

vagrant | 'vāgrənt
a person without a settled
home or regular work who
wanders from place to place
and lives by begging

@CraigMcEldowney
+
@SteveRifkin
=
@crifkin

- Ruby-powered command line interface to VirtualBox
- Deployment environment for local virtual environments

According to those who know:

"Create and configure lightweight, reproducible and portable environments."

Vagrant: What is it?

- Virtualized development environment
 - aka Stop installing client crap on your local machine
- Built-in dev environment sandboxing
 - Client A stops messing with Client B
 - Client A's hipster retro version of Varnish 1.2.x doesn't trump Client B's awesome future branch of Varnish 4.x
- Multi-VM Host Environment
 - aka Run a full production stack on your local machine for testing
- Package Virtual Environments
 - aka Sick of troubleshooting a fellow dev's environment—WHY DOESN'T SOLR WORK!@\$@!#-- just give them the entire environment

Vagrant: Why? (Use Cases)

- Download and install VirtualBox
 - <https://www.virtualbox.org/wiki/Downloads>
- Download and install Vagrant
 - <http://vagrantup.com>
- Download base Box (aka AMI, Base Image, etc)
 - vagrant box add precise64 <http://files.vagrantup.com/precise64.box>
 - Community-curated list of Vagrant boxes: <http://www.vagrantbox.es/>

Vagrant: What?

- My First Vagrant machine
- `mkdir mymachine`
- `cd mymachine`
- `vagrant init precise64`
 - Creates a general Vagrantfile
- `vagrant up`
 - Create/spin up virtual machine instance
- `vagrant ssh`
 - Connect to virtual machine instance

My First Vagrant...

- `config.vm.box = "precise64"`
 - Base image your server built on.
 - This association only exists on first spin-up, then this instance becomes its own standalone
- `config.vm.network :hostonly, "192.168.100.10"`
 - Specify an IP address that can be used by other Virtual Machines in this VirtualBox environment
 - http://vagrantup.com/v1/docs/host_only_networking.html
 - No Security built-in. All ports open!
- `config.vm.network :bridged`
 - Make VM appear as a physical device on your current network
 - http://vagrantup.com/v1/docs/bridged_networking.html
- `config.vm.forward_port 80,8080`
 - Forward port from VM to localhost, e.g. I go to localhost:8080 in my browser to see VM-hosted website

My Vagrantfile: Basic

- Super simple portability:
cd myserver
vagrant package --output grammas-present.box

Give resulting grammas-present.box to your closest friends and they can spin up the environment

To use on remote machine:

- Copy Vagrant box locally
- vagrant box add grammas-present.box
- vagrant init grammas-present
- vagrant up

Pack Your Box and Hit the Road

- vagrant reload
 - Reload settings from Vagrantfile, run any provisioners (more later)
- Need to rerun config?
 - Just tear it down and start again
 - vagrant destroy && vagrant up

Tweak existing Vagrant VM

- vagrant suspend
 - Save the current running state of VM and then stop it!
 - Resume working with resume
- Vagrant halt
 - Graceful shutdown of machine
 - Resume working with vagrant up
- Vagrant destroy
 - I hate this machine. I will destroy it.
 - Resume working (from scratch) with vagrant up

I'm done with this Vagrant box

- Vagrant automatically shares the root directory for vagrant file to VM at /vagrant
- `mkdir myserver && cd myserver`
- `vagrant init precise64`
- `vagrant up`
- `touch hi.txt`
- `vagrant ssh`
- `> ls -al /vagrant`
 - You'll see hi.txt smiling back at you.

Other Awesome: NFS Mounts

- Use other tools to provision the box:
 - Chef-solo
 - Chef-server
 - Puppet
 - PuppetServer
 - Shell
 - Others...

Now the fun stuff-- Provisioning

```
config.vm.provision :chef_solo do |chef|
  chef.cookbooks_path = "../my-recipes/
  cookbooks"
  chef.roles_path = "../my-recipes/roles"
  chef.data_bags_path = "../my-recipes/
  data_bags"
  chef.add_recipe "mysql"
  chef.add_role "web"
  # You may also specify custom JSON attributes:
  # chef.json = { :mysql_password => "foo" }
end
```

Provisioning: Chef-Solo

- Define Chef Server URL, validation key path, run list, client name, environment...
- Override any node attributes!!

```
chef.json.merge!({
  :cloud => { :private_ips => "192.168.100.11" },
  :drupal => {
    :db => {
      :database => "crifkin",
      :host => "192.168.100.100",
      :user => "drupal",
      :password => "drupal1"
    }
  }
})
```

Provisioning: Chef-server

- Remember to delete client/node between runs
- Vagrant provision == chef-client
 - OR `chef-client -c /tmp/vagrant-chef-1/client.rb`
- `ipaddress`
 - Vagrant assigns internal IP address to `eth0`
 - Host-only address is assigned, but not saved to `ohai` var for `ipaddress`
 - Work-around: Overwrite desired `ohai` value with `chef.json.merge!({`
 `:cloud => {`
 `:private_ips => "192.168.100.11"`
 `})`

Chef-Server Gotchas

```
Vagrant::Config.run do |config|
  config.vm.define :web do |web_config|
    web_config.vm.box = "web"
    web_config.vm.forward_port 80, 8080
  end
  config.vm.define :db do |db_config|
    db_config.vm.box = "db"
    db_config.vm.forward_port 3306, 3306
  end
end
End
```

- To spin up:
- cd mystack
- vagrant up
- OR do them one at a time with
 - Vagrant up db
 - Vagrant up web

More fun... Multiple VMs!!!

- Vagrant up doesn't always work for inexplicable reasons
 - That's why there is:
 - vagrant destroy -f && vagrant up
- VirtualBox sometimes gets overzealous in creating backups/clones
 - Check ~/VirtualBox VMs for any folders not matching actual Virtual Machines and clean out periodically
- Documentation is good, but sparse. No comprehensive resource yet on all Vagrantfile params, etc...
- Vagrants are known to steal cats.
 - Just something I heard on the interwebs

General Vagrant Gotchas

- Basic Tutorial:
<http://vagrantup.com/v1/docs/getting-started/index.html>
- General Docs:
<http://vagrantup.com/v1/docs/index.html>
- Some great use cases and advanced tutorial
<http://devops.me/2011/10/05/vagrant/>
- <http://beacon.wharton.upenn.edu/404/2011/12/keeping-your-machine-clean-with-vagrant-chef/>
- <http://lumberjaph.net/misc/2010/11/22/vagrant-rocks.html>

Awesome Links

- Fancy box packaging
 - <http://devops.me/2011/10/06/building-baseboxes/>
 - <http://wiki.opscode.com/display/chef/Vagrant>
- `postinstall.sh`
 - Custom scripts to run after spin-up
 - Cheap/easy way to configure the box other than chef/other sources...
 - Create DBs, load files, etc... true end-to-end bootstrap

Advanced topics