

Solución de Problemas de Hardware en el Clúster 5G SMI CNDP y Mantenimiento

Contenido

[Introducción](#)

[Prerequisites](#)

[Requirements](#)

[Componentes Utilizados](#)

[Antecedentes](#)

[¿Qué es SMI?](#)

[¿Qué es SMI-BM o CNDP?](#)

[¿Qué es SMI Cluster Manager?](#)

[¿Qué es SMI Cluster Deployer?](#)

[Problema](#)

[Procedimiento de mantenimiento](#)

Introducción

Este documento describe el procedimiento para realizar el mantenimiento (sustitución o mantenimiento de hardware), como la actualización del firmware (FW), etc., en la infraestructura de microservicios de suscriptor 5G (SMI) Plataforma de implementación nativa en la nube (CNDP) Conjunto de dispositivos (POD).

Prerequisites

Requirements

Cisco recomienda que tenga conocimiento sobre estos temas:

- SMI de Cisco
- Arquitectura 5G CNDPA o SMI-Bare-metal (BM)
- Dockers y kubernetes
- Servidores de la serie Cisco UCS C220

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
- Cisco UCS C220-M5SX-CM

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 de las unidades empresariales de Cisco Mobility, Cable y BNG, todas ellas con funciones similares de gestión de suscriptores y requisitos de almacenamiento de datos similares.

Atributos:

- 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.
- Entorno de ejecución común compartido por todas las aplicaciones para funciones que no son de aplicación (almacenamiento de datos, implementación, configuración, telemetría, alarma). Esto proporciona una interacción y experiencia uniformes para todos los puntos de contacto y puntos de integración de los clientes.
- Las aplicaciones y el entorno de ejecución común se implementan en contenedores de microservicios y se conectan con una malla de servicio inteligente.
- API expuesta para implementación, configuración y gestión, para habilitar la automatización.

¿Qué es SMI-BM o CNDP?

Cisco SMI-Bare Metal o CNDP es una plataforma sin software específico que proporciona la infraestructura necesaria para implementar las funciones de red virtual (VNF) y las funciones nativas de la nube (CNF), y permite las unidades empresariales de Cisco Mobility, Cable y BNG.

Atributos:

- Sin software específico que elimina la sobrecarga relacionada con VIM
- Rendimiento mejorado Más núcleos para la aplicación Ejecución de aplicaciones más rápida
- Flujo de trabajo de implementación automatizado; integrado con NSO CFP
- Pila curada para implementar Cisco 5G NF
- Guía de pedidos e implementación simplificada

¿Qué es SMI Cluster Manager?

Un cluster manager es un cluster keepalived de 2 nodos utilizado como punto inicial para la implementación del cluster del plano de control y del plano de usuario. Ejecuta un clúster kubernetes de un solo nodo y un conjunto de POD responsables de toda la configuración del clúster. Sólo el administrador principal del clúster está activo y el secundario sólo se hace cargo en caso de fallo o se desactiva manualmente para mantenimiento.

¿Qué es SMI Cluster Deployer?

SMI Deployer es un servicio en Cluster Manager que puede crear VM, personalizar Host OS, crear clúster K8, iniciar K8 Master, configurar clúster e iniciar Aplicaciones, etc.

Problema

El mantenimiento del hardware, como la falla de hardware o la actualización de software/fw, etc., necesita tiempo de inactividad en los servidores. ¿Qué procedimiento debe seguirse para el mantenimiento que se realizará en el POD? Cómo detener correctamente los servicios para evitar tiempos de inactividad no deseados en la aplicación.

Procedimiento de mantenimiento

Obtenga el VIP del administrador del clúster, el VIP principal de Kubernetes (para la aplicación correspondiente), UCS CIMC IP, UCS CIMC Name y el nombre de host del servidor (nombre de host del sistema operativo) en el que se realizará el mantenimiento.

