

Procédure de résolution de l'alerte de fragmentation MongoDB dans CPS

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Informations générales](#)

[Problème](#)

[Procédure de résolution de l'alerte de fragmentation MongoDB](#)

Introduction

Ce document décrit la procédure pour résoudre l'alerte de fragmentation MongoDB dans Cisco Policy Suite (CPS).

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Linux
- CPS
- base de données mongole

Note: Cisco recommande que vous ayez un accès racine privilégié à l'interface de ligne de commande CPS.

Components Used

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- CPS 20.2
- MongoDB v3.6.17
- UCS (Unified Computing System)-B

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

Informations générales

CPS utilise MongoDB où les processus mongod qui s'exécutent sur les machines virtuelles (VM) Sessionmgr constituent sa structure de base DataBase.

Lorsque des documents sont déplacés ou supprimés, ils laissent des trous. MongoDB essaie de réutiliser ces trous pour de nouveaux documents quand c'est possible, mais avec le temps, il se retrouve avec beaucoup de trous lentement et régulièrement, qui ne peuvent pas être réutilisés parce que les documents ne peuvent pas y tenir. Cet effet, appelé fragmentation, est courant dans tous les systèmes qui allouent de la mémoire, y compris votre système d'exploitation.

La fragmentation a pour effet de gaspiller de l'espace. En raison du fait que MongoDB utilise des fichiers mappés en mémoire, toute fragmentation sur le disque se reflète également dans la fragmentation en RAM. Cela entraîne la création d'une quantité moindre de « plage de travail » dans la mémoire vive et entraîne l'échange d'un plus grand nombre de disques.

CPS prend en charge les KPI pour surveiller la fragmentation du niveau MongoDB par l'utilisation de Grafana et génère une alarme SNMP (Simple Network Management Protocol) lorsque le pourcentage de fragment MongoDB dépasse une valeur spécifiée.

LES `/etc/collectd.d/dbMonitorList.cfg` le fichier présent sur les machines virtuelles sessionmgr contient la liste des bases de données et leurs valeurs de pourcentage de seuil de fragmentation respectives. Par défaut, le seuil de fragmentation est de 40 %. La valeur du seuil de fragmentation par défaut peut être modifiée si nécessaire.

Les statistiques de fragmentation pour les bases de données `session_cache`, `sk_cache`, `diameter` et `Subscriber Profile Repository (SPR)` (par l'utilisation de membres principaux) peuvent être vérifiées avec cette commande :

```
[root@installer ~]# diagnostics.sh --get_frag
CPS Diagnostics HA Multi-Node Environment
-----

Ping check for qns03 Adding to IGNORED_HOSTS...[FAIL]
|-----|
|-----|
| Mongo:v3.6.17 DATABASE LEVEL FRAGMENTATION STATUS INFORMATION Date : 2022-09-17 07:19:29 |
| SET TYPE : HA [MEMBER_ROLE : PRIMARY] |
|-----|
|-----|
| setname dbName storageSize(MB) datasize(MB) indexSize(MB) fileSize(MB) derivedFS(MB) frag% |
|-----|
|-----|
| ADMIN:set06 |
| Status via sessionmgr01:27721 |
| set06 diameter 9.56 0.04 0.05 64.00 0 NoFrag |
|-----|
|-----|
| BALANCE:set02 |
| Status via sessionmgr01:27718 |
| set02 balance_mgmt db not found - - - - - |
|-----|
|-----|
| SESSION:set01 |
| Status via sessionmgr01:27717 |
| set01 session_cache 0.02 0.00 0.02 16.00 0 NoFrag |
|-----|
|-----|
```

```
| SESSION:set01 |
| Status via sessionmgr01:27717 |
| set01 sk_cache 0.02 0.00 0.01 16.00 0 NoFrag |
|-----|
|-----|
| SPR:set04 |
| Status via sessionmgr01:27720 |
| set04 spr 0.04 0.00 0.13 64.00 0 NoFrag |
|-----|
|-----|
[root@installer ~]#
```

Problème

Lorsque le pourcentage de fragmentation du membre principal du jeu de réplicas dépasse la valeur de fragmentation de seuil configurée, cette alarme est générée. Si la valeur de seuil n'est pas configurée, l'alarme est déclenchée si le pourcentage de fragmentation dépasse la valeur par défaut (40 %).

Exemple d'alerte « La fragmentation MongoPrimaryDB a dépassé la valeur de seuil » :

```
id=7100,values={sub_id=7107, event_host=sessionmgr01, status=down, msg=MongoPrimaryDB
fragmentation exceeded the threshold value, CURR_FRAG = 40%, THRESHOLD = 40% at
sessionmgr01:27717 for session_cac
```

Procédure de résolution de l'alerte de fragmentation MongoPrimaryDB

Pour réduire le pourcentage de fragmentation, réduisez la base de données lorsqu'une alarme est générée. Une fois la base de données réduite (le pourcentage de fragmentation diminue), une alarme claire est envoyée.

