 Universidad AUTÓNOMA de Occidente	UNIVERSIDAD AUTONOMA DE OCCIDENTE			
	FACULTAD DE INGENIERIA <i>DEPARTAMENTO DE AUTOMATICA Y ELECTRONICA</i>		CONFIGURACION AMBIENTE DE DESARROLLO	
	ESTUDIANTES:		Fecha:	
CONFIGURACION DEL AMBIENTE DE DESARROLLO				

OBJETIVOS

- Configurar el ambiente de desarrollo Vagrant + VirtualBox + Centos que se usará para la configuración de servicios de red.
- Aprender como se publican boxes en Vagrant Cloud.

1. Instalar VirtualBox 6.1.40

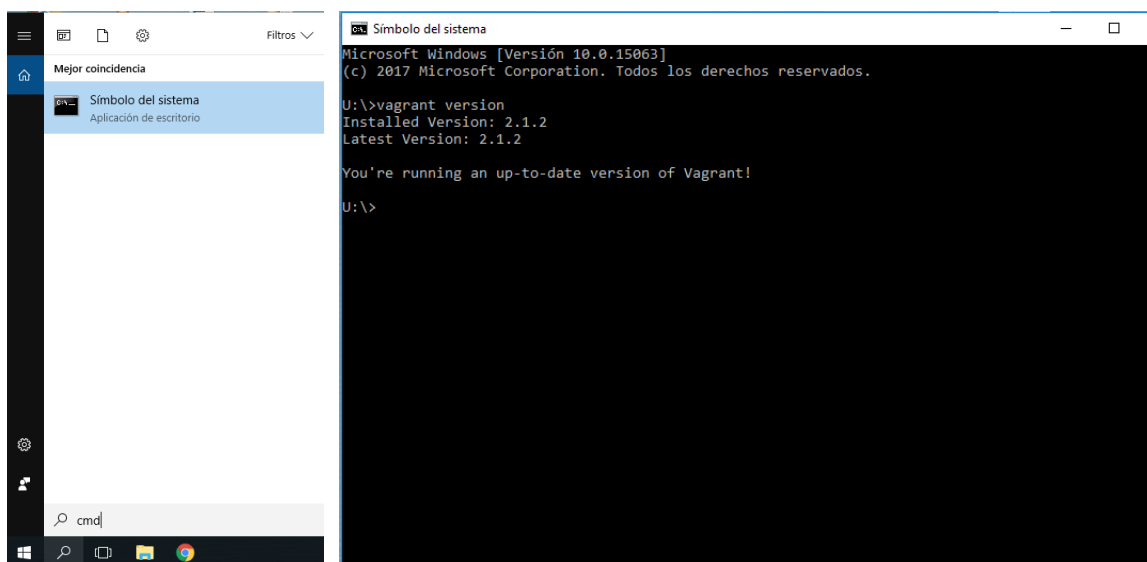
- Descargue e instale VirtualBox del sitio <https://www.virtualbox.org/>

2. Instalar Vagrant

- Descargue e instale Vagrant desde <https://releases.hashicorp.com/vagrant/2.3.4/>

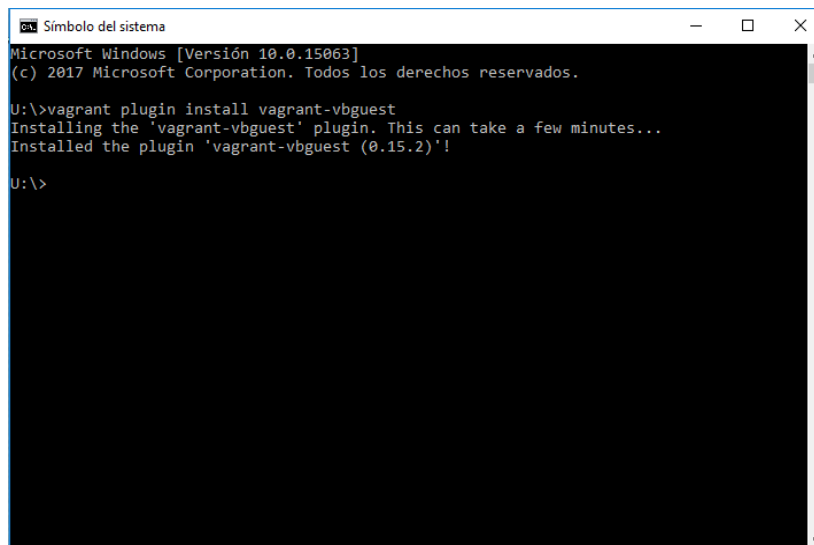
NOTA: Usaremos la versión 2.3.4 para 64 bits (vagrant_2.3.4_windows_amd64.msi) (para usuarios mac use el archivo con extensión .dmg)

