

在CPS中設定會話副本的情況下，使用者會話處理

目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[背景資訊](#)

[問題](#)

[解決方案](#)

簡介

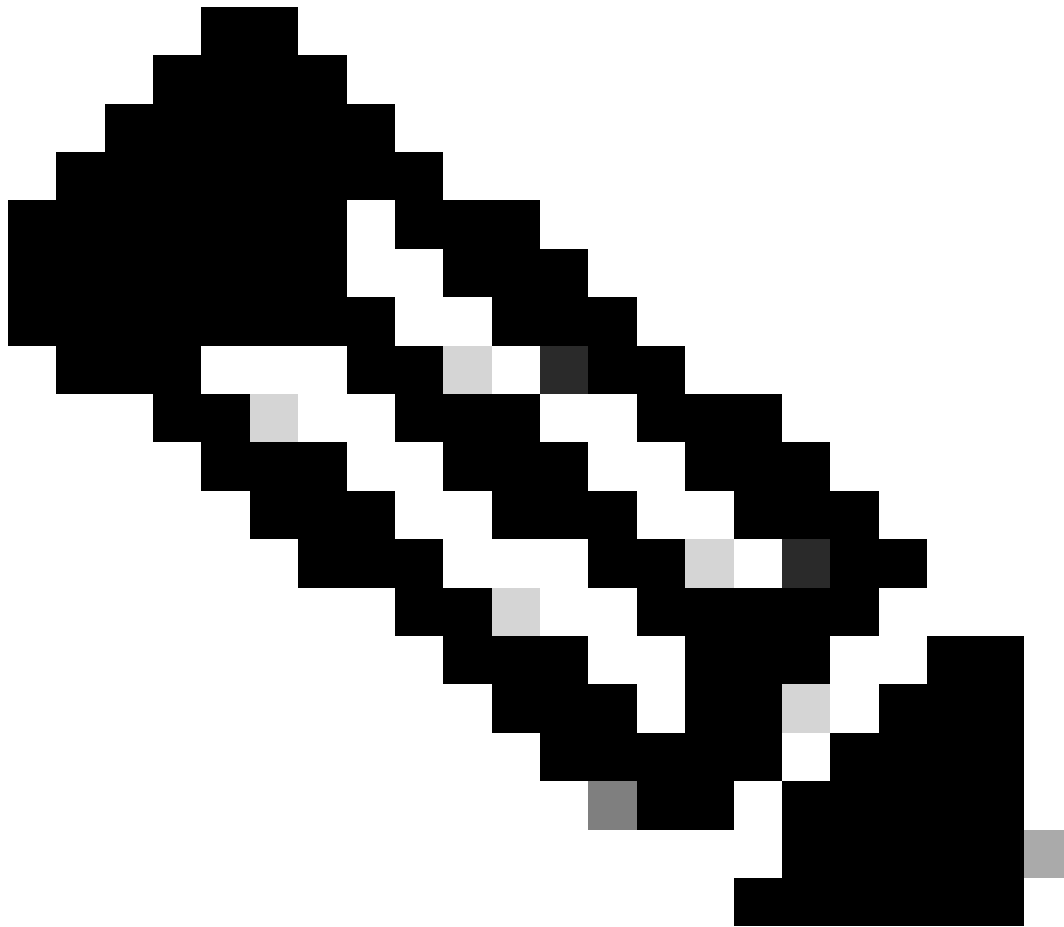
本檔案介紹在思科原則套件(CPS)中設定作業階段副本的情況下，使用者作業階段處理。

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

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，以便構建其基本資料庫結構。

為使副本集具有高可用性，建議的最小配置是一個包含三個資料承載成員的三成員副本集：一個主要成員和兩個輔助成員。在某些情況下（例如您有一個主要仲裁人和一個輔助仲裁人，但是成本限制禁止新增另一個輔助仲裁人），您可以選擇包括一個仲裁人。仲裁員參與選舉，但不儲存資料（即不提供資料冗餘）。對於CPS，通常以這種方式配置會話DB。

