Come funziona Linux:
l'amministrazione di sistema

In questa puntata introdurremo il problema dell'amministrazione di un sistema Linux. Anche se si utilizza da soli un computer, è comunque buona norma distinguere nettamente il ruolo di utente da quello di amministratore, in quanto tale distinzione consente di avere sempre un sistema sicuro ed affidabile.

Settima parte

di Giuseppe Zanetti

Abbiamo già visto che in Linux non tutti gli utenti possono fare le stesse cose. Il meccanismo dei permessi infatti permette di creare un ambiente sicuro, in cui solamente l'utente amministratore di sistema (root) o le persone a cui egli ne concede la possibilità possono modificare i file importanti o di configurazione.

Oltre al vantaggio di poter lasciare tranquillamente utilizzare la macchina anche ad utenti inesperti senza che possano fare danni, il meccanismo delle protezioni mette al sicuro anche da eventuali virus e cavalli di Troia. Infatti, salvo che si debba modificare un'impostazione, durante il lavoro normale le operazioni vengono compiute collegandosi a Linux come un utente non privilegiato. Eventuali operazioni errate o programmi maliziosi potranno così al massimo intaccare i file per i quali il singolo utente ha i permessi di scrittura e non quelli di sistema.

Collegarsi come root anche durante il lavoro normale può sembrare comodo, ma è probabilmente la più importante fonte di problemi. Anche per l'utente esperto, infatti, sbagliare è facile e non sempre è possibile tornare sui propri passi...

Single user mode

I sistemi UNIX, da cui Linux concettualmente deriva, non furono inizialmente pensati per funzionare su personal computer ed è perciò normale aspettarsi che una semplice operazione di manutenzione o installazione di software non abbia influenza sulla continuità d'utilizzo del sistema. Sarebbe infatti assurdo, in un sistema multiutente, fermare il lavoro di centinaia di persone solamente per cambiare l'indirizzo di un DNS o per installare un nuovo modem.

Nonostante ciò, capita di voler eseguire alcune operazioni importanti senza il rischio che altri utenti accedano contemporaneamente alla macchina. Ad esempio, la riparazione di un filesystem danneggiato mediante fsck dovrebbe essere fatta essendo sicuri che nessun altro processo vada a modificare il disco mentre ne stiamo testando l'integrità.

Abbiamo visto parlando dei runlevel che per far ciò esiste un modo semplice e pulito, ovvero quello di portare il sistema nello stato di "single user". In tale modalità di funzionamento l'accesso è permesso solamente dalla console di sistema all'utente root e sono disabilitati tutti i servizi. La mancanza dei servizi a cui si è abituati durante il normale lavoro rende, però, la modalità di single user abbastanza limitante ed è perciò conveniente ricorrervi solamente quando è strettamente necessario. La maggior parte delle procedure di amministrazione di un sistema Linux può tranquillamente essere eseguita a sistema funzionante e senza necessità di reboot.

Per entrate in single user mode si può fare un reboot del sistema e inserire il runlevel desiderato nella linea di comando di LILO di seguito al nome dell'immagine del kernel che si vuole caricare:

```
LILO: linux S
```

oppure è possibile in qualunque momento passare dal runlevel corrente a quello di single user senza spegnere la macchina mediante il comando init:

```
# init S
INIT: Going single user
INIT: Sending processes the KILL signal
INIT: Sending processes the TERM signal
Stopping pcmcia                          [  OK  ]
Stopping irda                            [  OK  ]
Stopping linuxconf                       [  OK  ]
```

In questo modo, sotto il controllo del programma init e del suo file di configurazione /etc/inittab, verranno terminati tutti i servizi avviati nel runlevel corrente (poniamo sia il 3): ognuno degli script Sxxxxx presente in /etc/rc.d/rc3.d viene pertanto richiamato col parametro "stop", così come quelli di tipo Kxxxxx in /etc/rc.d/rc1.d. In seguito vengono fatti partire i servizi necessari per il single user, ovvero in

/etc/rc.d/rc1.d/ vengono lanciati col parametro "start", gli script che iniziano con S.

Per tornare in modalità multiutente al termine delle operazioni si potrà semplicemente dare il comando "inverso": init 3.

