

# Module event-manager-reader

## Sommaire

- Concept
- Activation du module
- Configuration
  - Exemple de fichier de configuration
  - Détails des sections composant le fichier de configuration
    - Identification du module
    - Accès à la base MongoDB
      - Configuration de l'URI de connexion et de l'authentification par mot de passe
      - Connexion directe au serveur MongoDB
      - Connexion par SSH au serveur MongoDB
    - Gestion de l'auto reconnexion avec un cluster MongoDB
    - Configuration pour l'interface de visualisation
    - Options Internes

## Concept

Le module de type **event-manager-reader** est un module qui permet au module WebUI ( *l'Interface de Visualisation* ) du Broker d'afficher le bac à événement ( voir la page [Vue - Les Événements](#) ).

- Il récupère les événements en base de donnée des événements.
- Il doit donc être listé dans la liste des modules de type **WebUI** ( voir la page [Module WebUI](#), propriété "modules" ).



Pour que le menu "bac à événement" apparaisse dans l'interface, il faut absolument que ce module soit activé.



Si vous avez plusieurs configurations du module WebUI présentes dans votre architecture, n'oubliez pas d'activer le module "event-manager-reader" dans chaque configuration.

Exemple :

- Plusieurs modules WebUI sur le même Broker.
- Un module WebUI différent sur différents Brokers qui n'utilise pas la même configuration.

## Activation du module

Le module **event-manager-reader** est un module qui peut être activé seulement sur le module WebUI.

- L'activation du module s'effectue en ajoutant le nom de ce module dans le fichier de configuration du module WebUI.
- Pour ce faire, il faut ouvrir le fichier de configuration du module WebUI du Broker à l'emplacement `/etc/shinken/module/ma_webui.cfg`, et ajouter le nom du module "event-manager-reader".

Exemple : par défaut, est livré un module dont le nom est "**event-manager-reader**":

```
define module {
    [...]
    modules          Module 1, Module 2, Module 3, event-manager-reader
    [...]
}
```

Pour prendre en compte le changement de configuration, redémarrer l'Arbiter:

```
service shinken-arbiter restart
```

## Configuration

La configuration du module livré par défaut se trouve dans le fichier `/etc/shinken/modules/event_manager_reader.cfg`

- Un fichier d'exemple est disponible ici : `/etc/shinken-user-example/configuration/daemons/brokers/modules/event_manager_reader/event_manager_reader-example.cfg`