您可以在/etc/broadhop/mongoConfig.cfg中驗證CPS設定的副本集配置。

```
[SESSION-SET1]
SETNAME=set01a
OPLOG_SIZE=5120
ARBITER1=arbitervip:27717
ARBITER_DATA_PATH=/var/data/sessions.27717
MEMBER1=sessionmgr01:27717
MEMBER2=sessionmgr02:27717
DATA_PATH=/var/data/sessions.1/d
SHARD_COUNT=4
SEEDS=sessionmgr01:sessionmgr02:27717
[SESSION-SET1-END]
```

```
[SESSION-SET7]
SETNAME=set01g
OPLOG_SIZE=5120
ARBITER1=arbitervip:37717
ARBITER_DATA_PATH=/var/data/sessions.37717
MEMBER1=sessionmgr02:37717
MEMBER2=sessionmgr01:37717
DATA_PATH=/var/data/sessions.1/g
SHARD_COUNT=2
SEEDS=sessionmgr02:sessionmgr01:37717
[SESSION-SET7-END]
```

MongoDB還有另一個名為Sharding的概念，可幫助群集實現冗餘和速度。將資料庫拆分為索引集，這樣可以大大加快寫入速度，從而提高資料庫的整體效能。分片資料庫通常設定為每個分片都是一個副本集。

- 會話分片：會話共用種子及其資料庫：

```
osgi> listshards
```

```
Shard Id Mongo DB State Backup DB Removed Session Count
```

```
1 sessionmgr01:27717/session_cache online false false 109306
2 sessionmgr01:27717/session_cache_2 online false false 109730
3 sessionmgr01:27717/session_cache_3 online false false 109674
4 sessionmgr01:27717/session_cache_4 online false false 108957
```

- 輔助金鑰分片：輔助金鑰碎片種子及其資料庫：

```
osgi> listskshards
```

```
Shard Id Mongo DB State Backup DB Removed Session Count
```

```
2 sessionmgr02:37717/sk_cache online false false 150306
3 sessionmgr02:37717/sk_cache_2 online false false 149605
```

問題

問題1.由於單個成員故障，資料成員記憶體消耗穩步增長。

當一個資料承載成員在3個成員（2個資料承載+1仲裁器）中發生故障時，唯一剩餘的資料承載成員承擔主節點和副本集繼續運行，但負載和副本集重，沒有任何資料庫冗餘。使用三成員PSA架構時，如果任何資料承載節點發生故障，快取記憶體壓力會增加。這會導致剩餘的資料承載節點（主）的記憶體消耗量穩步增加，如果無人管理，則可能導致節點因可用記憶體耗盡而失敗，最終導致副本集故障。

```
-----
|[SESSION:set01a]|
|[Status via sessionmgr02:27717 ]|
|[Member-1 - 27717 : 192.168.29.100 - ARBITER - arbitervip - ON-LINE
|[Member-2 - 27717 : 192.168.29.35 - UNKNOWN - sessionmgr01 - OFF-LINE 19765 days
|[Member-3 - 27717 : 192.168.29.36 - PRIMARY - sessionmgr02 - ON-LINE
-----
```

```
-----
|[SESSION:set01g]|
|[Status via sessionmgr02:37717 ]|
|[Member-1 - 37717 : 192.168.29.100 - ARBITER - arbitervip - ON-LINE
|[Member-2 - 37717 : 192.168.29.35 - UNKNOWN - sessionmgr01 - OFF-LINE 19765 days
|[Member-3 - 37717 : 192.168.29.36 - PRIMARY - sessionmgr02 - ON-LINE
-----
```

問題2.雙成員故障對會話處理的影響。

當兩個資料承載成員（Sessionmgr01和sessionmgr02）在此類副本集中（雙故障）關閉時，整個副本集關閉，其基本資料庫功能受損。

```
Current setup have problem while connecting to the server on port : 27717
```

```
Current setup have problem while connecting to the server on port : 37717
```

在會話副本集的情況下，此副本集失敗會導致呼叫失敗，因為CPS呼叫處理進程(Quantum Network Suite(qns)進程)無法訪問已儲存在那些失敗的副本集中的會話。

解決方案

方法1.針對單成員故障。

您必須在短時間內在主輔助仲裁器(PSA)體系結構中恢復失敗的副本整合員。如果在PSA體系結構中恢復故障的資料承載成員需要時間，則必須刪除發生故障的。

步驟1.1.使用三成員PSA體系結構識別特定副本集中的故障資料承載成員。從群集管理器運行此命令。

```
#diagnostics.sh --get_r
```

步驟1.2.從特定副本集中刪除失敗的資料承載成員。

Syntax:

```
Usage: build_set.sh <--option1> <--option2> [--setname SETNAME] [--help]
```

option1: Database name

option2: Build operations (create, add or remove members)

Example:

```
#build_set.sh --session --remove-failed-members --setname set01a --force  
#build_set.sh --session --remove-failed-members --setname set01g --force
```

