

Les logs du Scheduler

Sommaire

Démarrage du démon
Chargement des broks initiaux par un regenerator (créateur d'objets des modules de broker) et vérifier que c'est bien la même configuration charger entre les regenerators / scheduler / arbiter
Quand un Scheduler reçoit une nouvelle configuration de l'Arbiter, il logue
Quand un Scheduler est en train de charger sa nouvelle configuration, il logue
Quand un Scheduler a fini de charger la nouvelle configuration reçu, il logue
Demande et génération des broks initiaux
Logs de chargement des modules
Communication entre Schedulers
Quand un Scheduler distant n'a pas reçu de configuration de l'Arbiter
Quand un Scheduler distant reçoit une configuration de l'Arbiter
Quand un Scheduler distant reçoit une nouvelle configuration de l'Arbiter
Requête d'export des données
Avec les noms des éléments
Sans les noms des éléments
Erreurs possibles
Le Scheduler est en phase de démarrage
Le Scheduler ne gère pas encore de configuration
La configuration interdit l'export de données
Impossible d'exporter les noms des éléments car le mot de passe d'accès n'a pas été configuré
Accès refusé suite à l'utilisation d'un mauvais mot de passe
Erreur de cohérence des périodes de maintenance
Exécution de commandes externes reçues d'un Receiver
Exemples
Exécution de commandes par les gestionnaires d'évènements
Échanges par paquet de taille limité des Broks avec le Broker
Log de performance de la boucle du scheduler
Surcharge serveur en activité disque, ralentissant l'écriture des logs
Logs d'erreur concernant des objets d'exécution de check qui restent en mémoire dans le Scheduler
Logs de WARNING concernant la suppression d'objets "notification" défectueux
Logs de WARNING concernant les notifications qui ne sont pas envoyées à cause des périodes de notification
Logs d'ERROR concernant l'arrêt du démon s'il n'arrive pas à charger les données de rétention
Log d'ERROR quand on charge depuis la rétention un nombre trop important de vérifications sur les hôtes ou les checks
Options du support Shinken
Forcer l'étalement des checks au démarrage

Les fichiers de log du Scheduler sont situés dans le dossier **/var/log/shinken/**. Pour plus d'informations, consultez la page [Fichiers Logs](#).

Démarrage du démon

Au démarrage et tous les jours à minuit, ce log indique la version ainsi que le numéro de patch cumulatif du démon.

Démarrage du daemon

```
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] Daemon version is: XX.XX.XX-release.fr culmulative-patch-YY
```

Lors du démarrage du démon, une ligne est disponible:

Démarrage du daemon

```
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [START-DAEMON] The daemon (version=02.08.02-release.fr) is now started as a daemon (detached from any shell) with pid=15412
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [ SYSTEM          ] System resource number of open files is set to (soft:1024 / hard:1024 ) (from parameter max_file_descriptor_limit)
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [ SYSTEM          ] System resource number of processes /threads is set to (soft:unlimited / hard:unlimited ) (set at system max values)
```

Avec comme informations principales:

- Sa version
- Son numéro de PID
- Ses limites systèmes en nombre de fichiers/socket ouvrables, et le nombre max de processus/threads
- Sur quelle(s) interface(s) le démon écoute

S'il écoute sur toutes les interfaces:

```
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [schedulerdaemon] The daemon listens on all network interfaces.
```

S'il écoute sur une interface précise:

```
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [schedulerdaemon] The daemon listens on the IP_INTERFACE: PORT_INTERFACE network interface.
```

Chargement des broks initiaux par un regenerator (créateur d'objets des modules de broker) et vérifier que c'est bien la même configuration charger entre les regenerators / scheduler / arbiter

Les logs suivants permettent de suivre le chargement de la configuration de supervision entre l'Arbiter et les Schedulers jusqu'aux interfaces : webui / livestatus / livedata.

Il existe 2 types d'identifiants de configuration (représentation de la configuration).

