

shinken-protected-fields-keyfile-export

Sommaire

- Concept
- Options
 - Options générales
 - Options concernant l'export de clé
 - Options de connexion à la base MongoDB
 - Options génériques
 - Options de connexion SSH
 - Options d'authentification
 - Options SSL/TLS
- Exemples

Concept

Cette commande permet de récupérer la clé courante utilisée pour le chiffrement des données de Shinken Enterprise.

- Il est important d'utiliser cette commande pour sauvegarder cette clef dans un endroit sécurisé.
- Si la clé sur le serveur venait à être corrompue, ou perdue, il est possible de la restaurer.

Options

Options générales

Option	Valeur par défaut	Description
-h	---	Affiche l'aide de la commande.

Options concernant l'export de clé

Option	Valeur par défaut	Description
-f <i>ARG</i>	---	Spécifier manuellement le chemin vers le fichier de la clé (si non spécifié, la commande exportera celle utilisée par le Synchronizer).

Options de connexion à la base MongoDB



Cette commande récupère les paramètres de connexion à la base MongoDB depuis la configuration.

- Il est nécessaire d'utiliser les options de la ligne de commande que si les fichiers de configuration ne correspondent pas à la base MongoDB sur laquelle, la commande doit être exécutée (

migration de base, test sur une préprod ...).

La commande dispose d'options de connexion à la base MongoDB qui peuvent être utilisés dans les cas suivants :

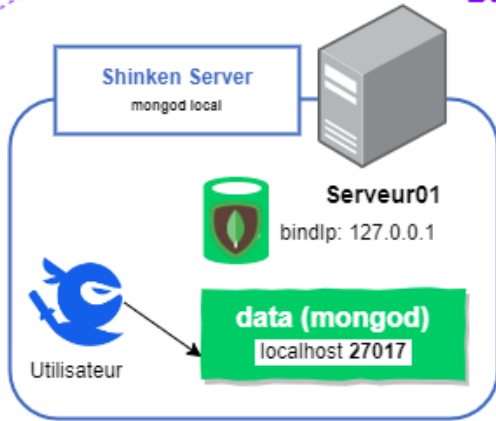
- La base de données MongoDB ne se trouve pas sur la machine qui exécute la commande.
- L'authentification par mot de passe à la base MongoDB est activée.
- Le port de MongoDB n'est pas celui par défaut (*défaut : 27017*).



La combinaison des options de connexion à MongoDB peut rapidement devenir complexe ; voici des paramètres adaptés aux cas les plus courants.

Options génériques

Base MongoDB en local

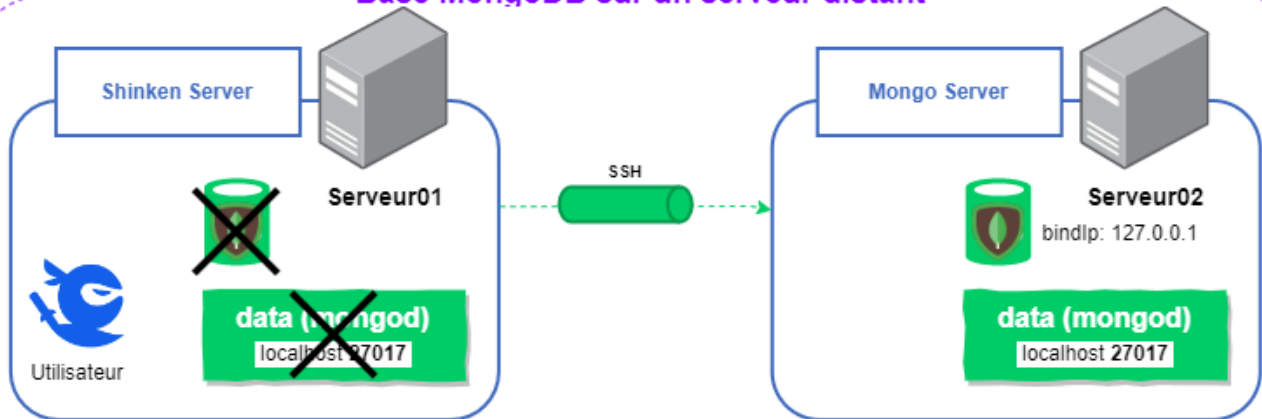


```
[root@serveur01 ~] shinken-commande --mongo-host 127.0.0.1 --mongo-port 27017 --mongo-database shinken
```

Option	Valeur par défaut	Description
--mongo-host <i>ARG</i>	localhost	Nom ou IP du serveur MongoDB.
--mongo-port <i>ARG</i>	27017	Port de la base MongoDB.
--mongo-database <i>ARG</i>	shinken (ou synchronizer si la commande concerne la base du Synchronizer)	Nom de la base de données à utiliser dans MongoDB. À n'utiliser que si la configuration du module ou du démon a changé la base utilisée par défaut.

