

在CPS副本集中管理仲裁器節點的過程

目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[背景資訊](#)

[問題](#)

[管理副本集中的仲裁器的過程](#)

簡介

本文檔介紹在思科策略套件(CPS)副本集中管理仲裁器節點的過程。

必要條件

需求

思科建議您瞭解以下主題：

- Linux
- CPS
- MongoDB

注意： 思科建議您必須具有對CPS CLI的超級使用者訪問許可權。

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
- 整合運算系統(UCS)-B
- MongoDB v3.6.17和v3.4.16

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構建其基本資料庫(DB)結構。它擁有多個副本集，用於各種用途：ADMIN、Subscriber Profile Repository(SPR)、BALANCE、SESSION、REPORTING、AUDIT。

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(HA)。由於資料在不同資料庫伺服器上有多個副本，因此允許負載共用讀取操作。

在某些情況下（例如您有一個主例項和一個輔助例項，但是成本限制禁止新增另一個輔助例項），您可以選擇將單例項新增到副本集中，作為仲裁人在選舉中投票。仲裁者只有一票。預設情況下，仲裁人的優先順序為0。

仲裁程式是屬於副本集但不儲存資料的單例項例項（這意味著它們不提供資料冗餘）。不過，他們可以參加選舉。仲裁員參與主節點的選舉，但仲裁員沒有資料集的副本，因此無法成為主節點。

仲裁程式具有最低的資源要求，不需要專用硬體。您可以在應用伺服器或僅監控網路的主機上部署仲裁器。

仲裁者不儲存資料，但在將仲裁者單目標進程新增到副本集之前，仲裁者的行為與任何其他單目標進程類似，並使用一組資料檔案和完整大小的日誌啟動。

以下是複製副本集示例， `set07`。

```
| 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 |
|-----|
```

問題

假設仲裁器有問題，或者需要更改副本集中的仲裁器，則必須刪除當前仲裁器並向副本集中新增新仲裁器。

管理副本集中的仲裁器的過程

步驟1.驗證CPS和新仲裁器中的mongo shell版本。從副本集和新仲裁器節點中的主sessionmgr運行此命令。

sessionmgr的輸出示例：

```
[root@sessionmgr02 ~]# mongo --version
MongoDB shell version v3.6.17
```

如果主sessionmgr和新仲裁程式中的mongo shell版本相同，或者新仲裁程式的mongo shell版本更高，則導航到步驟6。

否則，如果新的仲裁器mongo shell版本較低，則必須設定 `featureCompatibilityVersion` 作為副本集管理資料庫中的較低值，並執行以下步驟。

新仲裁器mongo shell版本低於CPS sessionmgr的示例案例：

```
[root@pcrfclient02 ~]# mongo --version
MongoDB shell version v3.4.16
```

步驟2.登入到複製副本集的主合併例項。

Command template:

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

Sample command:

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

步驟3. 運行此命令可檢視當前 featureCompatibilityVersion 在副本集的admin資料庫中。

```
set07:PRIMARY> db.adminCommand( { getParameter: 1, featureCompatibilityVersion: 1 } )
{
  "featureCompatibilityVersion" : {
    "version" : "3.6"
  },
  "ok" : 1,
  "operationTime" : Timestamp(1663914140, 1),
  "$clusterTime" : {
    "clusterTime" : Timestamp(1663914140, 1),
    "signature" : {
      "hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
```

步驟4. 運行此命令 setfeatureCompatibilityVersion 在副本集的admin資料庫中為3.4。

```
set07:PRIMARY> db.adminCommand( { setFeatureCompatibilityVersion: "3.4" } )
{ "ok" : 1 }
```

步驟5. 運行此命令以驗證 featureCompatibilityVersion 在副本集的admin資料庫中已更改為3.4。

```
set07:PRIMARY> db.adminCommand( { getParameter: 1, featureCompatibilityVersion: 1 } )
{ "featureCompatibilityVersion" : { "version" : "3.4" }, "ok" : 1 }
```

步驟6. 登入到群集管理器並修改 /var/qps/config/deploy/csv/AdditionalHosts.csv 包含新仲裁器詳細資訊的檔案。

```
#vi /var/qps/config/deploy/csv/AdditionalHosts.csv
```

Provide new arbiter details in this format:

Host Alias IP Address

```
new-arbiter new-arbiter xx.xx.xx.xx
```

步驟7. 匯入CSV配置。

```
#/var/qps/install/current/scripts/import/import_deploy.sh
```

步驟8. 驗證 /etc/hosts 已經更新了新仲裁人的資訊。

```
#cat /etc/hosts | grep arbiter
```

步驟9. 運行此命令進行同步 /etc/hosts.

```
#!/var/qps/bin/update/synchosts.sh
```

```
Syncing to following QNS Servers:
```

```
lb01 lb02 sessionmgr01 sessionmgr02 qns01 qns02 pcrfclient01 pcrfclient02
```

```
Do you want to Proceed? (y/n):y
```

```
lb01
```

```
lb02
```

```
sessionmgr01
```

```
sessionmgr02
```

```
qns01
```

```
qns02
```

```
pcrfclient01
```

```
pcrfclient02
```

步驟10.驗證pcrfclient VM上是否停止了mon_db指令碼。

```
#monsum | grep mon_db_for
```

如果停止，則輸出如下：

```
mon_db_for_lb_failover Not monitored Program
```

```
mon_db_for_callmodel Not monitored Program
```