步驟1.3.檢驗故障成員是否已從副本集中刪除。

```
#diagnostics.sh --get_r
```

方法2.雙成員故障。

當兩個資料承載成員在3 PSA副本集中發生故障時，這不是一種永久的解決方法。相反，這是一種臨時的解決方法，通過從各自的副本集中移除失敗的成員、會話共用和共用sk，可以避免或減少呼叫失敗並確保無感覺的流量處理。您必須儘快致力於恢復失敗的成員，從而避免產生任何進一步的不良影響。

步驟2.1.由於Sessionmgr09和Sessionmgr10已關閉，會話副本集已關閉，因此您必須從OSGI控制檯中的會話共用和共用儲存中刪除這些副本集條目：

```
#telnet qns0x 9091
```

```
osgi> listshards
```

```
Shard Id Mongo DB State Backup DB Removed Session Count
```

```
1 sessionmgr01:27717/session_cache online false false 109306
2 sessionmgr01:27717/session_cache_2 online false false 109730
3 sessionmgr01:27717/session_cache_3 online false false 109674
4 sessionmgr01:27717/session_cache_4 online false false 108957
```

```
osgi> listskshards
```

```
Shard Id Mongo DB State Backup DB Removed Session Count
```

```
2 sessionmgr02:37717/sk_cache online false false 150306
3 sessionmgr02:37717/sk_cache_2 online false false 149605
```

步驟2.2.移除以下作業階段分割塊：

```
osgi> removeshard 1
osgi> removeshard 2
osgi> removeshard 3
osgi> removeshard 4
```

步驟2.3.刪除以下磁碟碎片：

```
osgi> removeskshard 2
osgi> removeskshard 3
```

步驟 2.4.在執行重新平衡之前，請驗證管理資料庫（檢查所有qns VM的例項版本是否匹配）：

```
<#root>
```

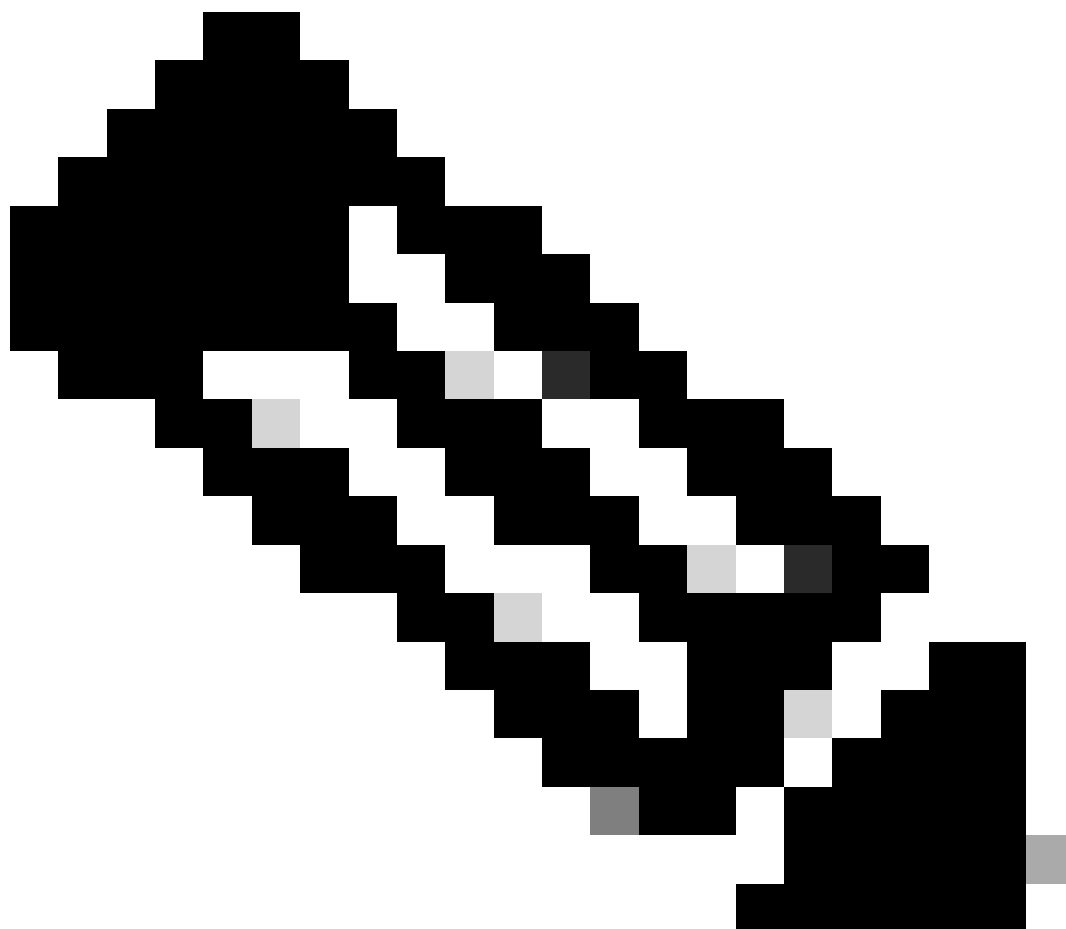
```
#mongo sessionmgrxx:xxxx/sharding. [Note: use the primary sessionmgr hostname and respective port
```

```
#set05:PRIMARY> db.instances.find()
{ "_id" : "qns02-1", "version" : 961 }
{ "_id" : "qns07-1", "version" : 961 }
{ "_id" : "qns08-1", "version" : 961 }
{ "_id" : "qns04-1", "version" : 961 }
{ "_id" : "qns08-1", "version" : 961 }
{ "_id" : "qns05-1", "version" : 961 }
```

