Vagrant 101 Workshop

You will learn (WIIFM)

- What environments are
- What Infrastructure as a Code (IaaC) is
- How to use Vagrant to spin off an environment

Requirements

- Basic knowledge of Linux

System Requirements

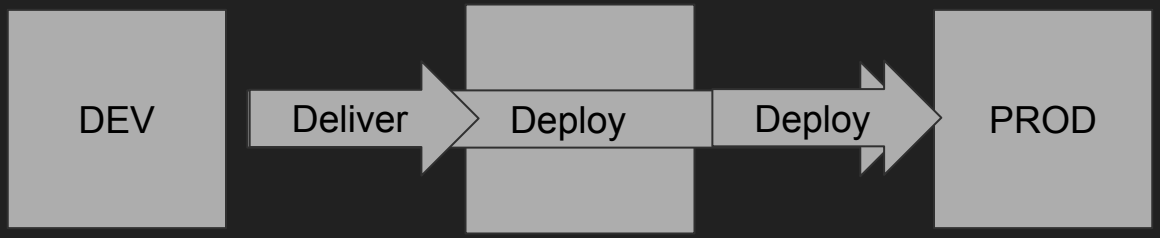
- Latest version of Vagrant installed
- Latest version of VirtualBox installed

Agenda

- Environment
- Vagrant introduction
- VirtualBox introduction
- Commands
- Workflow

Environment

Deployment Pipeline



Our DEV environment

- Got a Laptop
- Installed: Python, pip, virtualenv
- Checked out code from GitHub

- Hardware + OS
- Middleware
- Application


AUTOMATE

What Vagrant Is?

Building and Managing VM environments in a single,easy-to-use workflow

https://www.vagrantup.com/

VAGRANT

From Top

- Command line tool
- Providers: Virtualbox, VMWare, Hyper-V, AWS, GCP, Azure ….
- Provisioners: shell, Chef, Ansible, Puppet ….
- Runs on: Linux, Windows, MacOS

VirtualBox

- A general purpose full virtualizer for x86 hardware
- Free
- Open Source
- Available on every major platform

https://www.virtualbox.org

Getting up and running

Did you know?
Hashicorp is Vagrant vendor
Xenial = Ubuntu 16.04

```
$ vagrant init ubuntu/xenial64
$ vagrant up
$ vagrant ssh
```

Did you know?
ssh - Secure Shell. Used to log onto remote systems

Command: vagrant init

Places a Vagrantfile is current directory

```
# -*- mode: ruby -*-
# vi: set ft=ruby :
Vagrant.configure("2") do |config|
      config.vm.box = "ubuntu/xenial64"
end
```

Run the init command in your project root directory. Many configuration options are relative to the project root folder

ubuntu/xenial64 - a box
A box is a base image

Command: vagrant up

- The box is downloaded and stored locally in ~/.vagrant.d/boxes
- A new VM is created and bootstrapped with the box
- The VM is booted and provisioned

Command: vagrant ssh

- Log onto the VM

Did you know?
You logon with the user ‘vagrant’ which was created by vagrant during the VM bootstrapping

Command: vagrant destroy

- Remove all traces of the guest machine from the host machine

Command: vagrant box add

- Catalog - https://app.vagrantup.com/boxes/search
- Stored locally in ~/.vagrant.d/boxes

```
$ vagrant box add ubuntu/trusty64
```

Did you know?
Trusty = Ubuntu 14.04

Do you remember?
The base box configuration option is in the Vagrantfile

Exercise

- Check which Ubuntu version is running - ‘lsb_release -a’
- Destroy the VM
- Spin up a new VM based on trusty64
- Check which Ubuntu version is running NOW

/vagrant - synced folder

- By default, the project root folder (host) is synced with the /vagrant folder (guest)
- Two way sync
- It is not vagrant home directory

