



Introduzione a GNU/Linux

Indice

- [Introduzione](#)
- [Descrizione generale](#)
 - ◆ [Le distribuzioni](#)
 - ◆ [Il sistema operativo](#)
 - ◆ [Accedere al sistema](#)
 - ◆ [La shell](#)
 - ◆ [La home directory](#)
- [File e directory](#)
 - ◆ [Il filesystem](#)
 - ◆ [I permessi](#)
 - ◆ [Trovare i file \(find\)](#)
 - ◆ [Archiviare i file \(tar\)](#)
 - ◆ [Il tipo dei file \(file\)](#)
 - ◆ [Trovare una parola nei file \(grep\)](#)
 - ◆ [Confrontare due file \(diff\)](#)
- [Gli utenti](#)
- [I processi \(ps\)](#)
- [Informazioni sui comandi \(man\)](#)
- [Input e output dei programmi](#)
- [Le variabili d'ambiente](#)
- [Un correttore ortografico \(ispell\)](#)

Data ultima modifica: 10/07/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux

[[Indice generale](#)]

Introduzione

GNU/Linux è un sistema operativo che mette a disposizione degli utenti di personal computer la potenza dei sistemi Unix; quello che comunemente si indica con Linux è l'unione del *kernel*, il cui nome è proprio Linux, e delle librerie e programmi di sistema sviluppati originariamente dal progetto GNU (<http://www.gnu.org/>) che, cronologicamente, precedono la nascita del kernel stesso.

Il kernel è stato sviluppato originariamente da Linus Torvalds, che oggi detiene il copyright sul nome "Linux"; Torvalds continua a coordinare l'attività degli sviluppatori del kernel, che è rilasciato con licenza [GNU Public License](#) (GPL). I tool del progetto GNU, che comprendono fra gli altri il compilatore C `gcc`, insieme con il kernel, formano il sistema operativo GNU/Linux; fatte queste precisazioni, nel seguito verrà per comodità usato il termine Linux per indicare il sistema operativo GNU/Linux. E' da notare che il kernel Linux non è l'unico kernel per i sistemi operativi basati sui tools GNU, essendo attivo il progetto [Hurd](#) (a questo proposito si veda anche il progetto [Debian GNU/Hurd](#)).

La comunità Linux ha recentemente visto l'ingresso di colossi dell'informatica quali [IBM](#), che stanno contribuendo a legittimare il successo di un prodotto che è il risultato di un modello di sviluppo aperto che ha dato frutti forse insperati, avendo ad esempio portato il server web [Apache](#), il cui sorgente è disponibile, in testa alla classifica dei server più diffusi (si veda <http://www.netcraft.com>).

Per avere un'introduzione al mondo del software libero, insieme con una descrizione del modello commerciale che esso propone, un sito di riferimento è <http://www.softwarelibero.it>; è utile fare notare come il modello di sviluppo che ha fatto nascere Linux non è limitato solamente a questo sistema operativo; lo stesso Apache e molti altri prodotti sono disponibili con la stessa licenza anche per i sistemi operativi Microsoft.

L'utilizzo di Linux nell'ambiente casalingo è ancora minoritario, nonostante gli sforzi della comunità degli sviluppatori che hanno ormai portato alla maturità ambienti grafici quali [KDE](#) o [Gnome](#), che per quantità di applicazioni e facilità d'uso hanno poco da invidiare a Windows. Linux troverà vasto campo di applicazione anche quale ambiente desktop quando fra gli utenti si farà strada la convinzione che la qualità di Linux non è inferiore a quella di Windows, e acquisterà importanza la consapevolezza che la conoscenza (anche solo potenziale) delle cose che ormai fanno parte della nostra vita quotidiana è una garanzia di libertà.

Un sito in lingua italiana dove trovare informazioni su Linux è "Appunti di Informatica Libera" (<http://a2.swlibero.org/>) (autore Daniele Giacomini)

E' possibile reperire un manuale in lingua italiana, distribuito con licenza Free Documentation License all'indirizzo <http://www.linuxfacile.org/> (autore Daniele Medri).

Un libro sulla programmazione in Linux, distribuito con licenza FDL, è accessibile all'indirizzo <http://www.firenze.linux.it/~piccardi/gapil/> (autore Simone Piccardi)

Data ultima modifica: 01/07/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della

GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux – Descrizione generale

[[Indice generale](#) | [Indice sezione](#) | << |
>>]

Indice

- [Le distribuzioni](#)
- [Il sistema operativo](#)
- [Accedere al sistema](#)
- [La shell](#)
- [La home directory](#)

Data ultima modifica: 17/06/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux – Descrizione generale

[[Indice generale](#) | [Indice sezione](#) | << | >>]

Le distribuzioni

Molte ditte di software hanno assemblato il kernel, i tools GNU e i programmi applicativi disponibili per il sistema operativo Linux creando le proprie distribuzioni; fra le molte esistenti, ricordiamo la [Slackware](#), la prima a raggiungere un buon livello di diffusione, la [Debian](#), la distribuzione che meglio incarna lo spirito del software libero, [Red Hat](#) e [SuSe](#), le distribuzioni che oggi vedono il maggior successo commerciale.

Le varie distribuzioni differiscono fra di loro per le versioni e la quantità dei programmi installati, per i vari tool che assistono l'utente e l'amministratore nella manutenzione del sistema e per alcune differenze nella configurazione del sistema operativo.

Non è praticamente possibile fare una classifica della qualità delle maggiori distribuzioni, in quanto ognuna di essa si focalizza su un particolare tipo di utilizzo, ma non esclude comunque gli altri. Una distribuzione che conosce un buon grado di popolarità è la [Mandrake](#), che aggiunge alla compatibilità con Red Hat alcuni tool che agevolano la configurazione del sistema all'utente meno esperto.

Praticamente tutte le ultime distribuzioni presentano dei programmi di installazione che rendono il procedimento dell'installazione quasi automatico, agevolando quindi l'approccio a Linux. E' comunque necessario tenere presente che Linux, come ogni sistema Unix, è un sistema che richiede un certo sforzo prima che se ne possa apprendere la potenzialità; questa iniziale difficoltà è presto ripagata quando si cominciano ad intravedere le possibilità che vengono messe a disposizione.

Ogni distribuzione Linux contiene molti più programmi rispetto a quelli che un utente utilizzerà nella propria attività. Dopo l'installazione, se si è scelta l'opzione "server" (o comunque si chiami nelle varie distribuzioni) il sistema conterrà almeno il server HTTP Apache, un server FTP, sarà configurato per poter essere utilizzato da firewall, e moltissime altre cose. Se si utilizza il computer quale workstation, si troveranno già installati programmi per il collegamento a Internet via modem, uno o più browser, un client per IRC, eccetera.

Questi strumenti sono già preconfigurati; per utilizzarli seriamente è comunque necessario lo sforzo di capirne il funzionamento attraverso la documentazione che è fornita assieme alla distribuzione, e che comunque è integrabile da quella facilmente accessibile su Internet.

Data ultima modifica: 01/07/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux – Descrizione generale

[[Indice generale](#) | [Indice sezione](#) | << | >>]

Il sistema operativo

Il sistema operativo è l'insieme dei programmi che permettono all'utente di fare girare dei programmi sul proprio computer. Le applicazioni quali i word processor, i browser web, i fogli di calcolo, hanno bisogno di un sistema operativo per potere accedere alle risorse del computer.

Un sistema operativo può essere schematicamente diviso in cinque strati, a partire dal più vicino all'utente:

1. L'interprete dei comandi (o [shell](#)): permette l'attivazione dei programmi utente e di sistema
2. Il [filesystem](#): gestisce l'accesso ai file
3. La gestione delle periferiche: gestisce il colloquio delle applicazioni con le periferiche (es. la stampante)
4. La gestione della memoria
5. La gestione dei [processi](#)

Gli ultimi quattro strati costituiscono il *kernel* del sistema. Il sistema operativo è l'insieme del kernel e dei programmi che garantiscono il funzionamento di base del sistema. Come già ricordato nell'[Introduzione](#), "Linux" è il nome del kernel, al sistema operativo si dovrebbe dare il nome "GNU/Linux" perchè le utility di base sono quelle sviluppate dal progetto [GNU](#).

Al momento dell'accensione del computer, deve essere caricato il sistema operativo; il più diffuso programma per caricare Linux (o, in alternativa, altri sistemi operativi che possono essere stati installati), è LILO (Linux LOader).

Le [distribuzioni](#) moderne, durante la fase di installazione, permettono di conservare eventuali installazioni preesistenti di altri sistemi operativi (molto frequente è il caso che si voglia avere una macchina dual boot che permetta di caricare alternativamente Windows e Linux).

Data ultima modifica: 01/07/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux – Descrizione generale

[[Indice generale](#) | [Indice sezione](#) | << | >>]

Accedere al sistema

Linux è un sistema operativo multiutente, ovvero permette l'accesso contemporaneo di più utenti al sistema stesso.

La modalità multiutente è la modalità di default di Linux, ma non è la sola. Rispondendo al prompt di LILO, al momento dell'accensione del computer, con `linux 1` si entra in modalità monoutente. **Attenzione:** in questo modo è possibile cambiare la password di `root` a meno che non si protegga LILO con una password.

