

---

## Computación en la Nube

Oscar Hernán Mondragón Martínez  
Universidad Autónoma de Occidente

# Vagrant Primeros Pasos

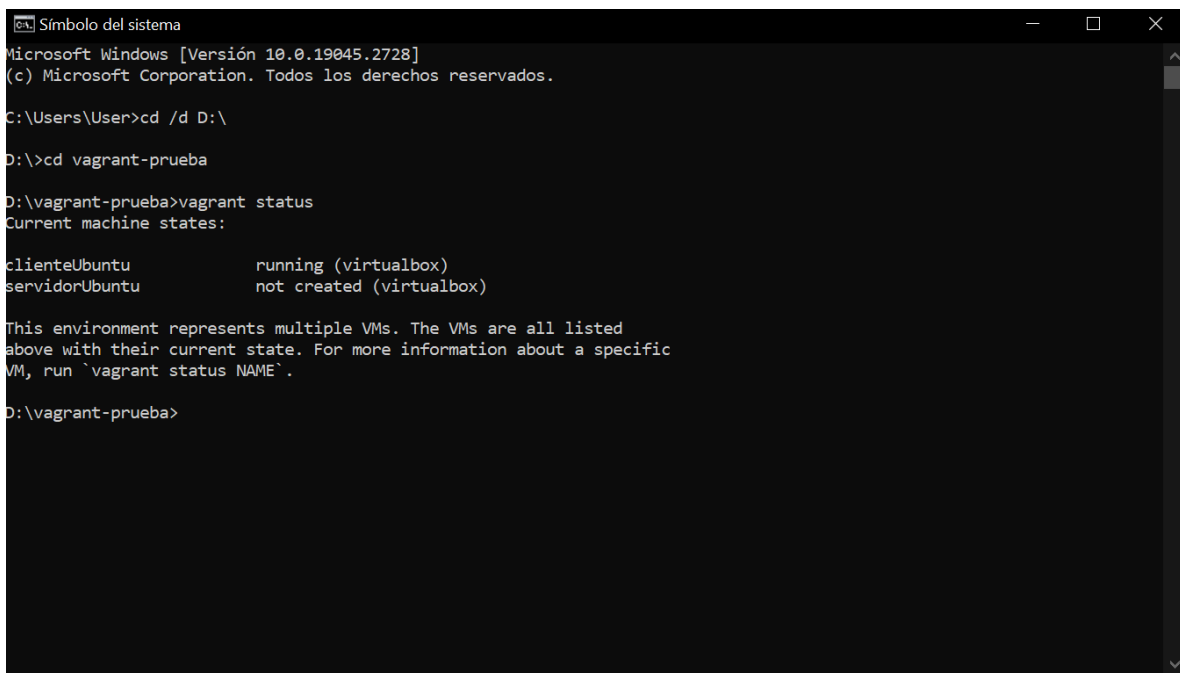
Diego Ivan Perea Montealegre, Sammir Hassan Ordoñez, Gabriel Angel Jeannot Viana

diego.perea@uao.edu.co, samir.hassan@uao.edu.co, gabriel.jeannot@uao.edu.co

Departamento de Automática y Electrónica

Universidad Autónoma de Occidente

Santiago de Cali, Valle del Cauca



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19045.2728]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\User>cd /d D:\

D:\>cd vagrant-prueba

D:\vagrant-prueba>vagrant status
Current machine states:

clienteUbuntu      running (virtualbox)
servidorUbuntu     not created (virtualbox)

This environment represents multiple VMs. The VMs are all listed
above with their current state. For more information about a specific
VM, run `vagrant status NAME`.

D:\vagrant-prueba>
```

Como no se creo servidorUbuntu maquina se hace:

```
Símbolo del sistema
are usually good hints as to what may be wrong.

If you're using a custom box, make sure that networking is properly
working and you're able to connect to the machine. It is a common
problem that networking isn't setup properly in these boxes.
Verify that authentication configurations are also setup properly,
as well.

If the box appears to be booting properly, you may want to increase
the timeout ("config.vm.boot_timeout") value.

D:\vagrant-prueba>
D:\vagrant-prueba>
D:\vagrant-prueba>vagrant up servidorUbuntu
Bringing machine 'servidorUbuntu' up with 'virtualbox' provider...
==> servidorUbuntu: Importing base box 'bento/ubuntu-22.04'...
==> servidorUbuntu: Matching MAC address for NAT networking...
==> servidorUbuntu: Checking if box 'bento/ubuntu-22.04' version '202303.13.0' is up to date...
==> servidorUbuntu: Setting the name of the VM: vagrant-prueba_servidorUbuntu_1681164612939_31197
==> servidorUbuntu: Fixed port collision for 22 => 2222. Now on port 2200.
==> servidorUbuntu: Clearing any previously set network interfaces...
==> servidorUbuntu: Preparing network interfaces based on configuration...
    servidorUbuntu: Adapter 1: nat
    servidorUbuntu: Adapter 2: hostonly
==> servidorUbuntu: Forwarding ports...
    servidorUbuntu: 22 (guest) => 2200 (host) (adapter 1)
==> servidorUbuntu: Booting VM...
==> servidorUbuntu: Waiting for machine to boot. This may take a few minutes...
    servidorUbuntu: SSH address: 127.0.0.1:2200
    servidorUbuntu: SSH username: vagrant
```

Maquinas creadas:

```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19045.2728]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\User>cd /d D:\

D:\>cd vagrant-prueba

D:\vagrant-prueba>vagrant status
Current machine states:

clienteUbuntu          running (virtualbox)
servidorUbuntu         not created (virtualbox)

This environment represents multiple VMs. The VMs are all listed
above with their current state. For more information about a specific
VM, run `vagrant status NAME`.

D:\vagrant-prueba>vagrant status
Current machine states:

clienteUbuntu          running (virtualbox)
servidorUbuntu         running (virtualbox)

This environment represents multiple VMs. The VMs are all listed
above with their current state. For more information about a specific
VM, run `vagrant status NAME`.

