# 2.Chef

1.Downloaded chefdk in Normal ubuntu System & installed

2.Check chef -v version details

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef**$ chef -v

Chef Development Kit Version: 3.2.30

chef-client version: 14.4.56

delivery version: master (6862f27aba89109a9630f0b6c6798efec56b4efe)

berks version: 7.0.6

kitchen version: 1.23.2

inspec version: 2.2.70

### 1.Creating Recipe in Chef

Chef writen in Ruby. Now we are going to create a recipe **“hello.rb”** , here .rb is Ruby extension

Collection of Recipes are called CookBoook

***1.Create chef\_repo folder, go into it***

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$

***2.Create a resource in System using Chef***

create hello.rb file inside **chef\_repo**

file 'motd' do

content 'Hello, World'

end

***3.Apply recipe to Current System.***

The above Code means, create a resorce(fille) with name “motd” with content as “hello world”

By executing **chef-apply hello.rd,** Chef will create a new file in our current infrasrure system

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ chef-apply hello.rb

Recipe: (chef-apply cookbook)::(chef-apply recipe)

\* file[motd] action create

- create new file motd

- update content in file motd from none to 03675a

--- motd 2018-09-26 22:42:58.885793277 +0530

+++ ./.chef-motd20180926-23053-ysyfqd 2018-09-26 22:42:58.885793277 +0530

@@ -1 +1,2 @@

+Hello, World

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ ls

hello.rb motd

4.Here chef is idempotent, that means multiple changes won’t change the result.

It we change the hello.rd file, chef wont consider the changes until we apply changes

***5.Delete file, using action***

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ gedit hello.rb

file 'motd' do

action:delete

end

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ chef-apply hello.rb

Recipe: (chef-apply cookbook)::(chef-apply recipe)

\* file[motd] action delete

- delete file motd

Complete Log

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ gedit hello.rb

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ cat hello.rb

file &apos;motd&apos; do

content &apos;Hello, World&apos;

end

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ ls

hello.rb

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ chef-apply hello.rb

Recipe: (chef-apply cookbook)::(chef-apply recipe)

\* file[motd] action create

- create new file motd

- update content in file motd from none to 03675a

--- motd 2018-09-26 22:42:58.885793277 +0530

+++ ./.chef-motd20180926-23053-ysyfqd 2018-09-26 22:42:58.885793277 +0530

@@ -1 +1,2 @@

+Hello, World

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ ls

hello.rb motd

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ gedit hello.rb

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ chef-apply hello.rb

Recipe: (chef-apply cookbook)::(chef-apply recipe)

\* file[motd] action delete

- delete file motd

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$

## 2. Chef : Recipes

### 1.Create a Resouce : Install a Software in Host Machine

In above ex, we just created a file as a resorce in host mechine. Now go to more advance install Softtware as a pkg in host machine

* Apache2 pkg should install in host mechine
* Apache2 Should enable & Auto Start
* Create index.html, & make it as apache Homepage

**1.edit hello.rd to perform above 3 steps on host system**

package 'apache2'

service 'apache2' do

action[:enable, :start]

end

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

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

end

**2.Do chef-apply**

**Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ sudo chef-apply hello.rb

[sudo] password for satya:

Recipe: (chef-apply cookbook)::(chef-apply recipe)

\* apt\_package[apache2] action install

- install version 2.4.29-1ubuntu4.1 of package apache2

\* service[apache2] action enable (up to date)

\* service[apache2] action start (up to date)

\* file[/var/www/html/index.html] action create

- update content in file /var/www/html/index.html from b66332 to c0086c

--- /var/www/html/index.html 2018-09-26 23:24:20.577635660 +0530

+++ /var/www/html/.chef-index20180926-27935-qxmg32.html 2018-09-26 23:24:28.713710303 +0530

@@ -1,376 +1,2 @@

**Cross Checking**

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ sudo chef-apply hello.rb

Recipe: (chef-apply cookbook)::(chef-apply recipe)

\* apt\_package[apache2] action install (up to date)

\* service[apache2] action enable (up to date)

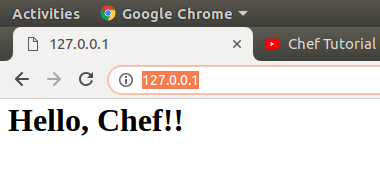
\* service[apache2] action start (up to date)

\* file[/var/www/html/index.html] action create (up to date)

**3.Staring apache Server**

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ service apache2 start**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$

4.Test by going : <http://127.0.0.1/>



Or using Curl

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef\_repo**$ curl localhost

<h1>Hello, Chef!!</h1>

## 3. Chef : Cookbooks

We are creating cookbooks folder, for working

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/cookbooks**$ PS1=’\W>’;

[shorten file path in terminal](https://askubuntu.com/questions/302667/shorten-file-path-in-terminal) PS1='\W> '

To generate CookBook Syntax is

chef generate cookbook <Cookbook\_Name>

**1.Createing Apache Cookbook**

cookbooks> chef generate cookbook chef\_apache2

Here a folder is created inside /cookbooks withthe name chef\_apache2

cookbooks> ls

**chef\_apache2**

cookbooks> cd chef\_apache2/

chef\_apache2> tree

**.**

├── Berksfile

├── CHANGELOG.md

├── chefignore

├── LICENSE

├── metadata.rb

├── README.md

├── **recipes**

│   └── default.rb

├── **spec**

│   ├── spec\_helper.rb

│   └── **unit**

│   └── **recipes**

│   └── default\_spec.rb

└── **test**

└── **integration**

└── **default**

└── default\_test.rb

7 directories, 10 files

**2.Create a Template called index.html in chef\_apache2 cookbook**

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/cookbooks**$

chef generate template chef\_apache2 index.html

Recipe: code\_generator::template

\* directory[./chef\_apache2/templates] action create

- create new directory ./chef\_apache2/templates

\* template[./chef\_apache2/templates/index.html.erb] action create

- create new file ./chef\_apache2/templates/index.html.erb

- update content in file ./chef\_apache2/templates/index.html.erb from none to e3b0c4

here new folder (template) inside **chef\_apache2** and places index.html inside template folder

