

Shinken-backup et Shinken-restore, les commandes de sauvegarde et de restauration

Sommaire

- Concept
- Sauvegardes et restaurations entre des distributions différentes
- Commande de sauvegarde (shinken-backup)
 - Utilisation
 - Options pour la sauvegarde
 - Options de connexion à la base MongoDB
 - Options génériques
 - Options de connexion SSH
 - Options d'authentification
 - Options SSL/TLS
 - Sauvegarde de données de configuration chiffrées
 - Sauvegarde d'une configuration anonymisée
- Exemples
 - Sauvegarde complète d'un serveur hébergeant l'ensemble des démons
 - Sauvegarde sur un serveur hébergeant le démon Synchronizer
- Erreur lors du Backup
- Commande de restauration (shinken-restore)
 - Utilisation
 - Options pour la restauration
 - Options de connexion à la base MongoDB
 - Options génériques
 - Options de connexion SSH
 - Options d'authentification
 - Options SSL/TLS
 - Restauration d'une base de données de configuration dont les données sensibles sont chiffrées
- Exemples
 - Restauration d'une sauvegarde complète de Shinken depuis le dossier ~/shinken-backups
 - Restauration d'une sauvegarde de la configuration de Shinken, lancé depuis le serveur hébergeant l'Arbiter /Synchronizer :
 - Restauration d'une sauvegarde des données de l'utilisateur "monutilisateur" de Shinken (portails, listes, favoris, tableaux de bord), lancé depuis le serveur hébergeant le Broker
 - Restauration avec une sauvegarde contenant seulement certaines parties
- Sauvegarde et restauration de la rétention du Scheduler
- Erreurs lors de la restauration

Concept

Afin de prévenir d'une perte de données après un crash d'un ou de plusieurs des serveurs utilisés dans l'architecture Shinken, il est conseillé d'utiliser les commandes shinken-backup et shinken-restore.

Ces deux commandes permettront de sauvegarder ou de restaurer tout ou une partie d'un serveur de l'architecture Shinken.

Sauvegardes et restaurations entre des distributions différentes

Shinken Enterprise supporte la sauvegarde et la restauration entre deux distributions différentes.

Cela signifie qu'une sauvegarde effectuée sous une distribution RedHat ou ses dérivées (CentOS, Alma, Rocky) peut être restaurée sur une distribution Debian et inversement.

	Sauvegarde sous RHEL / CentOS 7	Sauvegarde sous RHEL / Alma / Rocky 8	Sauvegarde sous RHEL / Alma / Rocky 9	Sauvegarde sous Debian 13
Restauration sous RHEL / CentOS 7	Oui ✓	Oui ✓	Oui ✓	Oui ✓
Restauration sous RHEL / Alma / Rocky 8	Oui ✓	Oui ✓	Oui ✓	Oui ✓
Restauration sous RHEL / Alma / Rocky 9	Oui ✓	Oui ✓	Oui ✓	Oui ✓

Restauration sous Debian 13	Oui ✓	Oui ✓	Oui ✓	Oui ✓
-----------------------------	-------	-------	-------	-------

Commande de sauvegarde (shinken-backup)

Utilisation

La commande de base pour la sauvegarde complète d'un serveur shinken est la suivante :

```
shinken-backup
```



La commande shinken-backup ne peut pas être exécutée dans les dossiers /etc/shinken et /etc/shinken-user

Options pour la sauvegarde

Option	Valeur par défaut	Description
<code>--help, -h</code>	---	Affiche l'aide de la commande.
<code>--output-directory, -od ARG</code>	Répertoire où est lancée la commande.	Répertoire où sera enregistrée la sauvegarde. - Le chemin fourni peut être absolu (<i>/root/backups/example</i>). - Le chemin peut être relatif. Le répertoire est alors créé à partir du répertoire où est exécutée la commande shinken-backup.
<code>--output-name, -on ARG</code>	YYYY-MM-DD_HH-MM-SS_ShinkenVersion	Nom du répertoire contenant la sauvegarde.
<code>--addons, -a</code>	---	Sauvegarde les configurations et les données des addons : nagvis : - sauvegarde les dossiers : <i>/opt/nagvis</i> <i>/etc/httpd/conf.d/</i> sous RHEL / CentOS 7 ou RHEL / Alma / Rocky 8 ou RHEL / Alma / Rocky 9 <i>/etc/apache2/conf-available/</i> sous Debian 13 nagvis-shinken-architecture : - sauvegarde les dossiers : <i>/var/lib/shinken-nagvis</i> <i>/etc/httpd/conf.d/</i> sous RHEL / CentOS 7 ou RHEL / Alma / Rocky 8 ou RHEL / Alma / Rocky 9 <i>/etc/apache2/conf-available/</i> sous Debian 13
<code>--configuration, -c</code>	---	Sauvegarde les données de configuration : - Les données du Synchronizer dans la base MongoDB. - Les fichiers de configuration dans <i>/etc/shinken/</i>. - Les fichiers de configuration dans <i>/etc/shinken-user</i>. - Les sondes utilisateurs dans <i>/var/lib/shinken-user/libexec</i> (⚠ <i>Si les sondes utilisent des dépendances systèmes, il faudra aussi qu'elles soient présentes lors de la restauration pour que les sondes soient opérationnelles</i>). - La rétention de l'Arbiter. La commande doit être exécutée sur la machine du Synchronizer et avoir accès à la base MongoDB.

