

Puppet:

Puppet is an open-source configuration management tool, which runs on many UNIX like systems and even in Microsoft windows and includes its own declarative language to describe system configuration. It was produced by Puppet Labs, in 2005. It is written in Ruby.

The user describes system resources and their state, either using Puppet's declarative language or a Ruby DSL (Domain Specific Language). This information is stored in files called "Puppet manifests". Puppet discovers system information via a utility called **Facter**, and compiles the Puppet **manifests** into a system-specific catalog containing resources and resource dependency, which are applied against the target systems. Any actions taken by Puppet are then reported.

For installing puppet we need two instances:

1. Puppet Master
2. Puppet Agent

Puppet master installation at Ubuntu Server 14.04:

To install Puppet master we will get latest repository and install the Puppet as follows:

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

```
root@ip-172-31-30-199:~# wget https://apt.puppetlabs.com/puppetlabs-release-trusty.deb
--2016-10-19 09:59:31-- https://apt.puppetlabs.com/puppetlabs-release-trusty.deb
Resolving apt.puppetlabs.com (apt.puppetlabs.com)... 198.58.114.168, 2600:3c00::f03c:91ff:fe69:6bf0
Connecting to apt.puppetlabs.com (apt.puppetlabs.com)|198.58.114.168|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 16944 (17K) [application/x-debian-package]
Saving to: 'puppetlabs-release-trusty.deb'

100%[=====] 16,944
2016-10-19 09:59:31 (281 KB/s) - 'puppetlabs-release-trusty.deb' saved [16944/16944]
```

dpkg -i puppetlabs-release-trusty.deb

```
root@ip-172-31-30-199:~# dpkg -i puppetlabs-release-trusty.deb
Selecting previously unselected package puppetlabs-release.
(Reading database ... 51172 files and directories currently installed.)
Preparing to unpack puppetlabs-release-trusty.deb ...
Unpacking puppetlabs-release (1.1-1) ...
Setting up puppetlabs-release (1.1-1) ...
```

Now update apt's list of available packages:

apt-get update

Now install puppet master package by using the command,

sudo apt-get install puppetmaster

```

root@ip-172-31-30-199:~# apt-get install puppetmaster
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  augeas-lenses debconf-utils facter hiera libaugeas-ruby libaugeas0
  libruby1.9.1 puppet-common puppetmaster-common ruby ruby-augeas ruby-json
  ruby-shadow ruby1.9.1 virt-what
Suggested packages:
  augeas-doc augeas-tools ruby-selinux libselinux-ruby1.8 librrd-ruby1.9.1
  librrd-ruby1.8 apache2 nginx puppet-el vim-puppet stompserver ruby-stomp
  libstomp-ruby1.8 rdoc ruby-ldap libldap-ruby1.8 puppetdb-terminus ri
  ruby-dev ruby1.9.1-examples ri1.9.1 graphviz ruby1.9.1-dev ruby-switch
The following NEW packages will be installed:
  augeas-lenses debconf-utils facter hiera libaugeas-ruby libaugeas0
  libruby1.9.1 puppet-common puppetmaster puppetmaster-common ruby ruby-augeas
  ruby-json ruby-shadow ruby1.9.1 virt-what
0 upgraded, 16 newly installed, 0 to remove and 53 not upgraded.
Need to get 4,574 kB of archives.
After this operation, 23.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] █

```

Once you press Y and hit enter the installation will continue, once the installation is done we can verify it by running the command:

puppet -V

```

root@ip-172-31-30-199:~# puppet -V
3.8.7
root@ip-172-31-30-199:~# █

```

We need to lock the version of puppet, since the change from version to version an occasionally cause your puppet environment to stop working properly. It can be done by editing the file:

vi /etc/apt/preferences.d/00-puppet.pref

```

root@ip-172-31-30-199:~# vi /etc/apt/preferences.d/00-puppet.pref

```

Add the entries in the newly created file as:

```

# /etc/apt/preferences.d/00-puppet.pref
Package: puppet puppet-common puppetmaster-passenger
Pin: version 3.8*
Pin-Priority: 501
~

