CHEF: Mise en oeuvre de chef server

Table of Contents

Le serveur Chef agit comme un concentrateur pour les données de configuration.

Le serveur Chef stocke les livres de recettes (cookbooks), les stratégies appliquées aux nœuds et les métadonnées qui décrivent chaque nœud enregistré géré par le chef-client.

Les nœuds utilisent le chef-client pour demander au serveur Chef des détails de configuration, tels

que des recettes, des modèles et des distributions de fichiers.

Le chef-client fait alors le plus de travail de configuration possible sur les nœuds eux-mêmes (et non sur le serveur Chef).

Cette approche évolutive répartit l'effort de configuration dans toute l'organisation.

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 paramétrage de chef-server

Prérequis

Configurer le fichier /etc/hosts .

Pour résoudre en local les différents

```
127.0.0.1 localhost
xx.xx.xxx.xxx chefserver.dgfip chefserver
xx.xx.xxx.xx chefworkstation.dgfip chefworkstation
```

Installer crontabs (s'il n'est pas déjà installé)

```
 yum install crontabs
```

Vider toutes les règles iptables et les sauvegarder sur tous les nœuds.

```
$ iptables -F
$ service iptables save
```

```
  iptables: Saving firewall rules to /etc/sysconfig/iptables:[  OK  ]
```

```
  $ iptables -L
```

```
  Chain INPUT (policy ACCEPT)
  target     prot opt source               destination
  Chain FORWARD (policy ACCEPT)
  target     prot opt source               destination
  Chain OUTPUT (policy ACCEPT)
  target     prot opt source               destination
  [root@chefnode Downloads]#
```

Définir selinux sur le mode Permissive sur tous les nœuds.

```
$ setenforce Permissive
$ getenforce
```

```
  Permissive
```

Installer le serveur Chef

Télécharger et installer chefdk

```
$ wget
https://packages.chef.io/repos/yum/stable/el/6/x86_64/chefdk-2.5.3-1.el6.x86
_64.rpm
$ rpm -ivh chefdk-2.5.3-1.el6.x86_64.rpm
```

Télécharger et installer chef-server-core

```
$ wget
https://packages.chef.io/repos/yum/stable/el/6/x86_64/chef-server-core-12.17
.33-1.el6.x86_64.rpm
```

```
$ rpm -ivh chef-server-core-12.6.0-1.el6.x86_64.rpm
```

Configurer le serveur

Reconfigurer le serveur.

```
$ chef-server-ctl reconfigure
```

Problème : common_name is required

```
00:00] ERROR: Exception handlers complete
Chef Client failed. 1 resources updated in 10.379691033 seconds
[2015-09-23T16:46:46+00:00] FATAL: Stacktrace dumped to
/opt/opscode/embedded/cookbooks/cache/chef-stacktrace.out
[2015-09-23T16:46:47+00:00] FATAL: Chef::Exceptions::ValidationFailed:
common_name is required
```

Solution

- S'assurer de la présence du nom d'hôte dans le fichier /etc/hosts, le nom d'hôte doit être résolu en adresse IP, cela n'a pas d'importance s'il s'agit d'un bouclage.
- Changer le nom d'hôte, sudo hostname chefserver, ici chefserver est le nouveau nom d'hôte.
- Modifier également le nom d'hôte dans le fichier /etc /sysconfig/network afin que le nom persiste après un redémarrage. Changez HOSTNAME = chefserver dans le fichier.
- Ajouter le nom d'hôte dans >> /etc/host : sudo echo “127.0.0.1 localhost chefserver” >> /etc/hosts.