-- configurati on- anonymou s, -ca	---	Sauvegarde les données de configuration anonymisées Les données du Synchronizer dans la base MongoDB anonymisées (Voir le chapitre : Sauvegarde d'une configuration anonymisée).  La commande doit être exécutée sur la machine du Synchronizer et avoir accès à la base MongoDB.
--events, -e	---	Sauvegarde les données du bac à événements Les données dans la base MongoDB du bac à événements.  La commande doit avoir accès à la base MongoDB.
--log, -l	---	Sauvegarde les logs. Tous les fichiers de logs du dossier /var/log/shinken.
-- modules, -M	---	Permet de sauvegarder les données de tous les modules NON fournis par Shinken Enterprise (exemple : le module <i>mssql_export</i>) Les données des modules dans MongoDB  La commande doit avoir accès à la base MongoDB.
- - m o d u l e- t y p e ARG	---	Permet de sauvegarder les données d'un seul type de module NON fourni par Shinken Enterprise. À utiliser en complément de l'option --modules/-M.
--list- modules, -L	---	Permet de lister tous les types de modules NON fournis par Shinken Enterprise qui peuvent être sauvegardés.  La commande doit avoir accès à la base MongoDB.
-- metrology, -m	---	Sauvegarde les données de métrologie. Les fichiers du dossier /opt/graphite/storage/whisper.  La commande doit être exécutée sur la machine avec le carbon-cache.

<code>--sla, -s</code>	---	Sauvegarde les données SLA. Les données dans la base MongoDB des SLA.  La commande doit avoir accès à la base MongoDB.
<code>--user, -u</code>	---	Sauvegarde les données des utilisateurs de l'interface de visualisation (<i>portails, listes, favoris, tableaux de bord...</i>) - Les données dans la base MongoDB des dashboards utilisateurs. - Les données dans la base MongoDB des listes utilisateurs. - Les données dans la base MongoDB des préférences utilisateurs. - Les données dans la base MongoDB du module service-weather. - Les données temporaires du module service-weather.  La commande doit être exécutée sur la machine d'un Broker et avoir accès à la base MongoDB.
<code>--database-name, -d</code>	---	Nom de la base de données MongoDB dans laquelle les données sont stockées. Cette option peut être utilisée en complément des options <code>--configuration</code>, <code>--configuration-anonymous</code>, <code>--events</code>, <code>--modules</code>, <code>--list-modules</code>, <code>--sla</code> ou <code>--user</code>, lorsque les données ne se trouvent pas dans la base par défaut.  Si elle est spécifiée, le même nom de base sera appliqué à l'ensemble des éléments sauvegardés. <pre>shinken-backup --sla --user --database-name shinken-France</pre> Dans cet exemple, la base de données shinken-France sera utilisée pour les sauvegardes des SLA et des utilisateurs.

Options de connexion à la base MongoDB

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

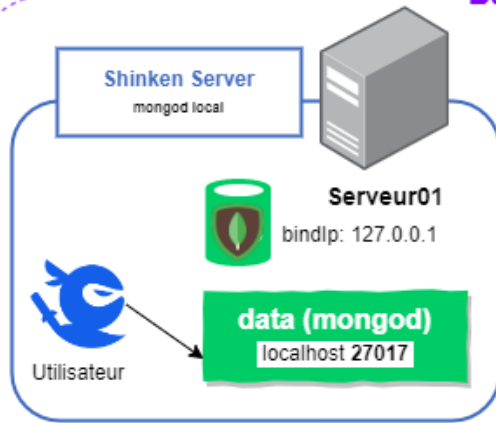
- La base de données MongoDB ne se trouve pas sur la machine qui exécute la commande.
- L'authentification par mot de passe à la base MongoDB est activée.
- Le port de MongoDB n'est pas celui par défaut (*défaut : 27017*).



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

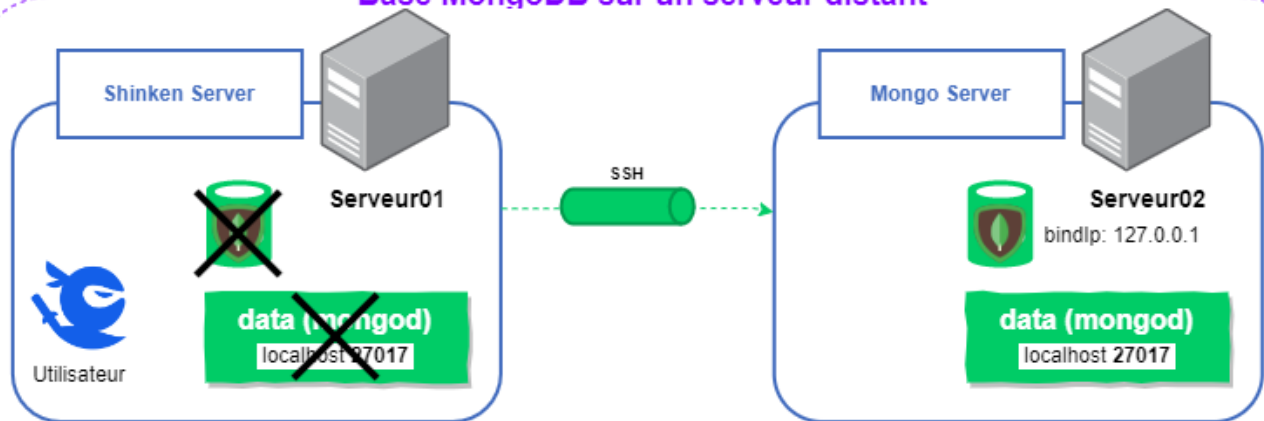


```
[root@serveur01 ~] shinken-commande --mongo-host 127.0.0.1 --mongo-port 27017 --mongo-database shinken
```