Options de connexion SSH

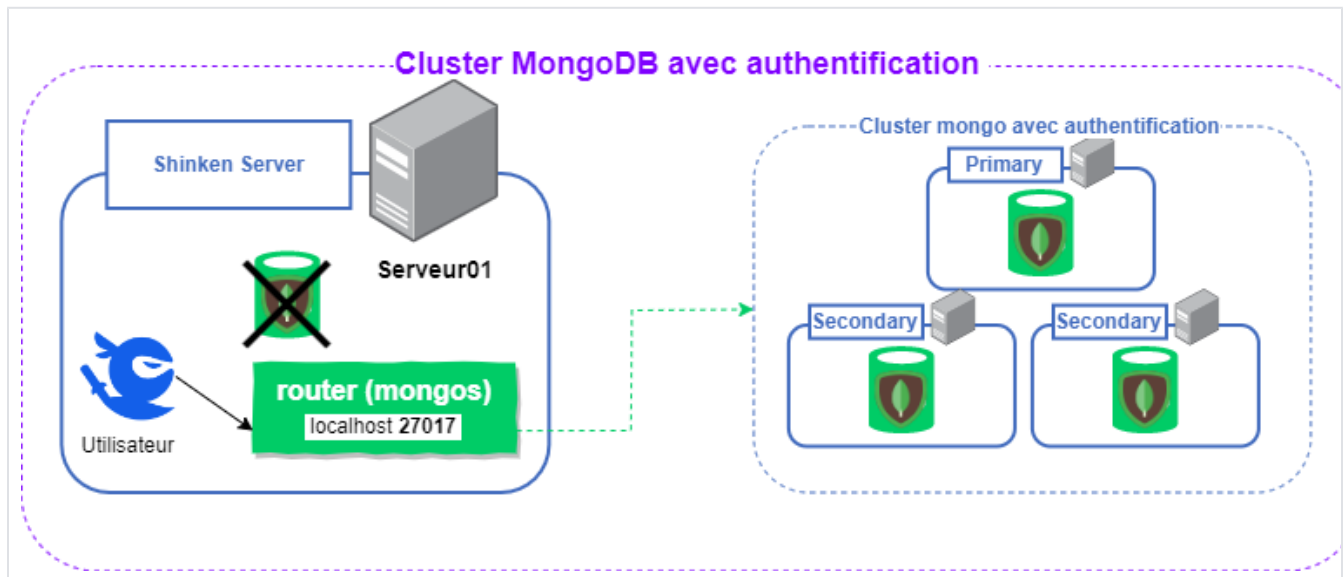
Base MongoDB sur un serveur distant



```
[root@serveur01 ~] shinken-command --mongo-host serveur02 --mongo-port 27017 --mongo-use-ssh --mongo-ssh-key /var/lib/shinken/.ssh/id_rsa --mongo-ssh-user shinken
```

Option	Valeur par défaut	Description
<code>--mongo-use-ssh</code>	---	Active la connexion SSH au serveur MongoDB.
<code>--mongo-ssh-key ARG</code>	<code>/var/lib/shinken/.ssh/id_rsa</code>	Clé privée SSH pour la connexion au serveur MongoDB. À utiliser en complément de l'option <code>--mongo-use-ssh</code> .
<code>--mongo-ssh-user ARG</code>	<code>shinken</code>	Utilisateur à utiliser pour la connexion SSH. À utiliser en complément de l'option <code>--mongo-use-ssh</code> .

Options d'authentification

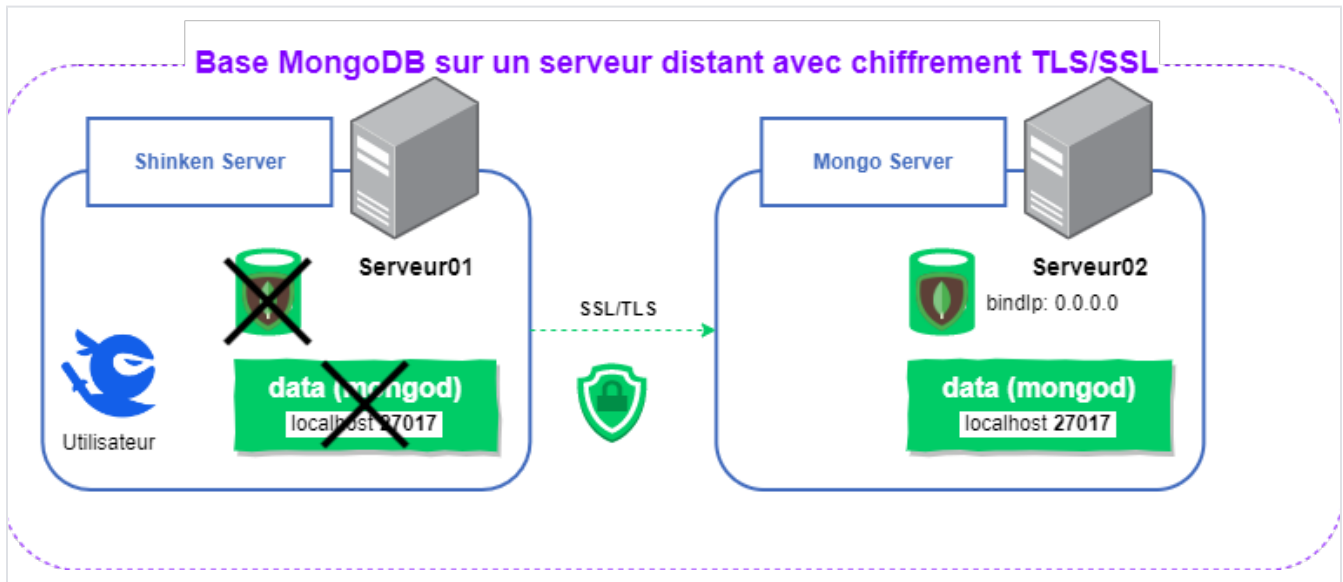


```
[root@serveur01 ~] shinken-command --mongo-host 127.0.0.1 --mongo-port 27017 --mongo-username shinken --mongo-password shinken
```

