PUPPET

Document History

#	Description	Owner/Date	Reviewer/Date	Approver/Date
1	Initial, Section - 1	Ajay/24-June-16		
2				

Index

1. Introduction

Puppet is an open source Configuration management tool, and puppet includes it's own declarative language to describe system configuration. Puppet is produced by Puppet Labs in 2005. It is written in Ruby
This document explains various use-case performed by the DevOps Enabler Team using puppet

2. Scope

The Scope of this document is to contribute to the knowledge base and share the best practices within the DevOps Enabler & Co that are followed while executing various tasks.

3.Environment

- Virtual machines - Centos 6.7
- Puppet - 3.8

4. Use Case 1: Puppet Master Installation

4.1 Overview:

In this use-case we install PUPPET Master (3.8.1) in Centos 6.7

4.2 Assumptions:

- **System must be with Name resolution:** Every node must have a unique hostname. **Forward and reverse DNS** must both be configured correctly.
- **Timestamp is working** : The time must be set accurately on the Puppet master server that will be acting as the certificate authority. You should probably use NTP.
- Puppet master tar file is already downloaded in server node
- Necessary admin permissions are provided to install puppet.

4.3 Scenario :

Installing puppet master in puppet server by using tar file

4.4 Steps :

1. Login to the target system to install PUPPET master.
2. You must login with admin permissions (root or privileged user).
3. Go to the location where the puppet master file(.tar file) is located.
4. Extract the tar file by using following command

   Command : $ tar -xvzf puppet-enterprise-3.8.1-el-6-x86_64.tar.tar
5. After extracting **puppet-enterprise-3.8.1-el-6-x86_64** directory will be appear, and enter into that directory

   Command : cd puppet-enterprise-2015.3.3-el-6-x86_64/
6. In that directory we want to execute **puppet-enterprise-installer** file, use following command

   Command : ./puppet-enterprise-installer
7. Then it will ask permission to install puppet packages, press “y”

8. It will take time to install packages, and after some time it will shows a message, i.e as shown below

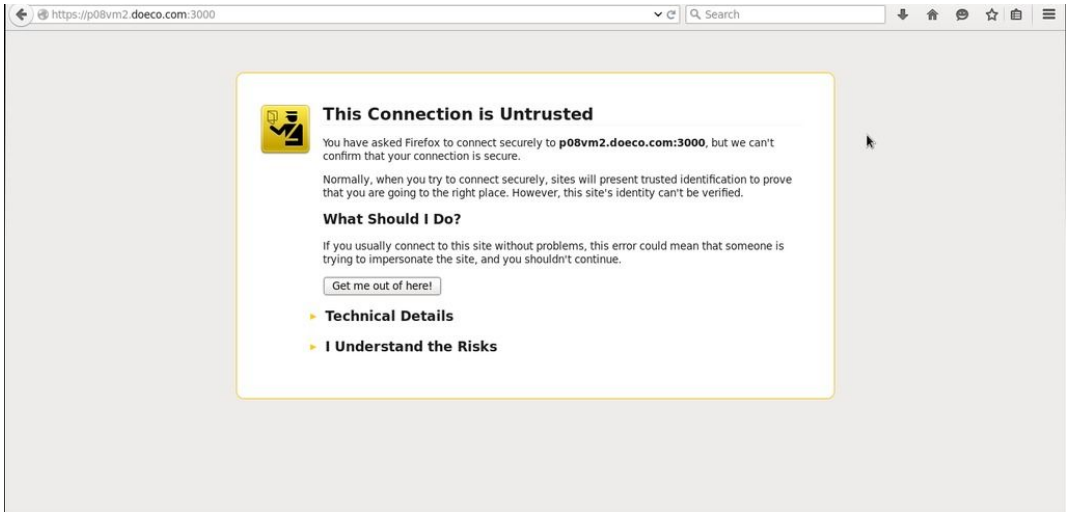
Please go to https://p08vm2.doeco.com:3000 in your browser to continue installation. Be sure to use https:// and that port 3000 is reachable through the firewall

9. When you saw the message, go to the url which shown in the above message

   Note : firewall should be inactive. To stop firewall use below command

```
service iptables stop
```