- **configuration_uuid**: uuid de configuration total généré par l'Arbiter
- **configuration_part_id**: id de la partie de configuration géré par un Scheduler

Quand un Scheduler reçoit une nouvelle configuration de l'Arbiter, il logue

```
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ CONFIGURATION ] The arbiter send us a new configuration: [ configuration_part_id=configuration_part_id, configuration_uuid=configuration_uuid, arbiter=arbiter_name, architecture=architecture_name, date=creation_date]
```

- **configuration_part_id**: id de la partie de configuration spécifiquement gérée par ce Scheduler (unique pour chaque Scheduler)
- **configuration_uuid**: uuid créé lors du démarrage de l'Arbiter qui correspond donc à l'id de la configuration géré par l'Arbiter
- **arbiter_name**: nom de l'Arbiter qui a créé cette configuration
- **architecture_name**: nom de l'architecture
- **creation_date**: date du démarrage de l'Arbiter

✗ Dans le cas où le scheduler n'est pas de la même version que l'arbiter et que l'option `*mismatch_version_error*` est activé sur l'arbiter:

```
[YYYY-MM-DD HH:MM:SS] ERROR : [ scheduler-name ] Incompatible daemon version : Your Arbiter daemon is in version [XX.XX.XX-release.fr culmulative-patch-YY] while this daemon is in version [XX.XX.XX-release.fr culmulative-patch-YY]. Refusing this configuration.
```

⚠ Dans le cas où le scheduler n'est pas de la même version que l'arbiter et que l'option `*mismatch_version_error*` est désactivé sur l'arbiter:

```
[YYYY-MM-DD HH:MM:SS] WARNING : [ scheduler-name ] Incompatible daemon version : Your Arbiter daemon is in version [XX.XX.XX-release.fr culmulative-patch-YY] while this daemon is in version [XX.XX.XX-release.fr culmulative-patch-YY].
```

Quand un Scheduler est en train de charger sa nouvelle configuration, il logue

Le log suivant apparaît si le Scheduler n'a pas fini de charger sa configuration est qu'il reçoit une demande de statut

```
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [scheduler][0] Someone asks to get the raw stats (daemon Health) but the scheduler is not initialized
```

Quand un Scheduler a fini de charger la nouvelle configuration reçu, il logue

```
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ CONFIGURATION ] The configuration [ shard_id=shard_id, scheduler=scheduler, configuration_uuid=configuration_uuid, arbiter=arbiter_name, architecture=architecture, date=creation_date, active=active] was loaded in [loading_time]s
```

- **shard_id**: id de la partie de configuration gérée par ce Scheduler (unique pour chaque Scheduler)
- **scheduler**: Nom du Scheduler qui reçoit la configuration
- **configuration_uuid**: uuid crée lors du démarrage de l'Arbiter qui correspond donc à l'id de la configuration géré par l'Arbiter
- **arbiter_name**: nom de l'Arbiter qui a créé cette configuration
- **architecture**: nom de l'architecture de l'Arbiter
- **creation_date**: date du démarrage de l'Arbiter
- **active**: définit si le démon est mis en tant que spare ou non
- **loading_time**: temps de chargement de la configuration

Exemple Scheduler chargement de la nouvelle configuration

```
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ CONFIGURATION ] The configuration [shard_id=256, scheduler=scheduler-master, configuration_uuid=c6e6edef648246c290ac252d623719f3, arbiter=arbiter-master, architecture=Shinken-groy-dev-02-07, date=09-06-2021 15:30:06, active=True] was loaded in [0.00109791755676]s
```

Demande et génération des broks initiaux

Les modules de Broker ont besoin des Broks dit "initiaux" afin d'avoir une image complète des éléments de supervision (hôtes, checks, mais aussi timeperiod et commandes). Quand une nouvelle configuration est chargée par le scheduler, le Broker le détecte et demande une génération des broks initiaux. Ceci est visible par la log suivant:

```
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ INITIAL BROKS ] [ REGISTERING ] [ broker-master ] The Broker is registering for initial broks generation. ( Currently 1 registered )
```