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/cookbooks/chef\_apache2**$ tree

**.**

├── Berksfile

├── CHANGELOG.md

├── chefignore

├── LICENSE

├── metadata.rb

├── README.md

├── **recipes**

│   └── default.rb

├── **spec**

│   ├── spec\_helper.rb

│   └── **unit**

│   └── **recipes**

│   └── default\_spec.rb

├── **templates**

│   └── index.html.erb

└── **test**

└── **integration**

└── **default**

└── default\_test.rb

**3.Write “hello world” inside index.html**

**chef\_apache2**$ gedit templates/index.html.erb

**4.edit recipe file recipes/default.rb to perform below 3 steps on host system**

* Apache2 pkg should install in host mechine
* Apache2 Should enable & Auto Start
* Create index.html, & make it as apache Homepage

**chef\_apache2**$ gedit recipes/default.rb

package 'apache2'

service 'apache2' do

action [:enable, :start]

end

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

source 'index.html.erb'

end

**5.Applay Cookbook to local System**

[**satya@satya-Aspire-E5-523**](mailto:satya@satya-Aspire-E5-523)**:~/Desktop/DevOps/chef$**

**sudo chef-client --local-mode --runlist ‘recipe[chef\_apache2]’;**

[sudo] password for satya:

[2018-09-27T20:53:43+05:30] WARN: No config file found or specified on command line, using command line options.

Starting Chef Client, version 14.4.56

resolving cookbooks for run list: ["chef\_apache2"]

Synchronizing Cookbooks:

- chef\_apache2 (0.1.0)

Installing Cookbook Gems:

Compiling Cookbooks...

Converging 3 resources

Recipe: chef\_apache2::default

\* apt\_package[apache2] action install (up to date)

\* service[apache2] action enable (up to date)

\* service[apache2] action start (up to date)

\* template[/var/www/html/index.html] action create

- update content in file /var/www/html/index.html from c0086c to 95dfba

--- /var/www/html/index.html 2018-09-26 23:24:28.713710303 +0530

+++ /var/www/html/.chef-index20180927-5228-95sw5q.html 2018-09-27 20:53:46.556287854 +0530

@@ -1,2 +1,2 @@

-<h1>Hello, Chef!!</h1>

+Hello Satya

Running handlers:

Running handlers complete

Chef Client finished, 1/4 resources updated in 03 seconds

**Check Apache Homepage**

**satya@satya-Aspire-E5-523:~/Desktop/DevOps/chef$ curl localhost**

Hello Satya

## 4. CookBooks + Chef Server + Nodes

So far we did changes on local system only.

Now we are going to execute our cookbooks in Hosted Nodes through Chef server

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

| **Chef** | ---------- Node 1

[CookBooks] ----> | **Server** | ---------- Node 2

**(Programmer Mechine) | | --------- Node 3**

**------------------**

### Chef Server : Create Online Chef server

1.Create Chef server by going [Hosted Chef Server - Manage Chef](https://manage.chef.io/signup)

2.Login to Chef Server : <https://manage.chef.io/>

3.Administration > Oraganization >smlcodes > Actions > Starter Kit > Download Starter Kit

4.Extract Downloaded zip (**Desktop/DevOps/chef/chef-starter**)

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef-starter**$ tree

**.**

└── **chef-repo**

├── **cookbooks**

│   ├── chefignore

│   └── **starter**

│   ├── **attributes**

│   │   └── default.rb

│   ├── **files**

│   │   └── **default**

│   │   └── sample.txt

│   ├── metadata.rb

│   ├── **recipes**

│   │   └── default.rb

│   └── **templates**

│   └── **default**

│   └── sample.erb

├── README.md

└── **roles**

└── starter.rb

10 directories, 8 files

5.<https://supermarket.chef.io/> where all the user created cookbooks are stored.

For example Search for learn\_chef\_apache2,

we are going to use above cookbook in this example

**6. Download cookbook using Knife**

i. Go to ***/DevOps/chef/chef-starter/chef-repo*** & run **knife** command to download cookbook.

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef-starter/chef-repo**$

**knife cookbook site download learn\_chef\_apache2**

Downloading learn\_chef\_apache2 from Supermarket at version 0.3.0 to /home/satya/Desktop/DevOps/chef/chef-starter/chef-repo/learn\_chef\_apache2-0.3.0.tar.gz

Cookbook saved: /home/satya/Desktop/DevOps/chef/chef-starter/chef-repo/learn\_chef\_apache2-0.3.0.tar.gz

ii.Exatract tar file to cookbooks folder

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef-starter/chef-repo**$ ls

**cookbooks** **learn\_chef\_apache2-0.3.0.tar.gz** README.md **roles**

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/chef/chef-starter/chef-repo**$

