# 在CPS中將CRD從錯誤狀態恢復的過程

## 目錄

## 簡介

本檔案介紹將思科原則套件(CPS)自訂參考資料(CRD)表從BAD狀態還原的過程。

## 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許可權訪問許可權：

- 對CPS CLI的Root訪問許可權
- 「qns-svn」使用者訪問CPS GUI（策略生成器和CPS Central）

### 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。

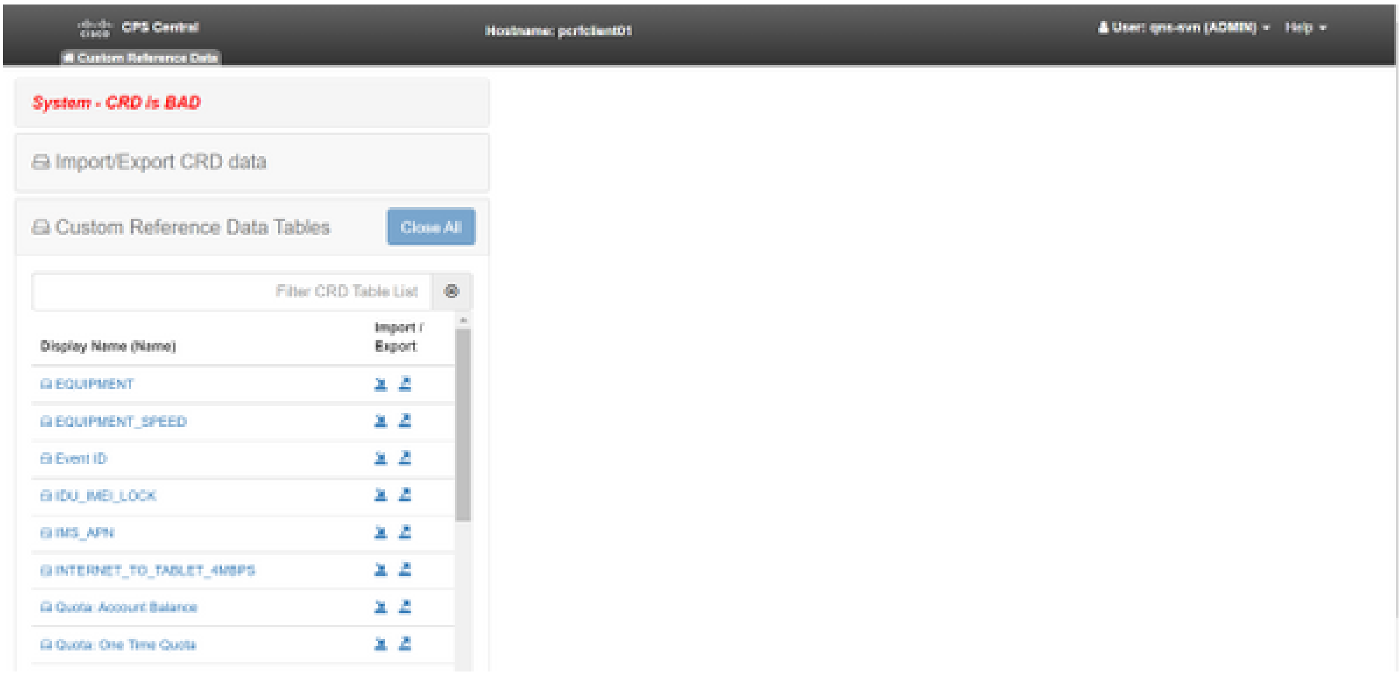
## 背景資訊

在CPS中，CRD表用於儲存從策略生成器發佈並與會話管理器上託管MongoDB例項中存在的CRD DB關聯的自定義策略配置資訊。通過CPS中央GUI在CRD表中執行匯出和匯入操作，以便處理CRD表資料。

