

Puppet [1] : Déploiement et automatisation avec Puppet

Puppet est l'outil de gestion de configuration le plus connu, il a vu le jour en 2005, bien avant ses principaux concurrents :

2009 pour Chef,
2011 pour Salt,
2012 pour Ansible

Il propose depuis maintenant longtemps une solution open-source en parallèle de la version entreprise (qui est évidemment payante) et il a eu le temps de grandir pour atteindre une taille plus que respectable.

Avant de continuer, n'oubliez pas que :

- il y a une seule chose importante à retenir pour le moment : la documentation est sur docs.puppet.com.

- mon serveur "master" se nomme `master.virtus.eu`
-
- mon serveur "slave1" ou "node1" s'appelle `slave1.virtus.eu`

N'oubliez donc pas de changer les noms des serveurs avec les vôtres au fur et à mesure de cet article.

Je vous déconseille BEAUCOUP de tenter le coup avec des configurations dans `/etc/hosts`, vous avez toutes les chances que ça foutent une pagaille mémorable avec les certificats

Installation

Nous allons installer un `puppet-server`

(que l'on pourrait appeler serveur maitre), c'est lui qui s'occupera de tout orchestrer. Puis, sur chaque node, on installera `puppet-agent`, ce qui leur permettra de communiquer avec le master.

PuppetLabs met à disposition puppet-collection pour les installations, c'est un ensemble d'outils (par exemple facter, hiera, ruby, openssl et mcollective pour puppet-agent) packagés. En utilisant ceci, nous sommes donc sûr d'avoir des logiciels compatibles entre eux.

MASTER / puppetserver

Vous pouvez voir les détails sur puppet-server et sur puppet_collections, nous allons faire ici une installation la plus simple possible (ah ah) sur une debian 8 (Jessie).

```
root@master:~# wget https://apt.puppetlabs.com/puppetlabs-release-pc1-jessie.deb
root@master:~# dpkg -i puppetlabs-release-pc1-jessie.deb
root@master:~# apt-get update
root@master:~# apt-get install puppetserver
```

Puis on démarre le service (si ça plante, allez voir la section "*memory allocation*") :

```
root@master:~# service puppetserver start
```

Attention ! `dpkg -l | grep puppet` nous donne les résultats suivants :

```
puppet-agent          1.9.1-1jessie
puppetlabs-release-pc1 1.1.0-4jessie
puppetserver          2.7.2-1puppetlabs1
```

On pourrait donc croire que l'on se retrouve avec un puppet v2.7 (on en est à la v4)... mais point du tout : Puppet Server 2.x supporte Puppet 4, alors que Puppet Server 1.x support Puppet 3.x et supérieur.

SLAVE / puppet-agent (node)

Vous pouvez voir les détails sur puppet-agent, l'installation se passe aussi sur une Debian 8 (Jessie):

```
root@slave1:~# wget https://apt.puppetlabs.com/puppetlabs-release-pc1-jessie.deb
root@slave1:~# dpkg -i puppetlabs-release-pc1-jessie.deb
root@slave1:~# apt-get update
root@slave1:~# apt-get install puppet-agent
```

Un peu de configuration

Voici une bonne chose de faite. En lisant la doc vous verrez que les exécutables de puppet sont maintenant dans `/opt/puppetlabs/bin/` (ce qui n'est pas dans notre PATH par défaut).

Pour régler ce problème pour le moment :

```
root@slave1:~# PATH=/opt/puppetlabs/bin:$PATH
```

puppet.conf

Pour les détails sur le fichier `puppet.conf`, voir `config_file_main.html`

- exemple de configuration pour un agent
- exemple de configuration pour un master

Par défaut, un node recherche son master sur un serveur qui s'appelle "puppet". Nous allons donc ajouter deux lignes pour spécifier notre configuration dans `/etc/puppetlabs/puppet/puppet.conf` sur le slave pour y ajouter ces deux lignes :

- **sur le MASTER**

```
[main]
certname = master.virtus.eu
server   = puppetmaster

[master]
dns_alt_names = puppet,puppetmaster,master.virtus.eu
vardir         = /opt/puppetlabs/server/data/puppetserver
logdir         = /var/log/puppetlabs/puppetserver
rundir         = /var/run/puppetlabs/puppetserver
pidfile        = /var/run/puppetlabs/puppetserver/puppetserver.pid
codedir        = /etc/puppetlabs/code
```

- **sur le SLAVE**

```
[main]
certname      = slave1.virtus.eu
server        = master.virtus.eu
environment   = production
```

Ajouter un node au master

Vous êtes prêt-e-s ? Il existe 2 méthodes pour dire à un node de se connecter au master, la première en "tâche de fond", et la seconde "à la main" que l'on peut mettre en cron.

sur le SLAVE

Utilisons la commande manuelle pour voir ce qui se passe :

```
root@slave1:~# /opt/puppetlabs/bin/puppet agent --test

Info: Creating a new SSL key for slave1.virtus.eu
Info: Caching certificate for ca
Info: csr_attributes file loading from /etc/puppetlabs/puppet/csr_attributes.yaml
Info: Creating a new SSL certificate request for slave1.virtus.eu
Info: Certificate Request fingerprint (SHA256):
55:A1:EE:A6:1D:D2:85:9D:7F:5F:69:72:7E:BF:54:BA:74:BD:7F:8E:43:0D:AC:A1:D1:1B:4A:FF:4A:F6:54:3B
Info: Caching certificate for ca
Exiting; no certificate found and waitforcert is disabled
```

Ce qui veut dire :

1. slave1 a créé un certificat pour lui et il arrive à se connecter au master
2. le master ne veut pas de lui vu qu'il ne le connaît pas, nous allons donc régler ça.

