

Module report-builder--module-sla-reader

Sommaire

Concept

Activation du module

Exemple d'activation du module nommé "report-builder--module-sla-reader" sur le module nommé broker--module-report-builder (configuration livrée par défaut par Shinken)

Créer un nouveau module de type "report-builder--module-sla-reader"

Configuration

Exemple de fichier de configuration

Détails des sections composant le fichier de configuration

Identification du module

Option de lecture du taux de SLA

Accès à la base MongoDB

Configuration des paramètres communs aux deux méthodes

Connexion directe au serveur MongoDB

Connexion par SSH au serveur MongoDB

Gestion de l'auto reconnexion avec un cluster MongoDB

Concept

Ce module permet de définir les paramètres d'accès aux données SLA pour la génération des rapports.

Il est possible, si vous avez plusieurs modules de type "broker__module_report_builder" d'avoir plusieurs configurations de ce module :

- Par exemple, un report_builder pour un royaume central et un, pour un sous-royaume, qui se connecteraient à des bases MongoDB différentes.



Les paramètres définissant le calcul du taux de disponibilité des SLA pour ce module doivent être les mêmes que ceux du module SLA attaché au Broker qui écrit les SLA pour le royaume.

Activation du module

Les modules de type "report_builder__module_sla_reader" sont des modules qui doivent être activés sur un module de type "broker__module_report_builder", qu'on appellera le **module parent**.

- L'activation du module s'effectue en ajoutant le **nom** de ce module dans le fichier de configuration du **module parent**.
 - Pour cela, ouvrir le fichier de configuration du **module parent** (de type "broker__module_report_builder"), et ajouter dans le paramètre **modules** le nom du module de type "report_builder__module_sla_reader".
- Il est possible de faire plusieurs modules de type "report_builder__module_sla_reader".
 - Cela permet, **par exemple**, d'avoir des configurations différentes en fonction des royaumes.
- S'il y a plusieurs modules de type "broker__module_report_builder" présents dans l'architecture, il ne faut pas oublier d'activer le module de type "report_builder__module_sla_reader" dans la configuration de chacun d'eux.
- **Contraintes** :
 - Activable uniquement sur un module de type "broker__module_report_builder" (voir la page [Module broker--module-report-builder](#))
 - Il ne peut y avoir qu'un seul module de type "report_builder__module_sla_reader" sur un module de type "broker__module_report_builder".

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

```
service-shinken-arbiter restart
```

Exemple d'activation du module nommé "report-builder--module-sla-reader" sur le module nommé broker--module-report-builder (configuration livrée par défaut par Shinken)

L'exemple suivant

- active le module "report-builder--module-sla-reader",
- sur le module "broker--module-report-builder", dont la configuration est dans le fichier **/etc/shinken/modules/broker--module-report-builder.cfg**.

Modification dans le fichier du module **/etc/shinken/modules/broker--module-report-builder.cfg**

```
define module {
    [...]
    modules          Module 1, Module 2, Module 3, report-builder--module-sla-reader
    [...]
}
```

Puis redémarrage de l'Arbiter

```
service-shinken-arbiter restart
```

Créer un nouveau module de type "report-builder--module-sla-reader"

Pour pouvoir configurer un module de type "report-builder--module-sla-reader", il faut faire un nouveau fichier de configuration grâce au fichier d'exemple fourni par défaut.

- Pour commencer, il faut choisir le nom du nouveau module.
 - Pour l'exemple, on l'appelle "Mon-Module-Report-Builder--Module-Sla-Reader".
 - Remplacer "Mon-Module-Report-Builder--Module-Sla-Reader" par le nom qui a été choisi.
- Puis il faut créer le fichier de configuration :
 - Copier le fichier de définition du module d'exemple **/etc/shinken-user-example/configuration/daemons/brokers/modules/broker__module_report_builder/modules/report_builder__module_sla_reader/report-builder--module-sla-reader-example.cfg** dans le répertoire de définition des modules **/etc/shinken/modules**.
(Exemple : **/etc/shinken/modules/report-builder--module-sla-reader__Mon-Module-Sla-Reader.cfg**)

```
cp /etc/shinken-user-example/configuration/daemons/brokers/modules
/broker__module_report_builder/modules/report_builder__module_sla_reader/report-builder--
module-sla-reader-example.cfg /etc/shinken/modules/report-builder--module-sla-reader__Mon-
Module-Sla-Reader.cfg
```