# 問題

如果在執行匯入所有操作時出現任何型別的錯誤，則CPS會停止該進程，將系統設定為BAD狀態並阻止CRD API執行。CPS向客戶端傳送錯誤響應，指示系統處於BAD狀態。如果系統處於BAD狀態，並且重新啟動量子網路套件(QNS)/使用者資料通道(UDC)伺服器，則使用golden-crd資料構建CRD快取。如果系統BAD狀態為FALSE，則使用MongoDB構建CRD快取。

以下是CPS中心錯誤影象以供參考。

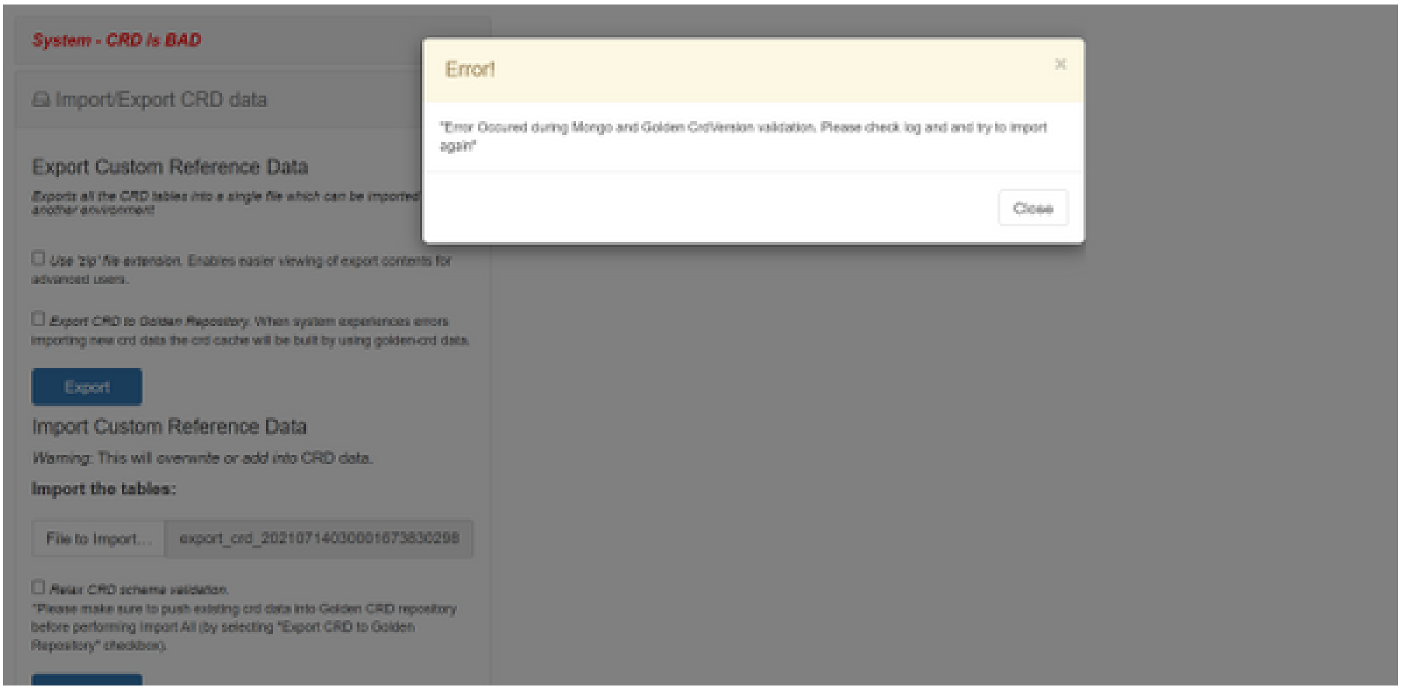


System - CRD is BAD

Import/Export CRD data

Custom Reference Data Tables

| Display Name (Name) | Import / Export |
|---|---|
| EQUIPMENT | |
| EQUIPMENT_SPEED | |
| Event ID | |
| IDU_IMEI_LOCK | |
| IMS_APN | |
| INTERNET_TO_TABLET_4MBPS | |
| Quota: Account Balance | |
| Quota: One Time Quota | |



System - CRD is BAD

Import/Export CRD data

Export Custom Reference Data

*Exports all the CRD tables into a single file which can be imported another environment*

□ *Use 'zip' file extension.* Enables easier viewing of export contents for advanced users.