10. When you go through the URL browser throws an security suggestion as follows

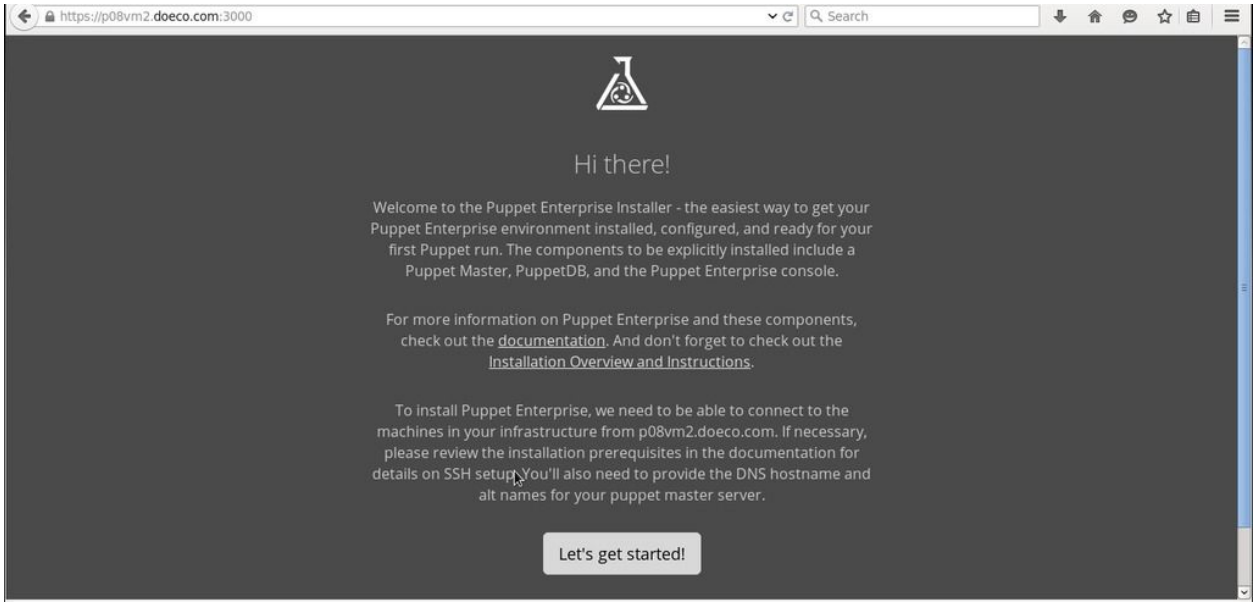


Then click on the **I Understand the Risks** option then it will shows an option **Add Exception** click on that option

Then a new window will open for confirmation

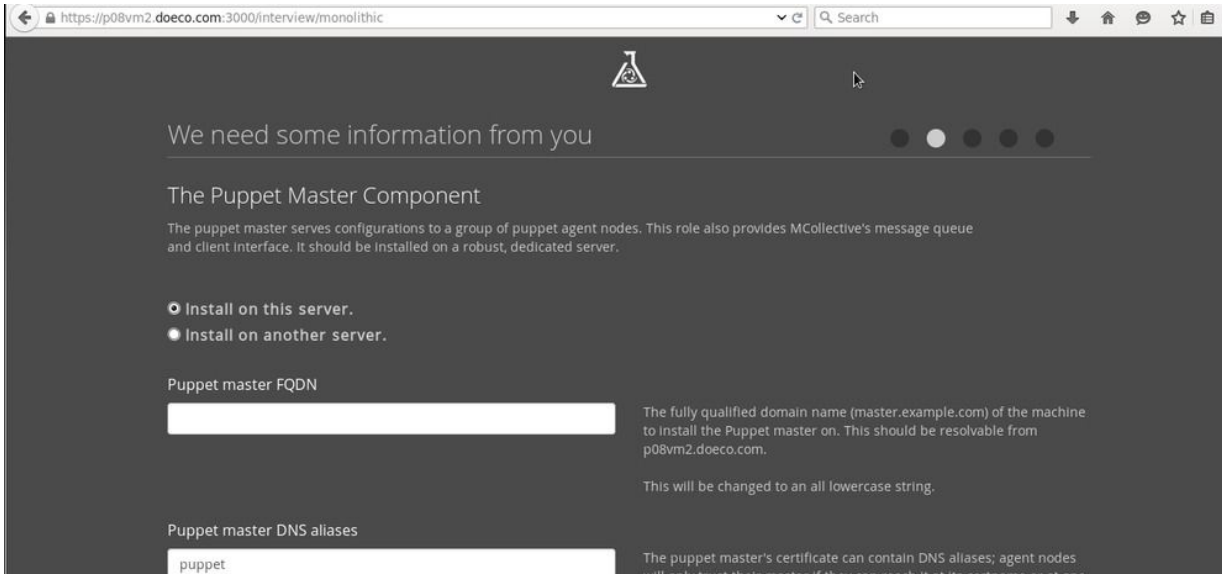
Then click on **Conform Security Exception button**

11. Now the puppet welcome page will be appeared in the browser. As shown below



Click on **Let's get started** button

12. In next step puppet will show two different options
    - Monolithic : it means if we want to install all puppet packages/modules in single system we can choose this option
    - Split : here we are splitting puppet, means if we want to install puppetDB on one system and remaining in another system choose this option
    - Presently I am using a single system so I go with Monolithic option
13. In next page we need to provide some basic information to puppet, the options are explained below

