

Definiciones

Vagrant

Es una herramienta que simplifica el trabajo para ejecutar y gestionar máquinas virtuales creada por Hashicorp

Vagrantfile

En este fichero está guardada la configuración de las MVs usadas

Atlas

Servicio usado por Vagrant para descargar 'base boxes'

Packer

Herramienta de línea de comandos que automatiza la creación de MVs con distintos sistemas y proveedores

Boxes

Vagrant utiliza imágenes preconstruidas de máquinas virtuales llamadas "cajas" (boxes)

Requisitos previos: VirtualBox instalado, estación de trabajo con más de 4 GB de RAM

Vagrant vs Docker

Si utilizas Virtualbox o Vmware como el proveedor para Vagrant, puede iniciar una máquina virtual según las configuraciones en el archivo Vagrantfile.

Si utilizas Docker como el proveedor para Vagrant, entonces lanzará contenedores de Docker según las configuraciones en el archivo Vagrantfile.

Comandos básicos

\$ vagrant --version	consultar la versión
\$ export VAGRANT_HOME=/ruta/al/directorio	configuración
\$ vagrant \$ vagrant -h	ayuda
\$ vagrant init	crear Vagrantfile

Configuración del Vagrantfile

Especificar la box a utilizar

```
config.vm.box = "ubuntu/bionic64"
```

Definir el nombre de la MV

```
config.vm.define "mi_vm"
```

Modificar RAM y nº de núcleos

```
config.vm.provider "virtualbox" do |vb|
  vb.memory = 1024 vb.cpus = 2 end
```

Configuración de red

```
config.vm.network "forwarded_port",
  guest: 80, host: 8080 config.vm.network
  "private_network", type: "dhcp"
```

Entorno gráfico

```
vb.gui = true
```

Configurar un disco

```
vb.customize ['createhd', '--filename',
  'additional_disk.vdi', '--size', 500 * 1024]
vb.customize ['storageattach', :id, '--storagectl', 'SATA Controller', '--port', 1, '--device', 0, '--type', 'hdd', '--medium',
  'additional_disk.vdi']
```

Boxes

\$ vagrant box add [nombre] [URL o ruta local]	agregar un nuevo box a Vagrant
\$ vagrant box list	mostrar la lista de boxes instaladas
\$ vagrant box remove [nombre]	eliminar un box específico
\$ vagrant box update	actualizar las boxes si es necesario
\$ vagrant box outdated	mostrar la lista de las actualizaciones requeridas
\$ vagrant box repackage [nombre] [versión]	empaquetar una box existente en una nueva versión

Inicio máquina virtual

El proceso de arranque de la máquina son 3 etapas:

1.- Descarga e instalación de la 'box' en el sistema anfitrión.

```
~/.vagr ran t.d /boxes
```

2.- Importación de la 'box' al proyecto.

```
~/Virt ualBox VMs/
```

3.- Arranque del sistema.

La imagen utilizada para la máquina virtual se almacena en un archivo llamado "box file" con extensión .box



By Carlos99

cheatography.com/carlos99/

Published 22nd September, 2023.

Last updated 22nd September, 2023.

Page 1 of 3.

Sponsored by CrosswordCheats.com

Learn to solve cryptic crosswords!

<http://crosswordcheats.com>

Compartir carpeta entre MV y Host

```
config.vm.synced_folder ".", "/vagrant"
type: "virtuoso"
```

Gestión MV

\$ vagrant up	iniciar la máquina virtual
\$ vagrant halt [ID]	detener la máquina virtual
\$ vagrant destroy [ID]	eliminar la máquina virtual

Los IDs de las máquinas virtuales se pueden consultar utilizando el comando **\$ vagrant global-status**

Crear entornos múltiples MVs

```
Fichero Vagrantfile, instalar y arrancar Ubuntu y Debian
Vagrant.configure("2") do |config|
  config.vm.define "ubuntu" do |ubuntu|
    ubuntu.vm.box = "bento/ubuntu-14.04"
  end
  config.vm.define "ubuntu" do |ubuntu|
    ubuntu.vm.box = "bento/ubuntu-14.04"
  end
  config.vm.define "ubuntu" do |ubuntu|
    ubuntu.vm.box = "bento/ubuntu-14.04"
  end
  config.vm.define "debian" do |debian|
    debian.vm.box = "bento/debian-7"
  end
end
```