Per avere accesso ad un computer Linux è necessario inserire una *login* ed una *password*. La *login* è il nome dell'utente che accede alle risorse del sistema, la password è utilizzata per proteggere i dati personali dell'utente.

L'autenticazione dell'utente avviene tramite il comando `login`, che viene eseguito dal sistema stesso al momento dell'avvio in modalità multiutente. Questo programma richiede all'utente l'inserimento della *login* e della password. Una volta che l'utente ha inserito i propri dati, il programma deve cercare all'interno del sistema le informazioni che gli servono per permettere l'accesso dell'utente al computer. Il file che contiene le informazioni necessarie all'autenticazione è `/etc/passwd`; si possono trovare informazioni su questo file con il comando `man -S 5 passwd`

In realtà, quanto segue si riferisce al più semplice modello di autenticazione dei sistemi Unix; esistono anche sistemi come le *shadow password*, o altri ancora più sicuri, ma, essendo lo scopo semplicemente quello di illustrare il procedimento di autenticazione, non li prendiamo in considerazione.

Ogni riga di questo file individua un utente abilitato ad accedere al sistema:

```
luca:AFg6YU7rtyt2w:500:100:Luca Fasolo:/home/luca:/bin/bash
```

I dati contenuti in questa riga che interessano l'autenticazione sono i primi due, che sono rispettivamente la *login* e la password in forma crittografata. Il programma `login` verifica che esista una riga per l'utente inserito, poi prende la password inserita dall'utente e la crittografa con un algoritmo che prende i primi due caratteri della stringa della password letta da `/etc/passwd`, e li usa quale "base" per generare la stringa della password crittografata. Se la stringa così generata è uguale a quella contenuta nel file `/etc/passwd`, l'utente è abilitato ad entrare nel sistema.

Questo metodo si basa sull'assunzione che, data una base e una stringa, la stringa crittografata generata dall'algoritmo è sempre uguale (maggiori informazioni si possono avere con il comando `man crypt`). Questo procedimento ha due conseguenze: per prima cosa, se due utenti hanno la stessa password nel computer sono memorizzate stringhe diverse perchè la base dalla quale sono generate cambia sempre; secondo e più importante, **la password in chiaro non è mai immagazzinata all'interno del computer.**

Il fatto che la *login* sia una stringa serve all'utente per potersela facilmente ricordare: in realtà il computer userà, dopo la fase di autenticazione, un identificativo numerico, che nella riga di esempio è 500. Questo identificativo prende il nome di UID (User Identifier).

La [variabile d'ambiente](#) che memorizza l'utente connesso è `USER`.

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux – Descrizione generale

[[Indice generale](#) | [Indice sezione](#) | << | >>]

La shell

Una volta che l'utente ha passato la [fase di autenticazione](#), è necessario che possa interagire con il computer. Se si utilizza Linux quale desktop, è probabile che l'utente acceda ad un sistema a finestre (probabilmente GNOME o KDE) per svolgere le proprie attività. Per il momento, "torniamo al passato" e supponiamo di accedere al computer in modalità non grafica.

Per accedere al sistema in modalità non grafica si usa la combinazione di tasti CTRL-ALT-F6 (o comunque CTRL-ALT-Fn con n fra 2 e 6): Linux lascia sempre a disposizione almeno un "terminale virtuale" non grafico. Per tornare allo schermo grafico, premere CTRL-ALT-F7. Quanto segue può essere applicato anche all'avvio di applicazioni grafiche quali `xterm` o `konsole`.

Il programma `login`, dopo avere autenticato l'utente, legge ancora `/etc/passwd` e fa partire il programma che è rappresentato dall'ultimo dato della riga corrispondente all'utente: nell'esempio, `/bin/bash`.

La `bash` è la shell di default dei sistemi Linux più diffusi; una *shell* è un programma che permette l'interazione fra l'utente e il computer (come il "prompt di MSDOS" dei sistemi Windows). La shell, alla pressione del tasto invio, interpreta i comandi scritti dall'utente, li esegue e ritorna il controllo all'utente.

"Bash" è un acronimo per Bourne Again SHell; esistono altri tipi di shell: i più comuni sono Ksh (Korn SHell) e Csh (C SHell). Anche se la `bash` è molto comoda, è necessario ricordarsi che essa è standard solo su Linux: se si pensa di utilizzare altri sistemi Unix, è consigliabile esercitarsi un po' almeno con la Ksh (che, almeno sulla distribuzione RedHat, non è installata di default, ma è comune su HP-UX, IBM AIX e anche Sun Solaris). L'elenco delle shell installate nel sistema è nel file `/etc/shells`.

E' possibile trovare nel file `/etc/shells` anche programmi strani, ad esempio `/bin/false`: questa "falsa" shell può essere usata per permettere ad un utente di accedere a determinati servizi (ad esempio, l'accesso via FTP) ma impedendogli di accedere normalmente al sistema: se un utente ha `/bin/false` nella sua riga del file `/etc/passwd`, il programma `login` autentica l'utente ma egli viene disconnesso subito dopo.

La [variabile d'ambiente](#) che indica quale shell si sta usando è `SHELL`.

Una shell è un programma piuttosto complesso, che ha al suo interno i comandi più comuni: ad esempio, il comando `cd` che serve per cambiare directory è all'interno della shell (non esiste l'eseguibile `cd`). Le shell inoltre sono programmabili, nel senso che al loro interno ci sono dei semplici linguaggi di programmazione che possono essere usati per automatizzare le operazioni più comuni. Ogni shell ha il proprio linguaggio di programmazione, anche se tutti hanno una base comune.

Dato che la shell è un eseguibile, è possibile lanciare una shell da un'altra shell; in questo caso le shell si innestano, e cambia il valore della [variabile d'ambiente](#) `SHELL` (shell level):

```
$ env | grep SHLVL
SHLVL=2
$ bash
$ env | grep SHLVL
SHLVL=3
$ exit
exit
$ env | grep SHLVL
SHLVL=2
```

Il comando [grep](#) serve per filtrare l'output del comando `env`.

La programmazione della shell è orientata ai file, non ci sono i tipi di dato dei linguaggi di programmazione usuali. Vediamo un esempio che illustra anche come si possano fare le stesse cose in molti modi: supponiamo di avere una directory che contenga tre file con estensione `txt`, e di voler esaminarne il contenuto con il comando `cat`.

Il contenuto della directory è:

```
$ ls
primo.txt  secondo.txt  terzo.txt
$
```

Potremmo stampare i file uno alla volta:

```
$ cat primo.txt
questo e' il primo
file
$ cat secondo.txt
questo invece
e' il secondo

$ cat terzo.txt
questo e' il
terzo
dei file
$
```

oppure potremmo dire alla shell di stampare i contenuto di tutti i file con estensione `txt`:

```
$ cat *.txt
questo e' il primo
file
questo invece
e' il secondo

questo e' il
terzo
dei file
$
```

In questo modo, però, abbiamo perso l'informazione su quali righe compongano i vari file. Proviamo ad usare un ciclo `for` per scorrere tutti i file con estensione `txt` e stamparne il nome e il contenuto (`echo` stampa una stringa):

```
$ for i in *.txt; do echo $i; cat $i; done
primo.txt
questo e' il primo
file
secondo.txt
questo invece
e' il secondo

terzo.txt
questo e' il
terzo
dei file
```

§

Senza entrare nei dettagli dell'esempio, l'importante è rendersi conto che con la shell si possono fare molte cose, soprattutto legate alle necessità di un amministratore di sistema che per comodità deve cercare di automatizzare le operazioni di routine.

Data ultima modifica: 01/07/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux – Descrizione generale

[[Indice generale](#) | [Indice sezione](#) | << | >>]

La home directory

Al termine del processo di [accesso al sistema](#), all'utente viene messa a disposizione la *home directory*, cui l'utente accede subito dopo l'autenticazione.

La home directory dell'utente è definita nel file `/etc/passwd`, ed è il penultimo valore della riga corrispondente all'utente:

```
$ grep luca /etc/passwd
luca:x:501:501:Luca Fasolo:/home/luca:/bin/bash
$
```

Una volta aperta una [shell](#), la directory di default è la home directory (il comando `pwd` stampa la directory corrente – `pwd`=Print Working Directory):

```
[luca@bobrock luca]$ pwd
/home/luca
[luca@bobrock luca]$
```

Quello che viene stampato prima del cursore, in questo caso, è il nome dell'utente seguito dal nome della macchina e il nome della directory corrente. La variabile che definisce questo formato è `$PS1`, e il file dove essa è definita, se si usa RedHat, è `/etc/bashrc`. Per provare a fare cambiamenti, è consigliabile usare il file `.bashrc` nella home directory, senza modificare il file di sistema. Per visualizzare i file il cui nome inizia con un punto, il comando è `ls -a`

La home directory è la "casa" dell'utente, il posto dove l'utente può memorizzare i propri file e dove di solito esegue i comandi. Linux memorizza la home directory dell'utente nella [variabile d'ambiente](#) `HOME`:

```
[luca@bobrock luca]$ echo $HOME
/home/luca
[luca@bobrock luca]$
```

