

Linux Base - Capitolo n. 1

Edizioni ByteMan (15-05-2005)

revisione: 25/01/2008

Introduzione

Linux è un sistema operativo multiutente, multiprocesso e multithread con le caratteristiche dei più avanzati ambienti Unix. Linux si caratterizza per essere stato sviluppato ex novo da un gruppo di volontari sparsi per il mondo in comunicazione tra loro principalmente grazie a Internet, secondo la filosofia del software "open source". Linux non appartiene a nessuna azienda né università ma è disponibile gratuitamente per chiunque intenda conoscerlo, utilizzarlo e anche modificarlo. L'hardware di riferimento è quello più economico e più diffuso nel mondo, e cioè i PC basati su microprocessori Intel, dall'80386 ai Pentium IV, fino ai più recenti. Esistono anche delle versioni che girano su piattaforme differenti come per esempio: Digital Alpha, PowerPC, Sun SPARC, Apple Macintosh, Atari ST/TT, Amiga, MIPS.

Il fatto che Linux sia un sistema operativo multiutente, multiprocesso e multitasking significa che più utenti possono lavorare sul medesimo computer, che ciascuno di essi può eseguire contemporaneamente più programmi e che ciascun programma può eseguire contemporaneamente più funzioni. Per ottimizzare l'utilizzo della memoria con molti processi attivi, il sistema può eseguire un programma leggendo dal disco solo un segmento per volta, detto pagina, necessario per l'esecuzione. Inoltre se uno stesso programma viene eseguito più volte contemporaneamente, per esempio da due utenti diversi, la parte di esso non soggetta a modifiche viene condivisa da tutte le istanze. Se poi la memoria diventa ugualmente insufficiente, Linux può depositare le pagine inattive sul disco in un'area dedicata, detta swap area (area di scambio). Come tutti i moderni sistemi operativi, prevede che l'accesso al disco utilizzi un'area di memoria, detta cache, per ottimizzare le prestazioni; a differenza di altri sistemi, però, quest'area non ha una dimensione fissa ma viene allocata dinamicamente a seconda delle necessità.

Dal punto di vista della connettività, Linux implementa completamente i protocolli TCP/IP (Transmission Control Protocol/Internet Protocol), compresi un buon numero di driver per le più diffuse schede Ethernet, SLIP (Serial Line Internet Protocol) e PPP (Point to Point Protocol) per la connessione via modem e PLIP (Parallel Line Internet Protocol). Vengono supportati tutti i servizi di base come TELNET, FTP (File Transfer Protocol), NFS (Network File System) e SMTP (Simple Mail Transfer Protocol).

Un progetto come Linux non avrebbe mai potuto avere successo se non fosse esistita Internet. E' grazie a Internet che persone di diversi continenti hanno potuto lavorare insieme per un progetto comune e diffonderne i risultati. In mancanza di canali commerciali, è ancora Internet il tramite tra gli sviluppatori e gli utilizzatori, ed è solo grazie a essa che si possono ottenere le informazioni più aggiornate sulle evoluzioni, i bug fix ecc. Collegandosi al sito web www.linux.org si possono trovare i link per la maggior parte dei siti riguardanti Linux. La documentazione "ufficiale" di Linux viene fornita normalmente con le distribuzioni.

Cenni storici

Per comprendere l'evoluzione di Linux è necessario fare almeno un accenno alla storia di Unix. La prima versione in monoprogrammazione di Unix venne sviluppata alla fine degli anni sessanta presso i laboratori Bell. Il nome Unix deriva dal fatto che le prime versioni consentivano l'utilizzo da parte di un solo utente per volta. Poiché la Bell Company si occupava principalmente di telefoni e non di computer, il nuovo sistema operativo fu progettato in modo tale da essere quanto più possibile adattabile a piattaforme hardware diverse.

Nel 1971 era pronta la versione funzionante in multiprogrammazione e nel 1973 gran parte del codice assembler venne sostituito con codice C. Il fatto di avere scritto il sistema operativo con un linguaggio di alto livello come il C, permetteva una più facile comprensione del suo funzionamento a chiunque. Per questo motivo e grazie alla semplice e coerente impostazione, Unix fu ritenuto dalle università americane uno strumento didattico ideale per i propri corsi di informatica. Dal canto loro, i laboratori Bell si dimostrarono disponibili a cedere, per fini didattici, la licenza d'uso e i sorgenti a cifre ragionate, per cui Unix si diffuse ampiamente in ambienti universitari e scientifici. Si susseguirono le nuove versioni, tra le quali fa spicco quella dell'Università di Berkeley, caratterizzata dalla sigla BSD (Berkeley Software Distribution).

Negli anni Ottanta Unix viene adottato come sistema operativo da molti importanti produttori di computer per soddisfare l'esigenza del mercato di una maggiore interoperabilità fra sistemi di marche e caratteristiche diverse. Vennero nello stesso periodo avviati dei processi per unificare i vari "dialetti" nati nel corso degli anni, ma gli interessi di parte dei promotori hanno ostacolato l'unificazione e a tutt'oggi questo traguardo sembra più distante che mai.

Nel 1983 Richard Stallman si rese conto che le politiche di protezione del software usate dalle aziende produttrici erano troppo restrittive per i propri gusti, e si convinse che questo comportamento frenava l'evoluzione dell'informatica, costringendo i programmatori a riscrivere daccapo sempre gli stessi programmi, invece di far evolvere quelli già esistenti. Decise quindi di sviluppare un sistema operativo, aiutato da quante più persone possibile, che fosse gratuito e i cui sorgenti fossero disponibili per chiunque. Nacque così la Free Software Foundation e il progetto del sistema operativo GNU. Come modello di riferimento fu preso il sistema operativo Unix in quanto solo in questo modo era possibile far lavorare su un progetto comune persone che neanche si conoscevano. Sin da principio si pensò al nuovo sistema operativo come a un'evoluzione di Unix piuttosto che a una semplice copia; l'acronimo GNU sta infatti per "Gnu's Not Unix" ("Gnu Non è Unix"). Stallman non volle che il software sviluppato nell'ambito del suo progetto diventasse di pubblico dominio in quanto in tal caso chiunque avrebbe potuto appropriarsene, farlo passare come proprio e andare contro la filosofia della fondazione. Fu stilato quindi un nuovo tipo di licenza d'uso appropriata che consentisse qualsiasi utilizzo del software a patto di riconoscerne la paternità e di non farlo passare come proprio apportando solo poche modifiche. Questo tipo di licenza è conosciuto come GNU GPL (General Public License) ed è stato adottato per numerosi pacchetti di software libero anche non direttamente impegnati nel progetto GNU, compreso lo stesso Linux.

