

# 處理CPS Mongo複製副本集中損壞的MongoDB的過程

## 目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[背景資訊](#)

[問題](#)

[修復損壞的資料庫成員的過程](#)

## 簡介

本文檔介紹在思科策略套件(CPS)副本集中處理損壞的蒙哥資料庫(DB)的過程。

## 必要條件

### 需求

思科建議您瞭解以下主題：

- Linux
- CPS
- MongoDB

**附註：** 思科建議您必須具有對CPS CLI的Root訪問許可權。

### 採用元件

本文中的資訊係根據以下軟體和硬體版本：

- CPS 20.2
- MongoDB v3.6.17
- UCS-B

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

## 背景資訊

MongoDB是一個源可用的跨平台面向文檔的資料庫(DB)程式。分類為NoSQL資料庫程式。MongoDB廣泛用於CPS中，用於管理其不同型別的資料庫，如SESSION、Subscriber Profile Repository(SPR)、Balance等。

# 問題

當aido\_client在sessionmgr內仍處於活動狀態時，當您執行不正確的資料庫碎片整理時，MongoDB會損壞。

這導致MongoDB將資料儲存在記憶體中，但無法在資料庫路徑上本地寫入資料。

如果在受影響的副本集上重新啟動主成員（mongo例項）或重新啟動sessionmgr VM，則可能導致資料丟失。

為了瞭解某個資料庫成員看起來是如何損壞的，您可以登入到一個有問題的成員，並執行提供的檢查。

步驟1.運行命令show dbs時，未返回資料庫清單的輸出。但是，當您檢查您所瞭解的資料庫內的計數時，它將返回該計數。

```
[root@lab-1-pcrfclient01 ~]# mongo --host sessionmgr05:27737
MongoDB shell version v3.6.17
connect to: mongodb://sessionmgr05:27737/?gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("a8f9b0eb-6e78-4bcd-bd63-60a9a9d813d0") }
MongoDB server version: 3.6.17
Server has startup warnings:
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten]
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten] ** WARNING: Access control is not
enabled for the database.
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten] **           Read and write access to
data and configuration is unrestricted.
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten] **
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten]
2022-03-09T00:53:26.949-0300 I REPL [replexec-0]
2022-03-09T00:53:26.949-0300 I REPL [replexec-0] ** WARNING: This replica set uses arbiters,
but readConcern:majority is enabled
2022-03-09T00:53:26.949-0300 I REPL [replexec-0] **           for this node. This is not a
recommended configuration. Please see
2022-03-09T00:53:26.949-0300 I REPL [replexec-0] **
2022-03-09T00:53:26.949-0300 I REPL [replexec-0]
set01e:PRIMARY>
set01e:PRIMARY> show dbs ## "no dbs reported"
set01e:PRIMARY> use session_cache ## "Switched to a known DB"
switched to db session_cache
set01e:PRIMARY> db.session.count()
223037 ## "DB has the content inside, hence the total record count is shown"
set01e:PRIMARY> use session_cache_2
switched to db session_cache_2
set01e:PRIMARY> db.session.count()
223643
set01e:PRIMARY> use session_cache_3
switched to db session_cache_3
set01e:PRIMARY> db.session.count()
222939
set01e:PRIMARY> use session_cache_4
switched to db session_cache_4
set01e:PRIMARY> db.session.count()
223692
set01e:PRIMARY>
set01e:PRIMARY> exit
bye
```

步驟2.運行diagnostics.sh —get\_shard時，應用程式分片顯示資料。它實際上儲存在記憶體中，而

不是在Sessionmgr虛擬機器(VM)的DBPATH中。

```
[root@lab-1-pcrfclient01 ~]# diagnostics.sh --get_shard
CPS Diagnostics GR Multi-Node Environment
```

```
|-----|
|-----|
|                                     SHARD STATUS INFORMATION
|      Date : 2022-03-09 11:00:23      |
|-----|
|-----|
```