Problème /etc/opscode/pivotal.pem missing

```
[2016-02-10T22: 09: 53 + 00: 00] FATAL:

------------------------------------------------- ---------------------
BOOT007: Le fichier de secrets (/etc/opscode/private-chef-secrets.json) est
présent
     mais le fichier /etc/opscode/pivotal.pem est manquant.

     S'assurer que private-chef-secrets.json est copié dans /etc/opscode
depuis le
     premier nœud Chef Server mis en ligne, puis exécuter
     'chef-server-ctl reconfigure' à nouveau.
```

——————————————————— ————————

Solution

- Renommer /etc/opscode/private-chef-secrets.json en /etc/opscode/private-chef-secrets.json.org

et essayez si le problème persiste, puis essayez l'option suivante.

— OU — (si la solution ci-dessus ne fonctionne pas) —

- Pour l'erreur /etc/opscode/pivotal.pem is missing.

Utiliser la commande ci-dessous.

```
 cp /opt/opscode/embedded/service/omnibus-ctl/spec/fixtures/pivotal.pem
/etc/opscode/
```

Créer un utilisateur pour le serveur.

```
 chef-server-ctl user-create USER_NAME FIRST_NAME LAST_NAME EMAIL 'PASSWORD'
--filename PATH_TO_FILE_NAME
```

Commande.

```
 chef-server-ctl user-create jacques Nougat Jacques
jacques.nouga@dgfip.finances.gouv.fr 'motdepasse' --filename
/etc/chef/jacques.pem
```

Créer une organisation

```
$ chef-server-ctl org-create short_name 'full_organization_name' --
association_user user_name --filename ORGANIZATION-validator.pem
```

Commande.

```
$ chef-server-ctl org-create ahmedinc 'dgfip' --association_user jacques --
filename /etc/chef/dgfip-validator.pem
```

Télécharger et installer chef-manage

```
$ wget
https://packages.chef.io/repos/yum/stable/el/6/x86_64/chef-manage-2.5.15-1.e
l6.x86_64.rpm
$ rpm -ivh chef-manage-2.5.15-1.el6.x86_64.rpm
```

```
  $ chef-server-ctl reconfigure
  $ chef-manage-ctl reconfigure
```

Vérifier linstallation

Vérifier l'état du serveur

```
$ chef-server-ctl status
run: bookshelf: (pid 21281) 1977939s; run: log: (pid 14630) 1994976s
run: nginx: (pid 12537) 1887643s; run: log: (pid 15024) 1994972s
run: oc_bifrost: (pid 21365) 1977938s; run: log: (pid 14119) 1994981s
run: oc_id: (pid 21411) 1977938s; run: log: (pid 12420) 1887644s
run: opscode-erchef: (pid 12424) 1887643s; run: log: (pid 14796) 1994975s
run: opscode-expander: (pid 21793) 1977935s; run: log: (pid 14538) 1994977s
run: opscode-solr4: (pid 21922) 1977934s; run: log: (pid 14356) 1994978s
run: postgresql: (pid 21972) 1977933s; run: log: (pid 14062) 1994982s
run: rabbitmq: (pid 22308) 1977926s; run: log: (pid 15032) 1994972s
run: redis_lb: (pid 9486) 1887685s; run: log: (pid 15017) 1994973s
```

Connexion à Chef-manage

Pour vérifier le bon fonctionnement de l'installation.

```
 http://ip_de_votre_serveur:port
```

Créer le référentiel chef-repo

Utiliser la commande **chef generate repo chef-repo** pour créer le référentiel Chef. Par exemple, pour créer un référentiel appelé chef-repo :

```
chef generate repo chef-repo
```

Créer un répertoire .chef

Le répertoire .chef est utilisé pour stocker trois fichiers:

- config.rb
- ORGANIZATION-validator.pem
- USER.pem

Ces fichiers doivent être présents dans le répertoire .chef pour que ChefDK puisse se connecter à un serveur Chef.