Sur le MASTER

```
root@master:~# /opt/puppetlabs/bin/puppet cert list
"slave1.virtus.eu" (SHA256)
55:A1:EE:A6:1D:D2:85:9D:7F:5F:69:72:7E:BF:54:BA:74:BD:7F:8E:43:0D:AC:A1:D1:1B:4A:FF:4A:F6:54:3B
```

Bonne nouvelle, le premier contact a vraiment eu lieu ! Le MASTER a bien reçu le certificat de SLAVE, et il nous attend pour le valider avec la commande suivante :

```
/opt/puppetlabs/bin/puppet cert sign slave1.virtus.eu

Signing Certificate Request for:
"slave1.virtus.eu" (SHA256)
55:A1:EE:A6:1D:D2:85:9D:7F:5F:69:72:7E:BF:54:BA:74:BD:7F:8E:43:0D:AC:A1:D1:1B:4A:FF:4A:F6:54:3B
Notice: Signed certificate request for slave1.virtus.eu
Notice: Removing file Puppet::SSL::CertificateRequest slave1.virtus.eu at
'/etc/puppetlabs/puppet/ssl/ca/requests/slave1.virtus.eu.pem'
```

Et si on relance notre `puppet agent -t` sur le slave :

```
root@slave1:~# /opt/puppetlabs/bin/puppet agent --test

Info: Caching certificate for slave1.virtus.eu
Info: Caching certificate_revocation_list for ca
Info: Caching certificate for slave1.virtus.eu
Info: Using configured environment 'production'
Info: Retrieving pluginfacts
Info: Retrieving plugin
Info: Caching catalog for slave1.virtus.eu
Info: Applying configuration version '1486920585'
Info: Creating state file /opt/puppetlabs/puppet/cache/state/state.yaml

Notice: Applied catalog in 0.07 seconds
```

Troubleshooting (install)

Vous avez peut être rencontré des problèmes au cours de cette installation, je vous conseille de regarder les points suivants pour vous aider à trouver une solution :

- `dns_alt_names` : liste de noms que votre **master** peut utiliser? Le nom que vos nodes utilisent (puppet, puppetmaster, master.virtus.eu...) **DOIT** etre inclus dedans ainsi que dans son certificat (voir aussi `dnsaltnames`),
- `puppet.conf`
- Configuration: Short list of important settings
- puppet troubleshooting ou l'original (bon courage)

Premier manifest

Personnellement, j'aime bien avoir mes outils sur mes serveurs, ainsi que quelques raccourcis, commençons donc par ça :

sur le MASTER

```
cd /etc/puppetlabs/code/environments/production/manifests
```

- Nous créons le fichier `site.pp` pour déclarer notre node et ce qu'il devra utiliser :

```
# /etc/puppetlabs/puppet/manifests/site.pp
node 'slave1.virtus.eu' {
    include common
}
```

- et nous allons faire notre premier manifest pour installer des paquets dans le fichier `common.pp` que nous créons aussi :

```
class common {  
  
  $packages_list = [  
    'apt-transport-https',  
    'bzip2',  
    'curl',  
    'deborphan',  
    'htop',  
    'less',  
    'lsof',  
    'ncdu',  
    'pbzip2',  
    'pigz',  
    'pwgen',  
    'rpl',  
    'screen',  
    'strace',  
    'sudo',  
    'tar',  
    'unzip',  
  ]  
}
```

```
'vim',  
'wget',  
'whois',  
'zip'  
]  
  
package {  
  $packages_list:  
    ensure => 'installed'  
}  
}
```

Puis nous relançons notre commande `/opt/puppetlabs/bin/puppet agent -t` sur notre slave1. Si vous n'avez pas d'erreurs, vous devriez voir quelque chose comme ceci qui s'affiche :

```
Info: Using configured environment 'production'
Info: Retrieving pluginfacts
Info: Retrieving plugin
Info: Caching catalog for slave1.virtus.eu
Info: Applying configuration version '1486926845'
Notice: /Stage[main]/Common/Package[deborphan]/ensure: created
Notice: /Stage[main]/Common/Package[ncdu]/ensure: created
Notice: /Stage[main]/Common/Package[pbzip2]/ensure: created
Notice: /Stage[main]/Common/Package[pigz]/ensure: created
Notice: /Stage[main]/Common/Package[pwgen]/ensure: created
Notice: /Stage[main]/Common/Package[rpl]/ensure: created
Notice: /Stage[main]/Common/Package[strace]/ensure: created
Notice: /Stage[main]/Common/Package[unzip]/ensure: created
Notice: /Stage[main]/Common/Package[whois]/ensure: created
Notice: /Stage[main]/Common/Package[zip]/ensure: created
Notice: Applied catalog in 39.43 seconds
```


Et voilà pour aujourd'hui !

À noter que :

- Vous pouvez utiliser la commande `puppet parser validate common.pp` si vous voulez voir si vous avez fait des erreurs de syntaxe.
- Vous pouvez utiliser puppet-lint pour voir si vous avez fait des erreurs de style.