

Procedimiento para restaurar CRD de estado incorrecto en CPS

Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Antecedentes](#)

[Problema](#)

[Procedimiento para restaurar CRD del estado BAD](#)

[Enfoque 1.](#)

[Enfoque 2.](#)

Introducción

Este documento describe el procedimiento para restaurar la tabla de datos de referencia personalizados (CRD) de Cisco Policy Suite (CPS) desde el estado BAD.

Prerequisites

Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- Linux
- CPS
- MongoDB

Cisco recomienda que tenga acceso privilegiado:

- Acceso raíz a CPS CLI
- acceso de usuario "qns-svn" a las GUI de CPS (Generador de políticas y CPS Central)

Componentes Utilizados

La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- CPS 20.2
- MongoDB v3.6.17
- UCS-B

The information in this document was created from the devices in a specific lab environment. All of

the devices used in this document started with a cleared (default) configuration. Si tiene una red en vivo, asegúrese de entender el posible impacto de cualquier comando.

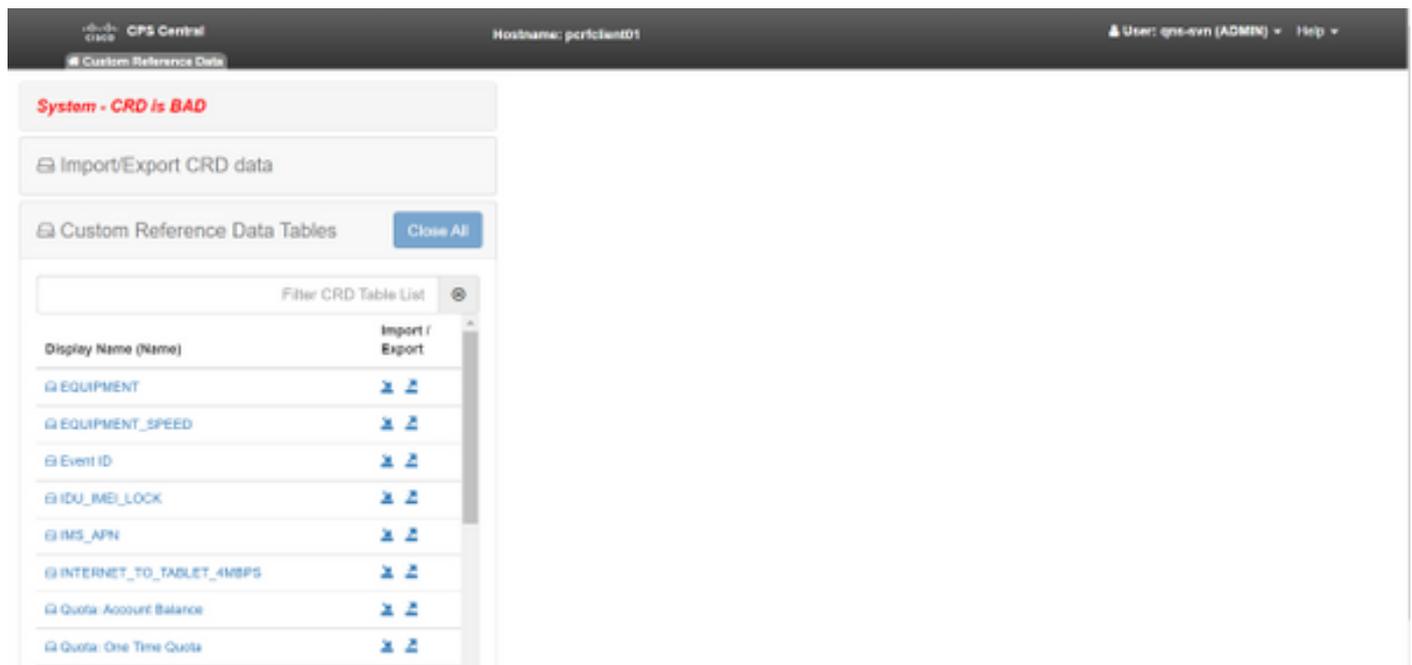
Antecedentes

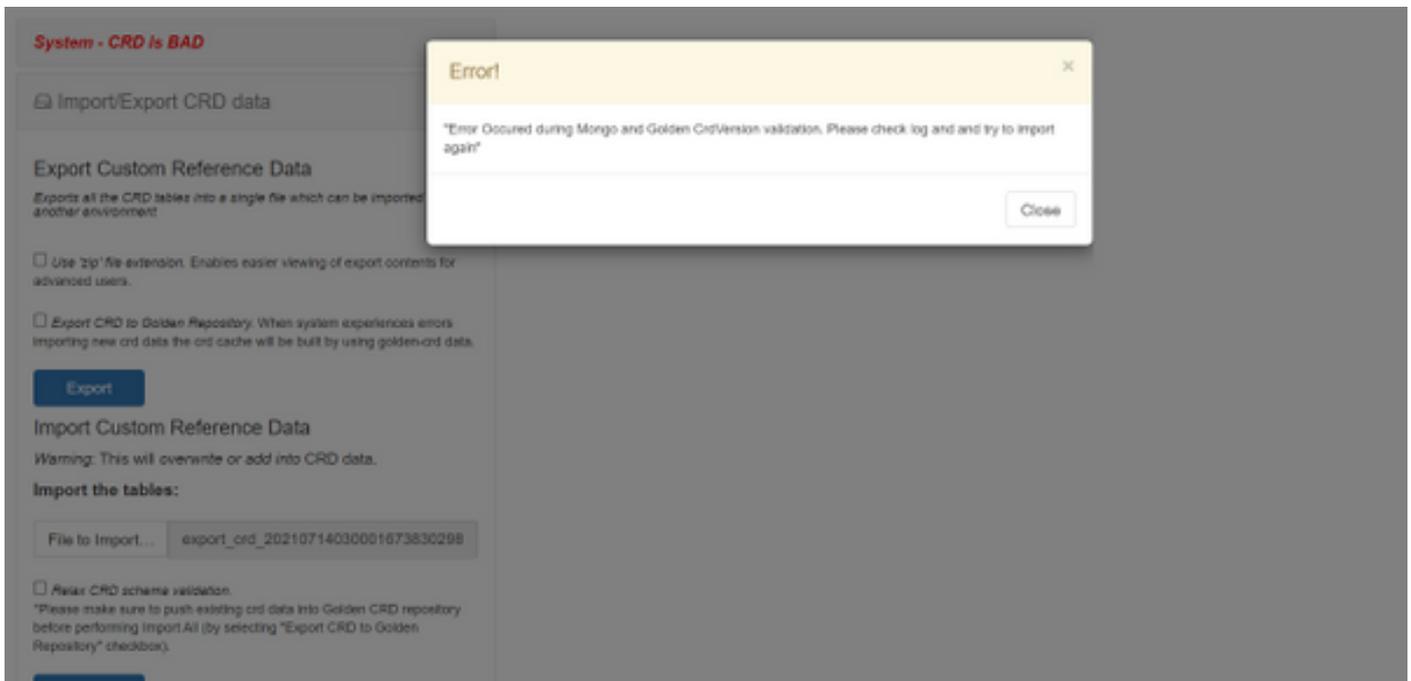
En CPS, la tabla CRD se utiliza para almacenar la información de configuración de políticas personalizada que se publica desde Policy Builder y se asocia con CRD DB que está presente en la instancia de MongoDB alojada en sessionmgr. Las operaciones de exportación e importación se realizan en la tabla CRD a través de la GUI central de CPS para manipular los datos de la tabla CRD.

Problema

Si se produce algún tipo de error al importar todas las operaciones, CPS detiene el proceso, establece el sistema en estado BAD y bloquea la ejecución de las API CRD. CPS envía una respuesta de error al cliente que indica que el sistema está en estado BAD. Si el sistema se encuentra en estado BAD y se reinicia el servidor Quantum Network Suite (QNS)/User Data Channel (UDC), la caché de CRD se genera mediante el uso de datos de código oro. Si el estado BAD del sistema es FALSO, la memoria caché CRD se genera con MongoDB.

