



UNIVERSITÀ
DEGLI STUDI
DI PALERMO



Shell di Linux

Installazione Vagrant e Apache

Prof. Roberto Pirrone

«Programmazione Web e Mobile»

a.a. 2021/2022

Vagrant

- Vagrant è un sistema di gestione integrata di macchine virtuali operate da diversi hypervisor (VirtualBox, VMWare, etc.) mirato alla definizione di *ambienti di sviluppo e deploy*
 - Usa appositi file di configurazione detti *Vagrantfile*
 - Installa separatamente una *vagrant* box, cioè una VM, in apposita cartella
 - Gestione di utenti, pacchetti, configurazione di rete, etc. tramite il Vagrantfile
 - La VM viene vista come una *macchina separata* sulla stessa rete del host o su una rete privata

Installazione di Vagrant

- Download e installazione di VirtualBox
 - <https://www.virtualbox.org/>
- Download e installazione di Vagrant
 - <https://www.vagrantup.com/>
- Da terminale digitare

```
$ vagrant init <nome della vagrant box>
```

```
$ vagrant up
```

 - Useremo la vagrant box *ubuntu/bionic64*

Comandi base di accesso a Vagrant

`$ vagrant up`

- Digitato nella cartella del Vagrantfile avvia la VM

`$ vagrant ssh`

- Connessione alla VM come utente vagrant che è un amministratore (sudoer) senza bisogno di password

`$ vagrant halt`

- Spegnimento della VM

Il nostro Vagrantfile di partenza

```
Vagrant.configure("2") do |config|  
  # For a complete reference, please see the online documentation at  
  # https://docs.vagrantup.com.  
  
  config.vm.box = "ubuntu/bionic64"  
  
  config.vm.network "forwarded_port", guest: 80, host: 8080  
  config.vm.network "forwarded_port", guest: 3306, host: 3306  
  
  config.vm.network "private_network", ip: "192.168.33.10"  
  config.vm.network "public_network", ip: "147.163.26.114"  
  
  config.vm.synced_folder "www/", "/home/vagrant/public_html"  
  config.vm.synced_folder "data/", "/home/vagrant/data"  
  
end
```

Introduzione minimale alla shell Linux

- I comandi della shell in Unix/Linux hanno il seguente formato generale:

```
$ <nome comando> [opzione] ... [file o cartella] ...
```

- opzione: -<lettera> oppure --<nome esteso opzione>
- Più opzioni possono essere concatenate come:
-<lettera opz1><lettera opz2>...

Introduzione minimale alla shell Linux

- Esempi di uso del comando di lista dei contenuti della cartella (directory) corrente:

```
$ ls -la
```

```
$ ls -l -a
```

```
$ ls --all -l
```

Introduzione minimale alla shell Linux

- Comandi sulle directory

\$ pwd *stampa il nome della directory corrente*

\$ cd <nome directory> *cambia directory e si porta in <nome directory>*

\$ mkdir <nome directory> *crea la directory <nome directory>*

- I percorsi dei file usano «/» come separatore delle directory
- I percorsi relativi iniziano con «.» (directory corrente) o «..» (directory superiore)

Introduzione minimale alla shell Linux

- Lista del contenuto di un file

```
$ more <nome file>
```

Interattivo da linea di comando

Barra spaziatrice → avanti

«b» → indietro

```
$ cat <nome file> non interattivo
```

```
$ less <nome file> nato dopo «more» ha molte più opzioni
```

Introduzione minimale alla shell Linux

- Ricerca di un pattern in un file

```
$ grep [-e <espressione regolare>] <nome file>
```

- Redirezione dell'output ed esecuzione in «pipeline»

```
$ ls -la > pippo.txt
```

```
$ more pippo.txt | grep --regexp=«\.[plb]»
```

Introduzione minimale alla shell Linux

- Comandi sui permessi

```
$ chmod <permessi> <file o directory>
```

Imposta i permessi di lettura/scrittura/esecuzione sul file o cartella

- Schema permessi *ugoa* (user, group, others, all)
- Modalità permessi simbolica: [ugoa][+--][r]*[w]*[x]*, ...
- Modalità permessi binaria: 755, 644, ... (r: 4, w:2, x: 1)

Introduzione minimale alla shell Linux

- Comandi sui permessi

```
$ chown [utente][:gruppo] [-R] <file o directory>
```

Cambia (ricorsivamente) il proprietario e il gruppo della risorsa

```
$ chgrp gruppo [-R] <file o directory>
```