La génération va commencer.

```
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ INITIAL BROKS ] [ GENERATING ] [ TIMING ] Generating initial broks for configuration [shard_id=256, scheduler=scheduler-name, configuration_uuid=60b36ae8df5f4090bd7ef5d576ee1a15, arbiter=arbiter-name, architecture=architecture-name, date=26-08-2022 16:13:27, active=True] (currently have 36 hosts and 463 services)
```

Pendant ce temps, d'autres demandes vont se rajouter à la première si on a plusieurs Brokers.

Une fois la génération finie, on aura le log suivant:

```
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ INITIAL BROKS ] [ GENERATING ] [ DONE ] [ Elapsed time=0.042s ] 241 initial broks are generated for 2 brokers:  
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ INITIAL BROKS ] [ GENERATING ] [ DONE ] - broker-master  
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ INITIAL BROKS ] [ GENERATING ] [ DONE ] - broker-secondary
```

Et les Brokers seront prévenus qu'ils peuvent désormais revenir télécharger les broks:

```
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ INITIAL BROKS ] [ GENERATING ] [ DONE ] [ broker-master ] The broker is warned that the generation is done and can be GET.  
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ INITIAL BROKS ] [ GENERATING ] [ DONE ] [ broker-secondary ] The broker is warned that the generation is done and can be GET.
```

Logs de chargement des modules

Les démons ont une phase de chargement des modules qui est décrite dans la page [GLOBAL - Les logs de gestion des modules - chapitre \[MODULES-MANAGER\]](#)

Communication entre Schedulers

Quand un Scheduler distant n'a pas reçu de configuration de l'Arbiter

Exemple Scheduler réception d'une nouvelle configuration

```
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [ CONFIGURATION ] The scheduler-distant has not yet received any configuration from Arbiter.
```

Quand un Scheduler distant reçoit une configuration de l'Arbiter

Exemple Scheduler réception d'une nouvelle configuration

```
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [ CONFIGURATION ] The scheduler-distant change it's configuration from configuration_uuid to new_configuration_uuid
```

- **configuration_uuid**: uuid créé lors du démarrage de l'Arbiter qui correspond donc à l'id de la configuration géré par l'Arbiter
- **new_configuration_uuid**: uuid créé lors du démarrage de l'Arbiter qui correspond donc au nouvel id la configuration géré par l'Arbiter

Exemple Scheduler réception d'une nouvelle configuration

```
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [ CONFIGURATION ] The scheduler-distant change it's configuration from 6ddbcbdd9260e40d9a8a48e1eabc875a5 to 32ab5f3457fb4c3fbe2415d873ae199e
```

Quand un Scheduler distant reçoit une nouvelle configuration de l'Arbiter

Exemple Scheduler réception d'une nouvelle configuration

```
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [ CONFIGURATION ] The scheduler-distant received a new configuration (uuid=073c072c6d524b38bb4c08b1fdfa7f89)
```

Requête d'export des données

La commande shinken-scheduler-export-data invoquée depuis le serveur de l'Arbiter permet de questionner les Schedulers pour extraire des données sur les éléments supervisés.

Lorsque cette commande est utilisée, les logs qui suivent permettent de voir la nature de la réponse faite par un Scheduler.

