non si trovino in una directory protetta da scrittura. Io lo uso lo stesso).

Se i file non sono tutti presenti non avrete il makefile per continuare l'installazione. In tal caso riguardatevi tutta la lista di operazioni eseguite su shell dopo il comando "**sudo auto-apt run ./configure**" e cercate cosa è che non trova. A questo punto scaricate voi stessi la roba mancante servendovi di Synaptic. Tutto questo può sembrare noioso ma è il prezzo che si deve pagare, spesso, nell'installazione da sorgenti.

A volte può essere difficile capire quale sia esattamente il file che manca. Ad esempio l'esecuzione del nostro comando "**auto-apt run ./configure**" potrebbe terminare con un errore del tipo "SDL could not be found". In questo caso lui vi sta dicendo in realtà che ciò che non sta trovando sono le "librerie di sviluppo SDL". Voi cercherete in Synaptic questo fantomatico pacchetto "SDL" che lui non trova e non lo troverete nemmeno voi, perchè in realtà quello che cerca è un pacchetto di nome "**libsdl1.2-dev**".

Si perchè in generale normalmente i file che mancano a **./configure** in una installazione sono pacchetti di tipo "librerie per sviluppatore", che sono pacchetti che normalmente cominciano con la parola "**lib**" e finiscono con la parola "**dev**".

Quindi non appena vi si presenta un problema di dipendenze, partite sempre dal presupposto che molto probabilmente (spesso ma non sempre) il pacchetto che voi state cercando in realtà comincia con "**lib**" e termina con "**dev**".

Se nonostante tutto avete ancora dei problemi a capire quale pacchetto manca ed è richiesto per completare una installazione, servitevi delle funzioni di ricerca di questo sito: <http://packages.ubuntu.com/>.

Utilizzando la procedura consigliata (cioè quella, che fa uso di **auto-apt run, make, checkinstall**), i sorgenti vengono trasformati in pacchetto in futuro facilmente eliminabile da **Synaptic** oppure da **Shell** digitando:

```
sudo dpkg -r nomepacchetto
```

## Disinstallare pacchetti su Ubuntu

### Guida: Passare da Windows a Linux (Sesta parte)

Se i pacchetti da voi installati erano dei **.deb**, o dei **sorgenti** installati usando il comando "**checkinstall**", dovrebbero essere facilmente rimovibili usando **Synaptic**.

Se avevate installato compilando dei sorgenti con la forma:

```
./configure  
make  
sudo make install
```

Cercare dove sono piazzati i sorgenti e provate a disinstallare usando "**make uninstall**".

Se non li trovate, provate a riscaricare i sorgenti (occhio alla versione del pacchetto, che sia assolutamente identico a quello che volete disinstallare) da dove li avevate presi in origine e a dare "make uninstall".

Il "make uninstall" cancella i files da dove sono stati piazzati dopo l'installazione.

Potrebbe non funzionare, dipende se il programmatore ha in origine previsto il "**make uninstall**".

Se non va bisogna andare a cancellare manualmente i file, nel posto dove sono stati installati.

Sempre partendo dai sorgenti, fare "**makefile**" e cercare i file.

Come rimedio "preventivo" a tutto questo, abituatevi, durante la procedura di installazione da sorgenti, ad usare il tool "**checkinstall**" che monitora le fasi di installazione dei programmi da sorgente e ne permette una rimozione molto più semplice.

**Se checkinstall non ha funzionato** durante la procedura di installazione, e quindi avete dovuto installare necessariamente con "make install" allora abituatevi a conservare i sorgenti scompattati dentro una apposita cartella "sorgenti", creata da voi, nella vostra home. In questo modo saprete sempre dove andarli a cercare.

---

## Gestire files, cartelle, permessi su Linux

### Guida: Passare da Windows a Linux (Settima parte)

In tutti i sistemi \*nix (Unix, Linux) esiste l'utente "root", che è l'amministratore del sistema, colui che può fare qualsiasi attività sul pc, compreso cancellare files importanti e modificare partizioni.

Alla fine della vostra installazione di ubuntu avete così due utenti: quello che avete impostato voi durante l'installazione e questo utente "root".

Tradizionalmente in Linux per fare tutte le operazioni importanti si effettua il login come utente "root". Su ubuntu questo utente è stato di default disabilitato dal login e non ha password.

Sicchè, gli utenti con i superpoteri sono quelli che fanno parte del gruppo "admin".

Questi utenti "admin" (che può anche essere uno solo, non per forza tanti utenti), quando devono fare operazioni pericolose, devono fare precedere tali comandi da "sudo" (verrà richiesta la password).

Se fate un utente nuovo, dall'apposito pannellino in "Sistema --> Amministrazione --> Utenti e gruppi", potete dare anche a lui la possibilità di usare "sudo", barrando la casellina che gli assegna i privilegi di amministratore.

L'assegnazione ad un gruppo, come "admin" si può fare anche da riga di comando con:

```
adduser nome_utente admin
```

Perchè è importante che lo sappiate fare anche da riga di comando oltre che da pannello? Risposta: Perchè questo potrebbe salvarvi. Fidatevi.

Detto questo, altra questione importante su Linux, è che ogni file ha assegnati dei "permessi". Ogni file in Linux ha tre parametri di "permessi":

- Utente proprietario owner del file
- Gruppo del proprietario
- Altri utenti

Per ogni parametro (ricordiamolo, sono tre), va impostato uno di questi valori:

0 = Nessun permesso	-
1 = Esecuzione	(x)
2 = Scrittura	(w)
3 = Esecuzione + Scrittura	(xw)
4 = Lettura	(r)
5 = Lettura + Esecuzione	(rx , è la somma dei valori 1+4)
6 = Lettura + Scrittura	(rw, 2+4)

7 = Lettura + Scrittura + Esecuzione (rwx, 1+2+4)

Per cambiare il permesso ad un file (o ad una cartella) si usa il comando **chmod**, esempi:

chmod 777 -R cartella	In questo modo abbiamo impostato il permesso di lettura scrittura ed esecuzione ( 4+2+1 = 7 ) rispettivamente per proprietario, gruppo e utenti. La -R serve per applicare i permessi anche su tutte le sottocartelle e i files all'interno.
chmod 644 ciccio	lettura e scrittura ( 4+2 = 6 ) per il proprietario e solo lettura ( 4 ) per gruppo e utenti. Equivalente a chmod u+w file
chmod 400 file	Se vogliamo che possa accedere a un file in lettura solo il proprietario (passare quindi da -r--r--r-- a -r-----). Equivalente a chmod go-r file
chmod 755 cartella	Così assegni permessi di lettura scrittura esecuzione al proprietario, lettura esecuzione al gruppo, lettura esecuzione agli altri.

Per sapere i permessi dei file in una cartella, usando la shell piazzatevi dentro una cartella (usate il comando "cd" per muovervi fra le cartelle), e date il comando:

ls -l

Verranno elencati i file presenti nella cartella, ogni riga avrà un aspetto simile a questo:

drwxr-xr-x 5 root root 4096 2007-01-05 18:08 home

Come potete vedere, quella è la riga della cartella "home", la prima colonna è quella che ci interessa per sapere i permessi.

Il primo carattere (d) indica l'attributo, ("d" vuole dire directory), saltatelo.

I successivi nove caratteri sono i permessi, vanno letti a 3 a 3, ovvero: (rwx)(r-x)(r-x) quindi, in questo caso, è come dire "755" (proprietario, gruppo, altri).

La terza colonna e la quarta (root e root) indicano rispettivamente il "proprietario" e il "gruppo" del file.