All'inizio degli anni Novanta il sistema operativo GNU era quasi completo tranne per l'aspetto più importante, e cioè mancava un kernel. Il kernel (nocciolo) è il cuore, o forse sarebbe meglio dire il cervello, di ogni moderno sistema operativo al quale possiamo pensare come al programma che permette agli altri programmi di funzionare e di accedere alle risorse del sistema. In quegli anni, uno studente dell'Università di Helsinki, Linus Torvalds, cominciò a sviluppare per hobby un kernel prendendo spunto da una versione ridotta di Unix, chiamata Minix. Il 5 ottobre 1991 Linus annunciò su Internet la prima versione "ufficiale" del suo nuovo sistema operativo, che chiamò Linux versione 0.02, e di cui mise a disposizione i sorgenti. Questa prima versione girava su PC basati su microprocessore 80386 e non permetteva di fare molto, ma invogliò diversi studenti e programmatori a essere partecipi del progetto di Linus, che da allora non fu più solo a estendere le funzionalità del nuovo sistema operativo. Dobbiamo arrivare al 1994 per avere una versione che meriti il numero 1.0, vale a dire una versione riconosciuta dai propri progettisti come completa e affidabile, ma in quel momento Linux è già conosciuto e usato in tutto il mondo. Il gruppo di sviluppo non ha però cessato il proprio lavoro, anzi, si è ampliato e oggi Linux è un sistema operativo completo che può competere ad armi pari con i rivali commerciali. Tra gli UNIX "blasonati" cioè "di marca" inteso per questo senso l'essere non liberamente distribuibili, forse il più famoso e potente è IRIX di Silicon Graphics, uno unix altamente scalabile e versatile.

Cronologia essenziale

Luglio 1991 - Tutto inizia in Finlandia, **Linus Benedict Torvalds**, ancora un giovane studente dell'Università di Helsinki, inizia a lavorare al suo hobby: Linux. Il 3 Luglio lo si sente informarsi su usenet:

Hello netlanders, Due to a project I'm working on (in minix), I'm interested in the posix standard definition. Could somebody please point me to a (preferably) machine-readable format of the latest posix rules? Ftp-sites would be nice.

Torvalds giustificò poi la folle impresa con queste parole:

I couldn't afford some of the commercial OSES and I didn't want to run DOS or Windows -- I don't even know, did Windows really exist then?

5 Ottobre 1991 - Nello stesso anno viene rilasciata la versione **0.02**. Il [post su usenet](#) che ne annuncia la presenza è diventato un classico. Grazie all'archivio di 20 anni di storia di Usenet su [Google](#) possiamo ricordare.

Gennaio 1992 - Viene rilasciata la versione **0.12**. **Risulta relativamente stabile** e supporta hardware di vario genere.

Da questa versione in poi la crescita di Linux inizia a diventare progressiva e dirompente, sia come numero di coder che supportano lo sviluppo, sia come utilizzatori.

Earlier kernel releases were very much only for hackers: 0.12 actually worked quite well

Aprile 1992 - Vengono rilasciate la versione 0.95 e la **0.96**. Il salto è diretto dalla 0.12. Nascono le prime distribuzioni: la MCC Linux e la SLS.

1994 - Viene rilasciata la prima versione definitiva **1.0**. Nascono RedHat, Debian, SUSE tutt'ora fra le distribuzioni più diffuse. Linux, che resta copyrighted by Linus Torvalds, diventa ufficialmente un software aperto, abbracciando in pieno la General Public License (GPL) del movimento GNU Open Source. Grazie all'aumento esponenziale dell'interesse da parte della comunità mondiale nascono i primi **LUGs** (Linux User Groups), ormai diffusi anche in **Italia**.

1995 - Compiono sul mercato delle nuove distribuzioni commerciali come Caldera Linux. Kernel **1.2** uscito in Marzo.

Dal kernel 1.3 in sviluppo si passerà direttamente al 2.0.

1996 - Viene rilasciata la versione **2.0**. Compiono le prime versioni tradotte in più lingue.

Linux ha bisogno di una **mascotte**: nasce **TUX**, il pinguino più famoso del mondo.

1997 - Da qui in poi la storia di Linux diventa sempre più indipendente da Linus che, nel 1997, lascia la Finlandia per raggiungere Santa Clara nella Silicon Valley, dove lo aspetta, nella misteriosa start-up **Transmeta**, un ruolo che ai più non è chiaro.

Per anni, prima di annunciare al pubblico di produrre microprocessori a basso consumo e quotarsi al NASDAQ, Transmeta rimane un segreto impenetrabile intorno al quale si accumulano rumours e misteri:

- E' la società dove lavora Linus Torvalds (che continua a sviluppare Linux e non si capisce per cosa venga pagato)
- Fra i soci finanziatori figura Paul Allen (Microsoft co-founder)
- Assume programmatori e tecnici di altissimo livello
- Sfoggia per anni una **home page** che è un capolavoro di anti-marketing.

1999 - Dopo lunga attesa vede la luce il kernel **2.2**

2001 - Agli inizi dell'anno, dopo varie preversioni, su kernel.org appare l'immagine da 19.788.626 byte del **2.4.0** la prima release di un altro stable thread.

Oggi - Linux è una reale alternativa al mondo Microsoft e Unix, si ritrova milioni di utenti, migliaia di sviluppatori e un mercato in rapidissima espansione.

E' presente in sistemi integrati, è usato per il controllo di dispositivi robotizzati e ha volato a bordo dello shuttle, praticamente gira su oggetti elettronici di tutti i tipi, dai palmari alle workstation Alpha, risultando l'OS in assoluto più soggetto a porting.

Nessuno ormai si sogna di considerarlo un progetto sperimentale che non possa essere usato in applicazioni mission-critical, IBM "lo monta sui suoi server" (e lo pubblicizza pure), Microsoft lo considera il principale *nemico* da combattere (e non lesina risorse nel farlo), Oracle ci fa girare sopra il suo DB.

Le Distribuzioni

Occorre tenere presente che Linux identifica solo il kernel; per avere un sistema operativo completo è necessario integrare il kernel con una vasta mole di software, la maggior parte del quale viene presa solitamente dal progetto GNU. Linux stesso, pur non essendo stato sviluppato all'interno del progetto GNU, viene distribuito in accordo alla GNU GPL.

All'inizio chi voleva disporre di un sistema operativo utilizzabile era costretto a recuperare i vari componenti da siti diversi e assemblarli sperando che si incastrassero l'uno con l'altro. Questo compito non era semplicissimo e solo persone esperte, che non avessero troppa fretta, potevano provare a mettere insieme una versione funzionante. Col tempo, alcuni che erano riusciti ad assemblare tutti i pezzi misero a disposizione il frutto del loro lavoro agli altri. Nacquero così le cosiddette distribuzioni, vale a dire raccolte di software libero basate su Linux, in grado di formare un sistema operativo funzionante, arricchite magari di un qualche programma di installazione e amministrazione.