- Ensuite, il faut modifier le fichier nouvellement créé pour configurer le nouveau module
 - Il faut vérifier que le fichier appartienne à l'utilisateur shinken et qu'il possède le droit d'édition. Si ce n'est pas le cas, il faut effectuer les commandes suivantes :

```
chown -R shinken:shinken /etc/shinken/modules/report-builder--module-sla-reader__Mon-Module-
Sla-Reader.cfg
chmod u+w /etc/shinken/modules/report-builder--module-sla-reader__Mon-Module-Sla-Reader.cfg
```

- On change le nom du module en "Mon-Module-Report-Builder--Module-Sla-Reader" dans le fichier **/etc/shinken/modules/report-builder--module-sla-reader__Mon-Module-Sla-Reader.cfg**

```
...
# Module name [ Must be unique ]                                     [
MANDATORY ]
#
...
module_name                  Mon-Module-Report-Builder--Module-Sla-
Reader
...

```

- Ensuite, il faut ajouter le nouveau module dans le module de type "broker__module_report_builder" correspondant.
 - Dans notre exemple, on ajoute le module "Mon-Module-Report-Builder--Module-Sla-Reader" au module "broker__module_report_builder" définie dans le fichier **/etc/shinken/modules/broker--module-report-builder.cfg**

```

define module {
    [...]
    modules
        Module 1, Module 2, Module 3, Mon-Module-Report-Builder--
Module-Sla-Reader
    [...]
}

```

- Pour finir il faut redémarrer l'Arbiter pour que le Broker puisse prendre en compte ce nouveau mdule.

```
service-shinken-arbiter restart
```

Configuration

Par défaut, La configuration du module se trouve dans le fichier `/etc/shinken/modules/report-builder--module-sla-reader.cfg`

- Un exemple de configuration est également disponible dans `/etc/shinken-user-example/configuration/daemons/brokers/modules/broker__module_report_builder/modules/report_builder__module_sla_reader/report-builder--module-sla-reader-example.cfg`

Exemple de fichier de configuration

```

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

#=====
# report-builder--module-sla-reader
#=====
# Modules that can load this module:
# - broker--module-report-builder
# This module reads SLA data from a mongodb database
#=====

define module {

    # #
    #     MODULE IDENTITY     #
    # #

    # Module name [ Must be unique ]                               [ MANDATORY ]
    #
    module_name                                     report-builder--module-sla-reader

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

    # #
    #     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 sla data
    #
    #     Default : shinken
    #
    # database                                 shinken

    # username/password to authenticate to MongoDB.

```

```

# Both parameters must be provided for authentication to function correctly.
#
# database__username

#
# database__password

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

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

# #
# SLA CALCULATION      #
# #

# Some status can impact
# -> positively (counted as OK/UP)
# -> negatively (counted as CRITICAL/DOWN)
# -> not impact the SLA
# (is not counted, meaning the period of study is reduced by the period that is not counted).
# This configuration aims at giving Shinken administrators a way to configure

