

dump_performance (top / ps / cpu / kernel / healthcheck)

Sommaire

[Mise en place](#)

Description

Les logs de la rétention MongoDB du Scheduler sont classés par catégorie afin de pouvoir différencier les types de log :

- Sauvegarde
- Chargement
- La suppression des lignes retentions obsolètes.

Sauvegarde en rétention

Pour la sauvegarde de la rétention, trois types de logs existent:

Section	Description
SAVE GLOBAL	Correspond au processus global de la sauvegarde
SAVE WORKERS	Corresponds à un sous-processus de SAVE GLOBAL, qui s'occupe de la file d'attente des différents workers de la sauvegarde
SAVE WORKER X	C'est un sous-processus de SAVE WORKERS, correspondant à un worker numéroté X qui permet de sauvegarder une partie des informations du Scheduler en base. Le nombre de workers est paramétrable dans les paramètres du module. (voir Rétention en base de donnée centralisée par royaume (Module MongoDBRetention))

SAVE GLOBAL

Les logs **SAVE GLOBAL** donnent des informations relatives au fonctionnement global du module ou de sa configuration.

Exemple

```
[2019-07-10 14:34:39] INFO : [scheduler-master] [ MongoDBRetention ] [ SAVE GLOBAL ] Starting to save retention data. [994:hosts] [994:checks] (Database used = mongodb://127.0.0.1/?safe=false, use ssh = False)
```

Erreurs

Les erreurs lors de la sauvegarde de la rétention sont aussi enregistrées dans les logs sous cette forme:

```
[2021-04-20 11:26:57] ERROR : [ scheduler-master ] [ MODULES-MANAGER ] The instance MongoDBRetention raised an error: [ SAVE GLOBAL ] FAILED Retention data could not be saved in mongodb. Total time 22.20 s. I disable it and set it to restart it later
```

Lors de la sauvegarde de la rétention, la base mongo est injoignable

```
[2021-04-20 11:26:57] ERROR : [ scheduler-master ] [ MODULES-MANAGER ] The instance MongoDBRetention raised an error: [ SAVE GLOBAL ] FAILED Retention data could not be saved in mongodb because mongo is unreachable. Total time 2.11s. I disable it and set it to restart it later
```

SAVE WORKERS

Les logs **SAVE WORKERS** donnent l'état de chaque worker de sa création à son succès/échec.

Exemple

```
[2019-07-10 14:34:44] INFO    : [scheduler-master] [ MongoDBRetention ] [ SAVE WORKERS           ]
Starting worker 0 with pid 14746. Try: 1/3
[2019-07-10 14:34:54] INFO    : [scheduler-master] [ MongoDBRetention ] [ SAVE WORKERS           ] The
worker 0 did SUCCESS (after 1 try)
```

SAVE WORKER X

Les logs **SAVE WORKER X** donne pour le worker ayant l'identifiant **X**, les statistiques sur les sauvegardes qu'il a effectuées : le nombre d'éléments, résultat et temps d'exécution.

Exemple

```
[2019-07-10 14:34:44] INFO    : [scheduler-master] [ MongoDBRetention ] [ SAVE WORKER 0           ] Will save
249 hosts and 249 checks
[2019-07-10 14:34:54] INFO    : [scheduler-master] [ MongoDBRetention ] [ SAVE WORKER 0           ] SUCCESS
did saved 249 hosts and 249 checks retention data into mongodb in 10.46s
```

Erreurs

En cas d'erreur, chaque worker essaiera de se lancer à nouveau en respectant le nombre de tentatives maximales définies dans le fichier de configuration du module. Si la rétention n'est pas sauvegardée après ces tentatives, le module sera en échec et le Scheduler s'arrêtera. Lorsque le SSH tunnel est activé :

Exemple d'erreurs

```
[2021-04-20 11:43:09] WARNING: [ scheduler-master ] [ MongoDBRetention ] [ SAVE WORKER 0           ] Mongo
connection failed 1/1 time, we will try again
[2021-04-20 11:43:10] ERROR   : [ scheduler-master ] [ MongoDBRetention ] [ SAVE WORKER 0           ] Mongo
connection failed 1/1 times, we stop trying
[2021-04-20 13:24:20] ERROR   : [ scheduler-master ] [ MongoDBRetention ] [ SAVE WORKER 0           ] Worker has an
error: [ Mongo connection failure : localhost:34925 ===(ssh tunnel)==> 192.168.1.132:22 ===(mongodb)==>
192.168.1.132:27017 ]
```

Sans le SSH tunnel

Exemple d'erreurs

```
[2021-04-20 11:43:09] WARNING: [ scheduler-master ] [ MongoDBRetention ] [ SAVE WORKER 0           ] Mongo
connection failed 1/1 time, we will try again
[2021-04-20 11:43:10] ERROR   : [ scheduler-master ] [ MongoDBRetention ] [ SAVE WORKER 0           ] Mongo
connection failed 1/1 times, we stop trying
[2021-04-20 12:06:15] ERROR   : [ scheduler-master ] [ MongoDBRetention ] [ SAVE WORKER 0           ] Worker has an
error: [ Mongo connection failure to mongodb://192.168.1.132/?safe=false ]
```

Chargement de la rétention

Les logs fournissent des informations liées au chargement de la rétention, permettant de suivre son avancée et l'état sur la connexion à Mongo.