Fichero Vagrantfile: instalar y arrancar dos nodos (nodo1 y nodo2) de una misma distribución centos7:

```
Vagrant.configure("2") do |config|
  config.vm.define "nodo1" do |nodo1|
    nodo1.vm.box = "centos/7"
    nodo1.vm.hostname = 'nodo1'
    nodo1.vm.network "public_network",
      ip: "192.168.1.24"
  end
end
```

Primera Imagen

Creación de la imagen:

\$ mkdir box-factory	creación del directorio
\$ cd box-factory	
\$ vagrant init -m ubuntu/trusty32	iniciar nuevo proyecto en una imagen existente
\$ vagrant up	levantar máquina virtual
\$ vagrant ssh	iniciar sesión ssh
\$ jekyll --version	comprobar versión
\$ vagrant package -o box-jekyll-1.box	crear la imagen de la máquina virtual

Gestión de imágenes:

\$ vagrant box list	consultar la lista de boxes (imágenes)
\$ vagrant box add [nombre] [Dirección]	añadir una nueva imagen al sistema
\$ vagrant box remove [nombre]	eliminar una imagen del sistema

Imágenes con chef/bento

```
git clone https://github.com/chef/bento.git
$ cd bento
$ packer build --only-ubuntu-14.04-i386.json
```

Seguridad

Riesgo #1 Descarga de la imagen de internet

Al descargar una imagen de una fuente no fiable para crear una máquina virtual

\$ vagrant box add [nombre] [URL o ruta local]	agregar una box a Vagrant
\$ vagrant box list	lista de boxes instaladas
\$ vagrant box remove [nombre]	eliminar una box

Riesgo #2 Exposición de puertos

Vagrant por defecto redirige solo un puerto, el SSH, desde el sistema invitado y no debe exponer otros servicios a la red

\$ config.vm.network :forwarded_port, guest: [puerto_invitado], host: [puerto_anfitrión], host_ip: "127.0.0.1"	configurar la redirección de puertos para el sistema invitado en el archivo Vagrantfile
---	---

Riesgo #3 Credenciales Predeterminadas

Las credenciales predeterminadas de Vagrant, como el usuario "vagrant" con permisos de sudo, pueden ser un riesgo de seguridad si no se cambian.

\$ vagrant ssh	iniciar una sesión SSH en la máquina virtual utilizando las credenciales predeterminadas
\$ vagrant ssh -i	modificar la contraseña del usuario "vagrant" en la máquina virtual

Riesgo #4 Llaves RSA predeterminadas

Todas las imágenes descargadas de Internet incluyen las mismas llaves RSA, lo que puede representar un riesgo si no se cambian.

```
$ vagrant ssh
$ vagrant ssh
```

iniciar una sesión SSH en la máquina virtual utilizando las llaves predeterminadas.

Proyecto Sinatra

```
$ vagrant ssh
$ ruby app.rb -o 0.0.0.0 &
$ logout
```

config.vm.network :forwarded_port, guest: 4567, host: 9090, host_ip: "127.0.0.1"

si el puerto está ocupado en nuestro sistema, hay que cambiarlo en el fichero Vagrantfile

```
$ vagrant reload
```

aplicar los cambios

Acceder a root directamente

Para acceder remotamente lo haremos mediante el SSH

```
cp -r /home/vagrant/.ssh/ /root/
chown -R root.root /root/.ssh/
```

Modificar el usuario SSH en Vagrantfile:
config.ssh.username = "root"

Usar Ansible para ejecutar comandos en máquinas Vagrant habilitadas con sudo

```
ansible -vv all -m command -a "id" -u vagrant --private-key /ruta/a/lla/clave/p-rivada -b --become-method sudo -e "ansible_become_pass=contraseña"
```

```
ansible -vv all -m command -a "id" -u vagrant --private-key /ruta/a/lla/clave/p-rivada -b
```



By Carlos99
cheatography.com/carlos99/

Published 22nd September, 2023.
Last updated 22nd September, 2023.
Page 3 of 3.

Sponsored by CrosswordCheats.com
Learn to solve cryptic crosswords!
<http://crosswordcheats.com>