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

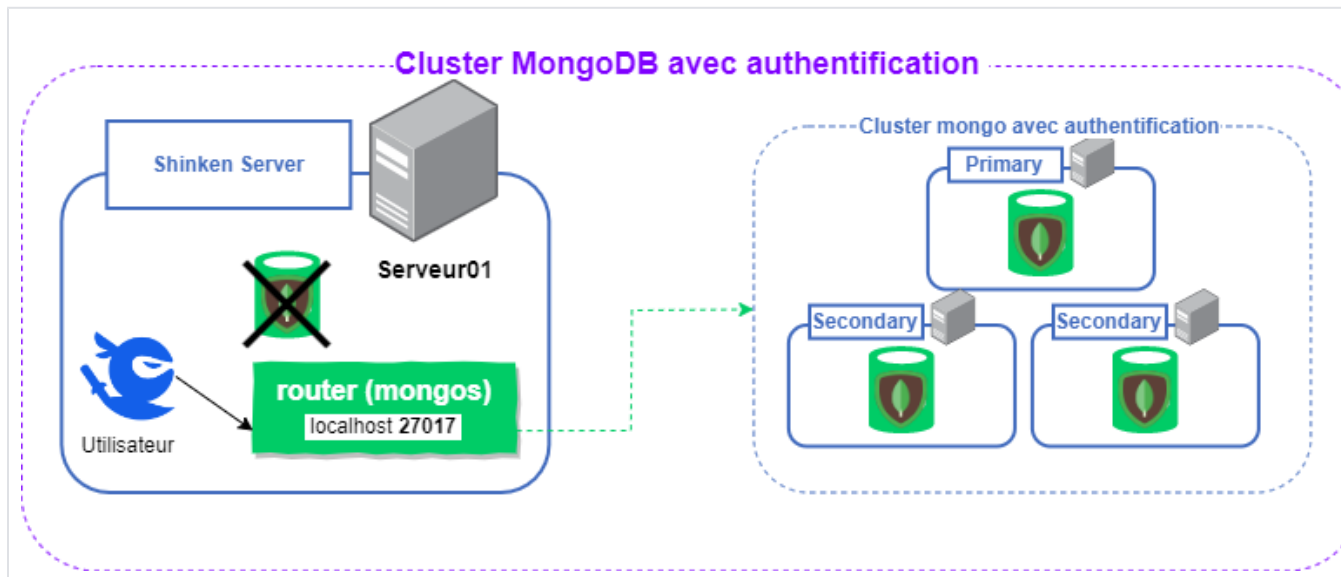
Base MongoDB sur un serveur distant



```
[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

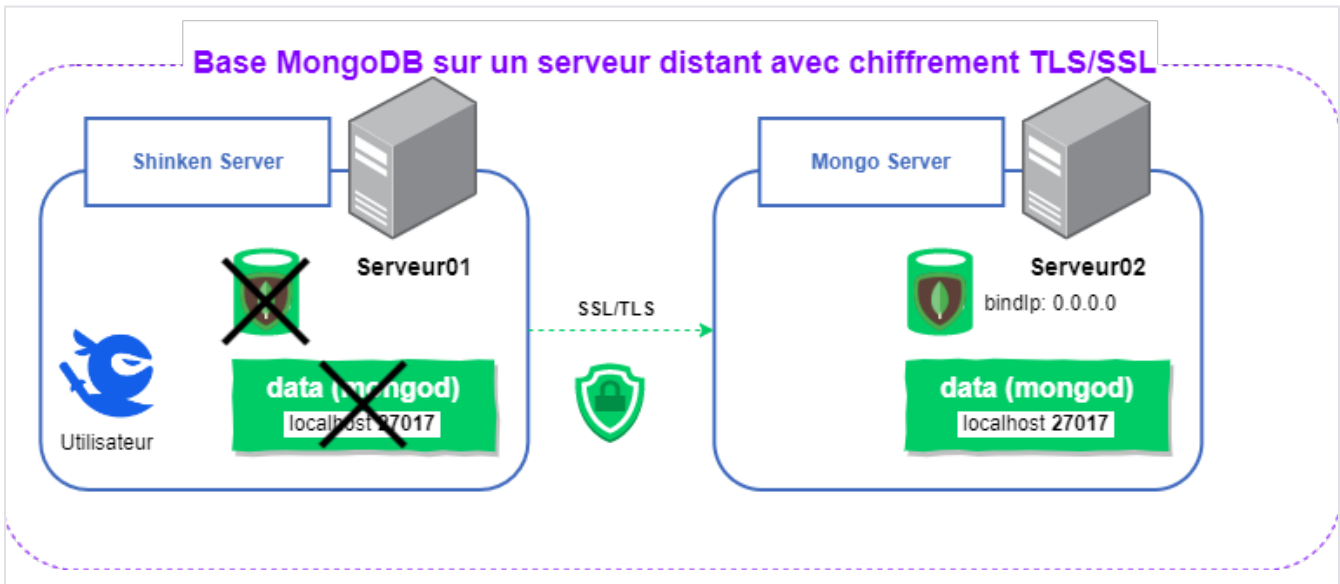


```
[root@serveur01 ~] shinken-command --mongo-host 127.0.0.1 --mongo-port 27017 --mongo-username shinken --mongo-password shinken
```

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

<pre>-- mongo -- password ARG</pre>	--- Mot de passe de l'utilisateur pour l'authentification avec mot de passe. À utiliser en complément de l'option <code>--mongo-username</code>. ✔ Si l'option --mongo-password est utilisée, le mot de passe risque d'être visible dans l'historique des commandes (<i>via la commande history</i>). 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 (<i>par exemple : <code>--mongo-password \$(cat my_file_with_password)</code></i>).
-------------------------------------	--

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> .

Sauvegarde de données de configuration chiffrées

Lors de la sauvegarde de données de configuration avec des données protégées, la commande "shinken-backup" affiche un avertissement si la clé de chiffrement de la base n'a pas été exportée.

La sauvegarde est tout de même effectuée, mais il est préconisé d'effectuer un export de la clé en utilisant la commande "shinken-protected-fields-keyfile-export" (voir la page [shinken-protected-fields-keyfile-export](#)) .