## Exemple de fichier de configuration

```
# CFG_FORMAT_VERSION 1 ( SHINKEN : DON'T TOUCH THIS LINE )

#####
# event manager
#####
# Modules that can load this module:
# - WebUI (to display events data to the users)
# This module read event from database to show you in the WebUI
#####

define module {

    # #
    #     MODULE IDENTITY     #
    # #

    # Module name [ Must be unique ]                                [ MANDATORY ]
    #
    module_name                event-manager-reader

    # Module type [ Do not edit ]                                    [ MANDATORY ]
    #
    module_type                event_container

    # #
    #     DATABASE CONNECTION     #
    # #

    # MongoDB parameters     #

    # MongoDB uri definition . You can find the mongodb uri syntax at
    # https://docs.mongodb.com/manual/reference/connection-string/
    #
    #     Default : mongodb://localhost/?w=1&fsync=false
    #
    # uri                    mongodb://localhost/?w=1&fsync=false

    # Which database contains events data
    #
    #     Default : event_container
    #
    # database                event_container

    # username/password to authenticate to MongoDB.
    # Both parameters must be provided for authentication to function correctly.
    #
    # broker_module_webui_module_event_manager_reader_database_username

    #
    # broker_module_webui_module_event_manager_reader_database_password

    # SSH tunnel activation to secure your mongodb connection
    # That will allow all mongodb to be encrypted & authenticated with SSH
    #
    #     ...      : Enable => 1 ( enable ssh tunnel )
    #     Default : Disable => 0 ( disable ssh tunnel )
    #
    # use_ssh_tunnel          0

    # If the SSH connection goes wrong, then retry use_ssh_retry_failure time before_shinken_inactive
    #
```

```

#           Default : 1 ( try )
#
# use_ssh_retry_failure                               1

# SSH user to connect to the mongodb server.
#
#           Default : shinken
#
# ssh_user                                           shinken

# SSH keyfile to connect to the mongodb server.
#
#           Default : ~shinken/.ssh/id_rsa
#
# ssh_keyfile                                       ~shinken/.ssh/id_rsa

# SSH Timeout used to test if the SSH tunnel is viable or not, in seconds.
#
#           Default : 10 ( seconds )
#
# ssh_tunnel_timeout                                10

#   AutoReconnect Management   #

# When MongoDB require you to reconnect ( For example, It can occur when a new PRIMARY is elected
# in a MongoDB cluster ), it will raised the MongoDB AutoReconnect exception.
#
# How many try to reconnect before module go in error
#
#           Default : 4 ( try )
#
# auto_reconnect_max_try                            4

# Time between each try
#
#           Default : 3 ( seconds )
#
# auto_reconnect_sleep_between_try                  3

# NOTE: Change these values only if you have a MongoDB cluster and you change the
# heartbeatTimeoutSecs of your MongoDB replica set
# The value of auto_reconnect_max_try * auto_reconnect_sleep_between_try must be higher than
# heartbeatTimeoutSecs in the rs.conf(); of your MongoDB replica set.

# #
# VISUALISATION UI   #
# #

# Specify the max retries when the Visualisation UI can't communicate with the event_manager_reader
# during the Excel export.
# Increase this parameter when users encounter difficulties when generating the Excel export
# of events. This will make the export more tolerant to network loss or database slowness,
# or restart of the broker.
# The value cannot be 0.
#
#           Default : 2 ( try )
#
# broker__module_webui__module_event_manager_reader__events_export__max_retry 2

# #
# INTERNAL OPTIONS   #
# #

# INTERNAL : DO NOT EDIT FOLLOWING PARAMETER WITHOUT YOUR DEDICATED SUPPORT
# Broker idle time before considering that Shinken is inactive.
# Use this if you have Broker loop time that exceeds 30 seconds
#
#           Default : 30 ( seconds )
#
# time_before_shinken_inactive                       30

```

```
}
```

## Détails des sections composant le fichier de configuration

### Identification du module

Il est possible de définir plusieurs instances de module de type "event-manager-reader" dans l'architecture Shinken.

- Chaque instance devra avoir un nom unique.

Nom	Type	Unité	Défaut	Description
module_name	Texte	---	event-manager-reader	<b>Il est conseillé de choisir un nom en fonction de l'utilisation du module pour que la configuration soit simple à maintenir.</b> Doit être unique.
module_type	Texte	---	event_container	Ne peut être modifié.

### Accès à la base MongoDB

Cette configuration s'effectue dans le fichier de configuration du module.

Pour se connecter à la base MongoDB utilisé pour le stockage des données, deux méthodes sont disponibles :

- **Connexion directe** : Par défaut, mais non sécurisée.
- **Tunnel SSH** : Shinken se connecte à la base MongoDB au travers d'un module SSH pour plus de sécurité.

### Configuration de l'URI de connexion et de l'authentification par mot de passe

```
# #
# DATABASE CONNECTION #
# #

# MongoDB parameters #

# MongoDB uri definition . You can find the mongodb uri syntax at
# https://docs.mongodb.com/manual/reference/connection-string/
#
# Default : mongodb://localhost/?w=1&fsync=false
#
# uri mongodb://localhost/?w=1&fsync=false

# Which database contains events data
#
# Default : event_container
#
# database event_container

# username/password to authenticate to MongoDB.
# Both parameters must be provided for authentication to function correctly.
#
# broker__module_webui__module_event_manager_reader__database__username

#
# broker__module_webui__module_event_manager_reader__database__password
```

