

Prima dispensa: software, licenze, nomenclatura minima

[(c)2002 – Jean François Panico]

Struttura del corso

Il corso di Linux che andiamo a realizzare segue questo tipo di indirizzo: cioè favorire l'uso e la conoscenza del software Open Source.

A chi è indirizzato

Il corso è indirizzato a principianti di Linux, a persone cioè che abbiano già conoscenza della struttura di un calcolatore e del software ma che non hanno mai usato né installato Linux. Nell'esperienza di tutti coloro che usano o hanno usato Linux, come anche altri sistemi operativi, esiste un periodo, detto di "spippolamento" durante il quale si impara a conoscere un po' il sistema provando e riprovando, facendo errori più o meno gravi. Lo scopo del corso è di limitare al massimo questo periodo facendo una prima installazione, spiegando la struttura del sistema, i comandi e come fare le cose più necessarie.

Inoltre nelle ultime lezioni verranno fatti vedere alcune applicazioni professionali di GNU/Linux: Linux come client di rete, come server, come gestore della sicurezza e come piattaforma di sviluppo affidabile.

Ci viene spesso chiesto di organizzare anche corsi avanzati come Linuz.it: come è spiegato nell'oggetto sociale della nostra azienda, noi accettiamo e cerchiamo di incentivare la partecipazione di chiunque.

Com'è strutturato

Il corso è suddiviso in otto seminari che tratteranno almeno i temi seguenti (per un programma completo e aggiornato dei seminari vi rimandiamo all'indirizzo www.linuz.it/download/prog_corso.doc).

- Introduzione al corso, presentazione di Linuz.it e di Linux.
- Primi comandi
- Struttura di un sistema Linux
- Documentazione e risorse su Internet
- Distribuzioni
- Esempi di installazione
- Il kernel
- Uso dei sorgenti e dei compilatori
- Rudimenti di reti tcp-ip e reti con Linux
- Interfaccia grafica X
- Applicazioni di uso più comune

Cos'è Linux

Il primo equivoco da eliminare è il nome: anche noi di Linuz.it, come gran parte della comunità informatica, subiamo questo piccolo fraintendimento. Linux è solo il kernel del sistema, cioè la parte centrale che permette al sistema operativo di funzionare, ma che non potrebbe servire a niente se non fosse accompagnato da tutta una serie di programmi che lo rendono usabile: la shell che prende i comandi dell'utente, tutti i demoni (programmi di sistema invisibili agli utenti) che eseguono ogni tipo di compito, i programmi veri e propri e via così. Tutto questo software è parte integrante di ogni sistema basato su Linux ma non è Linux!

Buona parte di questi programmi, come ad esempio la shell più diffusa - bash - e il compilatore, sono prodotti da un gruppo di programmatori che si riconoscono nel gruppo della Free Software Foundation (abbreviata comunemente in F.S.F.), fondata nel 1984 da Richard Stallman, il cui scopo è quello di creare un sistema operativo, il più possibile completo, detto GNU. Questo progetto prevede lo sviluppo di tutta una serie di programmi fra cui la shell di comandi (bash), il compilatore (il gcc) e molte altre cose fra cui anche un kernel, che andrebbe a sostituire Linux detto HURD: purtroppo almeno fino ad oggi lo sviluppo di questo kernel non è ancora ad un livello che ne permetta l'uso normale per cui il gruppo della F.S.F. suggerisce l'uso di Linux. Oltre al software la F.S.F. ha sviluppato anche una particolare licenza che è la più diffusa di tutto l'ambiente "Linux", la licenza GNU Public License, detta G.P.L. di cui parlerò più avanti. Però ormai è invalso l'uso di indicare con il termine Linux tutto ciò che normalmente troviamo nei CD-Rom per cui anche questo documento seguirà quest'uso indicando con Linux tutta la famiglia di software che gira con il kernel Linux e con kernel il vero e proprio Linux.

GNU

Linux e il progetto GNU di Richard Stallman

Molti utenti di calcolatore fanno girare ogni giorno una versione modificata del sistema GNU, senza rendersene conto. Per uno strano succedersi di eventi, la versione più diffusa al giorno d'oggi del sistema GNU è più spesso nota con il nome di "Linux", e molti utenti non si rendono conto di quanto questo sia collegato al Progetto GNU.

