Experiment-1

Lab Exercise: Introduction to Vagrant and Vagrantfile

Name: Aditya Tomar
SAP: 500106015
R.no.: R2142221060
Batch: 2(DevOps)

This exercise will guide them through setting up a virtual environment using Vagrant, configuring the environment via a Vagrantfile, and managing the virtual machines (VMs) with basic Vagrant commands.

Objective:

- Learn how to set up and configure virtual environments using Vagrant.
- Understand the structure and components of a Vagrantfile.
- Gain hands-on experience in managing virtual machines using Vagrant commands.

Prerequisites:

- Basic knowledge of virtualization concepts.
- Familiarity with command-line interfaces.
- Installation of Vagrant and VirtualBox (or any other supported provider) on your local machine.

Step-by-Step Exercise:

1.Setting Up the Environment:

Install Vagrant:

- Download and install Vagrant from the official website.
- Ensure you have VirtualBox installed as it is a commonly used provider with Vagrant.
- Verify Installation:
- Open a terminal or command prompt.
- Run the following commands to verify the installation:

```
vagrant –version
```

2. Creating a New Vagrant Project:

- Create a Project Directory:
- In your terminal, create a new directory for your Vagrant project and navigate into it:

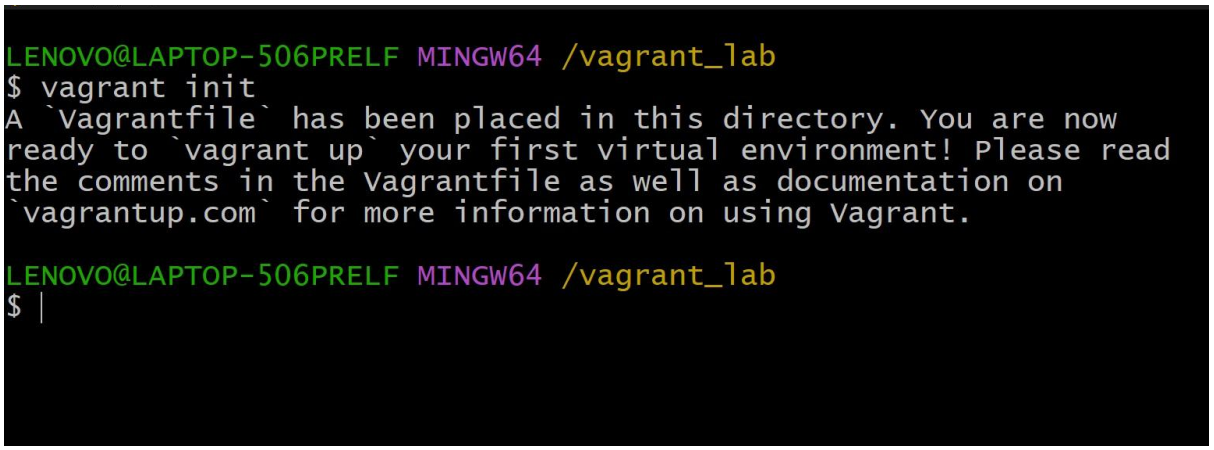
```
mkdir vagrant_lab
cd vagrant_lab
```

Initialize Vagrant:

- Run the following command to initialize a new Vagrantfile in your project directory:

```
vagrant init
```

This command will generate a Vagrantfile in the current directory.



3. Understanding the Vagrantfile:

- Open the Vagrantfile:
- Open the Vagrantfile in a text editor of your choice.
- The Vagrantfile is a Ruby-based configuration file used to define the virtual environment.
- Basic Vagrantfile Configuration:
- Modify the Vagrantfile to configure a basic virtual machine. For example:

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

  config.vm.box = "ubuntu/bionic64" # Specifies the base box to use (Ubuntu 18.04) config.vm.network "private_network", type: "dhcp" # Configures a private network config.vm.provider "virtualbox" do |vb|
```

```
    vb.memory = "1024" # Allocates 1GB of RAM to the
VM end
end
```

4. Launching and Managing the VM:

Start the VM:

In the terminal, start the VM using the following command:

```
vagrant up
```

```
LENOVO@LAPTOP-5O6PRELF MINGW64 /vagrant_lab
$ vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Box 'ubuntu/xenial64' could not be found. Attempting to find and install...
    default: Box Provider: virtualbox
    default: Box Version: >= 0
==> default: Loading metadata for box 'ubuntu/xenial64'
    default: URL: https://vagrantcloud.com/api/v2/vagrant/ubuntu/xenial64
==> default: Adding box 'ubuntu/xenial64' (v20211001.0.1) for provider: virtualbox
    default: Downloading: https://vagrantcloud.com/ubuntu/boxes/xenial64/versions/20211001.0.1/providers/virtu
albox/unknown/vagrant.box
Download redirected to host: cloud-images.ubuntu.com
Progress: 12% (Rate: 176k*/s, Estimated time remaining: 0:25:25)