La shell prevede diversi modi per ricondurre l'utente alla home directory. Il comando `cd` senza parametri è uno di questi modi:

```
[luca@bobrock luca]$ pwd
/home/luca
[luca@bobrock luca]$ cd /tmp
[luca@bobrock /tmp]$ pwd
/tmp
[luca@bobrock /tmp]$ cd
[luca@bobrock luca]$ pwd
/home/luca
[luca@bobrock luca]$
```

Il simbolo che rappresenta la home directory è ~ (se si ha una tastiera italiana questo carattere si genera premendo contemporaneamente i tasti `Alt Gr` e `ì`); quindi:

```
[luca@bobrock luca]$ pwd
/home/luca
[luca@bobrock luca]$ cd /tmp
[luca@bobrock /tmp]$ pwd
/tmp
[luca@bobrock /tmp]$ cd ~
[luca@bobrock luca]$ pwd
/home/luca
[luca@bobrock luca]$
```

Usando la variabile d'ambiente `HOME`, invece:

```
[luca@bobrock luca]$ pwd
/home/luca
[luca@bobrock luca]$ cd /tmp
[luca@bobrock /tmp]$ pwd
/tmp
[luca@bobrock /tmp]$ cd $HOME
[luca@bobrock luca]$ pwd
/home/luca
[luca@bobrock luca]$
```

Data ultima modifica: 01/07/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux – File e directory

[[Indice generale](#) | [Indice sezione](#) | << | >>]**Indice**

- [Il filesystem](#)
- [I permessi](#)
- [Trovare i file \(find\)](#)
- [Archiviare i file \(tar\)](#)
- [Il tipo dei file \(file\)](#)
- [Trovare una parola nei file \(grep\)](#)
- [Confrontare due file \(diff\)](#)

Data ultima modifica: 10/07/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Il filesystem

Il *filesystem* è l'insieme delle directory e dei file gestiti dal sistema operativo. Il filesystem è una struttura ad albero, che ha una radice, detta *root* (da non confondere con l'utente *root*), il cui simbolo è */*.

Il filesystem di Linux non è esattamente un albero perchè è possibile creare dei collegamenti (*link*) da una directory all'altra; più esattamente si dovrebbe quindi parlare di una struttura a *grafo*.

La struttura del filesystem di Linux si può visualizzare con il comando `tree`

```
$ cd /
$ tree -L 2 -d (Nota: non e' riportato tutto l'output)
.
|-- bin          Programmi di sistema
|-- boot
|-- dev
|  |-- cpu
|-- etc          File di configurazione del sistema
|  |-- X11
|-- home        Area a disposizione degli utenti
|  |-- luca
|  |-- mnt
|  |-- soft
|  `-- tmp
|-- lib          Librerie di sistema
|  |-- modules
|  `-- security
|-- lost+found
|-- misc
|-- mnt          Area a disposizione delle periferiche
|  |-- cdburner
|  |-- cdrom
|  |-- disk
|  `-- floppy
|-- net
|-- opt
|  `-- htmldoc-1.8.12
|-- proc
|-- root|-- sbin
|  `-- pam_filter
|-- tmp          Area per i file temporanei
|-- usr          Area dei programmi utente
|  |-- X11R6
|  |-- bin
|  |-- doc
|  |-- etc
|  |-- lib
|  |-- man
|  |-- sbin
|  |-- share
|  |-- src
`-- var
    |-- cache
```

Informazioni sul filesystem di Linux si possono trovare alla pagina http://www.ziobudda.net/Recensioni/il_fs_di_linux.php; informazioni più dettagliate (in inglese) alla pagina <http://www.pathname.com/fhs/2.1/fhs-toc.html>.

Data ultima modifica: 29/06/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



I permessi

In un sistema multiutente, è auspicabile che ogni utente possa modificare solo i propri file, ed è prudente, dal punto di vista dell'affidabilità e stabilità del sistema, che ogni operazione sia effettuata con il minimo di permessi possibile: un utente non amministratore non ha la necessità di modificare i file di sistema per effettuare le proprie attività.

In Linux, viene predefinito un utente speciale, `root`, che svolge il ruolo di amministratore del sistema e può accedere a tutte le parti del sistema stesso. Gli altri utenti, invece, hanno il permesso di leggere e modificare i file nella loro home directory, e hanno accesso in lettura agli altri file non critici del sistema. Un utente normale non può in alcun modo accedere alle parti critiche del sistema operativo.

Ad ogni file o directory in un sistema Linux sono associati i permessi per il possessore del file stesso, per i membri del suo *gruppo* (un insieme di utenti cui è possibile associare proprietà comuni; i gruppi sono definiti in `/etc/group`; le informazioni su questo file si possono ottenere con `man -S 5 group`) e per gli utenti che non appartengono al gruppo stesso. Il comando per vedere i permessi associati ad un file è `ls -l`:

```
$ ls -l
total 3
-rw-r--r--  1 luca    users      20 Feb 12 20:46 dati.txt
drwxr-xr-x  2 luca    users     1024 Feb 12 20:47 dir1
-rwxr-xr-x  1 luca    users      69 Feb 12 20:46 eseguibile
```

I primi dieci caratteri di una riga di output del comando `ls -l` rappresentano i permessi di un file o directory (rappresentate dalla lettera `d` in prima posizione). Saltando il primo carattere, i successivi tre caratteri rappresentano l'accesso in lettura, l'accesso in scrittura e l'eseguibilità del file (l'accessibilità nel caso di directory). `r` sta per "read", `w` per "write", `x` per "execute": il file `dati.txt` nell'esempio è accessibile in lettura e scrittura da parte del possessore (`luca`), in lettura dal gruppo (`users`) e dal resto degli utenti.

I permessi di lettura (`-rwxrwxrwx`), scrittura (`-rwxrwxrwx`) ed eseguibilità (`-rwxrwxrwx`) sono rappresentati con le lettere per comodità di lettura. Il computer, internamente, usa una rappresentazione in cui un bit posto a 1 significa "permesso accordato", a 0 significa "permesso negato". Ad esempio, la stringa `-r-xr-xr--` sarà `0101101100`: il segno `-` è uno 0 e una lettera è un 1.

Questa rappresentazione non è comoda da leggere, e non viene usata, ma serve per introdurre un'altra notazione.

Se si convertono in decimale le tre cifre binarie che compongono i permessi per i vari tipi di utenti, si ottiene, nel caso di `101101100`, il numero 554 ($101=5$, $101=5$, $100=4$).

Il file `dati.txt` dell'esempio avrà quindi permessi 644.

Il comando per cambiare i permessi è `chmod`. Per consentire il permesso di modificare il file `dati.txt` agli utenti del gruppo, si può fare:

```
$ chmod g+w primo.txt
$ ls -l primo.txt
-rw-rw-r--  1 luca    users      24 Feb 15 18:55 primo.txt
$
```

oppure

```
$ chmod 664 primo.txt
$ ls -l primo.txt
-rw-rw-r--  1 luca      users          24 Feb 15 18:55 primo.txt
$
```

Il modo di usare il comando `chmod` dipende dalle preferenze personali, io preferisco usare i numeri ma conosco altri che trovano più chiaro usare le lettere. In questo caso, per togliere il permesso di scrittura al gruppo il comando è:

```
$ chmod g-w primo.txt
[luca@bobrock prova]$ ls -l primo.txt
-rw-r--r--  1 luca      users          24 Feb 15 18:55 primo.txt
$
```

Usando le lettere, `u` indica l'utente, `g` il gruppo, `o` (others) gli altri utenti. Per togliere il permesso di lettura agli "altri" e aggiungere la scrittura al gruppo, si può fare:

```
$ chmod o-r,g+w primo.txt
$ ls -l
total 3
-rw-rw----  1 luca      users          24 Feb 15 18:55 primo.txt
$
```

oppure

```
$ chmod 660 primo.txt
$ ls -l
total 3
-rw-rw----  1 luca      users          24 Feb 15 18:55 primo.txt
$
```

Data ultima modifica: 24/06/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Trovare i file (**f**ind)

Capita spesso di avere la necessità di dovere trovare un file, senza sapere in quale directory si trovi. Se si sa che il file dovrebbe essere in qualche sottodirectory di quella dove ci troviamo, è possibile usare il comando `tree`:

```
$ tree
.
|-- dir1
|   |-- dir2
|   `-- terzo.html
|-- numeri.txt
|-- primo.html
|-- provalocale.c
|-- secondo.html
`-- somma.sh
```

Il comando `tree` non è comunque standard su sistemi Unix che non siano Linux, e comunque è difficile utilizzarne l'output in qualche script; il comando più adatto è `find`. Ad esempio, so che da qualche parte sotto la [home directory](#) c'è un file che si chiama `appunti.txt`, ma non so esattamente dove:

```
$ pwd
/home/luca
[luca@bobrock luca]$ find . -name "appunti.txt" -print
./TEST/appunti/old/appunti.txt
$
```

Il file è nella sottodirectory `TEST/appunti/old/`. Il comando dato sopra si può leggere "trova, a partire dalla directory corrente, il file che si chiama `appunti.txt` e stampa il risultato".

"." rappresenta la directory corrente (che si ottiene con `pwd`), `-name "<nome>"` è il nome del file e `-print` descrive l'operazione da fare sul risultato ottenuto. E' possibile anche cercare tutti i file con una data estensione:

```
$ find ./TEST/ -name "*.txt" -print
./TEST/appunti/old/appunti.txt
./TEST/appunti/nuovo.txt
$
```

In questo modo, abbiamo dato il comando "trova, a partire dalla sottodirectory `TEST`, tutti i file che hanno estensione `txt`".