Provider - VirtualBox configuration

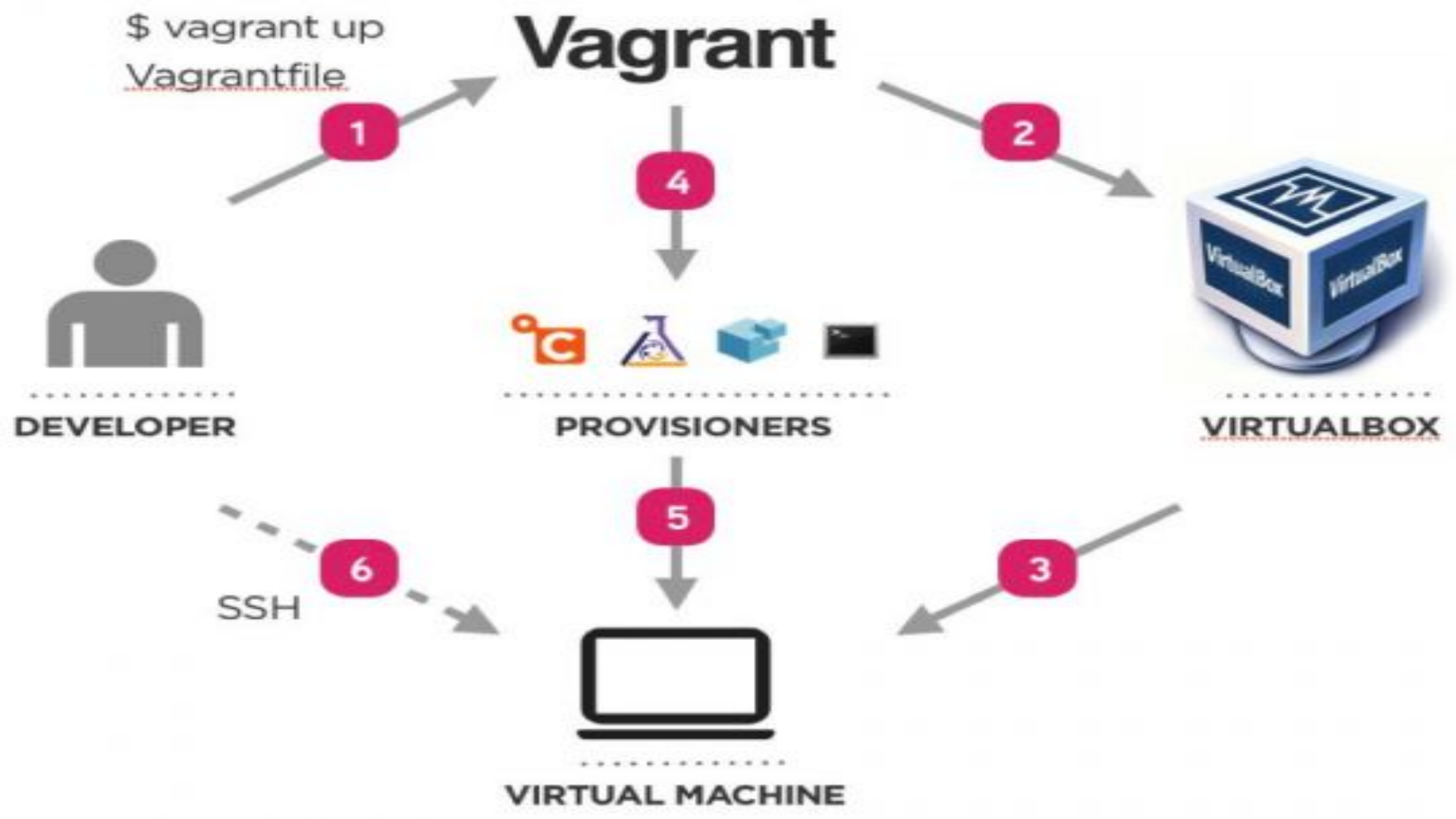
```
# -*- mode: ruby -*-
# vi: set ft=ruby :
Vagrant.configure("2") do |config|
    config.vm.box = "ubuntu/xenial64"
    config.vm.provider "virtualbox" do |v|
        v.name = "my_vm"
        v.memory = 2048
        v.cpus = 4
    end
end
```

Exercise

- Destroy current VM
- Spin up a new VM with different memory and cpu settings
- Check the new VM

Recap

- vagrant init user/box
- vagrant up
- vagrant destroy
- vagrant ssh
- vagrant box add user/box


$ vagrant up
Vagrantfile
Vagrant
1
2
3
4
5
6
SSH
DEVELOPER
PROVISIONERS
VIRTUALBOX
VIRTUAL MACHINE

shell provisioner

In the following Vagrant workshop