

Etapas para RMA do servidor AIO baseado em RCM na implantação do CNDP

Contents

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Conhecer o Esquema IP do RCM](#)

[Procedimento de backup](#)

[Fazer backup da configuração](#)

[Procedimento de pré-verificação](#)

[Pré-verificações da OIA](#)

[Exemplo de saída de pré-verificações](#)

[Procedimento de execução](#)

[Etapas para Executar no RCM Antes de Desligar o Nó AIO](#)

[Etapas para Executar no Nó Kubernetes Antes de Desligar o Nó AIO](#)

[Procedimento de manutenção do servidor](#)

[Procedimento de restauração do Kubernetes](#)

[Etapas para executar no nó Kubernetes pós-alimentação no nó AIO](#)

[Procedimento de restauração do RCM](#)

[Etapas para executar nos centros de operações CEE e RCM para restaurar o aplicativo](#)

[Procedimento de verificação](#)

Introdução

Este documento descreve o procedimento detalhado de RMA (Return Material Authorization, Autorização de Devolução de Material) para o servidor All-in-One (AIO) baseado em RCM (Redundancy Configuration Manager) na implantação da CNDP (Cloud Native Deployment Platform, plataforma de implantação nativa de nuvem) para qualquer problema de hardware ou atividades relacionadas à manutenção.

Pré-requisitos

Requisitos

A Cisco recomenda que você tenha conhecimento destes tópicos:

- RCM
- Kubernetes

Componentes Utilizados

As informações neste documento são baseadas na versão RCM - rcm.2021.02.1.i18

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

Conhecer o Esquema IP do RCM

Este documento explica o projeto RCM que consiste em dois nós AIO com dois Opscenters RCM e um RCM CEE, um para cada nó AIO.

O nó RCM AIO de destino para a RMA neste artigo é AIO-1 (AI0301) que contém os centros de acesso RCM no estado PRIMÁRIO.

POD_NAME	NODE_NAME	IP_ADDRESS	DEVICE_TYPE	OS_TYPE
UP0300	RCE301	10.1.2.9	RCM_CEE_AIO_1	opscenter
UP0300	RCE302	10.1.2.10	RCM_CEE_AIO_2	opscenter
UP0300	AI0301	10.1.2.7	RCM_K8_AIO_1	linux
UP0300	AI0302	10.1.2.8	RCM_K8_AIO_2	linux
UP0300	RM0301	10.1.2.3	RCM1_ACTIVE	opscenter
UP0300	RM0302	10.1.2.4	RCM1_STANDBY	opscenter

UP0300	RM0303	10.1.2.5	RCM2_ACTIVE	opscenter
UP0300	RM0304	10.1.2.6	RCM2_STANDBY	opscenter

Procedimento de backup

Fazer backup da configuração

Para começar, colete o backup de configuração da configuração atual dos centros de acesso RCM que são executados no nó AIO de destino.

```
# show running-config | nomore
```

Colete a configuração atual dos centros de postagem RCM CEE que são executados no nó AIO de destino.

```
# show running-config | nomore
```

Procedimento de pré-verificação

Pré-verificações da OIA

Colete a saída do comando de ambos os nós AIO e verifique se todos os pods estão no estado Running (Em execução).

```
# kubectl get ns
# kubectl get pods -A -o wide
```

Exemplo de saída de pré-verificações

Observe que os dois opscenters RCM e um opscenter RCM CEE são executados no nó AIO-1

```
c1oud-user@up0300-aio-1-master-1:~$ kubectl get ns
NAME                STATUS  AGE
```

```
cee-rce301      Active  110d  <--
default        Active  110d
istio-system   Active  110d
kube-node-lease Active  110d
kube-public    Active  110d
kube-system    Active  110d
nginx-ingress  Active  110d
rcm-rm0301     Active  110d  <--
rcm-rm0303     Active  110d  <--
registry       Active  110d
smi-certs      Active  110d
smi-node-label Active  110d
smi-vips       Active  110d
cloud-user@up0300-ai0-1-master-1:~$
```

Faça login no RCM opscenter da AIO-1 e verifique o status.

```
[up0300-ai0-1/rm0301] rcm# rcm show-status
message :
{"status":[" Fri Oct 29 07:21:11 UTC 2021 : State is MASTER"]}
[up0300-ai0-1/rm0301] rcm#
```

```
[up0300-ai0-1/rm0303] rcm# rcm show-status
message :
{"status":[" Fri Oct 29 07:22:18 UTC 2021 : State is MASTER"]}
[up0300-ai0-1/rm0303] rcm#
```