D:\vagrant-prueba>
```

```
vagrant@servidorUbuntu: ~  
D:\vagrant-prueba>vagrant ssh servidorUbuntu  
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-67-generic x86_64)  
  
 * Documentation:  https://help.ubuntu.com  
 * Management:    https://landscape.canonical.com  
 * Support:       https://ubuntu.com/advantage  
  
System information as of Mon Apr 10 10:15:09 PM UTC 2023  
  
System load:  0.0322265625      Processes:            157  
Usage of /:   12.0% of 30.34GB   Users logged in:      0  
Memory usage: 12%              IPv4 address for eth0: 10.0.2.15  
Swap usage:   0%                IPv4 address for eth1: 192.168.100.3  
  
This system is built by the Bento project by Chef Software  
More information can be found at https://github.com/chef/bento  
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-67-generic x86_64)  
  
 * Documentation:  https://help.ubuntu.com  
 * Management:    https://landscape.canonical.com  
 * Support:       https://ubuntu.com/advantage  
  
System information as of Mon Apr 10 10:15:09 PM UTC 2023  
  
System load:  0.0322265625      Processes:            157  
Usage of /:   12.0% of 30.34GB   Users logged in:      0  
Memory usage: 12%              IPv4 address for eth0: 10.0.2.15  
Swap usage:   0%                IPv4 address for eth1: 192.168.100.3  
  
This system is built by the Bento project by Chef Software  
More information can be found at https://github.com/chef/bento  
vagrant@servidorUbuntu:~$
```

```
vagrant@servidorUbuntu: ~  
More information can be found at https://github.com/chef/bento  
vagrant@servidorUbuntu:~$ sudo -i  
sudo: -i: command not found  
vagrant@servidorUbuntu:~$ -i  
-bash: -i: command not found  
vagrant@servidorUbuntu:~$ sudo apt-get install net-tools  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following NEW packages will be installed:  
  net-tools  
0 upgraded, 1 newly installed, 0 to remove and 2 not upgraded.  
Need to get 204 kB of archives.  
After this operation, 819 kB of additional disk space will be used.  
Get:1 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 net-tools amd64 1.60+git20181103.0eebece-1ubuntu5 [204 kB]  
Fetched 204 kB in 2s (107 kB/s)  
Selecting previously unselected package net-tools.  
(Reading database ... 44187 files and directories currently installed.)  
Preparing to unpack .../net-tools_1.60+git20181103.0eebece-1ubuntu5_amd64.deb ...  
Unpacking net-tools (1.60+git20181103.0eebece-1ubuntu5) ...  
Setting up net-tools (1.60+git20181103.0eebece-1ubuntu5) ...  
Processing triggers for man-db (2.10.2-1) ...  
Scanning processes...  
Scanning linux images...  
  
Running kernel seems to be up-to-date.  
  
No services need to be restarted.  
  
No containers need to be restarted.  
  
No user sessions are running outdated binaries.  
  
No VM guests are running outdated hypervisor (qemu) binaries on this host.  
vagrant@servidorUbuntu:~$
```

```

vagrant@servidorUbuntu: ~
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  net-tools
0 upgraded, 1 newly installed, 0 to remove and 2 not upgraded.
Need to get 204 kB of archives.
After this operation, 819 kB of additional disk space will be used.
Get:1 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 net-tools amd64 1.60+git20181103.0eebece-1ubuntu5 [204 kB]
Fetched 204 kB in 2s (107 kB/s)
Selecting previously unselected package net-tools.
(Reading database ... 44187 files and directories currently installed.)
Preparing to unpack .../net-tools_1.60+git20181103.0eebece-1ubuntu5_amd64.deb ...
Unpacking net-tools (1.60+git20181103.0eebece-1ubuntu5) ...
Setting up net-tools (1.60+git20181103.0eebece-1ubuntu5) ...
Processing triggers for man-db (2.10.2-1) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
vagrant@servidorUbuntu:~$ sudo apt-get install vim
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
vim is already the newest version (2:8.2.3995-1ubuntu2.3).
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
vagrant@servidorUbuntu:~$

```

## Ifconfig

```

vagrant@servidorUbuntu: ~
No VM guests are running outdated hypervisor (qemu) binaries on this host.
vagrant@servidorUbuntu:~$ sudo apt-get install vim
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
vim is already the newest version (2:8.2.3995-1ubuntu2.3).
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
vagrant@servidorUbuntu:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe10:b45 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:10:0b:45 txqueuelen 1000 (Ethernet)
    RX packets 22317 bytes 30601782 (30.6 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 3181 bytes 326313 (326.3 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.3 netmask 255.255.255.0 broadcast 192.168.100.255
    inet6 fe80::a00:27ff:fe00:c9c7 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:00:c9:c7 txqueuelen 1000 (Ethernet)
    RX packets 2 bytes 486 (486.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 14 bytes 1096 (1.0 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 46 bytes 5437 (5.4 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 46 bytes 5437 (5.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

```

## Ahora con el cliente Ubuntu

```
vagrant@clienteUbuntu: ~  
=> clienteUbuntu: flag to force provisioning. Provisioners marked to run always will still run.  
  
D:\vagrant-prueba>vagrant status  
Current machine states:  
  
clienteUbuntu          running (virtualbox)  
servidorUbuntu         running (virtualbox)  
  
This environment represents multiple VMs. The VMs are all listed  
above with their current state. For more information about a specific  
VM, run `vagrant status NAME`.  
  
D:\vagrant-prueba>vagrant ssh clienteUbuntu  
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-67-generic x86_64)  
  
 * Documentation:  https://help.ubuntu.com  
 * Management:    https://landscape.canonical.com  
 * Support:       https://ubuntu.com/advantage  
  
System information as of Tue Mar 14 02:29:44 UTC 2023  
  
System load: 1.3974609375      Processes:            160  
Usage of /:  15.7% of 30.34GB  Users logged in:     0  
Memory usage: 11%             IPv4 address for enp0s3: 10.0.2.15  
Swap usage:  0%  
  
This system is built by the Bento project by Chef Software  
More information can be found at https://github.com/chef/bento  
vagrant@clienteUbuntu:~$
```

```
vagrant@clienteUbuntu: ~  
  
This system is built by the Bento project by Chef Software  
More information can be found at https://github.com/chef/bento  
vagrant@clienteUbuntu:~$ sudo apt-get install net-tools  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following NEW packages will be installed:  
  net-tools  
0 upgraded, 1 newly installed, 0 to remove and 2 not upgraded.  
Need to get 204 kB of archives.  
After this operation, 819 kB of additional disk space will be used.  
Get:1 http://us.archive.ubuntu.com/ubuntu jammy/main amd64 net-tools amd64 1.60+git20181103.0eebece-1ubuntu5_amd64.deb 204 kB  
Fetched 204 kB in 2s (133 kB/s)  
Selecting previously unselected package net-tools.  
(Reading database ... 44187 files and directories currently installed.)  
Preparing to unpack .../net-tools_1.60+git20181103.0eebece-1ubuntu5_amd64.deb ...  
Unpacking net-tools (1.60+git20181103.0eebece-1ubuntu5) ...  
Setting up net-tools (1.60+git20181103.0eebece-1ubuntu5) ...  
Processing triggers for man-db (2.10.2-1) ...  
Scanning processes...  
Scanning linux images...  
  
