**BASIC COMMANDS**

which chef

pwd - print working directory

ls - directory files

ls / -root directory

touch – to choose file (touch file.txt)

ls -l - to check privileges

ls –lrt

ls –lrt / - working directory

chmod 777 filename

chmod 744 filename

rm filename

rmdir nodes

rm nodes/\*

rmdir nodes

ctrl c – to cancel

which package (tree) – to check installation

tree- to display the directory as tree structure

**TO CREATE RECIPIE :**

Use vi editor (vi filename.rb) (Eg:hello.rb)

Insert mode -- press i

Delete -- (dd) or delete key

Undo -- esc:q!

Save – esc:wq

------------------------------------------------------------

**hello.rb file**

file '/home.txt' do

content "Hello,World!"

end

-----------------------------------------------------------

To print -- cat filename.rb

**To RUN RECIPIE :**

chef-client --local-mode filename.rb curl localhost

sudo chef-client --local-mode filename.rb

cat /home.txt

-------------------------------------------------

file '/home.txt' do

content "Hello,World!"

end

**TO INSTALL PACKAGE :**

package 'tree' do

action :install

end

package 'httpd'

------------------------------------------------------------------------------------------------------------------------------------------

EXAMPLE:

**setup.rb**

package 'tree' do

action :install

end

package 'ntp'

file '/etc/motd' do

content 'This server is the property of Aisha'

action :create

owner 'root'

group 'root'

end

**To START THE SERVICE :**

service ‘httpd’ do

action [:enable, :start]

end

------------------------------------------------------------------------------------------------------------------------------------------

ps –ef | grep httpd --- to look the running processes

vi shutdown.rb – to stop the services

cat nodes/filename -- to view the files in node

mkdir cookbooks

cd cookbooks/

chef generate cookbook update\_machine

update\_machine then goto cd update\_machine

execute “update-upgrade” do

command “yum –y update”

action :run

end

mkdir cookbooks – to make directory

chef generate cookbook cookbooks/workstation

goto cookbook --- cd cookbooks

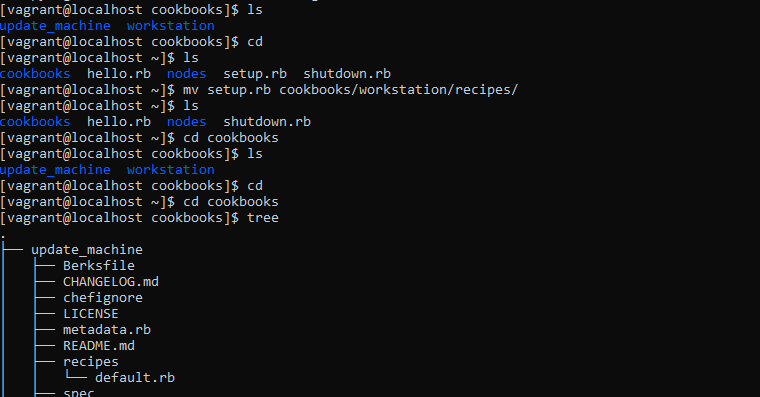
to view ------tree

to view inside ---- cat workstation/metadata.rb

cat workstation/README.md md----- mark down

cat workstation/recipes/default.rb

mv setup.rb cookbooks/workstation/recipes/ --- to move the file to the cookbook

--

**TO INSTALL GIT**

vi recipes/setup.rb

package ‘git’ do

action: install

end

cat cookbook/workstation/recipes/setup.rb

sudo chef-client -z recipes/setup.rb

which git

git init

git status

ls –a -------to see all the files in the workstation

git add .

git commit -m “initial cookbook commit”

git status- -clean

**To modify the file to setup.rb:**

vi recipe/setup.rb

File “hello.txt”

cat recpies/setup.rb

git add recipes/setup.rb (add the modified file to git)

git status (shows file modified)

git commit -m “added file hello.txt to setup.rb” (to commit the changes) (this is a local repository)