```

Next we will change the configuration file as follows:

vi /etc/puppet/puppet.conf

```

[main]
logdir=/var/log/puppet
vardir=/var/lib/puppet
ssldir=/var/lib/puppet/ssl
rundir=/var/run/puppet
factpath=$vardir/lib/facter
templatedir=$confdir/templates

[master]
# These are needed when the puppetmaster is run by passenger
# and can safely be removed if webrick is used.
ssl_client_header = SSL_CLIENT_S_DN
ssl_client_verify_header = SSL_CLIENT_VERIFY

```

Just comment the line **templatedir=\$confdir/templates** and save the file.

```

[main]
logdir=/var/log/puppet
vardir=/var/lib/puppet
ssldir=/var/lib/puppet/ssl
rundir=/var/run/puppet
factpath=$vardir/lib/facter
# templatedir=$confdir/templates

[master]
# These are needed when the puppetmaster is run by passenger
# and can safely be removed if webrick is used.
ssl_client_header = SSL_CLIENT_S_DN
ssl_client_verify_header = SSL_CLIENT_VERIFY

```

Now we will stop and then start the services:

```

root@ip-172-31-30-199:~# service puppetmaster stop
* Stopping puppet master
root@ip-172-31-30-199:~# service puppetmaster start
* Starting puppet master
root@ip-172-31-30-199:~#

```

Now the master puppet server is ready.

Puppet Agent installation at Ubuntu Server 14.04:

Download the puppet agent node by using the command:

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

```

root@ip-172-31-47-74:~# wget https://apt.puppetlabs.com/puppetlabs-release-trusty.deb
--2016-10-19 10:09:24-- https://apt.puppetlabs.com/puppetlabs-release-trusty.deb
Resolving apt.puppetlabs.com (apt.puppetlabs.com)... 198.58.114.168, 2600:3c00::f03c:91ff:fe69:6bf0
Connecting to apt.puppetlabs.com (apt.puppetlabs.com)[198.58.114.168]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 16944 (17K) [application/x-debian-package]
Saving to: 'puppetlabs-release-trusty.deb'

100%[=====] 16,944

2016-10-19 10:09:25 (272 KB/s) - 'puppetlabs-release-trusty.deb' saved [16944/16944]

root@ip-172-31-47-74:~#

```

```
# dpkg -i puppetlabs-release-trusty.deb
```

```
root@ip-172-31-47-74:~# dpkg -i puppetlabs-release-trusty.deb
Selecting previously unselected package puppetlabs-release.
(Reading database ... 51172 files and directories currently installed.)
Preparing to unpack puppetlabs-release-trusty.deb ...
Unpacking puppetlabs-release (1.1-1) ...
Setting up puppetlabs-release (1.1-1) ...
root@ip-172-31-47-74:~#
```

Now update apt's list of available packages:

```
# apt-get update
```

Install puppet by:

```
# apt-get install puppet
```

```
root@ip-172-31-47-74:~# apt-get install puppet
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  augeas-lenses debconf-utils facter hiera libaugeas-ruby libaugeas0
  libruby1.9.1 puppet-common ruby ruby-augeas ruby-json ruby-shadow ruby1.9.1
  virt-what
Suggested packages:
  augeas-doc augeas-tools puppet-el vim-puppet ruby-selinux libselenium-ruby1.8
  librrd-ruby1.9.1 librrd-ruby1.8 ri ruby-dev ruby1.9.1-examples ri1.9.1
  graphviz ruby1.9.1-dev ruby-switch
Recommended packages:
  rdoc
The following NEW packages will be installed:
  augeas-lenses debconf-utils facter hiera libaugeas-ruby libaugeas0
  libruby1.9.1 puppet puppet-common ruby ruby-augeas ruby-json ruby-shadow
  ruby1.9.1 virt-what
0 upgraded, 15 newly installed, 0 to remove and 53 not upgraded.
Need to get 4,560 kB of archives.
After this operation, 23.7 MB of additional disk space will be used.
Do you want to continue? [Y/n]
```

Once you press **Y** and hit enter the installation will continue, once the installation is done we can verify it by running the command:

```
# puppet -V
```

```
root@ip-172-31-47-74:~# puppet -V
3.8.7
root@ip-172-31-47-74:~#
```