```
[~]# shinken-backup --configuration

Saving Configuration

For security reasons, you are responsible for saving and storing the Protected fields key separately.
Please run the following command to export the key so you can save the result in a secure location :
shinken-protected-fields-keyfile-export

Failure to do so will result in the loss of your database when you restore from backup

Configuration save size: 44M

Done : your backup directory is /root/shinken-backups/2025-05-27__10-28-32__VC_Charlie-005.0-B25.fr
```

Sauvegarde d'une configuration anonymisée

Afin de transmettre une configuration au support, il est possible d'anonymiser la configuration lors de la sauvegarde.

L'anonymisation est réalisée pour les données présentes dans la base de données du Synchronizer.

- Si l'option `--configuration-anonymous` est utilisée, les autres options (`--sla` par exemple) sont ignorées.
- Les fichiers de configuration des démons et de l'architecture de Shinken ne sont pas sauvegardés.

Cette étape est réalisée entre la récupération des données dans la base et l'écriture des fichiers de sauvegarde, la base de données n'est donc pas impactée par ce fonctionnement.

Le tableau suivant présente les propriétés qui seront remplacées avant d'être écrites dans les fichiers de sauvegarde.

Type de donnée	Valeur remplacée	Description
address	clean address	Permet de masquer les adresses des équipements.
check_command	clean_command	Permet de masquer les commandes : une fois restaurée, la configuration pourra démarrer sans effectuer de check.
poller_tag		La valeur est supprimée pour permettre à la configuration de démarrer.
reactionner_tag		La valeur est supprimée pour permettre à la configuration de démarrer.
realm		La valeur est supprimée pour permettre à la configuration de démarrer.
password	clean	Les mots de passe sont effacés.

Toutes les données utilisateur, chiffrées ou non, présentes dans le système de champs protégé sont également remplacées par la valeur "clean".

La liste des champs protégés est disponible via la commande **shinken-protected-fields-data-manage**.

Le nom de la sauvegarde générée sera succédé de "--anonymous" afin de le distinguer des sauvegardes par défaut.

 Exemple de nom de sauvegarde anonyme : 2020-10-27__10-49-55__V02.08.02-001_BUILD066.fr--anonymous

 Les données ne sont ni effacées ni modifiées sur le serveur sur lequel le shinken-backup est effectué. Elles sont simplement remplacées avant d'être écrites dans les fichiers de sauvegardes.

 La sauvegarde réalisée n'est pas chiffrée, car toutes les données présentant un risque ont été remplacées.

Exemples

Sauvegarde complète d'un serveur hébergeant l'ensemble des démons

```
root@vm-shinken: ~
$ shinken-backup
Saving Events
  Events save size: 51M
Saving Sla
  Sla save size: 147M
Saving service-weather
  service-weather size: 2M
Saving User
  User save size: 136K
Saving Configuration
  Configuration save size: 17M
Saving Metrology
  Metrology save size: 47M
Saving Logs
  Logs save size: 1.1G
Saving addons data
  Addons save size: 4.8M
Saving custom modules data
  There is no custom modules in database to backup.

Done : your backup directory is /root/shinken-backups/2021-02-02__08-22-45__V02.08.02-002.fr
```

Sauvegarde sur un serveur hébergeant le démon Synchronizer

De la configuration:

```
[root@bmar-dev2 ~]# shinken-backup --configuration
Saving Configuration
  Configuration save size: 12M

Done : your backup directory is /root/shinken-backups/2021-03-27 01-10-28 V02.07.06-release.fr-Patched-09
```

Sauvegarde complète :

```
[root@bmar-dev2 ~]# shinken-backup
Saving Sla
  Skipped : No sla data in database
Saving User
  Skipped : No user data in database
Saving Configuration
  Configuration save size: 16M
Saving Metrology
  Metrology save size: 127M
Saving Logs
  Logs save size: 8.0K
Saving addons data
  Addons save size: 4.6M
Saving custom modules data
  Skipped : There is no custom modules in database to backup.

Done : your backup directory is /root/shinken-backups/2021-03-26__21-10-05__V02.07.06-release.fr-Patched-09
```

Erreur lors du Backup

Si le backup nécessite l'archivage de données depuis la base MongoDB, il est essentiel de vérifier la connectivité avant de lancer la sauvegarde de Shinken. En cas d'inaccessibilité de la base, un message s'affichera, indiquant les paramètres utilisés pour la tentative de connexion.

```
Unable to backup Shinken because MongoDB is unreachable. Please ensure that the MongoDB settings provided are correct before trying again

Parameters used to connect to MongoDB:
--mongo-host      : ( localhost )
--mongo-port      : ( 27017 )
```

Lorsqu'une erreur survient pendant la sauvegarde, un message d'erreur indique la section où l'erreur s'est produite. Le dossier qui aurait dû contenir le contenu de la sauvegarde contiendra les logs d'erreurs.

Dans ce message est précisé le nom du dossier créé :

```
$ shinken-backup
Saving Events
  Events save size: 392K
Saving Sla
Mongodb database dump failed, please look at /root/shinken-backups/2022-08-26_14-01-11_V02.08.02-MCO07.Release-Service.Step2-Openwrt2.831.fr/sla/backup-sla..
```



La partie floutée du chemin de la capture d'écran correspond au dossier créé par le shinken-restore. Ce nom de dossier est créé à partir de la date, du nom de la version de Shinken et de la langue.

Commande de restauration (shinken-restore)

Utilisation