Exemple

```
[2019-07-10 14:35:36] INFO    : [scheduler-master] [ MongodbRetention ] [ LOAD RETENTION      ] Starting
to load the retention data from mongodb. (Database used = mongodb://172.16.0.12/?safe=false, use ssh = False)
[2019-07-10 14:35:37] INFO    : [scheduler-master] [ MongodbRetention ] [ LOAD RETENTION      ] We found
994 hosts in the retention, took 0.374s.
[2019-07-10 14:35:37] INFO    : [scheduler-master] [ MongodbRetention ] [ LOAD RETENTION      ] We found
994 services in the retention, took 0.082s.
[2019-07-10 14:35:37] INFO    : [scheduler-master] [ MongodbRetention ] [ LOAD RETENTION      ] loaded in
0.457s ( 1988 elements ) which were saved in the previous retention phase ( => 1988 elements currently
managed by this scheduler. )
[2019-07-10 14:35:37] INFO    : [scheduler-master] [ MongodbRetention ] [ LOAD RETENTION      ] SUCCESS
Retention data loaded successfully in 0.662s.
```

Erreurs

Les erreurs lors du chargement de la rétention sont aussi enregistrées dans les logs sous cette forme:

Exemple

```
[2019-07-10 16:19:10] ERROR   : [scheduler-master] [ MongodbRetention ] [ LOAD RETENTION      ] The
mongodb connection failed to initialise. We cannot load the retention data
```

Suppression des anciennes rétentions

Les logs de suppression permettent de voir le nombre d'objets supprimés (triés par hôtes et checks) ainsi que la date à partir de laquelle la rétention est conservée.

Exemple avec des objets à supprimer

```
[2019-07-10 15:54:53] INFO    : [scheduler-master] [ MongodbRetention ] [ DELETE OLD RETENTION ] We will
delete all retention data that were saved before the 2019-07-07 13:54 UTC (3 days)
[2019-07-10 15:54:53] INFO    : [scheduler-master] [ MongodbRetention ] [ DELETE OLD RETENTION ] -
Deleting 994 hosts from old retention [1000 by 1000]
[2019-07-10 15:54:53] INFO    : [scheduler-master] [ MongodbRetention ] [ DELETE OLD RETENTION ] - 994 -
hosts deleted in 0.188s
[2019-07-10 15:54:53] INFO    : [scheduler-master] [ MongodbRetention ] [ DELETE OLD RETENTION ] -
Deleting 994 services from old retention [1000 by 1000]
[2019-07-10 15:54:53] INFO    : [scheduler-master] [ MongodbRetention ] [ DELETE OLD RETENTION ] - 994 -
services deleted in 0.091s
[2019-07-10 15:54:53] INFO    : [scheduler-master] [ MongodbRetention ] [ DELETE OLD RETENTION ] Total time
for deleting 1988 entries = 0.280s
```

Exemple sans objets à supprimer

```
[2019-07-10 14:35:13] INFO    : [scheduler-master] [ MongodbRetention ] [ DELETE OLD RETENTION ] We will
delete all retention data that were saved before the 2019-07-07 12:35 UTC (3 days)
[2019-07-10 14:35:16] INFO    : [scheduler-master] [ MongodbRetention ] [ DELETE OLD RETENTION ] There is
no data to delete
[2019-07-10 14:35:16] INFO    : [scheduler-master] [ MongodbRetention ] [ DELETE OLD RETENTION ] Total time
for deleting 0 entries = 0.019s
```

Erreurs

Si une erreur est rencontrée lors de la suppression, elles seront indiquées dans les logs, comme ceux-ci:

Exemple

```
[2019-07-10 16:19:10] ERROR : [scheduler-master] [ MongoddbRetention ] [ DELETE OLD RETENTION ] We have
been disconnected of mongo. Will retry [1/3]
[2019-07-10 16:19:10] ERROR : [scheduler-master] [ MongoddbRetention ] [ DELETE OLD RETENTION ] We have
been disconnected of mongo. Will retry [2/3]
[2019-07-10 16:19:10] ERROR : [scheduler-master] [ MongoddbRetention ] [ DELETE OLD RETENTION ] We have
been disconnected of mongo. Will retry [3/3]
[2019-07-10 16:19:10] ERROR : [scheduler-master] [ MongoddbRetention ] [ DELETE OLD RETENTION ] After 3
tries, we couldn't connect to mongo
```