Avec les noms des éléments

```
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [ EXPORT DATA ] full export of XXX elements in X.XXXs from Arbiter IPXXX.XXX.XXX.XXX
```

Sans les noms des éléments

```
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [ EXPORT DATA ] anonymous export of XXX elements in X.XXXs from Arbiter IPXXX.XXX.XXX.XXX
```

Erreurs possibles

Le Scheduler est en phase de démarrage

```
[YYYY-MM-DD HH:MM:SS] WARNING: [ scheduler-name ] [ EXPORT DATA ] Request from Arbiter IPXXX.XXX.XXX.XXX fails. Scheduler is not ready (initialisation ongoing)
```

Le Scheduler ne gère pas encore de configuration

```
[YYYY-MM-DD HH:MM:SS] WARNING: [ scheduler-name ] [ EXPORT DATA ] Request from Arbiter IPXXX.XXX.XXX.XXX fails. Scheduler is not ready (waiting for configuration)
```

La configuration interdit l'export de données

```
[YYYY-MM-DD HH:MM:SS] WARNING: [ scheduler-name ] [ EXPORT DATA ] Request from Arbiter IPXXX.XXX.XXX.XXX fails. Export is disabled by configuration parameter scheduler__export_data__enabled
```

Impossible d'exporter les noms des éléments car le mot de passe d'accès n'a pas été configuré

```
[YYYY-MM-DD HH:MM:SS] WARNING: [ scheduler-name ] [ EXPORT DATA ] Request from Arbiter IPXXX.XXX.XXX.XXX fails. The scheduler scheduler__export_data__passwordparameter is missing or void, that is forbidden for not anonymous request
```

Accès refusé suite à l'utilisation d'un mauvais mot de passe

```
[YYYY-MM-DD HH:MM:SS] WARNING: [ scheduler-name ] [ EXPORT DATA ] Request from Arbiter IPXXX.XXX.XXX.XXX fails. Access token is not valid
```

Erreur de cohérence des périodes de maintenance

Les périodes de maintenance ont une incohérence entre non identifiée qui consiste à ce qu'une période de maintenance va être démarrée deux fois, ce qui va poser soucis lors de son arrêt (l'hôte/check sera dans un état incohérent). Pour l'instant, la source de l'incohérence n'a pas été trouvée, mais des protections ont été mises en place pour les éviter.

Au chargement de la rétention, on aura une entrée "ERROR":

```
[YYYY-MM-DD HH:MM:SS] ERROR : [ scheduler-name ] [ DOWNTIME-INCOHERENCY ] The host Linux has bad downtime values (saved number of downtime=1, actual=0). We are fixing the values. Please report it to the support.
```

Lors du second démarrage de la période de maintenance:

```
[YYYY-MM-DD HH:MM:SS] ERROR : [ scheduler-name ] [ DOWNTIME-INCOHERENCY ] The downtime [Downtime id=1596532528600972 active=active type=fixed start=Tue Aug 4 11:16:06 2020 - end=Tue Aug 4 12:15:06 2020] on moi got activated twice. This is a bug and MUST be reported to support for investigation. Thanks.
```

Dans les deux cas, il faut récupérer les logs du Scheduler, et les donner en analyse au support Shinken.

Exécution de commandes externes reçues d'un Receiver

Lorsque le Receiver envoie des commandes externes au Scheduler, il est possible que ce dernier n'ait pas reçu de configuration de la part de l'Arbiter et qu'il ne soit pas en capacité d'exécuter ces commandes. Dans ce cas-là, cette entrée est renvoyé :

```
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ BUS COMMANDS ] Get shinken external commands (like recheck, set acknowledge, etc) from the Receiver receiver-master but i am not ready. Waiting for configuration from Arbiter.
```

Dans le cas où, les commandes ont pu être exécutées :

```
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ BUS COMMANDS ] Running 200 external commands (like recheck, set acknowledge, etc) received from the Receiver receiver-master
```

Les commandes externes différentes des mise à jour de statut sont également listées :

```
[YYYY-MM-DD HH:MM:SS] INFO    : [ scheduler-name ] [ EXTERNAL COMMAND ] received command [ COMMAND_NAME ] on  
[ ELEMENT ] with args [ OPTIONAL_PARAMETERS ]
```

Les valeurs les plus courantes pour COMMAND_NAME sont les suivantes :