3. Abra una consola de Windows y verifique la versión de Vagrant



- En la misma consola de Windows ejecute el siguiente comando para instalar el plugin vbguest

```
vagrant plugin install vagrant-vbguest
```



```
Microsoft Windows [Versión 10.0.15063]
(c) 2017 Microsoft Corporation. Todos los derechos reservados.

U:\>vagrant plugin install vagrant-vbguest
Installing the 'vagrant-vbguest' plugin. This can take a few minutes...
Installed the plugin 'vagrant-vbguest (0.15.2)'!

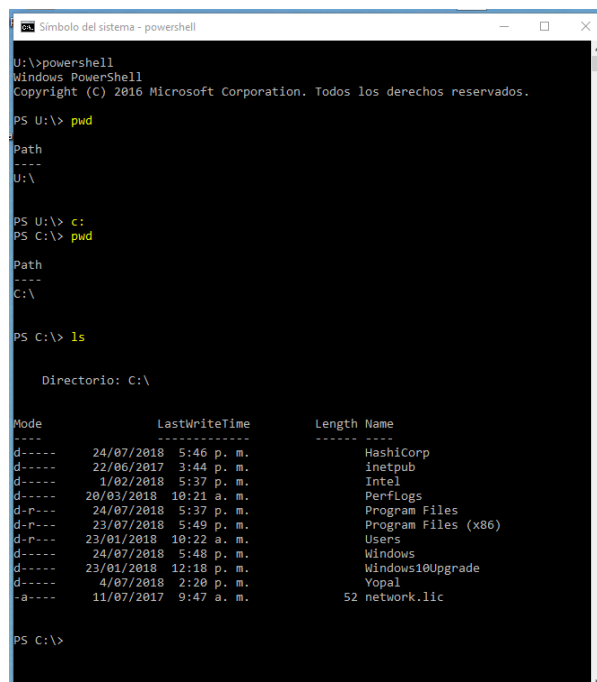
U:\>
```

Este es un plugin de Vagrant para mantener las adiciones de los guest de VirtualBox actualizadas.

4. Opcional. Powershell

Para algunos usuarios puede ser mas cómodo usar Powershell, el cual es similar a los Shells de Unix. Este ya viene preinstalado en Windows 10.

Para ejecutarlo, puede usar el buscador de aplicaciones de Windows o ejecutar “powershell” directamente en una consola de Windows, como se muestra a continuación.



```
U:\>powershell
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. Todos los derechos reservados.

PS U:\> pwd

Path
----
U:\

PS U:\> cd c:
PS C:\> pwd

Path
----
C:\

PS C:\> ls

Directorio: C:\

Mode                LastWriteTime         Length Name
----                -
d-----         24/07/2018   5:46 p. m.      HashiCorp
d-----         22/06/2017   3:44 p. m.      inetpub
d-----          1/02/2018   5:37 p. m.      Intel
d-----         20/03/2018  10:21 a. m.      PerfLogs
d-r-----         24/07/2018   5:37 p. m.      Program Files
d-r-----         23/07/2018   5:49 p. m.      Program Files (x86)
d-r-----         23/01/2018  10:22 a. m.      Users
d-----         24/07/2018   5:48 p. m.      Windows
d-----         23/01/2018  12:18 p. m.      Windows10Upgrade
d-----          4/07/2018   2:20 p. m.      Yopal
-a-----         11/07/2017   9:47 a. m.      52 network.lic

PS C:\>
```

Se puede ver en este ejemplo que se ejecutaron comandos como pwd, ls al estilo Unix. El comando pwd permite ver el directorio actual de trabajo, mientras que el ls el contenido de dicho directorio.

4. Configuración del ambiente de desarrollo

Los servicios de red que configuraremos usan el modelo cliente-servidor. Usaremos siempre dos maquinas virtuales, una que alojara los servicios configurados y otra que los consumirá.

Para las prácticas usaremos un Vagrantfile similar al siguiente.

```
Vagrant.configure("2") do |config|

  if Vagrant.has_plugin? "vagrant-vbguest"
    config.vbguest.no_install = true
    config.vbguest.auto_update = false
    config.vbguest.no_remote = true
  end

  config.vm.define :cliente do |cliente|
    cliente.vm.box = "centos/stream8"
    cliente.vm.network :private_network, ip: "192.168.50.2"
    cliente.vm.hostname = "cliente"
  end

  config.vm.define :servidor do |servidor|
    servidor.vm.box = "centos/stream8"
    servidor.vm.network :private_network, ip: "192.168.50.3"
    servidor.vm.hostname = "servidor"
  end

end
```

Este Vagrantfile define dos maquinas virtuales, una llamada servidor con dirección ip 192.168.50.3 y la otra cliente con dirección ip 192.168.50.2, ambas instanciadas desde un box en el repositorio de bento llamado "centos/stream8".

