

# CPS Mongo 복제본 세트에서 손상된 MongoDB를 처리하는 절차

## 목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[배경 정보](#)

[문제](#)

[손상된 DB 멤버 복구 절차](#)

## 소개

이 문서에서는 Cisco CPS(Policy Suite) 복제본 세트에서 손상된 DB(MongoData Base)를 처리하는 절차에 대해 설명합니다.

## 사전 요구 사항

### 요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- 리눅스
- CPS
- 몽고DB

**참고:** CPS CLI에 대한 권한 루트 액세스 권한이 있어야 합니다.

### 사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

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

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 네트워크가 작동 중인 경우 모든 명령의 잠재적인 영향을 이해해야 합니다.

## 배경 정보

MongoDB는 소스 사용 가능한 플랫폼 간 DB(문서 지향 데이터베이스) 프로그램입니다. NoSQL DB

프로그램으로 분류됩니다. MongoDB는 CPS에서 다양한 유형의 DB를 관리하기 위해 광범위하게 사용됩니다(SSPR(Subscriber Profile Repository), 균형 등).

## 문제

sessionmgr 내에서 aido\_client가 계속 활성화되어 있는 동안 잘못된 db 조각 모음을 수행하면 MongoDB가 손상되었습니다.

이렇게 하면 MongoDB가 메모리에 데이터를 보유하지만 DB 경로에 로컬로 쓸 수는 없습니다.

영향을 받는 복제본 세트에서 주 멤버(mongo 인스턴스)를 다시 시작하거나 sessionmgr VM을 다시 시작하면 데이터가 손실될 수 있습니다.

DB 멤버가 손상된 것처럼 보이는 방법을 이해하려면 문제가 있는 멤버 중 하나에 로그인하고 제공된 검사를 수행할 수 있습니다.

1단계. show dbs 명령을 실행하면 DB 목록의 출력이 반환되지 않습니다. 그러나 알고 있는 DB 내의 카운트를 확인하면 카운트가 반환됩니다.

```
[root@lab-1-pcrfclient01 ~]# mongo --host sessionmgr05:27737
MongoDB shell version v3.6.17
connect to: mongod://sessionmgr05:27737/?gssapiServiceName=mongod
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
```

2단계. **diagnostics.sh --get\_shard**를 실행하면 애플리케이션 샤드에는 데이터가 표시됩니다. 이는 실제로 Sessionmgr 가상 머신(VM)의 DBPATH가 아닌 인 메모리에 저장됩니다.

```
[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

3단계. 이 출력은 실제 데이터가 저장되어야 하는 DB PATH 내에 콘텐츠가 없음을 보여줍니다.

```
[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]
```

SSH(Secure Shell)를 연결된 sessionmgr에 연결하고 mongo 컨피그레이션에 언급된 DB\_PATH로 이동합니다. DB\_PATH 내의 콘텐츠가 비어 있는 것을 확인할 수 있습니다.

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

이러한 검사를 통해 MongoDB가 손상되었다는 결론을 내릴 수 있습니다.

## 손상된 DB 멤버 복구 절차

1단계. 문제가 있는 복제본 세트의 주 구성원에 대한 SSH입니다.

2단계. `aido_client`를 중지합니다(set01e에 속하는 복제 세트의 모든 구성원에서 `aido` 클라이언트를 중지하려면 확인).

3단계. set01e의 mongo 셸에 연결하고 다음 단계를 실행합니다.

```
# 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,"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "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
```

4단계. 동일한 복제본 인스턴스로 다시 연결하고 모든 session\_cache\_dbs에서 이 명령을 수행합니다. 여기서 session\_cache DB의 샘플을 설명합니다.

```
# 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: mongod://127.0.0.1:27737/?gssapiServiceName=mongod
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

```

**참고:** 나머지 session\_cache DB에 대해 4단계를 반복합니다.

5단계. 동일한 mongo 인스턴스를 다시 연결할 때 show dbs에 모든 DB가 나열되는지 확인합니다.

```

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

```

6단계. 이제 DB 경로에 sessionmgr 내부의 모든 데이터가 로컬로 포함되어 있는지 확인합니다. 복제본 세트의 각 데이터 경로를 확인할 수 있습니다. 이 경우 /var/data/sessions.1/set01e입니다.

```
[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
```

7단계. 동일한 사이트 보조 멤버에 대한 SSH를 수행하고 기본 멤버와 데이터 경로의 로컬 동기화를 수행합니다.

```
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

데이터 경로 `/var/data/sessions.1/set01e`가 비어 있는지 확인하고 비어 있지 않으면 `rm -rf /var/data/sessions.1/set01e/*`를 사용하여 제거한 다음 mongo 프로세스를 시작합니다.

**# /etc/init.d/sessionmgr-27737 시작**

```
[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 ]
```

8단계. 데이터가 이제 `/var/data/sessions.1/set01e`에 로컬로 복사되었는지 확인합니다.

```
[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]#

```

**참고:** 지역 사이트 보조 멤버에 대해 7단계 및 8단계를 반복합니다. 이 실습에서는 lab-2-sessionmgr05 및 lab-2-sessionmgr06의 멤버입니다.

9단계. 모든 보조 DB가 복구되면(로컬 및 지역 사이트) 기본 멤버에서 mongo 서비스를 다시 시작합니다.

```

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

```

10초 동안 기다린 후 기본 스위치가 성공했는지 확인합니다.

```

[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,"AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
"keyId" : NumberLong(0)
}
}
}
```

10단계. 이전의 주 멤버였던 lab-1-sessionmgr05에서 mongo 서비스를 다시 시작합니다.

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

11단계. 2단계에서 중지된 set01e 복제본 세트의 모든 복제본 멤버에 대해 aido\_client를 시작합니다.