

Linux dalla *console* — 2

Gestione utenti, pacchetti, servizi; comandi per la rete e di rete

Emiliano Vavassori

22 marzo 2017

FabLab Bergamo

Patronato San Vincenzo, Bergamo

Bergamo Linux Users Group

Demo di riepilogo

L'utente **root**

Gestione utenti

Gestione gruppi

Gestione pacchetti

Gestione servizi

Comandi per la configurazione di rete

Comandi di rete

L'utente **root**

L'utente **root**

L'utente **root** (anche *superutente*) è l'utente amministratore di sistema per eccellenza. È l'utente che tutto può nel sistema.

Per usare l'utente **root**:

```
$ su -
```

```
$ sudo -i
```

Ancora due parole su sudo...

C'è sempre la possibilità di lanciare un comando con i permessi di superutente da una shell utente:

```
$ sudo <comando>
```

```
$ su -c "<comando>"
```

Attenzione!

Fate molta attenzione ai comandi che eseguite con i permessi di *superutente*: potreste danneggiare *irrimediabilmente* il vostro sistema e perdere dati!

`sudo` può essere configurato in maniera più granulare di `su`, che si limita invece a “cambiare” utente.

Si può assegnare la possibilità di utilizzare `sudo` a qualsiasi utente tramite il gruppo `sudo` (o `wheel`, a seconda dei sistemi) o utilizzando i file di configurazione `/etc/sudoers`, `/etc/sudoers.d/*` o il comando `visudo`.

Gestione utenti

Utenti

Creazione:

```
# useradd -m <utente>
```

Assegnazione di password:

```
# passwd <utente>
```

Rimozione dell'utente:

```
# userdel -r <utente>
```

Gestione gruppi

Creazione:

```
# groupadd <gruppo>
```

Rimozione del gruppo:

```
# groupdel <gruppo>
```

Gestione appartenenza a gruppi

Aggiunta di un utente al gruppo:

```
# gpasswd -a <utente> <gruppo>
```

```
# usermod -a -G <gruppo> <utente>
```

Rimozione dell'utente dal gruppo:

```
# gpasswd -d <utente> <gruppo>
```

C'è sempre la possibilità di manipolare il file `/etc/groups`.

Gestione pacchetti

Dimmi qual è la tua distro e ti dirò chi sei...

Ogni distribuzione ha il suo sistema di *gestione dei pacchetti*, basati su standard più o meno riconosciuto e utilizzato (`.rpm`, `.deb` o `.tar.*`).

Sulla base dei vostri sistemi operativi, i comandi saranno differenti, ma i concetti spiegati sono trasversali.

Terminologia

Repository Archivio remoto di pacchetti binari catalogati

Pacchetti binari anche semplicemente *pacchetti*, sono applicazioni complete distribuite in forma binaria e con documentazione

Formato RPM formato di pacchetti binari tipico delle distribuzioni *RedHat-based*: Fedora, CentOS, OpenSUSE, Mageia

Formato DEB formato di pacchetti binari tipico delle distribuzioni *Debian-based*: Debian, *buntu, Mint, Elementary OS

Pacchetto sorgente Solitamente è un archivio in formato `.tar.gz` o `.tar.bz2` (più raramente `.zip` e `.tar.xz`): contiene il *codice sorgente* da compilare

Praticamente nulla: le distribuzioni RPM-based e DEB-based hanno storicamente un differente layout dei file pacchettizzati all'interno del filesystem del sistema.

Esiste però **alien** che, almeno, ci prova. Non consigliato per ambienti di produzione!

Sono stati sviluppati diversi formati intercompatibili, l'ultimo dovrebbe essere **Snap**. Fin'ora non hanno avuto gran successo.

Aggiornamento con apt

```
# apt update
```

```
# apt upgrade
```

```
# apt dist-upgrade
```

Installazione pacchetti

```
# apt search <qualcosa>
```

```
# apt show screen
```

```
# apt policy screen
```

```
# apt install screen vim ncftp
```

Rimozione pacchetti

```
# apt remove cmatrix
```

```
# apt purge emacs
```

```
# apt autoremove
```

Gestione pacchetti non da repository

```
# wget http://url/del/pacchetto \  
-O /tmp/pacchetto-ver.deb
```

```
# dpkg -i /tmp/pacchetto-ver.deb
```

Gestione servizi

Demoni? Chiamate l'esorcista!

I demoni (in inglese *daemons*) sono semplicemente *servizi*: processi che vengono eseguiti da una macchina (tipicamente un *server*) e che forniscono un certo tipo di funzionalità.

Alcuni demoni sono:

cron permette l'esecuzione di comandi temporizzati

cups *Common Unix Printing System*, gestisce la stampa

udev e dbus Servizi che gestiscono l'hardware e la comunicazione fra le varie periferiche e il kernel

networking "Servizio" che gestisce la parte di rete

samba Servizio per la condivisione dei file con protocollo SMB (*Server Message Block*)

ssh *Secure SHell*, permette l'accesso remoto

Avviare un servizio

```
# systemctl start ssh.service
```

```
# service sshd start
```

Fermare un servizio

```
# systemctl stop ssh.service
```

```
# service sshd stop
```

Attivare/disattivare un servizio al boot

Attivare:

```
# systemctl enable ssh.service
```

```
# update-rc.d sshd defaults
```

Disattivare:

```
# systemctl disable ssh.service
```

```
# update-rc.d sshd remove
```

Comandi per la configurazione di rete

Come si configura una rete?

Componenti fondamentali:

NIC *Network Interface Card*, scheda di rete

MAC address Anche *indirizzo fisico*, univoco per NIC (L2, *Datalink Layer*)

IP address Indirizzo “di rete”, univoco per NIC (L3, *Network Layer*)

Netmask/Prefix indica alla NIC quale dispositivi può trovare senza contattare un *gateway* (L3)

Gateway È la macchina che il dispositivo contatta se deve raggiungere host fuori dalla sua rete. Anche *default router* (L3)

Server DNS Più di uno. Permette la risoluzione dei nomi a dominio in indirizzi IP (L7, *Application Layer*)

Facciamo un esempio

NIC Realtek 8139+, nome dispositivo `eth0`

MAC address `00:E0:4C:aa:bb:cc`

IP address `192.168.1.100`

Netmask `255.255.255.0`

Prefix `/24`

Gateway `192.168.1.1`

Server DNS `8.8.8.8, 8.8.4.4`

Identificazione delle NIC

```
# ip link show
```

Attivazione/disattivazione NIC

```
# ip link set dev eth0 up
```

```
# ip link set dev eth0 down
```

Assegnazione indirizzi L3

Se esiste un DHCP server (assegnazione automatica):

```
# dhclient eth0
```

Se indirizzamento manuale:

```
# ip address add 192.168.1.100/24 dev eth0
```

```
# ip route add default via 192.168.1.1
```

Estratto: `/etc/resolv.conf`

```
nameserver 8.8.8.8
```

```
nameserver 8.8.4.4
```

Test di funzionalità

```
$ ping 192.168.1.100
```

```
$ ping 192.168.1.1
```

```
$ ping 8.8.8.8
```

```
$ ping www.google.com
```

Comandi di rete

- ping
- traceroute
- nslookup
- dig
- nmap

Il demone `ssh` è molto importante e offre, appena installato:

- Shell remota
- Autenticazione sicura (password, chiave pubblica/privata)
- Servizio di trasferimento file criptato (SFTP)
- Servizio di copia file sicuro (SCP)
- Servizio di tunneling delle connessioni

Vediamone subito una demo.