```
shinken-restore DIRECTORY-TO-RESTORE
```

Le dossier "DIRECTORY-TO-RESTORE" doit contenir au moins un des dossiers de sauvegarde suivant:

- configuration/
- context/
- events/
- logs/
- metrology/
- sla/
- user/
- weather/

Options pour la restauration

Option	Valeur par défaut	Description
<code>--help, -h</code>	---	Affiche l'aide de la commande.
<code>--addons, -a</code>	---	Restaure les configurations et les données des addons (<i>exemple: nagvis</i>) présent dans l'archive : • nagvis : ◦ restaure les dossiers : ▪ /opt/nagvis ▪ /etc/httpd/conf.d/ sous RHEL / CentOS 7 ou RHEL / Alma / Rocky 8 ou RHEL / Alma / Rocky 9 ▪ /etc/apache2/conf-available/ sous Debian 13 • nagvis-shinken-architecture : ◦ restaure les dossiers : ▪ /var/lib/shinken-nagvis ▪ /etc/httpd/conf.d/ sous RHEL / CentOS 7 ou RHEL / Alma / Rocky 8 ou RHEL / Alma / Rocky 9 ▪ /etc/apache2/conf-available/ sous Debian 13
<code>--configuration, -c</code>	---	Restaure les données de configuration. : • Les données du Synchronizer dans MongoDB. • Les fichiers de configuration dans /etc/shinken/. • Les fichiers de configuration dans /etc/shinken-user. • Les sondes utilisateurs dans /var/lib/shinken-user/libexec ( <i>Si les sondes utilisent des dépendances systèmes, il faudra aussi qu'elles soient présentes lors de la restauration pour que les sondes soient opérationnelles</i>). • La rétention de l'Arbiter.  La commande doit être exécutée sur la machine du Synchronizer et avoir accès à la base MongoDB.
<code>--with-key-backup</code> ARG	---	Pour une sauvegarde contenant des données protégées, restaure également le hash de la clé de chiffrement spécifiée. Il s'agit du résultat de la commande shinken-protected-fields-keyfile-export . À utiliser en complément de l'option <code>--configuration, -c</code>.
<code>--restore-only-configuration-database, --rocd</code>	---	Cette option restaure uniquement les données de la configuration, sans restaurer les fichiers de configuration (<i>.cfg</i>). Cette option est pratique pour transférer la politique de supervision d'une préproduction vers une production (ou inversement), sans changer les fichiers de définition des démons.  La commande doit être exécutée sur la machine du Synchronizer et avoir accès à la base MongoDB.

<code>--events, -e</code>	---	Restaure les données du bac à événements.  La commande doit avoir accès à la base MongoDB.
<code>--log, -l</code>	---	Restaure les logs. • Restaure les fichiers de logs dans le dossier <code>/var/log/shinken</code>. • Supprime les fichiers présents dans <code>/var/log/shinken</code> avant la restauration des logs.
<code>--metrology, -m</code>	---	Restaure les données de métrologie. • Restaure les fichiers du dossier <code>/opt/graphite/storage/whisper</code>. • Supprimer les fichiers présents dans <code>/opt/graphite/storage/whisper/</code> avant la restauration des logs.  La commande doit être exécutée sur la machine avec le carbon-cache.
<code>--modules, -M</code>	---	Restaure les données de tous les modules NON fournis par Shinken Enterprise (<i>exemple : le module <code>mssql_export</code></i>) • Les données des modules dans MongoDB.  La commande doit avoir accès à la base MongoDB.
<code>--module-type ARG</code>	---	Permet de restaurer les données d'un seul type de module NON fournis par Shinken Enterprise. À utiliser en complément de l'option <code>--modules, -M</code>.  La commande doit avoir accès à la base MongoDB.
<code>--list-modules, -L</code>	---	Permet de lister tous les types de modules NON fournis par Shinken Enterprise présent dans l'archive de sauvegarde et qui peuvent être restaurés.
<code>--sla, -s</code>	---	Restaure les données SLA. • Les données mongo des SLA.  La commande doit avoir accès à la base MongoDB.

--user, -u	---	Restaure les données des utilisateurs de l'interface de visualisation (<i>portails, listes, favoris, tableaux de bord...</i>) • Les données mongo des dashboards utilisateurs. • Les données mongo des listes utilisateurs. • Les données mongo des préférences utilisateurs. • Les données mongo du module service-weather. • Les données temporaires du module service-weather.  La commande doit être exécutée sur la machine d'un Broker et avoir accès à la base MongoDB.
--restore-only-user <i>ARG</i>	---	Restaure les données de l'interface de visualisation (<i>portails, listes, favoris, tableaux de bord...</i>) pour l'utilisateur spécifié. La valeur à utiliser est le nom (<i>contact_name</i>) de l'utilisateur. • Les données mongo des dashboards de l'utilisateur. • Les données mongo des listes de l'utilisateur. • Les données mongo des préférences de l'utilisateur. • Les données mongo des météo de l'utilisateur. 💡 Cette option ne demande pas le redémarrage de Shinken. À utiliser en complément de l'option --user, -u.  La commande doit avoir accès à la base MongoDB.

Options de connexion à la base MongoDB

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

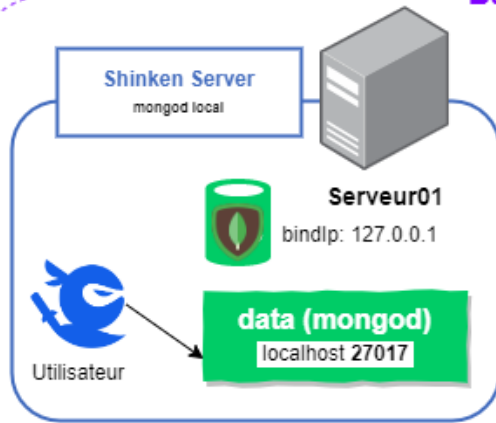
- La base de données MongoDB ne se trouve pas sur la machine qui exécute la commande.
- L'authentification par mot de passe à la base MongoDB est activée.
- Le port de MongoDB n'est pas celui par défaut (*défaut : 27017*).



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

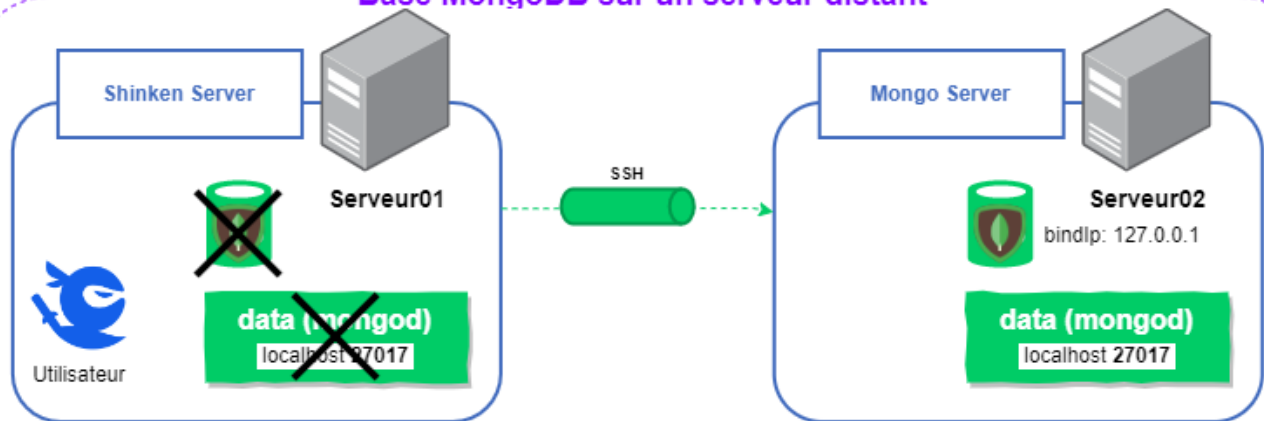