Il disco di rescue

Alle volte, nel caso di problemi gravi, ad esempio se il disco dove risiede il filesystem principale è seriamente danneggiato, non è possibile neppure fare il boot del sistema in modalità single user. Per fronteggiare situazioni di questo tipo è bene tenere sempre sotto mano un dischetto di rescue (salvataggio), che contenga al proprio interno un mini sistema Linux bootabile da floppy, un editor di testi (di solito "vi") e le principali utility di manutenzione del filesystem (mount, fsck, ...). Tale strumento è incluso in quasi tutte le distribuzioni di Linux, ma, nel caso non lo fosse, è possibile trovare in rete parecchie distribuzioni di Linux su un singolo floppy che possono essere utilizzate allo scopo. Ne potete trovare una lista completa su http://freshmeat.net/appindex/console/mini%20distributions.html.

E' addirittura possibile creare il proprio disco di rescue personalizzato utilizzando l'utility BYLD (http://byld.sourceforge.net/).

Il metodo di utilizzare questi strumenti è quello di fare il boot da floppy e poi di riparare manualmente il filesystem rovinato usando fsck:

```
# fsck -t ext2 -r /dev/hda1
```

Nel caso il problema fosse diverso e il sistema non partisse, ad esempio perché si è persa la password di root, è possibile usare il dischetto di rescue per montare il filesystem principale e andare a modificare direttamente il file delle password (/etc/passwd oppure /etc/shadow) usando un editor:

```
# mount -t ext2 /dev/hda1 /mnt
# vi /mnt/etc/passwd
```

La soluzione è quella di togliere completamente la password all'utente root:

```
root:x:0:0:root:/root:/bin/bash
```

Fatto ciò, sarà possibile fare un reboot del sistema ed entrare come root premendo semplicemente il tasto return alla richiesta della password. A questo punto si assegna una nuova password a root e il gioco è fatto:

```
freddy login: root
Password:
Last login: Tue Jun 20 16:43:48 on tty1
You have new mail.
# passwd root
Changing password for user root
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully
```

Ovviamente tale possibilità può essere pericolosa dal punto di vista della sicurezza.

Conviene perciò disabilitare il boot da floppy disk e proteggere con una password l'accesso al BIOS della macchina.

I compiti dell'amministratore di sistema

Vedremo in questa e nelle prossime puntate del corso, che le operazioni di manutenzione di un sistema Linux sono essenzialmente le seguenti:

- ✔ installazione di una nuova periferica hardware
- ✔ installazione e configurazione di nuovo software o servizio
- ✔ analisi dello stato del sistema e dei log, gestione della sicurezza
- ✔ manutenzione ordinaria (backup, pulizia dei dischi, gestione utenti, ...)
- ✔ manutenzione straordinaria (recovery di incidenti, ...).

Tenere un diario delle modifiche

Durante tutte queste operazioni, specialmente nel caso si modifichino dei file di configurazione, conviene tenere traccia delle modifiche apportate, in modo da poter tornare indietro nel caso di malfunzionamenti o effetti collaterali indesiderati. Ciò permetterà anche eventualmente di costruire molto velocemente un sistema simile al nostro o di sapere quali configurazioni si devono fare nel caso di reinstallazione. Il metodo migliore è quello di segnare le modifiche su un notes oppure su un file, che tuttavia potrebbe andare perso o non essere disponibile in caso di problemi. E' anche utilissimo fare una copia dei file prima di modificarli:

```
# cp /etc/sendmail.cf /etc/sendmail.cf.10.05.2000
# vi /etc/sendmail.cf
```

Generalmente i file di configurazione di Linux si trovano nella directory /etc (oppure /usr/etc o /usr/local/etc), tuttavia alcuni programmi hanno i propri file di configurazioni in altre posizioni nel filesystem.

Il metodo classico di configurazione di un sistema di tipo UNIX è quello di modificare direttamente i file di sistema utilizzando un comune editor di testi. L'operazione è facilitata dal fatto che praticamente tutti i file di configurazione sono file di testo. Questo formato ha i vantaggi di essere più comprensibile per chi deve modificare una configurazione rispetto ad un file binario ed inoltre di essere meno propensi a problemi durante la trasmissione, ad esempio via e-mail.