Running kernel seems to be up-to-date.  
  
No services need to be restarted.  
  
No containers need to be restarted.  
  
No user sessions are running outdated binaries.  
  
No VM guests are running outdated hypervisor (qemu) binaries on this host.  
vagrant@clienteUbuntu:~$ sudo apt-get install vim  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done
```

```

vagrant@clienteUbuntu: ~
vagrant@clienteUbuntu:~$ sudo apt-get install vim
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
vim is already the newest version (2:8.2.3995-1ubuntu2.3).
0 upgraded, 0 newly installed, 0 to remove and 2 not upgraded.
vagrant@clienteUbuntu:~$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.2.15 netmask 255.255.255.0 broadcast 10.0.2.255
    inet6 fe80::a00:27ff:fe10:b45 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:10:0b:45 txqueuelen 1000 (Ethernet)
    RX packets 1693 bytes 397212 (397.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 1201 bytes 186509 (186.5 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.100.2 netmask 255.255.255.0 broadcast 192.168.100.255
    inet6 fe80::a00:27ff:fe2d:2895 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:2d:28:95 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 12 bytes 956 (956.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 18 bytes 2231 (2.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 18 bytes 2231 (2.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

vagrant@clienteUbuntu:~$

```

Conectadas al ping

```

vagrant@clienteUbuntu: ~
    loop txqueuelen 1000 (Local Loopback)
    RX packets 18 bytes 2231 (2.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 18 bytes 2231 (2.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

vagrant@clienteUbuntu:~$ ping 192.168.100.3
PING 192.168.100.3 (192.168.100.3) 56(84) bytes of data.
64 bytes from 192.168.100.3: icmp_seq=1 ttl=64 time=3.41 ms
64 bytes from 192.168.100.3: icmp_seq=2 ttl=64 time=2.23 ms
64 bytes from 192.168.100.3: icmp_seq=3 ttl=64 time=1.70 ms
64 bytes from 192.168.100.3: icmp_seq=4 ttl=64 time=2.61 ms
64 bytes from 192.168.100.3: icmp_seq=5 ttl=64 time=2.93 ms
64 bytes from 192.168.100.3: icmp_seq=6 ttl=64 time=6.71 ms
64 bytes from 192.168.100.3: icmp_seq=7 ttl=64 time=1.13 ms
64 bytes from 192.168.100.3: icmp_seq=8 ttl=64 time=2.24 ms
64 bytes from 192.168.100.3: icmp_seq=9 ttl=64 time=2.37 ms
64 bytes from 192.168.100.3: icmp_seq=10 ttl=64 time=3.48 ms
64 bytes from 192.168.100.3: icmp_seq=11 ttl=64 time=4.11 ms
64 bytes from 192.168.100.3: icmp_seq=12 ttl=64 time=2.78 ms
64 bytes from 192.168.100.3: icmp_seq=13 ttl=64 time=2.63 ms
64 bytes from 192.168.100.3: icmp_seq=14 ttl=64 time=2.18 ms
64 bytes from 192.168.100.3: icmp_seq=15 ttl=64 time=2.09 ms
64 bytes from 192.168.100.3: icmp_seq=16 ttl=64 time=3.16 ms
64 bytes from 192.168.100.3: icmp_seq=17 ttl=64 time=3.82 ms
64 bytes from 192.168.100.3: icmp_seq=18 ttl=64 time=2.04 ms
64 bytes from 192.168.100.3: icmp_seq=19 ttl=64 time=2.30 ms
64 bytes from 192.168.100.3: icmp_seq=20 ttl=64 time=4.80 ms
64 bytes from 192.168.100.3: icmp_seq=21 ttl=64 time=2.55 ms
64 bytes from 192.168.100.3: icmp_seq=22 ttl=64 time=2.38 ms
^C
--- 192.168.100.3 ping statistics ---
22 packets transmitted, 22 received, 0% packet loss, time 21137ms
rtt min/avg/max/mdev = 1.127/2.891/6.711/1.161 ms
vagrant@clienteUbuntu:~$

```

## BOXES VAGRANT

```
D:\vagrant-prueba>vagrant package servidorUbuntu --output mynew.box
==> servidorUbuntu: Attempting graceful shutdown of VM...
==> servidorUbuntu: Clearing any previously set forwarded ports...
==> servidorUbuntu: Exporting VM...
==> servidorUbuntu: Compressing package to: D:/vagrant-prueba/mynew.box

D:\vagrant-prueba>ls
Vagrantfile  mynew.box

D:\vagrant-prueba>
```

```
D:\vagrant-prueba>ls
Vagrantfile  mynew.box

D:\vagrant-prueba>vagrant box add mynewbox mynew.box
==> box: Box file was not detected as metadata. Adding it directly...
==> box: Adding box 'mynewbox' (v0) for provider:
    box: Unpacking necessary files from: file:///D:/vagrant-prueba/mynew.box
    box:
==> box: Successfully added box 'mynewbox' (v0) for 'virtualbox'!

D:\vagrant-prueba>vagrant box list
bento/ubuntu-22.04 (virtualbox, 202303.13.0)
mynewbox           (virtualbox, 0)

