DevOps

Vagrant – Introduction, Installation & Provisioning

Collabera®
Value. Accelerated.

Collabera®
TACT
360° Training

Trainer: Abhijith V G – AWS, eCommerce, Mobile & DevOps Architect

Magento Certified DEVELOPER PLUS

amazon web services | Certified

Developer - Associate

VAGRANT: Introduction: Why and What is Vagrant

Vagrant provides easy to configure, reproducible, and portable work environments built on top of industry-standard technology and controlled by a single consistent workflow to help maximize the productivity and flexibility of you and your team.

To achieve its magic, Vagrant stands on the shoulders of giants. Machines are provisioned on top of VirtualBox, VMware, AWS, or any other provider. Then, industry-standard provisioning tools such as shell scripts, Chef, or Puppet, can automatically install and configure software on the virtual machine.

VAGRANT

VAGRANT: Introduction: Uses of Vagrant in an Environment

For Developers

If you are a developer, Vagrant will isolate dependencies and their configuration within a single disposable, consistent environment, without sacrificing any of the tools you are used to working with (editors, browsers, debuggers, etc.). Once you or someone else creates a single Vagrantfile, you just need to vagrant up and everything is installed and configured for you to work. Other members of your team create their development environments from the same configuration, so whether you are working on Linux, Mac OS X, or Windows, all your team members are running code in the same environment, against the same dependencies, all configured the same way. Say goodbye to "works on my machine" bugs.

VAGRANT: Introduction: Uses of Vagrant in an Environment

For Operators

If you are an operations engineer or DevOps engineer, Vagrant gives you a disposable environment and consistent workflow for developing and testing infrastructure management scripts. You can quickly test things like shell scripts, Chef cookbooks, Puppet modules, and more using local virtualization such as VirtualBox or VMware. Then, with the same configuration, you can test these scripts on remote clouds such as AWS or RackSpace with the same workflow.

VAGRANT: Introduction: Uses of Vagrant in an Environment

For Designers

If you are a designer, Vagrant will automatically set everything up that is required for that web app in order for you to focus on doing what you do best: design. Once a developer configures Vagrant, you do not need to worry about how to get that app running ever again. No more bothering other developers to help you fix your environment so you can test designs. Just check out the code, vagrant up, and start designing.

VAGRANT: Introduction: Uses of Vagrant in an Environment

For Everyone

Vagrant is designed for everyone as the easiest and fastest way to create a virtualized environment!

VAGRANT: Introduction: Alternatives of Vagrant

Vagrant vs. CLI Tools

Virtualization software like **VirtualBox** and **VMware** come with command line utilities for managing the lifecycle of machines on their platform. Many people make use of these utilities to write their own automation. Vagrant actually uses many of these utilities internally.

The difference between these CLI tools and Vagrant is that Vagrant builds on top of these utilities in a number of ways while still providing a consistent workflow.

Other than this **Docker** and **Terraform** are other tools competing with Vagrant!

VAGRANT: Introduction: Versions of Vagrant

vagrant --version
Vagrant 1.9.3

vagrant version
Installed Version: 1.9.3
Latest Version: 2.0.1

To upgrade to the latest version: https://www.vagrantup.com/downloads.html

In **Vagrantfile**
Vagrant.require_version ">= 1.9.3”
Vagrant.require_version ">= 1.8.5", "< 1.9.0"

VAGRANT: Introduction: Versions of Vagrant

Configuration Version (For backward compatibility)

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

Currently, there are only two supported versions: "1" and "2". Version 1 represents the configuration from Vagrant 1.0.x. "2" represents the configuration for 1.1+ leading up to 2.0.x.

VAGRANT: Installation and Configuration of Virtual Box

https://www.virtualbox.org/wiki/Downloads

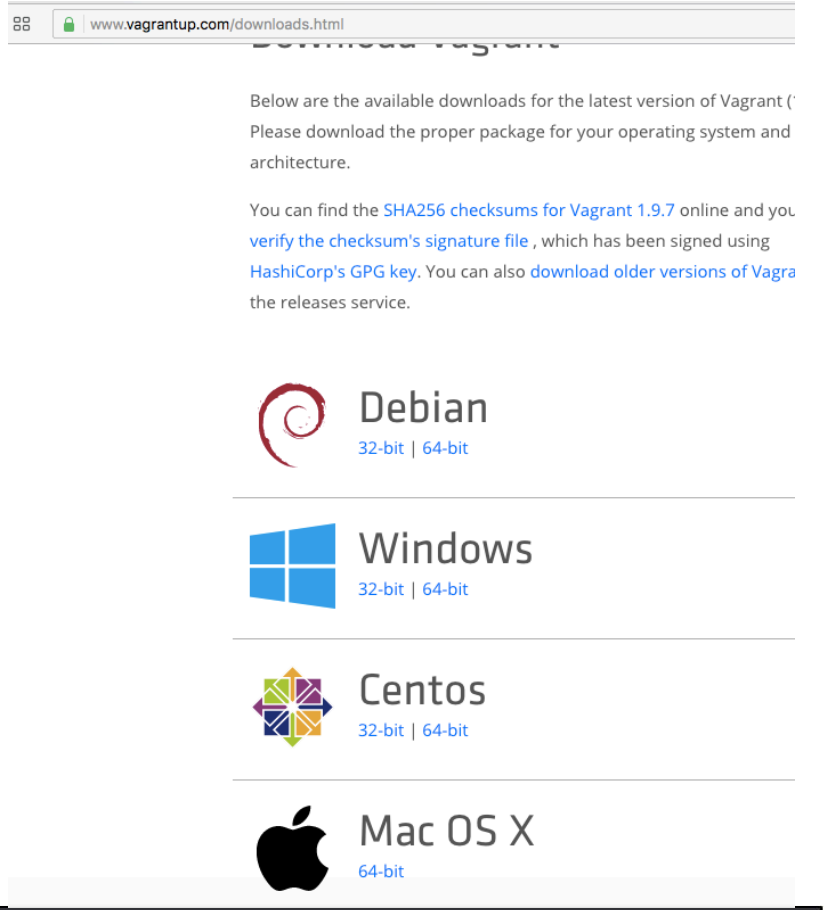
VAGRANT: Installation and Configuration of Virtual Box

