

# Solución de problemas de POD de posturas CEE 5G SMI

## Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Antecedentes](#)

[¿Qué es SMI?](#)

[¿Qué es SMI CEE?](#)

[¿Qué son los POD de CEE?](#)

[¿Qué es PGpool POD?](#)

[¿Qué es Postgres POD?](#)

[Problema](#)

[Troubleshoot](#)

[Solución Alternativa](#)

[Cierre CEE](#)

[Eliminar contenido de carpetas](#)

[Restauración de CEE](#)

[Comprobaciones posteriores](#)

[Verificar que las alertas se borren de CEE](#)

## Introducción

Este documento describe cómo implementar la solución alternativa para los problemas de reinicio del grupo de dispositivos (POD) (pgpool) de la infraestructura de microservicios de suscriptor (SMI).

## Prerequisites

## Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- Cisco SMI CEE (Ultra Cloud Core CEE)
- Plataforma de implementación nativa en la nube (CNDP) 5G o arquitectura SMI Bare Metal (BM)
- Dockers y Kubernetes

## Componentes Utilizados

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

hardware.

- SMI 2020.02.2.35
- Kubernetes v1.21.0

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

### ¿Qué es SMI?

Cisco SMI es una pila por capas de tecnologías y estándares de nube que permiten utilizar aplicaciones basadas en microservicios desde las unidades empresariales de Cisco Mobility, Cable y Broadband Network Gateway (BNG), todas ellas con funciones similares de gestión de suscriptores y requisitos de almacenamiento de datos similares.

Los atributos son:

- Pila de nube de capa (tecnologías y estándares) para proporcionar implementaciones de principio a fin y también para adaptarse a las infraestructuras de nube actuales de los clientes.
- Todas las aplicaciones comparten CEE para funciones que no son de aplicación (almacenamiento de datos, implementación, configuración, telemetría y alarma). Esto proporciona una interacción y experiencia uniformes para todos los puntos de contacto y puntos de integración de los clientes.
- Las aplicaciones y CEE se implementan en contenedores de microservicios y se conectan con una malla de servicio inteligente.
- API expuesta para la implementación, configuración y administración para habilitar la automatización.

### ¿Qué es SMI CEE?

CEE es una solución de software desarrollada para supervisar las aplicaciones móviles y de cable que se implementan en SMI. La CEE captura información (métricas clave) de las aplicaciones de forma centralizada para que los ingenieros puedan depurar y solucionar problemas.

CEE es el conjunto común de herramientas que se instalan para todas las aplicaciones. Incluye un centro de operaciones dedicado, que proporciona la interfaz de usuario (CLI) y API para administrar las herramientas de supervisión. Sólo hay un CEE disponible para cada clúster.

### ¿Qué son los POD de CEE?

Un POD es un proceso que se ejecuta en su clúster de Kubernetes. El POD encapsula una unidad granular que se conoce como contenedor. Un POD contiene uno o varios contenedores.

Kubernetes implementa uno o varios POD en un único nodo que puede ser una máquina física o virtual. Cada POD tiene una identidad discreta con una dirección IP interna y un espacio de puerto. Sin embargo, los contenedores dentro de un POD pueden compartir los recursos de

almacenamiento y red. CEE tiene una serie de POD que tienen funciones únicas. Pgpool y postgres se encuentran entre varios POD CEE.

## ¿Qué es PGpool POD?

Pgpool administra el conjunto de recursos Postgres para conexión, replicación, equilibrio de carga, etc. Pgpool es un middleware que funciona entre servidores PostgreSQL y una base de datos PostgreSQL.

## ¿Qué es Postgres POD?

Postgres admite la base de datos de lenguaje de consulta estructurado (SQL) con redundancia para almacenar alertas y paneles Grafana.

## Problema

Los POD de pgpool se reinician regularmente mientras los POD postgresql se ejecutan sin problemas.

Para mostrar las alertas, ingrese este comando:

```
show alerts active summary | include "POD_|k8s-pod-"
```