Cambia (ricorsivamente) il gruppo della risorsa

Introduzione minimale alla shell Linux

- Comandi di copia, spostamento, cancellazione

\$ cp <sorgente> <destinazione> *copia il file/cartella sorgente nella destinazione*

\$ mv <sorgente> <destinazione> *sposta il file/cartella sorgente nella destinazione*

\$ rm <nome file o directory> *rimuove il file/directory*

- cp, ls, rm, mv necessitano di *-r* o *-R* per la copia/elencazione/cancellazione/spostamento ricorsivo di una cartella

Introduzione minimale alla shell Linux

- Esecuzione da amministratore (super user)

```
$ sudo <comando>
```

- Serve aiuto?

```
$ man <nome comando>
```

```
$ <nome comando> --help
```

quasi tutti i comandi lo hanno: riepiloga brevemente le opzioni

Introduzione minimale alla shell Linux

- Installazione pacchetti

```
$ apt-get [opzioni]
```

Applicazione specifica di installazione, rimozione aggiornamento dei pacchetti

```
$ apt [opzioni]
```

Interfaccia di alto livello per ricerca pacchetti: ha tutte le opzioni di apt-get più quelle di ricerca

Introduzione minimale alla shell Linux

- Opzioni apt (e apt-get per la parte installazione/rimozione)
 - `list` – elenco pacchetti
 - `--installed` elenco pacchetti installati
 - `--upgradable` elenco pacchetti obsoleti che possono essere aggiornati
 - `search <espressione regolare>` - ricerca di pacchetti
 - `show <nome pacchetto>` - mostra informazioni sul pacchetto
 - `install, remove, purge <espressione regolare>` - installa, rimuove, rimuove insieme ai file di configurazione, i pacchetti che corrispondono all'espressione regolare
 - `update` – aggiorna le liste dei pacchetti disponibili
 - `upgrade` – installa gli aggiornamenti dei pacchetti disponibili

Installazione e configurazione dello stack LAMP

- Sui sistemi «Debian» (cioè con gestore di pacchetti di tipo Debian) è possibile installare *meta-pacchetti*
 - Un meta-pacchetto ha un nome che termina convenzionalmente con il carattere «^»
 - Lo stack LAMP è uno di questi: `lamp-server^`

```
$ sudo apt install lamp-server^
```

Installazione e configurazione dello stack LAMP

- Sui sistemi «Debian» (cioè con gestore di pacchetti di tipo Debian) è possibile installare *meta-pacchetti*
 - E' possibile installare il «gestore di task» `tasksel` usando `apt` e installare successivamente lo stack LAMP con il comando:

```
$ sudo tasksel install lamp-server
```

Installazione e configurazione dello stack LAMP

- E' possibile installare il «gestore di task» `tasksel` usando `apt` e installare successivamente lo stack LAMP:

```
$ sudo apt install tasksel
```

```
$ sudo tasksel install lamp-server
```
- E poi è comodo installare l'applicazione `phpMyAdmin` per la gestione web del server MySQL, *ma con il db già configurato con il suo utente di amministrazione – non root*

```
$ sudo apt install phpmyadmin
```

Installazione e configurazione dello stack LAMP

- MySQL nelle ultime versioni su Linux non supporta più l'autenticazione remota dell'utente root
- E' necessario autenticarsi solo in locale essendo root o un *sudoer*:
`$ sudo mysql -u root`
- Conviene creare un utente con tutti i privilegi per le connessioni remote che possa essere usato anche da *phpMyAdmin*:

```
mysql> CREATE USER 'vagrantdb'@'localhost' IDENTIFIED BY  
'vagrantdb';  
mysql> GRANT ALL ON *.* TO 'vagrantdb'@'localhost' WITH GRANT  
OPTION;  
mysql> FLUSH PRIVILEGES;
```

Installazione e configurazione dello stack LAMP

- La configurazione di accesso a phpMyAdmin si trova nel file

`/etc/phpmyadmin/config-db.php`

- Qui si possono configurare le variabili `$dbuser` e `$dbpass` se non già impostate automaticamente con la routine `dbconfig-common` invocata dall'installer

Installazione e configurazione dello stack LAMP

- Il file di configurazione vero e proprio di phpMyAdmin è

`/etc/phpmyadmin/config-inc.php`

- Qui si possono configurare le righe:

```
$cfg[ 'Servers' ][$i][ 'controluser' ] = $dbuser;
```

```
$cfg[ 'Servers' ][$i][ 'controlpass' ] = $dbpass;
```

Sostituendo esplicitamente username e password al posto delle variabili `$dbuser` e `$dbpass` usate da `dbconfig-common`

Installazione e configurazione dello stack LAMP

- L'engine PHP e la sua interazione con Apache è configurato attraverso il file:

`/etc/php/<numero_versione>/apache2/php.ini`

- Questo file contiene informazioni sul logging degli errori, sulle dimensioni massime degli upload, sul tempo di esecuzione degli script etc.