Altri comandi utili ed importanti per file e cartelle:

cd	Il comando cd serve a spostarsi tra le directory. Ad esempio, facendo "cd /etc" ci si sposta nella directory "etc". "cd .." : serve invece a spostarsi nella cartella superiore a quella in cui vi trovate.
df -h	Visualizza lo spazio libero rimasto nelle partizioni. Oppure nel pannello "applicazioni-->accessori" di GNOME c'è un analizzatore dello spazio occupato.
du -s /home	Restituisce la dimensione totale complessiva di una cartella, in questo caso la

	cartella "home".
free	Memoria di sistema libera
sudo gedit /etc/fstab	Vi mostra il file in cui vengono indicati i filesystem utilizzati e le directory ad essi associate.
gksudo kdesu	Vedi "su" e "sudo". Se dovete avviare applicazioni grafiche con i privilegi di <b>root</b> (cioè dell'amministratore del sistema), se usate " <b>sudo</b> " potreste avere qualche problema. Usare invece <b>gksudo</b> per applicazioni Gnome e <b>kdesu</b> per quelle KDE in modo che carichi come root anche le librerie grafiche che servono a quei programmi. Utilizzo tipico: "gksudo nautilus". Facendo così si possono creare e modificare file in cartelle protette senza usare la shell.
ls -a	ls serve per elencare il contenuto di una directory. L'opzione -a serve a vedere anche i file nascosti
mv	Il comando mv serve per spostare, o rinominare, file e directory.
rm -r nomecartella	Rimuove la cartella con il suo contenuto. Il comando rm viene usato anche per cancellare files.
mkdir nomecartella	Crea una nuova cartella
su	passare ad utente di root. Se la password non va reimpostata con "sudo passwd root"
sudo	sudo (SUperuser DO) è un programma che consente di eseguire dei comandi con i privilegi di amministratore (sempre che vi sia abilitato). Facendo <b>su</b> si diventa a tutti gli effetti root. sudo invece assume i privilegi di root solo per un tempo limitato (15 minuti, di base).
which	Mostra il percorso completo di un file

Un'ultima cosa: un file o una cartella, per essere resa **invisibile**, bisogna semplicemente rinominarla con un puntino davanti. Ad esempio la cartella "prova", invisibile sarà ".prova".

In Windows, e quindi nei file systems FAT e NTFS, esiste un attributo "h" per rendere il file "invisibile". Qui su linux (file systems ext2, ext3 ecc..), non funziona l'attributo "h" dal momento che per rendere invisibile file o cartelle si usa il sistema "puntino".

Il sistema dei permessi impostati mediante **chmod** NON funziona nei file system NTFS (quindi Windows), dove tutti i file sono di norma leggibili ed eseguibili da tutti.

---

## Apache2 e vsftpd (server FTP)

### Guida: Passare da Windows a Linux (Ottava parte)

**Apache** è un server web, **Vsftpd** è invece un server ftp.

Servono a chi vuole pubblicare pagine web e files su internet.

**Apache** può anche essere utilissimo semplicemente per provare le vostre pagine web, prima di pubblicarle nel vostro spazio web su internet.

Apache, ma soprattutto **Vsftpd** possono essere usati per scambiare grossi file attraverso internet, da un pc ad un altro.

Apache2 ha, rispetto ad Apache alcune migliorie, e vi dovrebbe bastare sapere che ha anche un modo diverso di gestire la configurazione.

Per farsi un server completo apache, bisogna installare da Synaptic i seguenti pacchetti:

apache2-mpm-prefork: il webserver  
mysql-server: pacchetto riguardante mysql  
libapache2-mod-php5: modulo PHP per Apache  
php5-mysql: estensione MySQL per PHP  
php5-gd: per installare le librerie "gd"

sudo /etc/init.d/apache2 stop	Ferma apache
sudo /etc/init.d/apache2 start	Fa partire apache
sudo /etc/init.d/apache2 restart	Fa ripartire apache
sudo gedit /etc/apache2/apache2.conf	<p><b>apache2.conf</b> è il file di configurazione di apache2. Molte opzioni in questo file di testo sono "commentate" (precedute dal simbolo "#").</p> <p>Ricordatevi che qualsiasi modifica voi facciate al file di configurazione, dopo dovrete riavviare apache2 facendo:</p> <pre>sudo /etc/init.d/apache2 restart</pre> <p>Un esempio di modifica è aggiungere queste righe ad apache2.conf:</p> <pre>ExtendedStatus On &lt;Location /server-status&gt; SetHandler server-status Order deny,allow</pre>

	<pre>Deny from all Allow from localhost &lt;/Location&gt;</pre> <p>Aggiungendo le righe di cui sopra, se nella URL del vostro browser scrivete <a href="http://localhost/server-status">http://localhost/server-status</a> potete ottenere numerose informazioni su chi è connesso al vostro server in questo momento (facendo <a href="http://localhost/server-status?refresh=5">http://localhost/server-status?refresh=5</a> la pagina si riaggiorna ogni 5 secondi).</p> <p>La direttiva <b>ExtendedStatus</b> quando attivata con l'argomento On permette di ottenere un maggior numero di informazioni.</p>
<pre>sudo gedit /etc/php5/apache2/php.ini</pre>	<p>Nel caso abbiate installato php5 con apache2, questo è il file di configurazione di php5</p>
<pre>sudo gedit /etc/apache2/sites-available/default</pre>	<p>Se vi dovesse interessare abilitare il <b>mod_rewrite</b> sul vostro server <b>apache2</b>, prima di tutto scrivete su shell la riga:</p> <pre>sudo a2enmod rewrite</pre> <p>Quindi modificate con un editor di testo, tipo gedit, il file <b>/etc/apache2/sites-available/default</b>. Qui cercate le righe:</p> <pre>&lt;Directory /var/www/&gt; Options Indexes FollowSymLinks MultiViews AllowOverride none Order allow,deny allow from all</pre> <p>E sostituite <b>AllowOverride none</b> con <b>AllowOverride all</b>.</p> <p>Quindi riavviate apache con <b>sudo /etc/init.d/apache2 restart</b>.</p> <p>Dovrebbe essere così installato il supporto a <b>mod_rewrite</b> e alle regole del file <b>.htaccess</b>.</p>

Una volta installato **apache2**, potrete accedere al vostro server web scrivendo **http://localhost**.

Dall'esterno, la gente può accedere al vostro server web (e così al vostro sito!) impostando **http://Vostro\_ip\_pubblico**.

Ma come fare a sapere quale è il vostro indirizzo ip pubblico? Vi sono migliaia di siti in rete che servono apposta per dirvi quale è il vostro "**indirizzo ip**".

Uno di questi siti è <http://whatismyip.com>.

Tutti i computer connessi in rete hanno un indirizzo ip, anche il vostro, e potrebbe essere un indirizzo ip **fisso**,



o che cambia ogni volta che accendete il vostro pc (ip "**dinamico**").

Attenzione: voi stessi non potete nel vostro browser fare **http://Vostro\_ip\_pubblico** ed accedere nel vostro stesso pc.

Se volete fare una prova del genere, servitevi di siti tipo <http://anonymouse.org> per "fingervi" utenti "esterni" al vostro pc.

Non funziona? Dall'esterno non riescono ad accedere al vostro sito? Se fate <http://localhost> invece riuscite ad accedere nel vostro sito?

Questo succede perchè Apache si apre all'esterno attraverso la **porta 80**. Assicuratevi che questa porta sia aperta.

Per fare questo prima di tutto su shell eseguite il comando "**ifconfig**", che eseguito senza opzioni vi elenca tutte le interfacce di rete nel vostro pc.

Eseguendo **ifconfig** otterrete le impostazioni di "**eth0**" (o "eth1" o simile), "**lo**" e magari anche altro.

"**eth0**" (o "eth1" o simile) è la vostra scheda di rete.