Shard Id	Mongo DB	State	Backup DB	Removed	Session Count
43	sessionmgr01:27717/session_cache	online	false	false	223873
1	sessionmgr01:27717/session_cache_2	online	false	false	222918
2	sessionmgr01:27717/session_cache_3	online	false	false	223720
3	sessionmgr01:27717/session_cache_4	online	false	false	223393
8	sessionmgr05:27737/session_cache	online	false	false	223188
9	sessionmgr05:27737/session_cache_2	online	false	false	223554
10	sessionmgr05:27737/session_cache_3	online	false	false	222920
11	sessionmgr05:27737/session_cache_4	online	false	false	223562
12	sessionmgr07:27747/session_cache	online	false	false	222663
13	sessionmgr07:27747/session_cache_2	online	false	false	222599
14	sessionmgr07:27747/session_cache_3	online	false	false	222475
15	sessionmgr07:27747/session_cache_4	online	false	false	223446
16	sessionmgr09:27757/session_cache	online	false	false	223246
17	sessionmgr09:27757/session_cache_2	online	false	false	223669
18	sessionmgr09:27757/session_cache_3	online	false	false	223711
19	sessionmgr09:27757/session_cache_4	online	false	false	223311
35	sessionmgr13:27717/session_cache	online	true	false	0
36	sessionmgr13:27717/session_cache_2	online	true	false	0
37	sessionmgr13:27717/session_cache_3	online	true	false	0
38	sessionmgr13:27717/session_cache_4	online	true	false	0

Rebalance Status: Rebalanced

步驟3.此輸出顯示DB PATH中沒有應儲存實際資料的內容。

```
[SESSION-SET3]
SETNAME=set01e
OPLOG_SIZE=5120
ARBITER=lab-1-arb-sessmgr15:27737
ARBITER_DATA_PATH=/var/data/sessions.1/set01e
PRIMARY-MEMBERS
MEMBER1=lab-1-sessionmgr05:27737
MEMBER2=lab-1-sessionmgr06:27737
SECONDARY-MEMBERS
MEMBER3=lab-2-sessionmgr05:27737
MEMBER4=lab-2-sessionmgr06:27737
DATA_PATH=/var/data/sessions.1/set01e ## "DB DATA Path of set01e replicaset"
[SESSION-SET3-END]
```

安全殼層(SSH)到關聯的sessionmgr並導航到蒙哥配置中提到的DB\_PATH。您可以看到DB\_PATH中的內容為空。

```
[root@lab-1-sessionmgr05 ~]# cd /var/data/sessions.1/set01e
[root@lab-1-sessionmgr05 ~]# ls -lrt
total 0
[root@lab-1-sessionmgr05 ~]#
```

通過這些檢查，您可以得出結論MongoDB已損壞。

## 修復損壞的資料庫成員的過程

步驟1.通過SSH連線到有問題的副本集的主要成員。

步驟2.停止aido\_client ( 確保在屬於set01e的副本集的所有成員上停止aido客戶端 )。

步驟3.連線到set01e的mongo shell並運行這些步驟。

```
# mongo --port 27737
# show dbs                               # Ensure this returns empty output.
# use admin
# db.repairDatabase()
# use config
# db.repairDatabase()
# exit
```

```
[root@lab-1-sessionmgr05 set01e]# mongo --port 27737
MongoDB shell version v3.6.17
connect to: mongod://127.0.0.1:27737/?gssapiServiceName=mongod
Implicit session: session { "id" : UUID("ff9df861-0b42-4e8a-99c1-3583670e1926") }
MongoDB server version: 3.6.17
Server has startup warnings:
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten]
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten] ** WARNING: Access control is not
enabled for the database.
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten] **           Read and write access to
data and configuration is unrestricted.
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten] **
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten] **
2022-03-09T00:53:26.949-0300 I REPL [replexec-0]
2022-03-09T00:53:26.949-0300 I REPL [replexec-0] ** WARNING: This replica set uses arbiters,
but readConcern:majority is enabled
2022-03-09T00:53:26.949-0300 I REPL [replexec-0] **           for this node. This is not a
recommended configuration. Please see
2022-03-09T00:53:26.949-0300 I REPL [replexec-0] **
2022-03-09T00:53:26.949-0300 I REPL [replexec-0]
set01e:PRIMARY> use admin
switched to db admin
set01e:PRIMARY> db.repairDatabase()
{
  "ok" : 1,
  "operationTime" : Timestamp(1647319246, 352),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1647319246, 352),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
set01e:PRIMARY>
set01e:PRIMARY> use config
switched to db config
set01e:PRIMARY> db.repairDatabase()
{
  "ok" : 1,
  "operationTime" : Timestamp(1647319301, 218),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1647319301, 218),
```

```

"signature" : {
"hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
"keyId" : NumberLong(0)
}
}
}
set01e:PRIMARY> show dbs
admin    0.031GB
config  0.031GB
set01e:PRIMARY> exit