Esiste un vero Linux, si tratta del nucleo del sistema che queste persone stanno usando. Ma non si può usare il nucleo (kernel) da solo, un kernel è solo utile in quanto parte di un sistema operativo completo. Il sistema operativo al cui interno di solito si trova Linux è una versione modificata del sistema GNU; in altre parole si tratta di un "sistema GNU basato su Linux".

Molti utenti non si rendono completamente conto della differenza che c'è tra il kernel (che sarebbe Linux) e il sistema completo, che viene chiamato Linux anch'esso. L'uso ambiguo del nome, però, non facilita la comprensione della questione.

I programmatori di solito sanno che Linux è il kernel, ma siccome anche loro hanno spesso sentito parlare del sistema completo con il nome di Linux, spesso credono che la storia rispecchi il nome. Per esempio, molti credono che una volta che Linus Torvalds ha finito di scrivere il kernel, i suoi amici si siano guardati intorno cercando altro free software, e quasi per caso tutto quello che serviva per fare un sistema simile a Unix era già disponibile.

Quello che hanno trovato non era lì per caso, si trattava del sistema GNU. Il software libero disponibile costituiva nel suo complesso un sistema completo perché il progetto GNU stava lavorando dal 1984 per creare tale sistema completo. Il manifesto GNU Manifesto aveva portato avanti lo scopo di creare un sistema operativo libero simile a Unix, che si chiamasse GNU. Nel momento in cui venne scritto Linux il sistema GNU era quasi completo.

La maggior parte dei progetti di programmi liberi hanno lo scopo di sviluppare un programma che svolga un particolare lavoro. Per esempio, Linus Torvalds si propose di scrivere un kernel simile a Unix (cioè Linux); Donald Knuth si propose di scrivere un impaginatore di testi (TeX); Bob Scheifler si propose di sviluppare in sistema a finestre (X Windows). È naturale misurare il contributo di questo tipo di progetti in base ai programmi specifici che questi hanno prodotto.

Se cerchiamo di misurare il contributo del progetto GNU in questo modo, cosa concluderemmo? Un distributore di CD-Rom ha misurato che nella sua "Distribuzione Linux", il software GNU costituiva il contributo maggiore, circa il 28% del totale del codice sorgente, includendo alcuni dei componenti più essenziali senza dei quali non ci sarebbe stato un sistema. Linux propriamente detto costituiva solo il 3%. Quindi se si dovesse scegliere un nome per il sistema operativo in base a chi ha scritto i programmi che ne fanno parte, la proposta più appropriata sarebbe "GNU".

Ma noi non crediamo che questo sia il giusto modo di considerare la questione. Il progetto GNU non era, e non è, il progetto di sviluppare specifici pacchetti software. Non era il progetto di sviluppare un compilatore C, anche se lo abbiamo fatto. Non era il progetto di sviluppare un editor di testi, anche se lo abbiamo fatto. Lo scopo del progetto GNU era di sviluppare un sistema completo e libero simile a Unix.

Molte persone hanno dato importanti contributi per software libero e completo, e tutti devono essere ringraziati. Ma il motivo per cui si tratta di un sistema, non solo una collezione di programmi utili, è perché il progetto GNU si è dato da fare perché fosse un sistema. Abbiamo fatto una lista dei programmi che servivano per fare un sistema libero e completo, e abbiamo sistematicamente trovato i programmi, o li abbiamo scritti, o abbiamo trovato persone che scrivessero tutto quello che costituiva la nostra lista. Abbiamo scritto parti essenziali ma non divertenti, come l'assemblatore e il linker, perché non ci può essere un sistema senza questi strumenti. Un sistema ha bisogno di altro oltre agli strumenti di programmazione; l'interprete di comandi "bash", l'interprete Postscript Ghostscript, e la libreria C GNU erano altrettanto importanti.

All'inizio degli anni '90 avevamo messo insieme il sistema completo, con l'eccezione del kernel (ma stavamo lavorando su un kernel, GNU Hurd, che si appoggia su Mach). Lo sviluppo di questo kernel è stato molto più difficile di quello che ci aspettavamo, e stiamo ancora lavorando per finirlo.

Per fortuna non occorre aspettare che sia terminato, perché Linux è già funzionante adesso. Quando Linus Torvalds ha scritto Linux, ha colmato l'ultima grossa lacuna. La gente ha quindi potuto mettere insieme Linux e il sistema GNU per ottenere un sistema libero completo: un sistema GNU basato su Linux (o sistema GNU/Linux, per brevità).