**tar -zxxvf learn\_chef\_apache2-0.3.0.tar.gz -C cookbooks**

learn\_chef\_apache2/

learn\_chef\_apache2/.kitchen.yml

learn\_chef\_apache2/Berksfile

learn\_chef\_apache2/Berksfile.lock

learn\_chef\_apache2/chefignore

learn\_chef\_apache2/metadata.json

learn\_chef\_apache2/metadata.rb

learn\_chef\_apache2/README.md

learn\_chef\_apache2/recipes/

learn\_chef\_apache2/templates/

learn\_chef\_apache2/templates/default/

learn\_chef\_apache2/templates/default/index.html.erb

learn\_chef\_apache2/recipes/default.rb

After Extraction remove .tar file

**rm -rf learn\_chef\_apache2-0.3.0.tar.gz**

Check ***learn\_chef\_apache2/recipes/default.rb*** it contains same code we configue before

apt\_update 'Update the apt cache daily' do

frequency 86\_400

action :periodic

end

package 'apache2'

service 'apache2' do

supports :status => true

action [:enable, :start]

end

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

source 'index.html.erb'

end

**7.Upload CookBook to Chef Server**

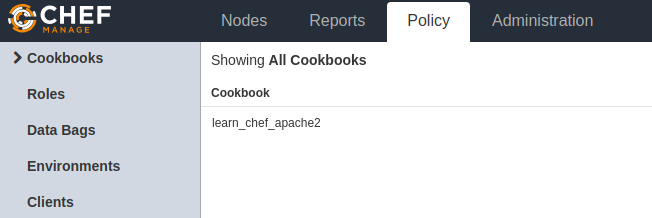
**go to** Desktop/DevOps/chef/chef-starter/chef-repo folder & upload cookbook to chef server

chef-repo> **knife cookbook upload learn\_chef\_apache2**

Uploading learn\_chef\_apache2 [0.3.0]

Uploaded 1 cookbook.

**8.Check Uploaded CookBook in server > policy tab**

****

**Now we are going to manage our nodes throgh Chef Server**

### Nodes : Manage Nodes using Chef server

We have two nodes created using vagrant, up them[**vagrant up, vagrant ssh**]

1. Ubuntu : **vagrant@vagrant-ubuntu-trusty-64:**
2. CentOS : **[vagrant@localhost ~]$**

**IpAddress : check ip’s by ping <ip>**

change the password using sudo passwd ubuntu (by default ubuntu user has sudo-permissions with NOPASSWD set)

Vagrant.configure("2") do |config|

config.vm.box = "ubuntu/trusty64"

config.vm.network "forwarded\_port", guest: 80, host: 5555

config.vm.network "public\_network"

end

1) wlp3s0 --> choose this

**2) enp2s0f1**

**Ubuntu : 5555**

* IpAddrees - **192.168.0.105**
* Username/pwd - ubuntu/ ubuntu

**CentOs:6666**

* IpAddrees - inet **192.168.0.107**
* Username/pwd - root / vagrant

**1.Working with Ubuntu Node (192.168.0.105)**

**1.Go to Clinet workstation coomandline**

/Desktop/DevOps/chef/chef-starter/chef-repo

**2.Boostrap Node1 – Ubuntu & run cookbook on Node1**

chef-repo> knife bootstrap 192.168.0.105 --ssh-user ubuntu --ssh-password ubuntu --sudo --use-sudo-password --node-name cnode1 --run-list &apos;recipe[learn\_chef\_apache2]&apos;

Node cnode1 exists, overwrite it? (Y/N) y

Client cnode1 exists, overwrite it? (Y/N) y

Creating new client for cnode1

Creating new node for cnode1

Connecting to **192.168.0.105**

192.168.0.105 -----> Existing Chef installation detected

192.168.0.105 Starting the first Chef Client run...

192.168.0.105 Starting Chef Client, version 11.8.2

192.168.0.105 resolving cookbooks for run list: ["learn\_chef\_apache2"]

192.168.0.105 Synchronizing Cookbooks:

192.168.0.105 - learn\_chef\_apache2

192.168.0.105 Compiling Cookbooks...

192.168.0.105 Converging 3 resources

192.168.0.105 Recipe: learn\_chef\_apache2::default

192.168.0.105 \* package[apache2] action install

192.168.0.105 - install version 2.4.7-1ubuntu4.20 of package apache2

192.168.0.105

192.168.0.105 \* service[apache2] action enable

192.168.0.105 - enable service service[apache2]

192.168.0.105

192.168.0.105 \* service[apache2] action start (up to date)

192.168.0.105 \* template[/var/www/html/index.html] action create

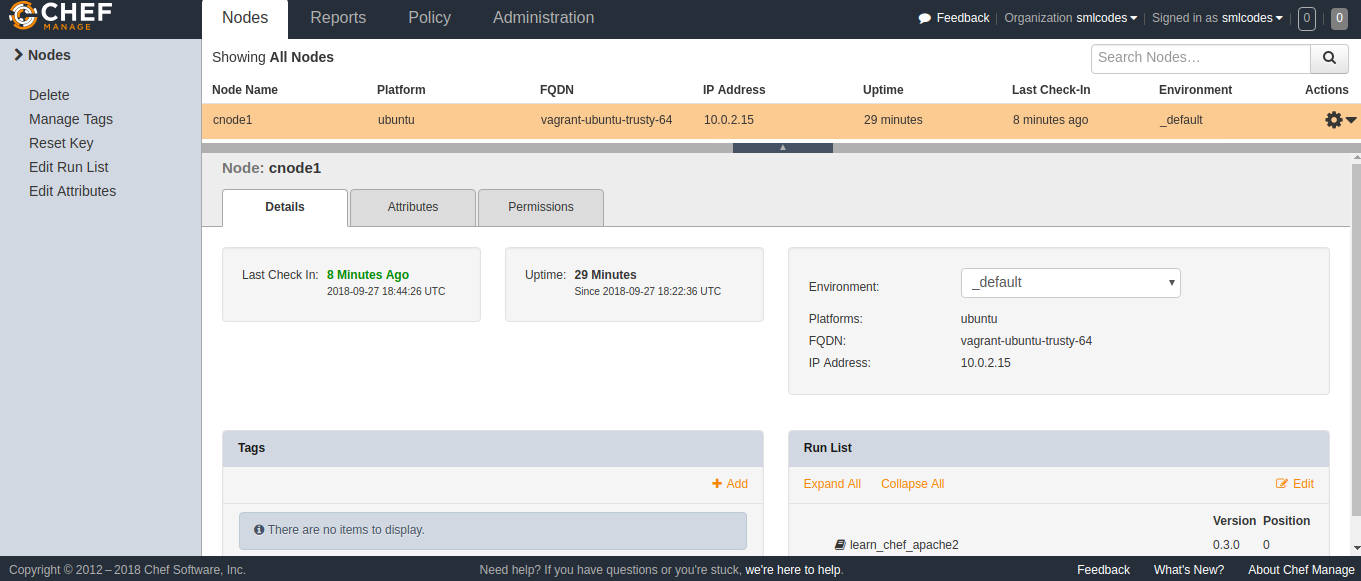
192.168.0.105 - update content in file /var/www/html/index.html from 538f31 to ef4ffd