"**lo**" è la "interfaccia logica" (detta anche "interfaccia di loopback"), che viene usata per le configurazioni locali (normalmente ha valore impostato a **127.0.0.1**).

Quello che ci interessa, è l'indirizzo indicato in **eth0** (o simile), che è qualcosa tipo **192.168.1.3** o molto simile.

Questo è l'indirizzo che dovreste usare nelle impostazioni del vostro router per "NATtare" la porta 80 (cercate nel manuale del vostro router come si creano le regole "NAT").

\* \* \*

Se dalla url richiamate una cartella del vostro server apache2, facendo tipo: "http://localhost/nome\_cartella" apache potrebbe mostrarvi a video una lista di tutti i file e le sottocartelle (funzione "**directory listing**"), questo può darvi dei problemi di sicurezza se vi interessa proteggere il contenuto di quei file e non volete che nessuno li scarichi.

Questo problema va risolto agendo su un file di configurazione di apache2, **etc/apache2/sites-enabled/000-default** (prima di modificarlo **fatevene sempre una copia di riserva**, e dopo ogni modifica assicuratevi che, se usate htaccess, il reindirizzamento funzioni ancora), oppure va risolto mediante .htaccess, oppure ancora, la via più rapida e indolore, piazzate un file index.html vuoto all'interno dentro ogni cartella che volete proteggere.

\* \* \*

Per quanto riguarda invece il server ftp "**vsftpd**" i comandi più importanti sono:

/etc/init.d/vsftpd stop	Ferma vsftpd
/etc/init.d/vsftpd start	Fa partire vsftpd
/etc/init.d/vsftpd restart	Fa ripartire vsftpd
gedit vsftpd.conf	Modificare il file di configurazione di vsftpd

Mentre per **apache** per comunicare con l'esterno dovete aprire la **porta 80**, per **vsftpd** è necessario aprire le **porte 20 e 21**.

---

## Stampe in Linux

### Guida: Passare da Windows a Linux (Nona parte)

Linux si serve di **CUPS** (Common Unix Printing System) per gestire le code di stampa ed inviare i dati alle stampanti (CUPS è un "server di stampa").

Di solito trovate CUPS già installato (pacchetti cupsys e cupsys-client).

Un'interfaccia per gestire CUPS è fornita da GNOME stesso sottoforma di un pacchetto chiamato **gnome-cups-manager** che di solito trovate già installato nel vostro sistema.

Tramite **gnome-cups-manager** potete modificare la qualità di stampa, numero di pagine per foglio ed altro.

Il vero pannello di impostazione di CUPS però è raggiungibile dal vostro browser (firefox, opera, konqueror o altro browser), inserendo l'indirizzo "**http://localhost:631/**" nella URL.

Non mi è mai capitato di avere problemi particolari con CUPS, eventualmente sappiate che il suo file di configurazione è **/etc/cups/cupsd.conf** ; non ho mai avuto bisogno di modificarlo manualmente.

Di solito gestisco le stampe usando il **gnome-cups-manager**, tuttavia vi confesso che non gestisce (almeno fino al momento in cui vi sto scrivendo) il modulo di stampa di giro automatico del foglio per stampa doppia faccia della mia stampante HP, funzione che invece è presente in **http://localhost:631** .

Vi consiglio comunque, per quanto possibile, di usare una sola delle due interfacce di gestione di CUPS perchè credo che fare regolazioni in entrambe le interfacce crei un pò di confusione nella configurazione della vostra stampante.

Personalmente uso sempre **gnome-cups-manager** e mi servo di **localhost:631** solo per abilitare l'opzione di giro automatico del foglio (quando mi serve) e non per altro.

**http://localhost:631/ --> E' il pannello di controllo di CUPS, eseguite questo link dal vostro browser (Firefox, Opera ecc.. ecc..)**

## QEMU, VMWARE, WINE

### Guida: Passare da Windows a Linux (Decima parte)

Con **qemu** e **vmware** potete creare dei "pc virtuali", in cui fare girare, se volete, un sistema operativo diverso da quello che usate normalmente.

In pratica create, all'interno del vostro pc, un pc del tutto "virtuale", con una sua scheda grafica, una sua scheda audio, una sua scheda di rete, un suo CD-ROM, un suo hard-disk ecc.. ecc (tutto rigorosamente "virtuale")..

In uno di questi pc virtuali potete fare girare quello che volete, Windows 2000, Windows XP e, perchè no, un'altra distribuzione di Linux.

Avviando uno di questi pc virtuali Windows, potete a tutti gli effetti installare del software Windows all'interno del vostro pc che di normale ha invece Linux.

Fra tutti i software di "virtualizzazione", Vmware sembra essere, almeno per ora, il più funzionante e performante.

Gli altri software di virtualizzazione, come **qemu**, vanno tenuti d'occhio nel tempo perchè migliorano sempre di più in fatto di stabilità, affidabilità e velocità.

Qemu è opensource, quindi liberamente scaricabile ed usabile, Vmware è invece un prodotto proprietario, e ve ne sono diverse versioni.

Vmware "**player**" è un software gratuito, la versione "**Server**" è pure gratuita (da poco tempo a questa parte), e la versione "**Workstation**" che invece è a pagamento.

Se vi interessa tutto questo per fare girare software Windows in Linux, tenete conto di due cose: tanto più il vostro pc è veloce, tanto meglio girano questi software di virtualizzazione, tanto più potete permettervi di usare software pesante.

Io devo dire che ho provato **Vmware Workstation**, in versione prova dimostrativa, ma non sono rimasto soddisfattissimo.. La velocità è piuttosto limitata (il mio pc non è velocissimo, è un Celeron 2.4 Ghz).

Non ho mai installato un'apposito tool di strumenti per Vmware che si chiama "**Vmware tools**" che servirebbe a migliorare molte cose, tra cui la qualità della grafica.

Tuttavia non immaginatevi mai chissà che cosa da Vmware.

Tutta un'altra musica è **Wine**, un software che permette di "adattare" il software per Windows qui su Linux.

Spesso la velocità di un software "emulato" da Wine è pari o superiore alla velocità dello stesso software che gira su Windows.

Il grosso problema di Wine è che, almeno fino al momento in cui vi sto scrivendo, è ancora un prodotto piuttosto lontano dall'essere perfetto.

I software per Windows, fatti girare su Wine: possono funzionare bene, possono funzionare così così, o possono non funzionare.

Possono arrivare ad installarsi ma non partire, o neppure arrivare a completare l'installazione.

L'installazione di qualcosa su Wine può essere semplicissima e non darvi alcun problema, oppure bisogna ricorrere a diversi trucchetti, a volte anche molto complessi, tra cui quello di installare il software prima nella vostra partizione Windows (**usando Windows**), e poi riavviare in Linux, leggere attraverso Linux la partizione Windows dove avete installato il programma per ricopiare nella cartella di Wine le cartelle del prodotto installato (e i suoi file di registro) qui su Linux.

Non vado oltre perchè ogni software ha spesso dei trucchetti tutti propri per partire, e vi sono dei siti sani che vi spiegano soltanto come fare partire anche uno solo di questi software su Wine.

I vostri siti di riferimento sono in primo luogo quello dello stesso Wine, in cui vi è la sezione "[App Db](#)" (in cui il software che gira meglio su Wine è raccolto in una categoria definita "**platinum**"), e il sito di "[Frank's Corner](#)", che sono pieni di consigli per voi.

Wine migliora sempre di più, sempre più software riesce a partire, e in futuro dovrebbe dare sempre meno noie. Attualmente Wine è giunto alla versione **0.9.43**.

Fra i programmi Windows che partono su **Wine** ricordiamo: Dreamweaver, flash, Office, Picasa, Google-earth, Photoshop CS, ma anche videogiochi come Call Of Duty ed Age of Empires.