```

步驟4.重新連線同一複製副本即時消息，並在所有session\_cache\_dbs上執行這些命令。下面介紹了session\_cache DB的示例。

```

# mongo --port 27737
# use session_cache
# db.session.count()           # Use this to check that session counts are still intact
# db.stats(1024*1024*1024)    # Use this to verify that the storage size is proper
# db.repairDatabase()
# exit

```

```

[root@lab-1-sessionmgr05 set01e]# mongo --port 27737
MongoDB shell version v3.6.17
connect to: mongodb://127.0.0.1:27737/?gssapiServiceName=mongodb
Implicit session: session { "id" : UUID("73794d11-0785-4520-ba82-19f0d2bba338") }
MongoDB server version: 3.6.17
Server has startup warnings:
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten]
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten] ** WARNING: Access control is not
enabled for the database.
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten] **           Read and write access to
data and configuration is unrestricted.
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten] **
2022-03-09T00:53:26.910-0300 I CONTROL [initandlisten]
2022-03-09T00:53:26.949-0300 I REPL   [replexec-0]
2022-03-09T00:53:26.949-0300 I REPL   [replexec-0] ** WARNING: This replica set uses arbiters,
but readConcern:majority is enabled
2022-03-09T00:53:26.949-0300 I REPL   [replexec-0] **           for this node. This is not a
recommended configuration. Please see
2022-03-09T00:53:26.949-0300 I REPL   [replexec-0] **
2022-03-09T00:53:26.949-0300 I REPL   [replexec-0]
set01e:PRIMARY>
set01e:PRIMARY>
set01e:PRIMARY>
set01e:PRIMARY> show dbs
admin    0.031GB
config  0.031GB
set01e:PRIMARY> use session_cache
switched to db session_cache
set01e:PRIMARY>
set01e:PRIMARY> db.stats(1024*1024*1024)
{
"db" : "session_cache",
"collections" : 3,
"views" : 0,
"objects" : 212467,
"avgObjSize" : 8175.252062673262,
"dataSize" : 1.6176805645227432,
"storageSize" : 2.471107453107834,
"numExtents" : 22,
"indexes" : 3,
"indexSize" : 0.30870679020881653,
"fileSize" : 0,

```

```

"nsSizeMB" : 16,
"extentFreeList" : {
"num" : 0,
"totalSize" : 0
},
"dataFileVersion" : {
"major" : 4,
"minor" : 22
},
"fsUsedSize" : 38.36811065673828,
"fsTotalSize" : 47.044921875,
"ok" : 1,
"operationTime" : Timestamp(1647321405, 102),
"$clusterTime" : {
"clusterTime" : Timestamp(1647321405, 103),
"signature" : {
"hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
"keyId" : NumberLong(0)
}
}
}
set01e:PRIMARY> db.repairDatabase()
{
"ok" : 1,
"operationTime" : Timestamp(1647321444, 84),
"$clusterTime" : {
"clusterTime" : Timestamp(1647321444, 84),
"signature" : {
"hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
"keyId" : NumberLong(0)
}
}
}
set01e:PRIMARY> show dbs
admin          0.031GB
config         0.031GB
session_cache  2.499GB

