

Zone d'entête - Météo

Sommaire

[Icônes de notifications de changements](#)
[La notification sonore](#)
[La notification visuelle de clignotement](#)

Démarrage du démon

Au démarrage le démon affiche plusieurs logs contenant ses informations dont :

- ses limites systèmes en nombre de fichier ouvrables, et nombre de threads/processus

```
[2020-05-18 05:19:18] INFO : [daemon-master] [ SYSTEM ] System resource number of open files is
set to (soft:1024 / hard:1024 ) (from parameter max_file_descriptor_limit)
[2020-05-18 05:19:18] INFO : [daemon-master] [ SYSTEM ] System resource number of process
/threads is set to (soft:unlimited / hard:unlimited ) (set at system max values)
```

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

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

Si le serveur hébergeant le démon est surchargé en terme 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 vous aurez dans les logs l'entrée suivante:

```
2020-05-04 00:00:51 WARNING : [ LOGGER ]
2020-05-04 00:00:51 WARNING : [ LOGGER ]
-----
2020-05-04 00:00:51 WARNING : [ LOGGER ] [ WRITING ] The log write time is very high (1.87s). Please look at
your log disk performance.
2020-05-04 00:00:51 WARNING : [ LOGGER ]
-----
2020-05-04 00:00:51 WARNING : [ LOGGER ]
```

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 permet de suivre le chargement de la configuration de supervision entre l'Arbiter les schedulers jusqu'au interface : 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 log

```
[1589548527] INFO : [scheduler] [schedulerdaemon] [ CONFIGURATION ] [ NEW ] Configuration received [
configuration_part_id=configuration_part_id, configuration_uuid=configuration_uuid,
arbiter=arbiter_name, date=creation_date, ]
```

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

Exemple Scheduler réception d'une nouvelle configuration

```
[1589548527] INFO : [scheduler] [schedulerdaemon] [ CONFIGURATION ] [ NEW ] Configuration received [ configuration_part_id=1280, configuration_uuid=e551f7f93f2d45bfafae77fc302db7a2, arbiter=arbiter-master1, date=15-05-2020 15:13:38 ]
```

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

```
[2020-05-15 15:15:30] INFO : [scheduler-master1] [schedulerdaemon] [ CONFIGURATION ] [ NEW ] Loaded in [loading_time]s => [ configuration_part_id=configuration_part_id, configuration_uuid=configuration_uuid, date=creation_date, author=arbiter_name ]
```

- **configuration_part_id**: id de la partit de configuration géré par ce Scheduler (unique par Scheduler)
- **configuration_uuid**: uuid crée lors du démarrage de l'Arbiter qui correspond donc à l'id de la configuration géré par l'Arbiter
- **creation_date**: date du démarrage de l'Arbiter
- **arbiter_name**: nom de l'Arbiter qui a créé cette configuration
- **loading_time**: temps de chargement de la configuration

Exemple Scheduler chargement de la nouvelle configuration

```
[2020-05-15 15:15:30] INFO : [scheduler-master1] [schedulerdaemon] [ CONFIGURATION ] [ NEW ] Loaded in [1.31168293953]s => [ configuration_part_id=1280 configuration_uuid=e551f7f93f2d45bfafae77fc302db7a2, author=arbiter-master1, date=15-05-2020 15:13:38]
```

Logs DEBUG des escalades

Si le scheduler est démarré avec la variable d'environnement

```
SHINKEN_SUPPORT_ENABLE_ESCALATION_DEBUG=1 /etc/init.d/shinken-scheduler start
```

Alors il va afficher plus d'informations DEBUG sur les escalades afin de pouvoir mieux les suivre.

Choix de l'escalade

Savoir quelle notification est prise, on a ce log. Attention, seule la dernière ligne sera utile (une escalade ayant pris le pas sur la précédente):

```
[2020-05-15 15:15:30] DEBUG : ESCALATION: [NOM HOTE/CHECK] The escalation NOM_ESCALADE give us a valid start time (TEMPS). Looking if other escalations are giving us earlier time.'
```

Choix de l'intervalle

Savoir quelle intervalle de notification est pris, il faut là encore prendre la dernière ligne (si plusieurs escalades sont possibles au même moment):

```
[2020-05-15 15:15:30] DEBUG : ESCALATION: [NOM HOTE/CHECK] The escalation NOM_ESCALADE is eligible and with a short notification interval, so we are using its interval: VALEUR_INTERVAL_ESCALADE.
```

Escalade avec un intervalle de notification à 0

Quand une escalade n'a pas de notification interval (ceci va envoyer une seule notification) on est mis au courant par cette ligne:

```
[2020-05-15 15:15:30] DEBUG : ESCALATION: [NOM HOTE/CHECK] The escalation NOM_ESCALADE have no notification interval, so disabling next notification.
```

Erreur de cohérence des downtimes

Les downtimes ont une incohérence entre non identifiée qui consiste à ce qu'un downtime va être démarré deux fois, ce qui va poser soucis lors de son arrêt (l'hôte/check sera dans un état incohérent). Pour l'instant nous n'avons pas trouvé la source de l'incohérence, mais nous avons mis des protections pour les éviter et corriger.

Au chargement de la retention, on aura une entrée ERROR:

```
[2020-08-04 11:03:34] ERROR : [scheduler-master] [ DOWNTIME-INCOHERENCY ] The host moi 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 du downtime:

```
[2020-08-04 11:16:06] ERROR : [scheduler-master] [ 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 à un dev.

Exécution de commandes reçues d'un Receiver

Lorsque le Receiver envoie des commandes externes au Scheduler, il est possible que ce dernier n'ai 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, nous renvoyons cette entrée :

```
[1596621351] INFO : [scheduler] Get external commands from a Receiver but i am not ready. I do not receive configuration from Arbiter.
```

Dans le cas où les commandes ont pu être exécutées :

```
[2020-08-05 11:56:37] INFO : [scheduler-master] Running command receives from a Receiver
```