El inicio de sesión en kubernetes master corresponde al servicio y asegúrese de que todos los POD estén en condición **En ejecución**.

Ejemplo de resultado

```
cloud-user@pod-name-smf-data-master-1:~$ kubectl get pods -A | grep -v Running
NAMESPACE NAME READY STATUS RESTARTS AGE
```

2. Inicie sesión en el clúster manager y acceda al centro de operaciones de implementación de clústeres SMI (aquí está el procedimiento para encontrar la IP del centro Ops).

```
kubectl get svc -n $(kubectl get ns | grep -i smi-cm | awk '{print $1}') | grep ^ops-center
(Here "smi-cm" is the namespace in which cluster deployer is hosted and the "ops-center" is the starting name of the cluster deployer service name which is "ops-center-smi-cluster-deployer" these names can vary based on the environment setup)
```

Ejemplo de resultado

```
cloud-user@tp-tam-deployer-cm-primary:~$ kubectl get svc -n $(kubectl get ns | grep smi-cm | awk '{print $1}') | grep ^ops-center
ops-center-smi-cluster-deployer ClusterIP 10.100.x.x <none>
8008/TCP,2024/TCP,2022/TCP,7681/TCP,3000/TCP,3001/TCP 154d
```

3. Inicie sesión con este comando.

```
ssh -p 2024 admin@10.100.x.x
(2024 is the port used to connect to cluster deployer)
```

4. Verifique que los servicios se correspondan con la aplicación con el comando **show cluster**.

Ejemplo de resultado

```
Welcome to the Cisco SMI Cluster Deployer on tp-tam-deployer-cm-primary
Copyright © 2016-2020, Cisco Systems, Inc.
All rights reserved.
```

```

admin connected from 192.x.x.x using ssh on ops-center-smi-cluster-deployer-5cdc5f94db-bnxqt
[tp-tam-deployer-cm-primary] SMI Cluster Deployer# show clusters
LOCK TO
NAME VERSION
-----
pod-name-smf-data -
pod-name-smf-ims -
pod1-name-smf-data -
pod1-name-smf-ims -
pod2-name-aio-1 -
pod2-name-aio-2 -
pod2-name-upf-data -
pod2-name-upf-ims -

```

5. Dibuje el nodo en el que realiza el mantenimiento con estos comandos y escriba **Yes** (esto evacuará los POD de forma correcta y se reiniciará en otros nodos según sea necesario).

Ejemplo de resultado

```

[cluster-name-cm-1] SMI Cluster Deployer# clusters cluster-name nodes worker-11 actions sync
drain remove-node true

```

This will run drain on the node, disrupting pods running on the node. Are you sure? [no,yes] yes
message accepted

6. Mueva el nodo al modo de mantenimiento con estos comandos (esto podría tardar hasta un máximo de 30 minutos).

Ejemplo de resultado

```

[cluster-name-cm-1] SMI Cluster Deployer# config
Entering configuration mode terminal
[cluster-name-cm-1] SMI Cluster Deployer(config)# clusters cluster-name
[cluster-name-cm-1] SMI Cluster Deployer(config-clusters-cluster-name)# nodes worker-11
[cluster-name-cm-1] SMI Cluster Deployer(config-nodes-worker1)# maintenance true
[cluster-name-cm-1] SMI Cluster Deployer(config-nodes-worker1)# commit
Commit complete.
[cluster-name-cm-1] SMI Cluster Deployer(config-nodes-worker1)# end

```

7. Verifique los registros para ver el estado.

```
clusters cluster-name nodes worker-11 actions sync logs
```

(In this we are dealing with the worker-11 node)