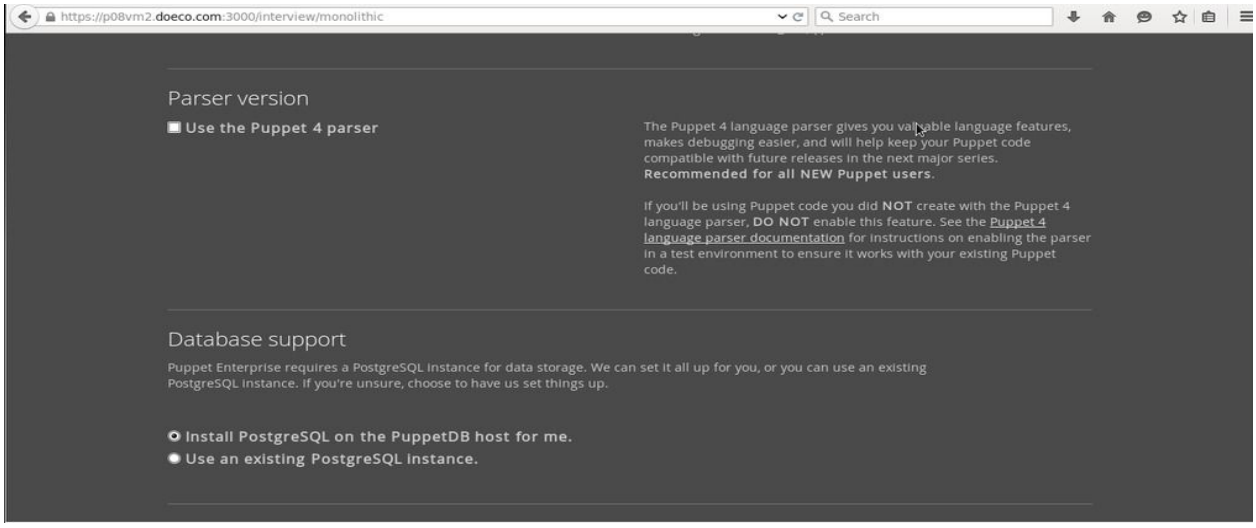


First it's ask about where puppet will install, in same system or other system.
Choose **Install on this server** option

puppet master FQDN : Provide your system hostname as puppet master FQDN

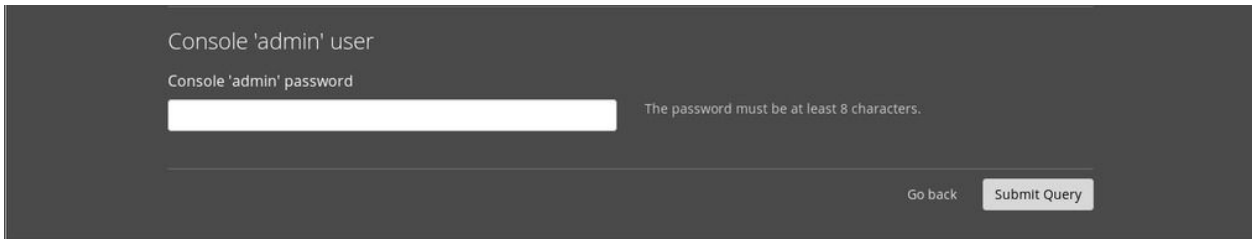
Example : p08vm1.doeco.com

Puppet master DNS aliases : provide an alias for puppet master DNS. By default it comes as puppet, If you want to change it you can



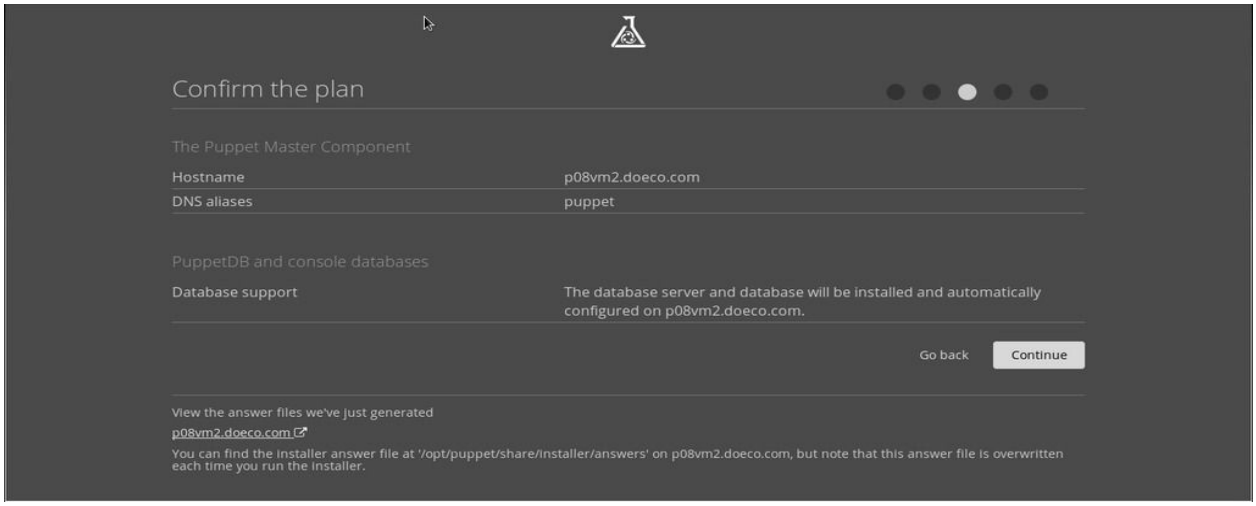
Parser version : present we don't need this version so leave this option

Database support : By default puppet has it's own DB if you want to use an external DB you can go with **Use an existing PostgreSQL instance** option.

