

EXPERIMENT-1

Introduction to Vagrant and Vagrantfile

Lab Exercise: Introduction to Vagrant and Vagrantfile

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.

Setting up Environment

- Download and Install Vagrant from Hashicorp.
- Ensure a hypervisor is installed in the system.
- Verify installation of vagrant by checking version of vagrant in terminal.

```
C:\Users\HP 15>vagrant --version
Vagrant 2.4.1
```

Creating a Vagrant Environment and Initialize Vagrant

- Create a new folder.
- Initialize Vagrant file.

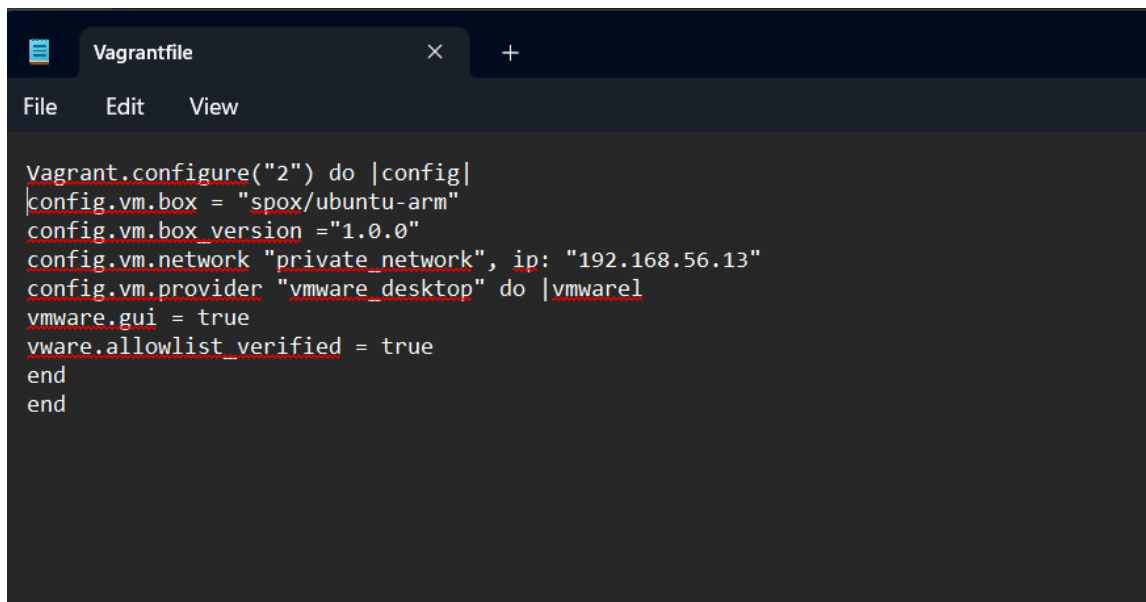
```
C:\Users\HP 15>cd C:\Users\HP 15\Desktop\Sem5\Docker\Exp1

C:\Users\HP 15\Desktop\Sem5\Docker\Exp1>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.
```

- A Vagrantfile will be generated.

Editing Vagrantfile

- Vagrantfile is a configuration file. It specifies how virtual environment is to be setup.
- Open it in a text editor.
- Modify Vagrantfile according to need.

A screenshot of a text editor window titled 'Vagrantfile'. The editor has a menu bar with 'File', 'Edit', and 'View'. The content of the file is a Vagrant configuration script in Ruby. It defines a virtual machine named '2' using the 'spox/ubuntu-arm' box with version '1.0.0'. The network is set to 'private_network' with IP '192.168.56.13'. The provider is 'vmware_desktop', and the GUI is enabled. The 'allowlist_verified' flag is set to true. The script ends with 'end' and 'end'.

4. Launching and Managing the VM:

Start the VM:

