

Gestione del Software

Installazione e gestione repository

Niccolò Izzo

izzo.niccolo@gmail.com



POLITECNICO OPEN
unix LABS

Come hack with us.

Benvenuti

Qui trovate queste slide

software.izzo.ovh

Fate tante domande!

Oggi vedremo come

Oggi vedremo come

Installare il Software

Oggi vedremo come

Installare il Software

Gestire i repository

Compilazione

Compilazione

Pro

- Codice ottimizzato e personalizzato
- Possibile il controllo sui sorgenti

Compilazione

Pro

- Codice ottimizzato e personalizzato
- Possibile il controllo sui sorgenti

Contro

- Lenta per software complessi
- Non tiene traccia del software

Compilazione

Pro

- Codice ottimizzato e personalizzato
- Possibile il controllo sui sorgenti

Contro

- Lenta per software complessi
- Non tiene traccia del software

Anche se in alcuni casi è l'unica possibilità.

I pacchetti precompilati

I pacchetti precompilati

Principalmente sono due:

- .rpm per RHEL, Fedora e le loro derivate
- .deb per Debian, Ubuntu e le loro derivate

I pacchetti precompilati

Principalmente sono due:

- .rpm per RHEL, Fedora e le loro derivate
- .deb per Debian, Ubuntu e le loro derivate

Ma esistono anche

- .tgz per Slackware
- .pkg.tar.xz per Arch Linux

Le operazioni base

Per installare un pacchetto

Debian based

```
# dpkg -i pacchetto.deb
```

RPM based

```
# rpm -i pacchetto.rpm
```

Le operazioni base

Per rimuovere un pacchetto

Debian based

```
# dpkg -r pacchetto
```

RPM based

```
# rpm -r pacchetto
```

Manca ancora qualcosa...

Manca ancora qualcosa...

Ora l'installazione non richiede la compilazione,
i programmi installati sono facilmente rimovibili.

Manca ancora qualcosa...

Ora l'installazione non richiede la compilazione,
i programmi installati sono facilmente rimovibili.

- Le dipendenze vanno soddisfatte manualmente
- I binari sono specifici per ogni architettura
- Non esistono aggiornamenti automatici

I Package Manager

Oggi sono diffusi tra vari OS e dispositivi.



I Package Manager su Linux

I Package Manager su Linux

Apt e Aptitude per gestire i .deb

Yum per gestire i pacchetti .rpm

I Package Manager su Linux

Apt e Aptitude per gestire i .deb

Yum per gestire i pacchetti .rpm

- Contengono moltissimi software utili
- Il software è verificato (da chi gestisce i repo)
- Le dipendenze vengono soddisfatte autonomamente
- Gli aggiornamenti sono automatici e rapidi

I Repository

Archivi web in cui sono raggruppati i software installabili
Ogni distribuzione ha i suoi e ne esistono di non ufficiali

Ubuntu 14.04~ 46000 pacchetti

Debian Sid~ 45000 pacchetti

Fedora 21~ 40000 pacchetti

Considerando i pacchetti precompilati per amd64.

Apt

Apt

Advanced Packaging Tool

Apt

Advanced Packaging Tool

É il package manager di Debian.

Ereditato da tutte le sue derivate (Ubuntu, Mint)

Apt

Advanced Packaging Tool

É il package manager di Debian.

Ereditato da tutte le sue derivate (Ubuntu, Mint)

Si può usare anche Aptitude: ha l'interfaccia grafica!

Apt - Comandi base

Per prima cosa aggiorniamo il database dei pacchetti

```
$ apt-get update
```

Ricerca nel database

```
$ apt-cache search keyword
```

Installazione di un pacchetto

```
# apt-get install pacchetto
```

Rimozione di un pacchetto

```
# apt-get remove pacchetto
```

Apt - Aggiornamento

Aggiornamento semplice dei pacchetti installati

```
# apt-get upgrade
```

Aggiornamento completo dei pacchetti installati

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

Il comando dist-upgrade aggiorna anche i pacchetti che richiedono l'installazione o la rimozione di altri pacchetti.

Apt - Automatic Security Updates

É possibile configurare apt per installare in modo automatico gli aggiornamenti.

Utile per ottenere gli aggiornamenti di sicurezza appena vengono rilasciati.

Guida dettagliata nelle wiki di [Ubuntu](#) e [Debian](#).

Apt - Altri comandi

Più informazioni su un pacchetto

```
$ apt-cache show pacchetto
```

Pulisce tutta la cache contenuta in `/var/cache/apt`

```
# apt-get clean
```

Rimuove un pacchetto e i suoi file di configurazione

```
# apt-get purge pacchetto
```

Elimina i pacchetti non più necessari al sistema

```
# apt-get autoremove
```

Apt - ora in Tecnicolor!

Per tutti i comandi precedenti possiamo sostituire le keyword "apt-get", "apt-cache" e "apt-file" con "apt".

