Supervision d'un cluster MongoDB

Sommaire

Supervision d'un cluster MongoDB

Voici les étapes à suivre pour superviser un cluster MongoDB avec Shinken :

- Assurez-vous qu'il existe un hôte dans Shinken pour chaque nœud de votre cluster MongoDB.
  - Si ce n'est pas le cas, créez les hôtes nécessaires.
- Accrocher le modèle d'hôte **mongodb3** sur chacun des hôtes du cluster.
- Modifiez les données des hôtes si nécessaire, par exemple en changeant le port d'écoute si votre cluster MongoDB n'est pas configuré sur le port par défaut.

Shinken recommande de superviser le cluster MongoDB créé lorsque l'on souhaite assurer une haute disponibilité pour la base de données de Shinken.

La supervision du cluster MongoDB dans ce cas nécessite un paramétrage spécifique (*voir la page* Haute disponibilité de la base MongoDB (mise en place d'un cluster)) .

Supervision d'un cluster MongoDB en SSL

Il est possible d'activer le chiffrement pour les communications au sein d'un cluster MongoDB (*voir la page* Activer le chiffrement (SSL) pour les communications d'un cluster MongoDB).

- Dans ce cas, le cluster MongoDB refuse les connexions d'un serveur qui ne fournit pas un certificat qu'il reconnait.

Le modèle **mongodb3** ainsi que les machines de supervision Shinken nécessitent d'être paramétrés pour superviser un Cluster MongoDB en SSL.

Création des certificats nécessaires à la supervision d'un cluster MongoDB en SSL

Pour superviser un cluster MongoDB en SSL, il faut que les machines qui exécutent les check de supervision du pack disposent de certificats acceptés par le cluster MongoDB.

Ces machines correspondent aux machines des **Pollers** et à la machine du **Synchronizer**.

- **Si vous disposez déjà de certificats :**
  - il faut vous assurer qu'ils ont été émis par la même autorité de certification que celle du cluster (*correspond à la clé* ***CAFile*** *dans le fichier de configuration du démon*) :

/etc/mongod.conf

```
net:
  ssl:
    mode: requireSSL
    PEMKeyFile: /etc/shinken/certs/mongodb/NODEX.pem
    CAFile: /etc/shinken/certs/mongodb/ca-cert.pem
```

  - Pour afficher les détails d'un certificat :

```
openssl x509 -in /chemin/vers/le/fichier/certificat.crt -text -noout
```

- **Si vous ne disposez pas de certificats et souhaitez en créer** :

> Les certificats, émis pour les serveurs de supervision du cluster, doivent répondre aux contraintes suivantes (*voir la page https://www.mongodb.com/docs/v3.0/core/security-x.509/#std-label-client-x509-certificates-requirements*) :
>
> - Ils doivent être émis par la même autorité de certification que celle du cluster.
> - Chaque serveur de supervision doit avoir son propre certificat.
> - parmi les attributs suivants des certificats, l'un d'entre eux doit avoir une valeur différente de celles utilisées pour le cluster :
>   - **O** (*Organization*)
>   - **OU** (*Organizational Unit*)
>   - **DC** (*Domain Components*)

  - Pour chaque serveur qui exécute les checks, lancer les commandes suivantes :
    - En remplaçant **nom_serveur** par le nom du serveur Shinken, tel qu'il est connu par le serveur DNS,
    - En remplaçant **Organization Inc.** par la valeur utilisée pour le certificat de l'autorité de certification,
    - **ca-cert.pem** et **ca-key.pem** correspondent à la clé de chiffrement et au certificat de votre autorité de certification.

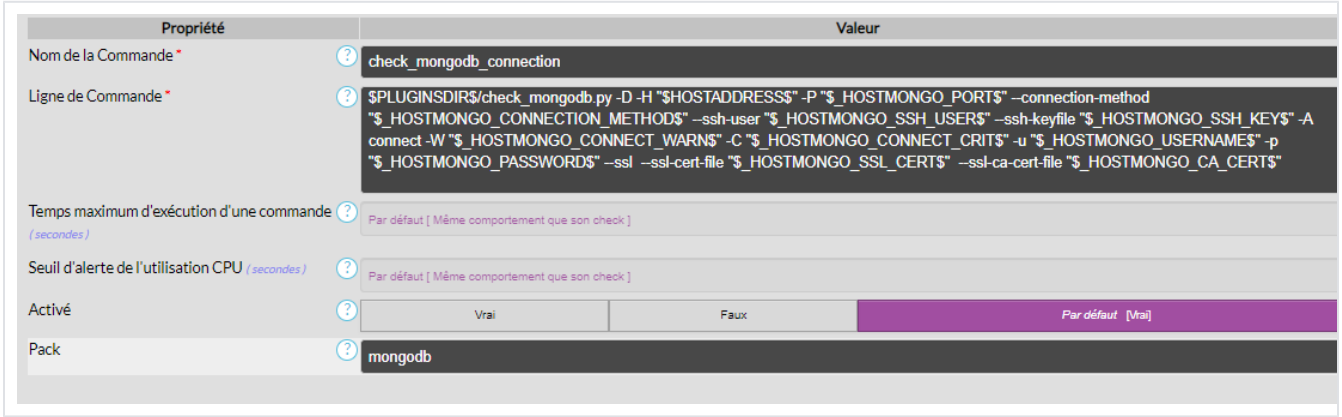
```
/opt/shinken/openssl/bin/openssl req -newkey rsa:2048 -nodes -days 365000 -keyout nom_serveur-key.pem -out
nom_serveur-req.pem -subj "/C=FR/L=Paris/O=Organization Inc./OU=Shinken MongoDB Supervision/CN=nom_serveur"
/opt/shinken/openssl/bin/openssl x509 -req -days 365000 -set_serial "0x`openssl rand -hex 8`" -in
nom_serveur-req.pem -out nom_serveur-cert.pem -CA ca-cert.pem -CAkey ca-key.pem
cat nom_serveur-key.pem nom_serveur-cert.pem > nom_serveur.pem
```

- **Sur les machines qui exécutent les check de supervision,**
  - vous devez disposer des deux certificats suivants :
    - **ca-cert.pem**, contient le certificat de l'autorité de certification, qui permet de certifier l'authenticité du cluster. Ce certificat permet de s'assurer que le cluster est bien celui qu'il prétend être,
    - **nom_serveur.pem**, contient le certificat qui certifie l'authenticité du serveur de supervision vis-à-vis du cluster MongoDB. Ce certificat permet au cluster de s'assurer que le serveur de supervision est bien celui qu'il prétend être.
  - nous conseillons de mettre vos certificats dans le dossier /etc/shinken/certs/mongodb, dans tous les cas, ils **doivent être placés au même endroit pour toutes les machines qui exécutent les checks**).

Paramétrage du modèle mongodb3

- Depuis l'interface de configuration, modifier **les commandes de tous les checks qui s'attachent au modèle d'hôte mongodb3** :
  - Ajouter à la fin de la ligne de chaque commande :

```
--ssl  --ssl-cert-file "$_HOSTMONGO_SSL_CERT$"  --ssl-ca-cert-file "$_HOSTMONGO_CA_CERT$"
```



- Sur les hôtes de chaque nœud du cluster MongoDB, ajouter les données MONGO_CA_CERT et MONGO_SSL_CERT qui pointent vers les certificats qui correspondent :

Nom	Valeur	Venant des modèles
MONGO_CA_CERT	/etc/shinken/certs/mongodb/ca-cert.pem	
MONGO_SSL_CERT	/etc/shinken/certs/mongodb/nom_serveur.pem	