

# Comment interpréter les informations et les erreurs courantes du Healthcheck

## Sommaire

- Le démon est configuré avec l'adresse "localhost"
- Le démon est injoignable
- Le démon est en cours d'arrêt
- Le démon n'a pas encore reçu sa configuration de l'Arbiter
- Le démon a chargé une configuration enregistrée
- Le démon et son Arbiter ont des versions différentes
- Conflit d'Arbiter sur un démon
- La dernière connexion de l'Arbiter remonte à trop longtemps
- La configuration de l'Arbiter n'a pas été trouvée
- Les serveurs ne sont pas à la même heure
- Un démon est bloqué et doit être redémarré
- Un démon est configuré en tant que Spare
  - Un démon master a un démon spare de désigné ( seulement pour les Brokers )
  - Un démon master autorise son spare à ne pas avoir la même liste de modules
  - Un démon master n'a pas de démon spare de désigné ( seulement pour les Brokers )
  - Un démon spare qui n'a pas de démon master ( seulement pour les Brokers )
- Informations et détections dans le cadre de virtualisation Vmware
  - Détection de la présence des "VMWare tools"
    - RHEL / CentOS 7
    - RHEL / Alma / Rocky 8 et 9
    - Debian 13
  - Détection d'un pourcentage élevé de "CPU Stolen"
  - Le démon Gatherer n'est pas disponible
- Un module a redémarré de manière imprévue
- Erreur de communication entre les démons
- Erreur d'encodage dans les fichiers de configuration d'un démon
- Un scheduler n'a pas de broker ou de poller ou de réagir
- Un module a mis trop de temps à répondre aux demandes d'informations/statistiques
- Erreurs de cohérence de configuration entre une WebUI et ses modules de rapports
  - Cas 1 : Un des modules de génération de rapport n'est pas joignable
  - Cas 2 : La WebUI ne peut pas s'authentifier, avec le jeton configuré, a au moins un des modules de génération des rapports configurés
  - Cas 3 : Au moins un des modules de génération des rapports configurés n'appartient pas à la bonne architecture Shinken ( il dépend d'un autre Arbiter ) .
  - Cas 4 : Au moins un des modules de génération des rapports configurés n'est pas dans le bon royaume.
  - Cas 5 : Au moins un des modules de génération des rapports configurés n'utilise pas les paramètres de calcul pour les SLA de l'Interface de Visualisation.
- Erreurs de configuration concernant les données de météorologie
  - Les erreurs d'écriture
    - Cas 1 : Les données d'un royaume ne sont pas sauvegardées
    - Cas 2 : Plusieurs brokers du même royaume écrivent dans une même base graphite
    - Cas 3 : Le module graphite sauvegarde uniquement des royaumes différents de celui du broker
    - Cas 4 : Le paramètre realm\_store\_only contient un royaume qui n'existe pas
    - Cas 5 : Le port de Graphite-Perfdata est invalide
    - Cas 6 : Le nom de l'hôte de Graphite-Perfdata est invalide
  - Les erreurs de lecture
    - Cas 1 : La configuration du module webui.cfg précise un royaume qui n'existe pas
    - Cas 2 : La configuration du module webui.cfg précise un royaume non géré par le broker
    - Cas 3 : Le serveur graphite précisé dans la configuration de webui ne gère pas les données de ce royaume
    - Cas 4 : Le serveur graphite précisé dans la configuration de webui n'est pas un serveur graphite connu
    - Cas 5 : Le broker gère des royaumes qui ne sont pas configurés dans les graphite\_backends de la webui
    - Cas 6 : Le broker rencontre des erreurs pour lire les données sur les serveurs graphite
    - Cas 7 : Le port de Graphite n'est pas correct
    - Cas 8 : L'adresse locale des serveurs de lecture ou d'écriture de Graphite n'a pas pu être résolu
- Erreurs de format dans le paramètre graphite\_backends de la WebUI
  - Cas 1 : Pas de royaume dans le backend
  - Cas 2 : Trop de séparateur de royaume
  - Cas 3 : Pas de protocole
  - Cas 4 : Pas de port HTTP
  - Cas 5 : Pas de nom d'hôte ou d'adresse IP
- Erreurs dans le contenu de la valeur de la clé graphite\_backend
  - Cas 1 : Port invalide
  - Cas 2 : Protocole non supporté
- Le démon a bloqué une tentative de chargement d'objet malveillant

Le healthcheck peut afficher de nombreux cas d'erreur différents. Aussi, certaines informations importantes sont présentées régulièrement.

Pour faciliter l'utilisation du Healthcheck, cette section présente ces différentes informations et erreurs courantes, accompagnées de leur interprétation.

## Le démon est configuré avec l'adresse "localhost"

Puisque les démons constituant une installation de Shinken Entreprise ont pour vocation à être utilisés dans une architecture distribuée, il ne faut pas définir les démons avec des adresses locales. Les démons vont devoir communiquer entre eux, il faut donc les configurer avec une adresse précise pour éviter tout problème dans une architecture distribuée.

Cette erreur est très fréquemment rencontrée dans le healthcheck puisque dans l'installation par défaut de Shinken Entreprise, les démons utilisent l'adresse localhost. Dès que des démons seront utilisés dans la même configuration sur plusieurs machines différentes, il est impératif d'utiliser des adresses précises.

```
- localhost (127.0.0.1):
*****
[scheduler: scheduler-vm3]
  AT RISK: scheduler-vm3 is defined with the localhost address. It is a problem in distributed mode. Please configure it with the LAN IP/FQDN address instead
  OK:      Connection to daemon is OK at port 7768
  OK:      Daemon version is: v 3.08.02-003.02.7e
  OK:      Correct connection from arbiter "arbiter-vm3" ( and no time shift )
Modules:
  OK:      Name: MongodbRetention      Type: mongodb_retention
```

## Le démon est injoignable

Le Scheduler "scheduler-vm3" est injoignable.

Dans ce cas, on exécute le healthcheck sur la machine vm3. On peut donc déduire que le démon est éteint.

Sur un démon hébergé sur une machine distante, il faudra d'abord déterminer si la machine est joignable avant de pouvoir affirmer que le démon est éteint. Un healthcheck local sur la machine hébergeant le démon peut confirmer cette hypothèse.

```
[scheduler: scheduler-vm3]
  ERROR:   Cannot contact daemon vm3:7768 ( [Errno 111] Connection refused )
  OK:      Configuration seems valid
```

## Le démon est en cours d'arrêt

Lorsque le démon est en cours d'arrêt, shinken-healthcheck le signale, et les informations relatives aux modules ne sont plus disponibles.

```
[ brokers ]
[broker: broker-master]
  AT RISK: The daemon is performing a shutdown.
  OK:      Configuration seems valid
  OK:      Connection to daemon is OK at port 7772
```

## Le démon n'a pas encore reçu sa configuration de l'Arbiter

Lorsque de l'on démarre ou redémarre un démon, celui-ci demande sa configuration à l'Arbiter. Il est possible que l'Arbiter mette du temps à l'envoyer.

Dans ce cas un message " AT RISK " sera affiché indiquant que le démon n'a pas encore été contacté par l'Arbiter.

```
[ brokers ]
[broker: broker-master]
  AT RISK: Daemon has not yet been contacted by an arbiter.
  OK:      Configuration seems valid
  OK:      Connection to daemon is OK at port 7772
  OK:      Daemon version is: v 3.08.02-003.02.7e
  INFO:    This daemon do not have any spare defined
```

## Le démon a chargé une configuration enregistrée

Lorsque le module *last-configuration-recorder* est actif, pour les démons de type :

- Broker,
- Poller,
- Reactionner,

- Receiver et
- Scheduler,

Au démarrage, le démon charge la précédente configuration qu'il a reçue de l'Arbiter, en attendant que ce dernier le contacte.

