**Academic Year: 2023-24 Semester: V**

**Class / Branch: TEIT Subject: DevOps Lab**

**Name of Instructor: Prof. Sonal Jain/Prof. Neha Deshmukh**

# Experiment No. 11

**Aim: To implement Software Configuration Management and provisioning using Puppet Blocks(Manifest, Modules)**

**Theory:**

In Puppet, all the programs which are written using Ruby programming language and saved with an

extension of .pp are called manifests. In general terms, all Puppet programs which are built with an

intension of creating or managing any target host machine is called a manifest. All the programs written

in Puppet follow Puppet coding style.The core of Puppet is the way resources are declared and how these resources are representing their state. In any manifest, the user can have a collection of different kind of resources which are grouped together

using class and definition

**Manifest File Workflow**

Puppet manifest consists of the following components −

• Files (these are plain files where Puppet has nothing to do with them, just to pick them up and place

them in the target location)

• Resources

• Templates (these can be used to construct configuration files on the node).

• Nodes (all the definition related to a client node is defined here)

• Classes

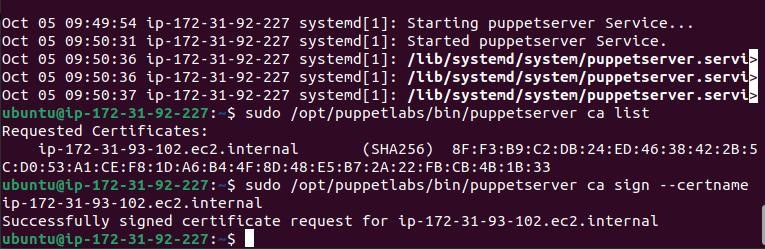
In Puppet, all manifest files use Ruby as their encoding language and get saved with .pp extension."Import" statement in many manifest are used for loading files when Puppet starts.In order to import all files contained in a directory, you can use the import statement in another way like import 'clients/\*'. This will import all .pp files inside that directory.

**Applying Manifests on Puppet master**

**Step 1**:Signing Certificates on the Puppet Master

List certificate requests

sudo /opt/puppetlabs/bin/puppetserver ca list



**Step2**: Sign a specific certificate request

sudo /opt/puppetlabs/bin/puppetserver ca sign --certname <Agent\_CertName>

Note: Agent Certname must be replaced with **Private IP DNS Name of puppet\_slave**

**Step3:** Create a Puppet manifest file

sudo nano /etc/puppetlabs/code/environments/production/manifests/site.pp



**Step4:** Add content like the following example:

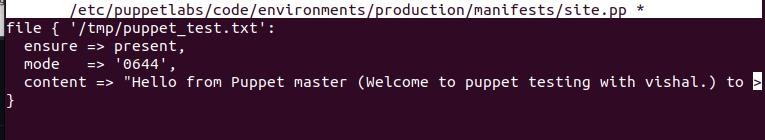
file { '/tmp/puppet\_test.txt':

ensure => present,

mode => '0644',

content => "Hello from Puppet master (Welcome to puppet testing with Sonal) to agent on IP address ${ipaddress\_eth0}\n",

}



**Step 5:** Trigger Puppet Agent to apply the changes (run on puppet Agent)

sudo /opt/puppetlabs/bin/puppet agent –test

**Step 6**:Check the content of the test file (run on puppet Agent)

sudo cat /tmp/puppet\_test.txt

# Conclusion:

In this experiment, we have successfully signed security certificate between puppet Master and Slave to run manifest from master and viewing on slave.