Il comando `find` può eseguire anche altre operazioni, oltre a stampare il risultato. Se vogliamo avere anche altre informazioni sui file che troviamo, ci ricordiamo che `ls -l` fornisce i dati sui file, e diamo il comando:

```
$ find ./TEST/ -name "*.txt" -exec ls -l {} \;
```

```
-rw-r--r--  1 luca  users      44 Mar  1 20:07 ./TEST/appunti/old/appunti.txt
-rw-r--r--  1 luca  users     108 Mar  2 19:05 ./TEST/appunti/nuovo.txt
$
```

Il comando si legge "trova, a partire dalla sottodirectory TEST, tutti i file che hanno estensione txt ed esegui (-exec) su quei file il comando ls -l". Le parentesi graffe sono l'argomento del comando ls -l.

Può capitare di dovere cercare dei file al di fuori della propria home directory. Ad esempio, per cercare in tutto il disco un file, il comando è

```
find / -name "nomefile" -print
```

Se non si è root, lo schermo si riempirà di errori perchè ci sono molte directory nelle quali non si ha il permesso di lettura. Per non vedere gli errori, si usa il comando

```
find / -name "nomefile" -print 2> /dev/null
```

Per il significato di 2> ci si può riferire alla sezione [Input e output dei programmi](#)

Data ultima modifica: 24/06/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux – File e directory

[[Indice generale](#) | [Indice sezione](#) | [<<](#) | [>>](#)]

Archiviare i file (**tar**)

L'utility standard dei sistemi Unix per archiviare dei file è il comando `tar`, Tape ARchive. Il nome svela l'origine del comando, che nacque per memorizzare su nastro il contenuto dei dischi per effettuare il backup dei dati in essi contenuti.

Linux utilizza il comando `tar` del progetto GNU; altri Unix utilizzano il proprio `tar`; se si utilizza Sun Solaris può essere necessario sostituire il `tar` nativo con quello GNU per poter gestire archivi che contengano file il cui path sia più lungo di un certo numero di caratteri.

Gli archivi in formato `tar` sono di solito diffusi in formato compresso, e in questo caso hanno estensione `.tgz` o `.tar.gz`: l'estensione indica che sono compressi con l'utility `gzip` (Gnu ZIP).

Alcuni archivi sono compressi con l'utility `bzip2`, più recente di `gzip`; in questo caso hanno estensione `.tar.bz2`.

Per vedere il contenuto di un file `tar` compresso con `gzip`, il comando è

```
tar tvfz nomearchivio.tgz
```

Per vedere il contenuto di un archivio `tar` compresso con `bzip`, il comando è

```
tar tvfI nomearchivio.tar.bz2
```

Se l'archivio non è compresso, il comando è

```
tar tvf nomearchivio.tar
```

Prima di estrarre i file da un archivio, è bene visualizzarne il contenuto per vedere dove effettivamente verranno estratti i file.

Per estrarre un archivio `tar` compresso rispettivamente con `gzip` e `bzip` i comandi sono

```
tar xvfz nomearchivio.tgz
tar xvfi nomearchivio.tar.bz2
```

Per creare un archivio, infine, il comando è

```
tar cvfz nomearchiviodacreare.tgz directorydaarchiviare/
```

Ad esempio, supponiamo di volere archiviare la directory `TEST` il cui contenuto è:

```
$ ls -lR TEST/
TEST/:
total 7
-rw-r--r--  1 luca  users      959 Mar  4 16:33 Prova.class
-rw-r--r--  1 luca  users      610 Mar  4 16:32 Prova.java
-rw-r--r--  1 luca  users      343 Mar  4 16:23 Triangolo.class
-rw-r--r--  1 luca  users      120 Mar  4 16:23 Triangolo.java
drwxr-xr-x  3 luca  users    1024 Mar  2 19:05 appunti
-rw-r--r--  1 luca  users      431 Feb 20 19:02 primo.class
```

```

-rw-r--r--  1 luca  users      121 Feb 20 19:01 primo.java

TEST/appunti:
total 2
-rw-r--r--  1 luca  users      108 Mar  2 19:05 nuovo.txt
drwxr-xr-x  2 luca  users     1024 Mar  2 19:05 old

TEST/appunti/old:
total 1
-rw-r--r--  1 luca  users       44 Mar  2 19:05 appunti.txt
$

```

I comandi seguenti creano l'archivio TEST.tgz e ne esaminano il contenuto:

```

$ tar cvfz TEST.tgz TEST/
TEST/
TEST/primo.java
TEST/primo.class
TEST/Triangolo.class
TEST/Triangolo.java
TEST/Prova.java
TEST/Prova.class
TEST/appunti/
TEST/appunti/old/
TEST/appunti/old/appunti.txt
TEST/appunti/nuovo.txt
$ tar tvfz TEST.tgz
drwxr-xr-x luca/users      0 2001-03-04 16:32:26 TEST/
-rw-r--r-- luca/users     121 2001-02-20 19:01:56 TEST/primo.java
-rw-r--r-- luca/users     431 2001-02-20 19:02:05 TEST/primo.class
-rw-r--r-- luca/users     343 2001-03-04 16:23:40 TEST/Triangolo.class
-rw-r--r-- luca/users     120 2001-03-04 16:23:08 TEST/Triangolo.java
-rw-r--r-- luca/users     610 2001-03-04 16:32:33 TEST/Prova.java
-rw-r--r-- luca/users     959 2001-03-04 16:33:22 TEST/Prova.class
drwxr-xr-x luca/users      0 2001-03-02 19:05:30 TEST/appunti/
drwxr-xr-x luca/users      0 2001-03-02 19:05:09 TEST/appunti/old/
-rw-r--r-- luca/users      44 2001-03-02 19:05:09 TEST/appunti/old/appunti.txt
-rw-r--r-- luca/users     108 2001-03-02 19:05:30 TEST/appunti/nuovo.txt
$

```

Per estrarre il contenuto di TEST.tgz nella directory /tmp:

```

$ cp TEST.tgz /tmp/
$ cd /tmp
$ tar xvfz TEST.tgz
TEST/
TEST/primo.java
TEST/primo.class
TEST/Triangolo.class
TEST/Triangolo.java
TEST/Prova.java
TEST/Prova.class
TEST/appunti/
TEST/appunti/old/
TEST/appunti/old/appunti.txt
TEST/appunti/nuovo.txt
$

```

La directory /tmp ora contiene la sottodirectory TEST.

In modo analogo, e senza copiare l'archivio nella directory dove deve essere scompattato, è possibile dare il comando

```
$ tar xvfz TEST.tgz -C /tmp
```

Data ultima modifica: 01/07/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux – File e directory

[[Indice generale](#) | [Indice sezione](#) | [<<](#) | [>>](#)]

Il tipo dei file (**file**)

Il comando `file` permette di determinare di che tipo è un file; questo può essere comodo, per esempio, per l'esecuzione di script automatizzati che basano il loro funzionamento sul tipo di file esaminati.

La sintassi di base è molto semplice: `file <nomefile>`. Ad esempio:

```
$ file Logo.gif unnamed.eps xhtml1.zip
Logo.gif:      GIF image data, version 89a, 296 x 294,
unnamed.eps:  PostScript document text conforming at level 2.0 - type EPS
xhtml1.zip:   Zip archive data, at least v1.0 to extract
$
```

Il file che contiene le definizioni dei tipi di file conosciuti è `/usr/share/magic`.

Un modo per vedere quali tipi di file sono registrati nel proprio sistema è il comando

```
$ cat /usr/share/magic | egrep -v ["#", ">"] - | \
> awk '{for(i=4;i
```

Data ultima modifica: 26/04/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux – File e directory

[[Indice generale](#) | [Indice sezione](#) | [<<](#) | [>>](#)]

Trovare una parola nei file (`grep`)

E' spesso necessario sapere quale file contiene una data parola; ad esempio, durante lo sviluppo di un programma si cambia il nome di una funzione ed è necessario individuare quali file stanno usando il nome vecchio. Il comando che si usa in questo caso è `grep`, la cui sintassi di base è

```
$ grep stringa file
```

`grep` stampa le righe dei file nei quali ha trovato la stringa; ad esempio, se si ha la necessità di vedere le occorrenze della parola "temp" nei file html in una directory, il comando è

```
$ grep temp *.html
primo.html:temporaneo
primo.html:temp
secondo.html:tempo
$
```

Si vede come il comando restituisca le righe dei file che contengono la *stringa* "temp", non la *parola*. Se si vogliono vedere solo quelle che contengono "temp" quale parola, bisogna usare l'opzione `-w` (`w` = word):

```
$ grep -w "temp" *.html
primo.html:temp
$
```

Se invece si vogliono vedere le righe che *non* contengono una data stringa, l'opzione da usare è `-v`:

```
$ grep -v "temp" *.html
primo.html:esempio
primo.html:es2
secondo.html:prima riga
secondo.html:altra
secondo.html:
```

Combinando le opzioni precedenti, si possono vedere le righe che non contengono una data parola:

```
$ grep -v -w "temp" *.html
primo.html:temporaneo
primo.html:esempio
primo.html:es2
secondo.html:prima riga
secondo.html:tempo
secondo.html:altra
secondo.html:
```

Per cercare ricorsivamente in tutte le sottodirectory una stringa, è possibile usare l'opzione `-r`:

```
$ grep -r "temp" *
dir1/terzo.html:temp
primo.html:temporaneo
primo.html:temp
secondo.html:tempo
$
```

Data ultima modifica: 24/06/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Confrontare due file (`diff`)

Il comando per confrontare due file è `diff`; per questo comando, è utile consultare `info diff` oltre a [man diff](#).