- Il permet un redémarrage du démon même si l'Arbiter ne peut pas joindre le démon ( *ex : coupure réseau* )
- Cela permet aussi de rendre le redémarrage du démon plus rapide ( *dans le cas où l'Arbiter n'a pas changé de version de configuration* ).

Un message " **AT RISK** " indique que le démon n'a pas encore été contacté par l'Arbiter

```
[ brokers ]
[broker: broker-master]
OK: Configuration seems valid
OK: Connection to broker-master is OK at port 7772
OK: Daemon version is: 02.04.01-003_BUILD04.fr
INFO: This daemon do not have any spare defined
Arbiters connection:
1.Architecture : Shinken-Production [loaded from retention] :
AT RISK: Arbiter arbiter-master : loaded from retention 23s ago [waiting for Arbiter connection]
```

## Le démon et son Arbiter ont des versions différentes

Pour garantir un bon fonctionnement, il faut que tous les démons présents dans la configuration possèdent la même version de Shinken Entreprise.

Dans ce cas, le Poller "poller-domtom" et l'Arbiter n'ont pas la même version installée, ce qui peut provoquer des dysfonctionnements.

```
[poller: poller-domtom]
ERROR: Daemon version is 02.04.01-003_BUILD04.fr and arbiter version is 02.04.01-003_BUILD05.fr. Please update
OK: Configuration seems valid
OK: Connection to daemon is OK at port 7771
OK: Correct connection from arbiter "arbiter-vm3" ( and no time shift )
Tag list:
None [This poller will handle all untagged checks (the "None" tag)]
Talk to:
```

## Conflit d'Arbiter sur un démon

Lors d'une mauvaise configuration, il arrive que 2 installations différentes de Shinken Entreprise pointent vers un même démon. Chaque Arbiter va détecter des incohérences, et ils vont constamment renvoyer des configurations ( *contradictoires entre elles* ) afin de les corriger, ce qui fait que le démon ne va jamais réussir à travailler correctement.

Dans ce cas, une erreur est affichée dans le shinken-healthcheck indiquant le conflit d'Arbiter.

Pour chaque Arbiter, est donné son nom, son uuid, son adresse IP ainsi que de quelle installation Shinken ( *Architecture* ) il provient, afin de les retrouver et de corriger immédiatement l'erreur de configuration.

```
ERROR: Arbiters CONFLICT :
1.Architecture : Production [loaded from retention] :
ERROR: Arbiter arbiter-master : last connection 8s ago. Defined on the server with uuid d2a358b0-cae3-4ce8-bc3f-edd241627bac (/var/lib/shinken/server.uuid)
2.Architecture : Testing [loaded from retention] :
ERROR: Arbiter arbiter-master : last connection 14s ago. Defined on the server with uuid 785e0227-455a-449d-b8f1-516897000e3b (/var/lib/shinken/server.uuid)
```

## La dernière connexion de l'Arbiter remonte à trop longtemps

Si la dernière connexion de l'Arbiter remonte à trop de temps, le démon va lever un **AT RISK** . Ceci peut être dû :

- Les Arbiters MASTER et SPARE sont réellement éteints
- Les Arbiter MASTER et SPARE sont en train d'envoyer des configurations à d'autres démons, et ne peuvent donc pas contacter ce démon pour l'instant.

Le temps pris en compte comme limite de dernière connexion est de  $check\_interval * max\_check\_attempts$  du démon ( *définis dans sa configuration* ). Les valeurs par défauts sont de  $60s * 3$ , soit 3 minutes.

```
Arbiters connection
Architecture DEV-01-01-01-01
AT RISK: Arbiter arbiter-master: Missed connection from arbiter since 17m 54s ( > daemon check_interval * max_check_attempts)
```

## La configuration de l'Arbiter n'a pas été trouvé

Ce message apparaît quand la commande healthcheck utilise l'option --locale ou -l et que la propriété `host_name` n'a été renseignée dans aucun fichier de configuration de ou des Arbiters(s).

Pour résoudre ce problème, il faut dans le fichier de configuration de l'Arbiter qui est présent sur cette machine (situé dans `/etc/shinken/arbiters/`), changer la propriété `host_name` avec le nom de la machine ( *qui est trouvable grâce à la commande `hostname`* )

```
[ arbiters ]
[arbiter: --]
ERROR: No configuration for this arbiter was found. Check if host name in arbiter configuration has the machine host name
```

## Les serveurs ne sont pas à la même heure

Si le serveur n'est pas à la même heure que le serveur Arbiter ( *qui fait office de référence* ), une erreur **CRITICAL** sera levée, car des temps différents sur les différents serveurs vont avoir des effets **désastreux** sur la cohérence des données de supervision.

```
Arbiters connection
Architecture DEV-
ERROR: Arbiter arbiter-spare: server times are different, time shift of 1 days 2h
```

## Un démon est bloqué et doit être redémarré

Si un démon arrive dans un état bloqué à cause d'un bug par exemple, il ne fonctionnera plus correctement et doit être redémarré. L'erreur suivante est alors remontée :

```
ERROR: The daemon have a lock, it's not working and MUST be restarted. Please contact your support to analyse the daemon lock
"Main loop" was locked more than 3600s, and detected at 2022-01-13 10:58:48 [ WATCH DOG ]
```

## Un démon est configuré en tant que Spare

Un démon peut être configuré en tant que Spare. Un démon Spare est par défaut inactif, et devient actif lorsqu'un démon principal n'est plus disponible, afin d'assurer une continuité du service.

Pour pouvoir identifier facilement les démons Spare dans l'architecture, une mention "SPARE" est présente à côté du nom du démon.

Si le démon prend la main pour assurer la continuité de service, alors une mention additionnelle "RUNNING" est affichée.

Dans le cas d'un démon Broker, une ligne supplémentaire va être affichée afin d'indiquer de quel démon master, il est le spare.

```
[broker: broker-spare] (SPARE)
OK: We are the SPARE of the master daemon => broker-master
OK: Configuration seems valid
OK: Connection to daemon is OK at port 9772
OK: Daemon version is:
OK: Correct connection from arbiter "arbiter-master" ( and no time shift )
```

```
[broker: broker-spare] (SPARE) (RUNNING)
OK: We are the SPARE of the master daemon => broker-master
OK: Configuration seems valid
OK: Connection to daemon is OK at port 9772
OK: Daemon version is:
OK: Correct connection from arbiter "arbiter-master" ( and no time shift )
```

## Un démon master a un démon spare de désigné ( seulement pour les Brokers )

Quand un démon master a un démon spare de désigné ( *via sa propriété `spare_daemon`* ) alors, il sera listé.

```
[broker: broker-master]
OK:      Our SPARE => broker-spare
OK:      Configuration seems valid
OK:      Connection to daemon is OK at port 7772
OK:      Daemon version is: 4.0.0-rc1
OK:      Correct connection from arbiter "arbiter-master" ( and no time shift )
```