□ *Export CRD to Golden Repository.* When system experiences errors importing new crd data the crd cache will be built by using golden-crd data.

Import Custom Reference Data

*Warning:* This will overwrite or add into CRD data.

**Import the tables:**

export_crd_20210714030001673830298

□ *Relax CRD schema validation.*
"Please make sure to push existing crd data into Golden CRD repository before performing Import All (by selecting "Export CRD to Golden Repository" checkbox).

Error!

"Error Occured during Mongo and Golden CrdVersion validation. Please check log and and try to import again"

如果CRD系統損壞，則：

1. CRD操作被阻止。您只能檢視資料。
2. 阻止除_import_all、_list和_query之外的CRD API。
3. QNS重啟從golden-crd位置提取CRD資料。
4. 重新啟動QNS/UDC既不修復系統BAD狀態也不修復呼叫丟棄，它只從golden-crd構建CRD快取。
5. 使用golden-crd資料構建的CRD快取。如果系統BAD狀態為FALSE，則使用MongoDB構建crd快取。

以下是CPS qns.log中的關聯消息：

```
qns02 qns02 2021-07-29 11:16:50,820 [pool-50847-thread-1]
INFO c.b.c.i.e.ApplicationInterceptor - System -
CRD is in bad state. All CRD APIs (except import all, list and query),
are blocked and user is not allowed to use.
Please verify your crd schema/crd data and try again!
qns02 qns02 2021-07-28 11:33:59,788 [pool-50847-thread-1]
WARN c.b.c.i.CustomerReferenceDataManager -
System is in BAD state. Data will be fetched from svn golden-crd repository.
qns01 qns01 2021-07-28 11:55:24,256 [pool-50847-thread-1]
WARN c.b.c.i.e.ApplicationInterceptor - ApplicationInterceptor: Is system bad: true
```

## 將CRD從BAD狀態還原的過程

### 方針1.

為了清除系統狀態，您需要從涉及從CPS中心匯入有效CRD資料的策略生成器中匯入有效且正確的CRD架構，如果全部匯入成功，則會清除系統狀態並解除阻止所有CRD API和操作。

