

# 處理CPS副本集中會話管理器的角色和優先順序的過程

## 目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[背景資訊](#)

[問題](#)

[將sessionmgr從副本集中的主和更改sessionmgr優先順序移動的過程](#)

[方針1](#)

[方針2](#)

## 簡介

本文檔介紹將sessionmgr從主角色移動的過程，以及更改Cisco Policy Suite(CPS)副本集中的sessionmgr優先順序。

## 必要條件

### 需求

思科建議您瞭解以下主題：

- Linux
- CPS
- MongoDB

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

CPS使用MongoDB，其中mongod進程在Sessionmgr虛擬機器(VM)上運行，以構成其基本資料庫結構。它擁有多個副本集，用於各種用途，包括管理、使用者配置檔案儲存庫(SCR)、平衡、會話、報告和稽核。

MongoDB中的副本集是一組維護相同資料集的單進程。副本集提供冗餘和高可用性。由於資料在不同資料庫伺服器上有多個副本，因此它允許負載共用讀取操作。

副本集包含多個承載資料的節點和一個可選的仲裁器節點。在承載資料的節點中，一個且只有一個成員被視為主節點，而其他節點被視為輔助節點（副本集可以有多个輔助節點）。主節點處理所有寫入操作。

輔助節點複製主節點的操作日誌(oplog)，並將這些操作應用到其資料集，以便輔助節點的資料集反映主節點的資料集。如果主節點不可用，符合條件的輔助節點將進行選擇以自己選擇新的主節點。仲裁員參加選舉，但不儲存資料。

若要獲取副本集狀態，請從ClusterManager或pcrfclient運行diagnostics.sh —get\_r命令。

此處提供了一個副本集示例。set07。

```
| SET NAME - PORT : IP ADDRESS - REPLICHA STATE - HOST NAME - HEALTH - LAST SYNC -PRIORITY
|-----|
| SESSION:set07 |
| Status via arbitervip:27727 sessionmgr01:27727 sessionmgr02:27727 |
| Member-1 - 27727 : - SECONDARY - sessionmgr01 - ON-LINE - 0 sec - 2 |
| Member-2 - 27727 : 192.168.10.146 - ARBITER - arbitervip - ON-LINE - ----- - 0 |
| Member-3 - 27727 : - PRIMARY - sessionmgr02 - ON-LINE - ----- - 3 |
|-----|
```

要獲取副本集配置資訊，請執行以下步驟。

步驟1.登入該副本集的主要MongoDB成員。從ClusterManager運行此命令。

```
Command template:
#mongo --host <sessionmgrXX> --port <Replica Set port>
```

```
Sample command:
#mongo --host sessionmgr02 --port 27727
```

步驟2.運行命令以獲取副本集配置資訊。

```
set07:PRIMARY> rs.conf()
{
  "_id" : "set07",
  "version" : 2,
  "members" : [
    {
      "_id" : 0,
      "host" : "sessionmgr01:27727",
      "arbiterOnly" : false,
      "buildIndexes" : true,
      "hidden" : false,
      "priority" : 2,
      "tags" : {
```

```

},
"slaveDelay" : NumberLong(0),
"votes" : 1
},
{
  "_id" : 1,
  "host" : "arbitervip:27727",
  "arbiterOnly" : true,
  "buildIndexes" : true,
  "hidden" : false,
  "priority" : 0,
  "tags" : {
  },
  "slaveDelay" : NumberLong(0),
  "votes" : 1
},
{
  "_id" : 2,
  "host" : "sessionmgr02:27727",
  "arbiterOnly" : false,
  "buildIndexes" : true,
  "hidden" : false,
  "priority" : 3,
  "tags" : {
  },
  "slaveDelay" : NumberLong(0),
  "votes" : 1
}
],
"settings" : {
  "chainingAllowed" : true,
  "heartbeatIntervalMillis" : 2000,
  "heartbeatTimeoutSecs" : 1,
  "electionTimeoutMillis" : 10000,
  "catchUpTimeoutMillis" : -1,
  "catchUpTakeoverDelayMillis" : 30000,
  "getLastErrorModes" : {
  },
  "getLastErrorDefaults" : {
    "w" : 1,
    "wtimeout" : 0
  },
  "replicaSetId" : ObjectId("61cdb17a80b097a2e7604c97")
}
}
set07:PRIMARY>

