

Utilizando Vagrant

Preparar el entorno de trabajo

Instalar Vagrant en debian/ubuntu de acuerdo a la bitácora en el sitio de la cátedra.

Creamos un directorio de trabajo y un Vagrantfile

Creamos un directorio de trabajo y un Vagrantfile:

```
mkdir vagrant-nginx-2020
cd vagrant-nginx-2020
vagrant init ubuntu/bionic64
```

Ajustamos el Vagrantfile por defecto

Quitamos todas las líneas comentadas con #, ajustando sólo la que comienza con

config.vm.box_url :

```
config.vm.box_url =
"https://vagrantcloud.com/ubuntu/boxes/bionic64/versions/20200324.0.0/
providers/virtualbox.box"
```

Ajustamos la configuración del provider virtualbox:

```
# Configuración específica para el provider virtualbox
config.vm.provider "virtualbox" do |v|
  v.name = "Vagrant con Nginx - 2020"
  v.customize ["modifyvm", :id, "--memory", "512"]
end
```

Definimos el provisioning de la máquina mediante shell script:

```
# Configuramos el provisioning mediante shell script
config.vm.provision "shell" do |s|
  s.path = "provision/setup.sh"
end
```

Creamos el shell script de provisioning en la carpeta **provision** con el nombre **setup.sh** y el siguiente contenido :

```
#!/bin/bash
export DEBIAN_FRONTEND=noninteractive

echo "Provisioning virtual machine..."
echo "Actualizando lista de paquetes"
apt-get update

echo "Instalando Git"
apt-get install git -y
```

```
echo "Instalando Nginx"
apt-get install nginx -y
```

Configuramos la redirección de puertos redirigiendo el puerto 8080 del host al 80 del guest :

```
# Configuramos la redirección del puerto 80 de la máquina virtual al
# 8080 del host
config.vm.network :forwarded_port, guest: 80, host: 8080,
auto_correct: true
```

Levantamos la maquina virtual

Levantamos la máquina virtual mediante el comando:

```
vagrant up
```

Para ingresar a la máquina virtual utilizamos:

```
vagrant ssh
```

Configuramos un Nginx con PHP y MySQL

Detenemos la máquina virtual en caso de que esté corriendo con:

```
vagrant halt
```

Configuramos el directorio del Vagrantfile para que sea accesible desde la VM

```
# Configuramos que el subdirectorio ./ en el host que esté mapeado
# al /var/www de la máquina virtual
config.vm.synced_folder "./", "/var/www", create: true, group: "www-
data", owner: "www-data"
```

Agregamos el archivo para el vhost nginx en **provision/config/nginx_vhost** con:

```
server {
    listen 80;
    server_name localhost;

    root /var/www/src/;
    index index.php index.html;

    # Important for VirtualBox
    sendfile off;

    location / {
        try_files $uri $uri/ =404;
    }

    location ~* \.php {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/var/run/php/php7.2-fpm.sock
    }
}
```

Agregamos una página web vacía en **src/index.php** con:

```
<?php echo "Hello World!"; ?>
```

Agregamos la configuración de los servicios al final del archivo **setup.sh**:

```
echo "Instalando PHP..."
apt-get install php-common php-cli php-fpm -y

echo "Instalando extensiones PHP..."
apt-get install curl php-curl php-gd php-mysql -y

echo "Configurando Nginx.."
cp /var/www/provision/config/nginx_vhost
/etc/nginx/sites-available/nginx_vhost > /dev/null
ln -s /etc/nginx/sites-available/nginx_vhost /etc/nginx/sites-enabled/
rm -rf /etc/nginx/sites-enabled/default
service nginx restart

echo "Instalando MySQL..."
apt-get install debconf-utils -y
debconf-set-selections <<< "mysql-server mysql-server/root_password
password root"
debconf-set-selections <<< "mysql-server
mysql-server/root_password_again password root"
apt-get install mysql-server -y
```

Volvemos a provisionar la máquina con:

```
vagrant up --provision
```

Otros comando útiles

Para ver todos los comandos: <https://www.vagrantup.com/docs/cli/>

Eliminar la máquina virtual:

```
vagrant destroy
```

Suspender/Reanudar la máquina virtual:

```
vagrant suspend
```

```
vagrant resume
```

Ver el estado de la máquina virtual:

```
vagrant status
```

Ver los mapeos de puertos de la máquina virtual:

```
vagrant port
```