A continuación se muestran las imágenes de error de CPS Central para referencia.





Si el sistema CRD es MALO, entonces:

1. La manipulación de CRD está bloqueada. Solo puede ver los datos.
2. Las API CRD, excepto `_import_all`, `_list`, `_query`, están bloqueadas.
3. El reinicio de QNS recoge los datos de CRD de la ubicación de la tarjeta dorada.
4. Un reinicio de QNS/UDC no corrige el estado BAD del sistema ni las caídas de llamadas, solamente genera la memoria caché CRD de la tarjeta dorada.
5. Caché CRD generada con datos de tarjeta dorada. Si el estado BAD del sistema es FALSO, entonces la memoria caché de crd se genera con MongoDB.

Estos son los mensajes asociados en CPS qns.log:

```
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.e.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
```

Procedimiento para restaurar CRD del estado BAD

Enfoque 1.

Para borrar el estado del sistema, debe importar un esquema CRD válido y correcto de Policy Builder que implique la importación de datos CRD válidos de CPS Central. Si la importación de todo es exitosa, borra el estado del sistema y se desbloquean todas las API CRD y todas las operaciones.

A continuación se indican los pasos detallados:

Paso 1. Ejecute este comando para realizar una copia de seguridad de la base de datos CRD.

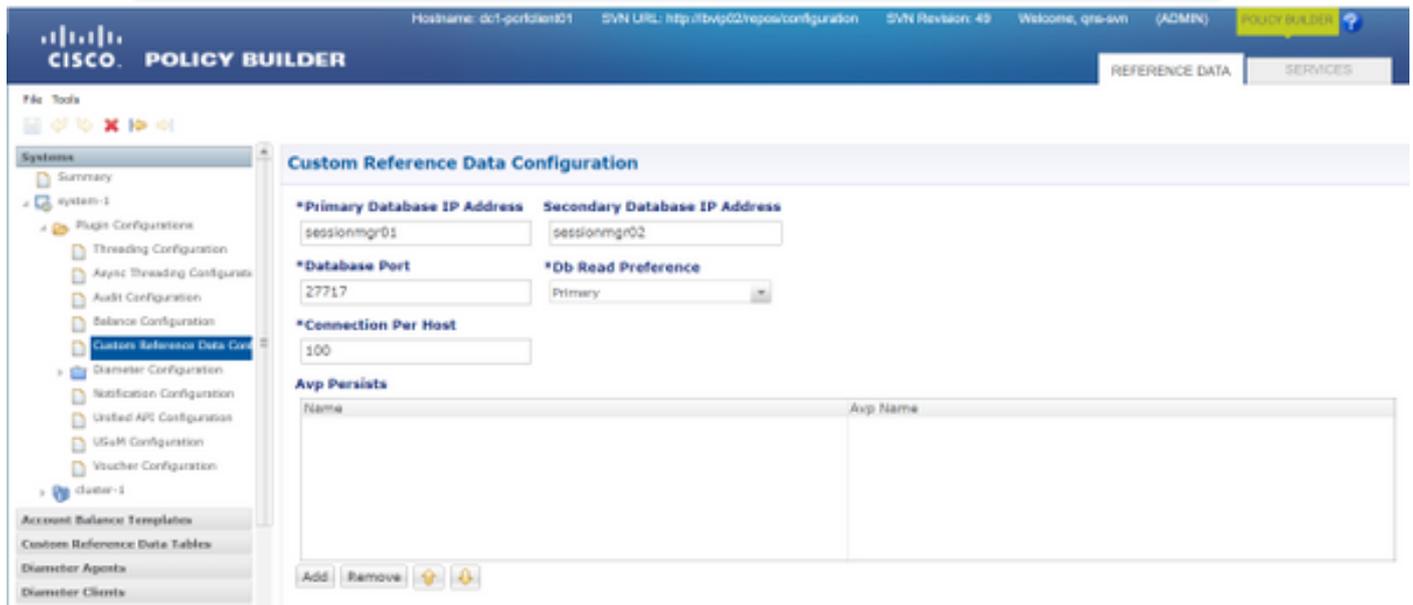
Command template:

```
#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
```

Nota: Para host y puerto CRD DB, consulte Configuración de datos de referencia personalizada en PB como se muestra en esta imagen.