```

**附註：**在副本集中具有最高優先順序的會話管理器作為主成員運行。

## 問題

假設sessionmgr在一個或多個副本集中執行主成員的角色，並且在這些情況下，您必須將副本集主角色移動到其他某個會話管理器，

1.無論何時執行任何涉及此sessionmgr VM關閉的活動，均可順利過渡。

2.如果sessionmgr運行狀況由於某種原因而降低，請使用其他正常的sessionmgr來維護副本集的正常功能。

## 將sessionmgr從副本集中的主和更改sessionmgr優先順序移動的過程

### 方針1

在此，副本集中的會話管理器優先順序直接在MongoDB級別更改。下面是將sessionmg02移出set07中主要角色的步驟。

選項1.更改sessionmgr02的優先順序。

步驟1.登入該副本集的主要MongoDB成員。

Command template:

```
#mongo --host <sessionmgrXX> --port <Replica Set port>
```

Sample command:

```
#mongo --host sessionmgr02 --port 27727
```

步驟2.運行命令以獲取複製副本集配置資訊。

```
set07:PRIMARY> rs.conf()
{
  "_id" : "set07",
  "version" : 2,
  "members" : [
    {
      "_id" : 0, -----> Position 0
      "host" : "sessionmgr01:27727",
      "arbiterOnly" : false,
      "buildIndexes" : true,
      "hidden" : false,
      "priority" : 2,
      "tags" : {

    },
    "slaveDelay" : NumberLong(0),
    "votes" : 1
  },
  {
    "_id" : 1, -----> Position 1
    "host" : "arbitervip:27727",
    "arbiterOnly" : true,
    "buildIndexes" : true,
    "hidden" : false,
    "priority" : 0,
    "tags" : {

  },
  "slaveDelay" : NumberLong(0),
  "votes" : 1
},
  {
    "_id" : 2, -----> Position 2
```

```

"host" : "sessionmgr02:27727",
"arbiterOnly" : false,
"buildIndexes" : true,
"hidden" : false,
"priority" : 3,
"tags" : {

},
"slaveDelay" : NumberLong(0),
"votes" : 1
}
],
"settings" : {
"chainingAllowed" : true,
"heartbeatIntervalMillis" : 2000,
"heartbeatTimeoutSecs" : 1,
"electionTimeoutMillis" : 10000,
"catchUpTimeoutMillis" : -1,
"catchUpTakeoverDelayMillis" : 30000,
"getLastErrorModes" : {

},
"getLastErrorDefaults" : {
"w" : 1,
"wtimeout" : 0
},
},
"replicaSetId" : ObjectId("61cdb17a80b097a2e7604c97")
}
}
set07:PRIMARY>

```

**附註：**請注意sessionmgr在rs.conf()輸出中的位置。

**步驟3.**運行此命令，將終端機移至配置模式。

```

set07:PRIMARY> cfg = rs.conf()
{
  "_id" : "set07",
  "version" : 2,
  "members" : [
    {
      "_id" : 0,
      "host" : "sessionmgr01:27727",
      "arbiterOnly" : false,
      "buildIndexes" : true,
      "hidden" : false,
      "priority" : 2,
      "tags" : {

    },
    "slaveDelay" : NumberLong(0),
    "votes" : 1
  },
  {
    "_id" : 1,
    "host" : "arbitervip:27727",
    "arbiterOnly" : true,
    "buildIndexes" : true,
    "hidden" : false,
    "priority" : 0,
    "tags" : {

