

CHEF: Mise en oeuvre de Chef Workstation sur centos

Table of Contents

- [CHEF: Mise en oeuvre de Chef Workstation sur centos](#)
- [Installation de l'hyperviseur](#)
- [Installation de ruby](#)
- [Installation de vagrant](#)
- [Installation et configuration de chef-workstation](#)
 - [Installations des paquets](#)
 - [Télécharger et installer chef-workstation](#)
 - [Installer les modules complémentaires](#)
 - [Configuration du poste de travail](#)
 - [Vérifier l'installation à l'aide du chef verify.](#)
 - [Obtenir le kit de démarrage.](#)
 - [Configuration de Knife](#)
- [Gestion des nodes.](#)
 - [Bootstrapping du noeud](#)
 - [Gestion des cookbooks](#)
 - [Uploader un cookbook sur le serveur.](#)
 - [Assigner le cookbook](#)

Chef-workstation est un ordinateur exécutant le kit Chef Development Kit (Chef DK) utilisé pour créer des livres de recettes, interagir avec le serveur Chef et interagir avec les nœuds.

Le poste de travail est l'endroit où les utilisateurs font la plupart de leur travail :

- Développer et tester des livres de recettes et des recettes
- Test du code du chef
- Garder le référentiel Chef synchronisé avec le contrôle de la source de la version
- Configuration de la stratégie organisationnelle, y compris la définition des rôles et des environnements, et vérification de la conservation des données critiques dans des sacs de données
- Interagir avec les nœuds, selon les besoins (tels que l'exécution d'une opération d'amorçage)
- Installer chef en fonction de la destination du nœud

Installation de l'hyperviseur

Appliquer le [cookbook d'installation de l'hyperviseur KVM](#)

Installation de ruby

Appliquer le [cookbook d'installation de ruby](#)

Installation de vagrant

Appliquer le [cookbook d'installation de vagrant](#)

Installation et configuration de chef-workstation

Installations des paquets

Chef-Workstation installe tout qui est nécessaire pour utiliser Chef.

Télécharger et installer chef-workstation

```
wget
http://packages.chef.io/files/stable/chef-workstation/0.1.120/el/6/chef-workstation-0.1.120-1.el6.x86_64.rpm
rpm -ivh chef-workstation-0.1.120-1.el6.x86_64.rpm
```

Installer les modules complémentaires

L'installation doit se faire dans le contexte ruby embarqué dans la distribution

```
eval "$(chef shell-init bash)"
gem install knife-kvm
```



Knife-kvm fournit une suite de plugin pour créer / bootstraper et détruire les vm invités kvm.

Configuration du poste de travail

Vérifier l'installation à l'aide du chef verify.

```
chef verify
  Running verification for component 'berkshelf'
  Running verification for component 'test-kitchen'
  Running verification for component 'tk-policyfile-provisioner'
  Running verification for component 'chef-client'
  Running verification for component 'chef-dk'
  Running verification for component 'chef-provisioning'
  Running verification for component 'chefspect'
  Running verification for component 'generated-cookbooks-pass-chefspect'
  Running verification for component 'rubocop'
  Running verification for component 'fauxhai'
  Running verification for component 'knife-spork'
  Running verification for component 'kitchen-vagrant'
  Running verification for component 'package installation'
  Running verification for component 'openssl'
  Running verification for component 'inspec'
  Running verification for component 'delivery-cli'
  Running verification for component 'git'
  Running verification for component 'opscode-pushy-client'
  Running verification for component 'chef-sugar'
  Running verification for component 'knife-supermarket'
  .....
  -----
  Verification of component 'fauxhai' succeeded.
  Verification of component 'kitchen-vagrant' succeeded.
  Verification of component 'openssl' succeeded.
  Verification of component 'delivery-cli' succeeded.
  Verification of component 'test-kitchen' succeeded.
  Verification of component 'rubocop' succeeded.
  Verification of component 'inspec' succeeded.
  Verification of component 'opscode-pushy-client' succeeded.
  Verification of component 'knife-supermarket' succeeded.
  Verification of component 'berkshelf' succeeded.
  Verification of component 'knife-spork' succeeded.
  Verification of component 'git' succeeded.
  Verification of component 'tk-policyfile-provisioner' succeeded.
  Verification of component 'chefspect' succeeded.
  Verification of component 'chef-sugar' succeeded.
  Verification of component 'chef-client' succeeded.
  Verification of component 'chef-dk' succeeded.
  Verification of component 'package installation' succeeded.
  Verification of component 'chef-provisioning' succeeded.
  Verification of component 'generated-cookbooks-pass-chefspect' succeeded.
```

Obtenir le kit de démarrage.

Aller au **chef-serveur**> **Onglet Administration**> Sélectionner une organisation> Cliquer sur **Kit de démarrage** dans le volet de gauche.

