shinken-protected-fields-data-manage

Sommaire

Concept

Cette commande permet de gérer la liste des motifs (*substrings*) permettant de déterminer quelles données des éléments de Shinken seront protégées.

Grâce à cette commande, il est possible de :

- Visualiser le paramétrage courant et la liste des données chiffrées correspondant aux motifs en se basant sur les éléments actuellement dans la configuration du Synchronizer.
- Ajouter un ou plusieurs motifs et prévisualisé la nouvelle liste des données chiffrées.
- Retirer un ou plusieurs motifs et prévisualisé la nouvelle liste des données chiffrées.

La commande affiche deux types d'informations principales :

- Les modifications apportées à la liste des motifs :
  - en blanc la liste de motifs actuel,
  - en vert les motifs ajoutés,
  - en rouge les motifs supprimés.
  - en jaune la nouvelle liste de motifs.
- La liste des données chiffrées en fonction de la liste des motifs :
  - en blanc la liste des données actuellement chiffrées,
  - en vert les données qui seront chiffrées,
  - en rouge les données qui ne seront plus chiffrées.

- Il est possible d'ajouter des motifs pour des données qui ne sont pas encore utilisés dans la base. Dans ce cas, la liste des donnée s ajoutées n'inclura aucune données liée à ce motifs.
- La nouvelle configuration n'est pas écrite dans le fichier de configuration du Synchronizer par défaut, mais dans le fichier de surcharge se trouvant dans l'arborescence **/etc/shinken-user/configuration/daemons/synchronizers /synchronizer_cfg_overload.cfg.** Cela permet de garder la configuration à travers les mises à jour

Options

Options générales

Option	Valeur par défaut	Description
-h, --help	--	Affiche le descriptif de la commande.
-c, --config	--	Fichier dans lequel lire la configuration actuelle (*par défaut, le fichier de configuration du Synchronizer qui prend en charge les éventuelles surcharges : /etc/shinken/synchronizer.cfg*).

--quiet	--	Réduit la quantité d'informations à afficher.

Options de gestions des données protégées

Option	Valeur par défaut	Description
-a, --add	--	Mot-clé à ajouter ; pour ajouter plusieurs mots-clés, répéter cette option autant de fois que nécessaire.
--remove	--	Mot-clé à retirer; pour retirer plusieurs mots-clés, répéter cette option autant de fois que nécessaire.

Les options **--add** et **–remove** peuvent être utilisées simultanément

Options de connexion à la base MongoDB

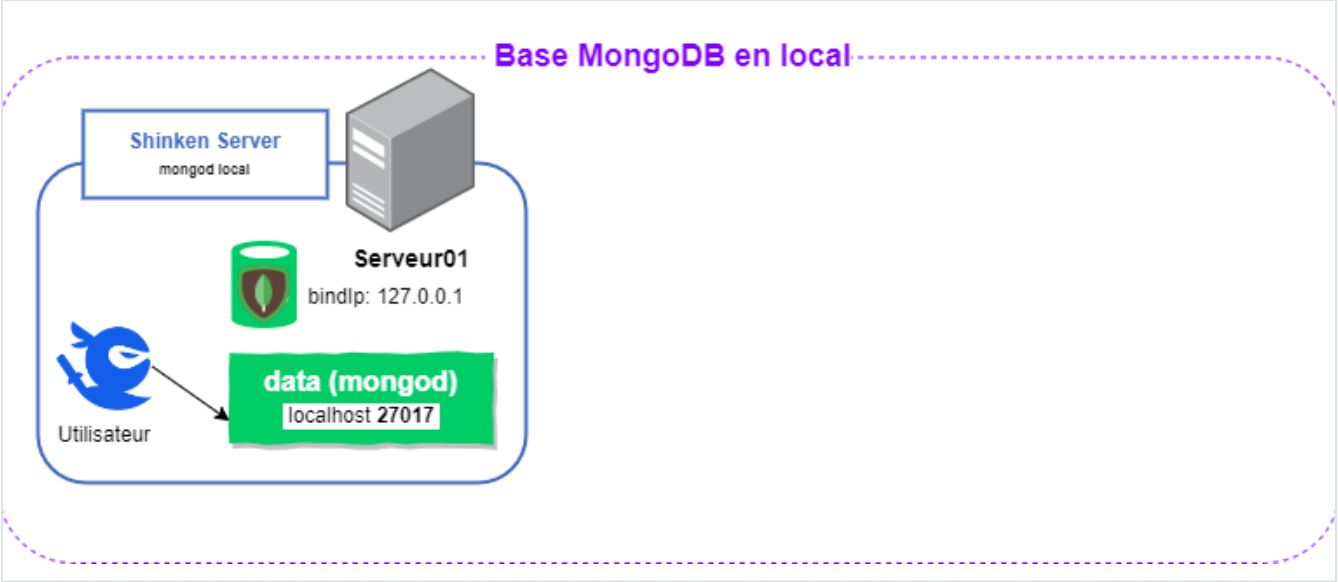
Cette commande récupère les paramètres de connexion à la base MongoDB depuis la configuration.
Il est nécessaire d'utiliser les options de la ligne de commande que si les fichiers de configuration ne correspond pas à la base MongoDB sur la quel la commande doit être exécutée (*migration de base, test 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.