```

```

},
"slaveDelay" : NumberLong(0),
"votes" : 1
},
{
  "_id" : 2,
  "host" : "sessionmgr02:27727",
  "arbiterOnly" : false,
  "buildIndexes" : true,
  "hidden" : false,
  "priority" : 3,
  "tags" : {

},
"slaveDelay" : NumberLong(0),
"votes" : 1
}
],
"settings" : {
  "chainingAllowed" : true,
  "heartbeatIntervalMillis" : 2000,
  "heartbeatTimeoutSecs" : 1,
  "electionTimeoutMillis" : 10000,
  "catchUpTimeoutMillis" : -1,
  "catchUpTakeoverDelayMillis" : 30000,
  "getLastErrorModes" : {

},
  "getLastErrorDefaults" : {
    "w" : 1,
    "wtimeout" : 0
  },
  "replicaSetId" : ObjectId("61cdb17a80b097a2e7604c97")
}
}
set07:PRIMARY>

```

**步驟4. 運行此命令以更改sessionmgr的優先順序。**

Command template:

```
cfg.members[X].priority = X --> put the position here in [].
```

sample command:

```
cfg.members[2].priority = 1
```

這裡，sessionmgr02當前是主要成員，其位置為2，優先順序為3。

為了將此sessionmgr02移出主角色，請提供大於0但小於具有最高優先順序的輔助成員優先順序的最低優先順序別，例如。1。

```
set07:PRIMARY> cfg.members[2].priority = 1
```

```
1
```

```
set07:PRIMARY>
```

**步驟5. 運行此命令以提交更改。**

```
set07:PRIMARY> rs.reconfig(cfg)
```

```
{
```

```

"ok" : 1,
"operationTime" : Timestamp(1641528658, 1),
"$clusterTime" : {
"clusterTime" : Timestamp(1641528658, 1),
"signature" : {
"hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
"keyId" : NumberLong(0)
}
}
}
2022-01-07T04:10:57.280+0000 I NETWORK [thread1] trying reconnect to sessionmgr02:27727
(192.168.10.140) failed
2022-01-07T04:10:57.281+0000 I NETWORK [thread1] reconnect sessionmgr02:27727 (192.168.10.140)
ok
set07:SECONDARY>

```

**步驟6.再次運行該命令以驗證sessionmgr優先順序中的更改。**

```

set07:SECONDARY> rs.conf()
{
  "_id" : "set07",
  "version" : 3,
  "members" : [
    {
      "_id" : 0,
      "host" : "sessionmgr01:27727",
      "arbiterOnly" : false,
      "buildIndexes" : true,
      "hidden" : false,
      "priority" : 2,
      "tags" : {

    },
    "slaveDelay" : NumberLong(0),
    "votes" : 1
  },
  {
    "_id" : 1,
    "host" : "arbitervip:27727",
    "arbiterOnly" : true,
    "buildIndexes" : true,
    "hidden" : false,
    "priority" : 0,
    "tags" : {

  },
  "slaveDelay" : NumberLong(0),
  "votes" : 1
},
{
  "_id" : 2,
  "host" : "sessionmgr02:27727",
  "arbiterOnly" : false,
  "buildIndexes" : true,
  "hidden" : false,
  "priority" : 1, --> Here priority has been changed from 3 to 1.
  "tags" : {

},
"slaveDelay" : NumberLong(0),
"votes" : 1
}
}

```

```

],
"settings" : {
"chainingAllowed" : true,
"heartbeatIntervalMillis" : 2000,
"heartbeatTimeoutSecs" : 1,
"electionTimeoutMillis" : 10000,
"catchUpTimeoutMillis" : -1,
"catchUpTakeoverDelayMillis" : 30000,
"getLastErrorModes" : {
},
"getLastErrorDefaults" : {
"w" : 1,
"wtimeout" : 0
},
"replicaSetId" : ObjectId("61cdb17a80b097a2e7604c97")
}
}
set07:SECONDARY>

```

步驟7.從ClusterManager或pcrfclient運行diagnostics.sh —get\_r命令以驗證複製副本集狀態的更改

```

o

| SET NAME - PORT : IP ADDRESS - REPLICA STATE - HOST NAME - HEALTH - LAST SYNC - PRIORITY |
-----
| SESSION:set07 |
| Status via arbitervip:27727 sessionmgr01:27727 sessionmgr02:27727 |
| Member-1 - 27727 : - PRIMARY - sessionmgr01 - ON-LINE - ----- - 2 |
| Member-2 - 27727 : 192.168.10.146 - ARBITER - arbitervip - ON-LINE - ----- - 0 |
| Member-3 - 27727 : - SECONDARY - sessionmgr02 - ON-LINE - 0 sec - 1 |
|-----|

