

CPS에서 세션 복제본이 설정된 경우의 가입자 세션 처리

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[배경 정보](#)

[문제](#)

[솔루션](#)

소개

이 문서에서는 CPS(Cisco Policy Suite)에서 세션 복제본을 설정한 경우의 가입자 세션 처리에 대해 설명합니다.

사전 요구 사항

요구 사항

다음 주제에 대한 지식을 보유하고 있으면 유용합니다.

- Linux
- CPS
- 몽고DB



참고: CPS CLI에 대한 루트 액세스 권한이 있어야 합니다.

사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

- CPS 20.2
- UCS(Unified Computing System)-B
- MongoDB-v3.6.17

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

배경 정보

CPS는 MongoDB를 사용하여 기본 데이터베이스 구조를 구성하기 위해 세션mgr 가상 컴퓨터

(VM)에서 Mongod 프로세스가 실행됩니다.

복제본 세트에 고가용성을 제공하기 위해 권장되는 최소 구성은 데이터 포함 멤버가 3개인 3개 멤버 복제본 세트입니다. 1개의 기본 멤버 및 2개의 보조 멤버. 일부 경우(예: 기본 및 보조 항목이 있지만 비용 제약으로 인해 다른 보조 항목을 추가할 수 없는 경우) 중재자를 포함하도록 선택할 수 있습니다. 중재자는 선거에 참여하지만 데이터를 보유하지 않습니다(즉, 데이터 중복성을 제공하지 않음). CPS의 경우, 일반적으로 세션 DB는 그러한 방식으로 구성된다.

/etc/broadhop/mongoConfig.cfg에서 CPS 설정에 대한 복제본 세트 컨피그레이션을 확인할 수 있습니다.

```
[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]
```

MongoDB에는 클러스터의 이중화와 속도를 돕는 Sharding이라는 또 다른 개념이 있습니다. 샤드는 데이터베이스를 인덱싱된 집합으로 분리하여 쓰기 속도를 크게 높여 전체적인 데이터베이스 성능을 향상시킵니다. 공유한 데이터베이스는 각 공유가 복제본 세트가 되도록 설정되는 경우가 많습니다.

- 세션 공유: 세션 샤드 시드 및 해당 데이터베이스:

```
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
```

- 보조 키 공유: 보조 키 샤드 시드 및 해당 데이터베이스:

```
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
```

문제

문제점 1. 단일 멤버 오류로 인한 데이터 멤버 메모리 소비의 꾸준한 증가.

데이터 베어링 멤버가 3개 멤버(2개 데이터 베어링 +1 중재자)에서 다운되면 나머지 데이터 베어링 멤버만 기본 및 복제본 세트의 역할을 수행하지만 DB 이중화 없이 무거운 로드 및 복제본 세트가 계속 작동합니다. 3개 멤버 PSA 아키텍처에서는 데이터 베어링 노드가 다운되면 캐시 압력이 증가합니다. 따라서 나머지 데이터 베어링 노드(기본)에 대한 메모리 소비량이 꾸준히 증가하므로 방치될 경우 사용 가능한 메모리가 고갈되어 노드가 실패하게 되고 결과적으로 복제본 세트 장애가 발생할 수 있습니다.

```
-----
|[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
-----
```

문제점 2. 이중 멤버 오류로 인해 세션 처리에 미치는 영향.

데이터 포함 멤버(Sessionmgr01 및 sessionmgr02)가 이러한 복제 집합에서 모두 다운되면(이중 실패) 전체 복제 집합이 다운되고 기본 데이터베이스 기능이 손상됩니다.

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

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

이 복제 세트 실패는 CPS 통화 처리 프로세스(qns(Quantum Network Suite) 프로세스)가 실패한 복제 세트에 이미 저장된 세션에 액세스할 수 없기 때문에 세션 복제 세트의 경우 통화 실패로 이어집니다.

솔루션

단일 멤버 장애의 경우 1번 접근 방식.

짧은 시간 동안 PSA(Primary-Secondary-Arbiter) 아키텍처에서 실패한 복제본 세트 멤버를 다시 가져올 수 있어야 합니다. PSA 아키텍처에서 장애가 발생한 데이터 베어링 멤버를 복원하는 데 시간이 걸리는 경우 장애가 발생한 멤버를 제거해야 합니다.

1.1단계. 3개 멤버 PSA 아키텍처로 특정 복제 데이터베이스 세트에서 장애가 발생한 데이터 베어링 멤버를 식별합니다. 클러스터 관리자에서 이 명령을 실행합니다.

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

1.2단계. 특정 복제 데이터베이스 집합에서 실패한 데이터 포함 멤버를 제거합니다.

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
```

1.3단계. 실패한 구성원이 복제 데이터베이스 집합에서 제거되었는지 확인합니다.

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

2중 멤버 실패의 경우 접근법 2.

3 PSA 복제 세트에서 두 데이터 베어링 멤버가 모두 다운되면 영구적인 해결 방법이 아닙니다. 각 복제본 세트, 세션 공유 및 sk에서 장애가 발생한 멤버를 제거하여 통화 오류를 방지 또는 줄이고 원활한 트래픽 처리를 보장하기 위한 임시 해결 방법입니다. 더 이상 원치 않는 영향을 피하기 위해 가능한 한 빨리 실패한 멤버의 복원 작업을 해야 합니다.

2.1단계. Sessionmgr09 및 Sessionmgr10이 다운되어 세션 복제 세트 항목이 다운되었으므로 OSGI 콘솔의 세션 샤드 및 스샤드에서 이러한 복제 세트 항목을 제거해야 합니다.