All'interno dei file che si modificano è bene inserire sempre dei commenti su cosa si è fatto, completi di data e del nome della persona che ha apportato la variazione. Il formato accettato dalla maggior parte dei programmi permette l'inserimento di commenti facendoli precedere dal carattere cancelletto "#":

```
# Abilita la access list (anti-SPAM)
# modificato da Giuseppe Zanetti (beppe@profuso.com) il 16/06/2000
Kaccess hash -o /etc/mail/access
```

Non si tratta, purtroppo, di una regola. In alcuni casi i commenti devono essere fatti precedere da altri caratteri, ad esempio il punto e virgola ";" o la scritta "rem".

Strumenti semplificati di configurazione

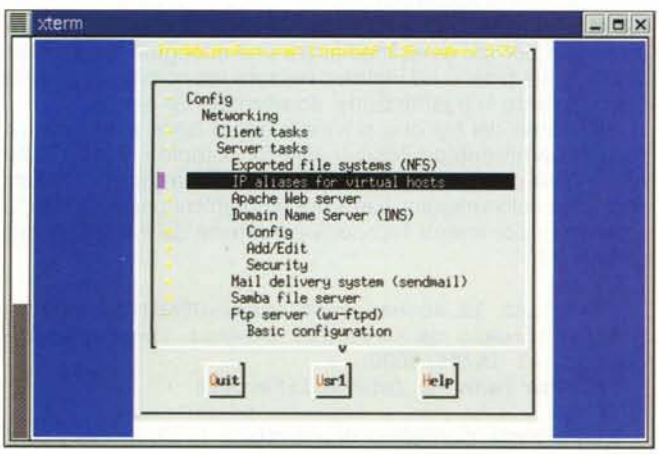
Configurare a mano un programma è spesso un'operazione ostica. Alcuni anni or sono, era considerato un "vero System Manager" solo chi riusciva ad ottenere una configurazione funzionante di sendmail (il programma che gestisce il protocollo SMTP della posta elettronica) modificando direttamente a mano il file /etc/sendmail.cf. Anche se la modifica manuale dei file di configurazione rimane il metodo più usato e più utile per comprendere come funzionano i diversi programmi, per fortuna ora sono disponibili diverse interfacce che permettono di configurare un computer Linux in modo semplice ed intuitivo.

Bisogna fare però attenzione al fatto che spesso questi strumenti non coesistono in modo accettabile con eventuali modifiche fatte a mano o usando altri programmi, con la possibilità che si sovrascrivano a vicenda le configurazioni.

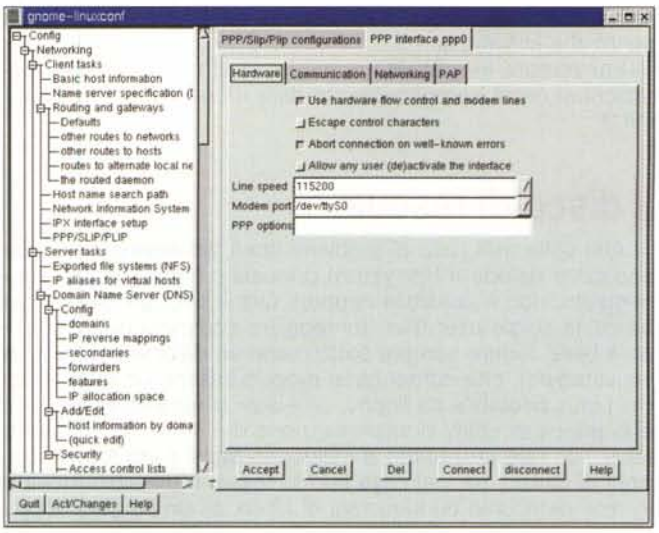
Ogni distribuzione di Linux offre i propri strumenti, ma esistono delle soluzioni indipendenti dalla versione utilizzata. In questo articolo tratteremo linuxconf, ma vale la pena menzionare anche webadmin (http://www.webmin.com/webmin), che ha la caratteristica di funzionare con una interfaccia Web e di essere disponibile per più sistemi (diverse distribuzioni di Linux, Solaris, HP-UX, FreeBSD, ...). Esso permette non solo di configurare, in modo coerente fra le diverse piattaforme, gli aspetti standard del sistema operativo (utenti e gruppi, stampanti, servizi di rete, condivisione di file, ...) ma anche di creare mailing list, gestire database MySQL, tenere sincronizzate le versioni dei programmi installati, ...

Nella tabella che segue sono riassunte le funzionalit‡ principali configurabili mediante Linuxconf.