Mettere insieme i due sembra semplice, ma non è stato un lavoro banale. La libreria C GNU (chiamata glibc per brevità) ha dovuto essere modificata considerevolmente. L'integrazione di un sistema completo come una distribuzione che funzionasse "così com'è" è stato un lavoro altrettanto impegnativo. Ha richiesto di confrontarsi con i problemi dell'installazione e dell'avvio del sistema, un problema che non avevamo ancora affrontato perché non eravamo ancora arrivati a quel punto. Le persone che hanno sviluppato le varie distribuzioni del sistema hanno dato un contributo sostanziale.

In aggiunta al progetto GNU, un altro progetto ha prodotto indipendentemente un sistema operativo libero simile a Unix. Questo sistema si chiama BSD ed è stato sviluppato all'università della California di Berkeley. Gli sviluppatori di BSD si sono ispirati all'esempio fornito dal progetto GNU e talvolta sono stati aiutati dagli attivisti di GNU; il loro reale lavoro, comunque, si sovrappone poco con GNU. I sistemi BSD di oggi usano del software GNU, proprio come il sistema GNU e le sue varianti usano del software BSD; ma se presi nel loro complesso i due sono sistemi differenti che si sono evoluti separatamente. Oggi un sistema operativo libero sarà quasi sicuramente o una variante del sistema GNU o un tipo di BSD.

Il progetto GNU supporta i sistemi GNU/Linux come il sistema GNU, anche con finanziamenti. Abbiamo finanziato la riscrittura delle estensioni alla libreria C che sono tipiche di Linux, così che adesso questo sono ben integrate e i sistemi GNU/Linux più recenti usano la versione attuale della libreria senza alcuna modifica. Abbiamo anche finanziato i primi stadi di sviluppo di Debian GNU/Linux.

Per la maggior parte del nostro lavoro oggi usiamo sistemi GNU basati su Linux, e speriamo che li usiate anche voi. Ma per favore non confondete il pubblico usando il termine Linux in modo ambiguo. Linux è il kernel, una delle componenti più importanti del sistema. Ma il sistema nel suo complesso è più o meno il sistema GNU.

Copyright 1997, 1998 Richard Stallman Questa è una traduzione non ufficiale del documento in inglese. La copia letterale e la redistribuzione sono permesse senza pagamento di diritti, purché questa nota sia mantenuta. Ultimo aggiornamento: 18 Marzo 1998, Richard Stallman Traduzione: 28 Luglio 1998, Alessandro Rubini

Linux (il kernel)

Linux è una implementazione gratuita di UNIX per personal computer (386-Pentium PRO), Digital Alpha, PowerPC, Sun SPARC, Apple Macintosh, Atari ST/TT, Amiga, MIPS, che supporta pieno multitasking, multi-user, multi-threading, X Windows, TCP/IP, ottima capacità di comunicazione con altri sistemi operativi (quali quelli della Microsoft, Apple, Novell) e molto altro ancora. Le sue radici sono profondamente radicate in Internet. Le prime versioni del programma sono state sviluppate interamente da Linus Torvalds (torvalds@transmeta.com) presso l'Università di Helsinki in Finlandia, ma ciò che rende Linux così diverso è il fatto di essere cresciuto grazie all'aiuto di moltissimi ed eterogenei gruppi di programmatori UNIX ed esperti di Sistemi Operativi che hanno messo a disposizione il codice da loro prodotto, gratuitamente. Questa eterogeneità si riferisce sia alle competenze tecniche e al settore di provenienza che alla dislocazione geografica. Affinché queste comunità di programmatori potessero lavorare insieme era

necessario un efficiente strumento di comunicazione. Lo strumento fu Internet ed essendo Linux il sistema operativo scelto da queste persone, significa che gli strumenti e le utility necessarie ad utilizzare la Rete furono tra le prime a comparire.

Oltre ad essere sviluppate appositamente per Linux, molte delle applicazioni furono portate da quello che era il miglior software disponibile in quel momento sul mercato. Per esempio, il compilatore C è gcc (della Free Software Foundation): si tratta di un compilatore che viene comunemente utilizzato su piattaforma HP e Sun.

