



IV Giornata Nazionale di Linux e del Software Libero

2004

# Installare programmi con Linux

A cura di

**isazi** <[isazi@olografix.org](mailto:isazi@olografix.org)>

[OPENLUG.ORG](http://OPENLUG.ORG)

## Indice: di cosa parleremo oggi ?

- Pacchetti o Sorgenti ?
- Installazione di software pacchettizzato
  - .rpm
  - .deb
  - .tgz
- Installazione di software non pacchettizzato
  - ./configure && make && make install
- Creiamo i nostri pacchetti
  - checkinstall

## Pacchetti o Sorgenti ?

- Nella maggior parte dei casi, durante l'installazione di una distribuzione linux ci viene chiesto (più o meno nello specifico) quali pacchetti software o quali tipologie di questi vogliamo installare
- Se successivamente vogliamo installare altri programmi non dobbiamo ripetere l'intero processo di installazione, ci basterà infatti scegliere tra una di queste due opzioni:
  - Installare software pacchettizzato per la nostra distribuzione, fornito sui cd di installazione o reperito in altro modo (es. internet)
  - Prendere i sorgenti del pacchetto che ci interessa e “compilarcelo” a partire da questi

## Pacchetti o Sorgenti ? (2)

- La scelta sarà naturalmente dettata da pochi (e personali) fattori:
  - La nostra esperienza con linux
  - La disponibilità o meno di un pacchetto precompilato
  - Nostre necessità particolari
- Di solito installare un pacchetto è sicuramente più semplice e porta via meno tempo; inoltre il pacchetto è sicuramente stato testato da altri e possiamo considerarlo “affidabile”
- Tuttavia compilare i sorgenti ci permette notevoli personalizzazioni e la possibilità di incrementare le prestazioni del software che andremo ad eseguire

## Installazione di software pacchettizzato: RPM

- Cominciamo con l'installazione e la gestione di pacchetti RPM (RedHat Package Manager)
- I pacchetti RPM sono stati inizialmente introdotti da Red Hat, e sono oggi utilizzati oltre che su RedHat e Fedora dalle loro derivazioni e da distribuzioni terze (SuSE o Mandrake ad esempio) che hanno scelto di non sviluppare un proprio sistema di pacchettizzazione
- Informazioni precise su questo formato possono essere reperite su <http://www.rpm.org>

## Installazione di software pacchettizzato: RPM (2)

- Installazione di un pacchetto:

```
rpm -ivh pacchetto.rpm
```

- Rimozione di un programma installato:

```
rpm -e nome_pacchetto
```

- Aggiornamento di un pacchetto:

```
rpm -Uvh pacchetto.rpm
```

- Qualche feature:

- Possiamo installare anche pacchetti remoti specificando il percorso remoto invece che il semplice nome del pacchetto
- RPM gestisce le dipendenze dei pacchetti

## Installazione di software pacchettizzato: DEB

- La distribuzione Debian e le sue derivate (Knoppix, etc.) utilizzano il formato DEB
- Installazione di un pacchetto e aggiornamento:  

```
dpkg -i pacchetto.deb
```
- Rimozione di un programma installato:  

```
dpkg -r nome_pacchetto (conserva configurazioni e altri file)
```

```
dpkg --purge nome_pacchetto (cancella tutto)
```
- La gestione dei pacchetti in Debian e nelle sue derivate, comunque, viene effettuata ormai attraverso un gestore dei pacchetti più avanzato: APT

## Installazione di software pacchettizzato: DEB (2)

- APT (Advanced Package Tool) è semplicemente un insieme di strumenti che ci permette di utilizzare con più semplicità i pacchetti DEB, scaricando i pacchetti per noi da fonti remote, risolvendo le dipendenze, etc.
- Le fonti vanno indicate nel file **/etc/apt/sources.list**
- Installare un pacchetto:  

```
apt-get install nome_pacchetto
```
- Aggiornamento lista pacchetti disponibili:  

```
apt-get update
```
- Aggiornamento dei programmi installati:  

```
apt-get upgrade
```



## Installazione di software pacchettizzato: DEB (3)

- Aggiornamento dell'intera distribuzione:

```
apt-get dist-upgrade
```

- Rimozione di un programma installato:

```
apt-get remove nome_pacchetto
```

- APT è stato ritenuto un sistema di gestione dei pacchetti talmente avanzato :) che ne è stata sviluppata una versione anche per gestire pacchetti RPM

## Installazione di software pacchettizzato: TGZ

- I pacchetti utilizzati da Slackware sono degli archivi compressi contenenti l'albero di installazione e uno script di installazione che crea i link, allinea i permessi et similia
- Questo sistema è indubbiamente il meno raffinato dei tre che stiamo descrivendo, ma ben si adatta allo spirito della distribuzione che lo utilizza