D:\vagrant-prueba>
```

Vagrant cloud (hashi corp)

### Edit Provider

**Provider**   
Vagrant will need to know how to use this provider.

**Checksum type**

**Checksum**

[Update provider](#) [Delete provider](#)

### Add Provider File

**Upload File**  mynew.box

**Status** Uploading 36.39/962.86MB (4%)



### Release Box Version

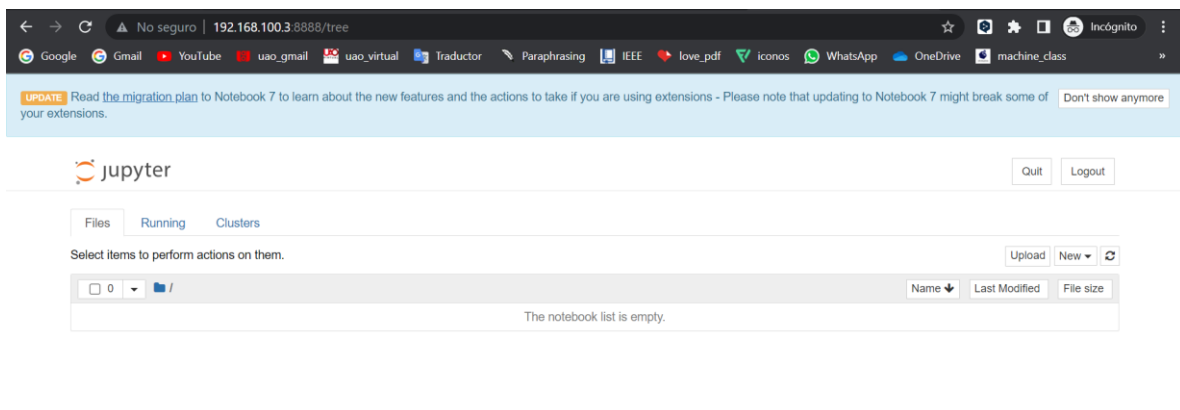
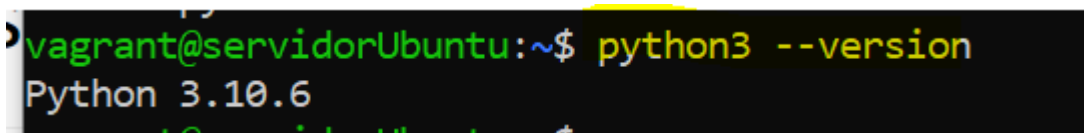
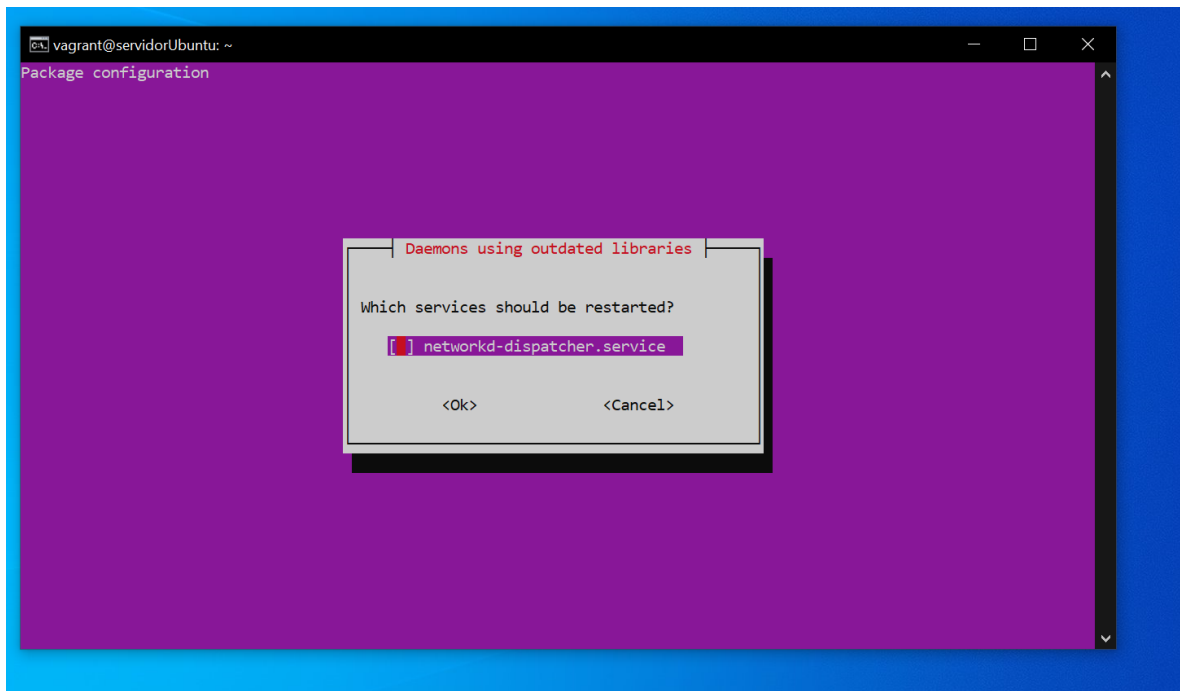
⚠ This version is unreleased. Upon releasing, it will be available to users from Vagrant.

[Release version](#)

### Update Box Version

**Version**





## INVESTIGACION:

directorios sincronizados de Vagrant.

Los directorios sincronizados de Vagrant son una funcionalidad que permite sincronizar una carpeta en la máquina anfitriona con la máquina virtual, lo que permite trabajar en los archivos del proyecto en la máquina anfitriona y utilizar los recursos de la máquina virtual para compilar o ejecutar el proyecto[1]. Vagrant sincroniza automáticamente las carpetas que se encuentran en el directorio de trabajo[2]. Las carpetas compartidas son el tipo predeterminado de carpeta sincronizada para los usuarios de VirtualBox[3]. Para sincronizar carpetas en Windows, se puede utilizar la herramienta

gratuita de línea de comandos rsync. Vagrant es una herramienta de línea de comandos que permite generar entornos de desarrollo reproducibles[4].

En resumen, los directorios sincronizados en Vagrant son una función que permite compartir archivos entre la máquina anfitriona y la máquina virtual creada con Vagrant. Esto es útil para editar archivos en la máquina anfitriona usando herramientas y entornos favoritos y que los cambios se reflejen automáticamente en la máquina virtual sin tener que copiar manualmente los archivos. Vagrant sincroniza por defecto el directorio en el que se encuentra el archivo Vagrantfile en la máquina anfitriona con un directorio en la máquina virtual, y se pueden configurar tantos directorios sincronizados como se necesite. Además, Vagrant soporta varios mecanismos de sincronización, como rsync, NFS y SMB.

Para configurar directorios sincronizados en Vagrant, debes agregar la siguiente línea en tu archivo Vagrantfile:

```
config.vm.synced_folder "directorio_anfitrion", "directorio_vm"
```

Donde "directorio\_anfitrion" es la ruta absoluta del directorio en tu máquina anfitriona que quieres sincronizar con la máquina virtual, y "directorio\_vm" es la ruta absoluta del directorio en la máquina virtual donde quieres que se sincronice el directorio del anfitrión.

Por ejemplo, si tienes un directorio llamado "mi\_proyecto" en tu máquina anfitriona en la ruta "/home/usuario/mi\_proyecto", y quieres sincronizarlo con un directorio en la máquina virtual en la ruta "/vagrant/mi\_proyecto", deberías agregar la siguiente línea en tu archivo Vagrantfile:

```
config.vm.synced_folder "/home/usuario/mi_proyecto", "/vagrant/mi_proyecto"
```

Recuerda que debes ejecutar el comando "**vagrant reload**" para que los cambios en el archivo Vagrantfile surtan efecto.