Le distribuzioni sono composte da molto software e quindi richiedono tempi piuttosto lunghi per essere scaricate via Internet. Alcune aziende hanno pensato così di metterle a disposizione su CD-ROM, facendo pagare agli acquirenti non la licenza dei software ma il lavoro necessario per la realizzazione e la distribuzione del CD-ROM in quanto tale. Tra le distribuzioni più conosciute attualmente troviamo, Debian, Slackware, Red Hat, Fedora, Mandrive (ex Mandrake), Suse, Caldera. Queste distribuzioni possono essere reperite nei negozi o in edicola a cifre bassissime, generalmente inferiori a 50 Euro, esclusa qualche distribuzione che ha aggiunto a Linux un supporto proprio per applicazioni commerciali ed un servizio assistenza e che quindi giustifica un prezzo superiore.

Dal menù **Riferimenti** è possibile accedere alle Home Page delle principali distribuzioni, mentre il sito www.distrowatch.com fornisce addirittura una classifica aggiornata in tempo reale delle prime 100 distribuzioni a livello internazionale.

Le Distribuzioni Live

Se si è appassionati di informatica o anche per semplice curiosità può capitare di essere interessati a provare Linux per iniziare a vedere direttamente come funziona, ma non si vuole procedere ad una normale installazione sul PC. In questi casi si può ricorrere alle cosiddette distribuzioni **LIVE** che permettono il caricamento di Linux direttamente dal CDROM, senza la necessità di installare alcunchè sul proprio prezioso e fragile hard disk.

Queste distribuzioni avviano direttamente un sistema Linux completo dal CDROM e possono essere usate in varie occasioni:

- Dimostrazioni dal vivo, in tempi rapidi, senza le potenziali complicazioni di un'installazione su Hard Disk;
- System Recovery, con la possibilità di operare sul sistema locale senza averlo bootato;
- Network monitoring in situazioni straordinarie, con la possibilità di sniffare, diagnosticare, eseguire test;
- Prove e valutazioni rapide e innocue di un sistema Linux sul proprio PC, senza il rischio di far danni.

Si raccomanda sempre di utilizzare le versioni più recenti e di considerare che l'esecuzione di molteplici applicazioni richiede memoria che va paginata sull'hard disk, se non basta la RAM. In genere è fondamentale che la distribuzione riconosca la propria scheda di rete o il modem; nel momento in cui questo avviene la configurazione del networking diventa semplice (tramite DHCP o configurazione manuale). Può capitare che alcune schede sonore o periferiche particolari (winmodem, schede integrate su motherboard ecc.) non vengano riconosciute.

Tra le più note ricordiamo: Knoppix, Mepis, Morphix, Dynebolic, DemoLinux.

Il Kernel

Per **kernel** si intende il cuore di un sistema operativo, il codice che gestisce le risorse presenti sul sistema e le rende disponibili alle **applicazioni**.

Il kernel si occupa principalmente di gestire:

- le comunicazioni con l'hardware del sistema
- i file system e la memoria
- l'accesso a queste risorse da parte dei processi (anche le applicazioni che girano sul sistema)

Le versioni del kernel Linux sono identificate con numeri dal significato ben preciso. Per esempio un kernel identificato da **2.4.12** ha:

2 come **major number** (I kernel della serie 1.x sono ormai piuttosto vecchi e pochissimo usati.

4 come **minor number**. Se è pari il kernel viene considerato **stable** e pronto per sistemi in produzione, se è dispari lo si considera in **development** e da usare con cautela o per sperimentazione. Le release stable sono sempre figlie delle devel precedenti. Per esempio al momento in cui si scrive, il kernel stable è alla versione 2.4.18, mentre quello in sviluppo è alla 2.5.7 (da cui deriverà la 2.6.x o direttamente la 3.0.x). Solitamente nei kernel stable si tende a fare maintenance ed a implementare solo le features strettamente necessarie, lasciando a quello in development lo sviluppo di nuove funzionalità.

12 è la revisione (patch) corrente. Questo è un numero progressivo che parte da 0. Da una revisione alla successiva possono passare da pochi giorni a varie settimane.

Esistono inoltre varie patch temporanee, anche non di Torwalds stesso (comuni sono le **-ac patch**, di Alan Cox) che rappresentano stadi intermedi prima della release di una revisione definitiva.

Ha senso utilizzarle subito solo in caso di utilizzo di kernel con gravi problemi di sicurezza o stabilità (sul proprio sistema) che vengono risolte con le relative patch parziali.

Esistono diversi tipi di kernel:

- **Monolitico** - E' un singolo file binario eseguibile in modalità *kernel* che contiene il gestore del processo, della memoria, del sistema e tutto il resto.
Esempi di tali sistemi sono **UNIX, Linux, MS-DOS**.
- **Microkernel** - E' un piccolo nucleo ridotto ai minimi termini che ha il compito di gestire le comunicazioni fra i gestori di sistema, processi separati eseguiti al di fuori dal kernel.
Esempi di tali sistemi sono **BE OS, GNU HURD, MINIX**.
- **Modulare** - Si intende un'estensione del kernel monolitico, con la capacità di caricare/scaricare parti di codice (moduli) secondo necessità e richieste.
Può esserlo **Linux** se lo si definisce in fase di configurazione pre-compilazione.

Il kernel **monolitico** è più veloce, poichè tutto il codice è già stato caricato al bootstrap dell'OS, ma di contro occupa maggiori risorse del sistema.

Un altro punto a favore è la maggiore stabilità: non richiede moduli evitando così pericolose dipendenze.

Il kernel **modulare** è quello utilizzato da tutte le distribuzioni in fase di installazione di LINUX su una macchina.

Knoppix Live

KNOPPIX è una raccolta di programmi GNU/Linux che gira completamente avviandosi da CD. E' in grado di riconoscere e di supportare automaticamente molti tipi di schede video, audio, SCSI e altre periferiche. E' il frutto del lavoro del tedesco Klaus Knopper che ha realizzato un autentico gioiello di immagine ISO, basata su Debian. Due Megabyte di software compresso su un CD di quasi 700 Mb, un boot rapido ed efficiente nel riconoscimento dell'hardware locale e la possibilità di eseguire moltissimi programmi da un ambiente grafico completo (KDE di default) ne fanno una distribuzione ideale. In caso di carenza di RAM crea dello SWAP space sull'hard disk e volendo lo si può installare anche su Hard Disk. KNOPPIX può essere adattato e usato per effettuare:

- Dimostrazioni e valutazioni di Linux
- Corsi di addestramento
- Recupero dati e sistema operativo (system recovery)
- Analisi di reti (network analysys)
- Installazione di Linux su Hard Disk

Per un uso dimostrativo non è necessaria nessuna installazione su disco fisso. Sul CD sono memorizzati in forma compressa circa 2 Gbyte di programmi e dati. Durante l'uso una decompressione trasparente, che poggia esclusivamente sulla RAM, consente l'utilizzo di tutti i programmi in esso contenuti.