Aquí se muestra una alerta de ejemplo de CEE.

```
[pod-name-smf-data/podname] cee# show alerts active summary | include "POD_|k8s-pod-"  
k8s-pod-crashing-loop 1d9d2b113073 critical 12-15T21:47:39 pod-name-smf-data-mas  
Pod cee-podname/grafana-65cbdb9846-krfgq (grafana) is restarting 1.03 times / 5 minutes.  
POD_Restarted 04d42efb81de major 12-15T21:45:44 pgpool-67f48f6565-vjt Container=  
k8s_pgpool_pgpool-67f48f6565-vjttd_cee-podname_a9f68607-eac4-40a9-86ef-db8176e0a22a_1474 of pod=  
pgpool-... POD_Restarted f7657a0505c2 major 12-15T21:45:44 postgres-0 Container=  
k8s_postgres_postgres-0_cee-podname_59e0a768-6870-4550-8db3-32e2ab047ce2_1385 of pod= postgres-0  
in name... POD_Restarted 6e57ae945677 major 12-15T21:45:44 alert-logger-d96644d4 Container=  
k8s_alert-logger_alert-logger-d96644d4-dsc8h_cee-podname_2143c464-068a-418e-b5dd-  
ce1075b9360e_2421 of po... k8s-pod-crashing-loop 5b8e6a207aad critical 12-15T21:45:09 pod-name-  
smf-data-mas Pod  
cee-podname/pgpool-67f48f6565-vjttd (pgpool) is restarting 1.03 times / 5 minutes. POD_Down  
45a6b9bf73dc major 12-15T20:30:44 pgpool-67f48f6565-qbw Pod= pgpool-67f48f6565-qbw52 in  
namespace=  
cee-podname is DOWN for more than 15min POD_Down 4857f398a0ca major 12-15T16:40:44 pgpool-  
67f48f6565-vjt Pod= pgpool-67f48f6565-vjttd in namespace=  
cee-podname is DOWN for more than 15min k8s-pod-not-ready fc65254c2639 critical 12-11T21:07:29  
pgpool-67f48f6565-qbw Pod  
cee-podname/pgpool-67f48f6565-qbw52 has been in a non-ready state for longer than 1 minute. k8s-  
pod-not-ready 008b859e7333 critical 12-11T16:35:49 pgpool-67f48f6565-vjt Pod  
cee-podname/pgpool-67f48f6565-vjttd has been in a non-ready state for longer than 1 minute.
```

## Troubleshoot

Desde el maestro de Kubernetes, ingrese este comando:

```
kubectl describe pods -n
```

Aquí se muestra el ejemplo de salida de la descripción de POD. El resultado está truncado.

Events:

Type Reason Age From Message

-----

```
Normal Scheduled 14m default-scheduler Successfully assigned cee-pod-name-11/postgres-2
to pod-name-master-3
Normal Pulling 14m kubelet Pulling image "docker.10.192.x.x.nip.io/cee-2020.02.2.i38/
smi-libraries/postgresql/2020.02.2/postgres:1.3.0-946d87d"
Normal Pulled 13m kubelet Successfully pulled image "docker.10.192.x.x.nip.io/cee-2020.02.2.i38/
smi-libraries/postgresql/2020.02.2/postgres:1.3.0-946d87d" in 29.048094722s
Warning Unhealthy 12m kubelet Readiness probe failed: [bin][h][ir] >>> [2021-10-11 18:09:48]
pod is not ready
Warning Unhealthy 10m kubelet Readiness probe failed: [bin][h][ir] >>> [2021-10-11 18:11:18]
pod is not ready
Warning Unhealthy 10m kubelet Readiness probe failed: [bin][h][ir] >>> [2021-10-11 18:11:48]
pod is not ready
Warning Unhealthy 9m49s kubelet Readiness probe failed: [bin][h][ir] >>> [2021-10-11 18:12:18]
pod is not ready
Warning Unhealthy 9m19s kubelet Readiness probe failed: [bin][h][ir] >>> [2021-10-11 18:12:48]
pod is not ready
Warning Unhealthy 8m49s kubelet Readiness probe failed: [bin][h][ir] >>> [2021-10-11 18:13:18]
pod is not ready
Warning Unhealthy 8m19s kubelet Readiness probe failed: [bin][h][ir] >>> [2021-10-11 18:13:48]
pod is not ready
Warning Unhealthy 7m49s kubelet Readiness probe failed: [bin][h][ir] >>> [2021-10-11 18:14:18]
pod is not ready
Warning Unhealthy 7m19s kubelet Readiness probe failed: [bin][h][ir] >>> [2021-10-11 18:14:48]
pod is not ready
Warning BackOff 6m44s kubelet Back-off restarting failed container
```

**O bien**

Events:

Type Reason Age From Message

-----

```
Warning FailedScheduling 13m default-scheduler 0/5 nodes are available: 2 node(s)
didn't match Pod's node affinity/selector, 3 node(s) didn't find available persistent
volumes to bind.
Normal Scheduled 13m default-scheduler Successfully assigned cee-pod-name-11/postgres-0
to pod-name-master-1
Warning FailedScheduling 13m default-scheduler 0/5 nodes are available: 2 node(s)
didn't match Pod's node affinity/selector, 3 node(s) didn't find available
persistent volumes to bind.
Normal Pulling 13m kubelet Pulling image "docker.10.192.x.x.nip.io/cee-2020.02.2.i38/
smi-libraries/postgresql/2020.02.2/postgres:1.3.0-946d87d"
Normal Pulled 12m kubelet Successfully pulled image "docker.10.192.x.x.nip.io/
cee-2020.02.2.i38/smi-libraries/postgresql/2020.02.2/postgres:1.3.0-946d87d"
in 43.011763302s
Warning Unhealthy 7m20s kubelet Liveness probe failed: [bin][h][imm] >>>
[2021-10-11 18:09:16] My name is pg-postgres-0
```

