

Practica07

Crear dos servidores de apache, un mysql y un balanceador de carga. Además en los servidores de apache instalaremos wordpress.

Paso 1

Crear la carpeta donde vamos a guardar todos los archivos de configuración.

```
mkdir practica-07
```

Una vez creada accedemos a la carpeta

```
cd practica-07
```

Paso 2

Una vez dentro ejecutamos el comando vagrant init para que nos cree el archivo Vagrantfile

```
vagrant init
```

Paso 3

Una vez tengamos el fichero **Vagrantfile** lo abrimos y agregamos las dos máquinas de Apache, la máquina de MySQL y el balanceador. Aquí tenéis el fichero **Vagrantfile** con sus correspondientes máquinas.

```
# -*- mode: ruby -*-
# vi: set ft=ruby :

Vagrant.configure("2") do |config|
  config.vm.box = "ubuntu/bionic64"

  # Apache HTTP Server
  config.vm.define "web1" do |app|
    app.vm.hostname = "web1"
    app.vm.network "private_network", ip: "192.168.33.11"
    app.vm.provision "shell", path: "provision-for-apache.sh"
  end
```

```
# Apache HTTP Server2
config.vm.define "web2" do |app|
  app.vm.hostname = "web2"
  app.vm.network "private_network", ip: "192.168.33.12"
  app.vm.provision "shell", path: "provision-for-apache.sh"
end

# MySQL Server
config.vm.define "db" do |app|
  app.vm.hostname = "db"
  app.vm.network "private_network", ip: "192.168.33.13"
  app.vm.provision "shell", path: "provision-for-mysql.sh"
end

# Balanceador
config.vm.define "balanceador" do |app|
  app.vm.hostname = "balanceador"
  app.vm.network "private_network", ip: "192.168.33.10"
  app.vm.provision "shell", path: "provision-for-balanceador.sh"
end
end
```

Paso 4

Una vez agregada las máquinas creamos los archivos de configuración para cada máquina, `provision-for-apache.sh` que será el mismo para las dos máquinas de apache, `provision-for-mysql.sh` para la máquina de mysql, `provision-for-balanceador.sh` para el balanceador.

Aquí tenemos el `provision-for-apache.sh`.

```
#!/bin/bash
#INSTALACION DE APACHE
apt-get update
apt-get install -y apache2
apt-get install -y php libapache2-mod-php php-mysql
sudo /etc/init.d/apache2 restart

#DESCARGAR UNZIP
apt-get install -y unzip

#DESCARGAMOS WORDPRESS
cd /tmp
rm -rf latest.zip
wget https://wordpress.org/latest.zip
rm -rf wordpress
unzip latest.zip

#CREAMOS EL ARCHIVO wp-config.php
cd wordpress
cp wp-config-sample.php wp-config.php
```

```
#CREAMOS VARIABLES
DB_NAME=wordpress_db
DB_USER=wordpress_user
DB_PASSWORD=wordpress_password
DB_HOST=192.168.33.13

#CONFIGURAMOS EL ARCHIVO wp-config.php
sed -i "s/database_name_here/$DB_NAME/" wp-config.php
sed -i "s/username_here/$DB_USER/" wp-config.php
sed -i "s/password_here/$DB_PASSWORD/" wp-config.php
sed -i "s/localhost/$DB_HOST/" wp-config.php

#COPIAMOS LOS TODOS LOS ARCHIVO DE WORDPRESS
cp . /var/www/html -R

#DAMOS PERMISOS
chown www-data:www-data /var/www/html/* -R

#BORRAMOS EL INDEX
rm -rf /var/www/html/index.html
```