## Un démon master autorise son spare à ne pas avoir la même liste de modules

Quand un master autorise son spare à ne pas avoir la même liste de modules, via la propriété "broker\_\_manage\_spare\_\_spare\_must\_have\_the\_same\_list\_of\_module\_type"

le texte entre parenthèses le précise ( voir la [Le Broker](#) )

```
[ brokers ]
[broker: broker-master]
OK:      My SPARE => broker-spare ( the broker configuration doesn't require the same list of modules on its spare )
OK:      Configuration seems valid
OK:      Connection to daemon is OK at port 7772
OK:      Daemon version is: 4.0.0-rc1
OK:      Correct connection from arbiter "arbiter-master" ( and no time shift )
```

## Un démon master n'a pas de démon spare de désigné ( seulement pour les Brokers )

Quand un démon master n'a pas de démon spare, ceci sera affiché.

```
[broker: broker-secondary]
OK:      Configuration seems valid
OK:      Connection to daemon is OK at port 8772
OK:      Daemon version is: 4.0.0-rc1
OK:      Correct connection from arbiter "arbiter-master" ( and no time shift )
INFO:    This daemon do not have any spare defined
```

## Un démon spare qui n'a pas de démon master ( seulement pour les Brokers )

Quand un démon spare n'est le spare d'aucun autre démon, il est noté comme inutilisé.

```
[broker: broker-spare-useless] (SPARE)
ERROR:   No master is using this spare dameon => UNUSED
OK:      Configuration seems valid
OK:      Connection to daemon is OK at port 11772
OK:      Daemon version is: 4.0.0-rc1
OK:      Correct connection from arbiter "arbiter-master" ( and no time shift )
```

## Informations et détections dans le cadre de virtualisation Vmware

Il est possible de virtualiser l'architecture Shinken, ou simplement quelques-uns des démons, avec **VMWare**.

Si c'est le cas, une ligne sera ajoutée dans la sortie du "shinken-healthcheck" et précisera que le satellite fonctionne sur une architecture VMWare.

```
Local daemons
[broker]
OK:      Connection to daemon is OK at port 7772
OK:      Daemon version is: 4.0.0-rc1
OK:      This satellite runs on a VMWare VM
OK:      Correct connection from arbiter "arbiter-master" ( and no time shift )
```

## Détection de la présence des "VMWare tools"



```
service-shinken-gatherer restart
```

```
[scheduler: scheduler-master]
  AT RISK: The VMWare stats file '/dev/shm/vmware_stats_export.dat' seems to be too old (not update since 0d 2h 2m 49s > 60s)
The shinken-gatherer daemon should be stopped. Please restart it with 'service shinken-gatherer start'.
```

## Un module a redémarré de manière imprévue

Certains démons de Shinken Entreprise utilisent des modules qui permettent d'étendre leurs fonctionnalités. Il arrive que ces modules redémarrent pour des raisons diverses et variées.


Lorsqu'un module a redémarré dans les 2 dernières heures, un avertissement est affiché pour le module en question, car il peut s'agir d'un problème récurrent et/ou potentiellement gênant. Une analyse des logs peut permettre d'avoir plus d'informations sur les erreurs du module.

```
[broker: broker-master]
  OK: Configuration seems valid
  OK: Connection to broker-master is OK at port 7772
  OK: Daemon version is: 2.12.0
  INFO: This daemon does not have any spare defined
Arbiters connection:
  1.Architecture : [shinken-arbiters] [1.0.0] :
  OK: Arbiter arbiter-master : last connection 26s ago.
Modules:
  AT RISK: The module [WebUI] has restarted [5 times] in the last 24 hours.
  Last 3 restarts:
    - 2026-03-05 15:34:56 : The module WebUI has gone down unexpectedly!
    - 2026-03-05 15:40:06 : The module WebUI has gone down unexpectedly!
    - 2026-03-05 15:40:07 : The module WebUI has gone down unexpectedly!
  AT RISK: Name: WebUI
  Type: webui
```

## Erreur de communication entre les démons

Les différents démons de Shinken Entreprise communiquent entre eux pour fonctionner. Il arrive que des erreurs de communications soient remarquées et génèrent des erreurs, qui sont alors notifiées dans le Healthcheck dans la section du démon concerné.

Il est à noter que ces erreurs ne sont pas forcément fatales au bon fonctionnement de Shinken Entreprise, mais qu'elles nécessitent une attention particulière.

 Les erreurs de communication peuvent ne pas être graves et ne pas avoir d'incidence sur l'architecture Shinken. Cependant, s'il y a des erreurs, en cas de doutes sur l'origine de ces problèmes de communication, par prévention, envoyer le log pour analyse au support Shinken.

```
[poller: poller-vm3]
  ERROR: Some API calls between daemons failed in the last 24 hours (2 errors). Please look at your daemon logs for more details about these errors.
  OK: Configuration seems valid
  OK: Connection to daemon is OK at port 7771
  OK: Daemon version is: 2.12.0
  OK: Correct connection from arbiter "arbiter-vm3" ( and no time shift )
Tag list:
  None [This poller will handle all untagged checks (the "None" tag)]
Talk to:
  OK: Reachable scheduler satellite (scheduler-vm3) at http://vm3:7768
```

## Erreur d'encodage dans les fichiers de configuration d'un démon

Si les fichiers de configuration des démons situés dans `/etc/shinken` ne pas encoder en utf-8.

Dans l'exemple ci-contre le fichier `arbiter-master.cfg` ne respecte pas le bon encodage ( `utf-8` ) et apparaît en erreur dans le résultat de la commande `shinken-healthcheck`.

**Correction** : Il suffit de réencoder le fichier au format utf-8

```
[ arbiters ]
[arbiter: arbiter-master]
  ERROR: Some characters could not be read in utf-8 in these files : /etc/shinken/arbiters/arbiter-master.cfg
  OK: Configuration seems valid
  OK: Connection to daemon is OK at port 7770
```

## Un scheduler n'a pas de broker ou de poller ou de reactionner

Les Schedulers produisent des jobs à effectuer pour les Pollers/Reactionners, ainsi que des données pour les brokers ( *par exemple pour l'interface de visualisation* ). Ils ont donc besoin de ces autres démons pour fonctionner.

En cas de mauvaise configuration de démons et/ou de royaumes, un Scheduler peut ne pas être défini comme cible d'un autre démon ( *Poller/Reactionner/Broker* ).



Si le Scheduler est dans un sous royaume et qu'un Poller/Reactionner/broker est présent dans un royaume supérieur, ceci signifie que ce dernier n'a pas l'option `manage_sub_realms` activée.

```
- localhost (192.168.56.102):
*****
[ schedulers ]
[scheduler: scheduler-master]
ERROR: The scheduler has no reactionner in its realm (or upper realm with the manage_sub_realms option enabled)
OK: Configuration seems valid
OK: Connection to daemon is OK at port 7768
OK: Daemon version is: 1.1.1
OK: Correct connection from arbiter "arbiter-master" ( and no time shift )
Modules:
OK: Name: PickleRetentionFile Type: pickle_retention_file
Talk to:
OK: Reachable reactionner satellite (reactionner-master) at http://localhost:7769
OK: Reachable scheduler satellite (scheduler-master) at http://192.168.56.102:7768
```

## Un module a mis trop de temps à répondre aux demandes d'informations/statistiques

En cas de surcharge d'un module, ce dernier peut mettre trop de temps pour répondre aux demandes d'informations ( *2 secondes, retentées une fois* ), ce qui va faire une erreur au niveau du module.

```
Modules:
OK: Name: sla Type: sla
Module WebUI:
ERROR: [MODULE-INFO ] Fail to send command call get_module_info for module WebUI because the module did timeout (2s)
OK: Connection to WebUI is OK at port 7767
OK: Auth_secret is a custom variable
OK: Mongodb server is available at: mongodb://localhost/?safe=true
```

## Erreurs de cohérence de configuration entre une WebUI et ses modules de rapports

Le module `webui__module_report_handler` vérifie que les modules de génération de rapport ( `broker__module_report_builder` ) configurés sont accessibles et utilisables par la WebUI pour générer un rapport.