```

現在，您會看到sessionmgr02已移動到輔助節點。為了使sessionmgr02再次成為主成員，請在步驟4中使用此命令運行上述步驟1到5。

cfg.members[2].priority =任何大於2但小於1001的數字 —>將優先順序置於高於示例中當前主要成員2的優先順序。

```

set07:PRIMARY> cfg.members[2].priority = 5
5
set07:PRIMARY> rs.reconfig(cfg)
{
"ok" : 1,
"operationTime" : Timestamp(1641531450, 1),
"$clusterTime" : {
"clusterTime" : Timestamp(1641531450, 1),
"signature" : {
"hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
"keyId" : NumberLong(0)
}
}
}
2022-01-07T04:57:31.247+0000 I NETWORK [thread1] trying reconnect to sessionmgr01:27727
(192.168.10.139) failed
2022-01-07T04:57:31.247+0000 I NETWORK [thread1] reconnect sessionmgr01:27727 (192.168.10.139)
ok
set07:SECONDARY>

```

運行命令以驗證sessionmgr優先順序中的更改。



```

set07:SECONDARY> rs.conf()
{
  "_id" : "set07",
  "version" : 4,
  "members" : [
    {
      "_id" : 0,
      "host" : "sessionmgr01:27727",
      "arbiterOnly" : false,
      "buildIndexes" : true,
      "hidden" : false,
      "priority" : 2,
      "tags" : {

    },
    "slaveDelay" : NumberLong(0),
    "votes" : 1
  },
  {
    "_id" : 1,
    "host" : "arbitervip:27727",
    "arbiterOnly" : true,
    "buildIndexes" : true,
    "hidden" : false,
    "priority" : 0,
    "tags" : {

  },
  "slaveDelay" : NumberLong(0),
  "votes" : 1
},
{
  "_id" : 2,
  "host" : "sessionmgr02:27727",
  "arbiterOnly" : false,
  "buildIndexes" : true,
  "hidden" : false,
  "priority" : 5, --> Here priority has been changed from 1 to 5.
  "tags" : {

},
"slaveDelay" : NumberLong(0),
"votes" : 1
},
],
"settings" : {
  "chainingAllowed" : true,
  "heartbeatIntervalMillis" : 2000,
  "heartbeatTimeoutSecs" : 1,
  "electionTimeoutMillis" : 10000,
  "catchUpTimeoutMillis" : -1,
  "catchUpTakeoverDelayMillis" : 30000,
  "getLastErrorModes" : {

},
  "getLastErrorDefaults" : {
    "w" : 1,
    "wtimeout" : 0
  },
  "replicaSetId" : ObjectId("61cdb17a80b097a2e7604c97")
}

```

```
set07:SECONDARY>
```

從ClusterManager或pcrfclient運行diagnostics.sh —get\_r命令以驗證複製副本集狀態的更改。

```
| SET NAME - PORT : IP ADDRESS - REPLICA STATE - HOST NAME - HEALTH - LAST SYNC -PRIORITY
|-----|
| SESSION:set07 |
| Status via arbitervip:27727 sessionmgr01:27727 sessionmgr02:27727 |
| Member-1 - 27727 : - SECONDARY - sessionmgr01 - ON-LINE - 14 sec - 2 |
| Member-2 - 27727 : 192.168.10.146 - ARBITER - arbitervip - ON-LINE - ----- - 0 |
| Member-3 - 27727 : - PRIMARY - sessionmgr02 - ON-LINE - ----- - 5 |
|-----|
|-----|
```