Note: if the sharding versions (previous output) are different for some QNS instances. For example, if y

```
{ "_id" : "qns08-1", "version" : 961 }
{ "_id" : "qns04-1", "version" : 962 }
```

在管理分片資料庫上運行此命令（使用正確的主機名）：



附註：如果您位於輔助成員上，請使用rs.slaveOk()來運行命令。

```
[root@casant01-cm csv]# mongo sessionmgr01:27721/shardin
set05:PRIMARY>
set05:PRIMARY> db.instances.remove({ "_id" : "$QNS_hostname" })
```

Example:

```
set05:PRIMARY> db.instances.remove({ "_id" : "qns04-1" })
set05:PRIMARY> exit
```

步驟2.5.現在運行會話共用重新平衡。

Login to osgi console.

```
#telnet qns0x 9091
```

```
osgi>listshards
```

```
osgi>rebalance
```

```
osgi>rebalancestatus
```

Verify shards:

```
osgi>listshards
```

步驟2.6.運行分片重新平衡：

Login to osgi console.

```
#telnet qns0x 9091
```

```
osgi>listskshard
```

```
osgi>rebalancesk
```

```
osgi>rebalanceskstatus
```

Verify shards:

```
osgi>listshards
```

步驟2.7.刪除副本集set01a和set01g (在群集上運行)：

```
#build_set.sh --session --remove-replica-set --setname set01a --force
```

```
#build_set.sh --session --remove-replica-set --setname set01g --force
```

步驟2.8.重新啟動qns服務 (在群集上運行)：

```
#restartall.sh
```

步驟2.9.從mongoConfig.cfg檔案中刪除set01a和set01g行。在Cluster Manager上運行此命令：

```
#cd /etc/broadhop/
```

```
#/bin/cp -p mongoConfig.cfg mongoConfig.cfg_backup_
```

```
#vi mongoConfig.cfg
```

```
[SESSION-SET1]
SETNAME=set01a
OPLOG_SIZE=5120
ARBITER1=arbitervip:27717
ARBITER_DATA_PATH=/var/data/sessions.27717
MEMBER1=sessionmgr01:27717
MEMBER2=sessionmgr02:27717
DATA_PATH=/var/data/sessions.1/d
SHARD_COUNT=4
SEEDS=sessionmgr01:sessionmgr02:27717
[SESSION-SET1-END]
```

```
[SESSION-SET7]
SETNAME=set01g
OPLOG_SIZE=5120
ARBITER1=arbitervip:37717
ARBITER_DATA_PATH=/var/data/sessions.37717
MEMBER1=sessionmgr02:37717
MEMBER2=sessionmgr01:37717
DATA_PATH=/var/data/sessions.1/g
SHARD_COUNT=2
SEEDS=sessionmgr02:sessionmgr01:37717
[SESSION-SET7-END]
```

步驟2.10.刪除線路後，儲存並退出。

在Cluster Manager上運行build_etc。

```
#!/var/qps/install/current/scripts/build/build_etc.sh
```

步驟2.11.驗證是否已通過診斷刪除副本集set01d。

```
#diagnostics.sh --get_r
```

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）。