### Cas 1 : Un des modules de génération de rapport n'est pas joignable

```
Module WebUI:
OK: Connection to WebUI is OK at port 7767
OK: Auth_secret is a custom variable
Submodules:
AT RISK: Name: webui--module-report-handler Type: webui__module_report_handler => Some report modules cannot
reached.
OK: Name: Mongodb Type: mongodb
OK: Name: webui-enterprise Type: webui_enterprise
OK: Name: sla Type: sla
OK: Name: Cfg_password Type: cfg_password_webui
OK: Name: event-manager-reader Type: event_container
OK: Name: webui-module-service-weather Type: webui__module_service_weather
Module Mongodb:
OK: Connect to MongoDB with address : [192.168.1.51]
Module webui--module-report-handler:
Configuration issues:
( unknown:http://localhost:24106 )
ERROR: The module is unreachable.
broker-master::broker--module-report-builder ( uri=http://localhost:24100 )
OK: The configuration is correct.
Talk to:
ERROR: Unreachable target at http://localhost:24106 (Connection error to http://localhost:24106 : [E
no 7] Failed to connect to localhost port 24106 after 0 ms: Could not connect to server)
OK: Reachable target (broker-master::broker--module-report-builder) at http://localhost:24100
```

### Cas 2 : La WebUI ne peut pas s'authentifier, avec le jeton configuré, a au moins un des modules de génération des rapports configurés

```

Module WebUI:
  OK: Connection to WebUI is OK at port 7767
  OK: Auth_secret is a custom variable
  Submodules:
    AT RISK: Name: webui--module-report-handler Type: webui__module_report_handler => Some report modules cannot
reached.
    OK: Name: Mongoddb Type: mongoddb
    OK: Name: webui-enterprise Type: webui_enterprise
    OK: Name: sla Type: sla
    OK: Name: Cfg_password Type: cfg_password_webui
    OK: Name: event-manager-reader Type: event_container
    OK: Name: webui-module-service-weather Type: webui_module_service_weather
  Module Mongoddb:
    OK: Connect to MongoDB with address : [192.168.1.51]
  Module webui--module-report-handler:
  Configuration issues:
    broker-master::broker--module-report-builder ( uri=http://localhost:24100 )
    OK: The configuration is correct.
    broker-master::broker--module-report-builder-2 ( uri=http://localhost:24101 )
    ERROR: The module is unreachable.
  Talk to:
    ERROR: Unreachable target (broker-master::broker--module-report-builder-2) at http://localhost:24101
[403] Authentication denied. Check the configuration of the calling module : WebUI::webui--module-report-handler
    OK: Reachable target (broker-master::broker--module-report-builder) at http://localhost:24100

```

**Cas 3 : Au moins un des modules de génération des rapports configurés n'appartient pas à la bonne architecture Shinken ( *il dépend d'un autre Arbitre* ).**

```

Module WebUI:
  OK: Connection to WebUI is OK at port 7767
  OK: Auth_secret is a custom variable
  Submodules:
    ERROR: Name: webui--module-report-handler Type: webui__module_report_handler => Some report modules cannot b
reached.
    OK: Name: Mongoddb Type: mongoddb
    OK: Name: webui-enterprise Type: webui_enterprise
    OK: Name: sla Type: sla
    OK: Name: Cfg_password Type: cfg_password_webui
    OK: Name: event-manager-reader Type: event_container
    OK: Name: webui-module-service-weather Type: webui_module_service_weather
  Module Mongoddb:
    OK: Connect to MongoDB with address : [127.0.1.1]
  Module webui--module-report-handler:
  Configuration issues:
    broker-master::broker--module-report-builder ( uri=http://127.0.0.1:24100 )
    OK: The configuration is correct.
    broker-master::broker--module-report-builder ( uri=http://172.16.0.180:24100 )
    ERROR: Monitoring configuration incarnation inconsistency between the WebUI and the report builder
    ERROR: |- From the WebUI: Monitoring incarnation [1c83f6f0e3594ea094dc6b0c6c9031a6] given by ar
er: [arbiter-master/bbdc4856-04e4-ff48-a02d-b027d7b78fc9] at: [2025-11-13 15:27:48]
    ERROR: |- From the report builder: Monitoring incarnation [261dca7914994df99b14e17722a942f2] gi
by arbiter: [arbiter-master/517b8587-d5ec-6c4b-8562-5dc05a601f3b] at: [2025-11-13 14:49:21]
    broker-sub::broker--module-report-builder-sub ( uri=http://127.0.0.1:34100 )
    OK: The configuration is correct.
  Talk to:
    OK: Reachable target (broker-master::broker--module-report-builder) at http://172.16.0.180:24100
    OK: Reachable target (broker-master::broker--module-report-builder) at http://127.0.0.1:24100
    OK: Reachable target (broker-sub::broker--module-report-builder-sub) at http://127.0.0.1:34100

```

**Cas 4 : Au moins un des modules de génération des rapports configurés n'est pas dans le bon royaume.**

```

Module WebUI:
  OK: Connection to WebUI is OK at port 7767
  OK: Auth_secret is a custom variable
  Submodules:
    ERROR: Name: webui--module-report-handler Type: webui__module_report_handler => Some report modules cannot be reached.
    OK: Name: Mongoddb Type: mongoddb
    OK: Name: webui-enterprise Type: webui_enterprise
    OK: Name: sla Type: sla
    OK: Name: Cfg_password Type: cfg_password_webui
    OK: Name: event-manager-reader Type: event_container
    OK: Name: webui-module-service-weather Type: webui_module_service_weather
  Module Mongoddb:
    OK: Connect to MongoDB with address : [127.0.1.1]
  Module webui--module-report-handler:
  Configuration issues:
    broker-master::broker--module-report-builder ( uri=http://127.0.0.1:24100 )
    OK: The configuration is correct.
    broker-sub::broker--module-report-builder-sub ( uri=http://127.0.0.1:34100 )
    ERROR: Configuration inconsistency between the WebUI and the report builder.
    ERROR: |- The realm [ "All" ] is not managed by the report builder.
  Talk to:
    OK: Reachable target (broker-master::broker--module-report-builder) at http://127.0.0.1:24100
    OK: Reachable target (broker-sub::broker--module-report-builder-sub) at http://127.0.0.1:34100