現在，您可以看到sessionmgr02再次成為主會話。

選項2.更改其他輔助會話管理器的優先順序，使其成為主要成員。這是sessionmgr01。

為了使sessionmgr01成為主成員，請在步驟4中使用此命令運行選項1中的上述步驟1到5。

cfg.members[0].priority =大於3但小於1001的任何數字 —>將優先順序置於高於示例中當前主要成員「3」的優先順序。

```
set07:PRIMARY> cfg.members[0].priority = 4
4
set07:PRIMARY> rs.reconfig(cfg)
{
"ok" : 1,
"operationTime" : Timestamp(1641540587, 1),
"$clusterTime" : {
"clusterTime" : Timestamp(1641540587, 1),
"signature" : {
"hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
"keyId" : NumberLong(0)
}
}
}
2022-01-07T07:29:46.141+0000 I NETWORK [thread1] trying reconnect to sessionmgr02:27727
(192.168.10.140) failed
2022-01-07T07:29:46.142+0000 I NETWORK [thread1] reconnect sessionmgr02:27727 (192.168.10.140)
ok
set07:SECONDARY>
```

運行命令以確認更改。

```
set07:SECONDARY> rs.conf()
{
"_id" : "set07",
"version" : 4,
"members" : [
{
"_id" : 0,
"host" : "sessionmgr01:27727",
"arbiterOnly" : false,
"buildIndexes" : true,
"hidden" : false,
"priority" : 4, --> Here priority has been changed from 2 to 4.
"tags" : {
```

```

},
"slaveDelay" : NumberLong(0),
"votes" : 1
},
{
"_id" : 1,
"host" : "arbitervip:27727",
"arbiterOnly" : true,
"buildIndexes" : true,
"hidden" : false,
"priority" : 0,
"tags" : {
},
"slaveDelay" : NumberLong(0),
"votes" : 1
},
{
"_id" : 2,
"host" : "sessionmgr02:27727",
"arbiterOnly" : false,
"buildIndexes" : true,
"hidden" : false,
"priority" : 3,
"tags" : {
},
"slaveDelay" : NumberLong(0),
"votes" : 1
}
],
"settings" : {
"chainingAllowed" : true,
"heartbeatIntervalMillis" : 2000,
"heartbeatTimeoutSecs" : 1,
"electionTimeoutMillis" : 10000,
"catchUpTimeoutMillis" : -1,
"catchUpTakeoverDelayMillis" : 30000,
"getLastErrorModes" : {
},
"getLastErrorDefaults" : {
"w" : 1,
"wtimeout" : 0
},
"replicaSetId" : ObjectId("61cdb17a80b097a2e7604c97")
}
}
set07:SECONDARY>

```

從群集管理器或pcrfcleint運行diagnostics.sh —get\_r命令以驗證複製副本集狀態的更改。

```

| SET NAME - PORT : IP ADDRESS - REPLICA STATE - HOST NAME - HEALTH - LAST SYNC -PRIORITY
|-----|
| SESSION:set07 |
| Status via arbitervip:27727 sessionmgr01:27727 sessionmgr02:27727 |
| Member-1 - 27727 : - PRIMARY - sessionmgr01 - ON-LINE - ----- - 4 |
| Member-2 - 27727 : 192.168.10.146 - ARBITER - arbitervip - ON-LINE - ----- - 0 |
| Member-3 - 27727 : - SECONDARY - sessionmgr02 - ON-LINE - 0 sec - 3 |
|-----|

```

現在，您可以看到sessionmgr01已變為主映像，而sessionmgr02變為輔助映像。

若要使sessionmgr02再次成為主成員，請使用步驟4中的此命令在**選項1**中運行上述步驟1到5。

cfg.members[0].priority = 小於3但大於0的任何數字 —>將優先順序置於低於示例中為「3」的sessionmgr02的優先順序。

```
set07:PRIMARY> cfg.members[0].priority = 1
1
set07:PRIMARY> rs.reconfig(cfg)
{
"ok" : 1,
"operationTime" : Timestamp(1641531450, 1),
"$clusterTime" : {
"clusterTime" : Timestamp(1641531450, 1),
"signature" : {
"hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
"keyId" : NumberLong(0)
}
}
}
2022-01-07T08:34:31.165+0000 I NETWORK [thread1] trying reconnect to sessionmgr01:27727
(192.168.10.139) failed
2022-01-07T08:34:31.165+0000 I NETWORK [thread1] reconnect sessionmgr01:27727 (192.168.10.139)
ok
set07:SECONDARY>
```