```
Vagrant.configure("2") do |config|
  config.vm.box = "centos/7"
  config.vm.synced_folder ".", "/home/vagrant/selftuts",type:"virtualbox"
end
```

Figura Configuración Vagrantfile para sincronizar folder

```
connection to 127.0.0.1 closed.
centos-7 } » ls
vagrantfile Vagrantfile~
centos-7 } » vim Vagrantfile
centos-7 } » vagrant reload
=> default: Attempting graceful shutdown of VM...
=> default: Checking if box 'centos/7' is up to date...
=> default: A newer version of the box 'centos/7' is available! You currently
=> default: have version '1611.01'. The latest is version '1704.01'. Run
=> default: `vagrant box update` to update.
=> default: Clearing any previously set forwarded ports...
=> default: Clearing any previously set network interfaces...
=> default: Preparing network interfaces based on configuration...
default: Adapter 1: nat
=> default: Forwarding ports...
default: 22 (guest) => 2222 (host) (adapter 1)
=> default: Booting VM...
=> default: Waiting for machine to boot. This may take a few minutes...
default: SSH address: 127.0.0.1:2222
```

Figura Vagrant reload para actualizar configuración de sincronización

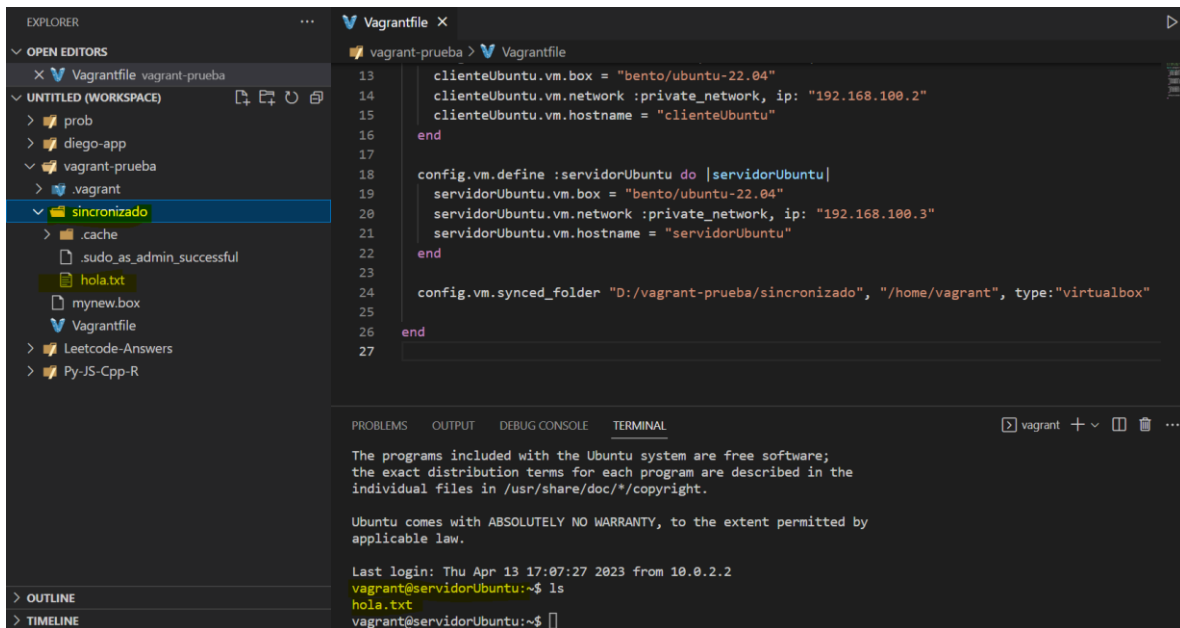
```

[vagrant@localhost ~]$ ls
selftuts
[vagrant@localhost ~]$ cd selftuts/
[vagrant@localhost selftuts]$ ls
Vagrantfile  Vagrantfile~
[vagrant@localhost selftuts]$ vim test.txt
-bash: vim: command not found
[vagrant@localhost selftuts]$ touch test.txt
[vagrant@localhost selftuts]$ exit
logout
Connection to 127.0.0.1 closed.
{ centos-7 } » ls
test.txt  Vagrantfile  Vagrantfile~
{ centos-7 } » ls
test.txt  Vagrantfile  Vagrantfile~
{ centos-7 } » ls
gimp-images  test.txt  Vagrantfile  Vagrantfile~
{ centos-7 } » cd gimp-images
{ gimp-images } master » ls
blog-images  git  images-master  node  sails  selftuts-site  vim  wordpress
{ gimp-images } master » cd ..
{ centos-7 } » ls
gimp-images  test.txt  Vagrantfile  Vagrantfile~
{ centos-7 } » vagrant ssh
Last login: Sun May 21 09:23:49 2017 from 10.0.2.2
[vagrant@localhost ~]$ cl

```

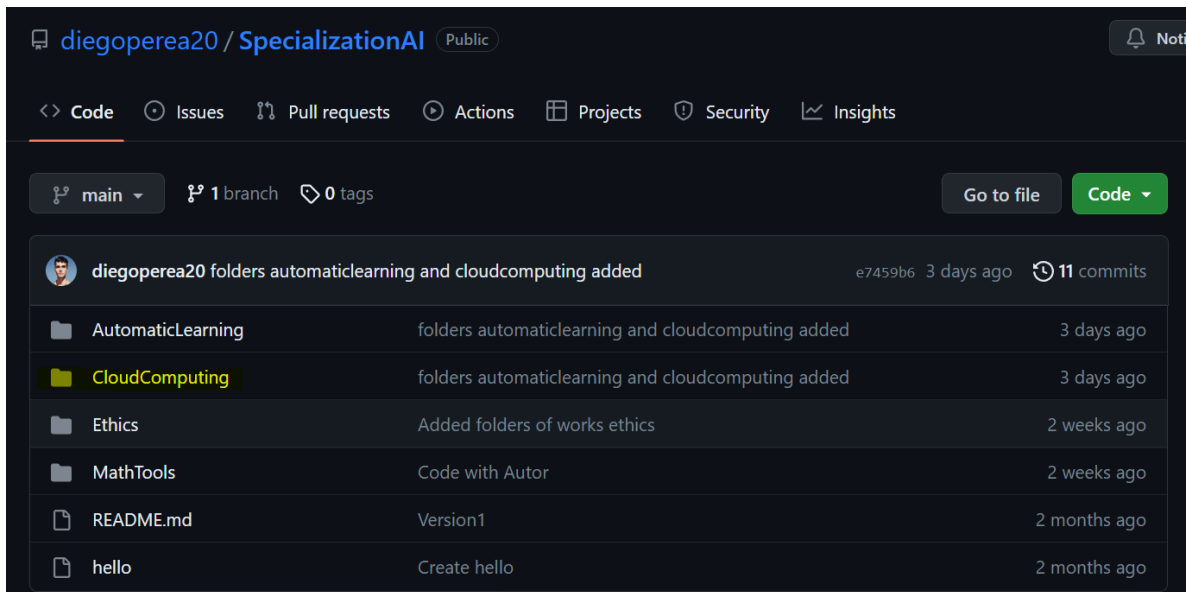
Figura Archivo y carpeta sincronizada

Comprobación :



Si pide la contraseña es “vagrant” por defecto

**Github**



## Linux

# Desarrollo

## 0. Abrir una ventana de terminal en Linux