Cette procédure permet de résoudre l'alerte de fragmentation MongoPrimaryDB dans l'exemple fourni.

Étape 1. Exécutez cette commande à partir de Cluster Manager ou de pcrfclient afin de vérifier l'état des membres principaux et secondaires dans le jeu de réplicas.

```
#diagnostics.sh --get_r
```

```
|-----|
|-----|
|SESSION:set01a|
|Status via sessionmgr01:27717 sessionmgr02:27717 |
|Member-1-27717 : 192.168.29.14-ARBITER-pcrfclient01- ON-LINE--0| -----|
|Member-2-27717 : 192.168.29.35-PRIMARY-sessionmgr01- ON-LINE--3| -----|
|Member-3-27717 : 192.168.29.36-SECONDARY-sessionmgr02- ON-LINE--2| 1 sec|
|-----|
|-----|
```

Étape 2. Exécutez cette commande à partir de Cluster Manager ou de pcrfclient pour modifier la priorité de sessionmgr01 et en faire un membre secondaire.

```
#sh set_priority.sh --db session --replSet set01a --asc
```

Expected output in #diagnostics.sh --get_r

```
|-----|
|-----|
|SESSION:set01a|
|Status via sessionmgr02:27717 sessionmgr01:27717 |
|Member-1-27717 : 192.168.29.14-ARBITER-pcrfclient01- ON-LINE--0| -----|
|Member-2-27717 : 192.168.29.35-PRIMARY-sessionmgr02- ON-LINE--3| -----|
|Member-3-27717 : 192.168.29.36-SECONDARY-sessionmgr01- ON-LINE--2| 1 sec|
|-----|
|-----|
```

Note: Assurez-vous que sessionmgr01 n'est plus principal (diagnostics.sh --get_r) et qu'un membre principal est disponible pour le jeu de réplicas.

Étape 3. Exécutez cette commande à partir de Sessionmgr01 pour arrêter le client AIDO.

```
#monit stop aido_client
```

Étape 4. Exécutez cette commande à partir de Sessionmgr01 pour arrêter l'instance Mongo respective (portNum est le numéro de port du membre fragmenté).

Command syntax:

```
#/etc/init.d/sessionmgr-<portNum> stop
```

Example:

```
#/etc/init.d/sessionmgr-27717 stop
```

Étape 5. Afin de nettoyer le répertoire de base de données dans sessionmgr01, supprimez le répertoire de données du chemin mentionné par rapport à l'attribut --dbpath de la commande mongo. Exécutez cette commande à partir de Sessionmgr01 afin de récupérer la valeur (utilisez le portNum du membre fragmenté).

Note: Le numéro de port et les répertoires associés aux autres bases de données Sessionmgr étant différents, assurez-vous que vous disposez des répertoires appropriés pour nettoyer les autres bases de données Sessionmgr.

Command syntax:

```
#grep -w DBPATH= /etc/init.d/sessionmgr-<portNum>
```

Example:

```
#grep -w DBPATH= /etc/init.d/sessionmgr-27717
```

Sample Output: DBPATH=/var/data/sessions.1/a

Copy the DBPATH from output.

Command syntax:

```
#rm -rf <DBPATH>/*
```

Example:

```
#rm -rf /var/data/sessions.1/a/*
```

Étape 6. Exécutez cette commande à partir de Sessionmgr01 pour démarrer l'instance Mongo correspondante.

Command syntax:

```
#!/etc/init.d/sessionmgr-<portNum> start
```

Example:

```
#!/etc/init.d/sessionmgr-27717 start
```

Étape 7. Exécutez cette commande à partir de Sessionmgr01 pour démarrer le client AIDO.

```
#monit start aido_client
```

Étape 8. Exécutez cette commande à partir de Cluster Manager ou de pcrfclient pour réinitialiser les priorités des membres du jeu de réplicas.

```
#sh set_priority.sh --db session --replSet set01a
```

Étape 9. Exécutez cette commande à partir de Cluster Manager ou de pcrfclient pour vérifier l'état des membres principaux et secondaires dans le jeu de réplicas.

```
#diagnostics.sh --get_r
```

```
|-----|
|-----|
|SESSION:set01a|
|Status via sessionmgr01:27717 sessionmgr02:27717 |
|Member-1-27717 : 192.168.29.14-ARBITER-pcrfclient01- ON-LINE--0| -----|
|Member-2-27717 : 192.168.29.35-PRIMARY-sessionmgr01- ON-LINE--3| -----|
|Member-3-27717 : 192.168.29.36-SECONDARY-sessionmgr02- ON-LINE--2| 1 sec|
|-----|
|-----|
```

À propos de cette traduction

Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.