```

## Cas 5 : Au moins un des modules de génération des rapports configurés n'utilise pas les paramètres de calcul pour les SLA de l'Interface de Visualisation.

```
Module WebUI:
  OK: Connection to WebUI is OK at port 7767
  OK: Auth_secret is a custom variable
  Submodules:
    ERROR: Name: webui--module-report-handler Type: webui_module_report_handler => No report module can be read, reports will not be available in WebUI.
    OK: Name: Mongoddb Type: mongoddb
    OK: Name: webui-enterprise Type: webui_enterprise
    OK: Name: sla Type: sla
    OK: Name: Cfg_password Type: cfg_password_webui
    OK: Name: event-manager-reader Type: event_container
    OK: Name: webui-module-service-weather Type: webui_module_service_weather
Module Mongoddb:
  OK: Connect to MongoDB with address : [192.168.1.51]
Module webui--module-report-handler:
  Configuration issues:
    broker-master::broker--module-report-builder ( uri=http://localhost:24100 )
    ERROR: SLA computation inconsistency on the WebUI [WebUI]:
    ERROR: | Computation options are different between the module [sla] of the WebUI and the module
report-builder--module-sla-reader] of the report builder
    ERROR: | - downtime_period option:
    ERROR: | --- WebUI::sla: [include]
    ERROR: | --- report-builder--module-sla-reader: [ok]
    ERROR: | - no_data_period option:
    ERROR: | --- WebUI::sla: [include]
    ERROR: | --- report-builder--module-sla-reader: [exclude]
    ERROR: | - recompute_old_sla option:
    ERROR: | --- WebUI::sla: [0]
    ERROR: | --- report-builder--module-sla-reader: [1]
    ERROR: | - unknown_period option:
    ERROR: | --- WebUI::sla: [include]
    ERROR: | --- report-builder--module-sla-reader: [exclude]
    ERROR: | - warning_counts_as_ok option:
    ERROR: | --- WebUI::sla: [0]
    ERROR: | --- report-builder--module-sla-reader: [1]
  Talk to:
  OK: Reachable target (broker-master::broker--module-report-builder) at http://localhost:24100
```

## Erreurs de configuration concernant les données de métrologie

Dans une configuration avec plusieurs royaumes, il est plus difficile de vérifier que les données de métrologie sont bien écrites et peuvent être lues pour chaque royaume.

Il existe 2 types d'erreurs :

- Les erreurs d'écriture
- Les erreurs de lecture

### Les erreurs d'écriture

#### Cas 1 : Les données d'un royaume ne sont pas sauvegardées

C'est le cas, si le message suivant apparaît :

```
[*] There is no broker configured to write data for this realm. This realm might produce data that are not stored anywhere (no Graphite backend configured)
```

**Exemple :** Un broker dans le royaume Toulouse qui n'a pas de module graphite, mais il a un module webui, il n'y a donc pas de broker qui sauvegarde les données du royaume **Toulouse**.

**Correction :** Ajouter le module Graphite-Perfdata sur le broker

```
Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      AT RISK: [ Toulouse ] There is no broker configured to write data for this realm. This realm might produce data that are not stored anywhere (no Graphite backend configured)
    OK: Realm [ France ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
```

#### Cas 2 : Plusieurs brokers du même royaume écrivent dans une même base graphite

C'est le cas, si le message suivant apparaît :

```
[*] This realm is stored on same backends by multiple brokers : by broker * on graphite *(*) and by broker * on graphite * (*)
```

Exemple : Deux brokers sur le même royaume ( *France* ) ont un même module graphite qui va écrire les données sur le même serveur graphite ( 192.168.1.23 )

Toutes les métriques sont reçues et gérées par le broker. Chaque broker écrira donc les données dans le serveur graphite. Il y aura donc des données en double sur le serveur graphite.

Correction : Un seul broker doit écrire les données d'un royaume sur un serveur graphite. Il faut donc :

- Enlever le module graphite\_perfdata d'un des deux brokers
- créer un deuxième module graphite\_perfdata qui écrira les données sur un serveur graphite différent. Afin de mettre en place de la haute disponibilité pour la base de métrologie; consulter la page [Haute disponibilité de la base de métrologie \(Graphite\)](#)

```
Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      ERROR: [ France ] This realm is stored on same backends by multiple brokers : by the module [ Graphite-Perfdata ] of the broker [ broker-
on graphite backend 192.168.1.23 and by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
```

### Cas 3 : Le module graphite sauvegarde uniquement des royaumes différents de celui du broker

```
Broker * has module* with realm [*] in parameter realm_store_only, but this broker handle only realms : *
```

Le broker à un module graphite\_perfdata qui sauvegarde les données uniquement pour un royaume qui est différent de celui ou ceux que le broker gère. Les royaumes gérés par le module sont cités dans le paramètre "realm\_store\_only". Si ce paramètre est commenté ou absent, le module prend en compte tous les royaumes que le broker gère.

**Correction** : Il faut changer le module de type graphite\_perfdata pour un autre module qui gère le/les mêmes royaumes que le broker. Il est possible de mettre un module graphite\_perfdata qui prend tous les royaumes ( *paramètre realm\_store\_only commenté* ).

```
Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      ERROR: Broker broke-master has module Graphite-Perfdata with realm [ Toulouse ] in parameter realm_store_only, but this broker handle only realm
Paris, Corse, Bordeaux, France
```

### Cas 4 : Le paramètre realm\_store\_only contient un royaume qui n'existe pas

```
Module * has realm [*] in parameter realm_store_only but this realm is unknown
```

Le paramètre "realm\_store\_only" contient un royaume qui n'est pas défini dans la configuration Shinken.

Correction : vérifier qu'il n'y a pas d'erreurs dans le ou les noms de royaumes qui sont définis dans le paramètre "realm\_store\_only" dans le module graphite\_perfdata cité dans la ligne d'erreur.

```
Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      ERROR: Module Graphite-Perfdata has realm [ Toulon ] in parameter realm store only but this realm is unknow
```

### Cas 5 : Le port de Graphite-Perfdata est invalide

```
The url of Graphite perfdata [ URL ] is invalid : The port [ HTTP_PORT ] is not valid. Valid values are integers from 0 to 65535.
```

Lors que le module Graphite-Perfdata est configuré avec un port qui n'est pas un port HTTP valide, l'erreur est remontée dans le shinken-healthcheck.

**Correction** : Changer le port de Graphite-Perfdata ( *par défaut dans /etc/shinken/modules/graphite.cfg* ) par un port HTTP valide.

```
Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      ERROR: The url of Graphite perfdata [ http://192.168.1.23:invalid ] is invalid : The port [ invalid ] is not valid
Valid values are integers from 0 to 65535.
```

## Cas 6 : Le nom de l'hôte de Graphite-Perfdata est invalide

```
The url of Graphite perfdata [ URL ] is invalid : The hostname or IP address is empty or not found.
```

Lorsque le module Graphite Perfdata est configuré avec un hôte vide, une erreur est remontée dans le shinken-healthcheck

**Correction** : Changer le nom d'hôte dans la configuration de Graphite Perfdata ( *par défaut dans /etc/shinken/modules/graphite.cfg* ) par un nom d'hôte ou une adresse IP valide

```
Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      ERROR: The url of Graphite perfdata [ http://:2003 ] is invalid : The hostname or IP address is empty or not found
```

## Les erreurs de lecture

### Cas 1 : La configuration du module webui.cfg précise un royaume qui n'existe pas

C'est le cas, si le message suivant apparaît :

```
Realm [*] : This realm is an unknown realm. Know realms are : *
```

Dans notre exemple : Dans le fichier `/etc/shinken/module/webui.cfg` la configuration précise un royaume qui n'existe pas.

```
graphite_backends      Bordeaux:192.168.1.23, France:192.168.1.23
```

**Correction** : Préciser un nom de royaume qui existe.

```
graphite_backends      *:192.168.1.23:80
```

```
Reading - UI Visualisation configuration(s): The * means that webui will read all realms that it's broker manage
broke-master (France):
WebUI
  AT RISK: Realm [Paris]      : This realm does not have neither hosts nor Graphite backend. If hosts are added, their graphics will not be display
  OK:     Realm [Bordeaux]    : The Graphite server on http://192.168.1.23:80 (itself) will be used
  OK:     Realm [France]     : The Graphite server on http://192.168.1.23:80 (itself) will be used
WebUI_01
  OK:     Realm [Bordeaux] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
  OK:     Realm [France] (*)  : The Graphite server on http://192.168.1.23:80 (itself) will be used
  OK:     Realm [Paris] (*)   : The Graphite server on http://192.168.1.23:80 (itself) will be used
```

### Cas 2 : La configuration du module webui.cfg précise un royaume non géré par le broker

C'est le cas, si le message suivant apparaît :

```
Realm [*] : This realm can't be accessed by the broker *. It can access to : *
```

Il y a deux cas possibles :

Exemple 1 : le broker ne gère pas les sous royaume, mais dans la configuration du module webui est précisée l'adresse d'un royaume fils

Exemple 2 : le broker broker-reunion reçoit un module Webui\_Reunion est configuré pour le royaume corse dont il ne s'occupe pas.