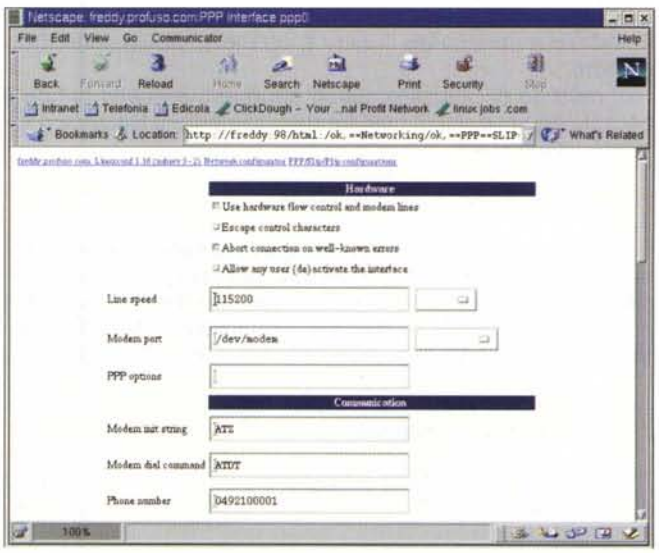
System time, time zone, CMOS clock
LILO
Basic networking
IPX interface setup
Static routing
Filesystems (/etc/fstab)
Routed daemon
NIS client (ypbind)
NFS server
PPP client
User accounts, groups
Shadow account policies
DNS named daemon (bind)
Sendmail, virtual email domain
Fire-walling (Packet filtering)
RARP
DHCP
IP Alias
UUCP
Disk quota



L'albero delle configurazioni possibili usando Linuxconf.



...sotto Gnome...



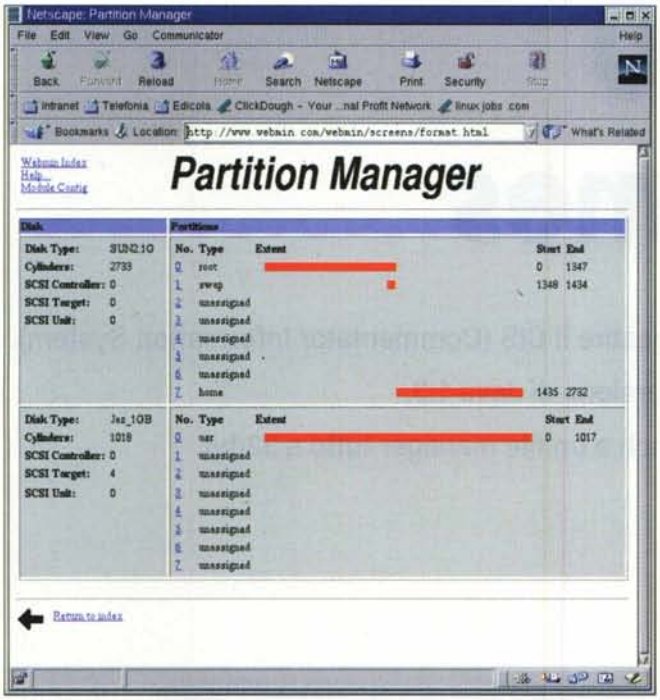
...e via web.

Linuxconf

Linuxconf (http://www.solucorp.qc.ca/linuxconf/) è un programma scritto specificamente per Linux, che ha la caratteristica di essere realizzato in maniera "modulare" ed indipendente dall'interfaccia che si sta usando (solo testo, grafica in ambiente X, via Web).

Una caratteristica interessante di Linuxconf è quella di poter creare diverse configurazioni per la stessa macchina. Tale funzione è utilissima nel caso di un computer portatile, che può essere configurato in modo diverso a casa o in ufficio.