In 11 anni Linux è diventato il sistema operativo più evoluto tecnicamente dell'intera categoria e - a causa della disponibilità dei sorgenti SCO. Inoltre molti degli sviluppatori di Solaris/QNX/AIX/SCO e NT, quando vanno a casa la sera - riscrivono e aggiungono le loro uniche e talvolta proprietarie tecnologie nel codice pubblico di Linux in modo che tutti gli altri possano utilizzarle.

Un po' di Storia

Il 1 Agosto 1991, Linus annunciò nel newsgroup comp.os.minix di aver iniziato a lavorare su una nuova versione di Minix e mise a disposizione i suoi risultati pubblicando i sorgenti della versione 0.01. Questo primo lavoro non conteneva nemmeno gli eseguibili, ma solo alcune parti base del kernel e dava per scontato che si avesse accesso ad una macchina Minix su cui compilarlo.

Il 5 Ottobre 1991 Linus annunciò la prima versione ufficiale: la 0.02. Questa era in grado di far girare bash (la famosa shell UNIX) e il compilatore gcc (GNU).

Da quel momento nuove versioni si susseguirono velocemente (grazie al contributo di molti altri programmatori) e nel Marzo 1992 Linus rilasciò la versione 0.95 ad indicare che il sistema era ormai maturo per la prima release (1.0). In realtà a più di un anno di distanza (nel Dicembre 1993) il kernel di Linux era ancora fermo alla versione 0.99.pl14. Oggi, nel momento in cui vi scrivo è stata rilasciata la versione 2.0.34 e ormai il sistema è così completo e stabile da poter competere e surclassare il nostro "amato" Ms Windows (di cui contiene un emulatore).

(Testo preso dalle FAQ del newsgroup it.comp.linux)

Che cosa rende Linux differente?

Una prima grossa differenza tra Linux e molti degli altri sistemi operativi è il prezzo: è gratuito. Ciò significa che può essere installato, copiato e ridistribuito senza dover pagare nessuna royalty. Non è assolutamente l'unico sistema operativo gratuito al mondo (esiste ad esempio BSD, sopra ricordato anche da Stallman che è uno Unix sviluppato all'Università di Berkley) ma sicuramente è quello più diffuso ed usato.

Comunque Linux ha molti altri vantaggi oltre all'essere gratuito. Il codice sorgente di Linux è a disposizione di chiunque (intendo i sorgenti dell'intero kernel!). Gli ultimi 5 anni nella storia dello sviluppo di Linux hanno dimostrato l'importanza di questa libertà.

Ciò ha comportato ad esempio uno straordinario livello di coinvolgimento di migliaia - addirittura centinaia di migliaia - di persone in tutto il mondo nello sviluppo sia del kernel che di molti altri programmi necessari per il suo utilizzo. Ha consentito ai produttori di hardware di sviluppare i driver per i loro particolari dispositivi senza dover ottenere una costosa licenza per il codice sorgente o firmare un vincolante accordo di non-disclosure (cioè il divieto di far conoscere ad altri il codice sviluppato e le caratteristiche tecniche della periferica in questione).

E ciò ha reso anche possibile agli studenti di informatica (ma non solo) di tutto il mondo di dare un'occhiata dall'interno ad un vero sistema operativo la cui qualità non è seconda a nessun prodotto commerciale. Per le sue caratteristiche Linux è oggi il sistema preferito dai programmatori UNIX e dagli hackers; oltre ai guru anche molte aziende, fra le quali anche molti giganti dell'informatica come IBM, e Internet Providers iniziano

a prenderlo in seria considerazione e a sviluppare potenti server utilizzando proprio questo Sistema Operativo.

Inoltre l'aver a disposizione i sorgenti permette a chiunque di modificare (o di far modificare) il kernel e tutto il resto alle proprie esigenze permettendo di avere sempre software calibrato sulle effettive necessità e di sapere sempre cosa gira sul proprio calcolatore. Quest'ultima caratteristica non è una banalità o una cosa limitata agli hacker come può sembrare ma una caratteristica utile a chiunque e ne faccio un esempio con la questione del numero seriale del Pentium III, saltata da qualche tempo agli onori della stampa di tutti i livelli: da sempre moltissimi modelli di CPU hanno un loro numero seriale unico, cosa che serviva per configurare più comodamente il computer che guidavano in rete e per impedire che un software registrato per una macchina venisse usato anche su altre e questa cosa non ha mai creato alcuna preoccupazione. Da quando è venuta fuori la questione del numero seriale del Pentium III molti si sono preoccupati dicendo che la possibilità di tracciare un processore in rete mette in pericolo la privacy dell'utente non dicendo però che il problema non è nell'hardware ma nel software; usando una cosa tipo Mac o Windows, non posso mai sapere cosa venga fatto effettivamente dal programma che sto usando perché di questo non potrò mai avere il codice sorgente. Da quando siamo abituati a tenere il nostro personal connesso ad una rete globale la pericolosità di questa ignoranza è diventata più evidente indipendentemente dal numero seriale.