La versione attuale (maggio 2005) è la 3.8.1 ed è scaricabile gratuitamente dal sito www.knoppix.org.

Inoltre essendo questo un autentico prodotto Open Source, soggetto alla GNU General Public License, è possibile prelevare anche i codici sorgenti per i pacchetti specifici su Internet all'indirizzo: www.knopper.net/knoppix/sources/.

I sorgenti dei pacchetti Debian installati possono essere trovati invece nei vari siti mirror di Debian.

Requisiti di sistema

- CPU Intel-compatibile (i486 o superiore)
- 20 MB di RAM per funzionare solo in modo testo, almeno 96 MB per utilizzare il modo grafico con KDE, ma per lanciare le varie applicazioni per ufficio sono raccomandati almeno 128 MB)
- Lettore CD-ROM/DVD di boot (IDE/ATAPI, Firewire, USB o SCSI), oppure floppy con CD-ROM standard (IDE/ATAPI o SCSI)
- Scheda video SVGA-compatibile
- Mouse seriale standard, o PS/2, o USB IMPS/2-compatibile

E' evidente che una maggiore quantità di RAM (da 256 MB in poi) rende il funzionamento più fluido, tuttavia si tenga presente che non è possibile confrontare le prestazioni LIVE con quelle dello stesso sistema installato su hard disk.

Disclaimer

Attenzione, riguardo al CD Knoppix 3.8.1 usato nel corso dei seminari, occorre tenere presente che l'uso del software avviene solo ed esclusivamente dietro la responsabilità dell'utente.

DICHIARAZIONE DI NON RESPONSABILITÀ: QUESTO È SOFTWARE SPERIMENTALE. USARE A PROPRIO RISCHIO. PER NESSUNA CIRCOSTANZA KNOPPER.NET, IL PROPRIETARIO DEL SITO, GLI ISTRUTTORI, POSSONO ESSERE RITENUTI RESPONSABILI DI DANNI ALL'HARDWARE O AL SOFTWARE, PERDITA DI DATI O ALTRI DANNI SIA DIRETTI O INDIRETTI CAUSATI DALL'USO DI QUESTO SOFTWARE.

IN ALCUNI STATI IL SOFTWARE CRITTOGRAFICO PUÒ ESSERE SOGGETTO A RESTRIZIONI DI IMPORTAZIONE/ESPORTAZIONE O SOGGETTO A BREVETTI SOFTWARE. QUESTO SOFTWARE NON PUÒ ESSERE USATO O DISTRIBUITO IN QUESTI STATI. ALTRIMENTI È SOGGETTO ALLE NORMALI REGOLE DELLA LICENZA GPL. SE NON SI ACCETTANO QUESTE REGOLE NON È POSSIBILE USARE O DISTRIBUIRE QUESTO SOFTWARE.

Avviare il CD Knoppix

Nell'ipotesi che la configurazione BIOS consenta il boot da CD, basta inserire il CD Live Knoppix nel lettore e riavviare il computer. Dopo l'avvio Knoppix 3.8 presenta una schermata simile alla seguente:



KNOPPIX consente all'utente di effettuare una serie di scelte all'avvio del sistema (lingua, interfaccia grafica, risoluzione video, esclusione di dispositivi, ecc.). Per effettuare tale operazione bisogna premere **F2** alla schermata di boot di KNOPPIX. Apparirà una schermata con l'elenco delle scelte possibili (CheatCodes), per effettuare una scelta bisogna scrivere:

knoppix opzione=valore

se si vogliono impostare più opzioni le si può scrivere una dopo l'altra separandole con uno spazio:

knoppix opzione1=valore1 opzione2=valore2 ...

L'elenco completo dei CheatCodes con una breve descrizione di ciascuno di essi (in inglese) si trova nel file **knoppix-cheatcodes.txt** nella directory **KNOPPIX** del CD, per comodità riportato anche dal menù **Guide & Tutorial**.

NOTA: in fase di avvio KNOPPIX utilizza la tastiera USA, per cui il carattere " = " non si trova sopra il tasto " 0 ", ma in corrispondenza del tasto " ` " (i accentata).

Per esempio volendo impostare la lingua italiana occorrerà digitare al boot:

knoppix lang=it [INVIO]

oppure volendo impostare anche la risoluzione video e la profondità di colore:

knoppix lang=it screen=1024x768 depth=24 [INVIO]

A questo punto inizia una fase molto delicata, il boot del sistema, che comprende anche il riconoscimento dell'hardware. Generalmente nel 90% dei casi tutto si conclude perfettamente prima con la visualizzazione di una schermata di configurazione simile a questa:



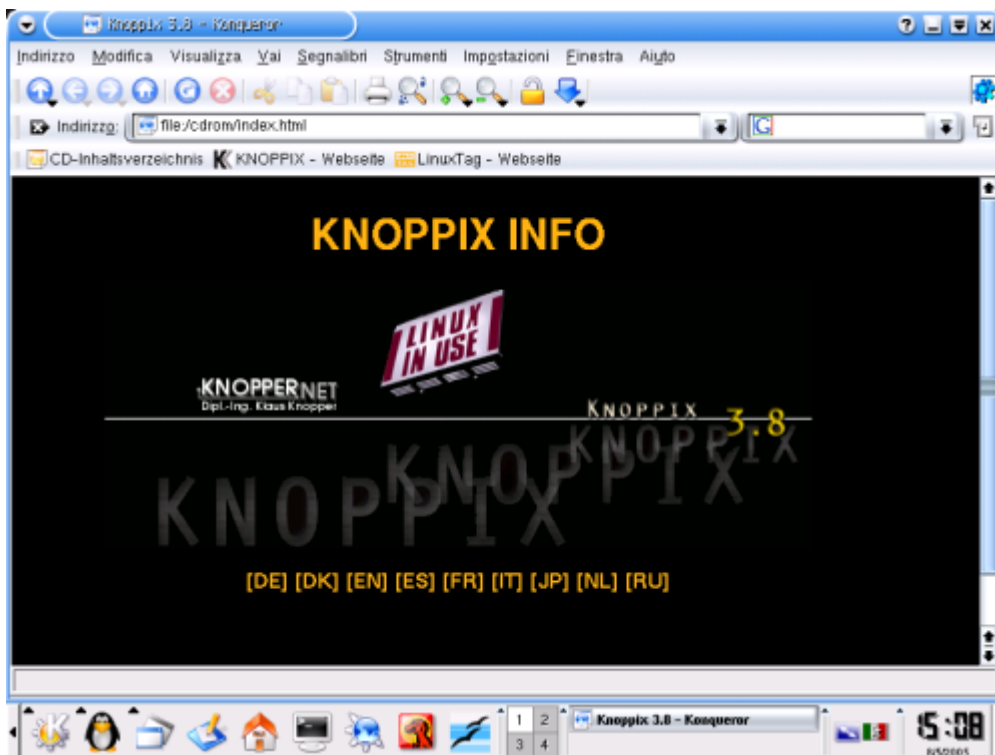
Welcome to the **KNOPPIX** live Linux-on-CD!

```
Found SCSI device(s) handled by BusLogic.o.
Scanning for USB/Firewire devices... Done.
Enabling DMA acceleration for: hda      [VMware Virtual IDE Hard Drive]
Enabling DMA acceleration for: hdc     [VMware Virtual IDE CDROM Drive]
Accessing KNOPPIX CDROM at /dev/scd0...
Total memory found: 256012 kB
Creating /randisk (dynamic size=199124k) on shared memory...Done.
Creating directories and symlinks on randisk...Done.
Starting init process.
INIT: version 2.78-knoppix booting
Running Linux Kernel 2.4.27.
Processor 0 is Genuine Intel(R) CPU 3.20GHz 3209MHz, 1024 KB Cache
ACPI Bios found, activating modules: ac battery button fan processor the:
USB found, managed by hotplug: (Re-)scanning USB devices... sync:[001 1]
Autoconfiguring devices... ██████████/
```