Ejemplo de resultado (truncado):

```
logs 2022-01-03 06:04:02.755 DEBUG cluster_sync.cluster-name.worker-11: Cluster name: cluster-
name
```

```
2022-01-03 06:04:02.755 DEBUG cluster_sync.cluster-name.worker-11: Node name: worker-11
```

```
2022-01-03 06:04:02.755 DEBUG cluster_sync.cluster-name.worker-11: debug: false
```

```
2022-01-03 06:04:02.755 DEBUG cluster_sync.cluster-name.worker-11: remove_node: false
```

```
PLAY [Check required variables] *****
```

```
TASK [Gathering Facts] *****
```

Monday 03 January 2022 06:04:06 +0000 (0:00:00.014)

0:00:00.014 *****

ok: [worker-11]

ok: [worker-13]

ok: [worker-11]

ok: [worker-16]

ok: [worker-18]

ok: [worker-17]

ok: [worker-12]

ok: [worker-10]

ok: [worker-19]

ok: [worker-2]

ok: [master-1]

ok: [worker-11]

ok: [worker-15]

ok: [master-3]

ok: [worker-20]

ok: [worker-22]

ok: [worker-21]

....

TASK [Check node_name] *****

Monday 03 January 2022 06:04:13 +0000 (0:00:07.086)

0:00:07.101 *****

skipping: [master-1]

skipping: [master-2]

skipping: [master-3]

skipping: [worker-1]

skipping: [worker-10]

skipping: [worker-11]

skipping: [worker-12]

skipping: [worker-13]

skipping: [worker-11]

skipping: [worker-15]

skipping: [worker-16]

skipping: [worker-17]
skipping: [worker-18]
skipping: [worker-19]
skipping: [worker-2]
skipping: [worker-20]
skipping: [worker-21]
skipping: [worker-22]

.....

PLAY [Wait for ready and ensure uncordoned] *****

TASK [Cordon and drain node] *****

Monday 03 January 2022 06:04:15 +0000 (0:00:01.116) 0:00:08.217 *****

skipping: [master-1]
skipping: [master-2]
skipping: [master-3]
skipping: [worker-11]
skipping: [worker-10]
skipping: [worker-12]
skipping: [worker-13]
skipping: [worker-1]
skipping: [worker-15]
skipping: [worker-16]
skipping: [worker-17]
skipping: [worker-18]
skipping: [worker-19]
skipping: [worker-2]
skipping: [worker-20]
skipping: [worker-21]
skipping: [worker-22]

.....

TASK [upgrade/cordon : Cordon/Drain/Delete node] *****

Monday 03 January 2022 06:04:16 +0000 (0:00:01.430) 0:00:09.647 *****

changed: [worker-11 -> 10.192.x.x]

PLAY RECAP *****

master-1		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
master-2		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
master-3		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-11		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-10		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-11		: ok=2	changed=1	unreachable=0	failed=0	skipped=1
rescued=0	ignored=0					
worker-12		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-13		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-1		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-15		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-16		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-17		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-18		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-19		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-2		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-20		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-21		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					
worker-22		: ok=1	changed=0	unreachable=0	failed=0	skipped=2
rescued=0	ignored=0					

.....

Monday 03 January 2022 06:04:17 +0000 (0:00:01.168) 0:00:10.815 *****

=====

2022-01-03 06:04:17.957 DEBUG cluster_sync.cluster-name.worker-11: **Cluster sync successful**

2022-01-03 06:04:17.958 DEBUG cluster_sync.cluster-name.worker-11: **Ansible sync done**

2022-01-03 06:04:17.961 INFO cluster_sync.cluster-name.worker-11: **_sync finished. Opening lock**

8. Verifique el nodo maestro de kubernetes y asegúrese de que el estado del nodo de trabajo ha cambiado.

Ejemplo de resultado

```
cloud-user@cluster-name-master-1:~$ kubectl get nodes
```