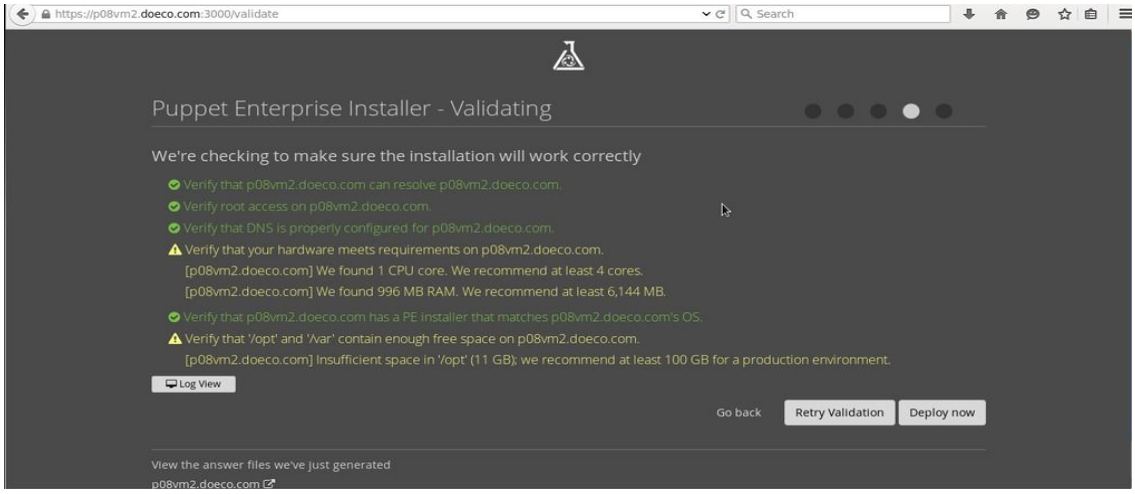


- Finally give puppet admin password and click on **Submit Query** button

And next a window will open with the information which is provided by us. Once check details and click on Continue option



14. In next step puppet validate the system configurations and provides report.



- In above image puppet validate system requirements, in above image 4 requirements are successful and two requirements are comes with error message .
- No problem with these errors because here i am using virtual machine so we can't that much RAM and etc.
- Next click on **Deploy now** option

15. In next step puppet will deploy all the dependencies and softwares. It will take around 30-40min time. After completion of deployment puppet shows you an option to start puppet, click on **Start** option

4.5 Testing process :

- Open browser and type URL as https://hostname.domain.com then puppet login page will be open, enter user name as admin and your admin password for login
- Open terminal and go to the path **/etc/puppetlabs/puppet/.** In this path you can find puppet.conf file

4.6 Results :

- Open terminal in puppetmaster and execute command **service pe-puppetserver status** then you can get output as follows

```
[root@p08vm1 ssl]# service pe-puppetserver status
pe-puppetserver (pid  19712) is running...
[root@p08vm1 ssl]#
```

5. Use Case 2: Puppet agent installation

5.1 Overview

In this use-case we install puppet agent software.

5.2 Assumptions

- Puppet master is installed in puppet server
- SSHD is up and running
- Timestamp (ntpd service) is up and running
- Vsftpd service is up and running in puppet server and ftp service is available in puppet agent system
- DNS is available,and nslookup is working
- Both systems hostnames are added to /etc/hosts file in both server and agent systems
- Puppet agent files are copied to directory in "/var/ftp/pub/" path
- Firewall service is stopped.
- Internet is available

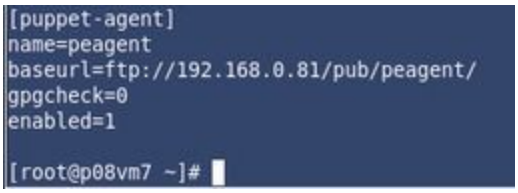
5.3 Scenario

Installing puppet agent by using vsftpd service

5.4 Steps

Note: There are different approaches to install puppet agent, in this use-case i am using **ftp** services

1. Open terminal and go to the path /etc/yum.repos.d/ and create a repo file
   ex : vi peagent.repo
2. Add path of the directory in puppet server where the puppet agent files copied. An example repo file is as shown below



3. After creating repo install puppet agent by using **yum.** Use following command

$ yum install pe-agent

4.

5.5 Testing process

- After pe-agent installed, check whether it's installed or not by using **"which puppet"** command. It will shows the path where puppet is stored

Error : When i execute "**which puppet**" it shows an error as follow

"bash: /usr/local/bin/puppet: No such file or directory"

Solution

The above error says that there is no puppet in bin, puppet is installed successfully but path is not set. To resolve this error simply export puppet path to bin

- First go to the path **/bin/** in bin export path of puppet by using following command

  $ export PATH=$PATH:/puppet path

  Example: $export PATH=$PATH:/opt/puppet/bin

  So here we are adding puppet bin path to system bin.

5.6 Results

- When we execute **"which puppet"** it will shows puppet path as follows

6. Use Case 3: Configuration of puppet master and agent, executing sample manifest for firewall

6.1 Overview

In this use-case we configure puppet master and puppet agent. And execute sample manifests

6.2 Assumptions:

- Puppet master is installed in server system (virtual machine or individual machine)
- Puppet agent is installed in client system (virtual machine or individual machine)
- Service ntpd is up and running
- Service SSH is up and running
- Hostnames of puppet master and agent added to hosts (/etc/puppet) file in both systems
- Firewall is stopped

6.3 Scenario

Configuring puppet master and agent, and executing sample manifest to make agent in desired state