Primi passi

La prima schermata

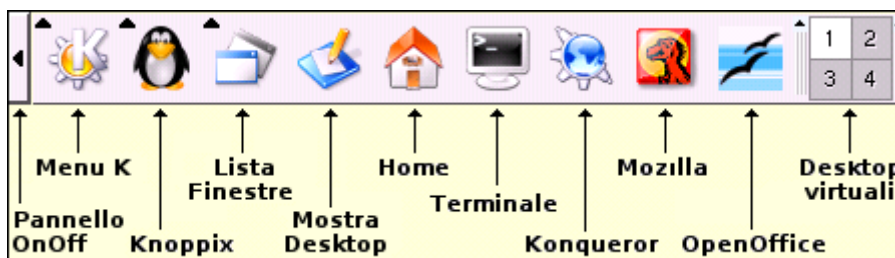
A questo punto Knoppix 3.8 è pronto per lavorare e mostra nella sua schermata iniziale il navigatore **Konqueror** con il quale è possibile sia esplorare il contenuto dell'albero delle directory sia navigare in Internet (se c'è una connessione attiva).



Per il momento possiamo però chiudere questa finestra ed analizzare invece le modalità base di utilizzo di Linux. Diciamo subito che per padroneggiare questo sistema operativo è necessario imparare ad effettuare le varie operazioni sia in modalità grafica sia in modalità testo.

La modalità grafica

Dalla modalità grafica si opera utilizzando le icone indicate in figura:

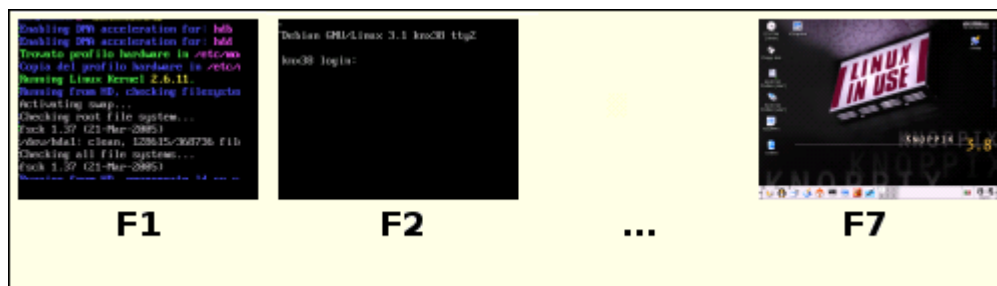


- **Menù K.** Permette di lanciare i programmi installati nel sistema. È l'equivalente del menu Start sotto Windows. I programmi sono organizzati in comode categorie, in modo da permettere di trovare facilmente l'applicazione desiderata. Consente inoltre una serie di **azioni** correlate allo spegnimento del sistema o al cambiamento di sessione.
- **Pannello OnOff** è un pulsantino comodo quando si vuole sfruttare tutta l'area video, il suo effetto è quello di togliere/ripristinare il pannello delle icone.

- **Knoppix** serve per la configurazione del sistema (scheda di rete, internet, etc...).
- **Mostra Desktop** Con un clic riduce a icona tutte le finestre aperte e visualizza il desktop. Un altro clic e riporta le finestre allo stato originale. Molto comodo quando tutto lo spazio è occupato dalle finestre aperte e si desidera accedere alle risorse del desktop.
- **Lista Finestre** Fornisce un elenco con tutte le finestre aperte per ciascun desktop virtuale.
- **Home** entra direttamente nella directory dell'utente,
- **Terminale** attiva un terminale virtuale per lavorare a linea di comando.
- **Konqueror** navigatore universale per internet e per il sistema.
- **Mozilla** browser specializzato per internet.
- **OpenOffice** attiva la suite per ufficio per videoscrittura, foglio di calcolo, presentazioni, web editor, etc...
- **Desktop** consente la selezione di ciascuno di 4 desktop indipendenti. Per modificare il numero dei desktop virtuali a disposizione, cliccare col tasto destro su una delle cifre identificative e scegliere la voce **Configura desktop virtuali**.

La modalità testo

L'altro modo per lavorare in Linux utilizza la modalità testo tramite un terminale non grafico. Occorre allora sapere che un terminale Linux è un luogo dove effettuare l'input ed ottenere l'output del sistema e di solito comprende almeno una tastiera ed un monitor. I sistemi Linux sono configurati in modo da avere un certo numero di console virtuali a cui si accede dal medesimo terminale di console fisico. Queste console sono identificabili tramite i tasti funzione: **F1**, **F2**, ..., **Fn**. Knoppix utilizza la console **F7** per lavorare in modalità grafica ed è, appunto, su questa console che ci troviamo subito dopo l'avviamento del sistema.



Quindi le operazioni di avvio avvengono su **F1**, mentre su **F7** viene attivato il server **X** per lavorare in modalità grafica. Tutte le altre console, dalla **F2** alla **F6** sono a disposizione per altri lavori. Il passaggio da una console all'altra si effettua con la combinazione di tasti:

- **Ctrl-Alt-Fn**

Si noti che lavorando con terminali **tty** o **vc** si può anche utilizzare la combinazione **Alt-Fn**.

Si sottolinea che non si tratta di semplici finestre per cui se si blocca il terminale video si perde il controllo del sistema, ma di autentici terminali virtuali. In sostanza, ammesso che si blocchi **F7** per una qualsiasi ragione, è possibile riprendere il controllo del sistema da una delle altre console a disposizione, per esempio la **F2**, digitando **Ctrl-Alt-F2** effettuando il **login** ed intervenendo nella maniera più opportuna, in modalità testo. E' uno dei motivi per cui si raccomanda vivamente di **imparare ad operare sia in modalità grafica sia in modalità testuale**.