Option	Valeur par défaut	Description
<code>--mongo-username ARG</code>	---	Utilisateur pour l'authentification avec mot de passe.

<pre>-- mongo - password ARG</pre>	---	<p>Mot de passe de l'utilisateur pour l'authentification avec mot de passe.</p> <p>À utiliser en complément de l'option <code>--mongo-username</code>.</p> <div style="border: 1px solid green; padding: 10px; margin-top: 10px;"> <p>✔ Si l'option <code>--mongo-password</code> est utilisée, le mot de passe risque d'être visible dans l'historique des commandes (<i>via la commande <code>history</code></i>).</p> <p>Pour éviter d'exposer le mot de passe, il est possible d'utiliser cette commande uniquement avec l'option <code>--mongo-username</code>. Un prompt interactif apparaîtra alors pour demander le mot de passe.</p> <p>Pour automatiser les commandes dans un script, il est possible de rediriger le contenu d'un fichier contenant le mot de passe (<i>par exemple : <code>--mongo-password \$(cat my_file_with_password)</code></i>).</p> </div>
------------------------------------	-----	---

Options SSL/TLS



```
[root@serveur01 ~] shinken-command --mongo-host serveur02 --mongo-port 27017 --mongo-ssl-ca-file /etc/shinken/certs/mongo/ca.pem --mongo-ssl-pem-key-file /etc/shinken/certs/mongo/client.pem
```

Option	Valeur par défaut	Description
<code>--mongo-ssl</code>	---	Active SSL/TLS pour les communications avec la base MongoDB.
<code>--mongo-ssl-ca-file ARG</code>	---	Chemin vers le fichier de l'autorité de certification (<i>CA</i>) utilisé pour vérifier le certificat SSL de MongoDB. À utiliser en complément de l'option <code>--mongo-ssl</code> .
<code>--mongo-ssl-pem-key-file ARG</code>	---	Chemin vers le fichier contenant le certificat SSL du client. À utiliser en complément de l'option <code>--mongo-ssl</code> .
<code>--mongo-ssl-pem-key-password ARG</code>	---	Mot de passe du certificat SSL du client. À utiliser en complément de l'option <code>--mongo-ssl</code> .

<code>--mongo-ssl-crl-file ARG</code>	---	Chemin vers le fichier CRL (<i>liste de révocation</i>) des certificats SSL à rejeter. À utiliser en complément de l'option <code>--mongo-ssl</code> .
<code>--mongo-ssl-allow-invalid-hostnames</code>	---	Accepter le certificat SSL de MongoDB même si le nom d'hôte du certificat ne correspond pas à celui du serveur. À utiliser en complément de l'option <code>--mongo-ssl</code> .
<code>--mongo-ssl-allow-invalid-certificates</code>	---	Accepter le certificat SSL de MongoDB même s'il est invalide, par exemple expiré. À utiliser en complément de l'option <code>--mongo-ssl</code> .

Exemples

```
$ shinken-protected-fields-keyfile-export
Checking consistency between the synchronizer configuration file and the currently running configuration... DONE
The following line represents the protected fields cipher key named : secret key
c2VjcmV0IGtleXxWeEJnYzQzdkZlR0oraDZGR1hpVm55T1B4M2tRZW8vNWZqN2F5YkdCQ2x3PQo=
You are responsible for saving it securely in a separate place from the Shinken backup.
You can restore this key export by running the following command :
shinken-protected-fields-keyfile-restore c2VjcmV0IGtleXxWeEJnYzQzdkZlR0oraDZGR1hpVm55T1B4M2tRZW8vNWZqN2F5YkdCQ2x3PQo=
```

La commande affichera :

- le nom de la clé
- un hash de la clé de chiffrement.

Ces valeurs affichées seront nécessaires lors de la restauration de la clé.



Il est important d'utiliser cette commande pour extraire les clés car :

- Des vérifications d'intégrité sont faites.
- Une compatibilité ascendante est assurée par l'utilisation de ces commandes si le principe de chiffrement de Shinken Entreprise venait à évoluer.



Dans le but d'un haut niveau de sécurité, il est conseillé de stocker cette clé dans un emplacement différent que les sauvegardes Shinken.