We need to lock the version of puppet, since the change from version to version an occasionally cause your puppet environment to stop working properly. It can be done by editing the file:

```
# vi /etc/apt/preferences.d/00-puppet.pref
```

```
root@ip-172-31-47-74:~# vi /etc/apt/preferences.d/00-puppet.pref
```


Add the entries in the newly created file as:

```
# /etc/apt/preferences.d/00-puppet.pref
Package: puppet puppet-common puppetmaster-passenger
Pin: version 3.8.3
Pin-Priority: 501
```

Next we will change the configuration file as follows:

```
# vi /etc/puppet/puppet.conf
```

Just comment the lines as mentioned above in puppet master installation and add the agent entries and save the file.

```
[main]
logdir=/var/log/puppet
vardir=/var/lib/puppet
ssldir=/var/lib/puppet/ssl
rundir=/var/run/puppet
factpath=$vardir/lib/facter
#templatedir=$confdir/templates

#[master]
# These are needed when the puppetmaster is run by passenger
# and can safely be removed if webrick is used.
#ssl_client_header = SSL_CLIENT_S_DN
#ssl_client_verify_header = SSL_CLIENT_VERIFY

[agent]
server = ip-172-31-30-199.us-west-2.compute.internal
```

Next we need to edit the file vi /etc/default/puppet and make changes from no to yes as show below:

```
# vi /etc/default/puppet
```

```
# Defaults for puppet - sourced by /etc/init.d/puppet

# Enable puppet agent service?
# Setting this to "yes" allows the puppet agent service to run.
# Setting this to "no" keeps the puppet agent service from running.
START=yes

# Startup options
DAEMON_OPTS=""
~
~
~
~
```

Now we are ready to start the puppet service, it will be done as follows:

```
# service puppet start
```

Now our client machine is ready to communicate with Puppet Master Server.

Cert exchange from Puppet master to puppet client:

To view such cert request run the command at Puppet Master Ubuntu server.

```
# puppet cert list
```

```
root@ip-172-31-30-199:~# puppet cert list
"ip-172-31-47-74.us-west-2.compute.internal" (SHA256) 44:4E:E7:E0:DA:53:C2:05:51:8C:5E:74:3C:29:70:20:B8:61:37:4E:9F:87:61:13:D0:CB:EF:74:B4:E4:97:48
root@ip-172-31-30-199:~#
```

Now the puppet master server must sign the cert requested from puppet client. It can be done as follows:

```
# puppet cert sign <private DNS>
```

```
root@ip-172-31-30-199:~# puppet cert sign ip-172-31-47-74.us-west-2.compute.internal
Notice: Signed certificate request for ip-172-31-47-74.us-west-2.compute.internal
Notice: Removing file Puppet::SSL::CertificateRequest ip-172-31-47-74.us-west-2.compute.internal at '/var/lib/puppet/ssl/ca/requests/ip-172-31-47-74.us-west-2.compute.internal.pem'
root@ip-172-31-30-199:~#
```

It means the request from Agent machine is accepted at Puppet master machine. We can check that with command as well:

```
# puppet cert list -all
```

```
root@ip-172-31-30-199:~# puppet cert list -all
+ "ip-172-31-30-199.us-west-2.compute.internal" (SHA256) 98:12:D4:00:73:93:2F:4E:77:58:6B:50:9F:41:15:D4:D1:A0:B6:30:55:3A:40:70:B4:65:59:E0:18:5E:20:C6 (alt names: "DNS:ip-172-31-30-199.us-west-2.compute.internal", "DNS:puppet", "DNS:puppet.us-west-2.compute.internal")
+ "ip-172-31-47-74.us-west-2.compute.internal" (SHA256) B3:93:36:F7:DD:EF:0D:CD:DA:64:A1:0F:2D:51:58:7D:24:60:7C:CC:39:BF:EC:D7:F9:89:46:59:58:90:75:AB
root@ip-172-31-30-199:~#
```

You will see a list of all of the requests. Signed requests are preceded by a + and unsigned requests do not have the +.