6.4 Steps

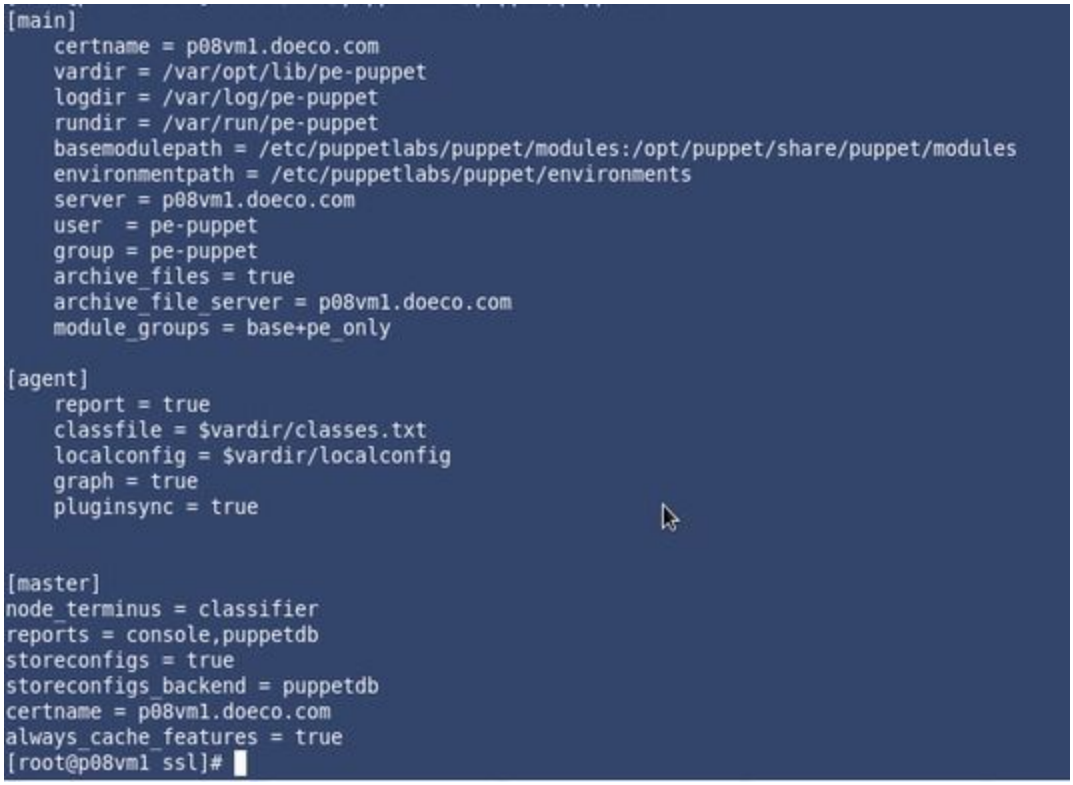
Puppet master configuration :

1. login to the puppet master system with necessary permissions.
2. We need to add server name to **puppet.conf** file which located in **/etc/puppetlabs/puppet/,** Because in puppet server is also client itself.
3. So we can add server name by two different ways
    a. We can execute command to set puppet server name

       Command : puppet config set server “FQDN of server”

       Example : puppet config set server p08vm1.doeco.com

    b. We can edit directly **puppet.conf** file. Go to the path /**etc/puppetlabs/puppet/** and open **puppet.conf** file (vi puppet.conf) add server name. The following image shows example **puppet.conf** file



4. Next execute "**puppet agent -t**" command. When we execute this command, system generates its catalog and sends that catalog to puppet master

Puppet agent configuration :

1. Login to the target system with necessary permissions.
2. Set puppet server to the system by executing following command

   Command : puppet config set server "FQDN of server"

   Example : puppet config set server p08vm1.doeco.com
3. Next execute "**puppet agent -t command**". This command generates a SSL key, this key is used as a certificate for communication between puppet agent and master.

NOTE : in server firewall should be in stopped state.

Communication between master and agent :

After generating SSL key we need to approve certificates from puppet master, for this

1. Login to the puppet master with necessary permissions.
2. Execute "**puppet cert list -all**" this shows all the certificates (SSL keys) stored in puppet master.
3. For accepting SSL key execute following command (in puppet server)

   puppet cert sign -a **#**this command approves all SSL keys at a time

   Note : we can approve SSL key for selected nodes also

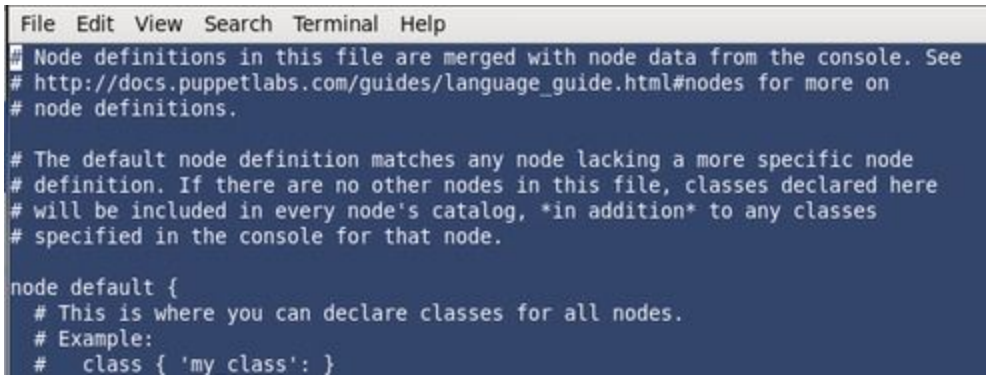
   Command :$puppet cert sign "hostname of node"

   Example : $puppet cert sign p08vm7.doeco.com

Executing sample manifest for firewall :

To check communication between puppet master and agent here we executing a sample manifest for firewall.