NAME	STATUS	ROLES	AGE	VERSION
cluster-name-master-1	Ready	control-plane,master	213d	v1.21.0
cluster-name-master-2	Ready	control-plane,master	213d	v1.21.0
cluster-name-master-3	Ready	control-plane,master	213d	v1.21.0
cluster-name-worker-11	Ready	<none>	213d	v1.21.0
cluster-name-worker-10	Ready	<none>	213d	v1.21.0
cluster-name-worker-11	Ready, SchedulingDisabled			
cluster-name-worker-12	Ready	<none>	213d	v1.21.0
cluster-name-worker-13	Ready	<none>	213d	v1.21.0
cluster-name-worker-11	Ready	<none>	213d	v1.21.0

9. En este paso, el nodo debe estar preparado para el mantenimiento (todos los POD de la aplicación deben haber sido desalojados, excepto los grupos de dispositivos administrados por daemonset/replicaset, etc., que se pueden ignorar).

10. Cierre el servidor desde Cisco Integrated Management Console (CIMC) o desde cualquier consola de gestión equivalente si el servidor pertenece a un proveedor diferente, y realice el mantenimiento del hardware.

Cuando el servidor vuelva a estar en línea después del mantenimiento y cuando todas las comprobaciones de estado estén en verde, realice esto.

11. Establezca el nodo de trabajo en Mantenimiento = "Falso" para volver a agregarlo y ejecutar una sincronización.

Ejemplo de resultado


```
[cluster-name-cm-1] SMI Cluster Deployer# config
Entering configuration mode terminal
[cluster-name-cm-1] SMI Cluster Deployer(config)# clusters cluster-name
[cluster-name-cm-1] SMI Cluster Deployer(config-clusters-cluster-name)# nodes worker-11
[cluster-name-cm-1] SMI Cluster Deployer(config-nodes-worker1)# maintenance false
[cluster-name-cm-1] SMI Cluster Deployer(config-nodes-worker1)# commit
Commit complete.
[cluster-name-cm-1] SMI Cluster Deployer(config-nodes-worker1)# end
```

12. Ejecute la sincronización del clúster para devolver el nodo en rotación y listo para servir.

Salida de muestra (truncada):

```
[cluster-name-cm-1] SMI Cluster Deployer# clusters cluster-name nodes worker-11 actions sync run
debug true
This will run sync. Are you sure? [no,yes] yes
message accepted
```

```
PLAY [Wait for ready and ensure uncordoned] *****
```

```
TASK [Wait for ready and ensure uncordoned] *****
```

```
Monday 03 January 2022 07:12:35 +0000 (0:00:01.151) 0:09:42.974 *****
```

```
skipping: [master-1] => (item=upgrade/wait-for-cluster-ready)
```

```
skipping: [master-1] => (item=upgrade/uncordon)
```

```
skipping: [master-2] => (item=upgrade/wait-for-cluster-ready)
```

```
skipping: [master-2] => (item=upgrade/uncordon)
```

```
skipping: [master-3] => (item=upgrade/wait-for-cluster-ready)
```

```
skipping: [master-3] => (item=upgrade/uncordon)
```

```
skipping: [worker-11] => (item=upgrade/wait-for-cluster-ready)
```

```
skipping: [worker-11] => (item=upgrade/uncordon)
```

```
skipping: [worker-10] => (item=upgrade/wait-for-cluster-ready)
```

```
skipping: [worker-10] => (item=upgrade/uncordon)
```

```
skipping: [worker-12] => (item=upgrade/wait-for-cluster-ready)
```

```
skipping: [worker-12] => (item=upgrade/uncordon)
```

```
skipping: [worker-13] => (item=upgrade/wait-for-cluster-ready)
```

```
skipping: [worker-13] => (item=upgrade/uncordon)
```

```
skipping: [worker-1] => (item=upgrade/wait-for-cluster-ready)
```

```
skipping: [worker-1] => (item=upgrade/uncordon)
```

```
.....
```

```
skipping: [worker-3] => (item=upgrade/wait-for-cluster-ready)
```

```
skipping: [worker-3] => (item=upgrade/uncordon)
```

```
skipping: [worker-4] => (item=upgrade/wait-for-cluster-ready)
```