```

**附註：**對session\_cache資料庫的其餘部分重複步驟4。

**步驟5.**確保show dbs現在列出將同一個mongo例項連線回時的所有DB。

```

mongo --port 27737
set01e:PRIMARY> show dbs
admin 0.031GB
config 0.031GB
session_cache 2.499GB
session_cache_2 2.499GB
session_cache_3 2.499GB
session_cache_4 2.499GB

```

**步驟6.**確保資料庫路徑現在包含sessionmgr本地的所有資料。您可以檢查副本集的相應資料路徑。在本例中為/var/data/sessions.1/set01e。

```

[root@lab-1-sessionmgr05 set01~]# cd /var/data/sessions.1/set01e
[root@lab-1-sessionmgr05 set01e]# ls
admin session_cache session_cache_2.1 session_cache_2.7 session_cache_3.1 session_cache_3.7
session_cache_4.1 session_cache_4.7 session_cache.8

```

```
admin.0 session_cache.0 session_cache_2.2 session_cache_2.8 session_cache_3.2 session_cache_3.8
session_cache_4.2 session_cache_4.8 session_cache.ns
admin.ns session_cache.1 session_cache_2.3 session_cache_2.ns session_cache_3.3
session_cache_3.ns session_cache_4.3 session_cache_4.ns _tmp
config session_cache.2 session_cache_2.4 session_cache.3 session_cache_3.4 session_cache.4
session_cache_4.4 session_cache.5
config.0 session_cache_2 session_cache_2.5 session_cache_3 session_cache_3.5 session_cache_4
session_cache_4.5 session_cache.6
config.ns session_cache_2.0 session_cache_2.6 session_cache_3.0 session_cache_3.6
session_cache_4.0 session_cache_4.6 session_cache.7
```

步驟7.使用SSH連線到同一站點輔助成員並執行與主成員的資料路徑的本地同步。

```
ssh to lab-1-sessionmgr06 (Secondary member)
Ensure to stop aido_client

# monit stop aido_client

Ensure to stop mongo processes

# /etc/init.d/sessionmgr-27737 stop # Wait for 10 seconds and start the service
```

back on

確保資料路徑/var/data/sessions.1/set01e為空，如果不是，則使用rm -rf /var/data/sessions.1/set01e/\*刪除這些路徑，然後啟動mongo進程。

```
# /etc/init.d/sessionmgr-27737 start
```

```
[root@lab-1-sessionmgr06 ~]# monit stop aido_client
[root@lab-1-sessionmgr06 ~]# monit status aido_client
Monit 5.26.0 uptime: 52d 20h 59m
```

```
Process 'aido_client'
status Not monitored
monitoring status Not monitored
monitoring mode active
on reboot start
data collected Wed, 23 Mar 2022 08:08:46
```

```
[root@lab-1-sessionmgr06 ~]#
[root@lab-1-sessionmgr06 ~]# /etc/init.d/sessionmgr-27737 stop
stop sessionmgr-27737 (via systemctl): [ OK ]
[root@lab-1-sessionmgr06 ~]# rm -rf /var/data/sessions.1/set01e/*
[root@lab-1-sessionmgr06 ~]# cd /var/data/sessions.1/set01e/
[root@lab-1-sessionmgr06 set01e]# ls
[root@lab-1-sessionmgr06 set01e]#
[root@lab-1-sessionmgr06 set01e]# /etc/init.d/sessionmgr-27737 start
Starting sessionmgr-27737 (via systemctl): [ OK ]
```

步驟8.驗證現在是否已將資料本地複製到/var/data/sessions.1/set01e。

```
[root@lab-1-sessionmgr06 ~]# cd /var/data/sessions.1/set01e/
[root@lab-1-sessionmgr06 set01e]# ls
admin.0 local.1 local.3 local.7 mongod.lock session_cache_2.3 session_cache_2.7
session_cache_3.1 session_cache_3.5 session_cache_3.ns
admin.ns local.10 local.4 local.8 session_cache_2.0 session_cache_2.4 session_cache_2.8
session_cache_3.2 session_cache_3.6 storage.bson
diagnostic.data local.11 local.5 local.9 session_cache_2.1 session_cache_2.5 session_cache_2.ns
session_cache_3.3 session_cache_3.7 _tmp
local.0 local.2 local.6 local.ns session_cache_2.2 session_cache_2.6 session_cache_3.0
```

```
session_cache_3.4 session_cache_3.8
[root@lab-1-sessionmgr06 set01e]#
```