To view the client cert fingerprint at Puppet Client Ubuntu desktop, run:

```
# puppet agent --fingerprint
```

```
root@ip-172-31-47-74:~# puppet agent --fingerprint
(SHA256) B3:93:36:F7:DD:EF:0D:CD:DA:64:A1:0F:2D:51:58:7D:24:60:7C:CC:39:BF:EC:D7:F9:89:46:59:58:90:75:AB
root@ip-172-31-47-74:~#
```

Puppet gathers facts about each of its nodes with a tool called `facter`. `Facter`, by default, gathers information that is useful for system configuration (e.g. OS names, hostnames, IP addresses, SSH keys, and more). It is possible to add custom facts if you need other facts to perform your configurations.

To see a list of facts that are automatically being gathered on your agent node, run the following command:

```
# facter
```

```
root@ip-172-31-30-199:~# facter
architecture => amd64
augeasversion => 1.2.0
bios_release_date => 05/12/2016
bios_vendor => Xen
bios_version => 4.2.amazon
blockdevice_xvda_size => 8589934592
blockdevices => xvda
domain => us-west-2.compute.internal
ec2_ami_id => ami-d732f0b7
ec2_ami_launch_index => 0
ec2_ami_manifest_path => (unknown)
ec2_block_device_mapping_ami => /dev/sda1
ec2_block_device_mapping_root => /dev/sda1
ec2_hostname => ip-172-31-30-199.us-west-2.compute.internal
ec2_instance_action => none
ec2_instance_id => i-0a2634cb90230ea55
ec2_instance_type => t2.micro
ec2_local_hostname => ip-172-31-30-199.us-west-2.compute.internal
ec2_local_ipv4 => 172.31.30.199
ec2_mac => 02:61:34:48:d3:4d
ec2_metadata => {"ami-id"=>"ami-d732f0b7", "ami-launch-index"=>"0", "a
```

How The Main Manifest Is Executed:

The puppet agent periodically checks in with the puppet master. During this time, it will send facts about itself to the master, and pull a current catalog--a compiled list of resources and their desired states that are relevant to the agent, determined by the main manifest. The agent node will then attempt to make the appropriate changes to achieve its desired state. This cycle will continue as long as the Puppet master is running and communicating with the agent nodes.

Immediate Execution on a Particular Agent Node:

It is also possible to initiate the check for a particular agent node manually, by running the following command (on the agent node):

```
# puppet agent --test
```

Running this will apply the main manifest to the agent running the test. You might see output like the following:

```
root@ip-172-31-47-74:~# puppet agent --test
Info: Retrieving pluginfacts
Info: Retrieving plugin
Info: Caching catalog for ip-172-31-47-74.us-west-2.compute.internal
Info: Applying configuration version '1476874311'
Notice: Finished catalog run in 0.02 seconds
```

A Simple Manifest:

The main manifest file on the Puppet master is located at:
`/etc/puppet/manifests/site.pp`

On the master, edit it now:

```
# sudo vi /etc/puppet/manifests/site.pp
```

Now add the following lines to describe a file resource:

```
file {'/tmp/example-ip':                                # resource type file and filename
  ensure => present,                                     # make sure it exists
  mode   => 0644,                                         # file permissions
  content => "Here is my Public IP Address: ${ipaddress_eth0}.\n", # note the ipaddress_eth0 fact
}
```

Now save and exit.

The inline comments should explain the resource that we are defining. In plain English, this will make ensure that all agent nodes will have a file at `/tmp/example-ip`, with `-rw-r--r--` permissions, and text that contains the node's public IP address.

You can either wait until the agent checks in with the master automatically, or you can run the `puppet agent --test` command (from one of your agent nodes). Then run the following command to print the file:

```
# cat /tmp/example-ip
```

```
root@ip-172-31-47-74:~# puppet agent --test
Info: Retrieving pluginfacts
Info: Retrieving plugin
Info: Caching catalog for ip-172-31-47-74.us-west-2.compute.internal
Info: Applying configuration version '1476875101'
Notice: /Stage[main]/Main/File[/tmp/example-ip]/ensure: created
Notice: Finished catalog run in 0.02 seconds
root@ip-172-31-47-74:~#
```

We can see the output:

Here is my Public IP Address: 128.131.192.11.