Usando invece un programma di cui ho a disposizione i sorgenti posso sempre sapere, o farmi dire da qualcuno che ha le conoscenze tecniche di cui ritengo di potermi fidare, cosa fa effettivamente. Nel caso in esame sappiate che è già uscita una versione di kernel che nasconde il numero seriale del nostro Pentium III, della qual cosa ce ne possiamo sincerare andandoci a leggere i sorgenti.

Inoltre avere a disposizione i sorgenti ha permesso il porting di questo software su moltissime piattaforme differenti come si può leggere nelle caratteristiche tecniche più giù, permettendo di avere i servizi di rete insieme a molte altre possibilità anche su computer i cui sistemi operativi originali non le prevedevano, come ad esempio Amiga. In un recente numero del Linux Journal c'è un'intervista a Linus in cui lui afferma di essersi "goduto" il porting di Linux per PalmPilot: provate ad immaginare un'affermazione del genere in bocca alla Microsoft, all'IBM o qualche altra ditta.

Un'ultima caratteristica del nostro software è che gode della migliore assistenza di tutti i prodotti commerciali a cui siamo abituati. Basta avere una connessione ad Internet e chiedere aiuto in una delle moltissime Mailing-list, newsgroup o siti dedicati a Linux e avrete la risposta che cercate: la diffusione dei sorgenti permette una conoscenza del funzionamento del software che usiamo più capillare rispetto a software proprietario. Anche qui farò un esempio: tempo fa si scoprì che una banalissima operazione fatta sul protocollo in uso su Internet poteva bloccare irrimediabilmente i server basati su qualunque sistema operativo, Linux compreso, questione detta del "*Ping of Death*". Questo però fu il primo per il quale fu disponibile la correzione grazie al lavoro degli sviluppatori volontari e il controllo degli utilizzatori più preparati: in effetti passarono pochi giorni fra la diffusione della notizia e la disponibilità di questa correzione, mentre per altri sistemi operativi si dovette aspettare, anche alcuni mesi, che la casa proprietaria facesse uscire la correzione ufficiale. Proprio in occasione di questa prova di superiorità una nota rivista informatica premiò Linux come il sistema operativo dotato del miglior supporto di assistenza di tutto il mondo informatico.

Caratteristiche tecniche di un sistema Linux

Da un punto di vista molto semplificato Linux è solo una versione di Unix, del quale mantiene molte caratteristiche, pregi e difetti. La differenza però con altri Unix è che Linux è in rapidissimo sviluppo ed è supportato da moltissime persone.

Queste comunque sono, sinteticamente e un po' tecnicisticamente le sue caratteristiche principali:

- multitasking: può eseguire più programmi nello stesso momento.
- multi-user: supporta più utenti sulla stessa macchina (senza richiedere licenze aggiuntive!).
- multiplatform: è in grado di girare su differenti piattaforme (non solo INTEL ma anche Digital Alpha, PowerPC, Sun SPARC, Macintosh e MIPS).
- gira in modalità protetta su processori 386 o superiori.
- prevede funzioni di protezione della memoria tra processi in maniera tale da non consentire ad un solo programma di mandare in crash l'intero sistema.