Créer le répertoire .chef dans le répertoire chef-repo :

```
  mkdir -p /path/to/chef-repo/.chef
```

Une fois le répertoire .chef créé, la structure de dossiers suivante sera présente sur la machine locale:

```
chef-repo/
  .chef/ << le répertoire caché
  certificats/
  config/
  cookbooks/
  data_bags
  environnements/
  roles/
```

Note de sécurité: Ajouter .chef au fichier .gitignore pour empêcher le téléchargement du contenu du dossier .chef sur GitHub. Par exemple:

```
$ echo '.chef' >> chef-repo/.gitignore
```

Créer le répertoire .chef/client.d

À partir de Chef 12.8, on peut créer un **.chef/client.d** dont le contenu sera copié sur le système en cours d'amorçage à l'aide de la commande knife bootstrap . On peut également définir l'option client_d_dir dans config.rb pour qu'elle pointe vers un autre répertoire au lieu de **.chef/client.d**, et le contenu de ce répertoire sera copié sur le système en cours d'initialisation. Tous les fichiers de configuration du répertoire client.d sont copiés dans **/etc/chef/client.d** du système en cours d'amorçage.

```
$ mkdir -p .chef/client.d
```

La présence de ce répertoire implique le chargement automatique de son contenu lors du bootstraping, s'il est vide cela provoquera un échec de la séquence de bootsrap. Ne créer ce dossier que si on en a effectivement besoin

Charger le Kit de démarrage

Le kit de démarrage permet de créer les fichiers de configuration nécessaires chez le clients: le répertoire **.chef**, les fichiers **config.rb**, **ORGANIZATION-validator.pem** et **USER.pem**.

Pour le mettre à disposition des clients télécharger simplement le kit de démarrage puis placer-le dans le répertoire partagé client.d.

Configurer le référentiel Chef

Avec WebUI

Utiliser les étapes suivantes pour configurer manuellement le chef-repo et utiliser la console de gestion Chef pour obtenir les fichiers .pem et config.rb .

Récupérer les fichiers de configuration

Pour une installation ChefDK qui interagira avec le serveur Chef (y compris le serveur Chef hébergé), se connecter au serveur et télécharger les fichiers suivants:

- **config.rb** Ce fichier de configuration peut être téléchargé à partir de la page Organisations .
- **ORGANIZATION-validator.pem** Cette clé privée peut être téléchargée à partir de la page Organisations .
- **USER.pem** Cette clé privée peut être téléchargée à partir de la section Changer le mot de passe de la page Gestion du compte .

Déplacer les fichiers de configuration

Les config.rb , ORGANIZATION-validator.pem et USER.pem doivent être déplacés vers le répertoire .chef après leur téléchargement à partir du serveur Chef.

Pour déplacer des fichiers dans le répertoire .chef :

```
cp /path/to/config.rb /path/to/chef-repo/.chef
```

et:

```
cp /path/to/ORGANIZATION-validator.pem /path/to/chef-repo/.chef
```

et:

```
 cp /path/to/USERNAME.pem /path/to/.chef
```

Sans WebUI

Utiliser les étapes suivantes pour configurer manuellement le référentiel Chef:

- Sur le serveur Chef, créer les fichiers ORGANIZATION-validator.pem et USER.pem à l'aide de l'outil de ligne de commande chef-server-ctl
- Sur le poste de travail, créer le fichier config.rb avec l’outil knife .

Créer une organisation

Sur la machine serveur Chef, créer ORGANIZATION-validator.pem en utilisant chef-server-ctl en ligne de commande :

```
$ chef-server-ctl org-create ORG_NAME ORG_FULL_NAME -f FILE_NAME
```

- **ORG_NAME** doit commencer par une lettre minuscule ou un chiffre, ne peut contenir que des lettres minuscules, des chiffres, des traits d'union et des caractères de soulignement, et doit comporter entre 1 et 255 caractères. Par exemple: chef
- **ORG_FULL_NAME** doit commencer par un caractère non ESPACE et doit comporter entre 1 et

1023 caractères. Par exemple: “Chef Software,Inc.”
- **-f FILE_NAME** : pour écrire ORGANIZATION-validator.pem dans un fichier FILE_NAME au lieu d'utiliser la sortie standard STDOUT .