Correction :

- Le royaume qui ne peut être accessible est un sous royaume du royaume qui est géré par le broker. Dans ce cas, il suffit de dire au broker de gérer les sous-royaumes : ajouter la gestion des sous-royaumes par le broker dans `/etc/shinken/brokers/broker-master3.cfg`

```
manage_sub_realms      1
```

- Le royaume n'est pas un sous royaume, il faut enlever le royaume non géré du module webui. Il est possible de spécifier le caractère \* dans la définition des backend afin de dire que ce module webui prendra tous les royaumes et sous-royaumes du broker vers le même `graphite_backend`

### Realm \* can't be accessed by the broker \*

```
Reading - UI Visualisation configuration(s): The * means that webui will read all realms that it's broker manage
broker-corse (Corse):
  WebUI
    OK: Realm [Corse] (*) : The Graphite server on 192.168.1.53 (itself) will be used
broker-metropole-01 (Metropole):
  webui-01-validation16-01
    OK: Realm [Corse] : The Graphite server on 192.168.1.53 will be used
    OK: Realm [Metropole] : The Graphite server on 172.16.0.181 will be used
    OK: Realm [Reunion] : The Graphite server on 192.168.1.148 will be used
  webui-02-validation16-01
    OK: Realm [Corse] : The Graphite server on 192.168.1.53 will be used
    OK: Realm [Metropole] : The Graphite server on 172.16.0.181 will be used
    OK: Realm [Reunion] : The Graphite server on 192.168.1.148 will be used
  webui-03-validation16-01
    OK: Realm [Corse] : The Graphite server on 192.168.1.53 will be used
    OK: Realm [Metropole] : The Graphite server on 172.16.0.181 will be used
    OK: Realm [Reunion] : The Graphite server on 192.168.1.148 will be used
broker-reunion (Reunion):
  WebUI
    OK: Realm [Reunion] (*) : The Graphite server on 192.168.1.148 (itself) will be used
  WebUI_Reunion
    ERROR: Realm [Corse] : This realm can't be accessed by the broker broker-reunion. It can access to : Reunion
    OK: Realm [Reunion] : The Graphite server on 192.168.1.148 (itself) will be used
```

### Cas 3 : Le serveur graphite précisé dans la configuration de webui ne gère pas les données de ce royaume

C'est le cas, si le message suivant apparaît :

```
Realm [*] : The Graphite server on * does not contain data for this realm
```

Exemple : Le serveur graphite qui est sur 192.168.1.53 sauvegarde uniquement les données du royaume Corse. Cependant, le broker broker-metropole-01 est configuré avec une webui qui va lire les données pour le royaume Metropole sur le serveur graphite 192.168.1.53.

```
graphite_backends          Corse:192.168.1.53, Reunion:192.168.1.148, Metropole:192.168.1.53
```

Correction : Préciser un serveur qui gère les données du royaume Metropole:

```
graphite_backends          Corse:192.168.1.53, Reunion:192.168.1.148, Metropole:172.16.0.181
```

### Realm [\*] : The Graphite server on \* does not contain data for this realm

```
Reading - UI Visualisation configuration(s): The * means that webui will read all realms that it's broker manage
broker-corse (Corse):
  WebUI
    OK: Realm [Corse] : There are no hosts in this realm.
broker-metropole-01 (Metropole):
  webui-01-validation16-01
    ERROR: Realm [Metropole] : The Graphite server on 192.168.1.53 does not contain data for this realm
    OK: Realm [Corse] : There are no hosts in this realm.
    OK: Realm [Reunion] : There are no hosts in this realm.
  webui-02-validation16-01
    OK: Realm [Corse] : There are no hosts in this realm.
    OK: Realm [Metropole] : The Graphite server on 172.16.0.181 will be used
    OK: Realm [Reunion] : There are no hosts in this realm.
  webui-03-validation16-01
    OK: Realm [Corse] : There are no hosts in this realm.
    OK: Realm [Metropole] : The Graphite server on 172.16.0.181 will be used
    OK: Realm [Reunion] : There are no hosts in this realm.
broker-reunion (Reunion):
  WebUI
    OK: Realm [Corse] : There are no hosts in this realm.
    OK: Realm [Reunion] : There are no hosts in this realm
```

### Cas 4 : Le serveur graphite précisé dans la configuration de webui n'est pas un serveur graphite connu

C'est le cas, si le message suivant apparaît :

```
The server on * is not a known Graphite server : There is no module graphite_perfdata that writes data on
backend *
```

Exemple : La configuration de webui précise que les données du royaume Metropole sont sur 172.16.0.182 qui n'est pas un serveur graphite connu (aucun module graphite\_perfdata n'est configuré pour écrire des données sur ce serveur)

```
graphite_backends      Corse:192.168.1.53, Reunion:192.168.1.148, Metropole:172.16.0.182
```

Correction : Préciser un serveur qui gère les données du royaume Metropole :

```
graphite_backends      Corse:192.168.1.53, Reunion:192.168.1.148, Metropole:172.16.0.181
```

### The server on \* is not a known Graphite server

```
Reading - UI Visualisation configuration(s): The * means that webui will read all realms that it's broker manage
broker-corse (Corse):
WebUI
OK:      Realm [Corse]      : There are no hosts in this realm.
broker-metropole-01 (Metropole):
webui-01-validation16-01
ERROR:   Realm [Metropole] : The server on 172.16.0.182 is not a known Graphite server : There is no module graphite_perfdata that writes data on backend 172.16.0.1
OK:      Realm [Corse]      : There are no hosts in this realm.
OK:      Realm [Reunion]    : There are no hosts in this realm.
webui-02-validation16-01
OK:      Realm [Corse]      : There are no hosts in this realm.
OK:      Realm [Metropole]  : The Graphite server on 172.16.0.181 will be used
OK:      Realm [Reunion]    : There are no hosts in this realm.
webui-03-validation16-01
OK:      Realm [Corse]      : There are no hosts in this realm.
OK:      Realm [Metropole]  : The Graphite server on 172.16.0.181 will be used
OK:      Realm [Reunion]    : There are no hosts in this realm.
broker-reunion (Reunion):
WebUI
OK:      Realm [Corse]      : There are no hosts in this realm.
OK:      Realm [Reunion]    : There are no hosts in this realm
```

### Cas 5 : Le broker gère des royaumes qui ne sont pas configurés dans les graphite\_backends de la webui

C'est le cas, si le message suivant apparaît :

```
Realm [*] : No graphics will be displayed as this realm is not present in "graphite_backends" parameter
```

Exemple : Le broker broker-metropole-01 gère le royaume Metropole et ses deux sous-royaume Reunion et Corse. Dans la définition de la webui webui-02-validation16-01, le royaume Reunion est manquant. On ne pourra donc pas accéder aux métriques sur les hôtes du royaume Reunion sur la webui webui-02-validation16-01

```
graphite_backends      Corse:192.168.1.53, Metropole:172.16.0.181
```

Correction : Il faut ajouter le royaume manquant spécifiquement, ou bien mettre le caractère \* pour indiquer que tous les royaumes se situent sur le même serveur

```
graphite_backends      Corse:192.168.1.53, Reunion:192.168.1.148, Metropole:172.16.0.181
```



Cette erreur n'est présente que s'il y a des hôtes dans ce royaume. Si le royaume n'a pas d'hôte, aucune métrique ne sera lue et donc il n'y aura pas d'erreurs.