Alcuni comandi utili:

<code>qemu -hda /home/nome_utente/vmware_img/windows.vmd k -m 192 -win2k-hack</code>	Avviare l'immagine di un hard disk
<code>qemu -boot d -cdrom /dev/hdd</code>	Avviare da un cdrom
<code>qemu -m 512M -boot c -cdrom /dev/hdb -hda nome-file.img -soundhw all</code>	Suono abilitato

Se disinstalli vmware player ricordati di cancellare la directory `/etc/vmware` e anche `/usr/lib/vmware`.

wine uninstaller	Per disinstallare programmi installati con Wine
winecfg	<p>Configurazione di Wine. Qui potete scegliere che versione windows usare (Win98? Win 2000? XP?).</p> <p>Fate attenzione alla sezione del pannello chiamata "<b>override</b>", vi è una lista selezionabile di dll.</p> <p>Non riuscite a fare partire il programma che vi interessa ma riuscite a capire quale dll potrebbe determinare l'incompatibilità?</p> <p>Cercate se quella dll è presente nella lista override. E' presente? Avete una copia di Windows originale installata in una partizione del vostro pc?</p> <p>Copiate la <b>dll originale</b> di Windows nella cartella di Wine corrispondente (cercatela nella vostra <code>/home/nome_utente/.wine</code>, dovreste trovarla) e segnalate da quel pannello che ne avete effettuato l'override.</p>

## Montare-smontare partizioni

### Guida: Passare da Windows a Linux (Decima parte)

Ogni sistema operativo ha il suo **"filesystem"**, ovvero il suo "modo di scrivere scrivere sull'hard disk" del pc.

Diciamo che il filesystem di Windows 9x (Windows 95,98 ecc..) si chiama **"FAT"**, quello di Windows XP si chiama **NTFS**.

In Linux attualmente il più diffuso filesystem è il **"ext2"** e si sta già cominciando ad usare il sistema **"ext3"**, che è usato ad esempio da Ubuntu Feisty.

Linux è capace di leggere e scrivere facilmente un filesystem FAT, per l'NTFS invece sono necessari dei "programmini" che permettano di leggere e scrivere queste architetture, come **NTFS-3G**, che ho sempre usato e non mi ha mai dato problemi.

Tenete presente che Linux è **"case sensitive"** (Windows no!), ovvero maiuscole e minuscole nei files sono considerati caratteri assolutamente distinti.

Quindi file.txt è diverso da File.txt e anche da FiLe.tXt.

Lo spazio di un **hard disk** può essere spezzato in tante parti (definite appunto **"partizioni"**), questo può essere molto utile per svariati motivi, tra cui quello di mettere più sistemi operativi in un solo hard disk, ad esempio mettere Windows e Linux in uno stesso hard disk (e scegliere all'avvio con quale sistema operativo avviare).

Per gestire le partizioni, Linux fornisce un utilissimo strumento chiamato **"gparted"**. Potete trovare gparted nella lista di Synaptic. Se non c'è in lista, è probabile che vi manchi qualche repository in **etc/apt/sources.list**.

Con gparted potete creare nuove partizioni, ridimensionare le esistenti, formattare partizioni secondo il filesystem che vi interessa (**ext3** nel caso di Feisty, ma anche ext2, o persino FAT o altro).

La gestione delle unità del sistema qui su Linux è diversa da Windows. Qui su Linux i dischi di sistema (che siano hard disk, lettori CD, DVD, o anche immagini iso), vengono **"montati"** su delle cartelle.

Digitate da shell il comando **"mount"**, vi uscirà un elenco delle periferiche ("devices") montate. Fra queste righe ve ne sarà una simile a questa:

```
/dev/hda1 on / type ext3
```

La particolarità di questa riga è quel segno "/" scritto tra le parole **"on"** e **"type"**, che è il segno che si usa in linux per indicare il percorso **"root"** ovvero la directory principale, ovvero quello che in Windows era spesso chiamato **"C:"**.

Dopo **"type"** vi è invece l'indicazione del tipo di filesystem (**"ext3"**).

Nella tabella che segue vi sono alcuni dei comandi più importanti ed usati, che hanno a che fare con il montare e smontare partizioni:

mkdir	Monta una partizione Windows ntfs <b>in sola lettura</b> nella directory /media/windows2, che avete creato con mkdir. La /dev/hdc1 può non essere quella giusta per voi,
-------	--

<pre> /media/windows2 mount /dev/hdc1 /media/windows2/ -t ntfs -o umask=0222 </pre>	<p>guardate quella giusta dal gestore periferiche di ubuntu.</p> <p>Se vi interessa che la partizione ntfs sia, oltre che in <b>lettura</b>, accessibile anche per la <b>scrittura</b> la procedura è un pò più complessa e serve NTFS-3G. Vi sono delle guide apposite in rete su come fare.</p>
<pre> mount /dev/hda1 /media/windows/ -t vfat -o umask=000 </pre>	<p>Per montare una partizione FAT (non ntfs!) Windows</p>
<pre> umount /media/windows/ </pre>	<p>Per smontare una periferica (in questo caso la partizione Windows, che ho precedentemente montato sulla cartella /media/windows).</p>
<pre> mount Iso_da_montare.iso /media/iso/ -t iso9660 -o loop </pre>	<p><b>Per montare una iso</b> in una cartella da voi appositamente creata (in questo caso /media/iso/).</p> <p>Bisogna avere il modulo "loop" aggiunto nel kernel, facendo:</p> <pre>sudo modprobe loop</pre> <p>Al termine potete smontare l'ISO facendo</p> <pre>sudo umount /media/iso</pre>
<pre> sudo gedit /etc/fstab </pre>	<p>Modificare il file di testo /etc/fstab <b>per montare le partizioni all'avvio</b> di ubuntu. In questo modo evitate di dovere rimontare le partizioni ad ogni riavvio.</p> <p>Se la partizione è <b>ntfs</b> aggiungere manualmente la riga:</p> <pre>dev/hda1 /media/windows ntfs umask=0222 0 0</pre> <p>Se invece è <b>FAT</b> e non ntfs aggiungere questa riga:</p> <pre>/dev/hda1 /media/windows vfat umask=000 0 0</pre> <p>Se c'è un cd che usate spesso (ad esempio un vocabolario di lingua straniera su cd), per evitare ogni volta di dovere usare il lettore cd (che è lento) potreste copiarvi il cd sul vostro HD creandovene una immagine <b>iso</b> e montarla in fstab in modo che ad ogni avvio del vostro pc questa venga montata in automatico.</p> <p>Per creare la iso cliccate col pulsante destro sulla icona del vostro cd e fate "<b>copia</b></p>

**disco"** e quindi **"copia disco"** su **"file immagine"**.

Fatta l'immagine, create una cartella su cui montarla(ad esempio **/media/iso/**) e aggiungete ad **etc/fstab**:

```
/percorso/immagine.iso /media/iso iso9660 ro,loop,auto 0 0
```

```
sudo fdisk -l | grep  
NTFS
```

Serve per vedere quali sono le vostre partizioni NTFS.



## Schede grafiche e Linux

### Guida: Passare da Windows a Linux (Undicesima parte)

"**xorg X11**" (o "**server X**") è il server grafico per il desktop di linux, derivato (open source) da Xfree86 (che invece non ha licenza GPL).

E' usato da GNOME, KDE e Xfce.

Il file di configurazione di xorg è **/etc/X11/xorg.conf**. E' possibile qui configurare video, tastiera, mouse ed altro.

Settando il "protocol" del mouse su "IMPS/2", il "device" su **"/dev/psaux"** ho risolto un problema del mio mouse PS/2 che sembrava inspiegabilmente ogni tanto impazzire sullo schermo.