La programmazione di plugin aggiuntivi è relativamente

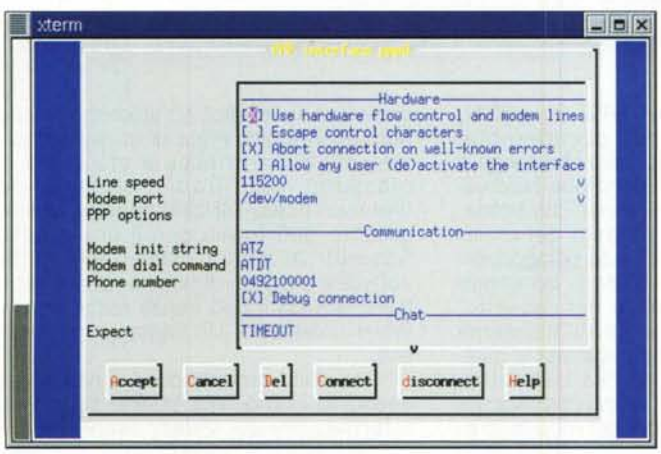


La gestione delle partizioni usando webadmin

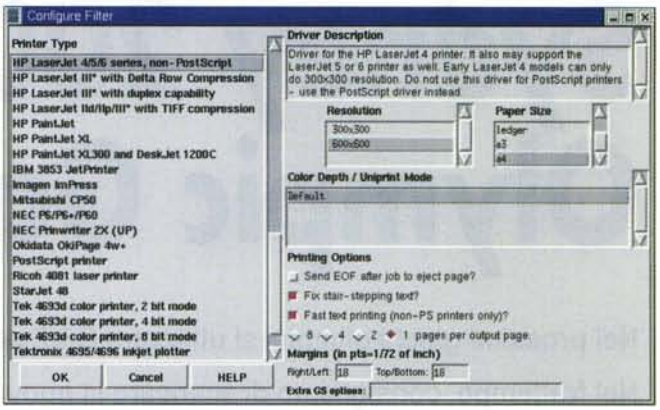
semplice ed indipendente dall'interfaccia che verrà utilizzata. Come si può vedere dalle foto allegate a questo articolo, le stesse maschere di configurazione possono essere usate indipendentemente sia in modalità testo che via Web o in ambiente X, in quanto non viene ogni volta riscritta l'interfaccia utente ma viene realizzata solamente una descrizione della stessa basata su un insieme di componenti standard (bottoni a scelta singola o multipla, caselle di inserimento testo, ...).

In questo modo è possibile anche demandare la creazione delle maschere di configurazione agli autori dei diversi programmi e fare in modo che ogni nuovo pacchetto software che si installa sulla macchina si porti dietro il proprio modulino di configurazione.

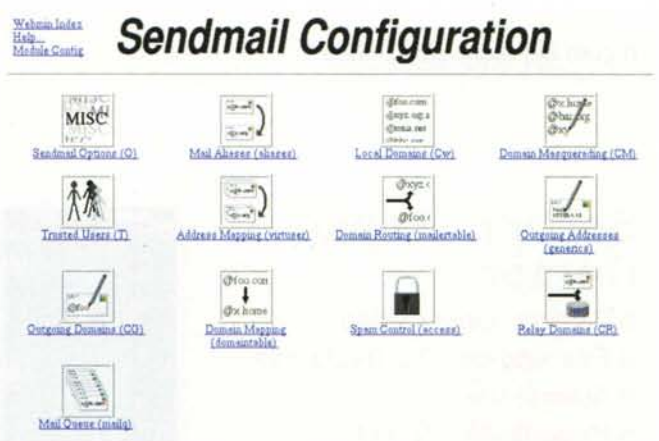
La possibilità di usare interfacce diverse permette di utiliz-



La stessa maschera di configurazione usando Linuxconf in modalit‡ testo...



Lo strumento per aggiungere una stampante "proprietario" di Red Hat.



Webadmin permette di gestire tutte le possibili opzioni di sendmail.

zare Linuxconf per fare la manutenzione anche di macchine remote.

Oltre ad essere un configuratore, Linuxconf è anche un "attivatore" di servizi, ed è in grado di sostituire molte delle funzionalità di init. Esso è in grado di sincronizzare il sistema in modo che siano attivi i servizi specificati dalla configurazione prescelta. Ciò è possibile lanciando il programma con la seguente linea di comando:

```
linuxconf –update
```

Oppure utilizzando l'opzione "Activate changes" dal menu interattivo.

E' inoltre possibile vedere la lista delle operazioni ancora da compiere per portare il sistema dalla condizione attuale a quella configurata, usando l'opzione —status.

Conclusioni

Grazie a Linux, UNIX ha perso in parte la sua fama di sistema eccessivamente complesso da configurare e mantenere. Nelle prossime puntate vedremo come in realtà le diverse attività da compiere come amministratore di sistema risultino in Linux semplificate e facilmente automatizzabili. MC