Nom	Type	Unité	Défaut	Description
-----	------	-------	--------	-------------

uri	Texte	URL	<b>mongodb://localhost/?safe=true</b>	La page suivante décrit la syntaxe de l'uri de MongoDB : <a href="https://docs.mongodb.com/manual/reference/connection-string/">https://docs.mongodb.com/manual/reference/connection-string/</a> .
database	Texte	---	<b>shinken</b>	Nom de la base de données où sont stockées les données événements.
broker__module_webui__module_event_manager_reader__database__username	Texte	---		Utilisateur pour l'authentification avec mot de passe à la base MongoDB.  Utile uniquement si l'activation par mot de passe a été activé ( voir la page <a href="#">MongoDB - activation de l'authentification par mot de passe</a> ).
broker__module_webui__module_event_manager_reader__database__password	Texte	---		Mot de passe de l'utilisateur utilisé pour l'authentification avec mot de passe à la base MongoDB.  Utile uniquement si l'activation par mot de passe a été activé ( voir la page <a href="#">MongoDB - activation de l'authentification par mot de passe</a> ).



### Connexion directe au serveur MongoDB

Par défaut, le module se connecte de manière directe à la base MongoDB pour y lire et écrire les données.

Dans la configuration du module, ceci correspond au paramètre "use\_ssh\_tunnel" à 0.

C'est la méthode de connexion par défaut lorsque la base est sur la même machine que le démon ( quand l'URL de la base est localhost ).

Si la base est sur une autre machine, il faudra alors se connecter à la base via un tunnel SSH. Cela permet à la base distante de rester en écoute réseau sur l'interface réseau local, ce qui la sécurise des accès extérieurs ( voir la page [Sécurisation des connexions aux bases MongoDB](#) ).

### Connexion par SSH au serveur MongoDB

```
# SSH tunnel activation to secure your mongodb connection
# That will allow all mongodb to be encrypted & authenticated with SSH
#
#     ...      : Enable => 1 ( enable ssh tunnel )
#     Default : Disable => 0 ( disable ssh tunnel )
#
# use_ssh_tunnel                                0

# If the SSH connection goes wrong, then retry use_ssh_retry_failure time before_shinken_inactive
#
#     Default : 1 ( try )
#
# use_ssh_retry_failure                          1

# SSH user to connect to the mongodb server.
#
#     Default : shinken
#
# ssh_user                                       shinken

# SSH keyfile to connect to the mongodb server.
#
#     Default : ~shinken/.ssh/id_rsa
#
# ssh_keyfile                                   ~shinken/.ssh/id_rsa

# SSH Timeout used to test if the SSH tunnel is viable or not, in seconds.
#
#     Default : 10 ( seconds )
#
# ssh_tunnel_timeout                            10
```

Nom	Type	Unité	Défaut	Description
-----	------	-------	--------	-------------

use_ssh_tunnel	Booléen	---	0	<ul style="list-style-type: none"> <li>• 1 : Connexion par tunnel SSH</li> <li>• 0 : Connexion directe</li> </ul>
use_ssh_retry_failures	Entier	Nombre d'essais	1	Spécifie le nombre supplémentaire de tentatives lors de l'établissement du tunnel SSH si ce dernier n'arrive pas à être établi
ssh_user	Texte	Utilisateur unix	shinken	L'utilisateur avec lequel le tunnel sera établi
ssh_keyfile	Texte	Chemin de fichier	~shinken/.ssh/id_rsa	La clé ssh privée présente sur le serveur Shinken qui sera utilisé pour établir le tunnel.
ssh_tunnel_timeout	Entier	Ssecondes	10	Spécifie le timeout en secondes de la vérification du tunnel SSH avant que la connexion vers MongoDB soit effectuée

## Gestion de l'auto reconnexion avec un cluster MongoDB

```
# AutoReconnect Management

# When MongoDB require you to reconnect ( For example, It can occur when a new PRIMARY is elected
# in a MongoDB cluster ), it will raised the MongoDB AutoReconnect exception.
#
# How many try to reconnect before module go in error
#
#     Default : 4 ( try )
#
# auto_reconnect_max_try                4

# Time between each try
#
#     Default : 3 ( seconds )
#
# auto_reconnect_sleep_between_try      3

# NOTE: Change these values only if you have a MongoDB cluster and you change the
#       heartbeatTimeoutSecs of your MongoDB replica set
#       The value of auto_reconnect_max_try * auto_reconnect_sleep_between_try must be higher than
#       heartbeatTimeoutSecs in the rs.conf(); of your MongoDB replica set.
```