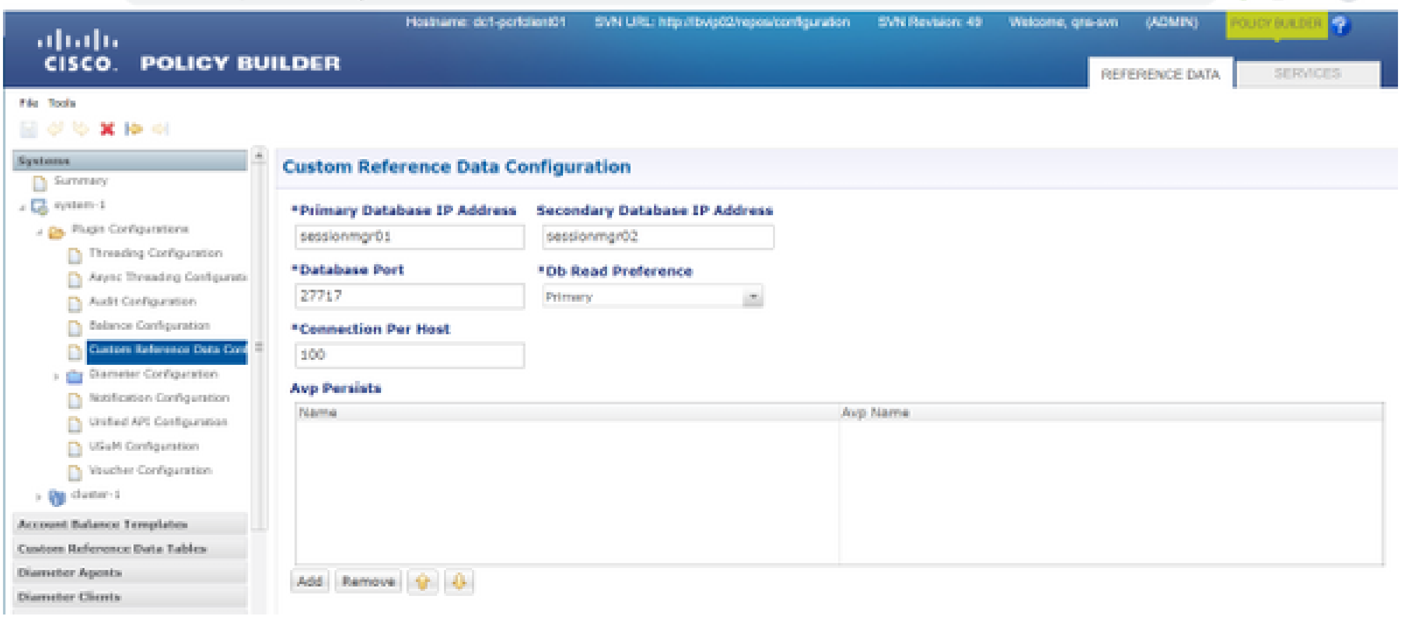
此處提供詳細步驟：

步驟1.運行此命令以備份CRD資料庫。

```
Command template:
#mongodump --host <session_manager> --port <cust_ref_data_port>
--db cust_ref_data -o cust_ref_data_backup

Sample command:
#mongodump --host sessionmgr01 --port 27717 --db cust_ref_data -o cust_ref_data_backup
```

**附註**：有關CRD DB主機和埠的資訊，請參閱PB中的自定義參考資料配置，如下圖所示。



Custom Reference Data Configuration

*Primary Database IP Address: sessionmgr01

Secondary Database IP Address: sessionmgr02

*Database Port: 27717

*Db Read Preference: Primary

*Connection Per Host: 100

Avp Persists

| Name | Avp Name |
|---|---|

步驟2.使用此過程刪除CRD表（整個DB）。

步驟2.1.登入到存在CRD DB的mongo例項。

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

Sample command:
#mongo --host sessionmgr01 --port 27717
```

步驟2.2.運行此命令以顯示mongo例項中存在的所有資料庫。

```
set01:PRIMARY> show dbs
admin 0.031GB
config 0.031GB
cust_ref_data 0.125GB
local 5.029GB
session_cache 0.031GB
sk_cache 0.031GB
set01:PRIMARY>
```

步驟2.3.運行此命令以切換到CRD DB。

```
set01:PRIMARY> use cust_ref_data
switched to db cust_ref_data
set01:PRIMARY
```

步驟2.4.運行此命令以刪除CRD DB。

```
set01:PRIMARY> db.dropDatabase()
{
"dropped" : "cust_ref_data",
"ok" : 1,
"operationTime" : Timestamp(1631074286, 13),
"$clusterTime" : {
"clusterTime" : Timestamp(1631074286, 13),
"signature" : {
"hash" : BinData(0,"AAAAAAAAAAAAAAAAAAAAAAAAAAA="),
"keyId" : NumberLong(0)
}}}
set01:PRIMARY>
```

步驟3.使用**show dbs**命令驗證不存在名為cust_ref_data的資料庫。