192.168.0.105 --- /var/www/html/index.html 2018-09-27 18:46:21.787423744 +0000

192.168.0.105 +++ /tmp/chef-rendered-template20180927-2085-27sx61 2018-09-27 18:46:35.617936153 +0000

192.168.0.105 @@ -1,379 +1,6 @@

**3.Now we can check, knife automatically register Ubuntu Node1 , with Chef Server**



**4.Go To Node1-Ubuntu Terminal, Check the Home page**

**vagrant@vagrant-ubuntu-trusty-64:~$ curl localhost**

<html>

<body>

<h1>hello world</h1>

</body>

</html>

**2.Working with Node 2 - CentOS (192.168.0.107)**

**1.Go to Clinet workstation coomandline**

/Desktop/DevOps/chef/chef-starter/chef-repo

**2.Boostrap Node2 – CentOs & run cookbook on Node2**

chef-repo> knife bootstrap 192.168.0.107 --ssh-user vagrant --ssh-password vagrant --sudo --use-sudo-password --node-name cnode2 --run-list &apos;recipe[learn\_chef\_apache2]&apos;

`rescue in new\_session&apos;: **Authentication failed for user vagrant@192.168.0.107@192.168.0.107 (Net::SSH::AuthenticationFailed)**

**ERROR:** Net::SSH::AuthenticationFailed: Authentication failed for user vagrant@192.168.0.107@192.168.0.107

Fixed it!

So when you are using hosted chef you need to pass in a private key with the bootstrap and have the public key in your autherized\_keys file....

1. install the ChefSDK
2. SCP your starter kit from hosted Chef
3. extract the starter kit to ~/chef-repo
4. generate a new keypair: ssh-keygen
5. add the public key to your autherized\_keys file: $ cat id\_rsa.pub >> authorized\_keys
6. run the knife bootstrap with the following:

sudo knife bootstrap {{server-ip}} --ssh-user {{your-server-user}} -i ~/.ssh/id\_rsa --sudo --node-name web1

That should work!

I would also suggest that the user you pass as the --ssh-user has passwordless sudo access.

**3.Now we can check, knife automatically register CentOS Node2 , with Chef Server**

**4.Go To Node2-CentOS Terminal, Check the Home page**

## More on Chef

To get no.of Nodes in Chef server

**chef-repo> knife node list**

cnode1

Get more Node Info

**chef-repo> knife node show cnode1**

**Node Name:** **cnode1**

Environment: \_default

FQDN: vagrant-ubuntu-trusty-64

IP: 10.0.2.15

Run List: recipe[learn\_chef\_apache2]

Roles:

Recipes: learn\_chef\_apache2

Platform: ubuntu 14.04

Tags:

# 3. Puppet

Using vagrant [vagrant@vagrant-ubuntu-trusty-64](mailto:vagrant@vagrant-ubuntu-trusty-64): box

* uname/pwd = vagrant / vagrant
* ipadrress = 192.168.0.105

By default puppet is installed in ubuntu, remove it , will start from scrach

root@vagrant-ubuntu-trusty-64:/home/vagrant/puppet# **puppet --version**

3.4.3

root@vagrant-ubuntu-trusty-64:/home/vagrant/puppet#

**apt-get purge puppet puppet-common -y**

Reading package lists... Done

Building dependency tree

Reading state information... Done

The following packages were automatically installed and are no longer required:

root@vagrant-ubuntu-trusty-64:/home/vagrant/puppet# **apt-get autoremove -y**

Reading package lists... Done

Building dependency tree

Reading state information... Done

The following packages will be REMOVED:

Cross check

vagrant@vagrant-ubuntu-trusty-64:~/puppet$ **ls -la /etc/puppet**

ls: cannot access /etc/puppet: No such file or directory

vagrant@vagrant-ubuntu-trusty-64:~/puppet**$ ls -al /var/lib/puppet**

ls: cannot access /var/lib/puppet: No such file or directory

**1: Enable the Puppet Package Repository**

The newest versions of Puppet can be installed from the [apt.puppetlabs.com](https://apt.puppetlabs.com/) package repository.

To enable the repository:

1. Download the “puppetlabs-release” package for your OS version.
   * You can see a full list of these packages on the front page of <https://apt.puppetlabs.com/>. They are all named puppetlabs-release-<CODE NAME>.deb. (For Ubuntu releases, the code name is the adjective, not the animal.)
   * Architecture is handled automatically; there is only one package per OS version.
2. Install the package by running dpkg -i <PACKAGE NAME>.
3. Run apt-get update to get the new list of available packages.