Installazione e configurazione dello stack LAMP

- Apache è un web server strutturato a «moduli» con la seguente struttura di cartelle:

```
/etc/apache2/  
|-- apache2.conf  
|   |-- ports.conf  
|-- mods-enabled  
|   |-- *.load  
|   |-- *.conf  
|-- conf-enabled  
|   |-- *.conf  
|-- sites-enabled  
|   |-- *.conf
```


Installazione e configurazione dello stack LAMP

- Comandi gestione Apache

```
$ sudo systemctl [status|start|stop|restart] apache2
```

Gestione dei servizi di sistema applicata ad apache2 (vale anche per il servizio mysql)

```
$ sudo apache2ctl [start|stop|restart]
```

Gestione diretta del servizio di sistema apache2

```
$ a2en{mod,site,conf} abilita un modulo, sito o configurazione
```

```
$ a2dis{mod,site,conf} disabilita un modulo, sito o configurazione
```

```
$ a2query richiede informazioni al server apache sulla sua configurazione
```

Installazione e configurazione dello stack LAMP

- Installazione della configurazione di phpMyAdmin

```
$ sudo ln -s /etc/phpmyadmin/apache.conf /etc/apache2/conf-available/phpmyadmin.conf
```

Creazione di una configurazione disponibile per phpmyadmin da abilitare successivamente

```
$ a2enconf phpmyadmin abilita la configurazione phpmyadmin
```

```
$ sudo apache2ctl restart
```

Riavvio di apache2 con la nuova configurazione

Installazione e configurazione dello stack LAMP

- Configurazione della cartella utente come possibile percorso del server nella forma `http://<indirizzo delserver>/~<nomeutente>/`
 - E' necessario installare il modulo `userdir`

```
$ sudo a2enmod userdir
$ sudo apache2ctl restart
```
 - Se si vogliono eseguire script php dalle cartelle utente, si deve commentare la direttiva che lo impedisce nel file `/etc/apache2/mods-available/php.<numero-versione>.conf`

```
<IfModule mod_userdir.c>
    <Directory /home/*/public_html>
        php_admin_flag engine Off
    </Directory>
</IfModule>
```

Installazione e configurazione dello stack LAMP

- Creazione di un Virtual Host cioè di un sito con nome pubblico stabilito da un DNS che in realtà punta ad una cartella gestita dal server
 - E' necessario abilitare un nuovo sito
 - Si crea un nuovo file di configurazione nella cartella `/etc/apache2/sites-available` con un nome del tipo `<numero progressivo sito>-<nome sito>.conf`
 - Nel file di configurazione del sito impostare correttamente le direttive `ServerName` e `DocumentRoot`

Installazione e configurazione dello stack LAMP

- Creazione di un Virtual Host cioè di un sito con nome pubblico stabilito da un DNS che in realtà punta ad una cartella gestita dal server
 - E' necessario abilitare un nuovo sito
 - Si deve configurare il DNS locale `/etc/hosts` della macchina da cui si naviga per puntare correttamente al nostro esempio, aggiungendo una riga siffatta (*operazione possibile solo come sudoer*):

```
192.168.1.253 www.example.com
```

Simulazione di alcune richieste al server

- Il client HTTP a linea di comando `curl` è in grado di inviare al server le richieste più complesse:

```
$ curl <URL> esegue la classica richiesta GET
$ curl <URL>?<nome>=<valore>&... richiesta GET con parametri
$ curl -G -d '?<nome>=<valore>&... '<URL> richiesta GET con
parametri
$ curl -d '?<nome>=<valore>&... '<URL> richiesta POST con parametri
$ curl -I <URL> richiesta HEAD
$ curl --request-target "*" -X OPTIONS <URL> richiesta
OPTIONS *
```

- In generale il parametro `-X` opportunamente configurato è quello che consente di inviare tutti gli altri metodi oltre a GET, POST e HEAD