

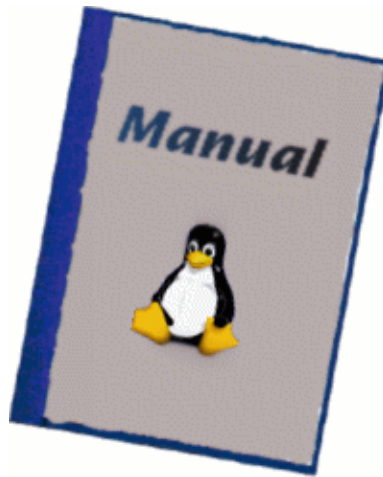
Scrivere pagine man



by Guido Socher ([homepage](#))

About the author:

A Guido piace Linux perché è molto flessibile e offre molte più possibilità di qualsiasi altro sistema operativo.



Abstract:

Ogni buon programma che possa essere usato dalla linea di comando di UNIX dovrebbe essere documentato nella sua man-page. Questo tutorial vi darà una veloce introduzione sulla scrittura delle pagine man.

Introduzione

La documentazione è spesso più importante del software stesso, specialmente se il software non deve essere usato dal suo autore. Anche se scrivo un programma che non voglio pubblicare, scrivo la sua documentazione perché un paio di mesi dopo potrei essermi dimenticato come usarlo e una buona documentazione strutturata mi dirà in pochi secondi come funziona.

Le utility a linea di comando tradizionali di Linux sono sempre state documentate con delle man-page. Un semplice `man comando` vi dirà come usare il comando.

Il vantaggio delle pagine man sulle altre forme di documentazione è che

1. Possono essere viste in pochi secondi da qualsiasi terminale
2. Possono essere facilmente convertite in altri formati: HTML, PDF, Postscript, testo, ...
3. Le pagine man possono essere viste non solo nelle finestre di terminale, ma anche in altri programmi come konqueror (scrivete semplicemente `man:comando`)

Le sezioni

Le pagine man sono strutturate in sezioni. Esattamente come un libro è strutturato in capitoli. Ci sono, per esempio, due pagine man per printf. Una per la funzione della libreria C (sezione 3) e una per il comando shell printf (sezione 1):

```
> whichman -0 printf
/usr/share/man/man1/printf.1.bz2
/usr/share/man/man3/printf.3.bz2
```

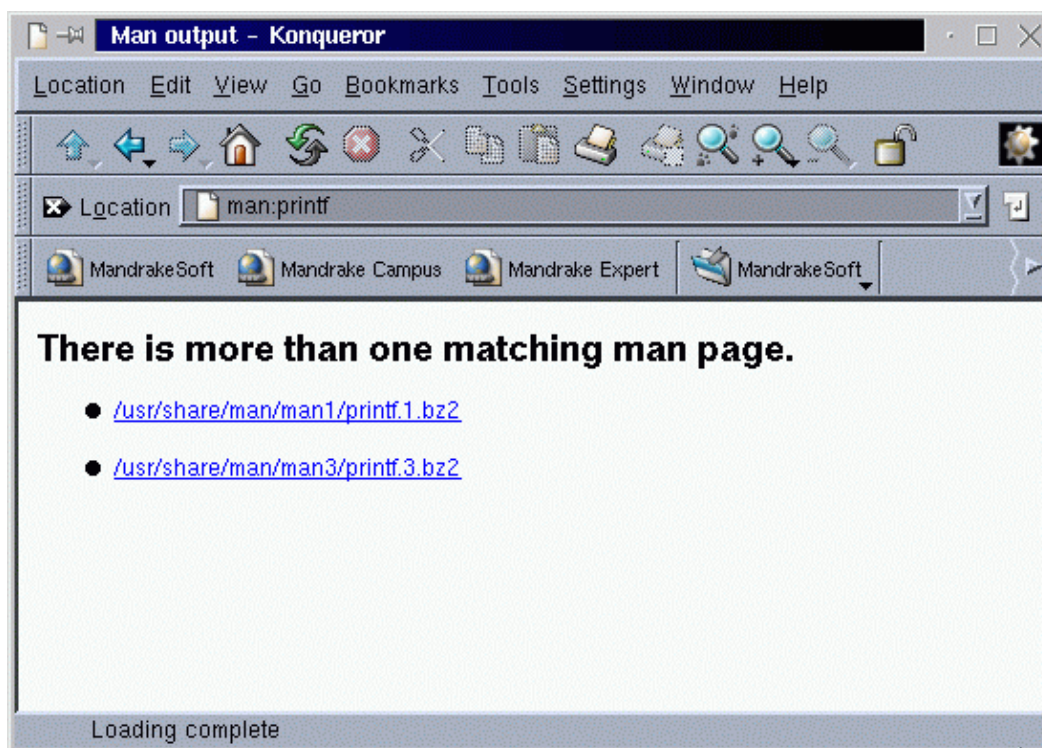
Le diverse sezioni sono:

Sezione

- 1 Comandi utente.
- 2 System call, funzioni fornite dal kernel.
- 3 Subroutine, funzioni di libreria.
- 4 Device, file speciali nella directory /dev.
- 5 Formati di file, per esempio /etc/passwd.
- 6 Giochi, si spiega da solo.
- 7 Miscellanea, per esempio pacchetti di macro, convenzioni, ...
- 8 Strumenti di amministrazione che solo root può usare.
- 9 Altro.
- n Nuova documentazione che potrebbe essere spostata nella sezione appropriata.
- l Documentazione locale per questo particolare sistema.