```
[root@serveur01 ~] shinken-commande --mongo-host 127.0.0.1 --mongo-port 27017 --mongo-database shinken
```

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

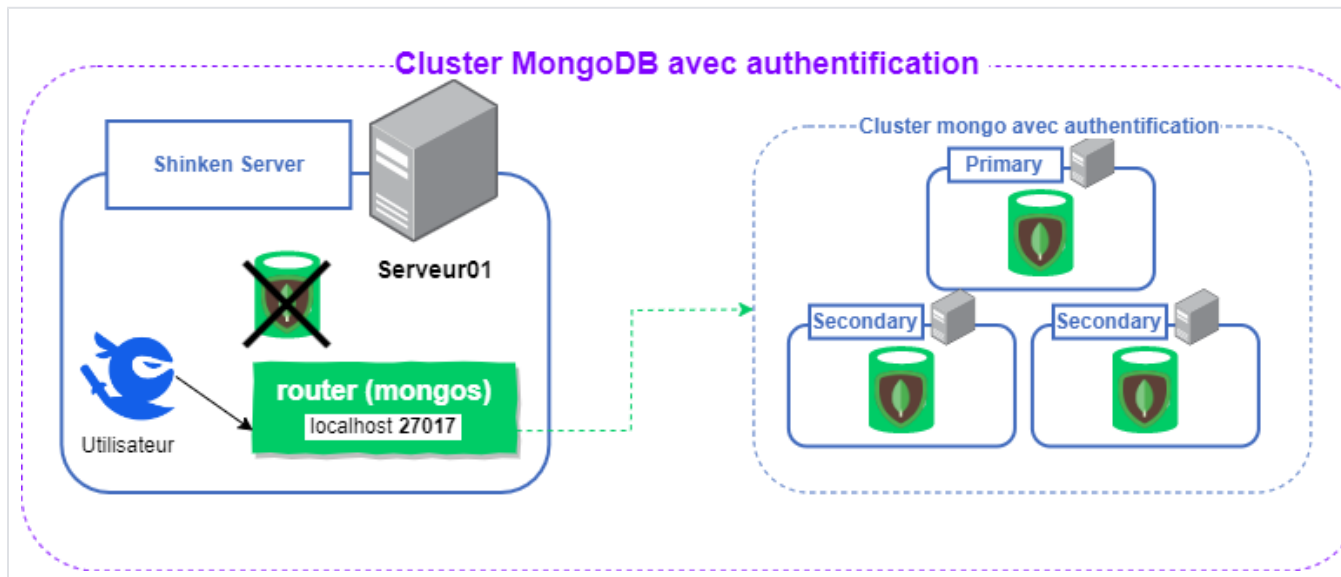
Base MongoDB sur un serveur distant