Vediamo l'utilizzo di `diff` su un esempio, applicando il comando ai due file `primo.txt` e `secondo.txt`:

```
$ cat primo.txt
questo e' il titolo
questa e' la prima riga
questa e' la seconda riga
questa e' la terza riga
c'e' anche una quarta riga
$ cat secondo.txt
questa e' la prima riga con modifiche
questa e' la seconda riga
questa anche e' la terza riga
c'e' anche una quarta riga
quinta riga
$
```

L'output del comando `diff` è

```
$ diff primo.txt secondo.txt
1,2c1
questo e' il titolo
questa e' la prima riga
---
> questa e' la prima riga con modifiche
4c3
questa e' la terza riga
---
> questa anche e' la terza riga
5a5
> quinta riga
$
```

L'output del comando riporta i comandi necessari per applicare le differenze al primo file ed ottenere il secondo. `1,2c1` significa "sostituisci le righe 1 e 2 del primo file con la riga 1 del secondo file"; `5a5` significa "aggiungi la riga 5 del secondo file dopo la riga 5 del primo file".

Seguendo le istruzioni e applicando man mano le differenze al primo file, si può ricostruire il secondo.

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux

[[Indice generale](#)]

Gli utenti

Il file dove sono definiti gli utenti è `/etc/passwd`; dopo l'installazione, il contenuto di questo file sarà simile al seguente (che è il file che si ottiene dopo l'installazione completa di una distribuzione Mandrake 7.2):

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:
daemon:x:2:2:daemon:/sbin:
adm:x:3:4:adm:/var/adm:
lp:x:4:7:lp:/var/spool/lpd:
...
nobody:x:99:99:Nobody:/:
xfs:x:100:102:X Font Server:/etc/X11/fs:/bin/false
htdig:x:101:104:./var/lib/htdig:
...
mysql:x:103:238:MySQL server:/var/lib/mysql:/bin/bash
zope:x:104:239:Zope Server:/var/zope:/bin/bash
luca:x:501:501:Luca Fasolo:/home/luca:/bin/bash
```

Si vede come il primo utente sia `root`, ovvero il superutente del sistema, e l'ultimo sia l'utente la cui creazione è di solito suggerita durante l'installazione.

E' assolutamente consigliabile collegarsi al sistema come utente normale (non `root`) e acquisire i diritti di `root` solo in caso di necessità. Per diventare `root` si usa il comando `su -`; il comando `su` (switch user) ha in generale la forma `su - utente` e permette, conoscendo la password, di assumere l'identità dell'utente desiderato.

Gli altri utenti definiti sono gli utenti di sistema e gli utenti usati dalle varie applicazioni per fare girare i propri processi (ad esempio, di solito l'utente `nobody` è il proprietario dei processi di Apache).

Si possono vedere gli utenti connessi al sistema con il comando `who`; gli ultimi `n` utenti che si sono connessi al sistema con `last -n` (ad esempio `last -10`).

Il comando `finger` permette di ottenere alcune informazioni sugli utenti registrati nel sistema; se nella directory dell'utente sono presenti i file `.plan` e/o `.project`, `finger` visualizza il testo contenuto in essi; ad esempio

```
$ finger luca
Login: luca                               Name: Luca Fasolo
Directory: /home/luca                     Shell: /bin/bash
On since Fri Jun 15 17:55 (CEST) on :0 (messages off)
No mail.
Project:
progetto assegnato a luca
Plan:
sto sistemando le dispense
$
```

Se gli utenti sono tenuti a mantenere aggiornati questi file, `finger` può essere uno strumento utile ad un amministratore di rete o a un capo progetto.

Altre informazioni sugli utenti si possono trovare nella sezione [I permessi](#).

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux

[[Indice generale](#)]

I processi (ps)

Un processo è un programma in esecuzione che dispone di un'area di memoria ad esso riservata; ogni processo in Linux ha un identificativo (PID, Process Identifier) e un owner (l'utente che ha lanciato il comando).

A volte l'owner del processo non è l'utente che ha lanciato il comando, in quanto è possibile in alcuni casi lanciare un processo assumendo l'identità di un altro. Ad esempio, l'owner dei processi `httpd` di Apache che si occupano di gestire le richieste al server web è `nobody` anche se il server viene fatto partire da `root`.

Il comando per visualizzare i processi in esecuzione è `ps`:

```
$ ps
  PID TTY          TIME CMD
  603 pts/1        00:00:00 bash
  621 pts/1        00:00:06 nedit
  622 pts/1        00:00:00 ps
$
```

Il comando `ps`, senza opzioni, stampa i processi dell'utente; se si vogliono vedere tutti i dettagli dei processi in esecuzione sul sistema, il comando è `ps aux`.

E' possibile usare anche il comando `ps -def` che usa le opzioni del programma `ps` in uso su altri tipi di Unix.

Altra opzione utile è `w`; ogni `w` aggiunge una riga all'output di ogni processo, e quindi `ps auxww` fornisce informazioni in più rispetto a `ps aux`; ad esempio,

```
$ ps aux | grep netscape
luca      581  0.2  10.5  22396  13732 ?        S    10:09   0:02 /usr/lib/netscape
$ ps auxww | grep netscape
luca      581  0.2  10.5  22396  13732 ?        S    10:09   0:02 /usr/lib/netscape/nets
cape-communicator -session-management -no-about-splash
```

Il comando `pstree` visualizza la dipendenza fra i processi:

```
$ pstree
init--cupsd
  |-8*[kdeinit]
  |-kdeinit--kdeinit
  |   |-kdeinit---bash---nedit
  |   |-kdeinit---bash---pstree
  |   |-kdeinit---bash---man---sh---sh--groff---grotty
  |   |   |
  |   |   `--less
  |   `--netscape-commun---netscape-commun
  |-kdeinit---cat
  |-kdm--X
  |   `--kdm---startkde---ksmserver
  |-kflushd
  |-klogd
  |-knotify
  |-kswapd
  |-kupdate
  |-mdrecoveryd
  |-6*[mingetty]
```

```
|-syslogd
`-xfs
```

E' possibile inviare dei *segnali* ai processi, tramite il comando `kill`, ad esempio per causarne la terminazione: in questo caso il comando è `kill -9 <PID>` (oppure `kill -s SIGKILL <PID>`), in quanto il segnale 9 (SIGKILL) causa la terminazione del processo. Maggiori informazioni sui segnali si trovano con [man -S 7 signal](#). Riprendendo l'esempio sopra, per terminare il processo di Netscape il comando è

```
$ kill -9 581
```

L'elenco dei segnali disponibili si ottiene con `kill -l`:

```
$ kill -l
 1) SIGHUP          2) SIGINT          3) SIGQUIT        4) SIGILL
 5) SIGTRAP        6) SIGABRT        7) SIGBUS         8) SIGFPE
 9) SIGKILL        10) SIGUSR1       11) SIGSEGV       12) SIGUSR2
13) SIGPIPE       14) SIGALRM       15) SIGTERM       17) SIGCHLD
18) SIGCONT       19) SIGSTOP       20) SIGTSTP       21) SIGTTIN
22) SIGTTOU       23) SIGURG        24) SIGXCPU       25) SIGXFSZ
26) SIGVTALRM    27) SIGPROF       28) SIGWINCH      29) SIGIO
30) SIGPWR       31) SIGSYS        32) SIGRTMIN      33) SIGRTMIN+1
34) SIGRTMIN+2   35) SIGRTMIN+3   36) SIGRTMIN+4   37) SIGRTMIN+5
38) SIGRTMIN+6   39) SIGRTMIN+7   40) SIGRTMIN+8   41) SIGRTMIN+9
42) SIGRTMIN+10  43) SIGRTMIN+11  44) SIGRTMIN+12  45) SIGRTMIN+13
46) SIGRTMIN+14  47) SIGRTMIN+15  48) SIGRTMAX-15  49) SIGRTMAX-14
50) SIGRTMAX-13  51) SIGRTMAX-12  52) SIGRTMAX-11  53) SIGRTMAX-10
54) SIGRTMAX-9   55) SIGRTMAX-8   56) SIGRTMAX-7   57) SIGRTMAX-6
58) SIGRTMAX-5   59) SIGRTMAX-4   60) SIGRTMAX-3   61) SIGRTMAX-2
62) SIGRTMAX-1   63) SIGRTMAX
$
```

Un utilizzo diverso dei segnali è la stampa dello stacktrace di una virtual machine Java; il comando `kill -3 <PID della vm>` stampa sullo standard error della virtual machine lo stacktrace.

Data ultima modifica: 24/06/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux

[[Indice generale](#)]

Informazioni sui comandi (man)

Le attuali interfacce grafiche disponibili per Linux (principalmente [KDE](#) e [Gnome](#)) sono giunte ad un ottimo livello di sviluppo e consentono di usare il computer per molte attività senza ricorrere all'uso di un terminale (o [shell](#)) dove scrivere i comandi. Molte cose però si debbono fare "a mano", e quindi è necessario conoscere l'uso dei comandi principali di Linux.

Linux, come tutti i sistemi Unix, ha una manualistica in linea che si attiva con il comando `man`. Ad esempio, per vedere la documentazione del comando `ls` il comando è:

```
$ man ls
```

Questo comando fa apparire la prima pagina del manuale relativo al comando `ls`; per avanzare di una pagina, premere `spazio`, per tornare indietro di una pagina, premere `b`, per uscire e tornare alla shell premere `q`.

La maggior parte della documentazione è in realtà mantenuta in file di tipo `info`. Imparare ad usare `man` è comunque utile perchè è lo standard sui sistemi Unix.

Il problema che si incontra subito è che non si può studiare qualcosa se non si sa cosa studiare. Linux ha moltissimi comandi; per averne un elenco si può usare il comando `xman`: questo comando apre una piccola finestra. Cliccando su `Manual Page` e selezionando `User Commands` dal menu a tendina `Sections`, si ottiene la lista dei comandi disponibili. Cliccando sul nome del comando, si apre una finestra con il manuale del comando stesso.

`xman` può essere utilmente utilizzato per curiosare fra i tantissimi comandi disponibili, scoprendone di nuovi e iniziando a navigare in un ambiente complesso che difficilmente si riesce ad esplorare fino in fondo.

Una cosa curiosa è che è disponibile in linea anche il manuale delle istruzioni del linguaggio C, contenuto nella sezione 3 della manualistica. Ad esempio,

```
$ man -S 3 printf
```

è la descrizione del funzionamento della istruzione C per stampare una stringa formattata.

`printf` è anche un esempio di uso delle sezioni della manualistica: `man printf` si riferisce al comando che si può dare a una shell (equivalente a `man -S 1 printf`), mentre `man -S 3 printf` è l'istruzione C.

Data ultima modifica: 14/06/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Input e output dei programmi

I programmi richiamabili da riga di comando hanno tre canali predefiniti per ricevere e trasmettere dati da e verso i dispositivi di input e output: lo *standard input* è il canale attraverso il quale il programma riceve i dati di ingresso; lo *standard output* è il canale di uscita del risultato dell'elaborazione del programma; lo *standard error* è dove il programma stampa eventuali errori occorsi durante l'esecuzione.

Gli operatori che gestiscono questi canali sono:

- `>` : reindirige lo standard output;
- `>>` : reindirige lo standard output verso un dispositivo aperto in modalità *append*;
- `<` : reindirige lo standard input;
- `2>` : reindirige lo standard error;
- `|` : operatore *pipe*, concatena standard output e standard input di due programmi.

Ad esempio, il comando `ls tmp/ > outls.txt` crea il file `outls.txt` che contiene l'output del comando `ls` nella directory `tmp/`:

```
$ ls tmp/ > outls.txt
$ cat outls.txt
primo.html
secondo.html
$ ls tmp/ >> outls.txt
$ cat outls.txt
primo.html
secondo.html
primo.html
secondo.html
```

L'operatore `<` è usato per mandare in ingresso ad un programma i dati contenuti in un file; ad esempio, lo script `somma.sh` legge da standard input due numeri e scrive la somma:

```
$ cat somma.sh
read a
read b
expr $a + $b
$
```

Se nel file `numeri.txt` ci sono due numeri, le operazioni di `read` saranno effettuate sui numeri contenuti in `numeri.txt`:

```
$ cat numeri.txt
5
4
$ ./somma.sh numeri.txt
9
$
```

L'operatore pipe | concatena lo standard input di un programma allo standard input di un altro. Ad esempio,

```
$ cat numeri.txt | ./somma.sh
9
$
```

L'operatore >> permette di aggiungere in coda ad un file l'output di un altro processo; ad esempio

```
$ cat numeril.txt
6
5
$ cat numeril.txt >> numeri.txt
$ cat numeri.txt
5
4
6
5
$
```

Data ultima modifica: 20/06/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux

[[Indice generale](#)]

Le variabili d'ambiente

Le *variabili d'ambiente* sono le variabili che permettono ad ogni utente di configurare il proprio ambiente di lavoro. Il comando per elencare quelle definite in un dato momento è `env`:

```
$ env
...
HOSTTYPE=i386
PATH=/usr/local/bin:/usr/bin:/bin:/usr/X11R6/bin:/usr/local/pgsql/bin:
/usr/local/bin/scripts:/opt/j2sdk1.3.0/bin
HOME=/home/luca
SHELL=/bin/bash
USER=luca
PGDATA=/home/pgsql/data
...
OSTYPE=Linux
...
```

Alcune di queste variabili sono utilizzate dalla maggior parte dei programmi più comuni, altre sono specifiche per un particolare applicativo; ad esempio la variabile `PGDATA` è significativa per il database Postgres.

I file di sistema nei quali sono definite le variabili d'ambiente sono `/etc/profile`, `/etc/bashrc`, `$HOME/.bashrc`, `$HOME/.bash_profile`.

Lo *scope* (visibilità) delle variabili d'ambiente è limitato alla [shell](#) attuale se la variabile stessa è definita da riga di comando, mediante il comando `export NOMEVAR=valore` se si usa la bash (oppure `setenv NOMEVAR=valore` con la C Shell).

Per fare in modo che una variabile sia visibile da tutto l'ambiente, usando la bash, è possibile definirla nel file `.bashrc`.

Se si inserisce una nuova variabile d'ambiente in `.bashrc`, essa sarà visibile da tutte le shell che saranno aperte da quel momento in poi. Perchè essa sia visibile anche dalla bash corrente, è sufficiente dare il comando `source ~/.bashrc` oppure `source $HOME/.bashrc`.

Per stampare il valore di una variabile d'ambiente, il comando è `echo $nomevariabile`, ad esempio

```
$ echo $HOME
/home/luca
```

Alcune importanti variabili d'ambiente sono [HOME](#), [USER](#), [SHELL](#), `PATH`.

Data ultima modifica: 21/06/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della

GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".



Introduzione a GNU/Linux

[[Indice generale](#)]

Un correttore ortografico (`ispell`)

Un programma molto utile quando si scrive un testo è un correttore ortografico. In Linux, è disponibile il programma `ispell`.

L'utilizzo di `ispell` è molto semplice; la sintassi di base è:

```
$ ispell nomefile
```

Dopo avere dato il comando, nella shell appaiono in sequenza le parole non inserite nel vocabolario utilizzato dal programma; il tasto `A` permette di accettare le parole apparentemente non corrette, il numero corrispondente alle possibili correzioni elencate le sostituisce alla parola errata, `I` inserisce la parola nel vocabolario dell'utente, che si va ad aggiungere a quello predefinito ed è memorizzato nel file `$HOME/.ispell_italian` se si utilizza il vocabolario italiano.

Nel caso il sistema non sia preconfigurato per usare la lingua italiana, si può usare il dizionario italiano con il comando

```
$ ispell -d /usr/lib/ispell/italian.hash nomefile
```

Il dizionario può essere memorizzato in un'altra directory; può essere localizzato con il comando [find](#).

Alla fine della correzione, nella directory si trovano i file