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

vagrant up

```
$ vagrant up
Bringing machine 'default' up with 'virtualbox' provider...
==> default: Box 'ubuntu/trusty64' could not be found. Attempting to find and install...
    default: Box Provider: virtualbox
    default: Box Version: >= 0
==> default: Loading metadata for box 'ubuntu/trusty64'
    default: URL: https://vagrantcloud.com/api/v2/vagrant/ubuntu/trusty64
==> default: Adding box 'ubuntu/trusty64' (v20191107.0.0) for provider: virtualbox
    default: Downloading: https://vagrantcloud.com/ubuntu/boxes/trusty64/versions/20191107.0.0/providers/virtual
box/unknown/vagrant.box
==> default: Box download is resuming from prior download progress
Download redirected to host: cloud-images.ubuntu.com
    default:
==> default: Successfully added box 'ubuntu/trusty64' (v20191107.0.0) for 'virtualbox'!
==> default: Importing base box 'ubuntu/trusty64'...
==> default: Matching MAC address for NAT networking...
==> default: Checking if box 'ubuntu/trusty64' version '20191107.0.0' is up to date...
==> default: Setting the name of the VM: vagrant_lab_default_1724991286583_46234
==> default: Clearing any previously set forwarded ports...
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:

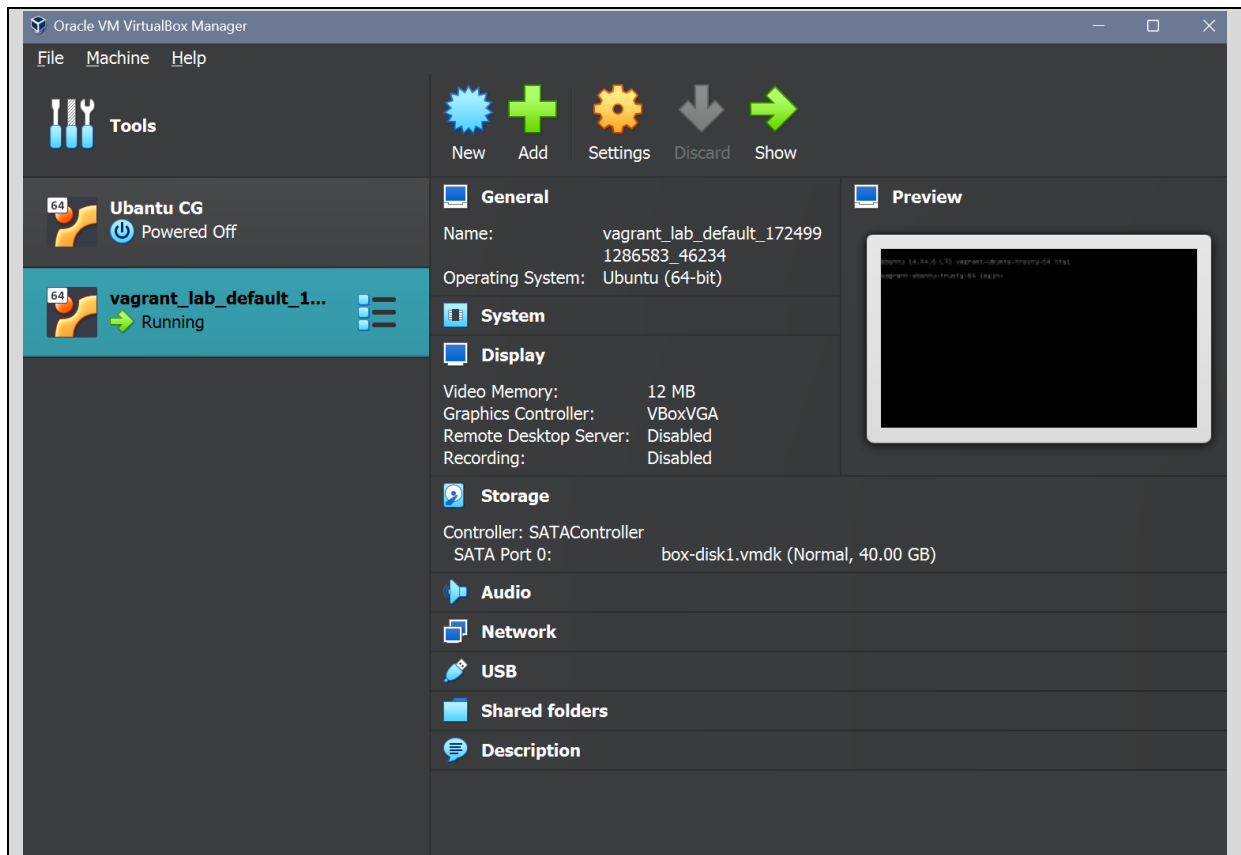
    https://www.virtualbox.org/manual/ch04.html#sharedfolders

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)
```



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

```
$ vagrant ssh
Welcome to Ubuntu 14.04.6 LTS (GNU/Linux 3.13.0-170-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

System information disabled due to load higher than 1.0

UA Infrastructure Extended Security Maintenance (ESM) is not enabled.

0 updates can be installed immediately.
0 of these updates are security updates.

Enable UA Infrastructure ESM to receive 64 additional security updates.
See https://ubuntu.com/advantage or run: sudo ua status

New release '16.04.7 LTS' available.
Run 'do-release-upgrade' to upgrade to it.
```

- 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
```

```
$ vagrant halt
==> default: Attempting graceful shutdown of VM...
```

Destroy the VM (optional):

To remove the VM completely, use the following command:

```
vagrant destroy
```

```
$ vagrant destroy
default: Are you sure you want to destroy the 'default' VM? [y/N] y
==> default: Destroying VM and associated drives...
```



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