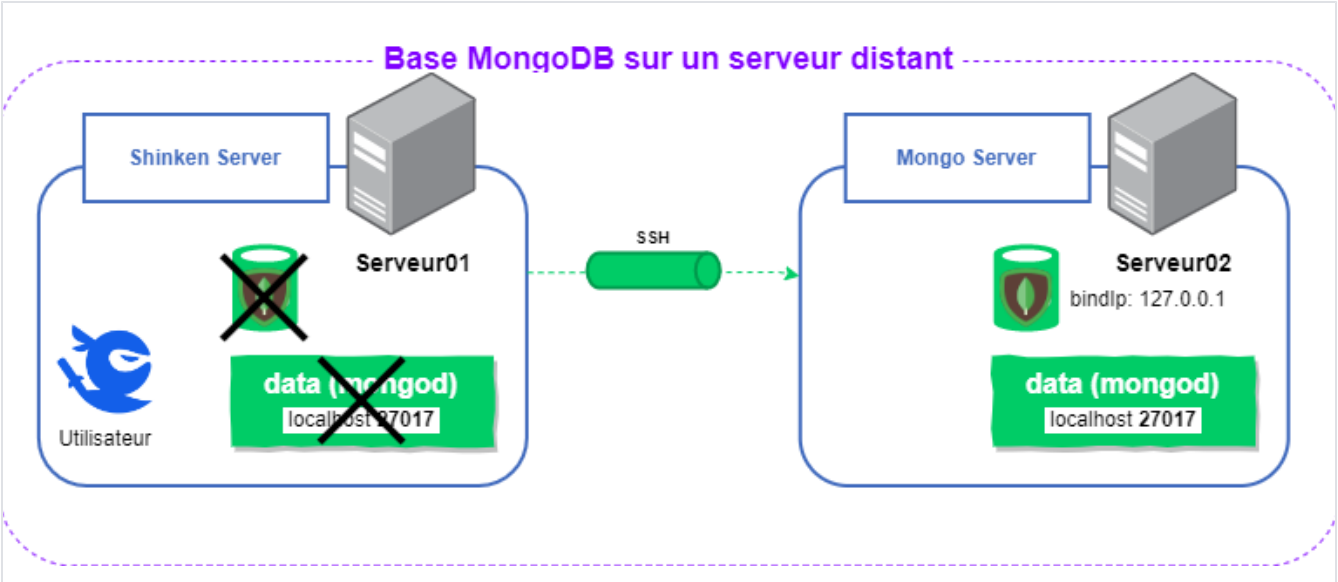
Options génériques




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

Option	Valeur par défaut	Description
--mongo-host *ARG*	**localhost**	Nom ou IP du serveur MongoDB.
--mongo-port *ARG*	**27017**	Port de la base MongoDB.
--mongo-database *ARG*	**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.