Se il mouse non funziona, per prima cosa bisogna accertarsi che venga rilevato correttamente dal kernel.

I mouse di **tipo PS/2** sono visti come il device **/dev/psaux**. Altri mouse (per esempio gli **USB**) sono visti come **/dev/input (o /dev/input/mice)**.

In ogni caso è possibile controllare se un device rappresenta il mouse scrivendo in shell **"cat /dev/psaux"** (se è PS/2 per esempio) controllando l'output del device e muovendo il mouse. Per terminare la sessione premere **Ctrl-C**.

E' molto facile che la vostra scheda video, Nvidia o ATI abbia il 3D disabilitato in automatico qui su Ubuntu perchè il produttore non ne ha rilasciato dei driver open source.

In questi casi cercate nel sito di Ubuntu le istruzioni specifiche su come fare per abilitarla.

Altre cosucce da sapere, in pillole, sulle schede grafiche in generale:

- **XGL** : acronimo di X over OpenGL. In sostanza è un'architettura creata per sfruttare l'accelerazione hardware 3d delle schede grafiche.
- **AIGLX** : Accelerated Indirect GLX ("AIGLX"). e' simile a XGL ma è opensource. Almeno fino al momento in cui vi sto scrivendo, le cose stanno così:  
**AiGLX funziona solo con gli ultimi driver beta di Nvidia**, per le **ATI funziona solo con i driver open** inclusi in Xorg ("mesa"), per cui per ora niente supporto da parte dei driver proprietari ATI nei riguardi di AiGLX.
- **Compiz** : E' un desktop 3d. Serve a creare degli effetti in 3d molto belli nel vostro schermo, che si abilitano dall'apposita opzione presente nel menù Preferenze -->Effetti Desktop
- **Beryl** : Era un progetto analogo e derivato da Compiz. Non più supportato
- **fglrx** : è il driver proprietario di ATI, per le sue schede video. Mi pare di capire che XGL funziona bene con fglrx, lo stesso non si può dire, almeno per ora, di AIGLX.
- **mesa** : è un driver driver grafico (come fglrx), che è opensource. Questo driver è molto utile per fornire l'accelerazione 3d alle schede di cui non si hanno driver proprietari e/o che siano vecchie (e quindi magari non più supportate dai proprietari).
- **Driconf** : è un tool grafico, per la configurazione del DirectRendering Infrastructure (DRI). Consente di personalizzare i settaggi di prestazione e qualità video del driver opengl. Esso consente di personalizzare i settaggi di prestazione e qualità video del driver opengl .

I comandi che seguono, nella tabella, sono molto importanti:

<code>/etc/init.d/gdm</code>	Serve a riavviare GNOME senza riavviare il computer.
------------------------------	--

restart	<p>GNOME è l'interfaccia grafica in Ubuntu. Se, per un motivo o per un altro sembra essere impazzita, non c'è bisogno di riavviare completamente il computer, riavviate solo GNOME con questa istruzione.</p> <p>Potete anche utilizzare la scorciatoia <b>Ctrl-Alt-Backspace</b>, ma questa scorciatoia potrebbe essere disabilitata.</p> <p>Per abilitarla, bisogna aggiungere ad xorg.conf le righe:</p> <pre> Section "ServerFlags" Option "DontZap" "yes" EndSection </pre>
fglrxinfo	Restituisce informazioni sul driver della scheda grafica ATI, se è installata correttamente
glxinfo   grep rendering	Se restituisce direct rendering: Yes vuol dire che l'accelerazione 3D è funzionante.
fgl_glxgears	Test per vedere se l'accelerazione 3D della scheda grafica è attiva.
fireglcontrol	Pannello di controllo per scheda grafica ATI.
dpkg-reconfigure xserver-xorg	E' il pannello di configurazione di <b>Xorg</b> . Se modificate questo file, attenzione ai valori <b>HorizSync</b> e <b>VertRefresh</b> nella sezione " <b>Monitor</b> ". Non modificate questi due valori a meno che non abbiate dato un'occhiata al manuale con le specifiche tecniche del vostro monitor, in quanto <b>rischiate di danneggiarlo</b> .

## Audio, midi su Linux

### Guida: Passare da Windows a Linux (Dodicesima parte)

Driver comunemente usati da linux per comunicare con la vostra scheda audio sono i driver "**ALSA**" (Advanced Linux Sound Architecture).

ALSA è un componente del kernel volto a sostituire le **OSS** (Open Sound System), che sono meno avanzate. Le ALSA possono comunque, se ve ne è esigenza, emulare le OSS mediante il programma "**aoss**" che fa parte del pacchetto "**alsa-oss**".

Il controllo dei volumi potete visualizzarlo con il comando **alsamixer**.

Se non riuscite a sentire i file **midi**, molto probabilmente questo succede perchè non avete il modulo installato.

Fate riferimento al sito della vostra distribuzione (in questo caso Ubuntu) per risolvere il problema, che potrebbe eventualmente spiegarvi come installare un software che si chiama **Timidity** che ha anche una interfaccia grafica molto carina, visualizzabile con un apposito parametro ("**timidity -ig file.mid**").

Timidity può fare uso di un file **.sf2** ("**soundfont**", è in pratica un archivio con i suoni degli strumenti) che potete salvare in qualsiasi posto del vostro HD, io l'ho messo nella mia **home**.

Per fare questo, bisogna configurare Timidity per riprodurre Soundfont esterni.

Queste sono le ultime righe del mio file di configurazione di Timidity, adattate per fare funzionare, a titolo di esempio, il soundfont Unison.sf2:

```
# source /etc/timidity/freepats.cfg
dir /home/attorianzo
soundfont unison.sf2 order=0
```

Se l'audio in riproduzione del midi è gracchiante o si sente male, provate a cambiare soundfont.

\* \* \*

Se non riuscite a far funzionare Timidity ma avete una **scheda audio Soundblaster** potete tuttavia provare questa strada:

Anzitutto vi serve un file **.sf2 (soundfont)**, se avete una partizione con windows, potete prelevarne uno da lì, ad esempio "windows/system32/drivers/2gmgsmt.sf2".

Prendete quel file, copiatelo dove vi fa comodo, e fate da riga di comando:

```
sfxload 2gmgsmt.sf2
```

Per quanto riguarda i file midi poi, nel pacchetto "**alsa-utils**" vi sono due utility: **aplaymidi** e **arecordmidi**. Sfruttano i driver ALSA per riprodurre e registrare file midi.

Con il comando "**aplaymidi -l**" ci viene mostrata la lista di porte di uscita MIDI presenti nel sistema (per

dubbi, fai "**man aplaymidi**" sulla riga di comando).

**"arecordmidi -l"** analogamente ci mostra le porte MIDI di ingresso.

Una volta letta la porta attraverso "**aplaymidi -l**", potete ascoltare un file midi (nel mio caso la porta che mi interessava era la 25:0, quella di "**Emu10k1 WaveTable**") facendo:

```
aplaymidi -p 25:0 'filemidi.mid'
```

Nel mio caso la porta midi era la 25:0, ma può essere diversa ( **-p serve a specificare la porta**).

---

## Boot del sistema e avvio automatico di applicazioni

### Guida: Passare da Windows a Linux (tredicesima parte)

Cosa succede quando accendete il vostro pc?

L'avvio del pc viene detto "**boot**" o "**bootstrap**".

La prima cosa che fa il "**BIOS**" del pc all'avvio è di cercare le periferiche avviabili leggendo i "settori di avvio" (o "**MBR**") di floppy e hard disk.

L'**MBR** (Master Boot Record) è un'area molto piccola del vostro hard disk, in cui linux posiziona un programmino detto "boot loader".

Il "**boot loader**" si occupa di caricare il "kernel", ovvero il cuore del vostro sistema operativo.