To enable the repository for Ubuntu 14.04 Trusty Tahr:

wget https://apt.puppetlabs.com/puppetlabs-release-trusty.deb

sudo dpkg -i puppetlabs-release-trusty.deb

sudo apt-get update

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/puppet**$

**wget https://apt.puppetlabs.com/puppetlabs-release-trusty.deb**

--2018-09-28 20:25:12-- https://apt.puppetlabs.com/puppetlabs-release-trusty.deb

puppetlabs-release-trusty.deb 100%[============================================================>] 16.55K --.-KB/s in 0.004s

2018-09-28 20:25:13 (3.70 MB/s) - ‘puppetlabs-release-trusty.deb’ saved [16944/16944]

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/puppet**$

**sudo dpkg -i puppetlabs-release-trusty.deb**

[sudo] password for satya:

Setting up puppetlabs-release (1.1-1) ...

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/puppet**$ **sudo apt-get update**

Ign:1 http://apt.puppetlabs.com trusty InRelease

Get:2 http://apt.puppetlabs.com trusty Release [101 kB]

Get:3 http://apt.puppetlabs.com trusty Release.gpg [819 B]

Here we Just added the puppetlab packages to our system only. We didn’t installed any thing here.

**2.Install PuppetMaster in Ubuntu Master System**

**satya@~/Desktop/DevOps/puppet**$ sudo apt-get install puppetmaster

**3.Check all Pupper folders are created or not in /var/lib/puppet & /*etc/*puppet folder**

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/puppet**$ ls -la /etc/puppet/

total 36

drwxr-xr-x 3 root root 4096 Sep 28 20:35 **.**

drwxr-xr-x 136 root root 12288 Sep 28 20:35 **..**

-rw-r--r-- 1 root root 5401 Feb 23 2018 auth.conf

drwxr-xr-x 2 root root 4096 Apr 20 02:36 **code**

-rw-r--r-- 1 root root 120 Feb 23 2018 hiera.yaml

-rw-r--r-- 1 root root 126 Feb 23 2018 puppet.conf

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/puppet**$ sudo ls -la /var/lib/puppet/

total 48

drwxr-x--- 12 puppet puppet 4096 Sep 28 20:35 .

drwxr-xr-x 68 root root 4096 Sep 28 20:35 ..

drwxr-x--- 2 puppet puppet 4096 Sep 28 20:35 bucket

drwxr-xr-x 2 root root 4096 Sep 28 20:35 facts.d

drwxr-xr-x 2 root root 4096 Sep 28 20:35 lib

drwxr-xr-x 2 root root 4096 Sep 28 20:35 locales

drwxr-x--- 2 puppet puppet 4096 Sep 28 20:35 preview

drwxr-x--- 2 puppet puppet 4096 Sep 28 20:35 reports

drwxr-x--- 2 puppet puppet 4096 Sep 28 20:35 server\_data

drwxrwx--x 8 puppet puppet 4096 Sep 28 20:35 ssl

drwxr-xr-t 2 root root 4096 Sep 28 20:35 state

drwxr-x--- 2 puppet puppet 4096 Sep 28 20:35 yaml

**4.Configure SSL in *var*/lib/puppet/ssl**

By defalut SSL is configured, remove SSL folder. We will configure manually

For doing that first stop puppetmaster service

**/Desktop/DevOps/puppet**$ service puppetmaster stop

Backup SSL folder & remove it (Changed name to ssl\_bkp)

**/Desktop/DevOps/puppet**$ sudo mv /var/lib/puppet/ssl /var/lib/puppet/ssl\_bkp

Change **/Desktop/DevOps/puppet**$ sudo gedit /etc/puppet/puppet.conf

[main]

ssldir = /var/lib/puppet/ssl

[master]

vardir = /var/lib/puppet

cadir = /var/lib/puppet/ssl/ca

cert = puppet

dns\_alt\_names = puppet,puppetmaster,puppetmaster.smlcodes.com

By Starting PuppetMaster SSL folder will create automatically

**/Desktop/DevOps/puppet**$ service puppetmaster start

Check that SSL folder will caretes

**satya@satya-Aspire-E5-523**:**~/Desktop/DevOps/puppet**$

**sudo ls -la /var/lib/puppet/ssl**

total 36

drwxrwx--x 8 puppet puppet 4096 Sep 28 20:55 .

drwxr-x--- 13 puppet puppet 4096 Sep 28 20:55 ..

drwxr-xr-x 5 puppet puppet 4096 Sep 28 20:55 ca

drwxr-xr-x 2 puppet puppet 4096 Sep 28 20:55 certificate\_requests

drwxr-xr-x 2 puppet puppet 4096 Sep 28 20:55 certs

-rw-r--r-- 1 puppet puppet 971 Sep 28 20:55 crl.pem

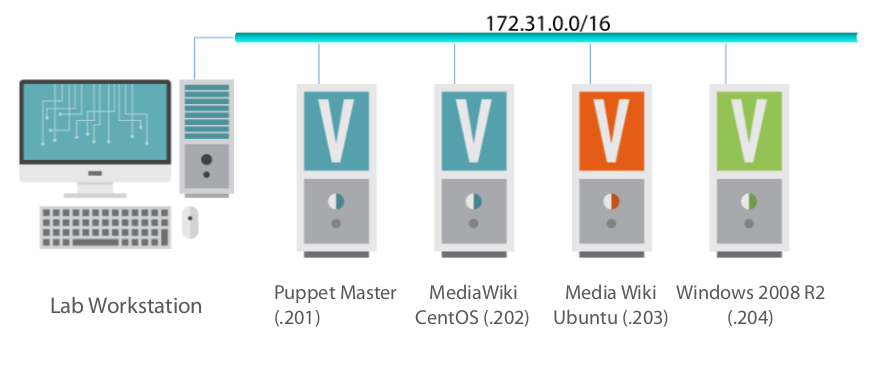
drwxr-x--- 2 puppet puppet 4096 Sep 28 20:55 private