COMMAND_NAME	Type d'élément	Action
Check acknowledge	check	Définit une prise en compte sur un check
Check acknowledge deletion	check	Supprime une prise en compte sur un check
Create check downtime	check	Définit une période de maintenance sur un check
Delete check downtime	check	Supprime une période de maintenance sur un check
Schedule immediate check	check	Force l'exécution de la vérification du statut d'un check
Host acknowledge	host	Définit une prise en compte sur un hôte
Host acknowledge deletion	host	Supprime une prise en compte sur un hôte
Create host downtime	host	Définit une période de maintenance sur un hôte
Delete host downtime	host	Supprime une période de maintenance sur un hôte
Schedule immediate host	host	Force l'exécution de la vérification du statut d'un hôte

Exemples

Mise en place d'une prise en compte sur un check

```
[2023-03-30 14:10:28] INFO    : [ scheduler-supdesup2 ] [ EXTERNAL COMMAND ] received command [ Check  
acknowledge ] on [ Int-II - CLOUD - int-google-1/Load Average SSH ] with args [ 2,True,True,bmourgues,En  
cours de résolution ]
```

Retrait d'une prise en compte sur un check

```
[2023-03-30 14:12:18] INFO : [ scheduler-supdesup2 ] [ EXTERNAL COMMAND ] received command [ Check acknowledge deletion ] on [ Int-II - CLOUD - int-google-1/Load Average SSH ] with args [ ]
```

Mise en place d'une période de maintenance sur un check

```
[2023-03-30 14:13:30] INFO : [ scheduler-supdesup2 ] [ EXTERNAL COMMAND ] received command [ Create check downtime ] on [ Int-II - CLOUD - int-google-1/Load Average SSH ] with args [ 1680178443,1680181983,True,0,0,bmourgues,Mise à jour en cours ]
```

Retrait d'une période de maintenance sur un check

```
[2023-03-30 14:15:42] INFO : [ scheduler-supdesup2 ] [ EXTERNAL COMMAND ] received command [ Delete check downtime ] on [ Downtime id=1680178410781868 active=active type=fixed start=Thu Mar 30 14:14:03 2023 - end=Thu Mar 30 15:13:03 2023 on Int-II - CLOUD - int-google-1/Load Average SSH ] with args [ ]
```

Check forcé

```
[2023-03-30 11:01:04] INFO : [ scheduler-supdesup2 ] [ EXTERNAL COMMAND ] received command [ Schedule immediate check ] on [ Int-II - FRANCE - integration-1/Broker - broker-france-webui - Memory consumption ] with args [ 1680166864 ]
```

Exécution de commandes par les gestionnaires d'évènements

Pour le gestionnaire d'évènements d'un hôte :

```
[YYYY-MM-DD HH:MM:SS] INFO: [ NOM DU SCHEDULER ] SERVICE EVENT HANDLER: NOM DE L'HÔTE; NOM DU SERVICE; STATUT DU SERVICE; CONFIRMATION DU STATUT; NOMBRE DE TENTATIVES DE CONFIRMATION DE STATUT; NOM DE LA COMMANDE
```

Pour le gestionnaire d'évènements d'un check :

```
[YYYY-MM-DD HH:MM:SS] INFO: [ NOM DU SCHEDULER ] HOST EVENT HANDLER: NOM DE L'HÔTE; STATUT DE L'HÔTE; CONFIRMATION DU STATUT; NOMBRE DE TENTATIVES DE CONFIRMATION DE STATUT; NOM DE LA COMMANDE
```

Échanges par paquet de taille limité des Broks avec le Broker

Quand le broker demande les broks au Scheduler, ce dernier va avoir dans ses logs:

```
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ GIVE BROKS ] [ broker-master ] Sending 5 broks (1.3kB)
```

Si on se retrouve dans un cas où la limite du **broks_packet_size** est atteinte (sur les broks initiaux par exemple), on va avoir la ligne suivante:

```
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ GIVE BROKS ] [ broker-master3 ] Sending 2588 broks (413.4 kB) [chunk, still 1698 to send]
```