Paso 2. Suelte la tabla CRD (toda la base de datos) con el uso de este procedimiento.

Paso 2.1. Inicie sesión en la instancia de mongo donde está presente CRD DB.

Command template:

```
#mongo --host <sessionmgrXX> --port <cust_ref_data_port>
```

Sample command:

```
#mongo --host sessionmgr01 --port 27717
```

Paso 2.2. Ejecute este comando para mostrar todas las bases de datos presentes en la instancia del mongo.

```
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>
```

Paso 2.3. Ejecute este comando para cambiar a CRD DB.

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

Paso 2.4. Ejecute este comando para Drop 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,"AAAAAAAAAAAAAAAAAAAAAAAAAAAA="),
      "keyId" : NumberLong(0)
    }
  }
}
set01:PRIMARY>
```

Paso 3. Verifique que no haya ninguna base de datos con el nombre cust_ref_data que exista con el comando **show dbs.**

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

Paso 4. Inicie sesión en Policy Builder con el usuario "qns-svn" y publique un esquema CRD válido.

Paso 5. Reinicie el proceso qns en todos los nodos con **restartall.sh desde Cluster Manager.**

Paso 6. Verifique que el diagnóstico esté bien y que no haya ninguna entrada en la tabla CRD. Sólo debe haber un esquema presente en las tablas CRD, es decir, sin ningún dato.

Paso 7. Inicie sesión en CPS Central con el usuario "qns-svn" e importe datos CRD válidos.

Paso 8. Verifique que, importar todos los mensajes devueltos correctamente y el mensaje de error "system - CRD is BAD" no se muestra en CPS Central.

Paso 9. Verifique que, todas las API de CRD ahora estén desbloqueadas, puede manipular los datos de CRD ahora.

Si el primer enfoque no funcionó, entonces siga el segundo.

Enfoque 2.

Paso 1. Identifique el host y el puerto en el que se aloja la instancia de ADMIN DB Mongo con el

comando `diagnostics.sh --get_r`.

```
[root@installer ~]# diagnostics.sh --get_r
CPS Diagnostics HA Multi-Node Environment
```

```
-----
Checking replica sets...
```

```
|-----|
| Mongo:v3.6.17 MONGODB REPLICASET STATUS INFORMATION Date : 2021-09-14 02:56:23 |
|-----|
| SET NAME - PORT : IP ADDRESS - REPLICASET 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 |
|-----|
|-----|
```

Paso 2. Inicie sesión en la instancia de mongo donde está presente ADMIN DB.

Command template:

```
#mongo --host <sessionmgrXX> --port <Admin_DB__port>
```

Sample Command:

```
#mongo --host sessionmgr01 --port 27721
```

Paso 3. Ejecute este comando para mostrar todas las bases de datos presentes en la instancia mongo.

```
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>
```

Paso 4. Ejecute este comando para cambiar a ADMIN DB.

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

Paso 5. Ejecute este comando para mostrar todas las tablas presentes en ADMIN DB.

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

Paso 6. Ejecute este comando para verificar el estado actual del sistema.

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

Aquí se puede ver que **"isSystemBad" : verdadero**. Por lo tanto, debe actualizar este campo a **"false"** para borrar el estado CRD BAD, con el comando proporcionado en el siguiente paso.

Paso 7. Actualice el campo "isSystemBAD" con el comando **db.state.updateOne({_id:"state"},{\$set:{isSystemBad:false}})**.

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

Paso 8. Ejecute el comando **db.state.find()** para verificar si el valor del campo **isSystemBad** ha cambiado a **false**.

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

Paso 9. Verifique que todas las API de CRD estén ahora desbloqueadas, puede manipular los datos de CRD ahora.