Chef Configuration

Chef Server

Chef-Client i.e. Nodes

Chef Workstation

* ChefDK
* Starter Kit

knife

1)Configure chef Server

a. Login to Chef Server **=>** https://manage.chef.io

username

password

b. Create Organization

Organization name **=>**

c. Download Starterkit from administrator

d. Install WinScp on local machine

2) configure workstation

a. create an EC2 machine in AWS => WorkstationC

b. Upload the Kit to WorkStation using SCP.

Username : ec2-user

Password : auth with ppk file

Hostname : public IP address of EC2 machine

c. Goto Workstation and Unzip the kit at Home directory

home directory path cd /home/ec2-user/

unzip chef-starter.zip

d. Install ChefDK

wget <https://packages.chef.io/files/stable/chefdk/2.0.28/el/6/chefdk-2.0.28-1.el6.x86_64.rpm>

rpm is basically stands for two things ― a software packaged in this format and a software package

file format

rpm -Uvh chefdk-2.0.28-1.el6.x86\_64.rpm => rpm file installed by using command rpm -Uvh where

-u =>update =>  don’t copy the files from source to destination if destination files are newer

-v =>verbose =>Verbose output

-h => human-readable => display the output numbers in a human-readable format

chefdk-4.2.0-1.el7.x86\_64.rpm(latest version of chef DK)

e. chef versions can be seen by syntax chef –version

**Displayed content**

Chef Development Kit Version: 2.0.28

chef-client version: 13.2.20

delivery version: master (17c1b0fed9be4c70f69091a6d21a4cbf0df60a23)

berks version: 6.2.0

kitchen version: 1.16.0

inspec version: 1.31.1

f. check the knife ssl certification

knife ssl check

content displayed

Connecting to host api.chef.io:443

Successfully verified certificates from `api.chef.io'

g. SSH connection between WS and Node

* Go to root level
* cd /root/.ssh or cd /home/ec2-user/.ssh
* **create ssh-keygen** - authentication key generation, management and conversion. The type of key to be generated is specified with the **-t** option.

ssh-keygen or ssh-keygen -t rsa

Enter file in which to save the key (/root/.ssh/id\_rsa): id\_rsa

id\_rsa.pub key file is generated which is public key. id\_rsa key file is also generated which is private key. private key is used to sign things, and your public key is used to verify your signature

* copy the key from the file i.e. cat id\_rsa.pub or vi id\_rsa.pub

3) configure chef client

a. create an EC2 machine in AWS => NodeC

b. Append copied key to /home/ec2-user/.ssh/authorized\_key i.e. vi authorized\_keys

c. ifconfig to get the inet addr:172.31.47.199

4) Workstation

1. Test it by doing "ssh ec2-user@<Node IP>" i.e. ssh 172.31.47.199

Thus, The authenticity of host '172.31.47.199 i.e. to the client is established.

Are you sure you want to continue connecting (yes/no)? yes

If permission is denied then perform step 5

5) chef client i.e. NodeC

In vi /etc/ssh/sshd\_config

Remove comment for **PermitRootLogin yes**

service sshd restart

BOOTSTRAP

=========

**workstation**

* cd /home/ec2-user
* cd chef-repo
* knife bootstrap <node-ip> --connection-user root --ssh-identity-file /root/.ssh/id\_rsa --node-name WebServer

i.e.

knife bootstrap 172.31.47.199 --connection-user root --ssh-identity-file /root/.ssh/id\_rsa --node-name WebServer

* Verify the Node status in the Chef Server
* Verify the chef-client version in Node i.e. in node ec2 instance

Type chef-client

Cookbook and Recipes

Workstation

1. Create a cookbook under home/ec2-user/chef-repo/cookbooks/
2. chef generate cookbook <cookbook name>

i.e chef generate cookbook myfirstCB

1. myfirstCB cookbook is created with recipe folder and default.rb file
2. Go to cookbook/recipes/default.rb file to write the recipe

i.e. cd myfirstCB/recipes

vi default.rb

file '/tmp/welcome.txt' do

action :create

content "Welcome to chef"

end

1. To execute the recipe of cookbook navigate to chef-repo/cookbook folder

* cd ../../
* cookstyle myfirstCB
* chef-client -z --runlist 'recipe[myfirstCB]'