## Realm [\*] : No graphics will be displayed as this realm is not present in "graphite\_backends" parameter

```
Reading - UI Visualisation configuration(s): The * means that webui will read all realms that it's broker manage
broker-corse (Corse):
  WebUI
    OK: Realm [Corse] (*) : The Graphite server on 192.168.1.53 (itself) will be used
broker-metropole-01 (Metropole):
  webui-01-validation16-01
    OK: Realm [Corse] : The Graphite server on 192.168.1.53 will be used
    OK: Realm [Metropole] : The Graphite server on 172.16.0.181 will be used
    OK: Realm [Reunion] : The Graphite server on 192.168.1.148 will be used
  webui-02-validation16-01
    ERROR: Realm [Reunion] : No graphics will be displayed as this realm is not present in "graphite_backends" parameter
    OK: Realm [Corse] : The Graphite server on 192.168.1.53 will be used
    OK: Realm [Metropole] : The Graphite server on 172.16.0.181 will be used
  webui-03-validation16-01
    OK: Realm [Corse] : The Graphite server on 192.168.1.53 will be used
    OK: Realm [Metropole] : The Graphite server on 172.16.0.181 will be used
    OK: Realm [Reunion] : The Graphite server on 192.168.1.148 will be used
broker-reunion (Reunion):
  WebUI
    OK: Realm [Reunion] (*) : The Graphite server on 192.168.1.148 (itself) will be used
  WebUI_Reunion
    OK: Realm [Reunion] : The Graphite server on 192.168.1.148 (itself) will be used
```

Dans le cas où le royaume n'a pas de graphite backend mais qu'il n'a pas d'hôtes, alors le message change et devient un " **AT RISK** " car il est nécessaire d'avoir des hôtes pour l'affichage des graphiques :

```
Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      OK: Realm [All] is stored by the module [Graphite-Perfdata] of the broker [broker-master] on graphite backend 192.168.1.72
      OK: Realm [France] is stored by the module [Graphite-Perfdata] of the broker [broker-master] on graphite backend 192.168.1.72
    Reading - UI Visualisation configuration(s): The * means that webui will read all realms that it's broker manage
    broker-master (All):
      WebUI
        AT RISK: Realm [France] : This realm does not have neither hosts nor graphite backend. If hosts are added, their graphics will not be displayed
        OK: Realm [All] : The Graphite server on http://192.168.1.72:80 will be used
```

## Cas 6 : Le broker rencontre des erreurs pour lire les données sur les serveurs graphite

C'est le cas, si le message suivant apparaît :

```
Graphite read error happened * times (*) during the last 24h, check the /opt/graphite/storage/whisper read access on the graphite host
```

Le broker a rencontré des erreurs lors des dernières 24 heures pour lire des métriques.

Correction : Il faut consulter les logs du broker concerné afin de connaître la nature de l'erreur. (problème de droits sur les dossiers de stockage u serveur graphite)

```
Server(s)
192.168.1.59:
  Write connexion status
    OK: success: broker-prod
  Read connexion status
    ERROR: failure: broker-prod - Graphite read error happened 1 times during the last 24h, check the /opt/graphite/storage/whisper/ read access on the graphite host
```

## Cas 7 : Le port de Graphite n'est pas correct

```
The Graphite backend [ GRAPHITE_BACKEND ] is_incorrect : The port [ HTTP_PORT ] is not valid. Valid values are integers from 0 to 65535.
```

Quand un port incorrect est utilisé dans le graphite backend, une erreur est remontée dans le shinken-healthcheck pour l'indiquer.

```
graphite_backends      *:192.168.1.23:invalid
```

Correction : Il faut changer le port du "graphite\_backend" concerné par un port qui existe ( *un entier compris entre 0 et 65535 inclus* )

```
graphite_backends      *:192.168.1.23:8080
```



Faites attention aux ports inférieurs à 1024, ceux-ci sont généralement réservés pour certains usages comme par exemple les ports 22 pour le protocole SSH ou 443 pour HTTPS.

```
Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      OK: Realm [ France ] is stored by the module [ Graphite-Perfdata-France ] of the broker [ broker-master ] on graphite backend 192.168.1.23
      OK: Realm [ Bordeaux ] is stored by the module [ Graphite-Perfdata-Bordeaux ] of the broker [ broker-bordeaux ] on graphite backend 192.168.1.128
    Reading - UI Visualisation configuration(s): The * means that webui will read all realms that it's broker manage
      broker-bordeaux (Bordeaux):
        WebUI_Bordeaux
          OK: Realm [ Bordeaux ] (*) : The Graphite server on http://192.168.1.128:80 will be used
      broker-master (France):
        WebUI_France
          ERROR: Realm [ France ] (*) : The Graphite backend [ 127.0.0.1:invalid ] is incorrect : The port [ invalid ] is not valid. Valid values are integers from 0 to 65535.
```

## Cas 8 : L'adresse locale des serveurs de lecture ou d'écriture de Graphite n'a pas pu être résolu

Lorsqu'un graphite\_backends d'un module WebUI ou qu'un module Graphite-Perfdata utilise une adresse locale ( *localhost* ou *127.0.0.1* ), il est possible que le shinken-healthcheck n'arrive pas à résoudre son adresse IP ( *remplacer localhost ou 127.0.0.1 par sa réelle IP* )

### Configuration du graphite\_backends en local d'une WebUI

```
graphite_backends      *:localhost
```

### Configuration d'un module graphite\_perfdata sur une adresse locale

```
host                   localhost
```

Dans ce cas, on remplace l'adresse par `_127.0.0.1_` et on avertit que son IP locale n'a pas pu être résolue :

```
Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      OK: Realm [ All ] is stored by the module [ Graphite-Perfdata ] of the broker [ broker-master ] on graphite backend 127.0.0.1 (192.168.1.23)
    Reading - UI Visualisation configuration(s): The * means that webui will read all realms that it's broker manage
      broker-master (All):
        WebUI
          OK: Realm [All] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
      Server(s)
        127.0.0.1:
          AT RISK: The local address could not be resolved
```

**Correction :** L'erreur provient du fait que le shinken-healthcheck a eu un problème lors de l'ouverture de socket. Cela peut provenir de la configuration réseau du serveur.

