排除5G SMI CNDP集群中的硬件问题并执行维护

目录

[简介](#)
[先决条件](#)
[要求](#)
[使用的组件](#)
[背景信息](#)
[什么是SMI?](#)
[什么是SMI-BM或CNDP?](#)
[什么是SMI Cluster Manager?](#)
[什么是SMI集群部署器？](#)
[问题](#)
[维护程序](#)

简介

本文档介绍在5G用户微服务基础设施(SMI)云本地部署平台(CNDP)设备池(POD)中执行维护(硬件更换或维护)的过程。

先决条件

要求

Cisco 建议您了解以下主题：

- 思科SMI
- 5G CNDPA或SMI裸机(BM)架构
- 多克斯和库伯内特
- Cisco UCS C220系列服务器

使用的组件

本文档中的信息基于以下软件和硬件版本：

- SMI 2020.02.2.35
- 库贝内特斯v1.21.0
- 思科UCS C220-M5SX-CM

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

背景信息

什么是SMI?

思科SMI是云技术和标准的分层堆栈，支持来自思科移动、电缆和BNG业务单元的基于微服务的应用 — 所有这些业务单元都具有相似的用户管理功能和类似的datastore要求。

Attributes:

- 层云堆栈（技术和标准），可提供自上而下的部署，并适应当前客户云基础设施。
- 通用执行环境由所有应用程序共享，用于非应用程序功能（数据存储、部署、配置、遥测、警报）。 这为所有客户触点和集成点提供一致的交互和体验。
- 应用和常见执行环境部署在微服务容器中，并与智能服务网状网连接。
- 用于部署、配置和管理的外露API，可实现自动化。

什么是SMI-BM或CNDP?

Cisco SMI — 裸机或CNDP是受管的裸机平台，它提供部署虚拟网络功能(VNF)和云本地功能(CNF)的基础设施，支持思科移动、电缆和BNG业务单元。

Attributes:

- 消除VIM相关开销的裸机
- 性能改善 更多应用核心加快应用执行
- 自动化部署工作流程；与NSO CFP集成
- 用于部署思科5G NF的托管堆栈
- 简化的订单和部署指南

什么是SMI Cluster Manager?

群集管理器是一个2节点保持连接的群集，用作控制平面和用户平面群集部署的初始点。它运行单个节点kubernetes群集和一组POD，负责整个群集设置。只有主群集管理器处于活动状态，辅助管理器仅在发生故障或手动关闭以进行维护时接管。

什么是SMI集群部署器？

SMI Deployer是群集管理器中的一项服务，可以创建VM、自定义主机操作系统、创建K8s群集、启动K8s主服务器、配置群集和启动应用程序等。

问题

硬件维护（如硬件故障或软件/防火墙升级等）需要服务器停机。要在POD中执行维护，需要遵循什么程序。如何平稳地停止服务以避免应用程序中出现不必要的停机。

维护程序

获取集群管理器VIP、Kubernetes master VIP（针对各个应用）、UCS CIMC IP、UCS CIMC名称以及要执行维护的服务器主机名（OS主机名）。

登录到kubernetes主设备与服务对应，并确保所有POD都处于**运行**状态。

示例输出：

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

2.登录集群管理器并访问SMI集群部署器运营中心（以下是查找运营中心IP的步骤）。

```
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)
```

示例输出：

```
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.使用此命令登录。

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

4.使用show clusters命令检查服务与应用程序对应。

示例输出：

```
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.使用这些命令排空您执行维护的节点并键入Yes（这将正常撤离POD，并根据需要在其他节点中重新启动）。

示例输出：

```
[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.使用这些命令将节点移至维护模式（最多可能需要30分钟）。

示例输出：

```
[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.检查日志的状态。

clusters cluster-name nodes worker-11 actions sync logs
(In this we are dealing with the worker-11 node)

输出示例（截断）：

```
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.检查kubernetes主节点并确保工作节点的状态已更改。

示例输出：

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.在此步骤中，节点应准备好进行维护（除可忽略的由守护进程/复制集等管理的Pod外，所有应用POD必须已被逐出）。

10.如果服务器属于其他供应商，则从思科集成管理控制台(CIMC)或任何等效管理控制台关闭服务器，并执行硬件维护。

当服务器在维护后恢复联机，并且所有运行状况检查均为绿色时，请执行此操作。

11.将Worker-Node设置为Maintenance = "False"，以便重新添加并运行同步。

示例输出：

```
[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.运行群集同步，使节点在旋转状态下恢复并准备提供服务。

输出示例（截断）：

```
[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.检查群集的状态。Pod-desired-count应与ready-count匹配。

```
[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
```