**附註：**對地理站點輔助成員重複步驟7和步驟8。在實驗室中，成員是lab-2-sessionmgr05和lab-2-sessionmgr06。

步驟9.恢復所有輔助DB（本地和地理站點）後，重新啟動主成員上的mongo服務。

```
[root@lab-1-sessionmgr05 ~]# /etc/init.d/sessionmgr-27737 stop
stop sessionmgr-27737 (via systemctl): [ OK ]
```

等待10秒，並確認主交換機切換成功。

```
[root@lab-1-sessionmgr06 ~]# mongo --port 27737
MongoDB shell version v3.6.17
connect to: mongod://127.0.0.1:27737/?gssapiServiceName=mongod
Implicit session: session { "id" : UUID("ba8e49fa-ad0f-4ac6-8ef8-b4da0a88fe33") }
MongoDB server version: 3.6.17
Server has startup warnings:
2022-03-15T02:54:29.546-0300 I CONTROL [initandlisten]
2022-03-15T02:54:29.546-0300 I CONTROL [initandlisten] ** WARNING: Access control is not enabled
for the database.
2022-03-15T02:54:29.546-0300 I CONTROL [initandlisten] ** Read and write access to data and
configuration is unrestricted.
2022-03-15T02:54:29.546-0300 I CONTROL [initandlisten] **
2022-03-15T02:54:29.546-0300 I CONTROL [initandlisten]
set01e:PRIMARY>
set01e:PRIMARY>
set01e:PRIMARY> show dbs
admin 0.031GB
config 0.031GB
local 5.029GB
session_cache 2.499GB
session_cache_2 2.499GB
session_cache_3 2.499GB
session_cache_4 2.499GB
set01e:PRIMARY> show dbs
admin 0.031GB
config 0.031GB
local 5.029GB
session_cache 2.499GB
session_cache_2 2.499GB
session_cache_3 2.499GB
session_cache_4 2.499GB
set01e:PRIMARY> rs.status()
{
  "set" : "set01e",
  "date" : ISODate("2022-03-15T06:13:19.991Z"),
  "myState" : 1,
  "term" : NumberLong(36),
  "syncingTo" : "",
  "syncSourceHost" : "",
  "syncSourceId" : -1,
  "heartbeatIntervalMillis" : NumberLong(2000),
  "optimes" : {
    "lastCommittedOpTime" : {
      "ts" : Timestamp(1647324799, 335),
      "t" : NumberLong(36)
    }
  },
}
```



```
"readConcernMajorityOpTime" : {
"ts" : Timestamp(1647324799, 335),
"t" : NumberLong(36)
},
"appliedOpTime" : {
"ts" : Timestamp(1647324799, 338),
"t" : NumberLong(36)
},
"durableOpTime" : {
"ts" : Timestamp(0, 0),
"t" : NumberLong(-1)
}
},
"members" : [
{
"_id" : 0,
"name" : "lab-2-sessionmgr06:27737",
"health" : 1,
"state" : 2,
"stateStr" : "SECONDARY",
"uptime" : 486,
"optime" : {
"ts" : Timestamp(1647324799, 94),
"t" : NumberLong(36)
},
"optimeDurable" : {
"ts" : Timestamp(0, 0),
"t" : NumberLong(-1)
},
"optimeDate" : ISODate("2022-03-15T06:13:19Z"),
"optimeDurableDate" : ISODate("1970-01-01T00:00:00Z"),
"lastHeartbeat" : ISODate("2022-03-15T06:13:19.267Z"),
"lastHeartbeatRecv" : ISODate("2022-03-15T06:13:18.270Z"),
"pingMs" : NumberLong(0),
"lastHeartbeatMessage" : "",
"syncingTo" : "lab-1-sessionmgr06:27737",
"syncSourceHost" : "lab-1-sessionmgr06:27737",
"syncSourceId" : 4,
"infoMessage" : "",
"configVersion" : 8
},
{
"_id" : 1,