Repita as mesmas etapas no nó AIO-2 em que os outros dois centros de postagem RCM correspondem ao nó AIO-1.

```
cloud-user@up0300-ai0-2-master-1:~$ kubectl get ns
NAME          STATUS  AGE
cee-rce302    Active  105d  <--
default       Active  105d
istio-system  Active  105d
kube-node-lease Active  105d
kube-public   Active  105d
kube-system   Active  105d
nginx-ingress Active  105d
rcm-rm0302    Active  105d  <--
rcm-rm0304    Active  105d  <--
registry      Active  105d
smi-certs     Active  105d
smi-node-label Active  105d
smi-vips      Active  105d
cloud-user@up0300-ai0-2-master-1:~$
```

Faça login no RCM opscenter da AIO-2 e verifique o status.

```
[up0300-aio-2/rm0302] rcm# rcm show-status
message :
{"status":[" Fri Oct 29 09:32:54 UTC 2021 : State is BACKUP"]}
[up0300-aio-2/rm0302] rcm#
```

```
[up0300-aio-2/rm0304] rcm# rcm show-status
message :
{"status":[" Fri Oct 29 09:33:51 UTC 2021 : State is BACKUP"]}
[up0300-aio-2/rm0304] rcm#
```

Procedimento de execução

Etapas para Executar no RCM Antes de Desligar o Nó AIO

1. Como ambos os RCMs em AIO-1 são MASTER, você pode migrá-los para BACKUP.

a. Para fazer isso, você deve executar o comando `rcm migrate primary` nos RCMs ativos antes de desligar o servidor AIO-1.

```
[up0300-aio-1/rm0301] rcm# rcm migrate primary
```

```
[up0300-aio-1/rm0303] rcm# rcm migrate primary
```

b. Verifique se o status agora é BACKUP em AIO-1.

```
[up0300-aio-1/rm0301] rcm# rcm show-status
```

```
[up0300-aio-1/rm0303] rcm# rcm show-status
```

c. Verifique se o status agora é MASTER em AIO-2 e se eles são MASTER.

```
[up0300-aio-1/rm0302] rcm# rcm show-status
```

```
[up0300-aio-1/rm0304] rcm# rcm show-status
```

d. Execute o desligamento do RCM no rm0301 e no rm0303.

```
[up0300-aio-2/rm0301] rcm# config
Entering configuration mode terminal
[up0300-aio-2/rm0301] rcm(config)# system mode shutdown
[up0300-aio-1/rce301] rcm(config)# commit comment <CRNUMBER>
```

```
[up0300-aio-2/rm0303] rcm# config
Entering configuration mode terminal
[up0300-aio-2/rm0303] rcm(config)# system mode shutdown
[up0300-aio-1/rce303] rcm(config)# commit comment <CRNUMBER>
```

2. Também precisamos desativar os CEE ops que são executados nos comandos AIO-1 usados.

```
[up0300-aio-1/rce301] cee# config
Entering configuration mode terminal
[up0300-aio-1/rce301] cee(config)# system mode shutdown
[up0300-aio-1/rce301] cee(config)# commit comment <CRNUMBER>
[up0300-aio-1/rce301] cee(config)# exit
```

Aguarde alguns minutos e verifique o sistema para mostrar 0,0%.

```
[up0300-aio-1/rce301] cee# show system
```

3. Verifique se não há pods para namespaces RCM e CEE, exceto para pods de documentação, smart-agent, ops-center-rcm e ops-center-cee

```
# kubectl get pods -n rcm-rm0301 -o wide
# kubectl get pods -n rcm-rm0303 -o wide
# kubectl get pods -n cee-rce302 -o wide
```

Etapas para Executar no Nó Kubernetes Antes de Desligar o Nó AIO

Desenhe o nó Kubernetes para que os pods e serviços associados sejam terminados normalmente. O agendador não selecionaria mais este nó do Kubernetes e removeria pods desse nó. Esvazie um único nó por vez.

