

Fehlerbehebung bei Fehlern bei mongoten Instanzen aufgrund einer erhöhten Platzauslastung von DATA_PATH

Inhalt

[Einleitung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Hintergrundinformationen](#)

[Problem](#)

[Wiederherstellen der Mongod-Instanz in Sessionmgr](#)

Einleitung

In diesem Dokument wird beschrieben, wie Sie das Problem der mongoten Instanzenfehler in der Cisco Policy Suite (CPS)-Sitzung aufgrund der erhöhten Platzauslastung DATA_PATH beheben können.

Voraussetzungen

Anforderungen

Cisco empfiehlt, dass Sie über Kenntnisse in folgenden Bereichen verfügen:

- Linux
- CPS
- MongoDB

Verwendete Komponenten

Die Informationen in diesem Dokument basierend auf folgenden Software- und Hardware-Versionen:

- CPS 20,2
- MongoDB v3.6.17
- UCS B

Die Informationen in diesem Dokument beziehen sich auf Geräte in einer speziell eingerichteten Testumgebung. Alle Geräte, die in diesem Dokument benutzt wurden, begannen mit einer gelöschten (Nichterfüllungs) Konfiguration. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die potenziellen Auswirkungen eines Befehls verstehen.

Hintergrundinformationen

CPS verwendet MongoDB, wo Mongod-Prozesse auf virtuellen Systemen von Sessionmgr (VMs) ausgeführt werden, um die grundlegende Datenbankstruktur zu bilden.

Mehrere mongote Instanzen werden auf einer Sitzung ausgeführt, und jeder dieser Instanzen wurde unterschiedliche Portnummern zugewiesen. Diese mongoten Instanzen nehmen an verschiedenen Replica Sets teil.

Problem

Wenn eine bestimmte Mongolinanz aufgrund der erhöhten Nutzung des DATA_PATH-Speicherplatzes des zugehörigen DATA PATH abbricht, wird dies auch in der Diagnose für diese SessionMgr bemerkt. Die Verbindung zum bestimmten Port ist fehlgeschlagen, und die /var/data/sessions.X Partition wird zu 100 % genutzt. Daher wird diese Mongolinanz in den OFF-LINE-Status im entsprechenden Replica Set verschoben. Anschließend wird der Teilnahmestatus in diesem Replica Set UNKNOWN.

In der Diagnose wird ein Beispielfehler ausgegeben. Geben Sie `diagnostics.sh` aus ClusterManager oder `prfclient`, um den aktuellen Status von mongod und Replica Set zu überprüfen.

```
Could not connect to port 27718 on sessionmgr02 (set02)...[FAIL]
Disk usage on sessionmgr02...[FAIL]
Disk usage is above critical threshold (97%) on sessionmgr02.
Results of: ssh root@sessionmgr02 -x 'df -hP -x iso9660'
```

```
-----
Filesystem Size Used Avail Use% Mounted on
/dev/sda2 95G 28G 62G 32% /
tmpfs 48G 0 48G 0% /dev/shm
tmpfs 57G 0 57G 0% /var/data/sessions.1
tmpfs 12G 12G 0 100% /var/data/sessions.2
-----
```

```
-----|
| BALANCE:set02 |
| Status via arbitervip:27718 sessionmgr01:27718 |
| Member-1 - 27718 : - UNKNOWN - sessionmgr02 - OFF-LINE - 19003 days - 2 |
| Member-2 - 27718 : - PRIMARY - sessionmgr01 - ON-LINE - ----- - 3 |
| Member-3 - 27718 : 192.168.10.146 - ARBITER - arbitervip - ON-LINE - ----- - 0 |
|-----|
```

Wiederherstellen der Mongod-Instanz in Sessionmgr

Im Abschnitt wird die Vorgehensweise zum Wiederherstellen der Mongod-Instanz in sessionmgr beschrieben, wenn sie aufgrund eines erhöhten Platzverbrauchs von DATA_PATH ausgefallen ist.

Bevor Sie dieses Verfahren beginnen, müssen Sie über folgende Berechtigungen verfügen:

- Root-Zugriff auf die CPS-CLI
- "qns-svn"-Benutzerzugriff auf die CPS-GUIs - Policy Builder und CPS Central

Hier ist das Verfahren für die Sitzungen mg02 und Port 27718 vorgesehen, die Teil von set02 sind.