如果未停止，則輸出如下：

```
mon_db_for_lb_failover OK Program
```

```
mon_db_for_callmodel OK Program
```

注意：如果mon_db指令碼未停止，請在各自的pcrfclient虛擬機器上運行這些命令以手動停止

。

```
#monit stop mon_db_for_lb_failover
```

```
#monit stop mon_db_for_callmodel
```

步驟11.從pcrfclient01運行此命令可從副本集中刪除當前仲裁器（此步驟中的示例為set07）。

```
#build_set.sh --session --remove-members --setname set07
```

```
Please enter the member details which you going to remove from the replica-set
```

```
Member:Port -----> arbitervip:27727
```

```
arbitervip:27727
```

```
Do you really want to remove [yes(y)/no(n)]: y
```

步驟12.從群集管理器運行此命令，以驗證是否已將仲裁器從 set07，輸出 set07 不能包含當前仲裁器

。

```
#diagnostics.sh --get_replica_status
```

```
Expected output:
```

```
-----|
```

```
|-----|
```

```
| SESSION:set07 |
```

```
| Status via sessionmgr01:27727 sessionmgr02:27727 |
```

```
| Member-1 - 27727 : - SECONDARY - sessionmgr01 - ON-LINE - 0 sec -|
```

```
| Member-2 - 27727 : - PRIMARY - sessionmgr02 - ON-LINE - ----- -|
```

```
|-----|
```

```
-----|
```

步驟13.更新 `mongoConfig.cfg` 在已修改的副本集中具有正確仲裁器的檔案。用新的仲裁器 (ARBITER=new-arbiter) 替換當前仲裁器 (ARBITER=arbiter)。從群集管理器運行此命令。

```
#vi /etc/broadhop/mongoConfig.cfg
```

當前配置：

```
[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]
```

所需配置：

```
[SESSION-SET2]
SETNAME=set07
OPLOG_SIZE=5120
ARBITER=new-arbiter:27727
ARBITER_DATA_PATH=/var/data/sessions.7
MEMBER1=sessionmgr02:27727
MEMBER2=sessionmgr01:27727
DATA_PATH=/var/data/sessions.1/2
[SESSION-SET2-END]
```

步驟14.複製更新的 `mongoConfig.cfg` 檔案到所有VM。從群集管理器運行此命令。

```
#copytoall.sh /etc/broadhop/mongoConfig.cfg /etc/broadhop/mongoConfig.cfg
```

步驟15.向set07新增新的仲裁器成員。從群集管理器，運行 `/var/qps/install/current/scripts/build/build_etc.sh` 命令以生成 `/etc/directory`。

步驟16.驗證在運行Oracle Replica Server之後，是否已將新仲裁器成員新增到副本集 `build_etc.sh` 現在，您必須等待AIDO伺服器使用新仲裁器建立/更新副本集。

```
#diagnostics.sh --get_replica_status
```

Expected Output:

```
| 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 : xx.xx.xx.xx - ARBITER - new-arbiter - ON-LINE - ----- - 0 |
| Member-3 - 27727 : - PRIMARY - sessionmgr02 - ON-LINE - ----- - 3 |
|-----|
```

注意：如果未新增新的仲裁器成員，請繼續執行後續步驟。否則，請導航至Step 18。

步驟17.從群集管理器運行此命令，以強制新增新的仲裁器成員。

```
#build_set.sh --DB_NAME --add-members --setname Setxxx --force
```

步驟18.如果仲裁器埠尚未啟動，請從新的仲裁器節點運行此命令以啟動該埠。

Command syntax:

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

Sample command:

```
#/etc/init.d/sessionmgr-27727 start
```

步驟19.驗證新仲裁程式是否新增成功。

```
#diagnostics.sh --get_replica_status
```

步驟20.從群集管理器運行此命令可相應地更新資料庫優先順序。

```
# cd /var/qps/bin/support/mongo/  
# ./set_priority.sh --db session  
# ./set_priority.sh --db spr  
# ./set_priority.sh --db admin  
# ./set_priority.sh --db balance  
# ./set_priority.sh --db audit  
# ./set_priority.sh --db report
```

步驟21.從群集管理器運行此命令以驗證複製副本集中的更改。

```
#diagnostics.sh --get_replica_status
```

Expected Output:

```
| 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 : xx.xx.xx.xx - ARBITER - new-arbiter - ON-LINE - ----- - 0 |  
| Member-3 - 27727 : - PRIMARY - sessionmgr02 - ON-LINE - ----- - 3 |  
|-----|
```

步驟22.驗證pcrfclient VM上是否還原了mon_db指令碼。否則，您必須手動啟動它們。

```
#monsum | grep mon_db_for
```

要啟用mon_db指令碼，請登入到所有pcrfclient虛擬機器並運行以下命令：

```
# monit start mon_db_for_lb_failover  
# monit start mon_db_for_callmodel
```

關於此翻譯

思科已使用電腦和人工技術翻譯本文件，讓全世界的使用者能夠以自己的語言理解支援內容。請注意，即使是最佳機器翻譯，也不如專業譯者翻譯的內容準確。Cisco Systems, Inc. 對這些翻譯的準確度概不負責，並建議一律查看原始英文文件（提供連結）。