```
$ ls
nomefile nomefile.bak
```

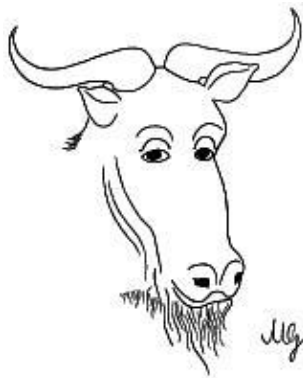
Il file con estensione `.bak` è la versione prima della correzione.

Data ultima modifica: 07/07/2001

Copyright (c) 2001 Luca Fasolo (l.fasolo@libero.it).

E' garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della GNU Free Documentation License, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation; con le Sezioni Non Modificabili [Introduzione](#), nessun testo di Copertina, e nessun testo di Retro Copertina. Una copia della licenza è acclusa nella sezione intitolata "[GNU Free Documentation License](#)".

GNU Free Documentation License



- [Cosa fare in caso di violazioni della licenza GFDL](#)
 - [Traduzioni di GFDL](#)
 - GNU Free Documentation License [in formato Docbook](#)
 - [Perché dovresti usare la GNU Free Documentation License](#)
 - [Come usare le caratteristiche opzionali della GFDL](#)
-

Contenuti

- [GNU Free Documentation License](#)
 - [Come usare questa licenza per i vostri documenti](#)
-

[GNU Free Documentation License](#)

Versione 1.1, Marzo 2000

Copyright (C) 2000 Free Software Foundation, Inc. 59 Temple Place,
Suite 330, Boston, MA 02111-1307 USA Chiunque può
copiare e distribuire copie letterali di questo documento di
licenza, ma non ne è permessa la modifica.

0. PREAMBOLO

Lo scopo di questa licenza è di rendere un manuale, un testo o altri documenti scritti "liberi" nel senso di assicurare a tutti la libertà effettiva di copiarli e redistribuirli, con o senza modifiche, a fini di lucro o no. In secondo luogo questa licenza prevede per autori ed editori il modo per ottenere il giusto riconoscimento del proprio lavoro, preservandoli dall'essere considerati responsabili per modifiche apportate da altri.

Questa licenza è un "copyleft": ciò vuol dire che i lavori che derivano dal documento originale devono essere ugualmente liberi. È il complemento alla GNU General Public License, che è una licenza di tipo "copyleft" pensata per il software libero.

Abbiamo progettato questa licenza al fine di applicarla alla documentazione del software libero, perché il software libero ha bisogno di documentazione libera: un programma libero dovrebbe accompagnarsi a manuali che forniscano la stessa libertà del software. Ma questa licenza non è limitata alla documentazione del software; può essere utilizzata per ogni testo che tratti un qualsiasi argomento e al di là dell'avvenuta pubblicazione cartacea. Raccomandiamo principalmente questa licenza per opere che abbiano fini didattici o per manuali di consultazione.

GNU Free Documentation License

1. APPLICABILITÀ E DEFINIZIONI

Questa licenza si applica a qualsiasi manuale o altra opera che contenga una nota messa dal detentore del copyright che dica che si può distribuire nei termini di questa licenza. Con "Documento", in seguito ci si riferisce a qualsiasi manuale o opera. Ogni fruitore è un destinatario della licenza e viene indicato con "voi".

Una "versione modificata" di un documento è ogni opera contenente il documento stesso o parte di esso, sia riprodotto alla lettera che con modifiche, oppure traduzioni in un'altra lingua.

Una "sezione secondaria" è un'appendice cui si fa riferimento o una premessa del documento e riguarda esclusivamente il rapporto dell'editore o dell'autore del documento con l'argomento generale del documento stesso (o argomenti affini) e non contiene nulla che possa essere compreso nell'argomento principale. (Per esempio, se il documento è in parte un manuale di matematica, una sezione secondaria non può contenere spiegazioni di matematica). Il rapporto con l'argomento può essere un tema collegato storicamente con il soggetto principale o con soggetti affini, o essere costituito da argomentazioni legali, commerciali, filosofiche, etiche o politiche pertinenti.

Le "sezioni non modificabili" sono alcune sezioni secondarie i cui titoli sono esplicitamente dichiarati essere sezioni non modificabili, nella nota che indica che il documento è realizzato sotto questa licenza.

I "testi copertina" sono dei brevi brani di testo che sono elencati nella nota che indica che il documento è realizzato sotto questa licenza.

Una copia "trasparente" del documento indica una copia leggibile da un calcolatore, codificata in un formato le cui specifiche sono disponibili pubblicamente, i cui contenuti possono essere visti e modificati direttamente, ora e in futuro, con generici editor di testi o (per immagini composte da pixel) con generici editor di immagini o (per i disegni) con qualche editor di disegni ampiamente diffuso, e la copia deve essere adatta al trattamento per la formattazione o per la conversione in una varietà di formati atti alla successiva formattazione. Una copia fatta in un altro formato di file trasparente il cui markup è stato progettato per intralciare o scoraggiare modifiche future da parte dei lettori non è trasparente. Una copia che non è trasparente è "opaca".

Esempi di formati adatti per copie trasparenti sono l'ASCII puro senza markup, il formato di input per Texinfo, il formato di input per LaTeX, SGML o XML accoppiati ad una DTD pubblica e disponibile, e semplice HTML conforme agli standard e progettato per essere modificato manualmente. Formati opachi sono PostScript, PDF, formati proprietari che possono essere letti e modificati solo con word processor proprietari, SGML o XML per cui non è in genere disponibile la DTD o gli strumenti per il trattamento, e HTML generato automaticamente da qualche word processor per il solo output.

La "pagina del titolo" di un libro stampato indica la pagina del titolo stessa, più qualche pagina seguente per quanto necessario a contenere in modo leggibile, il materiale che la licenza prevede che compaia nella pagina del titolo.

Per opere in formati in cui non sia contemplata esplicitamente la pagina del titolo, con "pagina del titolo" si intende il testo prossimo al titolo dell'opera, precedente l'inizio del corpo del testo.

2. COPIE ALLA LETTERA

Si può copiare e distribuire il documento con l'ausilio di qualsiasi mezzo, per fini di lucro e non, fornendo per tutte le copie questa licenza, le note sul copyright e l'avviso che questa licenza si applica al documento, e che non si aggiungono altre condizioni al di fuori di quelle della licenza stessa. Non si possono usare misure tecniche per impedire o controllare la lettura o la produzione di copie successive alle copie che si producono o distribuiscono. Però si possono ricavare compensi per le copie fornite. Se si distribuiscono un numero sufficiente di copie si devono seguire anche le condizioni della sezione 3.

Si possono anche prestare copie e con le stesse condizioni sopra menzionate possono essere utilizzate in pubblico.

3. COPIARE IN NOTEVOLI QUANTITÀ

Se si pubblicano a mezzo stampa più di 100 copie del documento, e la nota della licenza indica che esistono uno o più testi copertina, si devono includere nelle copie, in modo chiaro e leggibile, tutti i testi copertina indicati: il testo della prima di copertina in prima di copertina e il testo di quarta di copertina in quarta di copertina. Ambedue devono identificare l'editore che pubblica il documento. La prima di copertina deve presentare il titolo completo con tutte le parole che lo compongono egualmente visibili ed evidenti. Si può aggiungere altro materiale alle copertine. Il copiare con modifiche limitate alle sole copertine, purché si preservino il titolo e le altre condizioni viste in precedenza, è considerato alla stregua di copiare alla lettera.

Se il testo richiesto per le copertine è troppo voluminoso per essere riprodotto in modo leggibile, se ne può mettere una prima parte per quanto ragionevolmente può stare in copertina, e continuare nelle pagine immediatamente seguenti.

Se si pubblicano o distribuiscono copie opache del documento in numero superiore a 100, si deve anche includere una copia trasparente leggibile da un calcolatore per ogni copia o menzionare per ogni copia opaca un indirizzo di una rete di calcolatori pubblicamente accessibile in cui vi sia una copia trasparente completa del documento, spogliato di materiale aggiuntivo, e a cui si possa accedere anonimamente e gratuitamente per scaricare il documento usando i protocolli standard e pubblici generalmente usati. Se si adotta l'ultima opzione, si deve prestare la giusta attenzione, nel momento in cui si inizia la distribuzione in quantità elevata di copie opache, ad assicurarsi che la copia trasparente rimanga accessibile all'indirizzo stabilito fino ad almeno un anno di distanza dall'ultima distribuzione (direttamente o attraverso rivenditori) di quell'edizione al pubblico.

È caldamente consigliato, benché non obbligatorio, contattare l'autore del documento prima di distribuirne un numero considerevole di copie, per metterlo in grado di fornire una versione aggiornata del documento.

4. MODIFICHE

Si possono copiare e distribuire versioni modificate del documento rispettando le condizioni delle precedenti sezioni 2 e 3, purché la versione modificata sia realizzata seguendo scrupolosamente questa stessa licenza, con la versione modificata che svolga il ruolo del "documento", così da estendere la licenza sulla distribuzione e la modifica a chiunque ne possieda una copia. Inoltre nelle versioni modificate si deve:

- **A.** Usare nella pagina del titolo (e nelle copertine se ce ne sono) un titolo diverso da quello del documento, e da quelli di versioni precedenti (che devono essere elencati nella sezione storia del documento ove presenti). Si può usare lo stesso titolo di una versione precedente se l'editore di quella versione originale ne ha dato il permesso.
- **B.** Elencare nella pagina del titolo, come autori, una o più persone o gruppi responsabili in qualità di autori delle modifiche nella versione modificata, insieme ad almeno cinque fra i principali autori del documento (tutti gli autori principali se sono meno di cinque).
- **C.** Dichiarare nella pagina del titolo il nome dell'editore della versione modificata in qualità di editore.
- **D.** Conservare tutte le note sul copyright del documento originale.
- **E.** Aggiungere un'appropriata licenza per le modifiche di seguito alle altre licenze sui copyright.
- **F.** Includere immediatamente dopo la nota di copyright, un avviso di licenza che dia pubblicamente il permesso di usare la versione modificata nei termini di questa licenza, nella forma mostrata nell'addendum alla fine di questo testo.
- **G.** Preservare in questo avviso di licenza l'intera lista di sezioni non modificabili e testi copertina richieste come previsto dalla licenza del documento.
- **H.** Includere una copia non modificata di questa licenza.

- **I.** Conservare la sezione intitolata "Storia", e il suo titolo, e aggiungere a questa un elemento che riporti al minimo il titolo, l'anno, i nuovi autori, e gli editori della versione modificata come figurano nella pagina del titolo. Se non ci sono sezioni intitolate "Storia" nel documento, createne una che riporti il titolo, gli autori, gli editori del documento come figurano nella pagina del titolo, quindi aggiungete un elemento che descriva la versione modificata come detto in precedenza.
- **J.** Conservare l'indirizzo in rete riportato nel documento, se c'è, al fine del pubblico accesso ad una copia trasparente, e possibilmente l'indirizzo in rete per le precedenti versioni su cui ci si è basati. Questi possono essere collocati nella sezione "Storia". Si può omettere un indirizzo di rete per un'opera pubblicata almeno quattro anni prima del documento stesso, o se l'originario editore della versione cui ci si riferisce ne dà il permesso.
- **K.** In ogni sezione di "Ringraziamenti" o "Dediche", si conservino il titolo, il senso, il tono della sezione stessa.
- **L.** Si conservino inalterate le sezioni non modificabili del documento, nei propri testi e nei propri titoli. I numeri della sezione o equivalenti non sono considerati parte del titolo della sezione.
- **M.** Si cancelli ogni sezione intitolata "Riconoscimenti". Solo questa sezione può non essere inclusa nella versione modificata.
- **N.** Non si modifichi il titolo di sezioni esistenti come "miglioria" o per creare confusione con i titoli di sezioni non modificabili.

Se la versione modificata comprende nuove sezioni di primaria importanza o appendici che ricadono in "sezioni secondarie", e non contengono materiale copiato dal documento, si ha facoltà di rendere non modificabili quante sezioni si voglia. Per fare ciò si aggiunga il loro titolo alla lista delle sezioni immutabili nella nota di copyright della versione modificata. Questi titoli devono essere diversi dai titoli di ogni altra sezione.

Si può aggiungere una sezione intitolata "Riconoscimenti", a patto che non contenga altro che le approvazioni alla versione modificata prodotte da vari soggetti—per esempio, affermazioni di revisione o che il testo è stato approvato da una organizzazione come la definizione normativa di uno standard.

Si può aggiungere un brano fino a cinque parole come Testo Copertina, e un brano fino a 25 parole come Testo di Retro Copertina, alla fine dell'elenco dei Testi Copertina nella versione modificata. Solamente un brano del Testo Copertina e uno del Testo di Retro Copertina possono essere aggiunti (anche con adattamenti) da ciascuna persona o organizzazione. Se il documento include già un testo copertina per la stessa copertina, precedentemente aggiunto o adattato da voi o dalla stessa organizzazione nel nome della quale si agisce, non se ne può aggiungere un altro, ma si può rimpiazzare il vecchio ottenendo l'esplicita autorizzazione dall'editore precedente che aveva aggiunto il testo copertina.

L'autore/i e l'editore/i del "documento" non ottengono da questa licenza il permesso di usare i propri nomi per pubblicizzare la versione modificata o rivendicare l'approvazione di ogni versione modificata.

5. UNIONE DI DOCUMENTI

Si può unire il documento con altri realizzati sotto questa licenza, seguendo i termini definiti nella precedente sezione 4 per le versioni modificate, a patto che si includa l'insieme di tutte le Sezioni Invarianti di tutti i documenti originali, senza modifiche, e si elenchino tutte come Sezioni Invarianti della sintesi di documenti nella licenza della stessa.

Nella sintesi è necessaria una sola copia di questa licenza, e multiple sezioni invarianti possono essere rimpiazzate da una singola copia se identiche. Se ci sono multiple Sezioni Invarianti con lo stesso nome ma contenuti differenti, si renda unico il titolo di ciascuna sezione aggiungendovi alla fine e fra parentesi, il nome dell'autore o editore della sezione, se noti, o altrimenti un numero distintivo. Si facciano gli stessi aggiustamenti ai titoli delle sezioni nell'elenco delle Sezioni Invarianti nella nota di copyright della sintesi.

Nella sintesi si devono unire le varie sezioni intitolate "storia" nei vari documenti originali di partenza per formare una unica sezione intitolata "storia"; allo stesso modo si unisca ogni sezione intitolata

"Ringraziamenti", e ogni sezione intitolata "Dediche". Si devono eliminare tutte le sezioni intitolate "Riconoscimenti".

6. RACCOLTE DI DOCUMENTI

Si può produrre una raccolta che consista del documento e di altri realizzati sotto questa licenza; e rimpiazzare le singole copie di questa licenza nei vari documenti con una sola inclusa nella raccolta, solamente se si seguono le regole fissate da questa licenza per le copie alla lettera come se si applicassero a ciascun documento.

Si può estrarre un singolo documento da una raccolta e distribuirlo individualmente sotto questa licenza, solo se si inserisce una copia di questa licenza nel documento estratto e se si seguono tutte le altre regole fissate da questa licenza per le copie alla lettera del documento.

7. RACCOGLIERE INSIEME A LAVORI INDIPENDENTI

Una raccolta del documento o sue derivazioni con altri documenti o lavori separati o indipendenti, all'interno di o a formare un archivio o un supporto per la distribuzione, non è una "versione modificata" del documento nella sua interezza, se non ci sono copyright per l'intera raccolta. Ciascuna raccolta si chiama allora "aggregato" e questa licenza non si applica agli altri lavori contenuti in essa che ne sono parte, per il solo fatto di essere raccolti insieme, qualora non siano però loro stessi lavori derivati dal documento.

Se le esigenze del Testo Copertina della sezione 3 sono applicabili a queste copie del documento allora, se il documento è inferiore ad un quarto dell'intero aggregato i Testi Copertina del documento possono essere piazzati in copertine che delimitano solo il documento all'interno dell'aggregato. Altrimenti devono apparire nella copertina dell'intero aggregato.

8. TRADUZIONI

La traduzione è considerata un tipo di modifica, e di conseguenza si possono distribuire traduzioni del documento seguendo i termini della sezione 4. Rimpiazzare sezioni non modificabili con traduzioni richiede un particolare permesso da parte dei detentori del diritto d'autore, ma si possono includere traduzioni di una o più sezioni non modificabili in aggiunta alle versioni originali di queste sezioni immutabili. Si può fornire una traduzione della presente licenza a patto che si includa anche l'originale versione inglese di questa licenza. In caso di discordanza fra la traduzione e l'originale inglese di questa licenza la versione originale inglese prevale sempre.

9. TERMINI

Non si può applicare un'altra licenza al documento, copiarlo, modificarlo, o distribuirlo al di fuori dei termini espressamente previsti da questa licenza. Ogni altro tentativo di applicare un'altra licenza al documento, copiarlo, modificarlo, o distribuirlo è deprecato e pone fine automaticamente ai diritti previsti da questa licenza. Comunque, per quanti abbiano ricevuto copie o abbiano diritti coperti da questa licenza, essi non ne cessano se si rimane perfettamente coerenti con quanto previsto dalla stessa.

10. REVISIONI FUTURE DI QUESTA LICENZA

La Free Software Foundation può pubblicare nuove, rivedute versioni della Gnu Free Documentation License volta per volta. Qualche nuova versione potrebbe essere simile nello spirito alla versione attuale ma differire in dettagli per affrontare nuovi problemi e concetti. Si veda <http://www.gnu.org/copyleft>.

Ad ogni versione della licenza viene dato un numero che distingue la versione stessa. Se il documento specifica che si riferisce ad una versione particolare della licenza contraddistinta dal numero o "ogni versione successiva", si ha la possibilità di seguire termini e condizioni sia della versione specificata che di ogni

versione successiva pubblicata (non come bozza) dalla Free Software Foundation. Se il documento non specifica un numero di versione particolare di questa licenza, si può scegliere ogni versione pubblicata (non come bozza) dalla Free Software Foundation.

Come usare questa licenza per i vostri documenti

Per applicare questa licenza ad un documento che si è scritto, si includa una copia della licenza nel documento e si inserisca il seguente avviso di copyright appena dopo la pagina del titolo:

