Practical Worksheet 7

Version: 1.0 Date: 12/04/2018 Author: David Glance

## Learning Objectives

1. Install and configure Chef
2. Create a Git repository to hold configuration and code files
3. Deploy a server with apache2 configured
4. Install and configure Python Fabric
5. Write a Fabric script to install and configure Chef as above

## Technologies Covered

Ubuntu

AWS

Python

Git

Chef

Fabric

## Background

The aim of this lab is to write a program that will:

[1] Understand how to use the basic commands for Github to create, update and retrieve source files from a repository

[2] Background and basics to Chef

[3] How to automatically deploy a server using Chef scripts

[4] How to use Fabric to execute commands remotely

## EC2 instance

## [Step 1] Create an EC2 instance

[1] Create an EC2 micro instance using Ubuntu 18.04

[2] Install Chef[[1]](#footnote-1)

mkdir chef

cd chef

curl -L https://www.opscode.com/chef/install.sh | bash

wget http://github.com/opscode/chef-repo/tarball/master

tar -zxf master

mv chef-chef-repo\* chef-repo

rm master

cd chef-repo/

chef-repo# ls

LICENSE README.md chefignore cookbooks data\_bags environments roles

mkdir .chef

echo "cookbook\_path [ '/home/ubuntu/chef/chef-repo/cookbooks' ]" > .chef/knife.rb

**Install ChefDK**

curl https://omnitruck.chef.io/install.sh | sudo bash -s -- -c current -P chefdk

**Generate app and apache2 cookbooks etc**

cd cookbooks

chef generate cookbook webapp

**The following downloads predefined cookbooks from http://community.opscode.com/cookbooks**

knife cookbook site download apache2

knife cookbook site download apt

<edit> webapp/metadata.rb

append:

**depends "apache2"**

<edit> webapp/recipes/default.rb

append:

**include\_recipe "apache2"**

**apache\_site "default" do**

**enable true**

**end**

cd ..

<edit> solo.rb

add:

**file\_cache\_path "/home/ubuntu/chef/chef-solo"**

**cookbook\_path "/home/ubuntu/chef/chef-repo/cookbooks"**

<edit> web.json

add:

**{**

**"run\_list": [ "recipe[apt]", "recipe[webapp]" ]**

**}**

Run:

chef-solo -c solo.rb -j web.json

This should install and configure apache2

Test that it has worked by navigating to <public dns name> in your browser

**[2] Create a repository on Github and import the files you have used in chef-repo**

**[3] Install and run Fabric**

**NOTE** do this on your VirtualBox VM

The easiest way to install fabric is to:

pip install fabric

You will need to create a config file in ~/.ssh with the contents:

Home chefserver

Hostname <EC2 instance public DNS>

User ubuntu

UserKnownHostsFile /dev/null

StrictHostKeyChecking no

PasswordAuthentication no

IdentityFile <path to your private key>

You can test fabric from the command line:

python

>>> from fabric import Connection

>>> c = Connection(‘chefserver’)

>>> result = c.run(‘uname -s’)

Linux

>>>

**TODO**

Write a python script using fabric to set up and configure the environment you defined above. Note – you should be able to do a ‘git clone’ to pull all of the pre-configured files from Github so this makes setting up the environment very much easier.

The documentation for Fabric is here: <http://docs.fabfile.org/en/2.0/>

## Submission and Quiz

Submit the python file you wrote in [3]– respond to the quiz

## Respond to the Quiz

[1] what are the default ports that ssh, http and https use?

[A] 80, 443, 22

[B] 80, 444, 32

[C] 8000, 443, 32

[D] 8000,44, 11

[2] Chef is a product that is built on top of Fabric which utilises SSH:

[A] True

[B] False

1. (Adapted from http://gettingstartedwithchef.com/first-steps-with-chef.html) [↑](#footnote-ref-1)