skipping: [worker-4] => (item=upgrade/uncordon)
skipping: [worker-5] => (item=upgrade/wait-for-cluster-ready)
skipping: [worker-5] => (item=upgrade/uncordon)
skipping: [worker-6] => (item=upgrade/wait-for-cluster-ready)
skipping: [worker-6] => (item=upgrade/uncordon)
skipping: [worker-7] => (item=upgrade/wait-for-cluster-ready)
skipping: [worker-7] => (item=upgrade/uncordon)
skipping: [worker-8] => (item=upgrade/wait-for-cluster-ready)
skipping: [worker-8] => (item=upgrade/uncordon)
skipping: [worker-9] => (item=upgrade/wait-for-cluster-ready)
skipping: [worker-9] => (item=upgrade/uncordon)

TASK [upgrade/uncordon : Restore cordoned node] *****

Monday 03 January 2022 07:12:37 +0000 (0:00:01.539) 0:09:44.513 *****

changed: [worker-11 -> 10.192.x.x]

PLAY RECAP *****

master-1		: ok=38	changed=4	unreachable=0	failed=0	skipped=73
rescued=0	ignored=0					
master-2		: ok=35	changed=3	unreachable=0	failed=0	skipped=73
rescued=0	ignored=0					
master-3		: ok=35	changed=3	unreachable=0	failed=0	skipped=73
rescued=0	ignored=0					
worker-1		: ok=64	changed=3	unreachable=0	failed=0	skipped=83
rescued=0	ignored=0					
worker-10		: ok=63	changed=3	unreachable=0	failed=0	skipped=83
rescued=0	ignored=0					
worker-11		: ok=218	changed=30	unreachable=0	failed=0	skipped=306
rescued=0	ignored=0					
worker-12		: ok=63	changed=3	unreachable=0	failed=0	skipped=83
rescued=0	ignored=0					
worker-13		: ok=63	changed=3	unreachable=0	failed=0	skipped=83
rescued=0	ignored=0					
worker-11		: ok=63	changed=3	unreachable=0	failed=0	skipped=83
rescued=0	ignored=0					
.....						
worker-3		: ok=63	changed=3	unreachable=0	failed=0	skipped=83
rescued=0	ignored=0					

```
worker-4      : ok=63  changed=3  unreachable=0  failed=0  skipped=83
rescued=0    ignored=0

worker-5      : ok=63  changed=3  unreachable=0  failed=0  skipped=83
rescued=0    ignored=0

worker-6      : ok=63  changed=3  unreachable=0  failed=0  skipped=83
rescued=0    ignored=0

worker-7      : ok=63  changed=3  unreachable=0  failed=0  skipped=83
rescued=0    ignored=0

worker-8      : ok=63  changed=3  unreachable=0  failed=0  skipped=83
rescued=0    ignored=0

worker-9      : ok=63  changed=3  unreachable=0  failed=0  skipped=83
rescued=0    ignored=0
```

```
Monday 03 January 2022  07:12:38 +0000 (0:00:00.967)          0:09:45.481 *****
```

```
=====
```

```
2022-01-03 07:12:38.854 DEBUG cluster_sync.cluster-name.worker-11: Cluster sync successful
```

```
2022-01-03 07:12:38.858 DEBUG cluster_sync.cluster-name.worker-11: Ansible sync done
```

```
2022-01-03 07:12:38.860 INFO cluster_sync.cluster-name.worker-11: _sync finished. Opening lock  
13. Verifique el estado del clúster. El recuento deseado de grupos de dispositivos debe coincidir  
con el recuento preparado.
```

```
[cluster-name-cm-1] SMI Cluster Deployer# clusters cluster-name actions k8s cluster-status  
pods-desired-count 678  
pods-ready-count 678  
pods-desired-are-ready true  
etcd-healthy true  
all-ok true
```