```
set01:PRIMARY> show dbs
admin 0.031GB
config 0.031GB
local 5.029GB
session_cache 0.031GB
sk_cache 0.031GB
set01:PRIMARY>
```

步驟4.使用「qns-svn」使用者登入策略生成器並發佈有效的CRD架構。

步驟5.使用**restartall.sh**從群集管理器重新啟動qns進程。

步驟6.驗證診斷是否正常，以及CRD表中沒有條目。CRD表中必須只有架構，即沒有任何資料。

步驟7.使用「qns-svn」使用者登入到CPS Central，然後匯入有效的CRD資料。

步驟8.驗證「import all returns successful」消息和「system - CRD is BAD」錯誤消息（未顯示在CPS Central中）。

步驟9.驗證是否已解除鎖定所有CRD API，您現在可以處理CRD資料。

如果第一種方法不起作用，則選擇第二種方法。

## 方針2.

步驟1.使用**diagnostics.sh —get_r**命令確定託管DB Mongo例項的主機和埠。

```
[root@installer ~]# diagnostics.sh --get_r
CPS Diagnostics HA Multi-Node Environment
---------------------------
Checking replica sets...
|-----------------------------------------------------------------------------------------------
-----------------------------------------|
| Mongo:v3.6.17 MONGODB REPLICA-SETS STATUS INFORMATION Date : 2021-09-14 02:56:23 |
|-----------------------------------------------------------------------------------------------
-----------------------------------------|
| SET NAME - PORT : IP ADDRESS - REPLICA STATE - HOST NAME - HEALTH - LAST SYNC - PRIORITY |
|-----------------------------------------------------------------------------------------------
-----------------------------------------|
| ADMIN:set06 |
| Status via arbitervip:27721 sessionmgr01:27721 sessionmgr02:27721 |
| Member-1 - 27721 : - PRIMARY - sessionmgr01 - ON-LINE - -------- - 3 |
| Member-2 - 27721 : - SECONDARY - sessionmgr02 - ON-LINE - 1 sec - 2 |
| Member-3 - 27721 : 192.168.10.146 - ARBITER - arbitervip - ON-LINE - -------- - 0 |
|-----------------------------------------------------------------------------------------------
```

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

步驟2.登入到存在ADMIN DB的mongo例項。

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

Sample Command:
#mongo --host sessionmgr01 --port 27721
```

步驟3.運行此命令以顯示mongo例項中存在的所有資料庫。

```
set06:PRIMARY> show dbs
admin 0.078GB
config 0.078GB
diameter 0.078GB
keystore 0.078GB
local 4.076GB
policy_trace 2.078GB
queueing 0.078GB
scheduler 0.078GB
sharding 0.078GB
set06:PRIMARY>
```

步驟4.運行此命令以切換到管理資料庫。

```
set06:PRIMARY> use admin
switched to db admin
set06:PRIMARY>
```

步驟5.運行此命令可顯示ADMIN DB中存在的所有表。

```
set06:PRIMARY> show tables
state
system.indexes
system.keys
system.version
set06:PRIMARY>
```

步驟6.運行此命令以檢查系統的當前狀態。

```
set06:PRIMARY> db.state.find()
{ "_id" : "state", "isSystemBad" : true, "lastUpdatedDate" : ISODate("2021-08-11T15:01:13.313Z")
}
set06:PRIMARY>
```

在這裡，您可以看到「**isSystemBad**」 :**真的**。因此，您必須將此欄位更新為「**false**」，才能使用下一步中提供的命令清除CRD BAD狀態。

步驟7.使用命令**db.state.updateOne({_id:"state"},{$set:{isSystemBad:false}})更新欄位 "isSystemBAD"**。

```
set06:PRIMARY> db.state.updateOne({_id:"state"},{$set:{isSystemBad:false}})
{ "acknowledged" : true, "matchedCount" : 0, "modifiedCount" : 0 }
set06:PRIMARY>
```

步驟8.運行命令**db.state.find()**以檢查**isSystemBad**欄位值是否已更改為**false。**

```
set06:PRIMARY> db.state.find()
{ "_id" : "state", "isSystemBad" : false, "lastUpdatedDate" : ISODate("2021-08-
11T15:01:13.313Z") }
set06:PRIMARY>
```

步驟9.驗證所有CRD API是否已解鎖，您現在可以處理CRD資料。