1. For executing manifests first go to the path /etc/puppetlabs/puppet/environments/production/manifests/
2. In manifest directory, you will see a **site.pp** file, this is the example manifest file which is comes automatically when you install puppet
3. We can create separate manifests for different nodes, but the manifest file extension should be **.pp**
4. Open the site.pp ($ vi site.pp) file go to the bottom of the file, It will appear as follows



5. Here "node default {" is starting section of manifest, "node default" means we can use this manifest on any agent node. If we want to create manifest for a particular nodes then start manifest with "node 'hostname' {"

Example : node p08vm7 {

6. Here class { 'my class':} is a default class, if we want use it uncomment the class (just remove # symbol), otherwise continue as default.
7. Use following manifest for firewall running, that means if you run this manifest in your agent node puppet master makes firewall running.

   Example :

   ```
   node default {
        service{ 'iptables':            #service is a resource type
       ensure => 'running',             #ensure is the desired state of file/service/package
       enable => 'true';
               }
       }
   ```

8. After writing manifest save the **site.pp** file

6.5Testing process

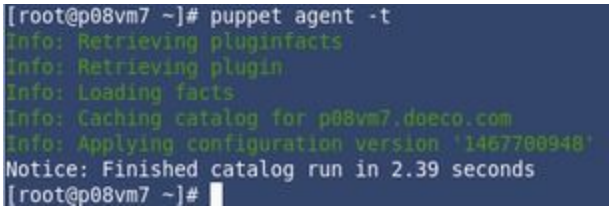
- Login to target agent node, check firewall status, If firewall is running stop file using following command

  Command : service iptables status #checking status of iptables

  Command : service iptables stop # stopping firewall
- Execute "**puppet agent -t**", after executing command check firewall status

6.6Results

- After executing manifest, the successful output will be like as follows



  If you want detailed information then execute "**puppet agent -t --debug**" command
- Check firewall status by using "**service iptables status**" command. Firewall should be active after executing manifest.

7.Use Case 4: Installing MYSQL using puppet master module

7.1 Overview

In this use-case Install MYSQL server using puppet master

7.2 Assumptions

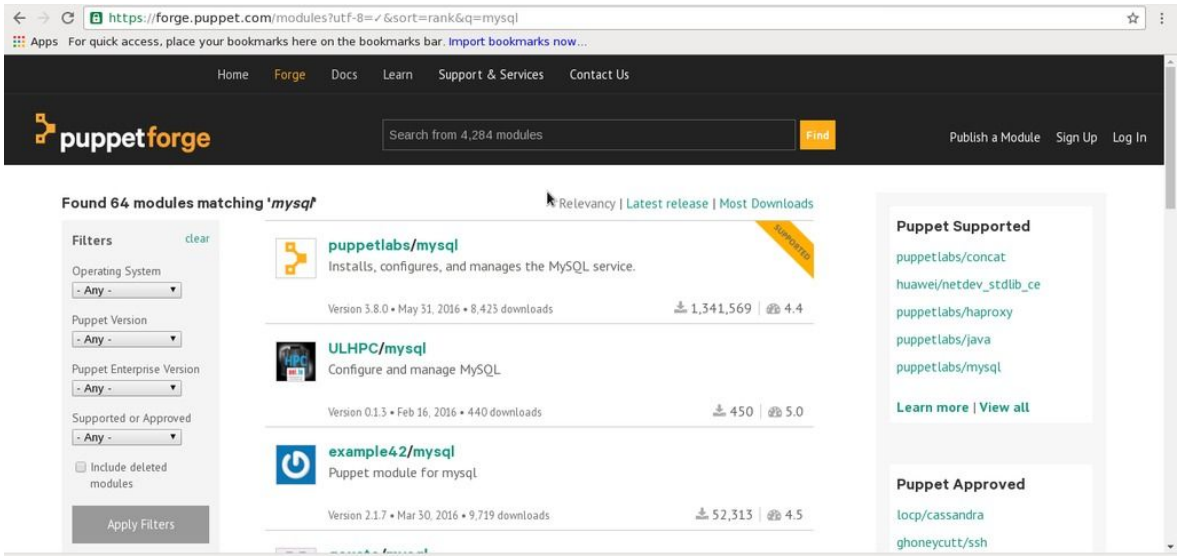
- Puppet master is installed in server system (virtual machine or individual machine)
- Puppet agent is installed in client system (virtual machine or individual machine)
- Service ntpd is up and running
- Service SSH is up and running
- Hostnames of puppet master and agent added to hosts (/etc/puppet) file in both systems
- Internet is working
- Puppet master manifests are executing in agent
- Should have basic knowledge about puppet modules