En cas de problème de communication, il se peut qu'un paquet de broks soit perdu, dans ce cas, une réémission est faite pour éviter cette perte, on aura un WARNING suivant:

```
[YYYY-MM-DD HH:MM:SS] WARNING: [ scheduler-name ] [ GIVE BROKS ] [ broker-master ] Packet number did mismatch "100d0ccf12de4665bf04f2150dcc97d5" != "0ca27bc3ea5440358c1194b5b7c3b4f4" : Re-sending broks (6.3kB)
```

Log de performance de la boucle du scheduler

Dans la boucle du Scheduler, les logs ci-dessous permettent de connaître le temps d'exécution passé pour chaque action.

```
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ SCHEDULER TIME ] [ === Loop start === ] [ Loop
number=7885 ] =====
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ SCHEDULER TIME ] took [ 0.005 ]s to schedule checks,
consume results and create broks
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ SCHEDULER TIME ] took [ 0.005 ]s to update items
context ( downtimes, acknowledge, flapping, root_problems, business values )
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ SCHEDULER TIME ] took [ 0.005 ]s to clean data (
cache, zombies, proxy, stats, notifications list )
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ SCHEDULER TIME ] took [ 0.005 ]s to check environment
and update stats ( satellite_thread, time, retention, orphan, modules )
[YYYY-MM-DD HH:MM:SS] INFO : [ scheduler-name ] [ SCHEDULER TIME ] [ === Loop stop === ] [ Loop
number=7884 ] [PERF] [ 0.082 ]s
```

Afin d'avoir toutes les informations de DEBUG sur les performances du Scheduler (boucle principale), il faut le lancer avec:

```
SHINKEN_LOG_SCHEDULER_RECURRENT_TIMES_FLAG=1 service-shinken-scheduler -d start
```

Surcharge serveur en activité disque, ralentissant l'écriture des logs

Si le serveur hébergeant le démon est surchargé en termes d'IO disques sur le volume qui héberge le fichier de log, alors ce dernier va mettre du temps à s'écrire et va ralentir tout le démon. Il faut alors si c'est faisable isoler le volume des disques sur un disque moins chargé pour ne pas ralentir le démon.

En cas de soucis, il y aura les logs l'entrée suivant:

```
[YYYY-MM-DD HH:MM:SS] WARNING : [ LOGGER ]
[YYYY-MM-DD HH:MM:SS] WARNING : [ LOGGER ]
-----
[YYYY-MM-DD HH:MM:SS] WARNING : [ LOGGER ] [ WRITING ] The log writes time is very high (1.87s). Please look
at your log disk performance.
[YYYY-MM-DD HH:MM:SS] WARNING : [ LOGGER ]
-----
[YYYY-MM-DD HH:MM:SS] WARNING : [ LOGGER ]
```

Logs d'erreur concernant des objets d'exécution de check qui restent en mémoire dans le Scheduler

Dans le cas où le Scheduler détecte que des objets d'exécution de checks ne sont pas bien nettoyés dans l'index de lancement (par rapport au temps).

Ceci peut résulter en deux logs d'ERROR :

- Signifie qu'un objet a été consommé, mais qu'il est peut-être encore présent dans l'index

```
[YYYY-MM-DD HH:MM:SS] ERROR : [ scheduler-name ] [ SCHEDULING ] [ JOB-EXECUTION FAST INDEX ] The check XXXX
is zombie without being cleaned (name=YYYY, was indexed at SSSSSS). Please report to your support.
```

- Suite au premier dans la minute suivante, le Scheduler corrigera l'objet en trop.

```
[YYYY-MM-DD HH:MM:SS] ERROR : [ scheduler-name ] [ JOB-EXECUTION FAST INDEX ] [ SCHEDULING ] [ JOB-EXECUTION
FAST INDEX ] [PERF: lock aquired in 0.000s, done in 0.001s] Did remove 9 forgotten execution from 2 past
seconds indexed (total number of seconds inside=63).
```

Cette situation ne devrait pas être présente, donc il est conseillé d'activer des logs de debug plus détaillés concernant ce mécanisme via la variable d'environnement **SHINKEN_LOG_SCHEDULER_JOB_EXECUTION_FAST_INDEX_FLAG**

```
SHINKEN_LOG_SCHEDULER_JOB_EXECUTION_FAST_INDEX_FLAG=1 service-shinken-scheduler -d restart
```

Il est conseillé de fournir ces logs détaillés au support pour analyse.