```
vagrant@servidorUbuntu: ~$
==> servidorUbuntu: waiting for machine to boot. This may take a few minutes...
servidorUbuntu: SSH address: 127.0.0.1:2200
servidorUbuntu: SSH username: vagrant
servidorUbuntu: SSH auth method: private key
==> servidorUbuntu: Machine booted and ready!
==> servidorUbuntu: Machine already provisioned. Run 'vagrant provision' or use the '--provision'
==> servidorUbuntu: flag to force provisioning. Provisioners marked to run always will still run.
PS D:\Documentos UAO\Especializacion\Compu Nube\Corte 1\Entornos virtuales> vagrant ssh servidorUbuntu
vagrant@127.0.0.1's password:
PS D:\Documentos UAO\Especializacion\Compu Nube\Corte 1\Entornos virtuales> vagrant ssh servidorUbuntu
vagrant@127.0.0.1's password:
vagrant@127.0.0.1's password:
Welcome to Ubuntu 22.04.2 LTS (GNU/Linux 5.15.0-67-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Fri Apr 14 05:59:43 PM UTC 2023

System load:  0.0               Processes:    140
Usage of /:   12.7% of 30.34GB   Users logged in: 0
Memory usage: 11%              IPv4 address for eth0: 10.0.2.15
Swap usage:  0%                IPv4 address for eth1: 192.168.100.3

* Introducing Expanded Security Maintenance for Applications.
  Receive updates to over 25,000 software packages with your
  Ubuntu Pro subscription. Free for personal use.

https://ubuntu.com/pro

This system is built by the Bento project by Chef Software
More information can be found at https://github.com/chef/bento
Last login: Thu Apr 13 18:21:43 2023 from 10.0.2.2
vagrant@servidorUbuntu: ~$
```

## 1. Listar directorio actual

```
vagrant@servidorUbuntu:~$ pwd
/home/vagrant
vagrant@servidorUbuntu:~$
```

## 2. Volver al directorio home desde cualquier ubicación

```
vagrant@servidorUbuntu:~$ cd ~
vagrant@servidorUbuntu:~$ pwd
/home/vagrant
vagrant@servidorUbuntu:~$
```

## 3. Crear un nuevo directorio

```
vagrant@servidorUbuntu:~$ pwd
/home/vagrant
vagrant@servidorUbuntu:~$ cd ~
vagrant@servidorUbuntu:~$ pwd
/home/vagrant
vagrant@servidorUbuntu:~$ mkdir temp
vagrant@servidorUbuntu:~$ mkdir temp/stuff
vagrant@servidorUbuntu:~$ mkdir temp/stuff/things
vagrant@servidorUbuntu:~$ mkdir -p temp/stuff/things/orange/pear/grape
vagrant@servidorUbuntu:~$
```

## 4. Cambiar de un directorio a otro usando cd

```
vagrant@servidorUbuntu:~$ cd temp
vagrant@servidorUbuntu:~/temp$ pwd
/home/vagrant/temp
vagrant@servidorUbuntu:~/temp$ cd stuff
vagrant@servidorUbuntu:~/temp/stuff$ pwd
/home/vagrant/temp/stuff
vagrant@servidorUbuntu:~/temp/stuff$ cd
vagrant@servidorUbuntu:~$ pwd
/home/vagrant
vagrant@servidorUbuntu:~$
```

## 5. Listar los contenidos de un directorio usando ls

```
vagrant@servidorUbuntu:~$ cd temp
vagrant@servidorUbuntu:~/temp$ ls
stuff
vagrant@servidorUbuntu:~/temp$ cd stuff
vagrant@servidorUbuntu:~/temp/stuff$ ls
things
vagrant@servidorUbuntu:~/temp/stuff$ cd things
vagrant@servidorUbuntu:~/temp/stuff/things$ ls
orange
vagrant@servidorUbuntu:~/temp/stuff/things$ cd
vagrant@servidorUbuntu:~$
```

## 6. Eliminar un directorio vacío

```
vagrant@servidorUbuntu:~$ cd temp
vagrant@servidorUbuntu:~/temp$ ls
stuff
vagrant@servidorUbuntu:~/temp$ cd stuff/things/orange/pear/grape/
vagrant@servidorUbuntu:~/temp/stuff/things/orange/pear/grape$ cd ..
vagrant@servidorUbuntu:~/temp/stuff/things/orange/pear$ rmdir grape
vagrant@servidorUbuntu:~/temp/stuff/things/orange/pear$ cd ..
vagrant@servidorUbuntu:~/temp/stuff/things/orange$ rmdir pear
vagrant@servidorUbuntu:~/temp/stuff/things/orange$ cd ..
vagrant@servidorUbuntu:~/temp/stuff/things$ ls
orange
vagrant@servidorUbuntu:~/temp/stuff/things$
```

## 7. Guardar su localización actual, ir a una nueva localización con pushd y retornar a la localización guardada con popd