I due principali "boot loader" di linux sono "**LILO**" e "**GRUB**" (GNU GRand Unified Bootloader). Ubuntu Feisty usa GRUB ed è proprio di GRUB che vi parlerò adesso.

L'MBR contiene lo "**stage 1**" di GRUB, che passa immediatamente la palla allo "stage 2" che invece è collocato nel disco vero e proprio (in una cartella tipo **/boot/grub/**).

Lo **stage 2** contiene il menù di avvio con la lista (**/boot/grub/menu.lst**) dei kernel avviabili.

Considerate che Linux aggiorna molto spesso i suoi kernel, o perchè ne correggono difetti, o perchè vi aggiungono nuovi driver, o perchè vi introducono nuove funzionalità; quindi una volta al mese è probabile che, fra gli aggiornamenti che scaricherete in automatico del vostro linux, vi ritroverete kernel nuovi, pronti per essere usati da voi.

Ogni volta che voi installate un kernel Linux nuovo, questo verrà posizionato nella directory "**/boot/**" ed automaticamente verrà aggiornata anche la lista dei kernel disponibili per GRUB (file "**/boot/grub/menu.lst**").

Nella lista dei kernel da avviare, potete persino inserire Windows. In questo modo potrete sempre scegliere se avviare da Win o da Linux, liberamente.

Mi sembra doveroso aggiungere due cosette ancora su GRUB:

quando voi installate Ubuntu installate anche GRUB.

..Ma se installate Windows (o effettuate un ripartizionamento che spinge Windows a ricontrollare la MBR) DOPO avere installato GRUB (e Linux), Windows modificherà e renderà inutilizzabile GRUB.

Il problema però si risolve facilmente riavviando il sistema dal vostro LiveCD (il cd usato per installare Ubuntu) e facendo una apposita sequenza di comandi di [ripristino di GRUB](#).

State tranquilli comunque perchè GRUB è affidabilissimo. L'**unica** volta che mi ha dato questo problema è stato (così come vi ho appena detto..) proprio quando ho cercato di ridimensionare la partizione di Windows per dare più spazio a Linux (operazione svolta peraltro egregiamente dal programma "**gparted**"). All'avvio di Windows (per provare se funzionava tutto a posto) lui mi sconvolse GRUB ma ripristinai tutto in pochi minuti con l'apposita procedura. La colpa purtroppo non è nè di Linux nè di GRUB, è Windows che è geloso e vuole avere per se tutti i miei dischi di sistema..

Una volta scelto da GRUB il kernel da caricare, il suo lavoro finisce qui, e da qui in poi il lavoro di avvio è svolto dal kernel, che cercherà di accedere a tutti i componenti dell'hardware, hard disk, schede di rete, scheda audio e tutto il resto.

In pratica il kernel cerca tutte le periferiche che sono compatibili con i suoi "drivers", che sono le "interfacce attraverso cui vengono pilotate le periferiche" (solo le periferiche compatibili col kernel possono funzionare).

Infine il kernel caricherà le applicazioni presenti nel "**runlevel**" tra cui vi è anche "**GDM**" (il "gestore di login" o "Gnome Display Manager") che è quella pagina in cui si può effettuare il login dell'utente (vedi oltre in questa sezione).

Da **GDM** mettendo il nome utente e password si può finalmente accedere a GNOME o altra "sessione" (vedi oltre in questa sezione).

Completato l'avvio, ecco alcuni comandi utili che potremmo eseguire da shell per avere qualche informazione riguardo alle operazioni di avvio del kernel:

dmesg   less	Il comando dmesg ripropone i messaggi visualizzati dal kernel durante l'avvio del sistema, consentendo di controllarli con tutta calma. Il comando less invece viene utilizzato, in generale quando si vuole velocemente prendere visione di un lungo file di testo.
who -r	Mostra quale è l'attuale runlevel in esecuzione.
runlevel	Come who -r, ci mostra il runlevel in esecuzione.

Se il vostro avvio è diventato lento, potrebbe essere utile per voi cercare di capire dove è che il vostro sistema perde tempo in avvio. C'è un modo per farvi mostrare tutte le operazioni che il vostro sistema compie durante l'avvio.

Fate **gedit /boot/grub/menu.lst** per modificare GRUB.

Individuate la riga in cui è specificato il kernel, dovrebbe essere simile a questa:

```
kernel /boot/vmlinuz-2.6.17-10-generic root=UUID=5ee00750-e638-423f-9d5d-44fdd7 42c7ba
ro quiet splash
```

Commentatela e riscrivetela sotto senza "**quiet**" e "**splash**" facendola quindi diventare:

```
#kernel /boot/vmlinuz-2.6.17-10-generic root=UUID=5ee00750-e638-423f-9d5d-44fdd7 42c7ba
ro quiet splash
kernel /boot/vmlinuz-2.6.17-10-generic root=UUID=5ee00750-e638-423f-9d5d-44fdd7 42c7ba
ro
```

In questo modo all'avvio di linux potete osservare e studiare in quale operazione il sistema perde più tempo all'avvio.

Se volete visualizzare tali operazioni ma nel contempo non volete rinunciare alla barra di scorrimento tipica dell'avvio, togliete quiet ma lasciate splash.

Come avrete visto, in quel codice scritto sopra, vi uno strano "**UUID**" seguito da codici strani. L'UUID è una sigla che serve ad identificare univocamente una periferica, nel nostro caso identifica la partizione che contiene il kernel considerato.

Si sarebbe potuto mettere, al posto di quel codice, anche `"/dev/hda1"` o `"/dev/hdc1"` o altro ma francamente a me ha dato meno problemi GRUB indicando direttamente l'UUID.

L'UUID della partizione che vi interessa (`"/dev/hda1"` o `"/dev/hdc1"` o altro) potete vederlo dal pannello in Gnome "**Gestione periferiche**", cercando nella partizione che vi interessa (o da shell, facendo "**sudo vol\_id -u /dev/XXX**". Se non avete il comando **vol\_id** installate da Synaptic il pacchetto "**volumeid**").

L'unica accortezza da tenere è che se modificate le partizioni (le ridimensionate con "gparted" o simili) è probabile che l'UUID cambi e quindi anche il file `/boot/grub/menu.lst` andrà risistemato manualmente.

\* \* \*

Potete scegliere voi stessi di mettere delle applicazioni che voi usate spesso in avvio "automatico" quando accendete il pc.

In generale, non conviene avviare in automatico applicazioni che non vi servono sempre, non tanto perchè possono rallentare inutilmente il sistema, anche di poco, ma quanto per i problemi di **sicurezza** che potrebbero portarvi.

Per controllare, disabilitare o abilitare applicazioni all'avvio, scaricate da synaptic e usate il pacchetto **bum** (boot up manager).

Se vi interessa mettere le mani sul boot di linux, in generale, è bene che sappiate cosa sono i **runlevels**.

Un runlevel è uno stato del sistema in cui possono venire compiute solo determinate operazioni. I runlevel sono sette, enumerati da 0 a 6, sebbene esistano anche il 7, l'8 ed il 9 che però i normali Unix (e Linux con loro) non utilizzano.

Runlevel 0: **/etc/rc0.d** Questo runlevel avvia la sequenza di arresto del sistema (shutdown).

Runlevel 1: **/etc/rc1.d** Questo runlevel rappresenta la modalità singolo utente, nessun altro utente può collegarsi, il servizio di rete è disabilitato.

Runlevel 2: **/etc/rc2.d** Per l'utente, Il 2 è il runlevel di default qui su Feisty.

Runlevel 3: **/etc/rc3.d** Per l'utente.

Runlevel 4: **/etc/rc4.d** Per l'utente.

Runlevel 5: **/etc/rc5.d** Per l'utente.