1. Melden Sie sich bei der entsprechenden Sitzung an.

2. Geben Sie diesen Befehl ein, um die Partition zu identifizieren, in der die Daten für diesen bestimmten set02 gespeichert wurden.

```
[root@dc1-sessionmgr02 ~]# cat /etc/broadhop/mongoConfig.cfg | grep -A6 set02 | grep "DATA_PATH"
ARBITER_DATA_PATH=/var/data/sessions.2
DATA_PATH=/var/data/sessions.2
```

3. Geben Sie diesen Befehl ein, um zu überprüfen, ob die `aido_client` Prozess ist vorhanden oder nicht.

```
[root@dc1-sessionmgr02 ~]# monsum
Monit 5.26.0 uptime: 11d 2h 9m
```

```
Service Name  Status  Type
-----
dc1-sessionmgr02  OK    System
whisper  OK    Process
snmpd  OK    Process
memcached  OK    Process
collectd  OK    Process
auditrpmsh.sh  OK    Process
aido_client  OK    Process
primary_db_frag  OK    Program
cpu_load_monitor  OK    Program
cpu_load_trap  OK    Program
gen_low_mem_trap  OK    Program
```

4. Wenn `aido_client` vorhanden ist, geben Sie `monit stop aido_client` um sie zu stoppen.

5. Geben Sie diesen Befehl ein, um zu überprüfen, ob der jeweilige Mongod-Instanzprozess noch aktiv ist oder nicht.

```
[root@dc1-sessionmgr02 ~]# ps -ef | grep 27718
root 12292 11114 0 02:05 pts/0 00:00:00 grep --color=auto 27718
root 19620 1 0 2021 ? 01:36:51 /usr/bin/mongod --ipv6 --syncdelay 1 --slowms 500 --storageEngine
mmapv1 --bind_ip_all --port 27718 --dbpath=/var/data/sessions.2 --replSet set02 --fork --pidfilepath
/var/run/sessionmgr-27718.pid --oplogSize 5120 --logpath /var/log/mongodb-27718.log --logappend --quiet
[root@dc1-sessionmgr02 ~]#
```

6. Wenn die Mongoinstanz noch aktiv ist, geben Sie diesen Befehl ein, um sie zu beenden.

```
[root@dc1-sessionmgr02 ~]# /etc/init.d/sessionmgr-27718 stop
```

```
Stopping sessionmgr-27718 (via systemctl): [ OK ]
[root@dc1-sessionmgr02 ~]#
```

7. Navigieren Sie zu DATA_PATH, das Sie in Schritt 1 erhalten haben.

```
[root@dc1-sessionmgr02 ~]# cd /var/data/sessions.2
[root@dc1-sessionmgr02 sessions.2]# ls -lrt
total 6616100
-rw----- 1 root root 16777216 Jun 22 2018 admin.ns
-rw----- 1 root root 67108864 Jun 22 2018 admin.0
-rw----- 1 root root 69 Nov 10 07:27 storage.bson
-rw----- 1 root root 16777216 Nov 10 07:27 vouchers.ns
-rw----- 1 root root 67108864 Nov 10 07:27 vouchers.0
-rw----- 1 root root 2146435072 Nov 10 07:27 local.2
drwx----- 2 root root 4096 Nov 10 07:27 local
-rw----- 1 root root 67108864 Nov 10 07:27 local.0
-rw----- 1 root root 16777216 Jan 7 14:38 config.ns
-rw----- 1 root root 67108864 Jan 7 14:38 config.0
-rw----- 1 root root 16777216 Jan 11 02:06 local.ns
-rw----- 1 root root 2146435072 Jan 11 02:06 local.1
drwx----- 2 root root 4096 Jan 11 02:06 diagnostic.data
-rw----- 1 root root 2146435072 Jan 11 02:06 local.3
-rw----- 1 root root 0 Jan 11 02:07 mongod.lock
drwx----- 2 root root 4096 Jan 11 02:08 journal
[root@dc1-sessionmgr02 sessions.2]#
```

8. Geben Sie den Befehl ein `rm -rf *`, um DATA_PATH zu löschen.

9. Geben Sie diesen Befehl ein, um die Mongod-Instanz zu starten. Dieser Befehl kann einige Minuten in Anspruch nehmen.

```
[root@dc1-sessionmgr02 ~]# /etc/init.d/sessionmgr-27718 start
Starting sessionmgr-27718 (via systemctl): [ OK ]
[root@dc1-sessionmgr02 ~]#
```

10. Wenn Sie `aido_client` in Schritt 3 eingeben, `monit start adio_client` um es erneut zu starten.

11. Geben Sie `diagnostics.sh` aus ClusterManager oder pcrfclient, um zu bestätigen, dass die jeweilige Mongod-Instanz wiederhergestellt wurde und zu ON-LINE in Replica Set wurde.

```
|-----|
----|
| BALANCE:set02 |
| Status via arbitervip:27718 sessionmgr01:27718 sessionmgr02:27718 |
| Member-1 - 27718 : - SECONDARY - sessionmgr02 - ON-LINE - 0 sec - 2 |
| Member-2 - 27718 : - PRIMARY - sessionmgr01 - ON-LINE - ----- - 3 |
| Member-3 - 27718 : XX.XX.XX.XX - ARBITER - arbitervip - ON-LINE - ----- - 0 |
|-----|
----|
```