Nota: SOLO SI ES USUARIO MAC OSX:

En la maquina anfitrión crear el directorio /etc/vbox y dentro el archivo networks.conf (/etc/vbox/networks.conf): con el siguiente contenido:
* 192.168.50.0/24

CONFIGURACION

1. Cree un directorio llamado “prueba”
2. Dentro del directorio prueba cree un archivo llamado “Vagrantfile”.

Para crear un Vagrantfile de ejemplo puede usar el siguiente comando dentro de la carpeta prueba:

```
vagrant init
```

Modifique el Vagrantfile para que tenga UNICAMENTE el contenido mostrado en el ejemplo de arriba (al inicio de la sección)

3. Cree y configure las maquinas mediante el comando `vagrant up` ejecutado desde consola
4. Verifique el Puerto de reenvío para cada maquina virtual. En el ejemplo mostrado abajo, el puerto de reenvío es el 2222

```
$ vagrant up

Bringing machine 'default' up with 'virtualbox' provider...==> default:
Importing base box 'hashicorp/precise64'...==> default: Forwarding ports...

default: 22 (guest) => 2222 (host) (adapter 1)==> default: Waiting for
machine to boot...
```

5. Verifique el estado de las maquinas creadas con el commando

```
vagrant status
```

6. Establezca una sesión ssh con la maquina servidor

```
vagrant ssh servidor
```

7. Autenticarse como super usuario

```
sudo -i
```

8. Instalar algunas herramientas para configuración de la red

```
yum install net-tools
```

9. Instalar el editor Vim

```
yum install vim
```

10. Repita los pasos 5 a 8 para la maquina cliente.
11. Confirme la ip de las maquinas virtuales usando `ifconfig` y pruebe conectividad con el comando `ping`.

2. Parte 2: Publicar Boxes en Vagrant Cloud

Una vez modificada la maquina virtual podemos crear un nuevo box y subirlo a Vagrant Cloud para usarlo posteriormente. Para eso seguiremos los siguientes pasos:

1. Re empaquetar la maquina virtual en un nuevo Vagrant Box

```
vagrant package servidor --output mynew.box
```

2. Agregar el box creado a su instalación de Vagrant.

El comando anterior creara un archivo mynew.box. Con el siguiente comando agregaremos el box a nuestra instalación de Vagrant:

```
vagrant box add mynewbox mynew.box
```

Esto permitirá usar el box desde cualquier ubicación en su computador.

3. Publique el box en Vagrant Cloud

- Diríjase a <https://app.vagrantup.com/boxes/new>
- Ingrese un nombre y una descripción para su box
- Cree la primera versión del box. Esta versión debe cumplir con el formato [0-9].[0-9].[0-9]. Por ejemplo 0.0.1.
- Cree un provider para el box. Virtualbox es el provider mas común.
- Cargue el archivo .box que corresponde al provider creado
- Una vez cargado el box puede encontrarlo en la sección de boxes de <https://app.vagrantup.com/>
- Antes de usar una versión del box, debe liberarlo “release”
- Una vez creado y liberado un box, puede liberar nuevas versiones dando click en “create new version” en el menú de versiones de la pagina del box.

Referencias

Creating a New Vagrant Box. <https://atlas.hashicorp.com/help/vagrant/boxes/create>

Ejercicios:

PARTE A

1. Crear el ambiente de trabajo inicial con una maquina cliente y una maquina servidor, de acuerdo al Vagrantfile mostrado en la seccion 4.
2. Instalar net-tools y vim
3. Subir la imagen modificada a Vagrant Cloud
4. Realizar el taller Linux disponible en Classroom
5. Investigue en qué consisten los **directorios sincronizados de Vagrant**. Demuestre su funcionamiento.

PARTE B

1. Investigue cómo funcionan los repositorios git.
 - a. Revise el video de YouTube sobre los pasos básicos para empezar a usar GitHub:
<https://www.youtube.com/watch?v=SqbGliTKVoE&t=39s>

- b. Si no lo tiene, cree un repositorio git en github (<https://github.com/>) y utilícelo desde su máquina virtual "servidor" para subir y descargar archivos.
- c. Cree una estructura de directorios en la cual subirá los archivos relacionados con cada una de sus prácticas a lo largo del Semestre. Por Ejemplo:

```
└─ servicios
   └─ Practica0_AmbienteTrabajo
   └─ Practica1_FTP
   └─ Practica2_HTTP
```

Nota: Recuerde primero instalar el cliente git en su máquina servidor.

REFERENCIAS

Vagrant. <https://www.vagrantup.com/>

GitHub. <https://github.com/>

Tutorial GIT. <https://git-scm.com/docs/gittutorial>

Ejercicios Linux. <https://learnpythonthehardway.org/book/appendix-a-cli/ex1.html>