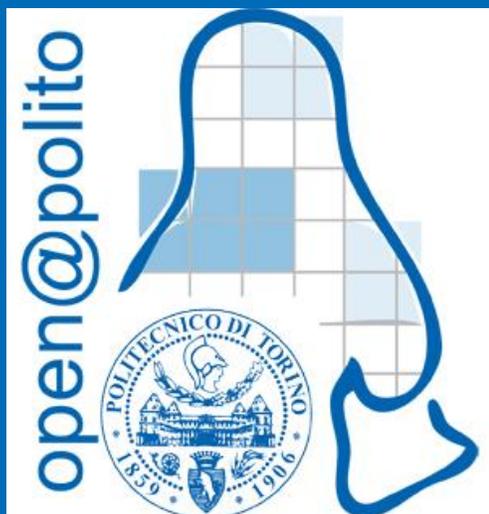




Con il supporto di:



Gestione pacchetti, utenti, permessi — The Linux Way

Rosario Antoci

08/12/2021

Installare <qualcosa>, come

- Compilare il codice sorgente
- Package manager back-end (dpkg/rpm)
- Package manager front-end (apt,yum (o dnf))

Compilare il codice sorgente

Pro

- Versione up-to-date
- Patching capillare
- Maggiori skill da devops

Contro

- Repo non centralizzati
- Maintenance difficoltosa
- Possibili conflitti con lo stack applicativo/sistemistico

Package manager (back-end)

Scaricare il pacchetto precompilato per l'OS (.deb/.rpm...) e installarlo con dpkg / rpm

Pro

- Non occorre compilare
- Maggiore disponibilita' di app non inserite ufficialmente nelle distro

Contro

- Repo non centralizzati
- Maintenance difficoltosa
- Possibili conflitti con lo stack applicativo/sistemistico

Package manager (front-end)

Installare pacchetto in modo centralizzato con apt/yum (o dnf)

Pro

- Non occorre compilare
- Ambiente stabile
- Gestione centralizzata

Contro

- Meno capillarita' (patching post-installazione)
- Meno versioni disponibili / app mancanti

Altri package manager

Sistema Operativo	Package manager
SUSE like	YaST/Zypper/RPM-based
Arch Linux	Pacman
Gentoo	Portage

Focalizzandoci sulle distribuzioni Debian-like, da segnalare anche Aptitude e Synaptic

Link per comparativa APT vs Aptitude

<https://www.tecmint.com/difference-between-apt-and-aptitude/>

Per chi vuole saperne di piu' sul package management di Arch Linux e SUSE:

- Arch Linux → <https://wiki.archlinux.org/index.php/Pacman>
- SUSE → YaST: <https://en.opensuse.org/Portal:YaST>
→ Zypper: <https://en.opensuse.org/Portal:Zypper>

Gestione permessi in Linux

- Gestione permessi:
 - Modalità assoluta
 - Modalità simbolica

`chmod` è il comando per definire i permessi di un file.

`chown` è il comando per definire la proprietà di un file (user owner e group owner)

Number	Permission
0	—
1	-x
2	-w-
3 (i.e. 2+1)	-wx
4	r—
5 (i.e. 4+1)	r-x
6 (i.e. 4+2)	rw-
7 (i.e. 4+2+1)	rwX

Modalita' assoluta

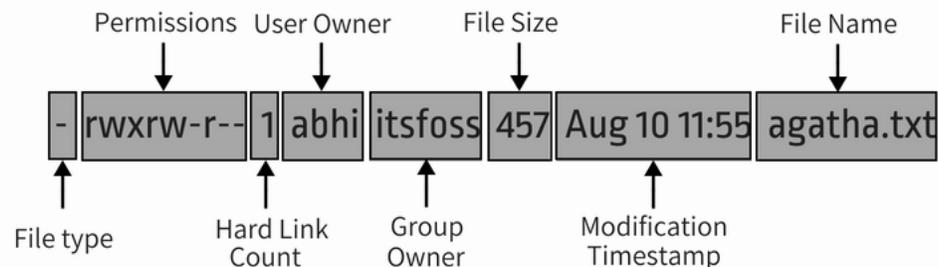
- E' il metodo standard, nonche' quello piu' complete per comprendere le dinamiche delle permission su sistemi Unix

Base ottale:

- User owner
- Group owner
- Others

Da tenere a mente:

- r (read) = 4
- w (write) = 2
- x (execute) = 1
- (no permission) = 0



Modalita' simbolica

- u = user owner
- g = group owner
- o = other
- a = all (user + group + other)

- + for adding permissions
- – for removing permissions
- = for overriding existing permissions with new value

Ex.

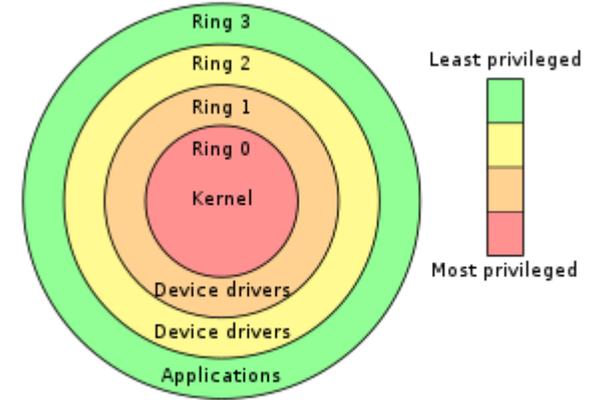
```
chmod g+x <file>
```

```
chmod g=rw <file>
```

Gestione utenti in Linux

- Utenti locali:
 - Permessi di super utente
 - Creazione, modifica, eliminazione
 - Cambio password

su e sudo: da grandi poteri...



Chi è lui? → cosa ci vuole insegnare?

su e' il comando che ci fara' loggare, se non lo siamo gia' come root.

Usare sudo permette ad un utente definito (direttamente e/o indirettamente) su /etc/sudoers di poter eseguire comandi privilegiati.

Da root: #: visudo (in automatico si creerà con l'\$EDITOR definito sulla \$SHELL uno spazio temporaneo, che una volta validato, sostituirà /etc/sudoers)

Vale il <<least privilege principle>> **PoLP**

Creazione utente

- `useradd` (o `adduser`?)

E' un binario il cui scopo e' quello di gestire la creazione di un utente ed eventuali caratteristiche, quali la shell, home directory ~, UID, gruppi di appartenenza ecc..

`adduser` invece e' o un link simbolico o uno script Perl

- `/etc/passwd`
- `/etc/shadow`
- `/etc/group`
- `/etc/gshadow`
- `/home/<home_user>`

Modifica utente

- `usermod`

E' un binario il cui scopo e' quello di gestire la modifica di un utente preesistente.

Esempio:

```
usermod -aG docker $USER
```

Domanda:

Cos'e' \$USER?

(scegliere tra il pubblico una risposta =)

Rimozione utente

- userdel *

E' un binario il cui scopo e' quello di gestire l'eliminazione di un utente preesistente.

*o deluser? =O

Ex.

userdel -f <user>

Domanda:

-f sta per?

Perche' potrebbe essere necessario usarlo?
(fai come domande precedenti)

Cambio password

- `passwd`

E' un binario il cui scopo e' quello di modificare la password di un utente preesistente.

Ex.

Da root:

```
passwd <user>
```

Extra:

- Forziamo l'utente a cambiare password con `passwd`
- Verifichiamo con `chage`

Domande / Esercitazioni