Quindi scrivendo "man 1 printf" vi darà la documentazione sul comando di shell printf e "man 3 printf" vi darà la descrizione della funzione della libreria C. Scrivendo semplicemente "man printf" vi darà la prima trovata (di solito dalla sezione 1).

Per controllare se ci sono più versioni delle pagine man potete sempre usare il comando whichman come mostrato sopra (scaricatelo dalla [pagina di Guido Sochere](#) o scrivete "man:printf" in konqueror e lui vi restituirà:



MANPATH

Il comando `man` cerca le pagine `man` in base al valore della variabile di ambiente `MANPATH`. Sfortunatamente molte distribuzioni Linux la impostano in modo errato. Spesso `/usr/lib/perl5/man`, che contiene molta documentazione su tutte le funzioni `perl`, non è inclusa. Potete aggiungerla al vostro `MANPATH` (in `.bashrc` o `.tcshrc` o ...) così

Bash:

```
MANPATH="/usr/local/man:/usr/man:/usr/share/man:/usr/X11R6/man:/usr/lib/perl5/man"
export MANPATH
```

Tcsh:

```
setenv MANPATH "/usr/local/man:/usr/man:/usr/share/man:/usr/X11R6/man:/usr/lib/perl5/man"
```

Dopo aver impostato il path potete usare "`man Pod::Man`" per vedere se riuscite a vedere una della pagine di manuale di `Perl`.

Le parole chiave di formattazione

Scrivere una pagina di manuale è molto semplice. Si tratta di un semplice linguaggio di `mark-up` dove le parole per il markup iniziano con un punto a inizio riga. Queste parole vengono anche chiamate macro. Le macro più importanti sono:

```
.TH -> Inizia il titolo/intestazione della pagina
.SH -> Titolo di sezione
.PP -> Nuovo paragrafo
." -> Linea di commento
.TP -> Indenta il testo che viene 2 righe dopo questa macro
```

La sintassi di `.TH` è:

```
.TH [nome del programma] [numero di sezione] [pié di pagina centrale] [pié di pagina sinistro] [intestazione centrale]
```

La sintassi di `.SH` è:

```
.SH testo dell'intestazione
```

La sintassi di `.PP` è molto semplice. Causa semplicemente un `fine-riga`.

A volte trovo comodo includere testo `pre-formattato` per gli esempi di codice. Può essere fatto così:

```
.nf
_il_vostro_testo_preformattato
_va_qui_____
.fi
```

Notate che sono macro `groff/nroff` e quindi non appartengono alle macro speciali per le pagine `man`. Sembrano comunque lavorare bene su tutti i sistemi Unix.

Tutte le macro di formattazione per le pagine man sono documentate nella man-page `groff_man(7)` ([Clicka qui per vedere una versione html della man-page di `groff_man\(7\)`](#)). Non spiegherò qui queste macro, mentre vi suggerisco di leggere la man page di `groff_man`, che è molto dettagliata e contiene tutto quello che dovete sapere.

I capitoli

Prima di iniziare a scrivere le nostre man-page dovete sapere che sono normalmente strutturate in capitoli. Per convenzione i possibili capitoli sono i seguenti:

NAME	Sezione del nome, il nome della funzione o del comando.
SYNOPSIS	Usò del comando.
DESCRIPTION	Descrizione generale
OPTIONS	Dovrebbe includere opzioni e parametri.
RETURN VALUES	Chiamate di funzione delle sezioni 2 e 3.
ENVIRONMENT	Descrive le variabili d'ambiente.
FILES	File associati con l'oggetto.
EXAMPLES	Esempi e suggerimenti.
DIAGNOSTICS	Normalmente usato per la diagnostica dei device della sezione 4.
ERRORS	Gestione dei segnali e degli errori nelle sezioni 2 e 3
SEE ALSO	Riferimenti incrociati e citazioni
STANDARDS	Conformità agli standard, se applicabile
BUGS	Errori e cavilli.
SECURITY CONSIDERATIONS	Argomenti di sicurezza di cui essere a conoscenza.
other	Capitoli personalizzati possono essere aggiunti dagli autori a loro discrezione