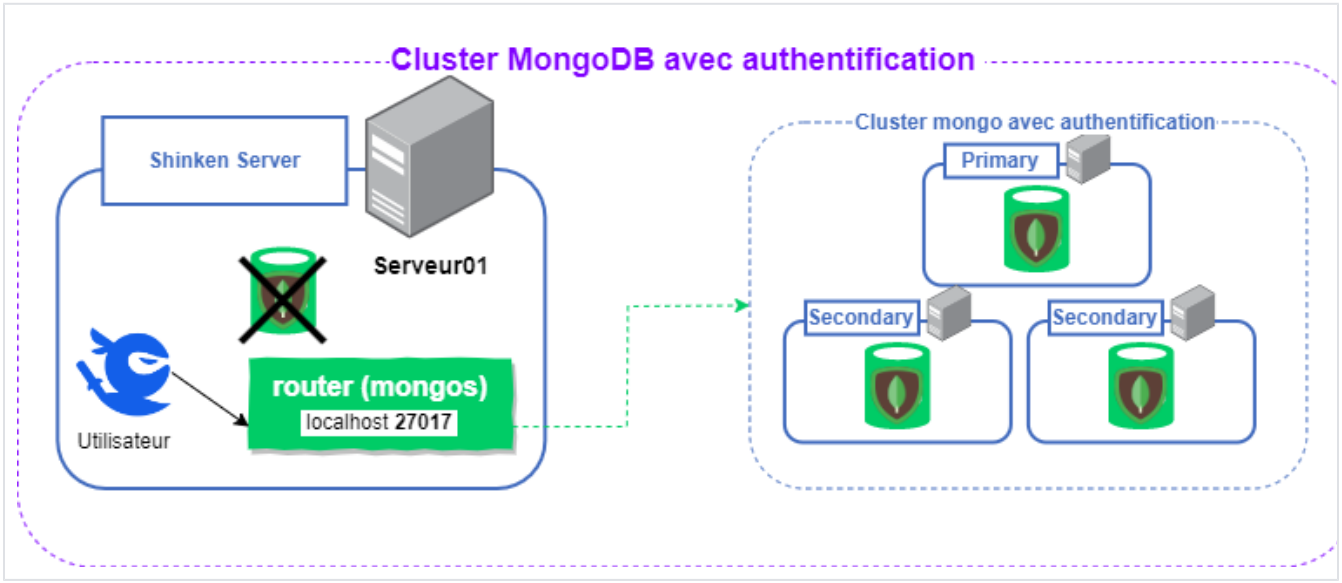
Options de connexion SSH



```
[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
--mongo-use-ssh	---	Active la connexion SSH au serveur MongoDB.
--mongo-ssh-key *ARG*	/var/lib/shinken/.ssh/id_rsa	Clé privée SSH pour la connexion au serveur MongoDB. À utiliser en complément de l'option --mongo-use-ssh.
--mongo-ssh-user *ARG*	shinken	Utilisateur à utiliser pour la connexion SSH. À utiliser en complément de l'option --mongo-use-ssh.

Options d'authentification

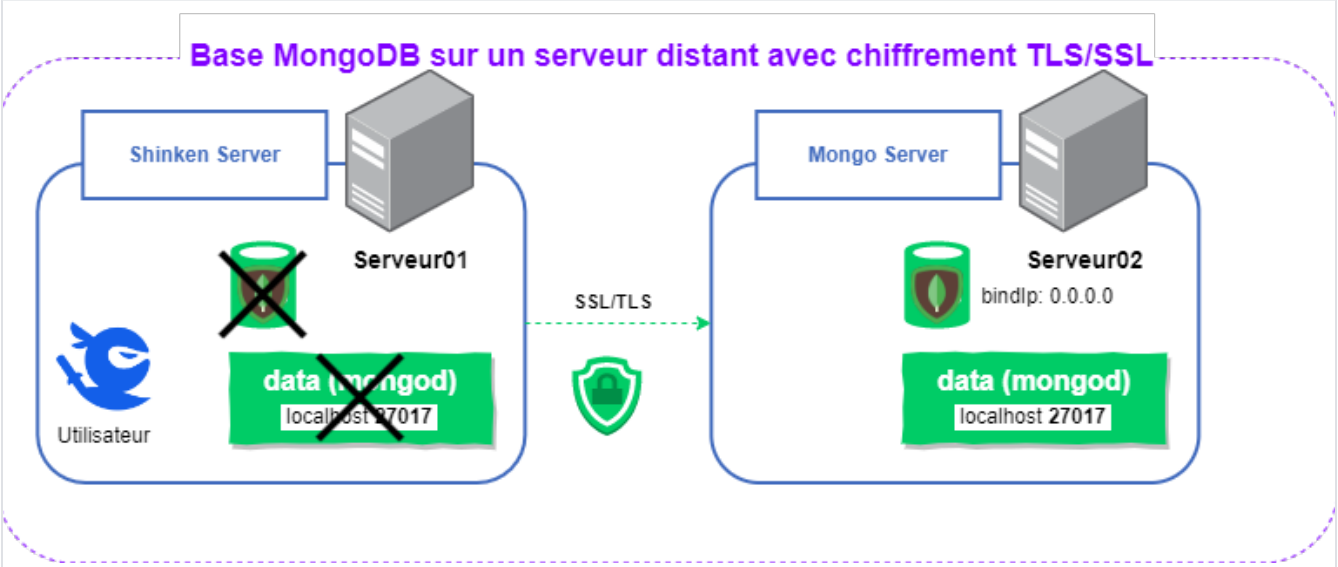