Spegnere il sistema

Questa azione va sempre fatta con la massima attenzione, la si effettua con 2 modalità diverse:

- Da terminale grafico tramite il **Menù K** selezionando le varie possibilità (spegnimento, riavvio,

sessione, etc...).

- Da console tramite i comandi:

shutdown -h now	(spegnere)
shutdown -r now	(riavviare)
Ctrl-Alt-Canc	(riavviare)

File System (parte 1)

L'albero delle directory

L'organizzazione del filesystem di GNU/Linux è piuttosto diversa da quella di Windows alla quale probabilmente si è abituati. A prima vista sembra criptica: non ci sono quelle belle cartelle con nomi esplicativi chiamate "Programmi" o "Documenti", però occorre tenere presente che UNIX è stato scritto da hacker per gli hacker, per cui "meno lettere possibile, ma significative". Come si può notare i nomi delle directory hanno nomi mediamente di quattro lettere e quando si lavora al terminale di un server senza interfaccia grafica, meno si scrive e più si è veloci.

Gli eseguibili, in particolare quelli di sistema, non vengono disseminati dappertutto, ma seguono delle regole ben precise. Si rinuncia ad un po' di arbitrio in cambio di una maggiore organizzazione. Unix a differenza di Win ha un **unico albero** nel quale vengono innestati i rami di ogni periferica che viene "montata". In Windows ogni **disco** (anche se astratto, di rete) invece è un albero separato individuato da una lettera (A:, B:, C:, D:, ...) ciascuno con la sua root e le sue directory, tutto ciò è di derivazione DOS (Disk Operating System) che è un sistema operativo basato sui dischi.

/	E' il punto di origine di tutto il filesystem e viene indicato con il nome di radice o root (Attenzione a non confonderla con la directory /root). Non deve contenere file (se non il kernel in casi particolari) ma solo le directory stabilite dallo standard FHS.
/bin	Contiene i programmi eseguibili fondamentali per la gestione del sistema (binaries, binari) che possono essere usati sia dall'amministratore (/root) sia dagli utenti (in /sbin che vedremo dopo vanno i binari di uso esclusivo del superutente). I comandi che impareremo ad usare più tardi sono tutti contenuti in questa directory!
/boot	Boot contiene tutto ciò che serve al processo di avvio (bootstrap) del sistema: per esempio il kernel (nocciolo del sistema operativo) stesso.
/dev	Device contiene i file speciali di dispositivo, uno per ogni dispositivo fisico del computer. In pratica è come se si trattasse di "driver" dei dispositivi. In Unix ogni cosa è un file.
/etc	Contiene i dati di configurazione del computer, distribuiti su molti file ed ulteriori sottodirectory dal nome a volte esplicativo a volte meno. Tutti i file sono di tipo ASCII e sono modificabili, con qualsiasi editor di testo, secondo la sintassi propria dell'applicazione. I comandi grafici per la gestione del sistema vanno a modificare i file che sono qui contenuti, quindi e' possibile una gestione mista grafica/testuale.
/home	Home contiene i dati personali di tutti gli utenti (tranne root) del sistema, suddivisi in una directory per ogni utente che ha solitamente come nome lo username dell'utente. Spessissimo si tratta di un filesystem secondario (un altro disco o partizione o condivisione di rete) montato in questo punto (per ragioni di sicurezza e di scalabilità)
/lib	Libraries contiene i file di libreria condivisi necessari per l'avvio del sistema e per i programmi di uso generale che si possono trovare in /bin e /sbin. Le librerie che riguardano solo programmi collocati in /usr non vanno qui. Prevede anche la subdirectory /lib/modules che contiene i moduli caricabili dal kernel.
/mnt	Mount è il punto di innesto standard per filesystem montati temporaneamente. Normalmente dovrebbe essere vuota e riservata ai montaggi temporanei (come specificato da FHS v2.3) ma finora è stato uso comune montarvi permanentemente i filesystem ospiti /mnt/floppy , /mnt/cdrom , /mnt/removable (zip, usb). Montare e smontare i filesystem ospiti è una caratteristica di Unix.
/opt	Optional serve per contenere applicazioni opzionali ed altri software, ognuno in una sua sottodirectory. I file di configurazione di questi pacchetti dovrebbero poi andare in /etc/opt e i file variabili in /var/opt . Peccato che non sempre questa convenzione venga rigorosamente rispettata, spesso in favore di un po' di subdirectory aggiunte in /usr .
/proc	Processes è una directory vuota che viene usata dal SO per montare il filesystem virtuale omonimo (solo in GNU/Linux non fa parte dell'HFS). I file e le directory contenuti in questo filesystem virtuale sono indispensabili ai programmi che hanno la necessità di accedere alle informazioni sul sistema.
/root	Questa è la home del superutente (root) contiene i dati (documenti) e le impostazioni dell'utente amministratore del sistema. Ci sono molti validi motivi per tenerla separata dalle home degli utenti comuni: impedire più facilmente accessi indesiderati, poterla

	mantenere nel filesystem principale anche se le home degli utenti sono su un filesystem separato in modo da essere in grado di accedervi anche quando il sistema viene avviato in condizioni di emergenza e non si possono montare altri filesystem.
/sbin	System Binaries contiene i binari di sistema, utility per l'amministrazione del sistema essenziali per avvio, restore, riparazione del sistema in aggiunta a quelli contenuti in /bin .
/tmp	Contiene i file temporanei dei programmi. Potrebbe addirittura essere collocata in un disco virtuale basato su memoria volatile (RAMdisk).
/usr	Costituisce la seconda maggior sezione del filesystem. Si compone di una struttura molto articolata, /usr contiene solo dati condivisibili e statici (ogni file di contenuto variabile o che contenga opzioni specifiche per la macchina locale dovrebbe stare in una delle directory che hanno questo scopo: /etc o /var). In pratica /usr replica l'albero principale per contenere tutti i programmi non essenziali per l'amministrazione del sistema (ovvero tutti quelli che usiamo di più se non facciamo il sysadmin di mestiere.)
/var	Contiene le directory e i file di contenuto variabile (cioè un po' di tutto di quello che non può stare in /usr perchè /usr deve poter funzionare anche in sola lettura).

Lo standard FHS

La disposizione di directory e file di Unix (e quindi di GNU/Linux) segue lo standard **FHS** (Filesystem Hierarchy Standard) [con qualche piccola eccezione...(ad es: /proc)]. FHS è una raccolta di requisiti e linee guida per la disposizione di file e directory nei sistemi operativi Unix-like. Il suo scopo è quello di garantire l'interoperabilità di applicazioni, strumenti di amministrazione, ambienti di sviluppo e scripts, così come una maggiore uniformità nella documentazione di questi strumenti.

Attenzione a / (slash) e \ (backslash). Internet, e non solo, usa la convenzione UNIX, ed è per questo che gli indirizzi URL usano lo slash, esattamente come nei pathname Unix.