```

```

# how the SLA are calculated.

# SLA are computed on a daily basis.
# SLA of the current day are always recomputed after a configuration change.
# SLA from days before are by default not recomputed.
#
#         Default : 0 => Disable ( old SLA will not be recalculated )
#         ...      : 1 => Enable ( old SLA will be recomputed with current settings )
#
# recompute_old_sla                                0

# Warning periods
#
#         Default : 0 => Warning counts as DOWN
#         ...      : 1 => Warning counts as UP
#
# warning_counts_as_ok                             0

# Unknown periods
#
#         Default : include => "Unknown" status is counted negatively in the SLA.
#         ...      : exclude => "Unknown" are not counted from SLA considered period.
#         ...      : ok       => "Unknown" are considered as UP periods
#
# unknown_period                                  include

# No_data periods ( "Missing data" and "Shinken inactive" status )
#
#         Default : include => Only status is considered. "Missing data" and "Shinken inactive"
#         ...      : exclude => No_data are not counted from SLA considered period.
#         ...      : ok       => No_data are considered as UP periods.
#
# no_data_period                                  include

# Downtime periods
#
#         Default : include => Only status is considered.
#         ...      : exclude => Downtimes are not counted from SLA considered period.
#         ...      : ok       => Downtimes are considered as UP periods.
#         ...      : critical => Downtimes are considered as DOWN periods.
#
# downtime_period                                 include
}

```

Détails des sections composant le fichier de configuration

Identification du module

```

# #
# #   MODULE IDENTITY   #
# #
# Module name [ Must be unique ] [ MANDATORY ]
#
module_name                                report-builder--module-sla-reader

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

```

Il est possible de définir plusieurs instances de module de type "broker__module_report_builder__module_sla_reader" dans une architecture Shinken.

- Chaque instance devra avoir un nom unique.

Nom	Type	Unité	Défaut	Description
module_name	Texte	---	report-builder--module-sla-reader	Il est conseillé de choisir un nom en fonction de l'utilisation du module pour que la configuration soit simple à maintenir. Doit être unique.
module_type	Texte	---	report_builder__module_sla_reader	Ne peut être modifié.

Option de lecture du taux de SLA

```

# #
#   SLA CALCULATION   #
# #

# Some status can impact
#   -> positively (counted as OK/UP)
#   -> negatively (counted as CRITICAL/DOWN)
#   -> not impact the SLA
# (is not counted, meaning the period of study is reduced by the period that is not counted).
# This configuration aims at giving Shinken administrators a way to configure
# how the SLA are calculated.

# SLA are computed on a daily basis.
# SLA of the current day are always recomputed after a configuration change.
# SLA from days before are by default not recomputed.
#
#   Default : 0 => Disable ( old SLA will not be recalculated )
#   ...     : 1 => Enable ( old SLA will be recomputed with current settings )
#
# recompute_old_sla                                0

# Warning periods
#
#   Default : 0 => Warning counts as DOWN
#   ...     : 1 => Warning counts as UP
#
# warning_counts_as_ok                             0

# Unknown periods
#
#   Default : include => "Unknown" status is counted negatively in the SLA.
#   ...     : exclude => "Unknown" are not counted from SLA considered period.
#   ...     : ok      => "Unknown" are considered as UP periods
#
# unknown_period                                  include

# No_data periods ( "Missing data" and "Shinken inactive" status )
#
#   Default : include => Only status is considered. "Missing data" and "Shinken inactive"
#   status are counted negatively in the SLA.
#   ...     : exclude => No_data are not counted from SLA considered period.
#   ...     : ok      => No_data are considered as UP periods.
#
# no_data_period                                  include

# Downtime periods
#
#   Default : include => Only status is considered.
#   ...     : exclude => Downtimes are not counted from SLA considered period.
#   ...     : ok      => Downtimes are considered as UP periods.
#   ...     : critical => Downtimes are considered as DOWN periods.
#
# downtime_period                                  include