```
[root@serveur01 ~] shinken-command  --mongo-host 127.0.0.1  --mongo-port 27017  --mongo-username shinken --mongo-password shinken
```

Option	Valeur par défaut	Description
--mongo-username *ARG*	---	Utilisateur pour l'authentification avec mot de passe.

--mongo-password *ARG*	---	Mot de passe de l'utilisateur pour l'authentification avec mot de passe. À utiliser en complément de l'option --mongo-username. Si l'option **--mongo-password** est utilisée, le mot de passe risque d'être visible dans l'historique des commandes (*via la commande* ***history***). Pour éviter d'exposer le mot de passe, il est possible d'utiliser cette commande uniquement avec l'option **--mongo-username**. Un prompt interactif apparaîtra alors pour demander le mot de passe. Pour automatiser les commandes dans un script, il est possible de rediriger le contenu d'un fichier contenant le mot de passe *(par exemple : --mongo-password $(cat my_file_with_password)*).

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
```

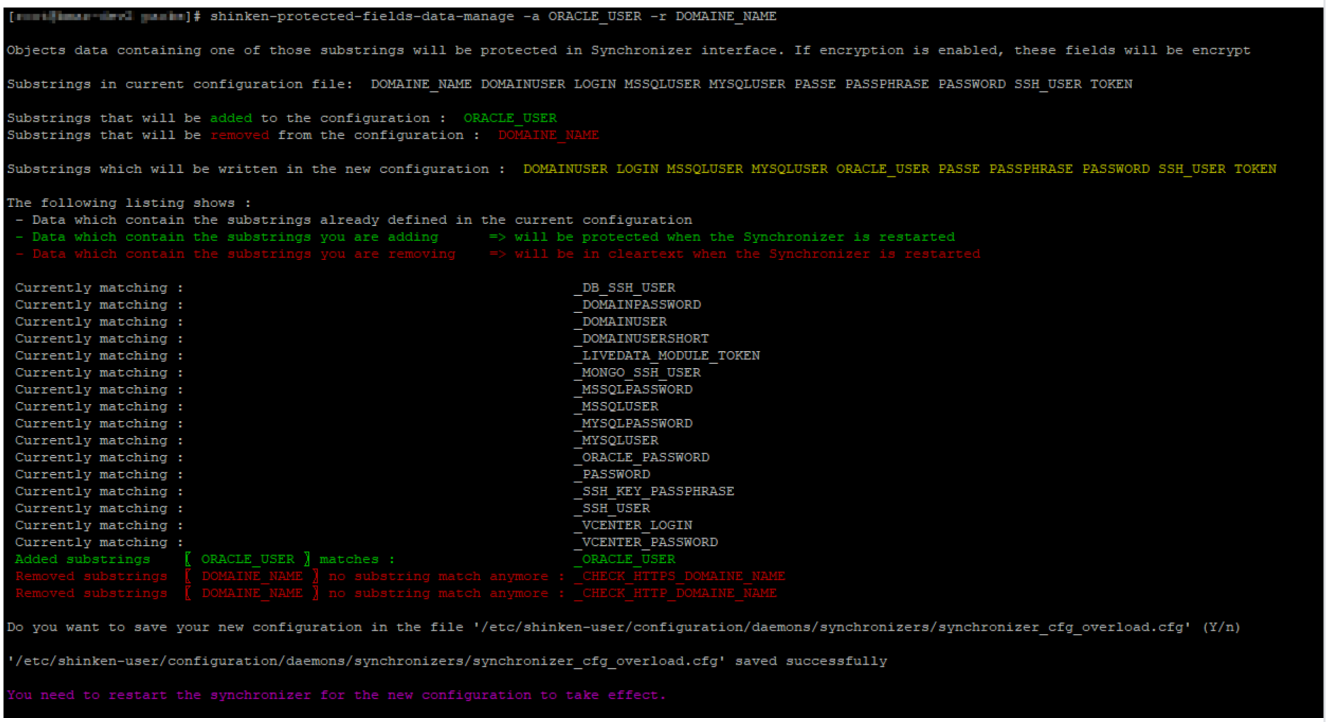
Option	Valeur par défaut	Description
--mongo-ssl	---	Active SSL/TLS pour les communications avec la base MongoDB.
--mongo-ssl-ca-file *ARG*	---	Chemin vers le fichier de l'autorité de certification (*CA*) utilisé pour vérifier le certificat SSL de MongoDB. À utiliser en complément de l'option --mongo-ssl.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