```
#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
```

2.2단계. 다음 세션 샤드를 제거합니다.

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

2.3단계. 다음 샤드를 제거합니다.

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

2.4단계. 리밸런스를 수행하기 전에 관리 DB를 확인합니다(모든 qns VM에 대해 인스턴스 버전이 일치하는지 확인).

<#root>

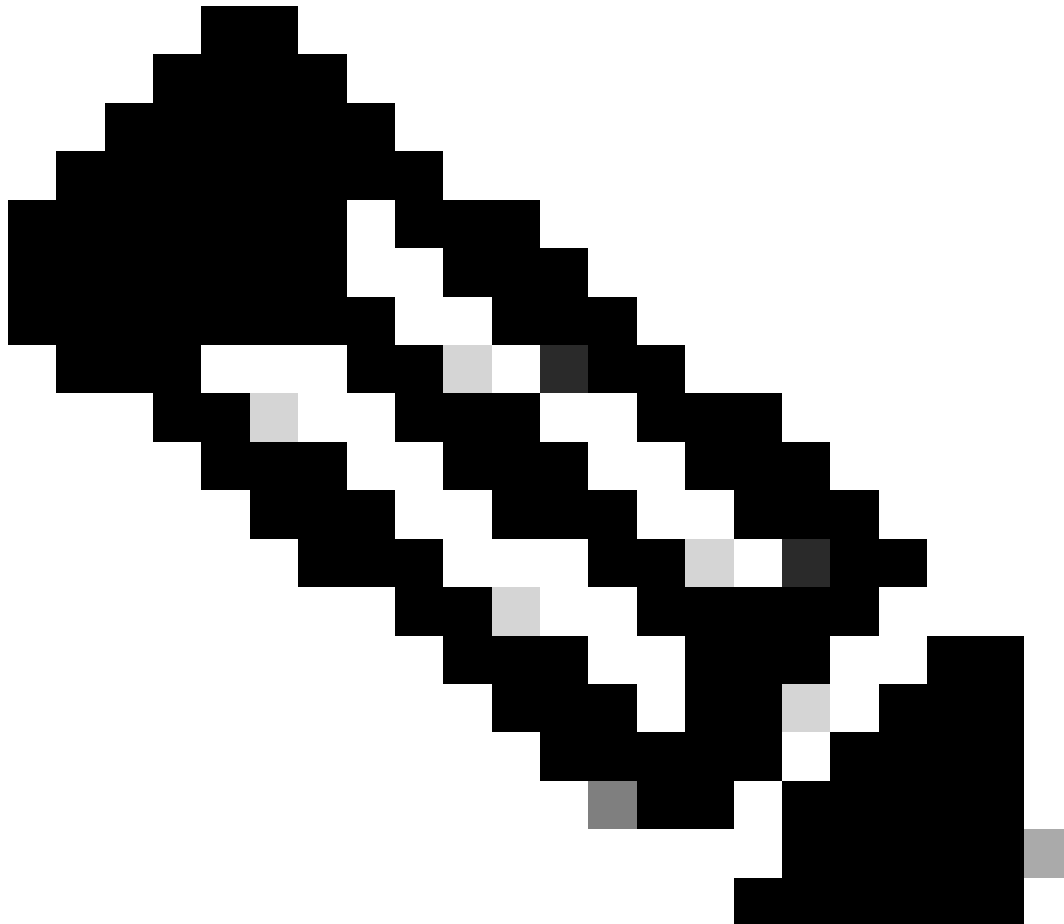
```
#mongo sessionmgrxx:xxxx/sharding. [Note: use the primary sessionmgr hostname and respective port]
```

```
#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 }
```

관리자 공유 DB에서 다음 명령을 실행합니다(올바른 호스트 이름 사용).



참고: 보조 멤버에 있는 경우 rs.slaveOk()를 사용하여 명령을 실행할 수 있습니다.

```
[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
```

2.5단계. 이제 세션 샤드 리밸런스를 실행합니다.

Login to osgi console.

```
#telnet qns0x 9091
```

```
osgi>listshards
```

```
osgi>rebalance
```

```
osgi>rebalancestatus
```

Verify shards:

```
osgi>listshards
```

2.6단계. SK 샤드 리밸런스를 실행합니다.

Login to osgi console.

```
#telnet qns0x 9091
```

```
osgi>listskshard
```

```
osgi>rebalancesk
```

```
osgi>rebalanceskstatus
```

Verify shards:

```
osgi>listshards
```

2.7단계. 복제 세트 set01a 및 set01g를 제거합니다(cluman에서 실행).

```
#build_set.sh --session --remove-replica-set --setname set01a --force  
#build_set.sh --session --remove-replica-set --setname set01g --force
```

2.8단계. qns 서비스를 다시 시작합니다(cluman에서 실행).

```
#restartall.sh
```

2.9단계. mongoConfig.cfg 파일에서 set01a 및 set01g 행을 제거합니다. 클러스터 관리자에서 실행:

```
#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]
```

2.10단계. 라인을 제거한 후 저장하고 종료합니다.

클러스터 관리자에서 build_etc를 실행합니다.

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

2.11단계. 진단 프로그램을 통해 복제 세트 set01d가 제거되었는지 확인합니다.

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

이 번역에 관하여

Cisco는 전 세계 사용자에게 다양한 언어로 지원 콘텐츠를 제공하기 위해 기계 번역 기술과 수작업 번역을 병행하여 이 문서를 번역했습니다. 아무리 품질이 높은 기계 번역이라도 전문 번역가의 번역 결과물만큼 정확하지는 않습니다. Cisco Systems, Inc.는 이 같은 번역에 대해 어떠한 책임도 지지 않으며 항상 원본 영문 문서(링크 제공됨)를 참조할 것을 권장합니다.