First Step

Install Vagrant and virtual box

Create vagrant file

Second step

vagrant init ubuntu/trusty64

Replace with below commands in vagrant file

Vagrant.configure(2) do |config|

config.vm.define "workstation" do |workstation|

workstation.vm.box = "ubuntu/trusty64"

workstation.vm.network "private\_network", ip: "192.168.0.252"

workstation.vm.hostname = "workstation.example.com"

end

config.vm.define "chef" do |chef|

chef.vm.box = "ubuntu/trusty64"

chef.vm.network "private\_network", ip: "192.168.0.253"

chef.vm.hostname = "chef.example.com"

chef.vm.provider "virtualbox" do |v|

v.memory = 4096

v.cpus = 2

end

end

config.vm.define "node" do |node|

node.vm.box = "ubuntu/trusty64"

node.vm.network "private\_network", ip: "192.168.0.3"

node.vm.hostname = "node.example.com"

end

end

vagrant up

Third Step

paste the chef developmentkit and chef server of ubuntu in vagrant folder

Fourth Step //Type below commands

Vegrant ssh workstation

Ls /vagrant/

sudo dpkg -i /vagrant/chefdk\_0.9.0-1\_amd64.deb

mkdir .chef

vim .chef/knife.rb

cookbook\_path ['/home/vagrant/cookbooks'] //In knife.rb

mkdir cookbooks

cd cookbooks

chef generate cookbook my\_cookbook

ls my\_cookbook/

cd my\_cookbook/recipes

vim default.rb

// In default.rb

file 'hello.txt' do

content 'hello world'

end

sudo chef-client -z --runlist 'recipe[my\_cookbook] //to verify my\_cookbook is working or not

exit

Fifth step

// go to root user of workstation

sudo -s

ls /etc

vi hosts

//add these three lines in hosts file

192.168.0.253 chef.example.com chef

192.168.0.252 workstation.example.com workstation

192.168.0.3 node.example.com node

//save it by using escape :wq

exit

Sixth Step

vagrant ssh chef

sudo -s

ls /etc

vi hosts

//add these three lines in hosts file

192.168.0.253 chef.example.com chef

192.168.0.252 workstation.example.com workstation

192.168.0.3 node.example.com node

//save it by using escape :wq

ping node // to verify node is connected to server - stop using ctrl+c

exit

Seventh Step

vagrant ssh node

sudo -s

ls /etc

vi hosts

//add these three lines in hosts file

192.168.0.253 chef.example.com chef

192.168.0.252 workstation.example.com workstation

192.168.0.3 node.example.com node

//save it by using escape :wq

exit

Eighth Step

vagrant ssh chef

sudo -s

ls /vagrant/

dpkg -i /vagrant/chef-server-core\_XXXXXXXX.deb //install chef server software in chef server

chef-server-ctl reconfigure

chef-server-ctl user-create admin admin admin admin@example.com LearnDevops -f admin.pem

chef-server-ctl org-create learndevops "Learn Devops Course" --association\_user admin

cp admin.pem org.pem /vagrant

exit

Ninth Step

vagrant ssh workstation

cp /vagrant/\*.pem .

ls

vim .chef/knife.rb

//Add below commands in knife.rb

current\_dir = File.dirname(\_\_FILE\_\_)

log\_level :info

log\_location STDOUT

node\_name "admin"

client\_key "/home/vagrant/admin.pem"

chef\_server\_url "https://chef.example.com/organizations/learndevops"

cookbook\_path ["/home/vagrant/cookbooks"]

cat .chef/knife.rb // to display all above commands - to verify

knife ssl fetch

knife client list // it will shows learndevops-validator

knife bootstrap node.example.com -N node -X vagrant --sudo

// it will ask vagrant password - enter "vagrant"

cd cookbooks/

ls

knife cookbook upload my\_cookbook

knife node run\_list set node 'recipe[my\_cookbook]' //it will show node run list

ssh node 'sudo chef-client' //give yes and enter vagrant node password in vagrant node - "vagrant"

Tenth Step

Verify Now hello.txt is copied in node