- shared copy-on-write pages among executables. Ciò significa che processi multipli possono utilizzare la stessa memoria per girare. Quando uno di questi cerca di scrivere nella parte di memoria condivisa, quella pagina (4KB di memoria) viene copiata da qualche altra parte. Copy-on-write ha due vantaggi: maggiori prestazioni e riduzione del consumo di memoria.
- gestione della memoria virtuale attraverso la paginazione (senza cioè dover scaricare l'intero processo) sul disco: su una partizione separata, su un file all'interno del file system, o entrambi, con la possibilità di aggiungere al volo - qualora ce ne fosse bisogno - ulteriori aree di swap (senza dover riavviare la macchina o interrompere il processo). Un totale di 16 aree di swap da 128MB possono essere utilizzate contemporaneamente per un teorico ammontare di 2 GB di spazio di swap utilizzabile.
- un pool di memoria unificato per i programmi e la memoria cache, in maniera tale che tutta la memoria libera possa essere utilizzata come cache, e tuttavia la cache possa essere ridotta qualora ci sia bisogno di lanciare un grosso programma.
- possiede librerie statiche e dinamiche (Dynamic Link Libraries - DLL).
- esegue il core dump consentendo un'analisi dopo il crash. In questo modo è possibile utilizzare un debugger non solo quando un programma sta girando, ma anche dopo che è andato in crash compatibile con POSIX, System V e BSD a livello di sorgente compatibile con SCO, SVR3, SVR4 a livello di eseguibili grazie all'adozione di un emulatore iBSC2-compatibile.
- completa disponibilità dei sorgenti, compreso l'intero kernel e tutti i driver, gli strumenti di sviluppo e tutti i programmi utente.
- inoltre è tutto liberamente distribuibile.
- è anno-2000 compatibile.
- POSIX job control.
- dispone di pseudoterminali (pty's).
- emulatore 387 a livello di kernel in maniera tale che nessun programma abbia bisogno del proprio emulatore... Ogni computer che utilizza Linux appare come se possedesse un coprocessore matematico. Ovviamente se il vostro computer possiede già una FPU, è possibile rimuovere l'emulatore dal kernel guadagnando alcuni Kb di memoria.
- supporto per diverse tastiere internazionali (tra cui quella Italiana) ed è possibile caricare e scaricare i driver al volo.
- consolle virtuali multiple: è possibile effettuare diversi login indipendenti dalla stessa postazione e passare da una all'altra con una semplice combinazione di tasti (spesso ALT-F1 - ALT-F2...). È possibile avere fino a 64 sessioni parallele, contemporanee e indipendenti.
- supporto nativo per molteplici file system: minix-1, Xenix System V, MS-DOS, HPFS (OS/2 2.x), VFAT (Windows 95), NT, HFS (Apple Mac), FFS (Amiga), CD-Rom, NFS...
- file system nativo che supporta partizioni fino a 4TB e nomi lunghi fino a 255 caratteri.
- networking TCP/IP nativo (comprendente ftp, telnet, NFS, etc.).
- può agire da server per reti AppleTalk.
- può agire da server per reti Microsoft (emulando LanManager, NT...) e da client (WfWg, W95, NT).
- può agire da client o server in una rete Novell Netware.
- può lavorare in clustering, cioè più macchine possono lavorare insieme su uno stesso programma contemporaneamente.
- e molto altro ancora...

Cosa si può fare con Linux

Come ho detto più volte Linux è nato in rete e grazie alla rete e questo comporta che per quanto riguarda le applicazioni di rete siamo copertissimi: sia dal lato server che dal lato client potete trovare praticamente ogni cosa in molte varianti. Con Linux potrete leggere la posta elettronica e i newsgroups, fare ftp, collegarvi ad altre macchine in rete sia in modalità testo che grafica con decine di programmi differenti.

Per quanto riguarda la trattazione di testi anche qui siamo messi benino in quanto abbiamo un potentissimo motore che si chiama TeX (si legge tech), il quale permette la produzione di documenti di stampa molto sofisticati ed eleganti. Originariamente questo era molto complesso da usare in quanto si dovevano scrivere tutti i campi a mano, in modo che assomiglia molto alla programmazione (d'altronde è stato sviluppato da programmatori), ma ormai esistono molti programmi che producono testo pronto per essere usato da TeX in modo molto più semplice e "visuale".

Oltre a questo ci sono molti programmi che lo coadiuvano, come ispell per la correzione ortografica (il quale viene usato anche da molti altri programmi in modo da avere un solo dizionario personalizzato), ghostview e dvi per la visualizzazione delle anteprime di stampa e per la stampa stessa.