運行此命令以驗證sessionmgr優先順序中的更改。

```
set07:SECONDARY> rs.conf()
{
"_id" : "set07",
"version" : 4,
"members" : [
{
"_id" : 0,
"host" : "sessionmgr01:27727",
"arbiterOnly" : false,
"buildIndexes" : true,
"hidden" : false,
"priority" : 1, --> Here priority has been changed from 4 to 1.
"tags" : {
},
"slaveDelay" : NumberLong(0),
"votes" : 1
},
{
"_id" : 1,
"host" : "arbitervip:27727",
"arbiterOnly" : true,
"buildIndexes" : true,
"hidden" : false,
"priority" : 0,
"tags" : {
},
"slaveDelay" : NumberLong(0),
"votes" : 1
},
{

```

```

"_id" : 2,
"host" : "sessionmgr02:27727",
"arbiterOnly" : false,
"buildIndexes" : true,
"hidden" : false,
"priority" : 3,
"tags" : {
},
"slaveDelay" : NumberLong(0),
"votes" : 1
},
"settings" : {
"chainingAllowed" : true,
"heartbeatIntervalMillis" : 2000,
"heartbeatTimeoutSecs" : 1,
"electionTimeoutMillis" : 10000,
"catchUpTimeoutMillis" : -1,
"catchUpTakeoverDelayMillis" : 30000,
"getLastErrorModes" : {
},
"getLastErrorDefaults" : {
"w" : 1,
"wtimeout" : 0
},
"replicaSetId" : ObjectId("61cdb17a80b097a2e7604c97")
}
set07:SECONDARY>

```

從ClusterManager或pcrfclient運行diagnostics.sh —get\_r命令以驗證複製副本集狀態的更改。

```

| SET NAME - PORT : IP ADDRESS - REPLICA STATE - HOST NAME - HEALTH - LAST SYNC -PRIORITY
|-----|
|
| SESSION:set07 |
| Status via arbitervip:27727 sessionmgr01:27727 sessionmgr02:27727 |
| Member-1 - 27727 : - SECONDARY - sessionmgr01 - ON-LINE - 14 sec - 1 |
| Member-2 - 27727 : 192.168.10.146 - ARBITER - arbitervip - ON-LINE - ----- - 0 |
| Member-3 - 27727 : - PRIMARY - sessionmgr02 - ON-LINE - ----- - 3 |
|-----|
|

```

現在，您可以看到sessionmgr02已變成主會話，而sessionmgr01則是輔助會話。

## 方針2

可以使用ClusterManager中的CPS指令碼set\_priority.sh來更改複製副本集中的會話管理器優先順序。預設情況下，成員的優先順序按順序設定（優先順序較高），如/etc/broadhop/mongoConfig.cfg中的ClusterManager中所定義。

以set07為例。

```

[root@installer broadhop]# cat mongoConfig.cfg
[SESSION-SET2]
SETNAME=set07
OPLOG_SIZE=5120

```

```
ARBITER=arbitervip:27727
ARBITER_DATA_PATH=/var/data/sessions.7
MEMBER1=sessionmgr02:27727
MEMBER2=sessionmgr01:27727
DATA_PATH=/var/data/sessions.1/2
[SESSION-SET2-END]
```

要獲取副本集狀態，請從ClusterManager或pcrfclient運行diagnostics.sh —get\_r命令。

```
| SET NAME - PORT : IP ADDRESS - REPLICA STATE - HOST NAME - HEALTH - LAST SYNC - PRIORITY |
|-----|
| SESSION:set07 |
| Status via arbitervip:27727 sessionmgr01:27727 sessionmgr02:27727 |
| Member-1 - 27727 : - SECONDARY - sessionmgr01 - ON-LINE - 0 sec - 2 |
| Member-2 - 27727 : 192.168.10.146 - ARBITER - arbitervip - ON-LINE - ----- - 0 |
| Member-3 - 27727 : - PRIMARY - sessionmgr02 - ON-LINE - ----- - 3 |
|-----|
```

比較上述結果時，可以看到sessionmgr02是/etc/broadhop/mongoConfig.cfg中set07的第一成員[MEMBER1]，因此sessionmgr02在預設情況下是set07中的主要成員。