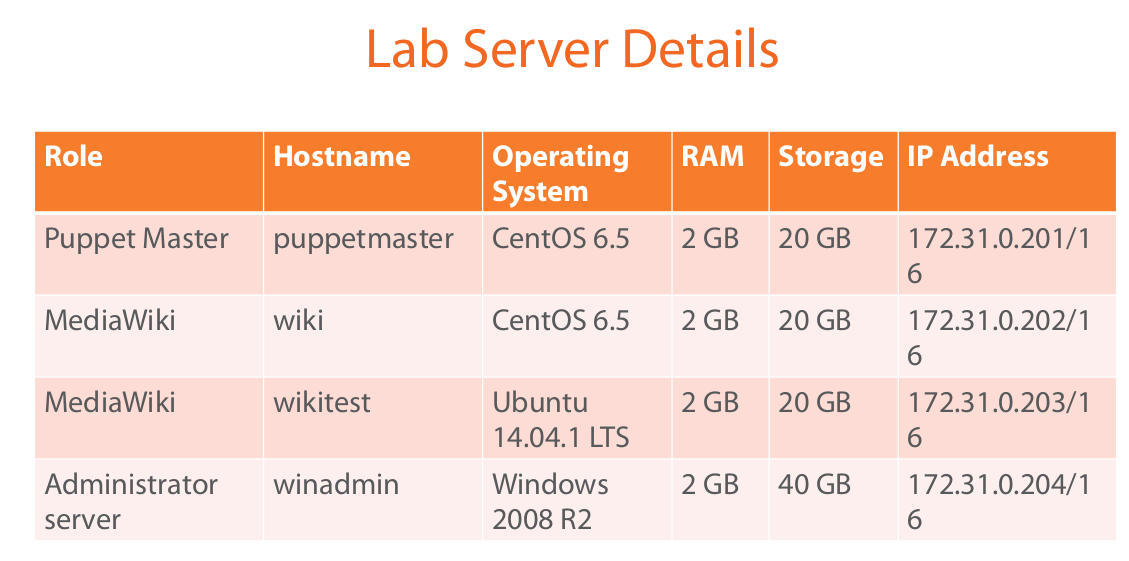
drwxr-x--- 2 puppet puppet 4096 Sep 28 20:55 private\_keys

drwxr-xr-x 2 puppet puppet 4096 Sep 28 20:55 public\_keys

5.Check , what are the all SSL cerfificates available to PuppetMaster by doing,

### PluralSight Puppet handson



<https://app.pluralsight.com/player?course=puppet-system-administrators-fundamentals&author=ben-piper&name=puppet-system-administrators-fundamentals-m2&clip=4&mode=live>

### Setting up PuppetMaster

**satya@satya**:**~/.../puppetlab**$ vagrant box add centos65-base centos65.box

**==> box: Box file was not detected as metadata. Adding it directly...**

**==> box: Adding box &apos;centos65-base&apos; (v0) for provider:**

box: Unpacking necessary files from: file:///home/satya/Desktop/DevOps/puppetlab/centos65.box

**==> box: Successfully added box &apos;centos65-base&apos; (v0) for &apos;virtualbox&apos;!**

**satya@satya**:**~/.../puppetmaster**$ vagrant up

Bringing machine &apos;puppetmaster&apos; up with &apos;virtualbox&apos; provider...

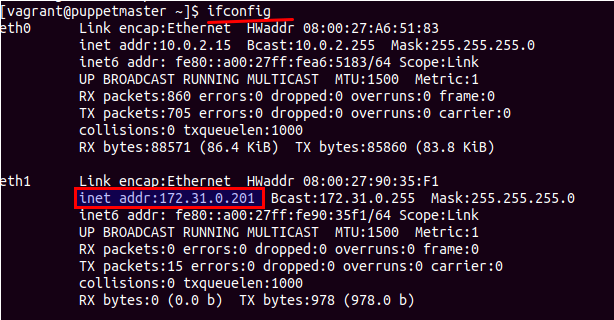
**==> puppetmaster: Importing base box &apos;centos65-base&apos;...**

**==> puppetmaster: Matching MAC address for NAT networking...**

**satya@satya**:**~/.../puppetmaster**$ vagrant ssh

Last login: Sun Dec 14 13:06:43 2014 from localhost

[vagrant@puppetmaster ~]$



**2.Install following pkgs: nano -text editor, git,ntp & Start ntp server**

The Network Time Protocol (**NTP**) is a protocol used to synchronize computer system clock automatically over a networks. ... The most common method to sync system time over a network in **Linux**desktops or servers is by executing the ntpdate command

[vagrant@puppetmaster ~]$ sudo yum -y install nano git ntp

[vagrant@puppetmaster ~]$ sudo service ntpd start

Starting ntpd: [ OK ]

[vagrant@puppetmaster ~]$

[vagrant@puppetmaster ~]$ sudo chkconfig ntp

[vagrant@puppetmaster ~]$ chkconfig | grep ntpd

ntpd 0:off 1:off 2:off 3:off 4:off 5:off 6:off

ntpdate 0:off 1:off 2:off 3:off 4:off 5:off 6:off

[vagrant@puppetmaster ~]$

**3. Installing Puppet Master**

Two things we need to do here

* Add Puppet Repositoty

[vagrant@puppetmaster ~]$

sudo yum -y install http://yum.puppetlabs.com/puppetlabs-release-el-6.noarch.rpm

* Install Pupper Server (PuppetMaster)

[vagrant@puppetmaster ~]$ sudo yum -y install puppet-server

On Succussfull install i will gives version number on check

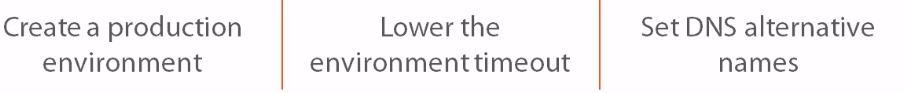
[vagrant@puppetmaster ~]$ puppet master --version