Aquí está el [provision-for-mysql.sh](#).

```
#!/bin/bash
#INSTALAMOS MYSQL
apt-get update
apt-get -y install debconf-utils

DB_ROOT_PASSWD=root
debconf-set-selections <<< "mysql-server mysql-server/root_password password
$DB_ROOT_PASSWD"
debconf-set-selections <<< "mysql-server mysql-server/root_password_again password
$DB_ROOT_PASSWD"

apt-get install -y mysql-server

#sed realiza buscar y reemplazar
sed -i -e "s/127.0.0.1/0.0.0.0/" /etc/mysql/mysql.conf.d/mysqld.cnf

#REINICIAMOS EL SERVICIO
/etc/init.d/mysql restart

#CREAMOS UN USUARIO GLOBAL
mysql -u root mysql -p$DB_ROOT_PASSWD <<< "GRANT ALL PRIVILEGES ON *.* TO root@'%'
IDENTIFIED BY '$DB_ROOT_PASSWD';"
mysql -u root mysql -p$DB_ROOT_PASSWD <<< "FLUSH PRIVILEGES;"

#CREAMOS LA BASE DE DATOS WORDPRESS

mysql -u root mysql -p$DB_ROOT_PASSWD <<< "DROP DATABASE IF EXISTS wordpress_db;"
mysql -u root mysql -p$DB_ROOT_PASSWD <<< "CREATE DATABASE wordpress_db CHARACTER
```

```
SET utf8;"
mysql -u root mysql -p$DB_ROOT_PASSWD <<< "GRANT ALL PRIVILEGES ON wordpress_db.*
TO wordpress_user@'%' IDENTIFIED BY 'wordpress_password';"
mysql -u root mysql -p$DB_ROOT_PASSWD <<< "FLUSH PRIVILEGES;"

#REINICIAMOS EL SERVICIO
/etc/init.d/mysql restart
```

Ahora el `provision-for-balanceador.sh`.

```
#!/bin/bash
apt-get update
apt-get install -y apache2
apt-get install -y php libapache2-mod-php php-mysql
sudo /etc/init.d/apache2 restart

#ACTIVAR MODULOS
a2enmod proxy
a2enmod proxy_http
a2enmod proxy_ajp
a2enmod rewrite
a2enmod deflate
a2enmod headers
a2enmod proxy_balancer
a2enmod proxy_connect
a2enmod proxy_html
a2enmod lbmethod_byrequests

#BORRAMOS EL ARCHIVO 000-default.conf
sudo rm -rf /etc/apache2/sites-enabled/000-default.conf

#COPIAMOS EL ARCHIVO 000-default.conf NUEVO
cp /vagrant/000-default.conf /etc/apache2/sites-enabled/

sudo /etc/init.d/apache2 restart
```

Paso 5

Una vez tengamos la provision de las cuatro máquinas, solo nos queda levantarlas y hacer las comprobaciones de cada máquina.

Paso 6

Para levantar las máquinas podemos hacerlas todas a la vez con `vagrant up`

```
vagrant up
```

O levantarla una a una con el comando `vagrant "nombre de la máquina"`

```
vagrant up db
```

Paso 7

Para aplicar cambios realizados en el `provision-for-apache.sh`, `provision-for-mysql.sh` o `provision-for-balanceador.sh` ejecutaremos el comando `vagrant provision "nombre de la máquina"`

```
vagrant provision web1
```

Paso 8

Para apagar la máquina hacemos un `vagrant halt "nombre de la máquina"`

```
vagrant halt db
```

Paso 9

Y si queremos destruir una máquina sería `vagrant destroy "nombre de la máquina"`

```
vagrant destroy web2
```

Paso 10

Para comprobar que el balanceador realiza su trabajo con los frontales, en el navegador web ponemos la dirección IP de uno de los frontales y nos tendría que salir la página de la base de datos para introducir, modificar o borrar datos.

Paso 11

Ahora comprobaremos que podemos instalar wordpress y crear entradas, para ello en el navegador deberemos de poner la ip del balanceador y se iniciara el instalador de wordpress. Ya solo nos quedaría instalarlo y crear entradas en wordpress.