7.3 Scenario

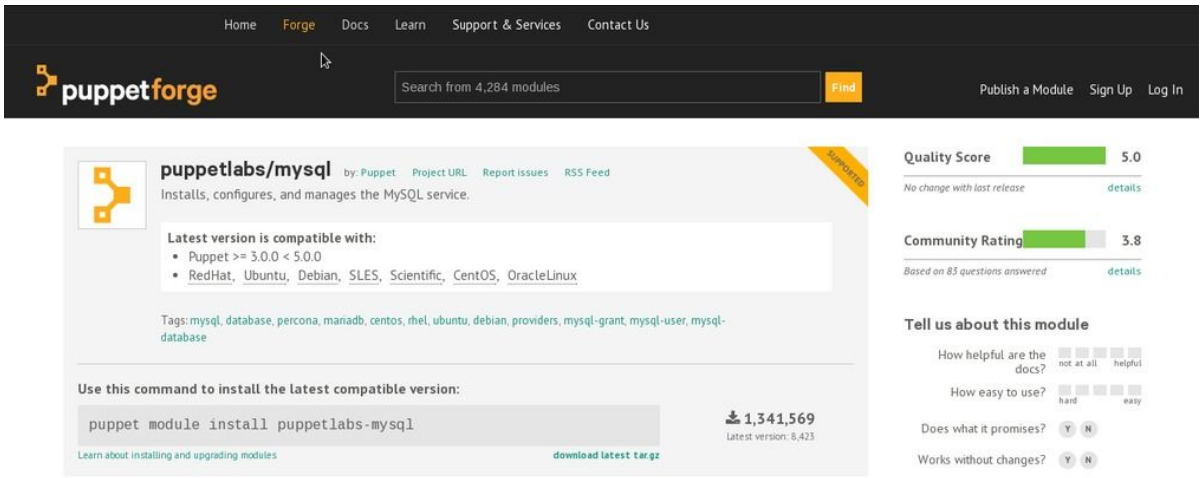
Installing MYSQL DB server in puppet agent by using puppet master module.

7.4 Steps

1. Login in to puppet master system with necessary permissions
2. Go to the path /etc/puppetlabs/puppet/environments/production/modules path and create a module (i created a module with the name of **mysqld**) for this process go to step 3 and continue the next
3. Open browser and go to https://forge.puppetlabs.com, search for mysql. After search finished you will see a list as follows



4. Click on first option “puppetlabs/mysql” you will see a new page as follows



5. Open terminal go to the path /etc/puppetlabs/puppet/environments/production/modules and execute command which shown in website in
   Command : puppet module install puppetlabs-mysql
6. mysql module will be downloaded change directory into module (which is created by us) and go to manifest, you will find **init.pp** file there, Open(vi init.pp) **init.pp** file, enter following code on the bottom of the init.pp file

```
node ‘hostname’{                          # replace your target system hostname here
        class {'::mysql::server':
        root_password => '$password',  #give your password here
                }
                }
```

7. Next go to the path /etc/puppetlabs/puppet/environments/production/manifests and add following code to **site.pp** file

   Code : node p08vm7{ #Here p08vm7 is hostname of targeted system
   class { “mysql”:}
   }

   Note : we can add code to **site.pp** file, otherwise we can create separate manifest files to separate nodes. For example we can create a separate manifest as **p08vm7.pp** and we can add code to that file.
8. Next login to target agent system and run “**puppet agent -t**” command

7.5Testing process

- Login to target agent system and execute “**puppet agent -t**” command
- After successfully executing above command check the status of mysql server using following command

  command : service mysqld status

  Output will be as follows

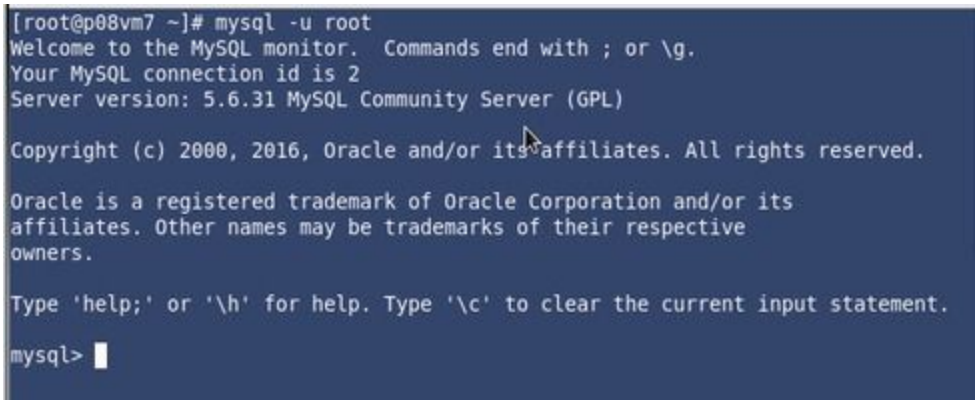
  [root@p08vm7 ~]# service mysqld status
  mysqld (pid 10528) is running...

- You can login into mysql with following command

  command : **mysql -u root**

7.6Results

- Finally we can login into mysql using “**mysql -u root**” command


[root@p08vm7 ~]# mysql -u root
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2
Server version: 5.6.31 MySQL Community Server (GPL)
Copyright (c) 2000, 2016, Oracle and/or its affiliates. All rights reserved.
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql>