Logs de WARNING concernant la suppression d'objets "notification" défectueux

Sur des installations datant d'avant la 02.03.03-U01, des objets "notifications" pouvaient rester indéfiniment en mémoire dans le Scheduler, mais sans être lancés.

- Ces objets de notification sont anciens et défectueux et ils sont vus comme "late" par le check du Scheduler.
- Une mise à jour via une version ou un patch, permet de supprimer ces objets.
- Ceci sera visible sur le premier démarrage du Scheduler avec le log suivant:

```
[YYYY-MM-DD HH:MM:SS] WARNING: [ scheduler-name ] [RETENTION] [myhost] The notification [Notification 557481183 status:scheduled command:VOID ref:unknown t_to_go:Tue Feb 2 07:49:19 2021 (creation=Fri Apr 28 02:19:56 2017) (is_master=False) (on myhost)] was detected as invalid (bug from old code), and was dropped.
```

Logs de WARNING concernant les notifications qui ne sont pas envoyées à cause des périodes de notification

```
[YYYY-MM-DD HH:MM:SS] WARNING: [ scheduler-name ] [ NOTIFICATION      ] The notification for NOM DE L'HOTE OU DU CHECK was not send because notification_period NOM DE LA PERIODE DE TEMPS do not provide a date in the next 366 days ( either in the past or nothing is defined, or days are excluded ).
```

Ce log sera affiché qu'une fois par jour au max par éléments afin d'éviter de saturer les logs.

Logs d'ERROR concernant l'arrêt du démon s'il n'arrive pas à charger les données de rétention

Si le démon n'arrive pas à charger les données de rétention, il va potentiellement perdre des données comme les notifications actuelles, les downtimes ou les acknowledge, ce qui est très visible pour les utilisateurs.

En l'état, on préfère tuer le démon afin qu'un spare puisse prendre le relai et fonctionner normalement.

Quand un scheduler s'arrête pour cette raison, il loggue:

```
[YYYY-MM-DD HH:MM:SS] ERROR : [ scheduler-master ] [ RETENTION      ] Failed to load the retention data. Shutting down daemon: we prefer to shut the daemon and leave a spare take the role instead of start and loose data like notifications or downtimes.
```

Log d'ERROR quand on charge depuis la rétention un nombre trop important de vérifications sur les hôtes ou les checks

Afin d'éviter un crash du Scheduler après le chargement d'une mauvaise rétention avec un check ou un hôte qui possède trop de vérifications, on ignore les vérifications et on log le message suivant :

```
[YYYY-MM-DD HH:MM:SS] ERROR : [ scheduler-name ] [ RETENTION ] [ ANALYSE ] [ CHECKS_IN_PROGRESS ] [ monhote /check The check have too much executions (503 > warning limit=100). This can be a bug, please contact your support with a backup of your retention for verification. [ uuid=b3e7d3307d2211eca52d080027940ca8-f9c118de7d2311ec80c7080027940ca8 ]
```

Options du support Shinken

Forcer l'étalement des checks au démarrage

Si, suite à une demande du support, l'option pour forcer l'étalement des checks a été activée, le log suivant apparaîtra lors du premier chargement d'une configuration par le Scheduler:

[YYYY-MM-DD HH:MM:SS] INFO : [scheduler-name] [MAINTENANCE] [CHECK SPREAD OUT] Scheduler will force all check spread out