**Solución Alternativa**

**Nota:** Este procedimiento no causa tiempo de inactividad en la aplicación.

## Cierre CEE

Para apagar la CEE, ingrese estos comandos de la CEE:

```
[pod-name-smf-data/podname] cee#  
[pod-name-smf-data/podname] cee# config terminal  
Entering configuration mode terminal  
[pod-name-smf-data/podname] cee(config)# system mode shutdown  
[pod-name-smf-data/podname] cee(config)# commit  
Commit complete.
```

Espera a que el sistema alcance el 100%

## Eliminar contenido de carpetas

Desde master-vip, SSH a cada una de las VM principales y elimine el contenido de estas carpetas: `/data/cee-podname/data-postgres-[0-2]`.

```
Master 1  
cloud-user@pod-name-smf-data-master-1:~$ sudo rm -rf /data/cee-podname/data-postgres-0  
Master 2  
cloud-user@pod-name-smf-data-master-2:~$ sudo rm -rf /data/cee-podname/data-postgres-1  
Master 3  
cloud-user@pod-name-smf-data-master-3:~$ sudo rm -rf /data/cee-podname/data-postgres-2
```

## Restauración de CEE

Para restaurar la CEE, ingrese estos comandos desde la CEE:

```
[pod-name-smf-data/podname] cee#  
[pod-name-smf-data/podname] cee# config terminal  
Entering configuration mode terminal  
[pod-name-smf-data/podname] cee(config)# system mode running  
[pod-name-smf-data/podname] cee(config)# commit  
Commit complete.
```

Espera a que el sistema alcance el 100%.

## Comprobaciones posteriores

Verifique a Kubernetes desde el maestro.

```
cloud-user@pod-name-smf-data-master-1:~$ kubectl get pods -A -o wide | egrep 'postgres|pgpool'  
All pods should display up and running without any restarts
```

## Verificar que las alertas se borren de CEE

Para verificar que las alertas se borren de la CEE, ingrese este comando:

```
show alerts active summary | include "POD_|k8s-pod-"
```

Además, puede ingresar este comando para asegurarse de que hay una base de datos maestra y dos en espera:

```

echo "0-----";kubectl
exec -it postgres-0 -n $(kubectl get pods -A | grep postgres | awk '{print $1}' | head -1)
-- /usr/local/bin/cluster/healthcheck/is_major_master.sh;echo "1-----"
-----";kubectl exec -it postgres-1 -n $(kubectl get pods -A | grep postgres | awk '{print
$1}'
| head -1) -- /usr/local/bin/cluster/healthcheck/is_major_master.sh;echo "2-----"
-----"; kubectl exec -it postgres-2 -n $(kubectl get pods -A | grep postgres |
awk '{print $1}' | head -1) -- /usr/local/bin/cluster/healthcheck/is_major_master.sh;
El ejemplo de resultado esperado es:

```

```

cloud-user@pod-name-smf-data-master-1:~$ echo "0-----";kubectl
exec -it postgres-0 -n $(kubectl get pods -A | grep postgres | awk '{print $1}' | head -1)
-- /usr/local/bin/cluster/healthcheck/is_major_master.sh;echo "1-----"
-----";kubectl exec -it postgres-1 -n $(kubectl get pods -A | grep postgres | awk '{print
$1}'
| head -1) -- /usr/local/bin/cluster/healthcheck/is_major_master.sh;echo "2-----"
-----"; kubectl exec -it postgres-2 -n $(kubectl get pods -A | grep postgres |
awk '{print $1}' | head -1) -- /usr/local/bin/cluster/healthcheck/is_major_master.sh;
0-----
[bin][h][imm] >>> [2021-12-15 22:05:18] My name is pg-postgres-0
[bin][h][imm] >>> My state is good.
[bin][h][imm] >>> I'm not a master, nothing else to do!
1-----
[bin][h][imm] >>> [2021-12-15 22:05:19] My name is pg-postgres-1
[bin][h][imm] >>> My state is good.
[bin][h][imm] >>> I think I'm master. Will ask my neighbors if they agree.
[bin][h][imm] >>> Will ask nodes from PARTNER_NODES list
[bin][h][imm] >>> Checking node pg-postgres-0
[bin][h][imm] >>>>>>> Count of references to potential master pg-postgres-1 is 1 now
[bin][h][imm] >>> Checking node pg-postgres-1
[bin][h][imm] >>> Checking node pg-postgres-2
[bin][h][imm] >>>>>>> Count of references to potential master pg-postgres-1 is 2 now
[bin][h][imm] >>> Potential masters got references:
[bin][h][imm] >>>>> Node: pg-postgres-1, references: 2
[bin][h][imm] >>> I have 2/2 incoming reference[s]!
[bin][h][imm] >>>> 2 - Does anyone have more?
[bin][h][imm] >>> Yahoo! I'm real master...so I think!
2-----
[bin][h][imm] >>> [2021-12-15 22:05:21] My name is pg-postgres-2
[bin][h][imm] >>> My state is good.
[bin][h][imm] >>> I'm not a master, nothing else to do!

```