此處提供了CPS高可用性選項，它們使用set\_priority.sh指令碼將sessionmgr02移出set07中的主成員角色。

步驟1.按升序設定優先順序。

Command template:

```
sh set_priority.sh --db arg --replSet arg --asc
```

where ,

--db arg --> arg is database name

[all|session|spr|admin|balance|report|portal|audit|bindings|session\_configs|bindings\_configs|spr\_configs]

--replSet arg -->arg is <setname>

Sample command:

```
sh set_priority.sh --db session --replSet set07 --asc
```

```
[root@installer ~]# sh set_priority.sh --db session --replSet set07 --asc
```

Set priorities is in progress. check log /var/log/broadhop/scripts/set\_priority.log to know the status

```
Setting priority for Replica-Set: SESSION-SET2
```

```
INFO Parsing Mongo Config file
```

```
INFO Priority set operation is completed for SESSION-SET2
```

```
INFO Priority set to the Database members is finished
```

```
INFO Validating if Priority is set correctly for Replica-Set: SESSION-SET2
```

```
WARNING Mongo Server trying to reconnect while getting config. Attempt #1
```

```
INFO Validated Priority is set correctly for Replica-Set: SESSION-SET2
```

```
Primary member sessionmgr01:27727 found for Replica SESSION-SET2
```

Set priorities process successfully completed.

```
[root@installer ~]#
```

步驟2.從ClusterManager或pcrfclient運行diagnostics.sh —get\_r命令以驗證更改。

```
| SET NAME - PORT : IP ADDRESS - REPLICAS STATE - HOST NAME - HEALTH - LAST SYNC - PRIORITY |
|-----|
| SESSION:set07 |
| Status via arbitervip:27727 sessionmgr01:27727 sessionmgr02:27727 |
| Member-1 - 27727 : - PRIMARY - sessionmgr01 - ON-LINE - ----- - 3 |
| Member-2 - 27727 : 192.168.10.146 - ARBITER - arbitervip - ON-LINE - ----- - 0 |
| Member-3 - 27727 : - SECONDARY - sessionmgr02 - ON-LINE - 0 sec - 2 |
|-----|
|-----|
```

現在，sessionmgr01已成為主成員，因為優先順序已按/etc/broadhop/mongoConfig.cfg中定義的升序設定。

若要使sessionmgr02再次成為主成員，請運行此命令。

```
[root@installer ~]# sh set_priority.sh --db session --replSet set07
```

```
Set priorities is in progress. check log /var/log/broadhop/scripts/set_priority.log to know the status
```

```
Setting priority for Replica-Set: SESSION-SET2
```

```
INFO Parsing Mongo Config file
```

```
INFO Priority set operation is completed for SESSION-SET2
```

```
INFO Priority set to the Database members is finished
```

```
INFO Validating if Priority is set correctly for Replica-Set: SESSION-SET2
```

```
WARNING Mongo Server trying to reconnect while getting config. Attempt #1
```

```
INFO Validated Priority is set correctly for Replica-Set: SESSION-SET2
```

```
Primary member sessionmgr02:27727 found for Replica SESSION-SET2
```

```
Set priorities process successfully completed.
```

```
[root@installer ~]#
```

**附註：**預設情況下，優先順序按降序設定。

從ClusterManager或pcrfclient運行diagnostics.sh —get\_r命令以驗證更改。

```
| SET NAME - PORT : IP ADDRESS - REPLICAS STATE - HOST NAME - HEALTH - LAST SYNC - PRIORITY |
|-----|
| SESSION:set07 |
| Status via arbitervip:27727 sessionmgr01:27727 sessionmgr02:27727 |
| Member-1 - 27727 : - SECONDARY - sessionmgr01 - ON-LINE - 0 sec - 2 |
| Member-2 - 27727 : 192.168.10.146 - ARBITER - arbitervip - ON-LINE - ----- - 0 |
| Member-3 - 27727 : - PRIMARY - sessionmgr02 - ON-LINE - ----- - 3 |
|-----|
|-----|
```

現在，您可以看到sessionmgr02已變成主會話，而sessionmgr01則是輔助會話。