Ultimamente la diffusione di Linux anche nelle case sta portando allo sviluppo di applicazioni che si trovano normalmente nei personal casalinghi, come programmi di scrittura più simili a quelli a cui siamo abituati su piattaforme di tipo Windows o Mac, di grafica (ottimo in questo campo è gimp), fogli elettronici (da vedere siag) e altro, e questo grazie sia allo sviluppo libero sia all'interesse di grandi case software come la Corel che ha rilasciato un ottimo programma di scrittura. Un problema molto grave in questo campo sono i giochetti: nonostante tutto siamo ancora molto indietro rispetto agli ambienti Windows e Mac e questo perché lo sviluppo di un gioco del livello di Quake (che comunque esiste anche per Linux ma è praticamente l'unico giochetto di questo livello esistente) è estremamente complesso e ancora le case produttrici di giochi non hanno dimostrato molto interesse.

Difetti di Linux

Non tutte sono rose e fiori però. In particolare Linux è:

- difficile da usare;
- vecchio come impostazione;
- nato per essere usato in situazioni diverse dall'uso domestico.

Queste caratteristiche negative sono fra loro legate e provengono tutte dall'origine di Linux come variante Unix. Questo comporta ad esempio una difficile gestione degli utenti e gruppi di uno stesso computer. La comodità di avere più utenti sullo stesso computer (perché di questo si tratta anche se all'inizio ci se ne può non rendere conto) è difficile da usare e da capire: vi troverete davanti ad un sistema che, se usato correttamente, non vi permetterà di fare alcune cose un po' delicate. Praticamente si perde un po' di elasticità d'uso in favore di un sistema più sicuro, stabile e protetto dagli errori degli utilizzatori.

Infine essendo un sistema relativamente nuovo manca ancora di molto software, cosa che però dovrebbe cambiare via via che questo prenderà maggior campo e che sta in effetti cambiando.

Questioni di licenze

La situazione delle licenze nel mondo Linux si è andata complicando via via che il sistema si diffondeva e acquisiva stabilità, usabilità e soprattutto notorietà, cose dalle quali è derivato interesse commerciale. Comunque la gran parte del software per questo sistema operativo rientra nel mondo dell'OpenSource, software di cui si hanno sempre a disposizione i sorgenti del programma stesso e che si possono modificare e ridistribuire. Allego a questa lezione la G.P.L. tradotta in italiano a cura del PLUTO. La traduzione non è ufficiale, nel senso che non è riconosciuta ufficialmente dalla Free Software Foundation (cosa per altro molto complicata) ma dà l'idea.

GPL

La GNU Public License è forse la più diffusa perché riguarda la maggior parte del nostro software, in particolare il kernel, il Perl e tutto il software comunemente detto GNU. È stata creata da Richard Stallman come è scritto in ciò che dice lui nel documento che ho presentato prima. In pratica vincola la distribuzione dei programmi che copre al fatto che insieme agli eseguibili siano sempre distribuiti o reperibili anche i sorgenti, e che tutto ciò che viene sviluppato usando parti di programmi coperti da G.P.L. usi lo stesso tipo di licenza

BSD

Tutti i sorgenti derivati da questi devono mantenere le note originali dell'originale. Software: BSD, Tcl/Tk, sendmail

Licenza Mit

Tutti i sorgenti derivati da questi devono mantenere le note di copyright degli originali. Software: X.

Altre

Oltre a queste ci sono molti altri tipi di licenza, da quelle proprietarie, il "normale" copyright, lo shareware, cardware e altri.

Case sensitiveness

I sistemi operativi Unix, come GNU/Linux, sono sensibili alla differenza tra le lettere maiuscole e minuscole. La differenza è sostanziale, per cui gli ipotetici file denominati: Ciao, clao, CIAO, ecc. sono tutti diversi. Non bisogna confondere questa caratteristica con quello che può succedere in altri ambienti, come per esempio MS-Windows 95/98/NT/2000, che preservano l'indicazione delle lettere maiuscole o minuscole, ma che poi non fanno differenza quando si vuole fare riferimento a quei file.

Il signor root, la directory root

Negli ambienti Unix si fa spesso riferimento al termine **root** in vari contesti e con significati differenti. *Root* è la radice, o l'origine, senza altri significati. A seconda del contesto, ne rappresenta l'origine, o il punto iniziale. Per esempio, si può avere:

- una directory *root*, che è la directory principale di un file system, ovvero la directory radice;
- un file system *root*, che è il file system principale di un gruppo che si unisce insieme;
- un utente *root*, che è l'amministratore;
- un dominio *root*, che è il dominio principale;
- una finestra *root* che è quella principale, ovvero la superficie grafica (*desktop*) su cui si appoggiano le altre finestre del sistema grafico X.