Ogni ramo (o anche sottoramo) può essere un filesystem diverso (periferica diversa o partizione diversa) che si dice **montato** (innestato) sull'albero.

In caso di installazioni toste si usa mettere alcune directory su dischi o partizioni separate (/boot, /home, /var, /usr, /usr/local, /tmp, /var) per mantenere intatti certi dati in caso di guasti hardware o upgrade del SO e mantenere il più leggero possibile il filesystem di base che deve partire nudo in caso di guai.

Il punto in cui si innesta nell'albero un filesystem separato si chiama mountpoint (punto di innesto).

Lo standard FHS v2.2 richiede, in particolare, che all'interno di / siano obbligatoriamente presenti (o soft-linkate) le seguenti directory: /bin /boot /dev /etc /lib /mnt /opt /sbin /tmp /usr /var. Sono richieste, solo se i relativi sottosistemi sono installati, anche: /home /lib (condivise per sistemi diversi) /root.

FHS v2.3 ha apportato alcune variazioni che alcune distribuzioni hanno già implementato: /media (punto di mount per filesystem ospitati costantemente floppy, cdrom, rimovibili), /srv.

Ed infine c'è sempre montata (ma non si vede) almeno una partizione di swap ovvero di memoria virtuale che si aggiunge alla RAM in caso di necessità. Nei sistemi Intel x86 si possono montare fino a 16 partizioni di swap contemporaneamente! In caso di problemi alla partizione che la ospita è possibile disattivare/attivare le aree di swap con i comandi **#swapoff** e **#swapon**.

Altre directory notevoli

Vediamo adesso alcune sottodirectory di particolare importanza ed alcune directory che sono state incluse recentemente dallo standard FHS o che vengono utilizzate solo da alcune distribuzioni.

/initrd	Fuori standard, usata da Mandrake solo durante l'avvio del sistema, poi praticamente vuota.
/media	Introdotta con lo standard FHS 2.3, è il punto di mount per filesystem ospitati costantemente floppy, cdrom, rimovibili.
/usr/bin	Strumenti ed applicazioni per gli utenti.
/usr/dict	Dizionari ed elenchi di parole. Viene lentamente sostituita dalla directory /usr/share/dict .
/usr/doc	Documentazione varia di sistema.

/usr/games	Giochi e passatempi.
/usr/info	File per il sistema ipertestuale GNU Info.
/usr/lib	Librerie utilizzate dagli strumenti ed applicazioni presenti in /usr/bin .
/usr/local	Gerarchia in cui vengono installati i programmi locali (che non possono essere eseguiti dalle macchine in rete con noi). L'albero che si dirama da questa directory non viene sovrascritto in caso di upgrade (aggiornamento) del sistema.
/usr/man	I manuali in linea che vengono letti con il comando man .
/usr/share	Contiene i dati che possono essere condivisi tra macchine con piattaforme hardware differenti (i386, alph, PPC...) (sono molto più di quelli che pensiamo possano esistere, l'interoperabilità è un miraggio solo nelle pubblicità dei SO proprietari.). Contiene anche le manpages in /usr/share/man (i comandi sono uguali e hanno le stesse opzioni su tutte le piattaforme. Se vi sedete davanti a un IBM S/370 o ad una macchina Sparc con Linux a bordo, sarete perfettamente in grado di usarla come il vostro portatile.). Alcune sue sottodirectory hanno l'equivalente in /usr .
/usr/src	Contiene il codice sorgente dei programmi da compilare e in /usr/src/linux il codice sorgente del sistema operativo stesso (prima o poi occorrerà imparare a ricompilarlo.)
/usr/tmp	Altra directory per file temporanei.
/usr/X11R6	Contiene la gerarchia riguardante il sistema grafico di Unix ed i suoi programmi. Nella directory /usr/X11R6/bin sono contenuti gli eseguibili attivati da X.
/var/local	Dati variabili relativi alla struttura /usr/local .
/var/log	Contiene i log dei programmi e di sistema (importante controllarli spesso).
/var/spool	Contiene le code di stampa e delle e-mail ancora da consegnare agli utenti.
/var/www	Contiene le pagine web se c'è installato apache o un altro server web .

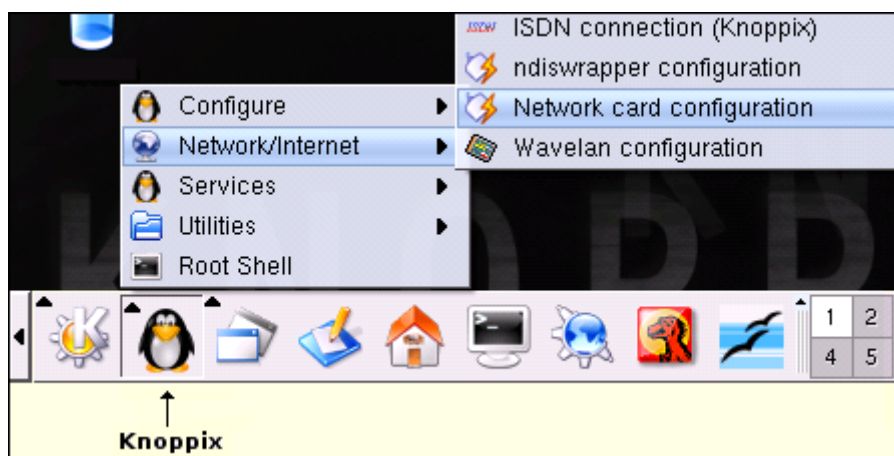
Configurazione scheda di rete

Introduzione

Oggi la gran parte del software disponibile per Linux si preleva direttamente da internet e quindi è indispensabile mettersi subito in condizione di collegarsi alla rete. Eseguiamo questa procedura nelle solite 2 modalità: grafica e testuale.

Configurazione da modo grafico

Si apre il **Menù Knoppix** e si selezionano le voci **Network/Internet** e poi **Network card configuration** come indicato in figura:



a cui seguiranno una serie di box con delle domande, queste le risposte da dare in base alla configurazione presente nell'aula:

Use DHCP broadcast ?	NO	
Please enter IP Address for eth0	192.168.12.nn	Postazione nn=13,14,15,...
Please enter Network Mask for eth0	255.255.255.0	
Please enter Broadcast Address for eth0	192.168.12.255	
Please enter Default Gateway	192.168.12.3	
Please enter Nameserver (s)	151.99.125.2	

Configurazione da modo testo

Quando eseguito graficamente può essere realizzato anche a linea di comando. Eseguire il LOGIN come ROOT, o prendere i privilegi di superutente da una sessione immettendo il comando:

```
su -
```

Il trattino orizzontale (segno meno) dopo il comando **su** e' consigliato in quanto serve per far caricare alla shell le variabili d'ambiente dell'utente che si sta tentando di impersonare (in questo caso root). A questo punto normalmente viene chiesta la password, ma la distribuzione Knoppix Live non la chiede.