--mongo-ssl-crl-file ***ARG***	---	Chemin vers le fichier CRL (*liste de révocation*) des certificats SSL à rejeter. À utiliser en complément de l'option --mongo-ssl.
--mongo-ssl-allow-invalid-hostnames	---	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 --mongo-ssl.
--mongo-ssl-allow-invalid-certificates	---	Accepter le certificat SSL de MongoDB même s'il est invalide, par exemple expiré. À utiliser en complément de l'option --mongo-ssl.

Exemple

Dans cet exemple, on ajoute le motif ORACLE_USER et on retire en même temps le motif DOMAINE_NAME. Dans la liste des données, on peut constater :

- En blanc, toutes les données qui ne seront pas impactées.
- En vert, les données présentes sur des éléments Shinken qui seront protégées après le redémarrage du Synchronizer.
- En rouge, les données présentes sur des éléments Shinken qui ne seront plus protégées après le redémarrage du Synchronizer.



```
[[redacted]]# shinken-protected-fields-data-manage -a ORACLE_USER -r DOMAINE_NAME

Objects data containing one of those substrings will be protected in Synchronizer interface. If encryption is enabled, these fields will be encrypt

Substrings in current configuration file:  DOMAINE_NAME DOMAINUSER LOGIN MSSQLUSER MYSQLUSER PASSE PASSPHRASE PASSWORD SSH_USER TOKEN

Substrings that will be added to the configuration :  ORACLE_USER
Substrings that will be removed from the configuration :  DOMAINE_NAME

Substrings which will be written in the new configuration :  DOMAINUSER LOGIN MSSQLUSER MYSQLUSER ORACLE_USER PASSE PASSPHRASE PASSWORD SSH_USER TOKEN

The following listing shows :
 - Data which contain the substrings already defined in the current configuration
 - Data which contain the substrings you are adding      => will be protected when the Synchronizer is restarted
 - Data which contain the substrings you are removing    => will be in cleartext when the Synchronizer is restarted

 Currently matching :                                        _DB_SSH_USER
 Currently matching :                                        _DOMAINPASSWORD
 Currently matching :                                        _DOMAINUSER
 Currently matching :                                        _DOMAINUSERSHORT
 Currently matching :                                        _LIVEDATA_MODULE_TOKEN
 Currently matching :                                        _MONGO_SSH_USER
 Currently matching :                                        _MSSQLPASSWORD
 Currently matching :                                        _MSSQLUSER
 Currently matching :                                        _MYSQLPASSWORD
 Currently matching :                                        _MYSQLUSER
 Currently matching :                                        _ORACLE_PASSWORD
 Currently matching :                                        _PASSWORD
 Currently matching :                                        _SSH_KEY_PASSPHRASE
 Currently matching :                                        _SSH_USER
 Currently matching :                                        _VCENTER_LOGIN
 Currently matching :                                        _VCENTER_PASSWORD
 Added substrings    [ ORACLE_USER ] matches :               _ORACLE_USER
 Removed substrings  [ DOMAINE_NAME ] no substring match anymore : _CHECK_HTTPS_DOMAINE_NAME
 Removed substrings  [ DOMAINE_NAME ] no substring match anymore : _CHECK_HTTP_DOMAINE_NAME

Do you want to save your new configuration in the file '/etc/shinken-user/configuration/daemons/synchronizers/synchronizer_cfg_overload.cfg' (Y/n)

'/etc/shinken-user/configuration/daemons/synchronizers/synchronizer_cfg_overload.cfg' saved successfully

You need to restart the synchronizer for the new configuration to take effect.
```

Dans cet exemple,

- Tant que le Synchronizer n'a pas redémarré, aucune modification n'est effectuée
- Si un utilisateur ajoute une donnée SUB_DOMAIN_NAME, elle sera protégée