

5G SMI CNDPクラスタのハードウェア問題のトラブルシューティングとメンテナンスの実行

内容

[概要](#)

[前提条件](#)

[要件](#)

[使用するコンポーネント](#)

[背景説明](#)

[SMIとは](#)

[SMI-BMまたはCNDPとは何ですか。](#)

[SMI Cluster Managerとは何ですか。](#)

[SMI Cluster Deployerとは何ですか。](#)

[問題](#)

[メンテナンスの手順](#)

概要

このドキュメントでは、5G Subscriber Microservices Infrastructure(SMI)Cloud Native Deployment Platform(CNDP)Pool of Devices(POD)におけるファームウェア(FW)のアップグレードなどのメンテナンス (ハードウェア交換またはメンテナンス) を実行する手順について説明します。

前提条件

要件

次の項目に関する知識があることが推奨されます。

- Cisco SMI
- 5G CNDPAまたはSMI-Bare-metal(BM)アーキテクチャ
- ドッカーとクベルネテス
- Cisco UCS C220シリーズサーバ

使用するコンポーネント

このドキュメントの情報は、次のソフトウェアとハードウェアのバージョンに基づいています。

- SMI 2020.02.2.35
- クベルネテスv1.21.0
- Cisco UCS C220-M5SX-CM

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このド

キュメントで使用するすべてのデバイスは、初期（デフォルト）設定の状態から起動しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

背景説明

SMIとは

Cisco SMIは、Cisco Mobility、Cable、およびBNGビジネスユニットのマイクロサービスベースのアプリケーションを可能にするクラウドテクノロジーと標準の階層型スタックであり、これらはすべて同様の加入者管理機能とデータストア要件を持ちます。

属性:

- 上位から下位に展開し、現在の顧客のクラウドインフラストラクチャにも対応するレイヤクラウドスタック（テクノロジーおよび標準）。
- アプリケーション以外の機能（データストレージ、導入、設定、テレメトリ、アラーム）のすべてのアプリケーションで共有されるCommon Execution Environment。これにより、すべての顧客のタッチポイントと統合ポイントに一貫したインタラクションとエクスペリエンスが提供されます。
- アプリケーションとCommon Execution Environmentは、マイクロサービスコンテナに導入され、インテリジェントサービスメッシュに接続されます。
- 展開、設定、および管理のための公開されたAPIにより、自動化が可能になります。

SMI-BMまたはCNDPとは何ですか。

Cisco SMI-Bare MetalまたはCNDPは、仮想ネットワーク機能(VNF)およびクラウドネイティブ機能(CNF)を導入し、Cisco Mobility、ケーブル、およびBNGビジネスユニットを有効にするインフラストラクチャを提供する、専用のベアメタルプラットフォームです。

属性:

- VIM関連のオーバーヘッドを排除するベアメタル
- パフォーマンスの向上 アプリケーションのコア数の増加アプリケーション実行の高速化
- 自動導入ワークフローNSO CFPと統合
- Cisco 5G NFを導入するためのキュレーテッドスタック
- 発注および導入ガイドの簡素化

SMI Cluster Managerとは何ですか。

クラスタマネージャは、コントロールプレーンとユーザプレーンのクラスタ展開の両方の初期ポイントとして使用される2ノードキープアライブクラスタです。単一ノードのKubernetesクラスタと、クラスタ全体のセットアップを担当する一連のPODを実行します。プライマリクラスタマネージャのみがアクティブで、セカンダリが引き継ぐのは、障害が発生した場合が、メンテナンスのために手動でダウンした場合だけです。

SMI Cluster Deployerとは何ですか。

SMI Deployerは、VMの作成、ホストOSのカスタマイズ、K8sクラスタの作成、K8s Masterの起

動、クラスタの設定、アプリケーションの起動などを可能にするCluster Managerのサービスです。

問題

ハードウェア障害やソフトウェア/ファームウェアのアップグレードなどのハードウェアメンテナンスには、サーバのダウンタイムが必要です。PODでメンテナンスを実行するには、どのような手順に従う必要があります。アプリケーションで不要なダウンタイムを回避するために、サービスを正常に停止する方法。

メンテナンスの手順

メンテナンスを実行するクラスタマネージャVIP、Kubernetes master VIP (各アプリケーション)、UCS CIMC IP、UCS CIMC名、サーバホスト名 (OSホスト名) を取得します。

kubernetes masterへのログインはサービスに対応し、すべてのPODがRunning状態であることを確認してください。

出力例 :

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

2.クラスタマネージャにログインし、SMIクラスタ導入オペレーションセンターにアクセスします (ここではオペレーションセンターIPを見つける手順を示します)。

```
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)
```

出力例 :

```
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.このコマンドでログインします。

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

4. **show clusters**コマンドを使用して、サービスがアプリケーションに対応していることを確認します。

出力例 :

```
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.これらのコマンドを使用してメンテナンスを実行するノードをドレインし、「Yes」と入力します (これにより、PODが正常に退避し、必要に応じて他のノードで再起動します)。

出力例：

```
[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.これらのコマンドを使用して、ノードをメンテナンスモードに移動します (これには最大30分かかります)。

出力例：

```
[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.ログでステータスを確認します。

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

出力例 (省略)：

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

.....

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. kubernetesマスターノードを確認し、ワーカーノードのステータスが変更されたことを確認します。

出力例：

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.このステップでは、ノードのメンテナンスの準備が整っている必要があります
(daemon/replicasetなどで管理されるポッドを除き、すべてのアプリケーションPODが削除されている必要があります。これは無視できます)。

10.サーバが別のベンダーに属している場合は、Cisco Integrated Management Console(CIMC)または同等の管理コンソールからサーバをシャットダウンし、ハードウェアメンテナンスを実行します。

メンテナンス後にサーバがオンラインに戻り、すべてのヘルスチェックが緑色になったら、これを実行します。

11. Worker-NodeをMaintenance = "False"に設定して再度追加し、同期を実行します。

出力例：

[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. クラスターの同期を実行して、ノードをローテーション状態に戻し、サービスの準備を整えます。

出力例 (省略) :

```

[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 rescued=0	ignored=0	: ok=63	changed=3	unreachable=0	failed=0	skipped=83
worker-5 rescued=0	ignored=0	: ok=63	changed=3	unreachable=0	failed=0	skipped=83
worker-6 rescued=0	ignored=0	: ok=63	changed=3	unreachable=0	failed=0	skipped=83
worker-7 rescued=0	ignored=0	: ok=63	changed=3	unreachable=0	failed=0	skipped=83
worker-8 rescued=0	ignored=0	: ok=63	changed=3	unreachable=0	failed=0	skipped=83
worker-9 rescued=0	ignored=0	: ok=63	changed=3	unreachable=0	failed=0	skipped=83

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.** クラスタのステータスを確認します。Pods-desired-countはready-countと一致する必要があります。

[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**