3.8.7

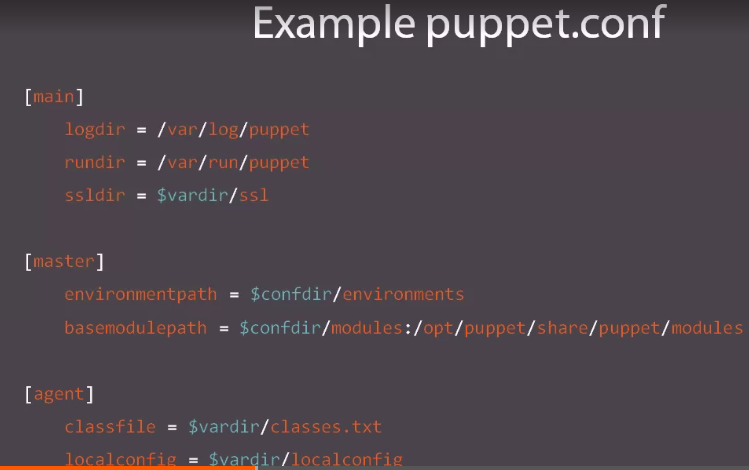
**4.Setting up Directory Enviroments**

* Puppet programs are stored in files called **manifests**. These manifests contain **node** **definitions** and **resource** **decorations**, node which tell Puppet exactly what configurations need to be applied to which nodes.
* By default Puppet Master search for manifest file deatils at /**etc/puppet/manifests/site.pp.(PP- Puppet Programming)**

Tasks



* Each directory environment is controlled by an environment configuration file. Directory environments are stored in subdirectories located under the **/etc/puppet/environments** directory.
  + - * + Producton : **etc/puppet/environments/production/ environment.conf**
        + **Test : etc/puppet/environments/test/ environment.conf**
* The Puppet configuration file is located at /**etc/puppet/puppet.conf.** Generally, there will be only one Puppet configuration file per Puppet Master or node. The file contains three section, **Master, Agent, and Main.**
  + - The Master section controls the settings specific to the Puppet Master.
    - The Agent contains settings specific to the Puppet Agent
    - Main section contains settings which are used if the more specific Master and Agent sections don't set a value for a given configuration setting.



***Creating our Production environmet manifet file***

[vagrant@puppetmaster ~]$ sudo mkdir -p /etc/puppet/environments/production/{modules,manifests}

[vagrant@puppetmaster ~]$ cd /etc/puppet/environments/production

[vagrant@puppetmaster production]$ ls

manifests modules

[vagrant@puppetmaster production]$ sudo nano environment.conf

modulepath = /etc/puppet/environments/production/modules

environment\_timeout = 5s

2.Tell to Puppet.conf about our environment

[vagrant@puppetmaster production]$ cd /etc/puppet/

[vagrant@puppetmaster puppet]$ sudo nano puppet.conf

[main]

# The Puppet log directory.

# The default value is &apos;$vardir/log&apos;.

logdir = /var/log/puppet

# Where Puppet PID files are kept.

# The default value is &apos;$vardir/run&apos;.

rundir = /var/run/puppet

# Where SSL certificates are kept.

# The default value is &apos;$confdir/ssl&apos;.

ssldir = $vardir/ssl

dns\_alt\_names = puppet,puppetmaster,puppetmaster.benpiper.com

[master]

environmentpath = $confdir/environments

basemodulepath = $confdir/modules:/opt/puppet/share/modules

[agent]

# The file in which puppetd stores a list of the classes

# associated with the retrieved configuratiion. Can be loaded in

# the separate ``puppet`` executable using the ``--loadclasses``

# option.

# The default value is &apos;$confdir/classes.txt&apos;.

classfile = $vardir/classes.txt

# Where puppetd caches the local configuration. An

# extension indicating the cache format is added automatically.

# The default value is &apos;$confdir/localconfig&apos;.

localconfig = $vardir/localconfig

[vagrant@puppetmaster puppet]$ [vagrant@puppetmaster puppet]$ sudo nano puppet.conf

-bash: [vagrant@puppetmaster: command not found

[vagrant@puppetmaster puppet]$

**5.PuppetMaster Security : Setting up SSL Certificates**

***i.Set SELinux***

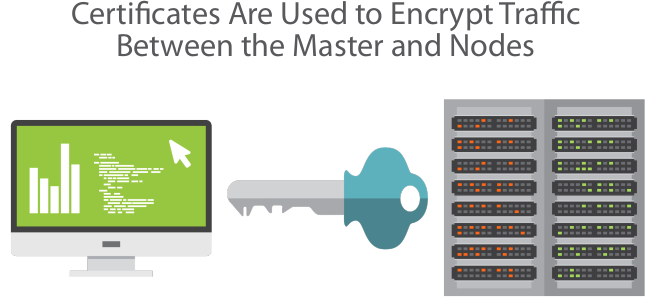
**[vagrant@puppetmaster puppet]$ sudo setenforce permissive**

[vagrant@puppetmaster puppet]$ sudo sed -i &apos;s\=enforcing\=permissive\g&apos; /etc/sysconfig/selinux

[vagrant@puppetmaster puppet]$ sudo getenforce

Permissive

***ii.Generating SSL certificates***

****

[vagrant@puppetmaster puppet]$ **sudo puppet master --verbose --no-daemonize**

Notice: /File[/etc/puppet/environments/production]/seluser: seluser changed &apos;unconfined\_u&apos; to &apos;system\_u&apos;