```
vagrant@servidorUbuntu:~$ cd temp
vagrant@servidorUbuntu:~/temp$ mkdir -p i/like/icecream
vagrant@servidorUbuntu:~/temp$ pushd i/like/icecream/
~/temp/i/like/icecream ~/temp
vagrant@servidorUbuntu:~/temp/i/like/icecream$ popd
~/temp
vagrant@servidorUbuntu:~/temp$ pwd
/home/vagrant/temp
vagrant@servidorUbuntu:~/temp$ pushd i/like
~/temp/i/like ~/temp
vagrant@servidorUbuntu:~/temp/i/like$ pwd
/home/vagrant/temp/i/like
vagrant@servidorUbuntu:~/temp/i/like$ pushd icecream
~/temp/i/like/icecream ~/temp/i/like ~/temp
vagrant@servidorUbuntu:~/temp/i/like/icecream$ pwd
/home/vagrant/temp/i/like/icecream
vagrant@servidorUbuntu:~/temp/i/like/icecream$ popd
~/temp/i/like ~/temp
vagrant@servidorUbuntu:~/temp/i/like$ pwd
/home/vagrant/temp/i/like
vagrant@servidorUbuntu:~/temp/i/like$ popd
~/temp
vagrant@servidorUbuntu:~/temp$ pushd i/like/icecream
~/temp/i/like/icecream ~/temp
vagrant@servidorUbuntu:~/temp/i/like/icecream$ pushd
~/temp ~/temp/i/like/icecream
vagrant@servidorUbuntu:~/temp$ pwd
/home/vagrant/temp
vagrant@servidorUbuntu:~/temp$ pushd
~/temp/i/like/icecream ~/temp
vagrant@servidorUbuntu:~/temp/i/like/icecream$ pwd
/home/vagrant/temp/i/like/icecream
vagrant@servidorUbuntu:~/temp/i/like/icecream$
```

## 8. Crear archivos vacíos con touch

```
vagrant@servidorUbuntu:~$ cd temp
vagrant@servidorUbuntu:~/temp$ touch grass.txt
vagrant@servidorUbuntu:~/temp$ ls
grass.txt  i  stuff
vagrant@servidorUbuntu:~/temp$
```



## 9. Copiar un archivo desde una localización a otra con cp

```
vagrant@servidorUbuntu:~/temp$  
vagrant@servidorUbuntu:~/temp$ cp grass.txt neat.txt  
vagrant@servidorUbuntu:~/temp$ ls  
grass.txt  i  neat.txt  stuff  
vagrant@servidorUbuntu:~/temp$ cd neat.txt awesome.txt  
-bash: cd: too many arguments  
vagrant@servidorUbuntu:~/temp$ cp neat.txt awesome.txt  
vagrant@servidorUbuntu:~/temp$ ls  
awesome.txt  grass.txt  i  neat.txt  stuff  
vagrant@servidorUbuntu:~/temp$
```

## 10. Mover un archivo de un lugar a otro usando mv

```
vagrant@servidorUbuntu:~/temp$ mv awesome.txt uncool.txt  
vagrant@servidorUbuntu:~/temp$ ls  
grass.txt  i  neat.txt  stuff  uncool.txt  
vagrant@servidorUbuntu:~/temp$
```

## 11. Cree un archivo con contenido de texto llamado test.txt y ábralo con less o con more. ¿Cuál es la diferencia?

La principal diferencia entre less y more es que less es más flexible y potente que more. less permite desplazarse hacia atrás y hacia adelante por el archivo, buscar patrones de texto, copiar texto del archivo y más. Además, less no carga todo el contenido del archivo en memoria de una sola vez, lo que lo hace más rápido y eficiente que more para archivos grandes.

## 12. Abra el archivo usando cat

```
Este es un texto de ejemplo  
test2.txt (END)
```

```
vagrant@servidorUbuntu:~/temp$ echo "Este es un texto de ejemplo" > test2.txt  
vagrant@servidorUbuntu:~/temp$  
vagrant@servidorUbuntu:~/temp$ ls  
grass.txt  i  neat.txt  stuff  test2.txt  test.txt  uncool.txt  
vagrant@servidorUbuntu:~/temp$ less test2.txt  
vagrant@servidorUbuntu:~/temp$ cat test2.txt  
Este es un texto de ejemplo  
vagrant@servidorUbuntu:~/temp$
```



### 13. Eliminar un archivo usando rm

```
vagrant@servidorUbuntu:~/temp$ ls
grass.txt  i  neat.txt  stuff  test2.txt  test.txt  uncool.txt
vagrant@servidorUbuntu:~/temp$
vagrant@servidorUbuntu:~/temp$ rm uncool.txt
vagrant@servidorUbuntu:~/temp$ ls
grass.txt  i  neat.txt  stuff  test2.txt  test.txt
vagrant@servidorUbuntu:~/temp$ rm neat.txt
vagrant@servidorUbuntu:~/temp$ ls
grass.txt  i  stuff  test2.txt  test.txt
vagrant@servidorUbuntu:~/temp$
```

### 14. Salir de la terminal

```
vagrant@servidorUbuntu:~/temp$
vagrant@servidorUbuntu:~/temp$ exit
logout
Connection to 127.0.0.1 closed.
PS D:\Documentos UAO\Especializacion\Compu Nube\Corte 1\Entornos virtuales>
```

## Tensorflow

Funcionamiento de Máquina virtual con Tensor Flow:

Se ingresa a cmd y se escoge una locación para crear el directorio “pruebaTensorflow”

```
Ca. Seleccíonar Símbolo del sistema - PowerShell
Microsoft Windows [Versíon 10.0.19045.2846]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Samir Hassan>PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Users\Samir Hassan> cd Desktop
PS C:\Users\Samir Hassan\Desktop> mkdir pruebaTensorflow

Directorio: C:\Users\Samir Hassan\Desktop

Mode                LastWriteTime         Length Name
----                -
d-----          14/04/2023   5:16 p. m.          pruebaTensorflow


PS C:\Users\Samir Hassan\Desktop> cd pruebaTensorflow
PS C:\Users\Samir Hassan\Desktop\pruebaTensorflow>
```

Se inicia vagrant en dicha carpeta