Runlevel 6: **/etc/rc6.d** Il runlevel 6 è quello di reboot.

Lo 0, l'1 ed il 6 sono riservati e corrispondono rispettivamente a: arresto del sistema (0), single-user mode (1) e reboot (6).

I runlevels dal 2 al 5 qui su ubuntu sono uguali, e potete modificarli se ne avete l'esigenza.

In Feisty c'è un sistema tutto nuovo per l'avvio del sistema, che non si basa più, come di solito in linux, sul file **/etc/inittab** (sistema "sysvinit"), ma su un sistema noto come **upstart**, che gestisce i demoni secondo le configurazioni impostate nella cartella **/etc/event.d/**.

Il runlevel di avvio usato di default in Feisty è il 2. Per cambiarlo bisogna editare manualmente il file **/etc/event.d/rc-default**.

In questo file `/etc/event.d/rc-default` infatti vi è il comando "telinit" che è serve a impostare il runlevel.

Gli script che di fatto permettono di gestire l'avvio o lo stop di un servizio sono in realtà nella directory `/etc/init.d/` (in questa cartella sono conservati i file binari, eseguibili, dei demoni installati) e possono essere utilizzati direttamente dall'utente per gestire i singoli processi.

I file presenti nelle cartelle dei runlevels `/etc/rcX.d` (X è il numero del runlevel) sono in realtà dei link ("symlink") agli script nella cartella `/etc/init.d/`.

Tali link sono preceduti da una S ("start") o una K ("Kill"), seguite da un numero. Le applicazioni precedute da "S" verranno avviate nel relativo runlevel.

Attualmente la documentazione in rete per upstart è piuttosto scarsa, ed in realtà, a meno che non abbiate delle necessità particolari, è piuttosto difficile che abbiate mai la necessità di ritoccare i file di configurazione di upstart e i demoni da avviare.

\* \* \*

Qualunque sia il runlevel usato, vi possono essere delle operazioni che voi eseguite spesso in riga di comando, e magari preferireste che il sistema all'avvio le eseguisse in automatico.

Aggiungete il comando che vi interessa nel file `/etc/rc.local` inserendolo prima della riga "Exit 0".

\* \* \*

C'è un altro metodo per avviare applicazioni in automatico in GNOME, è un metodo molto semplice ed è quello che più vi consiglio.

Basta andare nel menu "Sistema --> Preferenze --> Sessioni --> Programmi d'avvio" e qui aggiungere il comando che intendiamo avviare.

Il sistema automaticamente aggiungerà il comando da voi inserito in un file nella cartella `"/home/NomeUtente/.config/autostart/`.

Ad esempio, basta aggiungergli `"/usr/bin/gdesklets"` per fare in modo che le **gdesklets** (sono dei piccoli tools per GNOME, da installare con Synaptic), vengano avviate in automatico ad ogni avvio di GNOME.



## Cominciare a programmare C con GTK e Java

### Guida: Passare da Windows a Linux (Quattordicesima parte)

Giusto due accenni sul cominciare a programmare in C, su cosa sono le GTK e come si potrebbe fare ad usarle in C.

Il **GTK+** (acronimo che sta per GIMP ToolKit) è un toolkit (insieme di strumenti, di cui il principale è la libreria **libgtk**) che serve per programmare usando le finestre, nello stile di GNOME. Per sviluppare facendo uso delle **gtk**, qui su Ubuntu, dovete installare da synaptic il pacchetto **libgtk2.0-dev**.

Il vostro programma, realizzato per essere compatibile con le GTK, va compilato con una istruzione simile a questa:

```
gcc -Wall -g primo.c -o primo `pkg-config --cflags --libs gtk+-2.0`
```

**GCC** (GNU Compiler Collection, in origine GNU C Compiler) è il compilatore C di linux. Alcuni parametri importanti:

-Wall : abilita tutti i messaggi di warning.

-g : aggiunge informazioni aggiuntive utili per il debugging.

-o nomefile : è il nome di come si chiamerà il file compilato.

pkg-config : è uno strumento che serve a comunicare al compilatore i percorsi giusti (PATH) che servono alle librerie. Nel nostro caso il comando pkg-config e i suoi argomenti sono contenuti fra degli apici storti che sono detti "backquotes". Questi vanno fatti premendo il tasto **ALTGR + `** (apice). Fra gli apici reversi va un comando (esattamente come se lo stessi eseguendo da shell).

pkg-config pesca le informazioni "gtk+-2.0" dall'apposito file che si trova nella cartella /usr/lib/pkgconfig/

Nel sito ho dedicato una apposita sezione alla programmazione con C e glade (designer di GUI).

Una piccola curiosità:

Invece di usare il comando gcc, spesso si usa il comando "cc", che in realtà è un link ad una versione del compilatore gcc che abbiamo installato (potremmo anche averne installata più di una).

Lo stesso comando gcc è un link.

Osservate:

```
attorianzo@attorianzo-desktop:~$ which gcc
/usr/bin/gcc
attorianzo@attorianzo-desktop:~$ ls -lah /usr/bin/gcc
lrwxrwxrwx 1 root root 7 2007-04-20 17:45 /usr/bin/gcc -> gcc-4.1
```

Come potete vedere, "gcc" punta alla versione 4.1.

Per quanto riguarda cc:

```
attoriano@attoriano-desktop:~$ which cc
/usr/bin/cc
attoriano@attoriano-desktop:~$ ls -lah /usr/bin/cc
lrwxrwxrwx 1 root root 20 2007-04-20 17:45 /usr/bin/cc -> /etc/alternatives/cc
attoriano@attoriano-desktop:~$ ls -lah /etc/alternatives/cc
lrwxrwxrwx 1 root root 12 2007-07-05 17:10 /etc/alternatives/cc ->
/usr/bin/gcc
```

Quindi "cc" punta a "gcc". (fonte: [forum Ubuntu](#))

\* \* \*

Per quanto riguarda il cominciare a sviluppare JAVA, scaricatevi da Synaptic il pacchetto **jdk1.4** (o versione più avanzata, se l'avete)

\* \* \*

Vi consiglio di scaricare anche il pacchetto **"python-wxgtk2.6"** (o versione più avanzata, se l'avete) che serve per installare le **"wxwidget"**.

---

## Gli script di shell, i file .sh

### Guida: Passare da Windows a Linux (Quindicesima parte)

Questi file .sh sono dei file di testo in cui vengono immesse delle operazioni da riga di comando. Un esempio di "rudimentale" file .sh:

```
#!/bin/sh
#Questo è un commento e non viene interpretato
echo "Questo verrà scritto"
```

Il primo rigo (**#!/bin/sh**) precisa che il file dovrebbe essere processato dall'interprete **sh**, che normalmente si trova appunto nella cartella **/bin/sh**.

La seconda riga è un semplice commento.

In terza riga vi è il comando **echo** che stampa il testo contenuto tra le virgolette.

Questi script .sh possono essere lanciati da una finestra di terminale facendo:

```
sh /percors/verso/ilfile.sh
```

oppure, se il file è "eseguibile", si può fare:

```
/percors/verso/ilfile.sh (se il file è "eseguibile")
```

oppure, se ci troviamo già nella cartella, si può eseguire semplicemente facendo:

```
./ilfile.sh (se il file è "eseguibile")
```

Per essere il file "eseguibile" bisogna renderlo tale facendo:

```
chmod +x ilfile.sh
```

## Il passaggio da Windows a Linux

### Guida: Passare da Windows a Linux (Sedicesima parte)

Se avete letto tutta la parte precedente di questa guida, sono certo che una minima idea su cosa voglia dire passare da Windows a Linux ve la sarete fatta.