- Onglet Admin
- Sélectionnez le kit de démarrage
- Réinitialiser les clés et télécharger

Télécharger le kit de démarrage (chef-repo) à partir du serveur et le décompresser dans le répertoire personnel sur WORKSTATION NODE.

```
unzip chef-starter.zip
tree chef-repo

chef-repo
├── cookbooks
│   ├── cheignore
│   └── starter
│       ├── attributes
│       │   └── default.rb
│       ├── files
│       │   └── default
│       │       └── sample.txt
│       ├── metadata.rb
│       ├── recipes
│       │   └── default.rb
│       └── templates
│           └── default
│               └── sample.erb
├── README.md
├── roles
│   └── starter.rb
└── 9 directories, 8 files
```

Configuration de Knife

Initialiser knife

```
knife configure
  Please enter the chef server URL:
[https://M013E13P-1249185.dgfip/organizations/myorg]
https://xx.xx.xxx.xxx:10443/organizations/dgfip/
  Please enter an existing username or clientname for the API:
[jacques.nougat] jacques
  Overwrite /home/user/jacques.nougat/.chef/credentials?? (Y/N) y
*****
  You must place your client key in:
```

```
/home/user/jacques.nougat/.chef/jacques.pem
Before running commands with Knife
*****
Configuration file written to
```

Ajuster le fichier knife.rb

```
vi ~/.chef/knife.rb

# See https://docs.getchef.com/config_rb_knife.html for more information
on knife configuration options
current_dir = File.dirname(__FILE__)
log_level          :info
log_location       STDOUT
node_name          "jacques"
client_key         "/etc/chef/jacques.pem"
chef_server_url    "https://xx.xx.xxx.xxx:10443/organizations/dgfip"
cookbook_path      ["#{current_dir}/../cookbooks"]
chef_repo_path     "/path/to/chef-repo"
```

Charger la clef privée de l'utilisateur

Charger la clef privé du compte utilisé pour la connexion au serveur chef dans le répertoire indiqué dans client_key

```
scp user@xx.xx.xxx.xxx:/chef-repo/.chef/jacques.pem ~/.chef/jacques.pem

user@xx.xx.xxx.xxx's password:
jacques.pem                  100% 1678      1.6KB/s
00:00
```

Récupérer la clé SSL du chef-serveur.

```
export no_proxy='xx.xx.xxx.xxx'
knife ssl fetch

WARNING: Certificates from xx.xx.xxx.xxx will be fetched and placed in
your trusted_cert
directory (/home/user/jacques.nougat/.chef/trusted_certs).
Knife has no means to verify these are the correct certificates. You
should
verify the authenticity of these certificates after downloading.
Adding certificate for xx_xx_xxx_xxx in
/home/user/jacques.nougat/.chef/trusted_certs/xx_xx_xxx_xxx.cr
```

Vérifier la configuration de knife

Lister les clients actuellement disponibles. Actuellement, nous ne voyons que le poste de travail.

```
knife client list

dgfip-validator
```

Gestion des nodes.

Bootstrapping du noeud

Le Bootstrapping consiste à configurer un Node de façon à ce que le Chef-server puisse le piloter.

Tout d'abord, vérifier l'accès ssh sur la machine, avec l'utilisateur root ou un utilisateur qui peut passer root avec sudo (vagrant par ex).

Ensuite lancer la commande suivante :

```
knife bootstrap [adresse-ip-du serveur] -x [username] -P [password]
```

ou s'il faut utiliser sudo : `knife bootstrap [adresse-ip-du serveur] -x [utilisateur] -P [password] --sudo`

```
knife bootstrap 192.168.121.139 -x root -P master -N dgfip-base
```

On doit avoir un retour similaire à ceci :

```
knife bootstrap 192.168.121.139 -x root -P master -N dgfip-base
  Creating new client for dgfip-base
  Creating new node for dgfip-base
  Connecting to 192.168.121.139
192.168.121.139 -----> Existing Chef installation detected
192.168.121.139 Starting the first Chef Client run...
192.168.121.139 Starting Chef Client, version 12.0.3
192.168.121.139 resolving cookbooks for run list: []
192.168.121.139 Synchronizing Cookbooks:
192.168.121.139 Compiling Cookbooks...
192.168.121.139 [2018-05-17T14:23:36+02:00] WARN: Node dgfip-base has an
empty run list.
192.168.121.139 Converging 0 resources
192.168.121.139
192.168.121.139 Running handlers:
```

```
192.168.121.139 Running handlers complete
192.168.121.139 Chef Client finished, 0/0 resources updated in
6.994744495 seconds
```

Vérifier le noeud en utilisant la commande ci-dessous.

```
knife client list
dgfip-base
dgfip-validator
```

Gestion des cookbooks

Uploader un cookbook sur le serveur.

```
knife upload cookbooks/nom_cookbook

Created cookbooks/starter
```

Assigner le cookbook

Sélectionner le cookbook et ajouter-le à la liste des courses en cours

Synchroniser le client avec le serveur en utilisant la commande chef-client.

Une fois assignés, nous pouvons nous connecter au nœud et exécuter chef-client. Cela communiquera au chef-server et obtiendra le cookbook et synchronisera le serveur.

```
chef-client
```

From:

<http://www.ouarte.garden/> - **dwndoc**

Permanent link:

<http://www.ouarte.garden/doku.php?id=vagrant:chef-workstation>

Last update: **2025/02/19 10:59**