https://www.virtualbox.org/wiki/Downloads

VAGRANT: Installation: Configuring Vagrant

https://www.vagrantup.com/downloads.html

The installer will automatically add vagrant to your system path so that it is available in terminals.



VAGRANT: Installation: Installing Putty and GitBash

http://www.putty.org

https://git-scm.com/downloads

VAGRANT: Provisioning with Vagrant: Creating first VM

vagrant init

Initialize Vagrant with a Vagrantfile and ./.vagrant directory, using no specified base image. Before you can do vagrant up, you'll need to specify a base image in the Vagrantfile.

vagrant init boxpath

Initialize Vagrant with a specific box. To find a box, go shopping. When you find one you like, just replace it's name with boxpath.

For example
vagrant init chef/centos-6.5
vagrant init ubuntu/trusty64

VAGRANT: Provisioning with Vagrant: Operations on VM

vagrant up #starts vagrant environment (also provisions only on the FIRST vagrant up) Equivalent to pressing the power buttons on your servers.
vagrant status #outputs status of the vagrant machine
vagrant halt #stops the vagrant machine
vagrant reload #restarts vagrant machine, loads new Vagrantfile configuration
vagrant provision #forces reprovisioning of the vagrant machine
vagrant ssh #connects to machine via SSH
vagrant destroy #stops and deletes all traces of the vagrant machine
vagrant suspend #Suspends a virtual machine (remembers state)
vagrant resume #Resume a suspended machine (vagrant up works just fine for this as well)
vagrant reload --provision #Restart the virtual machine and force provisioning

VAGRANT: Provisioning with Vagrant: Connecting to the VM

Login
vagrant ssh <name>

vagrant port displays information about guest port mappings

VAGRANT: Provisioning with Vagrant: Adding Images

Vagrant Boxes are prebuilt VM images. (You never modify your box images)

vagrant box list #List the installed boxes
vagrant box add <name> <box path/HTTP URI> #Add the box for later use
vagrant box remove <name> virtualbox #delete a box
vagrant box outdated #Check for updates vagrant box update
vagrant box update … # Add newest version

This URL has a list of common boxes you can use:
https://app.vagrantup.com/boxes/search

VAGRANT: Provisioning with Vagrant: Manually Install Apache2

```
mkdir t1
cd t1
vagrant init ubuntu/trusty64
vagrant up
vagrant ssh
sudo apt-get update
sudo apt-get install –y apache2
curl localhost
```

Additional experiments… create a new file and confirm that it it browsable
Check if the webserver is accessible from outside the VM
Port forwarding to enable access
Destroy and up again and see what happens

VAGRANT: Provisioning with Vagrant: Automated Apache2 install

Method 1: Inline shell script in the Vagrantfile

```
config.vm.provision "shell", inline: <<-SHELL
    apt-get update
    apt-get install -y apache2
 SHELL
```

Method 2: Path to the shell script (relative to where your Vagrantfile is).

```
config.vm.provision "shell", path: "path to your shell script"
```

To run this either do: vagrant destroy and then vagrant up

Or you can do: vagrant provision

VAGRANT: Provisioning with Vagrant: Automated Apache2 install

bootstrap.sh in the same directory as your Vagrantfile:

```
#!/usr/bin/env bash
apt-get update
apt-get install -y apache2
if ! [ -L /var/www/html ]; then
  rm -rf /var/www/html
  ln -fs /vagrant /var/www/html
fi
```

NOTE: This command will map the /vagrant volume to /var/www/html

ln -fs /vagrant /var/www/html

At the end of the Vagrantfile before “end”

```
config.vm.box = "hashicorp/precise64”
config.vm.provision :shell, path: "bootstrap.sh”
config.vm.network :forwarded_port, guest: 80, host: 4567
```

VAGRANT: Home work

Install VirtualBox
Install Vagrant
Install Putty (only for Windows)
Install GitBash (Only for Windows)

Create an Ubuntu/Trusty64 Linux box
SSH into the box
Install Apache2
Install Apache with your custom index page loaded automatically
Destroy and Up again to confirm that the state is maintained

(The web server should be brows-able from outside the VM)

For technical support:

Call : +91 75730 27611

E-mail : tactsupport@collabera.com

Visit us at : http//:www.collaberatact.com

Collabera