```
[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

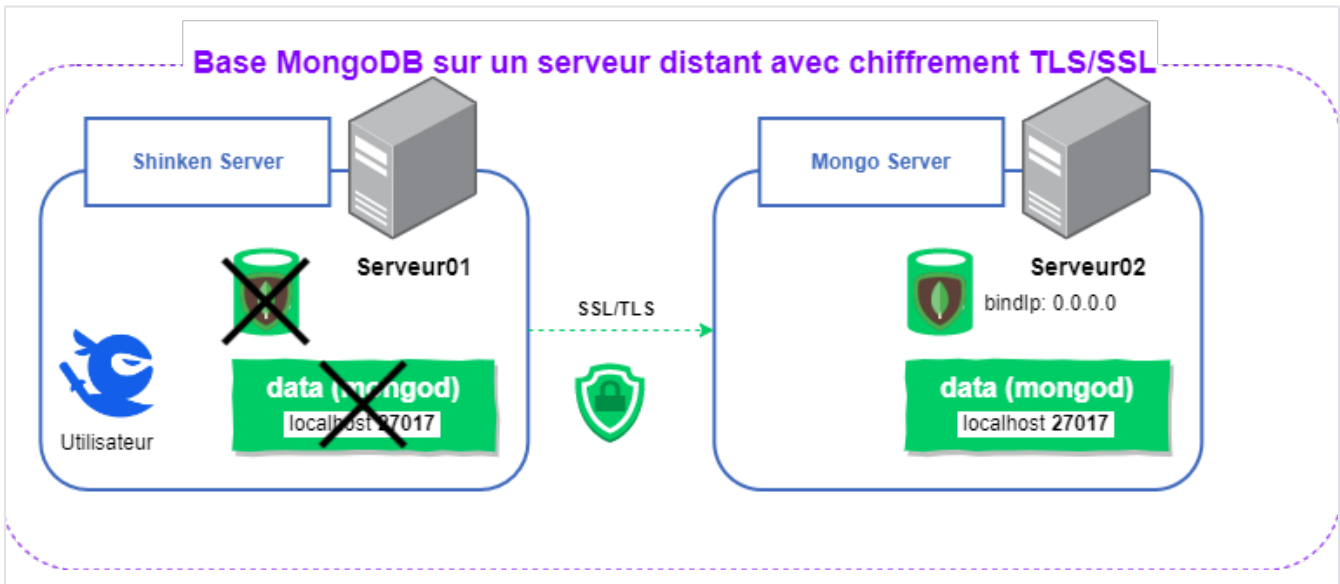


```
[root@serveur01 ~] shinken-command --mongo-host 127.0.0.1 --mongo-port 27017 --mongo-username shinken --mongo-password shinken
```

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

<pre>-- mongo -- password ARG</pre>	--- Mot de passe de l'utilisateur pour l'authentification avec mot de passe. À utiliser en complément de l'option <code>--mongo-username</code>. ✔ Si l'option --mongo-password est utilisée, le mot de passe risque d'être visible dans l'historique des commandes (<i>via la commande history</i>). 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 (<i>par exemple : <code>--mongo-password \$(cat my_file_with_password)</code></i>).
-------------------------------------	--

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 <i>ARG</i></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> .

Restauration d'une base de données de configuration dont les données sensibles sont chiffrées

Si une clé est déjà présente sur le serveur et qu'elle est identique à celle de la sauvegarde, **shinken-restore** restaurera cette sauvegarde en utilisant la clé.

Si la clé n'a pas été exportée, **shinken-restore** affichera un avertissement signalant qu'elle est automatiquement exportée dans un fichier temporaire, en enjoignant de le déplacer en lieu sûr.

```
$ shinken-restore /root/shinken-backups/2019-01-16__11-58-28
Restoring from 02.06.00-001_BUILD67.fr to 02.06.00-001_BUILD67.fr
-Restoring Configuration
Checking encryption status for your current database : WARNING

The current key was never exported
  I'm exporting it to /root/protected_fields_key-export-2019-01-16__12-00-47
  The current key will be replaced by the one from the backup.
  => Remove this file as soon as possible to prevent security holes

  The keyfile used in the backup is the same as the one from the current configuration.
Stopping Shinken before restoring
The backup you are restoring contains a encrypted Synchronizer database.
```

Si les deux clés sont différentes, il est nécessaire de spécifier l'option **--with-key-backup** suivi de l'export de la clé, qui permet de faire la restauration et de placer automatiquement la clé fournie.

En suivant, redémarrer le Synchronizer.

- **Le Synchronizer refusera de démarrer tant que la clé n'est pas restaurée.**

Si la clé est égarée, il est possible de la restaurer par l'intermédiaire du support Shinken (voir la page [shinken-protected-fields-keyfile-rescue-from-backup](#)).

```
$ shinken-restore --with-key-backup dmFsaWRhdGlvbWJ3BM3c0ZVRREMQ2STVzVVVHWWxMRcTdEhrZ0dIb2JZZTY2YTZ2REVKKQVNKaz0K /root/shinken-backups/2019-01-16__11-58-28
Restoring from 02.06.00-001_BUILD67.fr to 02.06.00-001_BUILD67.fr
-Restoring Configuration
Checking encryption status for your current database : ENCRYPTED

Stopping Shinken before restoring
Checking protected fields key : OK

The backup you are restoring contains a encrypted Synchronizer database.

Setting addons state
nagvis : ENABLED
nagvis-shinken-architecture : ENABLED
=> Restoring configuration DONE
```

Exemples

Restauration d'une sauvegarde complète de Shinken depuis le dossier ~/shinken-backups

```
root@vm-shinken: ~/shinken-backups

$ shinken-restore 2020-10-27__10-47-43__V02.08.02-001_BUILD066.fr__crypted-backup2

Stopping Shinken before restoring
Restoring from 02.08.02-001_BUILD066.fr to 02.08.02-001_BUILD066.fr
-Restoring Sla                DONE
-Restoring User                DONE
-Restoring Configuration      DONE
-Restoring Metrology          DONE
-Restoring Logs                DONE
-Restoring Weather            DONE

Sanatizing your restored data
  fix_double_link              : skip (unecessary)
  fix_double_sync_keys         : skip (unecessary)
  fix_default_item_se_uuid     : skip (unecessary)
  fix_remove_shinken_core      : skip (unecessary)
  fix_remove_deprecated_check  : skip (unecessary)
  fix_remove_undefined_aix_templates : skip (unecessary)
  fix_flapping_thresholds      : skip (unecessary)
  fix_business_impact          : skip (unecessary)

Done. You can restart your shinken with service-shinken start
```



Après la restauration des données, des actions automatiques (*sanitizes*) sont lancés. Ces actions permettent, si nécessaire, de mettre les données à jours vers la version de Shinken où sont restaurées les données.