Par exemple pour une organisation nommée **chef**, portant le nom complet de **Chef Software,Inc** et le fichier **ORGANIZATION-validator.pem** enregistré dans **/tmp/chef.key** :

```
$ chef-server-ctl org-create chef "Chef Software, Inc."  -f /tmp/chef.key
```

Créer un utilisateur

Dan le serveur Chef, créer USER.pem partir de la ligne de commande en utilisant chef-server-ctl :

```
$ chef-server-ctl user-create USER_NAME FIRST_NAME LAST_NAME EMAIL PASSWORD
-f FILE_NAME

* **USER\_NAME** est le nom d'utilisateur
* **FIRST\_NAME** est son nom
* **LAST\_NAME** est son prénom
* **EMAIL** est sond adresse mail
* **PASSWORD** est sont mot de passe
* **-f FILE_NAME** écrit le fichier USER.pem dans un fichier au lieu de
STDOUT .
```

Par exemple: un utilisateur nommé **grantmc**, avec le prénom et le nom **Grant McLennan** , une adresse électronique **grantmc@chef.io**, un mot de passe **achanger** et un fichier USER.pem enregistré dans **/tmp/grantmc.key** :

```
$ chef-server-ctl utilisateur-créer grantmc Grant McLennan grantmc@chef.io
achanger -f /tmp/grantmc.key
```

Déplacer des fichiers .pem

Placer les fichiers **ORGANIZATION-validator.pem** et **USER.pem** à partir du serveur Chef dans le répertoire .chef:

```
$ cp /path/to/ORGANIZATION-validator.pem /path/to/chef-repo/.chef
```

et:

```
$ cp /path/to/USERNAME.pem /path/to/chef-repo/.chef
```

Créer le fichier config.rb

Dans le répertoire chef-repo/.chef créer le fichier config.rb à l'aide de l'outil knife configure. Le fichier doit être créé dans le dossier .chef . Il devrait ressembler à:

```
current_dir = Fichier .  dirname ( __FILE__ )
log_level : info
log_location STDOUT
node_name 'nom_noeud'
client_key "#{current_dir}/USER.pem"
validation_client_name 'ORG_NAME-validator'
validation_key "#{current_dir}/ORGANIZATION-validator.pem"
chef_server_url 'https://api.chef.io/organizations/ORG_NAME'
cache_type 'BasicFile'
cache_options ( :path => "#{ENV['HOME']}/.chef/checksums" )
cookbook_path ["#{current_dir}/../cookbooks" ]
```

Au minimum, on doit mettre à jour les paramètres suivants avec les valeurs appropriées:

- **client_key** doit pointer sur l'emplacement du fichier .pem l'utilisateur du serveur Chef
- **validation_client_name** doit être mis à jour avec le nom de l'organisation souhaitée créée sur le serveur Chef.
- **validation_key** doit pointer sur l'emplacement du fichier .pem de l'organisation sur le serveur ChefDK.
- **chef_server_url** doit être mis à jour avec le domaine ou l'adresse IP utilisé pour accéder au serveur Chef.

Récupérer les certificats SSL

Le serveur Chef 12 active la vérification SSL par défaut pour toutes les demandes adressées au serveur, telles que celles effectuées par knife et le chef-client. Le certificat généré lors de l'installation du serveur Chef est auto-signé, ce qui signifie qu'il n'y a pas d'autorité de certification de signature à vérifier. De plus, ce certificat doit être téléchargé sur toute machine à partir de laquelle knife et/ou le chef-client adressera des demandes au serveur Chef.

Utiliser la sous-commande knife ssl fetch pour extraire le certificat SSL du serveur Chef:

```
$ knife ssl fetch
```

Configuration de Knife

Pour utiliser knife depuis le serveur il faut configurer le dossier de l'utilisateur par défaut pour pouvoir s'authentifier au serveur

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:/jacques.pem /etc/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
```