## Erreurs de format dans le paramètre graphite\_backends de la WebUI

### Cas 1 : Pas de royaume dans le backend

#### Configuration du graphite\_backends sans nom de royaume

```
graphite_backends      http://localhost:80, France:http://192.168.1.68
```

Lorsque aucun royaume n'a été trouvé dans la définition d'un backend, ou bien que le caractère servant à séparer le nom du royaume de l'adresse n'est pas trouvé ( *caractère =* ), l'erreur suivante est remontée :

```
Graphite Backend [ BACKEND ] format error. No realm name found. Expected format :
<REALM>=<PROTOCOL>://<HOSTNAME>:<PORT>
```

```

Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      OK: Realm [ All ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
      OK: Realm [ France ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
      OK: Realm [ Bordeaux ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
    Reading - UI Visualisation configuration(s): The * means that weui will read all realms that it's broker manage
      broke-master (All):
        WebUI
          ERROR: Realm [ <UNKNOWN> ] : Graphite Backend [ http://192.168.1.23:80 ] format error. No realm name found. Expected format : <REALM>=<PROTOCOL>://<HOSTNAME>:<PORT>
          OK: Realm [ All ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
          OK: Realm [ Bordeaux ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
          OK: Realm [ France ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used

```

**Correction :** Rajouter un nom de royaume et faire en sorte que chaque backend suive le format : **<Realm>=<Protocol>://<Hostname>:<Port>**

### Cas 2 : Trop de séparateur de royaume

#### Configuration du graphite\_backends avec trop de séparateur de royaume

```
graphite_backends      France=Belgique=http://192.168.1.23:80
```

Lorsque le caractère servant à séparer le nom du royaume et l'adresse ( *caractère =* ) est présent plus d'une fois dans le backend, l'erreur suivante est remontée :

```
Graphite Backend [ BACKEND ] format error. Too much realm separators found ( = ). Expected format : <REALM>=<PROTOCOL>://<HOSTNAME>:<PORT>
```

```

Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      OK: Realm [ All ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
      OK: Realm [ France ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
      OK: Realm [ Bordeaux ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
    Reading - UI Visualisation configuration(s): The * means that weui will read all realms that it's broker manage
      broke-master (All):
        WebUI
          ERROR: Realm [ <UNKNOWN> ] : Graphite Backend [ France=Belgique=http://192.168.1.23:80 ] format error. Too much realm separators found ( = ). Expected format : <REALM>=<PROTOCOL>://<HOSTNAME>:<PORT>
          OK: Realm [ All ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
          OK: Realm [ Bordeaux ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
          OK: Realm [ France ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used

```

**Correction :** Enlever le caractère "=" en trop pour que le backend respecte le format suivant : **<Realm>=<Protocol>://<Hostname>:<Port>**

### Cas 3 : Pas de protocole

#### Configuration du graphite\_backends sans protocole

```
graphite_backends      France=192.168.1.23:80
```

Lorsque le protocole n'a pas été trouvé dans le backend ( *ou que la chaîne "://" n'est pas trouvée* ), l'erreur suivante est remontée :

```
Graphite Backend [ BACKEND ] format error. No protocol found ( http:// ). Expected format : <REALM>=<PROTOCOL>://<HOSTNAME>:<PORT>
```

```
Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      OK: Realm [ All ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
      OK: Realm [ France ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
      OK: Realm [ Bordeaux ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
    Reading - UI Visualisation configuration(s): The * means that webui will read all realms that it's broker manage
      broke-master (All):
        WebUI
          ERROR: Realm [ <UNKNOWN> ] : Graphite Backend [ France=192.168.1.23:80 ] format error. No protocol found ( http:// ). Expected
format : <REALM>=<PROTOCOL>://<HOSTNAME>:<PORT>
          OK: Realm [ All ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
          OK: Realm [ Bordeaux ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
          OK: Realm [ France ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
```

**Correction :** Rajouter le protocole manquant pour respecter le format suivant : **<Realm>=<Protocol>://<Hostname>:<Port>**

#### Cas 4 : Pas de port HTTP

##### Configuration du graphite\_backends sans port HTTP

```
graphite_backends      France=http://192.168.1.23
```

Lorsque le port HTTP n'est pas précisé dans le backend, l'erreur suivante est remontée :

```
Graphite Backend [ BACKEND ] format error. No HTTP port found. Expected format :
<REALM>=<PROTOCOL>://<HOSTNAME>:<PORT>
```

```
Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      OK: Realm [ All ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
      OK: Realm [ France ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
      OK: Realm [ Bordeaux ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
    Reading - UI Visualisation configuration(s): The * means that webui will read all realms that it's broker manage
      broke-master (All):
        WebUI
          ERROR: Realm [ France ] : Graphite Backend [ France=http://192.168.1.23 ] format error. No HTTP port found. Expected forma
t : <REALM>=<PROTOCOL>://<HOSTNAME>:<PORT>
          OK: Realm [ All ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
          OK: Realm [ Bordeaux ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
```

**Correction :** Rajouter le port manquant pour respecter le format suivant : **<Realm>=<Protocol>://<Hostname>:<Port>**

#### Cas 5 : Pas de nom d'hôte ou d'adresse IP

##### Configuration du graphite\_backends sans nom d'hôte

```
graphite_backends      France=http://:80
```

Lorsque aucun nom d'hôte ou d'adresse IP n'est précisée dans l'adresse du backend, l'erreur suivante est remontée :

```
Graphite Backend [ BACKEND ] format error. No hostname or IP address found. Expected format :
<REALM>=<PROTOCOL>://<HOSTNAME>:<PORT>
```

```

Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      OK: Realm [ All ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
      OK: Realm [ France ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
      OK: Realm [ Bordeaux ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
    Reading - UI Visualisation configuration(s): The * means that weui will read all realms that it's broker manage
      broke-master (All):
        WebUI
          ERROR: Realm [ France ] : Graphite Backend [ France=http://:80 ] format error. No hostname or IP address found. Expect
format : <REALM>=<PROTOCOL>://<HOSTNAME>:<PORT>
          OK: Realm [ All ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
          OK: Realm [ Bordeaux ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used

```

**Correction :** Ajouter un nom d'hôte ( *ou une adresse IP* ) pour respecter le format suivant : **<Realm>=<Protocol>://<Hostname>:<Port>**

## Erreurs dans le contenu de la valeur de la clé graphite\_backend

### Cas 1 : Port invalide

#### Configuration du graphite\_backends avec un port incorrect

```
graphite_backends      France=http://localhost:invalid
```

Lorsqu'un port non valide a été renseigné dans un backend, l'erreur suivante est remontée :

```
Graphite Backend [ BACKEND ] URL error : The port [ invalid ] is not valid. Valid values are integers from 0 to 65535.
```

```

Storage
[graphite]
  Realm(s)
    Writing - Data location(s):
      OK: Realm [ All ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
      OK: Realm [ France ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
      OK: Realm [ Bordeaux ] is stored by the module [ Graphite-Perfdata ] of the broker [ broke-master ] on graphite backend 192.168.1.23
    Reading - UI Visualisation configuration(s): The * means that weui will read all realms that it's broker manage
      broke-master (All):
        WebUI
          ERROR: Realm [ France ] : Graphite Backend [ France=http://192.168.1.23:invalid ] URL error : The port [ invalid ] is not val
Valid values are integers from 0 to 65535.
          OK: Realm [ All ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used
          OK: Realm [ Bordeaux ] (*) : The Graphite server on http://192.168.1.23:80 (itself) will be used

```

**Correction :** Remplacer le port erroné par un entier entre 0 et 65535 ( *inclus* ).

### Cas 2 : Protocole non supporté

#### Configuration du graphite\_backends avec un protocole non supporté

```
graphite_backends      France=tcp://localhost:80
```

Lorsque le protocole renseigné dans un backend n'est pas un protocole supporté, l'erreur suivante est remontée :

```
Graphite Backend [ BACKEND ] URL error : Protocol not supported. Supported protocol list : [ http ]
```