Le situazioni in cui si presenta questa definizione possono essere molte di più. L'importante, per ora, è avere chiara l'estensione del significato di questa parola.

GNU/Linux, come gli altri sistemi derivati da Unix, è multiutente. La multiutenza implica una distinzione tra i vari utenti. Fondamentalmente si distingue tra l'amministratore del sistema, o *superuser*, e gli altri utenti. L'amministratore del sistema è quell'utente che può fare tutto ciò che vuole, soprattutto rischia di produrre gravi danni anche solo per piccole disattenzioni. L'utente comune è quello che utilizza il sistema senza pretendere di organizzarlo e non gli è possibile avviare programmi o accedere a dati che non lo riguardano.

Il sistema operativo GNU/Linux, così come tutti i sistemi operativi Unix, è composto essenzialmente da:

- un sistema di avvio o *boot*;
- un kernel;
- un file system;
- un sistema di inizializzazione e gestione dei processi in esecuzione;
- un sistema di gestione della rete;
- un sistema di gestione delle stampe;
- un sistema di registrazione e controllo degli accessi;
- una shell (interprete dei comandi);
- alcuni programmi di servizio (*utility*) per la gestione del sistema;
- strumenti di sviluppo software (C/C++).

Il boot

E' il modo con cui un sistema operativo può essere avviato quando l'elaboratore viene acceso. Di solito, il software registrato su ROM degli elaboratori basati sull'uso di dischi, è fatto in modo da eseguire le istruzioni contenute nel primo settore di un dischetto, oppure, in sua mancanza, del cosiddetto MBR (*Master boot record*) che è il primo settore del primo disco fisso. Il codice contenuto nel settore di avvio di un dischetto o del disco fisso, provvede all'esecuzione del kernel (lo avvia). Con GNU/Linux installato in un elaboratore i386, la configurazione e la gestione del sistema di avvio viene fatta principalmente attraverso tre modi possibili:

- LILO, che è in grado di predisporre un settore di avvio su un dischetto, sull'MBR o sul primo settore della partizione contenente GNU/Linux;
- GRUB, che è funzionalmente simile a LILO;
- Loadlin, che permette di avviare l'esecuzione di un kernel Linux da una sessione Dos.

Il kernel

Come suggerisce il nome, è il nocciolo del sistema operativo. I programmi utilizzano il kernel per le loro attività e in questa maniera sono sollevati dall'agire direttamente con la CPU. Di solito, è costituito da un file unico, il cui nome potrebbe essere vmlinuz (oppure zImage, bzImage e altri), ma può comprendere anche moduli aggiuntivi, per la gestione di componenti hardware specifici che devono poter essere attivati e disattivati durante il funzionamento del sistema. Quando il kernel viene avviato (attraverso il sistema di avvio), esegue una serie di controlli diagnostici in base ai tipi di dispositivi (componenti hardware) per il quale è stato predisposto, quindi monta (*mount*) il file system principale (*root*) e infine avvia la procedura di inizializzazione del sistema (Init).

Il file system

Il file system è il modo con cui sono organizzati i dati all'interno di un disco o di una sua partizione. Nei sistemi operativi Unix non esiste la possibilità di distinguere tra un'unità di memorizzazione e un'altra, come avviene nel Dos, in cui ogni disco o partizione sono contrassegnati da una lettera dell'alfabeto (A:, B:, C:). Nei sistemi Unix, tutti i file system cui si vuole poter accedere devono essere concatenati assieme, in modo da formare un solo file system globale. Quando un sistema Unix viene avviato, si attiva il file system principale, o *root*, quindi possono essere collegati a questo altri file system a partire da una directory o sottodirectory di quella principale. Dal momento che per accedere ai dati di un file system diverso da quello principale occorre che questo sia collegato, nello stesso modo, per poter rimuovere l'unità di memorizzazione contenente questo file system, occorre interrompere il collegamento. Ciò significa che, nei sistemi Unix, non si può inserire un dischetto, accedervi immediatamente e toglierlo quando si vuole: occorre dire al sistema di collegare il file system del dischetto, quindi lo si può usare come parte dell'unico file system globale. Al termine si deve interrompere questo collegamento e solo allora si può rimuovere il dischetto.

(Abbiamo estratto alcune pagine da “Appunti di Informatica Libera” – scaricabile all'indirizzo <http://a2.swlibero.org/>)