Digitare quindi i seguenti 3 comandi:

```
ifconfig eth0 192.168.12.nn netmask 255.255.255.0 broadcast 192.168.12.255
route add default gw 192.168.12.3
echo "nameserver 151.99.125.2" >> /etc/resolv.conf
```

E adesso un minimo di approfondimenti.

Il comando **ifconfig** permette di gestire i parametri delle schede di rete, per esempio digitato da solo produce un output simile al seguente:

```

ifconfig
eth0 Link encap:Ethernet HWaddr 00:11:2F:85:F0:B0
      inet addr:192.168.0.10 Bcast:192.168.0.255 Mask:255.255.255.0
      inet6 addr: fe80::211:2fff:fe85:f0b0/64 Scope:Link
      UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
      RX packets:1821 errors:0 dropped:0 overruns:0 frame:0
      TX packets:2241 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:1000
      RX bytes:988652 (965.4 KiB) TX bytes:341820 (333.8 KiB)
      Interrupt:5 Memory:feaf8000-0

lo    Link encap:Local Loopback
      inet addr:127.0.0.1 Mask:255.0.0.0
      inet6 addr: ::1/128 Scope:Host
      UP LOOPBACK RUNNING MTU:16436 Metric:1
      RX packets:691 errors:0 dropped:0 overruns:0 frame:0
      TX packets:691 errors:0 dropped:0 overruns:0 carrier:0
      collisions:0 txqueuelen:0
      RX bytes:392947 (383.7 KiB) TX bytes:392947 (383.7 KiB)

```

Si tratta di una tabella che riporta le configurazioni attuali delle schede installate. Si può notare che oltre le informazioni riguardanti l'interfaccia **eth0** ci sono quelle riguardanti l'interfaccia **lo**, che è l'interfaccia di loopback presente di default. La sigla eth0 è utilizzata dai sistemi operativi LINUX per definire l'interfaccia di rete primaria. Se ci sono altre schede di rete appariranno altre informazioni per ogni scheda di rete. In particolare si notino l'indirizzo IP e la NETMASK assegnati alla scheda di rete in uso.

Il comando **route** permette invece di gestire la tabella di routing, si osservi ad esempio l'output prodotto dal comando seguente:

```

route -n
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.0.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0
0.0.0.0 192.168.0.1 0.0.0.0 UG 0 0 0 eth0

```

Il compito di risolvere i nomi di dominio è affidato al file **/etc/resolv.conf**, è sufficiente, infatti, inserire in questo file una riga con l'informazione desiderata. Per far questo si potrebbe ricorrere all'uso di un editor, ma è più semplice utilizzare il comando **echo** e la **ridirezione** dell'output, come fatto in precedenza.

Comandi Base (parte 1)

Ricordiamo che in Linux è basilare saper padroneggiare il sistema da terminale a linea di comando. Iniziando una rassegna dei comandi di base che ci permetteranno l'interazione con il s.o. vogliamo ricordare che, per evidenti motivi, non è possibile spiegarli tutti e con tutte le possibili opzioni, pertanto per ogni informazione aggiuntiva si può utilizzare il manuale interno digitando:

```
man <comando>
```

In Linux c'è una netta separazione tra l'amministratore di sistema (**root**), che ha i pieni poteri, e gli utenti normali con poteri limitati. Tutti però hanno a disposizione una **directory** per tenervi dentro i propri file. Per l'amministratore c'è una directory riservata (**/root**), mentre gli altri utenti dispongono di una directory personale posta all'interno della directory **/home**, per esempio: **/home/user1**, **/home/user2**, etc.

Una volta entrati nel sistema ci si ritrova a lavorare nella propria home directory, ma spesso si ha la necessità di muoversi fra le directory. Il comando **cd** è quello che serve.

```
cd /tmp
```

serve per entrare nella directory **/tmp**.

```
cd
```

digitato senza argomenti, riporta alla propria home directory.

```
cd -
```

utilizzato con l'argomento - riporta alla directory di provenienza, infatti in questo caso ci si ritrova nuovamente nella directory **/tmp**.

Per sapere tutto su **cd** basta consultare il manuale interno:

```
man cd
```

Il secondo comando è **dir**, che elenca file e directory presenti nella directory corrente, se non altrimenti specificato. Quasi sempre al suo posto si usa il comando **ls** (list). Essi hanno la stessa funzione e la stessa sintassi, basta confrontare le pagine di manuale relative all'uno ed all'altro comando.

Normalmente **ls** visualizza i file in ordine alfabetico e non visualizza i file nascosti, quelli cioè il cui nome inizia con un punto: per vedere anche questi file, occorre utilizzare l'opzione **-a** (all) che svelerà la presenza anche di due directory: "." e ".."; la prima è la rappresentazione della directory corrente, la seconda della directory superiore nell'albero delle directory. Ecco alcuni casi:

```
ls                lista directory corrente
ls -a            lista directory corrente, inclusi i file nascosti
ls .             lista directory corrente
ls ..           lista directory superiore
ls /tmp         lista directory /tmp
```

Un caso particolare è costituito dal "long listing format", usato per ottenere maggiori informazioni sui file:

```
ls -l            long listing
```

ed un possibile output potrebbe essere questo:

```
-rw-r--r--  1 usr1  usr1    9306 mag 28 18:41 prova.txt
drwxr-sr-x  5 usr1  usr1    4096 gen 13  2005 GNUstep
drwx--S---  2 usr1  usr1    4096 apr 18 02:08 Mail
drwxr-sr-x  2 usr1  usr1    4096 lug 21 17:23 mirrored
drwx--S---  2 usr1  usr1    4096 gen  8  2005 nsmail
```

```
-rw-r----- 1 usr1  usr1  155403 mag 12 11:42 out.jpeg
drwxr-sr-x   2 usr1  usr1    4096 apr  9 01:56 prove
drwxr-sr-x   2 usr1  usr1    4096 mar  8 15:38 public_html
```

Qui notiamo molte informazioni in più su ogni file: oltre al nome è possibile vedere, ad esempio, a chi appartiene, la sua grandezza e la data di creazione. Il blocco di informazioni riportate sulla sinistra sarà esaminere in seguito, è un insieme di caratteri che definiscono le caratteristiche di ogni file.

Altre opzioni, forse meno utilizzate, ma sempre utili sono le seguenti:

```
ls -S          ordina i file in ordine di dimensioni
ls -t          ordina i file in ordine di creazione discendente (i più recenti in alto)
ls --color     distingue tramite colori i vari tipi di file
ls -h         esprime la dimensione del file in un formato più leggibile
               (ad esempio, 2K, 32M, 4G, etc.); non ha senso se utilizzato
               da solo, ma può essere utile se abbinato con -l
```