Una manpage di esempio

Questa è una semplice man-page di esempio. Notate che `\-` viene richiesto per diversificare il trattino dagli a-capo. Scrivete tutto dentro il vostro editor e salvatelo come `cdspeed.1`

```
.TH cdspeed 1 "September 10, 2003" "version 0.3" "USER COMMANDS"
.SH NAME
cdspeed \- decrease the speed of you cdrom to get faster access time
.SH SYNOPSIS
.B cdspeed
[\-h] [\-d device] \-s speed
.SH DESCRIPTION
Modern cdrom drives are too fast. It can take several seconds
on a 60x speed cdrom drive to spin it up and read data from
the drive. The result is that these drives are just a lot slower
than a 8x or 24x drive. This is especially true if you are only
occasionally (e.g every 5 seconds) reading a small file. This
utility limits the speed and makes the drive more responsive
when accessing small files.
.PP
cdspeed makes the drive also less noisy and is very useful if
you want to listen to music on your computer.
.SH OPTIONS
.TP
\-\h
display a short help text
.TP
```

```

\ -d
use the given device instead of /dev/cdrom
.TP
\ -s
set the speed. The argument is a integer. Zero means restore maximum
speed.
.SH EXAMPLES
.TP
Set the maximum speed to 8 speed cdrom:
.B cdspeed
\ -s 8
.PP
.TP
Restore maximum speed:
.B cdspeed
\ -s 0
.PP
.SH EXIT STATUS
cdspeed returns a zero exist status if it succeeds to change to set the
maximum speed of the cdrom drive. Non zero is returned in case of failure.
.SH AUTHOR
Guido Socher (guido (at) linuxfocus.org)
.SH SEE ALSO
eject(1)

```

Clickate [qui \(cdspeed.html\)](#) per vedere la pagina sopra descritta.

Vedere e formattare le vostre man-page

Mentre scrivete una pagina man dovreste, di tanto in tanto, vederla per controllare che appaia bene. Scrivete:

```
nroff -man la_vostra_manpage.1 | less
```

o

```
groff -man -Tascii la_vostra_manpage.1 | less
```

Per convertire una man-page in testo pre-formatato (per esempio per controllare la sintassi) usate:

```
nroff -man la_vostra_manpage.1 | col -b > xxxx.txt
```

Per convertirla in Postscript (per stamparla o riconvertirla in PDF) usate:

```
groff -man -Tps la_vostra_manpage.1 > la_vostra_manpage.ps
```

Per convertirla in html usate:

```
man2html la_vostra_manpage.1
```

Usare Perl POD per generare le manpage

So che molte persone ritengono strano editare le manpage con un text-editor. Vogliono generarle. Il formato di documentazione Perl POD è una buona scelta. Potete scrivere la pagina in formato POD e lanciare il comando

```
pod2man la_vostra_manpage.pod > la_vostra_manpage.1
```

La sintassi del linguaggio di documentazione Perl POD è descritta in una pagina man di nome perlpod. La pagina di esempio precedente in formato pod somiglia a quella mostrata sotto. Notate che POD è sensibile agli spazi e che le linee vuote attorno a "=head" non necessarie.

```
=head1 NAME
```

```
cdspeed - decrease the speed of you cdrom to get faster access time
```

```
=head1 SYNOPSIS
```

```
cdspeed [-h] [-d device] -s speed
```

```
=head1 DESCRIPTION
```

```
Modern cdrom drives are too fast. It can take several seconds on a 60x speed cdrom drive to spin it up and read data from the drive. The result is that these drives are just a lot slower than a 8x or 24x drive. This is especially true if you are only occasionally (e.g every 5 seconds) reading a small file. This utility limits the speed and makes the drive more responsive when accessing small files.
```

```
cdspeed makes the drive also less noisy and is very useful if you want to listen to music on your computer.
```

```
=head1 OPTIONS
```

```
B<-h> display a short help text
```

```
B<-d> use the given device instead of /dev/cdrom
```

```
B<-s> set the speed. The argument is a integer. Zero means restore maximum speed.
```

```
=head1 EXAMPLES
```

```
Set the maximum speed to 8 speed cdrom:
```

```
cdspeed -s 8
```

```
Restore maximum speed:
```

```
cdspeed -s 0
```

```
=head1 EXIT STATUS
```

```
cdspeed returns a zero exist status if it succeeds to change to set the maximum speed of the cdrom drive. Non zero is returned in case of failure.
```

```
=head1 AUTHOR
```

```
Guido Socher
```

```
=head1 SEE ALSO
```

```
eject(1)
```

Riferimenti

- Man-page HOWTO
- groff man(7), man-page macros

Webpages maintained by the LinuxFocus Editor team

© Guido Socher

"some rights reserved" see linuxfocus.org/license/
<http://www.LinuxFocus.org>

Translation information:

en --> -- : Guido Socher ([homepage](#))

en --> it: Alessandro Pellizzari <alex/at/neko.it>

2005-01-10, generated by lfparsr_pdf version 2.51