Progress: 23% (Rate: 167k*/s, Estimated time remaining: 0:21:50)

    default:
==> default: Successfully added box 'ubuntu/xenial64' (v20211001.0.1) for 'virtualbox'!
==> default: Importing base box 'ubuntu/xenial64'...
==> default: Matching MAC address for NAT networking...
==> default: Checking if box 'ubuntu/xenial64' version '20211001.0.1' is up to date...
==> default: Setting the name of the VM: vagrant_lab_default_1723445417679_44014
Vagrant is currently configured to create VirtualBox synced folders with
the `SharedFoldersEnableSymlinksCreate` option enabled. If the Vagrant
guest is not trusted, you may want to disable this option. For more
information on this option, please refer to the VirtualBox manual:

  https://www.virtualbox.org/manual/ch04.html#sharedfolders

This option can be disabled globally with an environment variable:

  VAGRANT_DISABLE_VBOXSYMLINKCREATE=1

or on a per folder basis within the Vagrantfile:

  config.vm.synced_folder '/host/path', '/guest/path', SharedFoldersEnableSymlinksCreate: false
==> 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: Running 'pre-boot' VM customizations...
==> default: Booting VM...
==> default: Waiting for machine to boot. This may take a few minutes...
    default: SSH address: 127.0.0.1:2222
    default: SSH username: vagrant
    default: SSH auth method: private key
```

Vagrant will download the specified box (if not already downloaded) and launch the VM.

- SSH into the VM:
- Connect to the running VM using SSH:

vagrant ssh

```
LENOVO@LAPTOP-506PRELF MINGW64 /vagrant_lab
$ vagrant ssh
Welcome to Ubuntu 16.04.7 LTS (GNU/Linux 4.4.0-210-generic x86_64)

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

UA Infra: Extended Security Maintenance (ESM) is not enabled.

0 updates can be applied immediately.

45 additional security updates can be applied with UA Infra: ESM
Learn more about enabling UA Infra: ESM service for Ubuntu 16.04 at
https://ubuntu.com/16-04

New release '18.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.




vagrant@ubuntu-xenial:~$
vagrant@ubuntu-xenial:~$
vagrant@ubuntu-xenial:~$
vagrant@ubuntu-xenial:~$
vagrant@ubuntu-xenial:~$ touch cxsffjhs
vagrant@ubuntu-xenial:~$ ls
cxsffjhs
vagrant@ubuntu-xenial:~$
```

- This command will log you into the VM's shell.
- Exploring the VM:
- Inside the VM, explore the filesystem, install packages, and run commands to understand the environment.
- Stop the VM:
- Exit the SSH session by typing exit.
- Stop the VM with the following command:

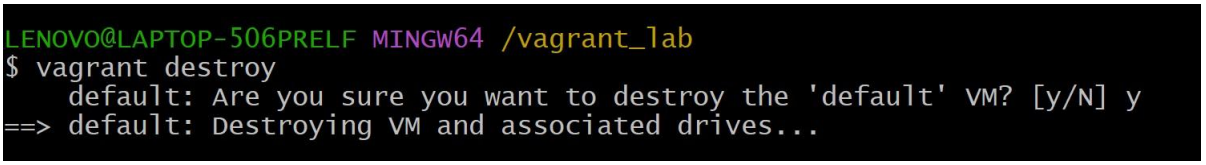
vagrant halt

```
LENOVO@LAPTOP-506PRELF MINGW64 /vagrant_lab
$ vagrant halt
==> default: Attempting graceful shutdown of VM...
```

Destroy the VM (optional):

To remove the VM completely, use the following command:

```
vagrant destroy
```



This will remove all traces of the VM, including any data stored on it.

Explore the benefits of using Vagrant for development and testing environments.

Submission:

- Submit a brief report including the Vagrantfile you configured, screenshots of the running VM, and the output of any commands run within the VM.
- Reflect on the learning experience and any challenges faced during the exercise.

This lab exercise provides a hands-on introduction to Vagrant, focusing on creating and managing virtual environments through a Vagrantfile. It offers both foundational learning and opportunities to explore more advanced features.