**To create a cookbook appache and a recipies server.rb**

* chef generate cookbook cookbooks/appache
* cd cookbooks ------tree
* vi cookbooks/appache/recipes/server.rb

package ‘httpd’ do

action: install

end

file ‘/var/www/html/index.html’ do

content ‘<h1>Hello,world!</h1>’

end

service ‘httpd’ do

action [:enable,:start ]

end

**To run**

* sudo chef-client -z -r "recipe[appache::server]"

**To use default.rb**

* vi cookbooks/appache/recipes/default.rb

include -recipe 'appache::server'

sudo chef-client –zr “recipe[appache]”

------------------------------------------------------------------------------------------------------------------------------------------

**NODES:**

hostname

hostname –I

cat /proc/cpuinfo

cat /proc/meminfo

**Accessing node object:**

Setup.rb(ohai)

package 'git' do

action :install

end

file '/etc/motd' do

content "This server is the property of the chef

HOSTNAME: #{node['hostname']}

IPADDRESS: #{node['ipaddress']}

CPU: #{node['cpu']['0']['mhz']}

MEMORY: #{node['memory']['total']}

"

action :create

end

sudo chef-client -z -r recipes["workstation"::setup]

**appache/server.rb:**

package 'httpd' do

action :install

end

file '/var/www/html/index.html' do

content '<h1>Hello,world!</h1>'

end

service 'httpd' do

action [:enable,:start ]

end

file '/var/www/html/index.html' do

content "<h1>Hello ,World!</h1>

<h2>IPADDRESS: #{node['ipaddress']}</h2>

<h2>HOSTNAME: #{node['hostname']}</h2>

"

end

sudo chef-client -z -r "recipe[appache::server]"

curl localhost(to see the response of the webpage)

------------------------------------------------------------------------------------------------------------------------------------------

**chef repo**

**C:\Devops\chef-repo>**knife client list

kcesoundarya-validator

**C:\Devops\chef-repo>**knife cookbook upload apache

Uploading apache [0.2.1]

Uploaded 1 cookbook.

**C:\Devops\chef-repo>**knife cookbook list

apache 0.2.1

**C:\Devops\chef-repo>**knife cookbook upload workstation

Uploading workstation [0.2.1]

Uploaded 1 cookbook.

**C:\Devops\chef-repo>**knife cookbook upload starter

Uploading starter [1.0.0]

Uploaded 1 cookbook.

**C:\Devops\chef-repo>**knife cookbook list

apache 0.2.1

starter 1.0.0

workstation 0.2.1

------------------------------------------------------------------------------------------------------------------------------------------

**knife**

Workstation and cookbook communicate ----knife

knife cookbook upload cookbook name

knife cookbook list

------------------------------------------------------------------------------------------------------------------------------------------

bootstraping –configuring the node to the required state(chef client)

converging - adding required resources

------------------------------------------------------------------------------------------------------------------------------------------

vagrant status (vw running status)

vagrant ssh-config (port details)

knife bootstrap localhost –ssh-port PORT –sh-user vagrant –sudo –identity-file /PATH/TO/KEY –N web1(provisioning to the node)

knife node list (nodes available)

knife node show web1(particular node details)

knife node run\_list add web1 “recipe[workstation],recipe[apache]”(particular node running recipes)

knife node show web1

vagrant ssh web1(for login)

Run the commands on the web1 vagrant instance(runlist recipes will run )

sudo chef-client

curl localhost

-----------------------------------------------------------------------------------------------------------------------------------------

**DOCKER**

Installing docker using vagrant

Go to cd C:\Devops\Docker

vagrant init

vagrant up

vagrant ssh

sudo yum install docker

sudo docker version

sudo docker info

sudo systemctl start docker

sudo systemctl status docker

sudo systemctl enable docker

sudo docker run hello-world

-----------------------------------------------------------------------------------------------------------------------------------------

**DOCKER**

Easily develop applications and deploy them anywhere

Scalabilitiy--- add more features to the applications whenever want

Light weigth---easily can be shifted (converting package)

container can be used anywhere to run eg:vmvare

containers—can be deployed any where