1. Upload the cookbook to Server using Knife (make sure you are in chef-repo folder)

knife upload <cookbook name>

i.e.

knife upload myfirstCB

Let see few examples

1. create a new directory

directory '/tmp/welcome/' do

action :create

end

file '/tmp/welcome/devops.txt' do

content 'Thank you'

end

1. Install httpd

package 'httpd' do

action :install

end

service 'httpd' do

action :restart

end

1. Set homepage

package 'httpd' do

action :install

end

service 'httpd' do

action :restart

end

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

content "welcome to Devops"

end

1. Html code

package 'httpd'

service 'httpd' do

action :restart

end

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

content '<html>

<body>

<h1>Hello World</h1>

<h1>Welcome to Devops</h1>

</body>

</html>'

end

1. Template creation
   1. chef generate cookbook <cookbook name>

i.e chef generate cookbook apache

* 1. chef generate template index.html - creates templates folder

index.html.erb file created inside template folder

* 1. write the html code in the erb file

i.e vi index.html.erb

<html>

<body>

<h1>Hello Devops world</h1>

<h1>Learning course</h1>

</body>

</html>

* 1. vi apache/recipes/default.rb

package 'httpd'

service 'httpd' do

action :restart

end

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

source 'index.html.erb'

end

1. install tomcat7 and proxypass settings

package 'httpd'

package 'tomcat7' do

action :install

end

package 'tomcat7-webapps' do

action :install

end

file 'etc/httpd/conf.d/proxy.conf' do

content "ProxyPass /examples http://localhost:8080/examples"

content "ProxyPassReverse /examples http://localhost:8080/examples "

end

service 'httpd' do

action :restart

end

service 'tomcat7' do

action :restart

end

using template file

package %w(httpd tomcat7 tomcat7-webapps) do

action :install

end

service 'tomcat7' do

action [:enable, :restart]

end

template 'etc/httpd/conf.d/proxy.conf' do

source 'proxy.conf.erb'

end

service 'httpd' do

action [:enable, :restart]

end

proxy.conf.erb file

ProxyPass /examples http://localhost:8080/examples

ProxyPassReverse /examples http://localhost:8080/examples

**Windows bootstrapping**

* 2 EC2 machine instances to be created

Workstation -Amazon Linux AMI 2018.03.0 (HVM), SSD Volume Type

Windows-Microsoft Windows Server 2012 R2 Base

* Workstation

starter kit

cd /home/ec2-user/

unzip chef-starter.zip

Install ChefDK

wget <https://packages.chef.io/files/stable/chefdk/2.0.28/el/6/chefdk-2.0.28-1.el6.x86_64.rpm>

rpm -Uvh chefdk-2.0.28-1.el6.x86\_64.rpm => rpm file installed by using command rpm -Uvh

* Windows
* Get IP address , username and password by right clicking on Windows EC2 machine instance and selecting connect option

|  |  |
| --- | --- |
| **Public DNS** | ec2-13-127-197-125.ap-south-1.compute.amazonaws.com |
| **User name** | Administrator |
| **Password** | 3m7)\*8eFN9 |

* + Download remote machine and Connect to the remote machine
  + Control Panel => system security =>Windows firewall => Turn off firewall
  + Powershell =>Services.msc

Restart Remote desktop configuration , Remote desktop services

* + Powershell

netsh advfirewall firewall add rule name="WinRM 5985" protocol=TCP dir=in localport=5985 action=allow

* Workstation

cd chef-repo

knife bootstrap windows winrm 13.127.197.125 -x Administrator -P 3m7\)\*8eFN9 -N WebWin2012

IP address 13-127-197-125of windows EC2 instance

Knife ssh node ???

https://computingforgeeks.com/how-to-configure-chef-knife-upload-cookbooks-and-run-a-recipe-on-client-nodes/