"name" : "lab-1-sessionmgr05:27737",
"health" : 1,
"state" : 2,
"stateStr" : "SECONDARY",
"uptime" : 885,
"optime" : {
"ts" : Timestamp(1647324799, 96),
"t" : NumberLong(36)
},
"optimeDurable" : {
"ts" : Timestamp(0, 0),
"t" : NumberLong(-1)
},
"optimeDate" : ISODate("2022-03-15T06:13:19Z"),
"optimeDurableDate" : ISODate("1970-01-01T00:00:00Z"),
"lastHeartbeat" : ISODate("2022-03-15T06:13:19.270Z"),
"lastHeartbeatRecv" : ISODate("2022-03-15T06:13:18.270Z"),
"pingMs" : NumberLong(0),
"lastHeartbeatMessage" : "",
"syncingTo" : "lab-1-sessionmgr06:27737",
"syncSourceHost" : "lab-1-sessionmgr06:27737",
```

```
"syncSourceId" : 4,
"infoMessage" : "",
"configVersion" : 8
},
{
  "_id" : 2,
"name" : "lab-1-arb-sessmgr15:27737",
"health" : 1,
"state" : 7,
"stateStr" : "ARBITER",
"uptime" : 1130,
"lastHeartbeat" : ISODate("2022-03-15T06:13:19.240Z"),
"lastHeartbeatRecv" : ISODate("2022-03-15T06:13:18.856Z"),
"pingMs" : NumberLong(0),
"lastHeartbeatMessage" : "",
"syncingTo" : "",
"syncSourceHost" : "",
"syncSourceId" : -1,
"infoMessage" : "",
"configVersion" : 8
},
{
  "_id" : 3,
"name" : "lab-1-sessionmgr05:27737",
"health" : 0,
"state" : 8,
"stateStr" : "(not reachable/healthy)",
"uptime" : 0,
"optime" : {
  "ts" : Timestamp(0, 0),
  "t" : NumberLong(-1)
},
"optimeDurable" : {
  "ts" : Timestamp(0, 0),
  "t" : NumberLong(-1)
},
"optimeDate" : ISODate("1970-01-01T00:00:00Z"),
"optimeDurableDate" : ISODate("1970-01-01T00:00:00Z"),
"lastHeartbeat" : ISODate("2022-03-15T06:13:19.299Z"),
"lastHeartbeatRecv" : ISODate("2022-03-15T06:11:58.086Z"),
"pingMs" : NumberLong(0),
"lastHeartbeatMessage" : "Connection refused",
"syncingTo" : "",
"syncSourceHost" : "",
"syncSourceId" : -1,
"infoMessage" : "",
"configVersion" : -1
},
{
  "_id" : 4,
"name" : "lab-1-sessionmgr06:27737",
"health" : 1,
"state" : 1,
"stateStr" : "PRIMARY",
"uptime" : 1130,
"optime" : {
  "ts" : Timestamp(1647324799, 338),
  "t" : NumberLong(36)
},
"optimeDate" : ISODate("2022-03-15T06:13:19Z"),
"syncingTo" : "",
"syncSourceHost" : "",
"syncSourceId" : -1,
"infoMessage" : "",
```

```
"electionTime" : Timestamp(1647324719, 72),
"electionDate" : ISODate("2022-03-15T06:11:59Z"),
"configVersion" : 8,
"self" : true,
"lastHeartbeatMessage" : ""
}
],
"ok" : 1,
"operationTime" : Timestamp(1647324799, 338),
"$clusterTime" : {
"clusterTime" : Timestamp(1647324799, 338),
"signature" : {
"hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
"keyId" : NumberLong(0)
}
}
}
```

步驟10.在lab-1-sessionmgr05上重新啟動之前為主要成員的mongo服務。

```
[root@lab-1-sessionmgr05 ~]# /etc/init.d/sessionmgr-27737 start
Starting sessionmgr-27737 (via systemctl): [ OK ]
```

步驟11.在步驟2中停止的set01e複製副本集的所有複製副本成員上啟動aido\_client。