Ma alcuni comandi sono leggermente diversi:

```
# apt upgrade
```

aggiorna i pacchetti anche se richiedono l'installazione ma non la rimozione di pacchetti aggiuntivi.

```
# apt full-upgrade
```

Svolge la funzione del precedente "dist-upgrade".

Apt - Repository

/etc/apt/sources.list

```
# See http://help.ubuntu.com/community/UpgradeNotes for how to upgrade to
# newer versions of the distribution.
deb http://archive.ubuntu.com/ubuntu trusty main
deb-src http://archive.ubuntu.com/ubuntu trusty main

## Major bug fix updates produced after the final release of the
## distribution.
deb http://archive.ubuntu.com/ubuntu trusty-updates main
deb-src http://archive.ubuntu.com/ubuntu trusty-updates main

## N.B. software from this repository is ENTIRELY UNSUPPORTED by the Ubuntu
## team. Also, please note that software in universe WILL NOT receive any
## review or updates from the Ubuntu security team.
deb http://archive.ubuntu.com/ubuntu trusty universe
deb-src http://archive.ubuntu.com/ubuntu trusty universe
deb http://archive.ubuntu.com/ubuntu trusty-updates universe
deb-src http://archive.ubuntu.com/ubuntu trusty-updates universe

## Uncomment the following two lines to add software from Canonical's
## 'partner' repository.
## This software is not part of Ubuntu, but is offered by Canonical and the
## respective vendors as a service to Ubuntu users.
# deb http://archive.canonical.com/ubuntu trusty partner
# deb-src http://archive.canonical.com/ubuntu trusty partner

deb http://security.ubuntu.com/ubuntu trusty-security main
deb-src http://security.ubuntu.com/ubuntu trusty-security main
deb http://security.ubuntu.com/ubuntu trusty-security universe
deb-src http://security.ubuntu.com/ubuntu trusty-security universe
# deb http://security.ubuntu.com/ubuntu trusty-security multiverse
# deb-src http://security.ubuntu.com/ubuntu trusty-security multiverse
```


Apt - Repository

Per aggiungere altre repository (di launchpad)

```
# add-apt-repository ppa:utente/nome_ppa
```

Oppure passando una stringa da inserire in sources.list

```
# add-apt-repository 'deb uri_distribution [component] [component]'
```

Altrimenti possiamo modificare a mano il file sources.list ricordandoci di aggiungere le relative chiavi GPG.

Aggiorniamo sempre il database dei pacchetti dopo ogni modifica ai repository.

Apt - Demo

DEMO!

YUM

Yellow dog Updater, Modified

E' il package manager per le distribuzioni basate su rpm
si trova incluso di default in Fedora, RHEL, CentOS

A differenza di apt, aggiorna da solo il suo DB dei pacchetti

YUM - Comandi base

Installazione di un pacchetto

```
# yum install pacchetto
```

Ricerca nel database dei pacchetti

```
$ yum search keyword
```

Rimozione di un pacchetto (come apt-get purge)

```
# yum remove pacchetto
```

Più informazioni su un pacchetto

```
$ yum info pacchetto
```

YUM - Aggiornamento

Aggiornamento dei pacchetti installati

```
# yum update
```

Sincronizzazione con i repository

```
# yum distro-sync
```

Pulizia della cache pacchetti

```
$ yum clean packages
```

YUM - AutoUpdates

Anche YUM supporta gli aggiornamenti automatici.

Ma non è necessariamente la scelta giusta abilitarli.

Trovate una guida completa nella [wiki di Fedora](#).

YUM - Repository

/etc/yum/repos.d/fedora.repo

```
[fedora]
name=Fedora $releasever - $basearch
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/releases/$releasever/Everything/$basearch/os/
metalink=https://mirrors.fedoraproject.org/metalink?repo=fedora-$releasever&arch=$basearch
enabled=1
metadata_expire=7d
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$releasever-$basearch
skip_if_unavailable=False

[fedora-debuginfo]
name=Fedora $releasever - $basearch - Debug
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/releases/$releasever/Everything/$basearch/debug/
metalink=https://mirrors.fedoraproject.org/metalink?repo=fedora-debug-$releasever&arch=$basearch
enabled=0
metadata_expire=7d
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$releasever-$basearch
skip_if_unavailable=False

[fedora-source]
name=Fedora $releasever - Source
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/releases/$releasever/Everything/source/SRPMS/
metalink=https://mirrors.fedoraproject.org/metalink?repo=fedora-source-$releasever&arch=$basearch
enabled=0
metadata_expire=7d
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$releasever-$basearch
skip_if_unavailable=False
```

YUM - Repository

Per aggiungere il file .repo da web

```
# yum-config-manager --add-repo url_repository
```

Quindi verifichiamo i repository presenti nel sistema

```
$ yum repolist all
```

Abilitiamo il repository appena inserito

```
# yum-config-manager --enable nome_repository
```

Per disabilitarlo...banalmente

```
# yum-config-manager --disable nome_repository
```


YUM - Demo

DEMO!

Grazie per l'attenzione.

Fine

Queste slides sono distribuite sotto [licenza MIT](#).