﻿Action plan:

1. Installing the ruby on rails, chef client and chef workstation.

2. we can write a cookbook containing all the recipes to install the webserver and replacing the httpd conf file.

3. create a certificate using openssl

4. running the test scripts to validate the server configuration.

Steps:

Step1:

First step I did was to set up an amazon cloud instance. Setting up an amazon instance can be

done with the help of console. I have chosen Ubuntu instance to do the job. When creating the instance I configured the instance so that only one port is open for public and certain ip addresses can only reach it.

Step2:

Updating the installed packages by using the following command.

sudo apt-get update

step3:

Making a directory ruby in the user home directory.

mkdir ~/ruby

navigating into the directory.

cd ~/ruby

step4:

Getting the ruby verison using wget.

wget http://cache.ruby-lang.org/pub/ruby/2.1/ruby-2.1.4.tar.gz

inflating the tar.

tar -xzf ruby-2.1.4.tar.gz

step5:

Navigating into the ruby directory.

cd ruby-2.1.4

running the configure file will check for the dependencies and it will create a make file, which

contains all that need to compile a code.

./configure

step6:

Running the make utility makes the makefile executable.

make

Running the make utility with the install parameter copies the binaries into the /usr/local/bin

directory.

sudo make install

Now the ruby is installed.

Step7:

Installing chef on the workstation.

We can download the chef package from the chef website and transfer the zip file onto the

workstation.

Unzip the file and we will get an chef development kit and the chef starter kit file.

Install the .deb file by using the command

Sudo dpkp -I chef dk\_0. 9. 0-1 amd64.deb

Then we have to create an chef repo in the root directory.

Mkdir chef-repo

In the chef-repo we can unzip the starter kit.

Unzip chef-starter-kit.zip

This will enable the connection between workstation and the hosted chef server.

Now in the chef repo we can generate cookbooks with the recipes in it.

I have used two recipes in the cookbook which are webserver.rb and httpd.rb. we can upload them into the hosted chef server.

We can do it by using command,

Knife cookbook upload webserver\_install

Now, we have all the cookbook in a runlist. When ever we need to spin an another server using

these packages, we can just bootstrap the node by using the IP address.

Knife bootstrap 10.0.2.15 --ssh-user sai --ssh-password --sudo --use-sudo-password --node -

name node1 --runlist

step 8:

creating the ssl certificate. To do that we need to execute the following commands,

openssl req -new -newkey rsa:2048 -nodes -keyout localhost.key -out localhost.csr

openssl x509 -req -day 365 -in localhost.csr -signkey localhost.key -out localhost.crt

step 9:

for redirecting the http requests to https, we have to do the changes in httpd.conf file. In order to automate them, I have written the httpd.conf file and a reciepe to insert the conf file into the location of /etc/http/conf

Step 10:

To test and validate the server configuration, I have writing chef spec test scripts. Those scripts are inserted into the cookbook itself. The test scripts are

Webserver\_spec.rb

Httpd\_spec.rb.

Hence, we have a webserver which has a ssl certificate and it redirects the http requests to https requests.