**Install Vagrant**

Lab 1 – Download and Install Vagrant (login- vagrant,password-vagrant)

1. Install VirtualBox (Oracle)
   1. <https://www.vagrantup.com/downloads.html> or <https://releases.hashicorp.com/vagrant/2.0.0/vagrant_2.0.0_x86_64.msi?_ga=2.192368618.1360667834.1506531603-1264789848.1506531603>
2. Download and add couple of images to vagrant
   1. Download the CentOS 6.5 Vagrant box to the puppet-fundamentals-lab directory: https://www.dropbox.com/s/206lcenz3o45c79/centos65.box?dl=1 or

http://benpiper.github.io/course-files/puppet-system-administrators-fundamentals/centos65.box.torrent

* 1. Ensure the file is saved as centos65.box.
  2. Add the CentOS 6.5 box you just downloaded:
     1. vagrant box add centos65-base centos65.box
  3. Download the Ubuntu 14.04 Vagrant box to the same directory: http://cloud-images.ubuntu.com/vagrant/trusty/current/trusty-server-cloudimg-amd64-vagrant-disk1.box
  4. Add the Ubuntu box:
     1. vagrant box add trusty64 trusty-server-cloudimg-amd64-vagrant-disk1.box

**Starting with Chef**

Lab 1 – Starting with ChefDK

1. Install VirtualBox (Oracle)
2. Verify installation (add c:\program files\oracle\virtualbox in path)
   1. VBoxManage -version
3. Install Vagrant
   1. (done in earlier lab)
   2. Vagrant –version (add c:\hashicorp\vagrant\bin in path)
4. Download an Ubuntu Vagrant Box and install chefDK
   1. vagrant box add ubuntu/trusty64
   2. in C:\demo\_chef run following command
   3. vagrant init ubuntu/trusty64
      1. this will create a vagrantfile in that folder
   4. vagrant up
   5. vagrant ssh
   6. on shell
      1. sudo apt-get update
      2. sudo apt-get -y install curl
      3. curl https://omnitruck.chef.io/install.sh | sudo bash -s -- -P chefdk -c stable -v 2.0.28
      4. chef --version
      5. Text editor (emacs or nano or vim.)
         1. sudo apt-get install nano
   7. vagrant destroy --force

Lab 2 – Configuring Resources

1. Setup working directory (ssh)
   1. mkdir ~/chef-repo
   2. cd ~/chef-repo
   3. create hello.rb in chef-repo

file '/tmp/hello.txt’ do

content 'hello world'

end

* 1. sudo chef-apply hello.rb
  2. cat /tmp/hello.txt

1. modify hello.rb to change content and run the same command again and observe messages + modified files
2. try creating bye.rb and test

file '/tmp/motd' do

action :delete

end

Lab 3 – Configuring packages / Services

<https://learn.chef.io/modules/learn-the-basics/ubuntu/virtualbox/configure-a-package-and-service#/>

**Starting with Puppet**

Lab 1 – Starting with Puppet Server

1. Download above file and extract it on c:\demo\_puppet
2. Go into the directory of puppetmaster
   1. Vagrant up
   2. Vagrant ssh
      1. Sudo yum -y install nano git ntp(timing)
      2. Sudo service ntpd start
      3. Sudo chkconfig ntpd on
      4. Sudo chkconfig | grep ntpd
   3. Install nano, Git, NTP
3. Add the puppet repository
   1. Sudo yum -y install http://yum.puppetlabs.com/puppetlabs-release-el-6.noarch.rpm
4. Install the puppet server
   1. Sudo yum -y install puppet-server
   2. Puppet master –version
   3. Create a production environment
      1. Sudo mkdir -p /etc/puppet/environment/production/{modules,manifests}
      2. Create environment.conf in production folder

modulepath = /etc/puppet/environments/production/modules

environment\_timeout = 5s

* + 1. Cd /etc/puppet
    2. Sudo nano puppet.conf
    3. Modify to include following section

[master]

environmentpath=$confdir/environments

basemodulepath = $confdir/modules:/opt/puppet/share/modules/

* + 1. Start with installation
       1. sudo yum -y install httpd httpd-devel mod\_ssl ruby-devel

rubygems, gcc gcc-c++ libcurl-devel openssl-devel

sudo gem install rubygems-update

sudo update\_rubygems

* + - 1. sudo gem install rack passenger

gem install rack -v=1.6.4

gem install passenger -v=5.0.10 --conservative

* + - 1. sudo passanger-install-apache2-module

1. Install and Configure Agent

**Install and Configure Ansible**

Lab 1 – Download and Configure Ansible

1. Install VirtualBox (Oracle)
2. Install Vagrant
3. Install and Configure Project environment
   1. Mkdir demo\_ansible
   2. Cd demo\_ansible
   3. Ls
   4. Vagrant init
      1. Will create a file
   5. Nano vagrantfile and change the contents to

# -\*- mode: ruby -\*-

# vi: set ft=ruby :

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

config.vm.define "acs" do |acs|

acs.vm.box = "ubuntu/trusty64"

acs.vm.hostname = "acs"

acs.vm.network "private\_network", ip: "192.168.33.10"

end

config.vm.define "web" do |web|

web.vm.box="nrel/CentOS-6.5-x86\_64"

web.vm.hostname = "web"

web.vm.network "private\_network", ip: "192.168.33.20"

web.vm.network "forwarded\_port", guest: 80, host: 8085

end

config.vm.define "db" do |db|

db.vm.box = "nrel/CentOS-6.5-x86\_64"

db.vm.hostname = "db"

db.vm.network "private\_network", ip: "192.168.33.30"

end

end

* 1. Vagrant up
  2. Vagrant ssh acs
     1. Sudo apt-get install ansible
     2. Ansible
  3. Vagrant ssh web
     1. sudo yum install epel-release
     2. sudo yum install ansible(optional)
  4. vagrant ssh acs
     1. mkdir ex1
     2. cd ex1
     3. ls
     4. vi inventory
        1. 192.168.33.20
        2. 192.168.33.30
     5. Ls
     6. Cat

ansible all -i inventory -u vagrant -m ping -k

* + 1. If ssh gives error add hosts

Ssh [vagrant@192.168.33.20](mailto:vagrant@192.168.33.20)

cat ~/.ssh/known\_hosts

* + 1. Append -v / -vv/-vvv for debugging details
    2. -m for module -> command or shell etc