8.Use Case 5: Installing JBoss server using puppet master module

8.1 Overview

In this use-case Install JBoss server using puppet master module

8.2 Assumptions

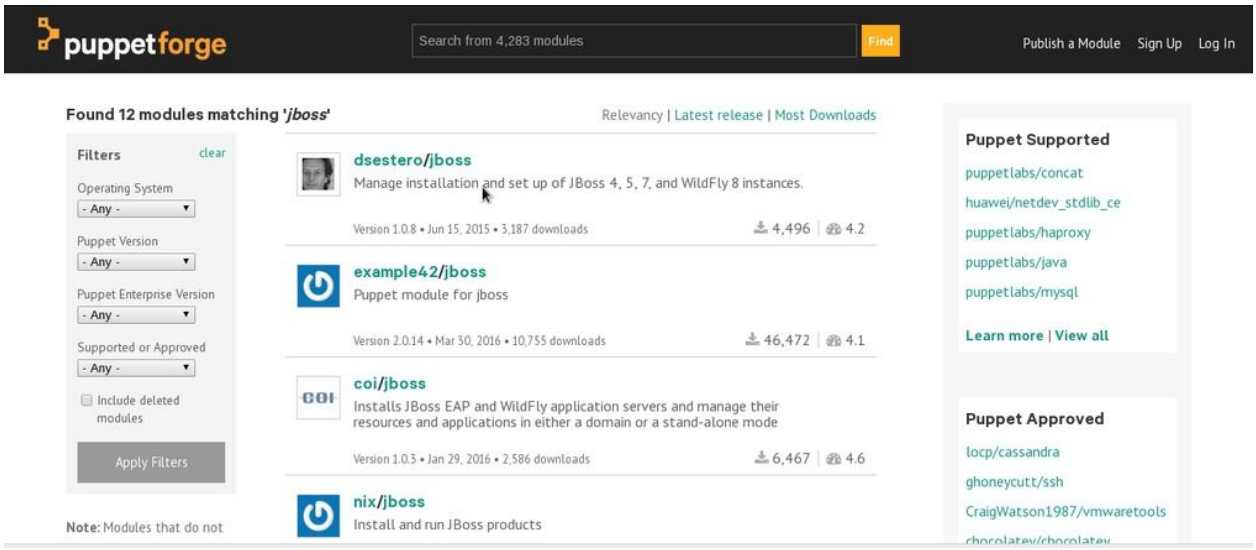
- Puppet master is installed in server system (virtual machine or individual machine)
- Puppet agent is installed in client system (virtual machine or individual machine)
- Service ntpd is up and running
- Service SSH is up and running
- Hostnames of puppet master and agent added to hosts (/etc/puppet) file in both systems
- Internet is working
- Puppet master manifests are executing in agent
- Having basic knowledge about puppet modules

8.3 Scenario

Installing JBoss server in puppet agent by using puppet master modules

8.4 Steps

1. Login in to puppet master system with necessary permissions
2. Go to the path /etc/puppetlabs/puppet/environments/production/modules path and create module (i created a module with the name of **jbossd**)
3. Open browser and go to https://forge.puppetlabs.com, search for **JBoss**. After search finished you will see a list as follows



4. Click on first option “example42/jboss”
5. Open terminal go to the path /etc/puppetlabs/puppet/environments/production/modules and execute command which shown in website in
   Command : puppet module install example42-jboss
6. JBoss module will be downloaded,change directory into module (which is created by us) and go to manifest, you will find **init.pp** file there and open(vi init.pp) **init.pp** file, enter following code on the bottom of the init.pp file\

```
node hostname{              #hostname of your agent node
  class { 'jboss':}
}
```

7. Next go to your manifest in etc/puppetlabs/puppet/environments/production/manifest path open your manifest, i created new manifest with my node name p08vm8.pp. Add following lines

```
node hostname{              #hostname of your agent node
  class { 'jboss':}
}
```

8.5 Testing process

- Login to target agent system and execute "**puppet agent -t**" command
- After successfully executing above command check the status of mysql server using following command
  command : service jboss status

  Output will be as follows

  [root@p08vm7 ~]# service jboss status

  Jboss-as is running (pid 2750)
- Open jboss in your browser with http://yourip:8080/ or http://localhost:8080/

8.6 Results

9.Use Case 6: Ensure both Virtual machines are in desired state

9.1 Overview

Ensure both virtual machines (MYSQL and JBOSS) are in desired state

9.2 Assumptions

- Puppet master is installed in server system (virtual machine or individual machine)
- Puppet agent is installed in client systems (virtual machine or individual machine)
- MYSQL is installed and running in puppet agent
- JBOSS service is installed and running puppet agent
- Service ntpd is up and running
- Service SSH is up and running
- Hostnames of puppet master and agent added to hosts (/etc/puppet) file in both systems
- Internet is working
- Puppet master manifests are executing in agent

9.3 Scenario

Ensuring desired state for MYSQL and JBOSS, that means MYSQL and JBOSS are up and running

9.4 Steps

Here i am testing whether my puppet agents are in desired state or not.
MYSQL :

1. Login to puppet agent machine where MYSQL is installed.
2. Check MYSQL service status by using "**service mysqld status**"
3. Now stop MYSQL services by using "**service mysqld stop**"
4. Now by using puppet i am starting **mysql** services. To ensure mysql just execute following command
   command : **puppet agent -t**
5. Now check status of MYSQL service. The following image shows you an overview about all this process

Jboss:

1. Login to puppet agent machine whereJBOSS is installed.
2. Check MYSQL service status by using "**service jboss status**"
3. Now stop MYSQL services by using "**service jboss stop**"
4. Now by using puppet i am starting **jboss** services. To ensure mysql just execute following command
   command : **puppet agent -t**
5. Now check status of JBOSS service. The following image shows you an overview about all this process