Info: Creating a new SSL key for ca

Info: Creating a new SSL certificate request for ca

Info: Certificate Request fingerprint (SHA256): C7:66:3C:22:A5:C2:33:10:9C:A3:9E:99:5C:87:4F:8A:09:5C:03:D7:CA:83:CB:0F:17:18:E4:93:7E:84:B2:03

Notice: Signed certificate request for ca

Info: Creating a new certificate revocation list

Info: Creating a new SSL key for puppetmaster

Info: csr\_attributes file loading from /etc/puppet/csr\_attributes.yaml

Info: Creating a new SSL certificate request for puppetmaster

Info: Certificate Request fingerprint (SHA256): 83:D5:C6:1E:6B:5B:14:BE:3E:90:DF:76:02:72:8D:05:B1:3D:B1:82:C5:D8:56:9B:31:38:69:37:36:89:16:5E

Notice: puppetmaster has a waiting certificate request

Notice: Signed certificate request for puppetmaster

Notice: Removing file Puppet::SSL::CertificateRequest puppetmaster at &apos;/var/lib/puppet/ssl/ca/requests/puppetmaster.pem&apos;

Notice: Removing file Puppet::SSL::CertificateRequest puppetmaster at &apos;/var/lib/puppet/ssl/certificate\_requests/puppetmaster.pem&apos;

Notice: Starting Puppet master version 3.8.7

^CNotice: Caught INT; exiting

Check SSL Certificate by going

[vagrant@puppetmaster puppet]$ **sudo ls -ls /var/lib/puppet/ssl**

total 28

4 drwxr-xr-x. 5 puppet puppet 4096 Sep 28 15:31 ca

4 drwxr-xr-x. 2 puppet puppet 4096 Sep 28 15:31 certificate\_requests

4 drwxr-xr-x. 2 puppet puppet 4096 Sep 28 15:31 certs

4 -rw-r--r--. 1 puppet puppet 958 Sep 28 15:31 crl.pem

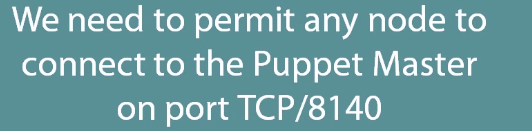
4 drwxr-x---. 2 puppet puppet 4096 Sep 28 15:31 private

4 drwxr-x---. 2 puppet puppet 4096 Sep 28 15:31 private\_keys

4 drwxr-xr-x. 2 puppet puppet 4096 Sep 28 15:31 public\_keys

[vagrant@puppetmaster puppet]$

**6.Configuring IPTable**



[vagrant@puppetmaster puppet]$ sudo nano /etc/sysconfig/iptables

add --> -A INPUT -m state --state NEW -m tcp -p tcp --dport 8140 -j ACCEPT

[vagrant@puppetmaster puppet]$ sudo service iptables restart

iptables: Setting chains to policy ACCEPT: filter [ OK ]

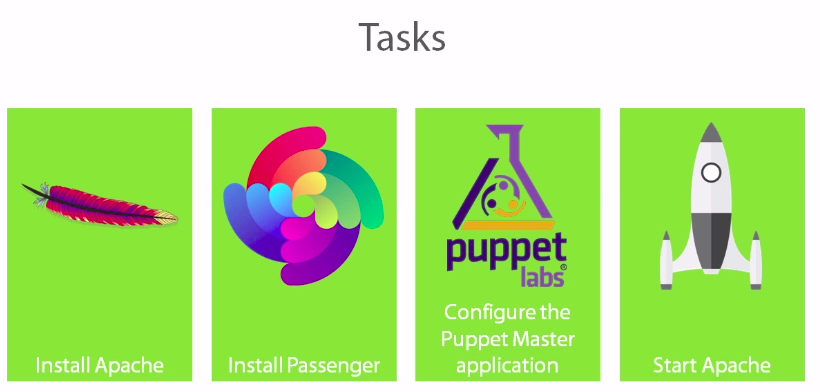
iptables: Flushing firewall rules: [ OK ]

iptables: Unloading modules: [ OK ]

iptables: Applying firewall rules: [ OK ]

[vagrant@puppetmaster puppet]$

**7.Install Apache**

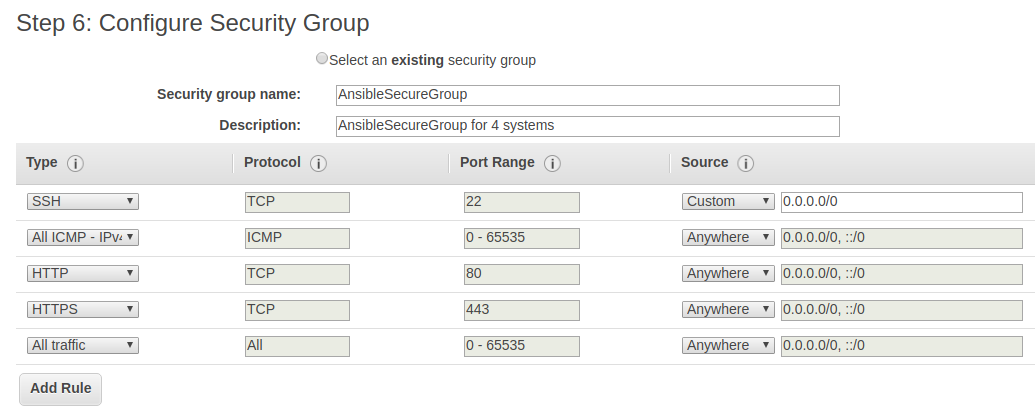


[vagrant@puppetmaster puppet]$ **sudo yum -y install httpd httpd-devel mod\_ssl ruby-devel rubygems gc-c++ libcurl-devel openssl-devel**

gem – is a pakage manager for Ruby

vagrant@puppetmaster puppet]$ sudo gem install rack passenger

# 5. Ansible Aws



# 10. Docker