```

Nom	Type	Unité	Défaut	Description
recompute_ol d_sla	Booléen	---	0	<ul style="list-style-type: none"> • 1 : Si la configuration a modifié les paramètres de calcul du taux de disponibilité, les SLA déjà archivés seront recalculés, puis réarchivés, lorsqu'ils seront consultés. • 0 : Les SLA déjà archivés sont lus tels quels et ne sont pas modifiés.
warning_coun ts_as_ok	Booléen	---	0	<ul style="list-style-type: none"> • 1 : On considère que le service est toujours rendu même de manière potentiellement dégradée. • 0 : On considère que si le service n'est pas rendu de manière optimale, il n'est pas bien rendu et donc fait baisser le taux de disponibilité.
unknown_peri od	String	---	include	<ul style="list-style-type: none"> • ok : On considère que le service a donné un état donc le service est encore rendu même de manière potentiellement dégradée. • include : On considère que si l'on ne peut savoir si le service est rendu c'est qu'il n'est pas rendu. • exclude : On considère l'état trop imprécis pour modifier le taux de disponibilité.
no_data_peri od	String	---	include	<ul style="list-style-type: none"> • ok : On considère que même si la supervision ne l'a pas confirmé, le service est rendu. • include : On considère que si l'on ne peut pas savoir si le service est rendu, c'est qu'il n'est pas rendu. • exclude : On considère que l'état de la supervision (<i>de Shinken</i>) n'impacte pas le taux de disponibilité.
downtime_per iod	String	---	include	<ul style="list-style-type: none"> • ok : On considère que les périodes de maintenance planifiées font partie du service, donc le service est rendu . • critical : On considère que lors d'une maintenance planifiée, le service n'est plus rendu. • include : On considère que la maintenance est planifiée et donc que cette période n'impacte pas le service. • exclude : On considère que seul l'état du service compte pour le taux de disponibilité.



Plus de détails sur ces paramètres et sur le fonctionnement des SLA sont disponibles sur cette page : [Calcul du taux de SLA](#)

Accès à la base MongoDB

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

Pour se connecter à la base MongoDB utilisée pour le stockage des données, 2 méthodes sont disponibles :

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

Configuration des paramètres communs aux deux méthodes

```

# #
# 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 sla data
#
# Default : shinken
#
# database shinken

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

#
# database__password

```

Nom	Type	Unité	Défaut	Description
uri	Texte	URL	mongodb://localhost/?w=1&fsync=false	La syntaxe des URI de MongoDB est disponible à l'adresse https://docs.mongodb.com/manual/reference/connection-string/
database	Texte	---	shinken	Nom de la base de données où sont stockés les données SLA
database__username	Texte	---		Dans le cas où une identification par mot de passe est utilisée (voir la page : MongoDB - activation de l'authentification par mot de passe) : <ul style="list-style-type: none"> nom de l'utilisateur pour se connecter à la base de données. Laisser ce paramètre vide sinon.
database__password	Texte	---		Dans le cas où une identification par mot de passe est utilisée (voir la page : MongoDB - activation de l'authentification par mot de passe) : <ul style="list-style-type: none"> mot de passe pour se connecter à la base de données. Laisser ce paramètre vide sinon.

Connexion directe au serveur MongoDB

Par défaut, le module se connecte de manière directe à la base MongoDB, avec les paramètres communs listés ci-dessus, et le paramètre "use_ssh_tunnel" défini à 0.

Connexion par SSH 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](#)).

```

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

# If the SSH connection goes wrong, then retry use_ssh_retry_failure time before_shinken_inactive
#
#         Default : 1 ( number of retry )
#
# 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"> 1 : Connexion par tunnel SSH. 0 : Connexion directe.
use_ssh_retry_failure	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 utilisée pour établir le tunnel.
ssh_tunnel_timeout	Entier	secondes	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 ( number of 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 : terme de MongoDB pour désigner un serveur maître, le serveur sur lequel il est possible de faire des requêtes d'écriture dans la base.

Élection : processus de MongoDB pour choisir un nouveau membre Primaire si le membre Primaire actuel devient inaccessible.

Voir : [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 SLA va se reconnecter automatiquement au cluster MongoDB. Il va faire un nombre d'essais égal 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 ne 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	essais	4	Nombre d'essais de reconnexion à la base
auto_reconnect_sleep_between_try	Entier	secondes	3	Temps entre chaque essai en secondes

Les valeurs par défaut 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.