

# Installation & Configuration of Puppet

I used 2 machines in which I installed the Puppet. One is used as the client and the other as the master (or server).

**The server** is an Ubuntu 12.04 LTS Server with the ip 192.168.2.112, and the following /etc/hosts file:

```
=====
127.0.0.1    localhost
127.0.1.1    leon-server
192.168.2.112 leon-server.localdomain leon-server    <- puppet-master
192.168.2.113 debian-server.myplace                <- puppet-client
```

# The following lines are desirable for IPv6 capable hosts

```
::1        ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

**The client** is a Debian 6 Server with the ip 192.168.2.113 and the following /etc/hosts file:

```
=====
127.0.0.1    localhost
127.0.1.1    debian-server.localdomain debian-server
192.168.2.113 debian-server.localdomain debian-server    <- puppet-client
192.168.2.112 leon-server.localdomain leon-server        <- puppet-master
```

# The following lines are desirable for IPv6 capable hosts

```
::1        ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

- \* Before we continue with the installation of the Puppet we have to be sure that the machines can reach and communicate each other. For this reason we use the `ping` command to send icmp requests from one machine to an other. Also it is important the two machines to have the same date. We can check it running the `date` command in each machine and compare the results. We can set a new date running for example `date -s "17 MAY 2013 11:15:00"`

I installed the **puppet-master** to the Ubuntu server: `sudo apt-get install puppetmaster`  
I installed the **puppet-client** to the Debian server: `sudo apt-get install puppet`

In each machine there is the `/etc/puppet/puppet.conf` file. I edit this file and **in the master** I added at the end the line: `dns_alt_names = puppet, leon-server.localdomain, leon-server`  
(`dns_alt_names`: A list of valid hostnames for the master, which will be embedded in its certificate.)

In the same file at the Debian machine ( **client** ) I added on the [main] block the lines:  
`server=leon-server.localdomain` <- The hostname of my puppet master-server  
`report=true`  
`pluginsync=true`

*\*\* We can use also the `certname` parameter (The sitewide unique identifier for this node. Defaults to the node's fully qualified domain name, which is usually fine.)*  
And after that I changed the value of `START` to yes at the `/etc/default/puppet` file.

The next step was to restart the puppet-master and puppet-client services:  
`sudo service puppet restart`  
`sudo service puppetmaster restart`

The two machines use the SSL protocol to communicate each other and for this reason I had to request a certificate from the master to the client and at the same time to sign this certificate. So I was connected to the puppet-client and with the  
`puppet agent --server leon-server.localdomain --waitforcert 60 --test --group 0` command the client asked for a certificate from the server.

Then I connected to the puppet-master (without to stop the previous command) and using the command `puppet cert --list` I had the list with the certificate requests. The certification was signed using the command `puppet cert --sign debian-server.localdomain`.

Looking at the puppet-client we will see something like:

```
info: Creating a new SSL key for debian-server.localdomain
info: Caching certificate for ca
info: Creating a new SSL certificate request for debian-server.localdomain
info: Certificate Request fingerprint (md5): 72:39:2F:EE:8D:05:9D:CA:21:96:3A:98:87:49:45:B4
info: Caching certificate for debian-server.localdomain
info: Retrieving plugin
info: Caching certificate_revocation_list for ca
puppet://leon-server.localdomain/plugins
info: Creating state file /var/lib/puppet/state/state.yaml
info: Caching catalog for debian-server.localdomain
info: Applying configuration version '1368732656'
notice: Finished catalog run in 0.02 seconds
```

This means that the client's certification was signed and saved on the master

Just to test...

**In the master**, I created two folders at the `etc/puppet/module` directory

`mkdir -p helloworld/manifests/`

and inside the manifests folder I created the `init.pp` file (which is a kind of module):

```
class helloworld {
    #for puppet class is a block of commands and it has nothing to do
    #with OOP
    file { ['/tmp/helloFromMaster':
        #create the file /tmp/helloFromMaster
        content => "See you at crowdpark!" #which contains something
    ]
}
```

at the `/etc/puppet/manifests/` I created the `site.pp` file which contains only the line:

`include helloworld` #the name of the class we want to use/execute at the client

After that I was connected to the client and I searched the `helloFromMaster` file

`ls -l /tmp/`

And I checked the content of the file `cat /tmp/helloFromMaster` ....

So, this is the main idea how puppet works. We can write manifest-modules at the puppet-master in order for example to install and configure web-servers or databases at the puppet-client.

## More reading...

<http://terokarvinen.com/2012/puppetmaster-on-ubuntu-12-04>

<http://www.unixmen.com/install-puppet-master-and-client-in-ubuntu/>

<https://help.ubuntu.com/12.04/serverguide/puppet.html>

<http://docs.puppetlabs.com/>

<http://docs.puppetlabs.com/guides/installation.html#post-install>

<http://honglus.blogspot.de/2012/01/force-puppet-agent-to-regenerate.html>

Troubleshooting...

[http://terminalinflection.com/puppet-client-error-ssl\\_connect-certificate-verify-failed/](http://terminalinflection.com/puppet-client-error-ssl_connect-certificate-verify-failed/)

<http://blog.adityapatawari.com/2012/02/puppet-and-common-errors.html>

[http://docs.puppetlabs.com/pe/2.0/maint\\_common\\_config\\_errors.html](http://docs.puppetlabs.com/pe/2.0/maint_common_config_errors.html)

<http://bitcube.co.uk/content/puppet-errors-explained>

Core types cheat sheet:

[http://docs.puppetlabs.com/puppet\\_core\\_types\\_cheatsheet.pdf](http://docs.puppetlabs.com/puppet_core_types_cheatsheet.pdf)

Type references:

<http://docs.puppetlabs.com/references/stable/type.html>

Module cheat sheet:

[http://docs.puppetlabs.com/module\\_cheat\\_sheet.pdf](http://docs.puppetlabs.com/module_cheat_sheet.pdf)