```
PS C:\Users\Samir Hassan\Desktop\pruebaTensorflow> vagrant init
A `Vagrantfile` has been placed in this directory. You are now
ready to `vagrant up` your first virtual environment! Please read
the comments in the Vagrantfile as well as documentation on
`vagrantup.com` for more information on using Vagrant.
PS C:\Users\Samir Hassan\Desktop\pruebaTensorflow> _
```


Se dirige a vagrant cloud, en donde se busca una MV que tenga incorporado Tensorflow

### Discover Vagrant Boxes



Provider any virtualbox vmware libvirt more ▾

Sort by Downloads Recently Created Recently Updated

|   |   |                         |                  |                              |
|---|---|-------------------------|------------------|------------------------------|
|  | gt50/tensorflow 0.9.0<br>tensorflow installed on bento/ubuntu-14.04 image | <span>virtualbox</span> | Downloads<br>152 | Released<br>over 6 years ago |
|---|---|-------------------------|------------------|------------------------------|

Se copia el contenido del vagrantfile

## gt50/tensorflow Vagrant box

How to use this box with [Vagrant](#):

Vagrantfile [New](#)

```
Vagrant.configure("2") do |config|
  config.vm.box = "gt50/tensorflow"
  config.vm.box_version = "0.9.0"
end
```

**v0.9.0** currently released version

This version was created over 6 years ago.

Ubuntu 14.04 Python 2.7.6 Tensorflow 0.9.0

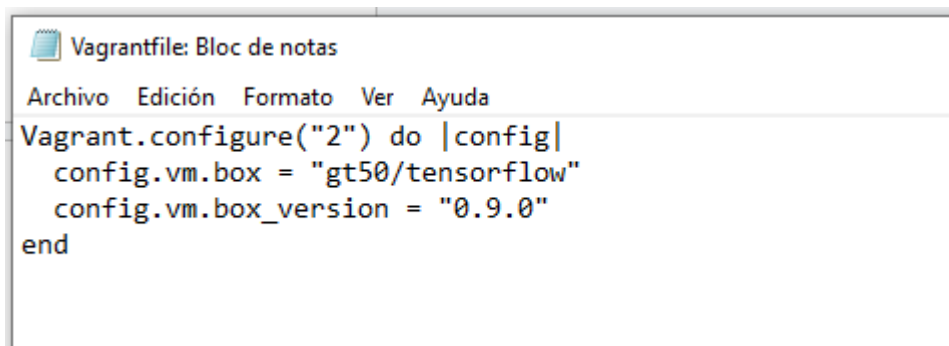
1 provider for this version.

**virtualbox** Hosted by Vagrant Cloud (572 MB)



[Homepage](#) [Terms](#) [DMCA](#) [Privacy](#) [Security](#) [API](#)

Se modifica el vagrantfile con el contenido copiado



```
Vagrantfile: Bloc de notas
Archivo Edición Formato Ver Ayuda
Vagrant.configure("2") do |config|
  config.vm.box = "gt50/tensorflow"
  config.vm.box_version = "0.9.0"
end
```

Se empieza a descargar la imagen del sistema y a configurarse en nuestra MV

```
PS C:\Users\Samir Hassan\Desktop\pruebaTensorflow> vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Box 'gt50/tensorflow' could not be found. Attempting to find and install...
default: Box Provider: virtualbox
default: Box Version: 0.9.0
==> default: Loading metadata for box 'gt50/tensorflow'
default: URL: https://vagrantcloud.com/gt50/tensorflow
==> default: Adding box 'gt50/tensorflow' (v0.9.0) for provider: virtualbox
default: Downloading: https://vagrantcloud.com/gt50/boxes/tensorflow/versions/0.9.0/providers/virtualbox.box
```

Se verifica el estado de la máquina creada, la cual efectivamente está funcionando.

```
PS C:\Users\Samir Hassan\Desktop\pruebaTensorflow> vagrant status
Current machine states:

default                running (virtualbox)

The VM is running. To stop this VM, you can run `vagrant halt` to
shut it down forcefully, or you can run `vagrant suspend` to simply
suspend the virtual machine. In either case, to restart it again,
simply run `vagrant up`.
```

Ingresamos como usuario a la MV

```
PS C:\Users\Samir Hassan\Desktop\pruebaTensorflow> vagrant ssh default
Welcome to Ubuntu 14.04.4 LTS (GNU/Linux 3.13.0-92-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
Last login: Mon Jul 25 00:01:00 2016 from 10.0.2.2
Welcome to Ubuntu 14.04.4 LTS (GNU/Linux 3.13.0-92-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
Last login: Mon Jul 25 00:01:00 2016 from 10.0.2.2
vagrant@vagrant:~$
```

Ingresamos el comando Python para abrir la consola de Python en la MV

```
vagrant@vagrant:~$ python
Python 2.7.6 (default, Jun 22 2015, 17:58:13)
[GCC 4.8.2] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

En la terminal de comandos de Python importamos TensorFlow

```
Type "help", "copyright", "credits" or "license" for more information.
>>> import tensorflow as tf
```

Y escribimos un pequeño código para evaluar el funcionamiento de TensorFlow

```
>>> hello = tf.constant('Hola, TensorFlow')
>>> sess = tf.Session()
>>> print(sess.run(hello))
Hola, TensorFlow
>>> _
```

Así, terminamos verificando el funcionamiento de la MV con TF.

#### Referencias:

1. HashiCorp. (2021). Synced Folders. [Online]. Disponible en: <https://developer.hashicorp.com/vagrant/docs/synced-folders> [Accedido el 9 Abril 2023].
2. Triviños, E. (2016). Vagrant III: Sincronización de carpetas. [Online]. Disponible en: <https://etrivinos.wordpress.com/2016/01/12/vagrant-iii-sincronizacion-de-carpetas/> [Accedido el 9 Abril 2023].
3. StackExchange. (2014). Symbolic links and synced folders in Vagrant. [Online]. Disponible en: <https://qastack.mx/programming/24200333/symbolic-links-and-synced-folders-in-vagrant> [Accedido el 9 Abril 2023].
4. RedesZone. (2018). Vagrant: instalación, configuración y ejemplos. [Online]. Disponible en: <https://www.redeszone.net/tutoriales/servidores/vagrant-instalacion-configuracion-ejemplos/> [Accedido el 9 Abril 2023].
5. “Part 4 : Synced folder in vagrant,” *YouTube*, 21-May-2017. [Online]. Available: [https://www.youtube.com/watch?v=oD-p\\_GbwiPA](https://www.youtube.com/watch?v=oD-p_GbwiPA). [Accessed: 11-Apr-2023]