```
Copyright (c) ANNO VOSTRO NOME.  
È garantito il permesso di copiare,  
distribuire e/o modificare questo documento  
seguendo i termini della GNU Free Documentation  
License, Versione 1.1 o ogni versione successiva  
pubblicata dalla Free Software Foundation;  
con le Sezioni Non Modificabili ELENCARNE I  
TITOLI, con i Testi Copertina ELENCO, e  
con i Testi di Retro Copertina ELENCO.  
Una copia della licenza è acclusa nella sezione  
intitolata "GNU Free Documentation License".
```

Se non ci sono Sezioni non Modificabili, si scriva "senza Sezioni non Modificabili" invece di dire quali sono non modificabili. Se non c'è Testo Copertina, si scriva "nessun Testo Copertina" invece di "il testo Copertina è ELENCO"; e allo stesso modo si operi per il Testo di Retro Copertina.

Se il vostro documento contiene esempi non banali di programma in codice sorgente si raccomanda di realizzare gli esempi contemporaneamente applicandovi anche una licenza di software libero di vostra scelta, come ad esempio la GNU General Public License, al fine di permetterne l'uso come software libero.

Ritorna alla [pagina principale di GNU](#).

Per informazioni e domande sulla FSF e GNU rivolgersi, possibilmente in inglese, a gnu@gnu.org. Altri [modi per contattare](#) la FSF.

Commenti su queste pagine web a webmasters@gnu.org, altre domande a gnu@gnu.org.

Copyright (C) 1996, 1997, 1998, 1999, 2000 Free Software Foundation, Inc., 59 Temple Place – Suite 330, Boston, MA 02111, USA

La copia letterale e la distribuzione di questo articolo nella sua integrità sono permesse con qualsiasi mezzo, a condizione che questa nota sia riprodotta.

Aggiornato: 20 Settembre 2000 Andrea Ferro, Leandro Noferini e Franco Vite.
