Starting the Puppet Lab on Docker:

docker run --rm -dit --expose 8140 --name puppet --hostname puppet --network ansible\_nw sreeharshav/puppetmaster:nocasetup

docker run --rm -dit --name puppetagent1 --hostname puppetagent1 --network ansible\_nw sreeharshav/puppetagent:V2

docker run --rm -dit --name puppetagent2 --hostname puppetagent2 --network ansible\_nw sreeharshav/puppetagent:V2

docker run --rm -dit --name puppetagent3 --hostname puppetagent3 --network ansible\_nw sreeharshav/puppetagent:V2

docker run --rm -dit --name puppetagent4 --hostname puppetagent4 --network ansible\_nw sreeharshav/puppetagent:V2

Puppet Agent Testing

#!/bin/bash

echo "Running on puppetagent1"

docker exec -it puppetagent1 puppet agent -t

echo "Running on puppetagent2"

docker exec -it puppetagent2 puppet agent -t

echo "Running on puppetagent3"

docker exec -it puppetagent3 puppet agent -t

echo "Running on puppetagent4"

docker exec -it puppetagent4 puppet agent -t

echo "AGENT RUNNING COMPLETED"

Puppet Parameters:

class users::params {

$user1 = 'sree'

$user2 = 'harsha'

$user3 = 'veerapalli'

$user1home = '/home/user1home'

$user2home = '/home/user2home'

$user3home = '/home/user3home'

}

# A description of what this class does

#

# @summary A short summary of the purpose of this class

#

# @example

# include users::newusers

class users::newusers inherits users::params {

user { $::users::params::user1:

ensure => 'present',

home => $::users::params::user1home,

password => '$1$3h12aiE4$zVINSwxSKTZ3p1FqXll/Z.',

password\_max\_age => '99999',

password\_min\_age => '0',

shell => '/bin/bash',

uid => '502',

}

user { $::users::params::user2:

ensure => 'present',

home => $::users::params::user2home,

password => '$1$3h12aiE4$zVINSwxSKTZ3p1FqXll/Z.',

password\_max\_age => '99999',

password\_min\_age => '0',

shell => '/bin/bash',

uid => '502',

}

user { $::users::params::user3:

ensure => 'present',

home => $::users::params::user3home,

password => '$1$3h12aiE4$zVINSwxSKTZ3p1FqXll/Z.',

password\_max\_age => '99999',

password\_min\_age => '0',

shell => '/bin/bash',

uid => '502',

}

}

node default {

notify {'DEFAULT NOTIFICATION':

message => "This is defaut node configuration"

}

}

node 'puppetagent5','puppetagent6','puppetagent7','puppetagent8' {

include users::newusers

include nginx::install

file { '/etc/motd':

ensure => file,

owner => 'root',

content => "

This server is a property of XYZ Inc.

SYSTEM INFO

============

Hostname : ${::fqdn}

IP Address : ${::ipaddress}

Memory : ${::memory['system']['total']}

Cores : ${::processors['count']}

OS : ${::os['distro']['description']}

"

}

}

===========================================================

CLASS PARAMETERS

#app.pp

node default {

notify {'NODE DEFAULT - NO CONFIGURATION':}

}

node 'puppetagent1','puppetagent2'{

class {'testusers':

user1 => 'SSree',

user2 => 'SHarsha',

user3 => 'SVeerapalli',

sure => 'present',

}

}

node 'puppetagent3','puppetagent4'{

class {'testusers':

user1 => 'STestuser1',

user2 => 'STestuser2',

user3 => 'STestuser3',

sure => 'present'

}

}

#params.pp

class users::params {

$user1 = 'testuser1'

$user2 = 'testuser2'

$user3 = 'testuser3'

}

#init.pp

class testusers (

$user1,

$user2,

$user3,

$sure,

) {

user { $user1:

ensure => $sure,

password => '$1$3h12aiE4$zVINSwxSKTZ3p1FqXll/Z.',

password\_max\_age => '99999',

password\_min\_age => '0',

shell => '/bin/bash',

#uid => '501',

}

user { $user2:

ensure => $sure,

password => '$1$3h12aiE4$zVINSwxSKTZ3p1FqXll/Z.',

password\_max\_age => '99999',

password\_min\_age => '0',

shell => '/bin/bash',

#uid => '502',

}

user { $user3:

ensure => $sure,

password => '$1$3h12aiE4$zVINSwxSKTZ3p1FqXll/Z.',

password\_max\_age => '99999',

password\_min\_age => '0',

shell => '/bin/bash',

#uid => '503',

}

}

<https://axelauvinen.wordpress.com/2013/11/23/puppet-defined-types-class-parameters-ubuntu-12-04/comment-page-1/>

=====================================================================

#/etc/puppetlabs/puppet/hiera.yaml - After editing restart puppetserver service

---

:backends:

- yaml

- json

:yaml:

:datadir: "/etc/puppetlabs/code/environments/production/hieradata"

:json:

:datadir: "/etc/puppetlabs/code/environments/production/hieradata"

:hierarchy:

- "nodes/%{::trusted.certname}"

- "%{::environment}"

- "%{::osfamily}"

- "common"

#/etc/puppetlabs/code/environments/production/manifests/init.pp

node default {

notify {hiera(motd):}

include users

}

#/etc/puppetlabs/code/environments/production/manifests/users.pp

class users {

$user\_defaults = { ensure => present }

$group\_defaults = { ensure => present }

create\_resources('user', hiera\_hash(users), $user\_defaults)

create\_resources('group', hiera\_hash(groups), $group\_defaults)

}

#/etc/puppetlabs/code/environments/production/hieradata/common.yaml

---

motd: 'Welcome to Pune'

groups:

abc: {}

xyz: {}

users:

foo:

gid: abc

bar:

gid: abc

foobar:

gid: abc

baz:

gid: xyz

groups: [ 'abc' ]

=========================================================================

# ~/setup.rb

package 'tree' do

action :install

end

package 'ntp'

file '/etc/motd' do

content 'This server is the property of ...'

end

service 'ntpd' do

action [:enable, :start]

end