### Définitions

- **Primaire**: nom de MongoDB pour désigner un serveur maître, le serveur sur lequel il est possible de faire des requêtes d'écriture dans la base.
- **Election** : processus de MongoDB pour choisir un nouveau membre Primaire si le membre Primaire devient inaccessible

( voir la page [Haute disponibilité de la base MongoDB \(mise en place d'un cluster\)](#) )

Dans le cas de l'utilisation d'un cluster MongoDB, lorsque le membre Primaire devient inaccessible, une nouvelle élection est déclenchée, ce qui provoque une coupure temporaire de l'accès à la base.

Dans le but de ne pas interrompre le service, le module "event-manager-reader" va se reconnecter automatiquement au cluster MongoDB. Pour ce faire, il va faire un nombre d'essais égaux au paramètre "auto\_reconnect\_max\_try" avec une pause de X secondes entre chaque essai ( correspondant au paramètre "auto\_reconnect\_sleep\_between\_try" ).

Par défaut pour MongoDB le temps maximum avant qu'un membre Primaire soit considéré comme indisponible et qu'une nouvelle élection ait lieu est de 10 secondes.

Voir : "**heartbeatTimeoutSecs**" donné par la commande rs.conf(); dans un shell de MongoDB.

Nom	Type	Unité	Défaut	Description
auto_reconnect_max_try	Entier	Nombre d'essais	4	Nombre d'essais de reconnexion à la base
auto_reconnect_sleep_between_try	Entier	Secondes	3	Temps entre chaque essai en seconde

Les valeurs par défauts du fichier laissent 12 secondes, ce qui est amplement suffisant avec la configuration par défaut de MongoDB.



Il est conseillé de ne pas modifier ces valeurs.

### Configuration pour l'interface de visualisation

```
# #
#     VISUALISATION UI     #
# #

# Specify the max retries when the Visualisation UI can't communicate with the event_manager_reader
# during the Excel export.
# Increase this parameter when users encounter difficulties when generating the Excel export
# of events. This will make the export more tolerant to network loss or database slowness,
# or restart of the broker.
# The value cannot be 0.
#
#     Default : 2 ( try )
#
# broker__module_webui__module_event_manager_reader__events_export__max_retry 2
```

Le paramètre "broker\_\_module\_webui\_\_module\_event\_manager\_reader\_\_events\_export\_\_max\_retry" permet de configurer le nombre maximal de tentatives lorsque l'interface de visualisation ne peut pas communiquer avec le module event\_manager\_reader pendant l'export des données vers Excel.

Cela rendra les exports plus tolérants aux pertes de réseau ou aux lenteurs de la base de données, ou aux redémarrages du Broker.

Nom	Type	Unité	Défaut	Description
broker__module_webui__module_event_manager_reader__events_export__max_retry	Entier	Nombre d'essais	2	Nombre d'essais d'appel pour l'export des données vers Excel. Ne peut pas valoir 0.

### Options Internes

```
# #
#   INTERNAL OPTIONS   #
# #

# INTERNAL : DO NOT EDIT FOLLOWING PARAMETER WITHOUT YOUR DEDICATED SUPPORT
# Broker idle time before considering that Shinken is inactive.
# Use this if you have Broker loop time that exceeds 30 seconds
#
#       Default : 30 ( seconds )
#
# time_before_shinken_inactive           30
```



Ces paramètres sont dédiés au fonctionnement interne au module, il est fortement recommandé de ne pas les modifier sans votre support dédié.

Nom	Type	Unité	Défaut	Description
time_before_shinken_inactive	Entier	Secondes	30	Temps d'inactivité du Broker avant de considérer que Shinken est inactif. Utilisez cette option si le temps de boucle du Broker dépasse 30 secondes.