Se può esservi sembrato tutto troppo complicato sappiate e ricordatevi che:

- Le guide sembrano molto più difficili lette sullo schermo di quanto non lo siano stampate su carta. E' questo uno dei motivi per cui vi consiglio di stampare questa guida e leggerla con calma.
- Si fa presto l'abitudine ad entrare nei meccanismi che regolano Linux, vi sembrerà incredibile ma è così, ed una volta entrati nel meccanismo vi sentirete così padroni del vostro sistema che ve lo adatterete come un guanto.
- Se non siete laureati in informatica ma avete sempre e solo usato Windows e le vostre conoscenze di Linux sono pari a zero, come così era per me, in una settimana di Linux avrete di certo configurato tutto il sistema, ed in un mese saprete tutto quello che c'è da sapere per usarlo in serenità, senza essere dei grandi esperti.
- Ricordate che quasi per ogni problema che incontrerete qualcuno, nel sito della vostra distribuzione e in centinaia di altri siti, blog e forum in rete, gratuitamente vi ha pensato ed ha scritto per voi una guida per risolvere il vostro problema.

Se nel vostro pc vi è ancora solo Windows e non avete ancora installato Linux, ma pensate di farlo, vi dò alcuni suggerimenti per una migrazione da Windows a Linux quanto più possibile serena:

Cominciate ad abituarvi, in Windows stesso, ad usare del software libero, in particolare quel software che ritroverete uguale da Windows a Linux.

Questo basterà già da solo a rendere il vostro passaggio più amichevole e meno traumatico.

Siete ancora in Windows giusto? Tanto per cominciare:

- Smettetela di usare **Internet Explorer** ed installatevi **Firefox**: è gratis
- Smettetela di usare **Outlook** per la posta elettronica, ed installatevi **Thunderbird**: è gratis.
- Smettetela di usare **Windows Media Player** per vedere i vostri film ed ascoltare musica, ed installatevi **VLC**: è gratis.

E' tutto software che potete **scaricare gratis dai rispettivi siti**.

Considerate che tutti i "siti preferiti" che avete salvato in Firefox su Windows, potrete tranquillamente passarli nel Firefox in Linux, e lo stesso vale per **Thunderbird: tutta la vostra posta elettronica, gli account, potrete ripassarli da Windows a Linux.**

Non appena avrete preso confidenza con questi tre software su Windows, che vi ritroverete uguali in Linux, scaricatevi una distribuzione Linux.

Quale fra le tante distribuzioni? Qui di campane ce ne sono tante, io personalmente vi consiglio **Ubuntu Feisty**, ma in molti consigliano, almeno per ora, di usare la versione "**Dapper**", che è la versione di Ubuntu più collaudata ed a lungo termine di supporto ("LTS").

Ubuntu Feisty è attualmente la versione di Ubuntu più avanzata al momento.

Io ho due hard disk nel mio pc, uno da 80 giga ed uno da 20 giga.

Ho lasciato Windows XP nell'hard disk da 80 ed ho dedicato l'intero HD da 20 giga ad Ubuntu.

Se però avete un solo, grosso Hard Disk nel vostro pc, dovrete fare in modo di liberare un pò di spazio per Linux.

Al momento dell'installazione sarà Linux stesso che vi chiederà quanto spazio, del vostro hard disk, volete dedicare a Linux e quanto volete rimanga a Windows. Così Linux stesso creerà la sua partizione.

Mi raccomando: Prima di fare QUALSIASI operazione sulle partizioni, **prima di installare Linux, fatevi una copia di ogni dato importante che avete in Windows**. Se combinate qualche sciocchezza durante la procedura di partizionamento e formattazione, rischiate di perdere tutti i dati.

Sicché salvatevi l'archivio della posta, i vostri mp3 e divx preferiti, documenti importanti, ecc. ecc..

Piccola nota: se il vostro hard disk è ancora vuoto e volete installare sia Windows XP che Linux, **installate sempre prima Windows** perché Windows vuole sentirsi sempre la prima donna nel vostro pc, e non accetta di essere installato in una partizione secondaria.

Personalmente, dopo avere avviato il pc dal CD di Ubuntu, **ho semplicemente seguito i passaggi del programma di installazione**, che mi formattò l'hard disk e vi installò rapidamente Ubuntu.

Il mio primo avvio di Ubuntu fu per me sorprendente: già tutto funzionava alla perfezione, porte USB, HUB USB, scheda grafica (ma senza accelerazione 3d), audio, scheda di rete e connessione ad internet mediante il mio router, e persino la webcam.

Se avete avuto la mia stessa fortuna, la prima cosa da fare è di cercarvi subito, in Synaptic, i pacchetti di installazione di **Firefox, Thunderbird, VLC** (Firefox dovrete trovarlo già attivo di default, gli altri non credo, quindi scaricateli con Synaptic).

Se tutto andrà bene, come lo è stato per me basterà poco tempo dal vostro iniziale entusiasmo che vi renderete conto che ci sono delle cose che non vanno.

Ecco i primi problemi in cui vi potreste imbattere:

- **Connessione ad internet non funzionante.** Soluzione: cercate nel sito della distribuzione su come configurare il vostro modem e la connessione ADSL.
- **Avviate un gioco 3d o google Earth e scoprite che non vi funziona l'accelerazione 3d.** Soluzione: prima guardate la sezione di questa guida relativa alle schede grafiche e poi cercate nel sito della vostra distribuzione come fare ad installare i driver giusti.
- **Non funzionano i file midi.** Soluzione: vedi l'apposita sezione in questa stessa guida.
- **Non funziona la riproduzione dei filmati o di qualche filmato, mpg, avi, DVD o altro.** Soluzione: Dovrebbe partire in automatico un tentativo di installazione del codec video giusto. Se questo non succede, leggi la sezione sulle schede grafiche in questa guida e poi vai al sito della distribuzione per cercare le soluzioni all'ultimo grido, fatte apposta per la tua distribuzione, per installare tutti i codec video.
- **Non funzionano le applet in JAVA ed i siti in FLASH su Firefox:** Soluzione: problema comune e di facile soluzione. Fai una ricerca nel sito della tua distribuzione su come installare Java e flash.
- **Non si sente l'audio nei siti in flash su Firefox.** Soluzione: vedi sezione "cartelle utilizzate da linux" in questa stessa guida( cerca "/etc/firefox/firefoxrc")

Per quanto riguarda il **cominciare** ad utilizzare Linux tutto quello che è scritto in questa guida dovrebbe essere sufficiente.

La mia personale raccomandazione è questa: siate pazienti. Linux è una grande comunità ed è piena di gente che vuole aiutarvi.

Visitate il forum della vostra distribuzione e non esitate a chiedere lì le vostre perplessità.

Abituatevi inoltre ad usare Google, Yahoo e gli altri motori di ricerca, compresi quelli dei forum della vostra distribuzione, per risolvere i vostri problemi quando questi si presentano.

Una ultima cosa: In questa guida vi sono elencati tutti i **principali problemini** a cui va incontro un utente che passa a Linux (in particolare Ubuntu), credetemi questa è **la guida che tanto avrei voluto avere io al momento del grande salto**.

Avete visto come per molti problemi la soluzione è stata dirvi "cercate nel forum" o "fate riferimento al sito della vostra distribuzione". Questo non per mia pigrizia o perchè mi annoiassi a spiegarvi le soluzioni ai vostri problemi, ma perchè i ragazzi dei forum tengono aggiornate le guide da loro scritte, specificatamente, per la distribuzione che avete voi, e vi danno il necessario supporto.

Nessuno mi ha pagato per scrivere questa guida, spero vi sia stata utile, e spero anche, che se riporterete dei brani di questa guida nei vostri siti, ne citerete la fonte.. Non è solo un fatto di onestà intellettuale, ma un fatto di.. Cuore.

Un buon linux a tutti da **Etnablog.com**