Une fois la restauration terminée, démarrer Shinken:

```
service-shinken start
```

Restauration d'une sauvegarde de la configuration de Shinken, lancé depuis le serveur hébergeant l'Arbiter /Synchronizer :

```
root@vm-shinken: ~/shinken-backups

$ shinken-restore --configuration 2020-10-27__10-47-43__V02.08.02-001_BUILD066.fr__crypted-backup2

Stopping Shinken before restoring
Restoring from 02.08.02-001_BUILD066.fr to 02.08.02-001_BUILD066.fr
-Restoring Configuration      DONE

Sanatizing your restored data
  fix_double_link              : executed [OK]
  fix_double_sync_keys         : skip (unecessary)
  fix_default_item_se_uuid     : skip (unecessary)
  fix_remove_shinken_core      : skip (unecessary)
  fix_remove_deprecated_check  : skip (unecessary)
  fix_remove_undefined_aix_templates : skip (unecessary)
  fix_flapping_thresholds      : skip (unecessary)
  fix_business_impact          : skip (unecessary)

Done. You can restart your shinken with service-shinken start
```

Restauration d'une sauvegarde des données de l'utilisateur "monutilisateur" de Shinken (portails, listes, favoris, tableaux de bord), lancé depuis le serveur hébergeant le Broker

```

root@vm-shinken: ~/shinken-backups

$ shinken-restore -u --restore-only-user monutilisateur 2020-10-27__10-47-43__V02.08.02-001_BUILD066.
fr__crypted-backup2
Restoring from 02.08.02-001_BUILD066.fr to 02.08.02-001_BUILD066.fr
-Restoring User
  Restoring only the user monutilisateur

Restore of the user monutilisateur data is OK

```

Restauration avec une sauvegarde contenant seulement certaines parties

```

root@vm-shinken: ~/shinken-backups

$ shinken-restore 2021-04-13__12-18-24__V02.08.02-004_BUILD08.fr/
Restoring from 02.08.02-004_BUILD08.fr to 02.08.02-004_BUILD08.fr
=> Restoring Configuration      :
    => Checking encryption status for your current database : NOT ENCRYPTED
    => The keyfile used in the backup is the same as the one from the current configuration.
    => Stopping Shinken before restoring : DONE
    => Restoring configuration database : DONE
The backup you are restoring contains a encrypted Synchronizer database.
    => Setting addons state
nagvis                          : ENABLED
- [ WARNING ] Cannot find local broker daemon configuration in /etc/shinken/brokers. The 'Livestatus'
module must be enabled manually in your broker configuration file for this addon to work properly.

nagvis-shinken-architecture    : ENABLED
- [ WARNING ] Cannot find local broker daemon configuration in /etc/shinken/brokers. The 'Livestatus'
module must be enabled manually in your broker configuration file for this addon to work properly.
- [ WARNING ] Cannot find local arbiter daemon configuration in /etc/shinken/arbiters. The
'architecture-export' module must be enabled manually in your arbiter configuration file for this addon to
work properly.

    => Restoring configuration files      : DONE
    => Restoring configuration            : DONE
=> Restoring Sla                  : SKIPPED
No sla data to restore in /tmp/shinken-backups/2021-04-13__12-18-24__V02.08.02-004_BUILD08.fr
=> Restoring service-weather      : SKIPPED
No service-weather data to restore in /tmp/shinken-backups/2021-04-13__12-18-24__V02.08.02-004_BUILD08.fr
=> Restoring Events               : SKIPPED
No events data to restore in /tmp/shinken-backups/2021-04-13__12-18-24__V02.08.02-004_BUILD08.fr
=> Restoring User                 : SKIPPED
No user data to restore in /tmp/shinken-backups/2021-04-13__12-18-24__V02.08.02-004_BUILD08.fr
=> Restoring Metrology            : SKIPPED
No metrology data to restore in /tmp/shinken-backups/2021-04-13__12-18-24__V02.08.02-004_BUILD08.fr
=> Restoring Logs                 : DONE
=> Restoring Addons               : DONE
=> Restoring custom modules data   : SKIPPED
No custom modules data to restore in /tmp/shinken-backups/2021-04-13__12-18-24__V02.08.02-004_BUILD08.fr

Sanatizing your restored configuration for this version of Shinken
fix_double_sync_keys           : skip (unnecessary)
fix_default_item_se_uuid       : skip (unnecessary)
fix_remove_shinken_core        : skip (unnecessary)
fix_remove_deprecated_check    : skip (unnecessary)
fix_remove_undefined_aix_templates : skip (unnecessary)
fix_flapping_thresholds        : skip (unnecessary)
fix_business_impact            : skip (unnecessary)
fix_host_name_inheritance      : skip (unnecessary)
fix_bp_rule_args               : skip (unnecessary)
fix_uuid_in_sla_info           : skip (unnecessary)
fix_synchronizer_install_missing_addons : skip (unnecessary)
fix_missing_skeletons          : skip (unnecessary)
fix_missing_sources            : executed [OK]
fix_update_protected_fields_parameters : skip (unnecessary)
fix_all_move_overload_files    : skip (unnecessary)
remove_hostgroups_in_working_area : skip (unnecessary)

```

Restoration done. You can restart your Shinken with `service-shinken start`

Pour l'instant, la rétention du Scheduler ne peut se sauvegarder et se restaurer que manuellement (voir la page [Sauvegarde et restauration manuelle de la rétention](#)).

Lorsqu'une erreur arrive pendant le restore, on a un message d'erreur indiquant qu'une erreur est survenue dans une section en particulier.

Un log d'erreur est alors créé dans le dossier du backup utilisé par le script.

⚠ La partie floutée du chemin de la capture d'écran correspond au dossier que l'on donne en paramètre au shinken-restore.