

5.x Puppet

This is a work in progress...

Creating a Puppet Master Server

For this instruction I am using 2 Virtual Machines on Ubuntu 16.04 LTS

```
wget https://apt.puppetlabs.com/puppetlabs-release-pc1-xenial.deb
sudo dpkg -i puppetlabs-release-pc1-xenial.deb
sudo apt-get update
sudo apt-get install puppetserver
```

Modifying the Memory Limit on Puppet Master

Puppet default memory use is 2GB edit the puppetserver file to change it to 512mb.

```
sudo vi /etc/default/puppetserver
```

Adjust the line

```
# Modify this if you'd like to change the memory allocation, enable JMX,
etc
JAVA_ARGS="-Xms2g -Xmx2g -XX:MaxPermSize=256m"
```

to

```
# Modify this if you'd like to change the memory allocation, enable JMX,
etc
JAVA_ARGS="-Xms512m -Xmx512m -XX:MaxPermSize=256m"
```

Defining the DNS for the Server

For the Puppet Agents to find the Puppet Master server the DNS needs to be defined in the configuration file.

```
sudo vi /etc/puppetlabs/puppet/puppet.conf
```

Add this to the end of the file since our server ip is 192.168.237.130 we will use this in our example.

```
dns_alt_names = hostname,192.168.237.130
[main]
certname = 192.168.237.130
server = 192.168.237.130
environment = production
runinterval = 5m
```

Start the Puppet Server and Enable Start on Reboot

```
sudo /opt/puppetlabs/bin/puppet resource service puppet ensure=running
enable=true
```

Should be followed by

```
Notice: /Service[puppet]/ensure: ensure changed 'stopped' to 'running'
service { 'puppet':
  ensure => 'running',
  enable => 'true',
}
```

Creating the Puppet Agent Node

```
wget https://apt.puppetlabs.com/puppetlabs-release-pc1-xenial.deb
sudo dpkg -i puppetlabs-release-pc1-xenial.deb
sudo apt-get update
sudo apt-get install puppet-agent
```

Configuring the Config File to Find the Puppet Master Server

```
sudo vi /etc/puppetlabs/puppet/puppet.conf

[main]
certname = puppetagent
server = 192.168.237.130
environment = production
runinterval = 20m
```

Start the Puppet Agent

```
sudo /opt/puppetlabs/bin/puppet resource service puppet ensure=running
enable=true
```

```
Notice: /Service[puppet]/ensure: ensure changed 'stopped' to 'running'
service { 'puppet':
  ensure => 'running',
  enable => 'true',
}
```

Checking if the Puppet Agent CSR Reached the Puppet Master Server

On the Puppet Master server run this command to check if it has received the puppet agent CSR.

```
setupadmin@bbsvc1:~$ sudo /opt/puppetlabs/bin/puppet cert list

"puppetagent" (SHA256)
52:3A:3A:13:EC:0C:1F:7D:EE:81:32:94:20:8F:DB:C8:9C:31:03:25:E2:A0:A6:8A:AE
:DF:05:79:CC:7C:35:96
```

Now that you have confirmed the certificate sign it to connect the node.

```
sudo /opt/puppetlabs/bin/puppet cert sign puppetagent

Signing Certificate Request for:
"puppetagent" (SHA256)
52:3A:3A:13:EC:0C:1F:7D:EE:81:32:94:20:8F:DB:C8:9C:31:03:25:E2:A0:A6:8A:AE
:DF:05:79:CC:7C:35:96
Notice: Signed certificate request for puppetagent
Notice: Removing file Puppet::SSL::CertificateRequest puppetagent at
'/etc/puppetlabs/puppet/ssl/ca/requests/puppetagent.pem'
```

To check all certificates signed with the Puppet Master.

```
sudo /opt/puppetlabs/bin/puppet cert list --all

+ "192.168.237.130" (SHA256)
F0:98:99:57:FA:C3:91:19:D4:7C:DA:20:7C:B6:78:FD:87:FA:44:0D:B0:F4:A1:4C:20
:2A:BB:56:4F:09:DB:BF (alt names: "DNS:192.168.237.130", "DNS:hostname",
"DNS:192.168.237.130")
+ "puppetagent" (SHA256)
ED:5C:43:36:69:6C:9F:9C:15:44:40:93:0D:3C:C4:64:7F:3D:78:CF:90:B0:2D:0B:AB
:D1:6F:54:E5:34:EC:DA
```

How to Execute Puppet Scripts

sudo puppet apply "puppet script"

Updating Ubuntu

```
exec { 'apt-update': # exec resource named 'apt-update'
  command => '/usr/bin/apt-get update' # command this resource will run
}
```

Installing Apache2

```
package { 'apache2':
  require => Exec['apt-update'], # require 'apt-update' before installing
  ensure => installed,
}
```

Ensuring Apache2 is running

```
service { 'apache2':
  ensure => running,
}
```

Creating a Group and Assigning a GID

```
group { 'serveradmin':
  ensure => 'present',
  gid => '3000',
}
```

Creating a User

```
user { 'serveradmin':
  ensure => 'present',
  managehome => 'true',
  home => '/home/serveradmin',
  comment => 'Server Admin',
  groups => 'serveradmin',
  password => 'serveradmin',
  password_max_age => '99999',
  password_min_age => '0',
  shell => '/bin/bash',
  uid => '3000',
}
```

Installing a Database

```
# install postgresql package
package { 'postgresql':
  require => Exec['apt-update'], # require 'apt-update' before installing
  ensure => installed,
}
```

```
# ensure postgresql service is running
service { 'postgresql':
  ensure => running,
}
```

Creating a Database Instance

```
class { 'postgresql::server': }

postgresql::server::db { 'testdb':
```

```
user => 'admin',
password => postgresql_password('admin', 'c00kies'),
owner => 'admin',
}
```

```
class mymodule::myclass{
file { 'my_bash_script':
ensure => 'file',
path => '/home/setupadmin/runthisfile.sh',
owner => 'setupadmin',
group => 'setupadmin',
mode => '0755', # Use 0700 if it is sensitive
notify => Exec['run_my_script'],
}
```

```
exec { 'run_my_script':
#path => ["/usr/bin", "/usr/sbin", "/bin/bash"],
command => '/home/setupadmin/runthisfile.sh',
refreshonly => true
}
}
```

```
include mymodule::myclass
```