- Installazione di un pacchetto:

```
installpkg pacchetto.tgz
```

- Rimozione di un programma installato:

```
removepkg nome_pacchetto
```

- Aggiornamento di un pacchetto:

```
upgradepkg pacchetto.tgz
```

## Installazione di software pacchettizzato: TGZ (2)

- Slackware non dispone nativamente di un tool come apt, ma tra gli “extras” della distribuzione è presente **swaret**

- Installazione di un pacchetto:

```
swaret --install nome_pacchetto
```

- Rimozione di un programma installato:

```
swaret --remove nome_pacchetto
```

- Aggiornamento lista pacchetti disponibili:

```
swaret --update
```

- Aggiornamento dei programmi installati:

```
swaret --upgrade
```

## Installazione di software non pacchettizzato

- Se non disponiamo del pacchetto necessario, o se vogliamo comunque personalizzare il software che andremo ad usare, possiamo compilarlo ed installarlo a partire dai sorgenti
- Naturalmente faremo a meno di tutte le comodità che molti tool offrono per la gestione delle dipendenze che dovremo andare a risolvere da soli
- La cosa migliore da fare è leggere il file INSTALL che accompagna quasi tutti i pacchetti, ci darà un'idea della procedura da seguire e delle dipendenze da soddisfare prima di poter riuscire nella nostra impresa

## Installazione di software non pacchettizzato (2)

- Dopo aver letto il file INSTALL (o un buon surrogato) e aver risolto le eventuali dipendenze siamo pronti a compilare il nostro programma
- Partiamo dal presupposto che stiamo compilando un programma che, anche se distribuito sotto forma di archivio di sorgenti, abbia un minimo di “struttura” e la cui installazione sia stata pensata per facilitarci la vita
- Se è presente nella directory ottenuta decomprimendo l'archivio eseguiamo il file **configure**, eventualmente con le dovute opzioni:

```
./configure --help
```

```
./configure
```

## Installazione di software non pacchettizzato (3)

- L'esecuzione dello script chiamato “configure” ci serve per creare un **Makefile**, un file dove sono elencate le istruzioni per il programma **make** che si occupa di eseguire i passaggi necessari alla compilazione nel giusto ordine
- Andiamo col make:  

```
make
```
- E per l'installazione (se abbiamo i privilegi di root):  

```
make install
```
- A questo punto dovremmo avere disponibile ed installato il nostro programma, a meno di eventuali errori che comunque dovrebbero esserci stati segnalati :)

## Creiamo i nostri pacchetti!

- Per ovviare a qualche simpatico inconveniente, come per esempio programmi dall'installazione invasiva che non possiedono una procedura di rimozione, o per unire i vantaggi (o gli svantaggi se Murphy ci assiste) dei due metodi che abbiamo presentato per l'installazione dei programmi, potremmo compilare tutto a partire dai sorgenti e da questi crearci un pacchetto personalizzato
- Umh, ma non sarà un lavoro troppo da sviluppatori ? No! (se ci “aiutiamo” con qualche script già fatto, ovviamente) :)
- Per prima cosa compiliamo il programma come descritto precedentemente, ma fermiamoci prima del “make install”...

## Creiamo i nostri pacchetti! (2)

- Per realizzare i nostri pacchetti utilizzeremo **checkinstall** al posto dell'installazione con “make install”
- L'utilizzo più semplice è questo:
  - `checkinstall -S <opzioni>` (per pacchetti slackware)
  - `checkinstall -R <opzioni>` (per pacchetti rpm)
  - `checkinstall -D <opzioni>` (per pacchetti deb)
- Le opzioni base, che sono comuni per tutti e tre i formati, sono:
  - `--pkgname=nome_programma`
  - `--pkgversion=versione_programma`
  - `--pkgarch=architettura_di_compilazione`
  - `--pkgrelease=release_del_pacchetto`



## Creiamo i nostri pacchetti! (3)

- Dovremmo avere, a questo punto, il nostro bel pacchetto gestibile con gli strumenti messi a disposizione dalla nostra distribuzione per il formato che abbiamo scelto
- Occhio però che così non abbiamo certo creato un pacchetto come quelli ufficiali, questi pacchetti creati con checkinstall andranno bene per noi e, se abbiamo un sistema molto simile, per qualche nostro amico, non andranno bene per essere distribuiti in rete
- Gestire la propria macchina in questo modo, comunque, permette di mantenere una notevole pulizia, laddove con la compilazione selvaggia si andava a perdere soprattutto questo, senza rinunciare ad una sana ottimizzazione :D

# URL

- Pacchetti RPM:  
<http://www.rpmfind.net>
- Pacchetti DEB:  
<http://www.apt-get.org>
- Pacchetti TGZ:  
<http://www.linuxpackages.net>
- Sorgenti:  
<http://freshmeat.net>

The end

# Domande ?