Faça login no Gerenciador de cluster SMI.

```
cloud-user@bot-deployer-cm-primary:~$ kubectl get svc -n smi-cm
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)
cluster-files-offline-smi-cluster-deployer	ClusterIP	10.102.108.177	<none>	8080/TCP
iso-host-cluster-files-smi-cluster-deployer	ClusterIP	10.102.255.174	192.168.0.102	80/TCP
iso-host-ops-center-smi-cluster-deployer	ClusterIP	10.102.58.99	192.168.0.100	3001/TCP
netconf-ops-center-smi-cluster-deployer	ClusterIP	10.102.108.194	10.244.110.193	3022/TCP, 22
ops-center-smi-cluster-deployer	ClusterIP	10.102.156.123	<none>	8008/TCP, 20
squid-proxy-node-port	NodePort	10.102.73.130	<none>	3128:31677/

```
cloud-user@bot-deployer-cm-primary:~$ ssh -p 2024 admin@<Cluster IP of ops-center-smi-cluster-deployer>
```

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

```
admin connected from 192.168.0.100 using ssh on ops-center-smi-cluster-deployer-686b66d9cd-nfzx8
```

```
[bot-deployer-cm-primary] SMI Cluster Deployer#
```

```
[bot-deployer-cm-primary] SMI Cluster Deployer# show clusters
```

```
                LOCK TO
NAME            VERSION
-----
cp0100-smf-data -
cp0100-smf-ims  -
cp0200-smf-data -
cp0200-smf-ims  -
up0300-aio-1    -    <--
up0300-aio-2    -
up0300-upf-data -
up0300-upf-ims  -
```

Desenhe o nó mestre:

```
[bot-deployer-cm-primary] SMI Cluster Deployer# clusters up0300-aio-1 nodes master-1 actions sync drain
This would run drain on the node, disrupting pods running on the node. Are you sure? [no,yes] yes
message accepted
```

Marque o nó master-1 no modo de manutenção:

```
[bot-deployer-cm-primary] SMI Cluster Deployer# config
Entering configuration mode terminal
[bot-deployer-cm-primary] SMI Cluster Deployer(config)# clusters up0300-aio-1
[bot-deployer-cm-primary] SMI Cluster Deployer(config-clusters-up0300-aio-1)# nodes master-1
[bot-deployer-cm-primary] SMI Cluster Deployer(config-nodes-master1)# maintenance true
[bot-deployer-cm-primary] SMI Cluster Deployer(config-nodes-master1)# commit
Commit complete.
[bot-deployer-cm-primary] SMI Cluster Deployer(config-nodes-master1)# end
```

Execute a sincronização do Cluster e monitore os logs para a ação de sincronização:

```
[bot-deployer-cm-primary] SMI Cluster Deployer# clusters up0300-aio-1 nodes master-1 actions sync
This would run sync. Are you sure? [no,yes] yes
message accepted
[bot-deployer-cm-primary] SMI Cluster Deployer# clusters up0300-aio-1 nodes master-1 actions sync logs
```

Exemplo de saída para logs de sincronização de cluster:

```

[installer-master] SMI Cluster Deployer# clusters kali-stacked nodes cmts-worker1-1 actions sync logs
Example Cluster Name: kali-stacked
Example WorkerNode: cmts-worker1
Logs 2020-10-06 20:01:48.023 DEBUG cluster_sync.kali-stacked.cmts-worker1: Cluster name: kali-stacked
2020-10-06 20:01:48.024 DEBUG cluster_sync.kali-stacked.cmts-worker1: Node name: cmts-worker1
2020-10-06 20:01:48.024 DEBUG cluster_sync.kali-stacked.cmts-worker1: debug: false
2020-10-06 20:01:48.024 DEBUG cluster_sync.kali-stacked.cmts-worker1: remove_node: true
PLAY [Check required variables] *****
TASK [Gathering Facts] *****
Tuesday 06 October 2020  20:01:48 +0000 (0:00:00.017)          0:00:00.017 *****
ok: [master3]
ok: [master1]
ok: [cmts-worker1]
ok: [cmts-worker3]
ok: [cmts-worker2]
ok: [master2]
TASK [Check node_name] *****
Tuesday 06 October 2020  20:01:50 +0000 (0:00:02.432)          0:00:02.450 *****
skipping: [master1]
skipping: [master2]
skipping: [master3]
skipping: [cmts-worker1]
skipping: [cmts-worker2]
skipping: [cmts-worker3]
PLAY [Wait for ready and ensure uncordoned] *****
TASK [Cordon and drain node] *****
Tuesday 06 October 2020  20:01:51 +0000 (0:00:00.144)          0:00:02.594 *****
skipping: [master1]
skipping: [master2]
skipping: [master3]
skipping: [cmts-worker2]
skipping: [cmts-worker3]
TASK [upgrade/cordon : Cordon/Drain/Delete node] *****
Tuesday 06 October 2020  20:01:51 +0000 (0:00:00.205)          0:00:02.800 *****
changed: [cmts-worker1 -> 172.22.18.107]
PLAY RECAP *****
cmts-worker1          : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0
cmts-worker2          : ok=1    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
cmts-worker3          : ok=1    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
master1               : ok=1    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
master2               : ok=1    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
master3               : ok=1    changed=0    unreachable=0    failed=0    skipped=2    rescued=0
Tuesday 06 October 2020  20:02:29 +0000 (0:00:38.679)          0:00:41.479 *****
=====
2020-10-06 20:02:30.057 DEBUG cluster_sync.kali-stacked.cmts-worker1: Cluster sync successful
2020-10-06 20:02:30.058 DEBUG cluster_sync.kali-stacked.cmts-worker1: Ansible sync done
2020-10-06 0:02:30.058 INFO cluster_sync.kali-stacked.cmts-worker1: _sync finished. Opening lock

```

Procedimento de manutenção do servidor

Desligue o servidor do CIMC normalmente. Prossiga com a atividade de manutenção relacionada ao hardware, conforme definido no MoP Hardware, e certifique-se de que todas as verificações de integridade sejam aprovadas depois que o servidor for ligado.

Observação: este artigo não cobre o MoP da atividade de manutenção ou hardware do servidor, pois eles são diferentes da descrição do problema

Procedimento de restauração do Kubernetes

Etapas para executar no nó Kubernetes pós-alimentação no nó AIO

Faça login no Gerenciador de cluster SMI:

```
cloud-user@bot-deployer-cm-primary:~$ kubectl get svc -n smi-cm
NAME                                     TYPE          CLUSTER-IP      EXTERNAL-IP      PORT(S)
cluster-files-offline-smi-cluster-deployer ClusterIP      10.102.108.177  <none>           8080/TCP
iso-host-cluster-files-smi-cluster-deployer ClusterIP      10.102.255.174  192.168.0.102   80/TCP
iso-host-ops-center-smi-cluster-deployer  ClusterIP      10.102.58.99    192.168.0.100   3001/TCP
netconf-ops-center-smi-cluster-deployer  ClusterIP      10.102.108.194  10.244.110.193  3022/TCP,22
ops-center-smi-cluster-deployer          ClusterIP      10.102.156.123  <none>           8008/TCP,20
squid-proxy-node-port                     NodePort       10.102.73.130   <none>           3128:31677/

cloud-user@bot-deployer-cm-primary:~$ ssh -p 2024 admin@<ClusterIP of ops-center-smi-cluster-deployer>
Welcome to the Cisco SMI Cluster Deployer on bot-deployer-cm-primary
Copyright © 2016-2020, Cisco Systems, Inc.
All rights reserved.
admin connected from 192.168.0.100 using ssh on ops-center-smi-cluster-deployer-686b66d9cd-nfzx8
[bot-deployer-cm-primary] SMI Cluster Deployer#
[bot-deployer-cm-primary] SMI Cluster Deployer# show clusters
                LOCK TO
NAME            VERSION
-----
cp0100-smf-data -
cp0100-smf-ims  -
cp0200-smf-data -
cp0200-smf-ims  -
up0300-aio-1    -    <--
up0300-aio-2    -
up0300-upf-data -
up0300-upf-ims  -
```

Desative o sinalizador de manutenção para que o master-1 seja adicionado de volta ao cluster.

```
[bot-deployer-cm-primary] SMI Cluster Deployer# config
Entering configuration mode terminal
[bot-deployer-cm-primary] SMI Cluster Deployer(config)# clusters up0300-aio-1
[bot-deployer-cm-primary] SMI Cluster Deployer(config-clusters-up0300-aio-1)# nodes master-1
[bot-deployer-cm-primary] SMI Cluster Deployer(config-nodes-master-1)# maintenance false
[bot-deployer-cm-primary] SMI Cluster Deployer(config-nodes-master-1)# commit
Commit complete.
[bot-deployer-cm-primary] SMI Cluster Deployer(config-nodes-master-1)# end
```

Restaure os pods e serviços do nó principal com a ação de sincronização do cluster.

```
[bot-deployer-cm-primary] SMI Cluster Deployer# clusters up0100-aio-1 nodes master-1 actions sync run
This would run sync. Are you sure? [no,yes] yes
```

message accepted

Monitore os logs para a ação de sincronização.

```
[bot-deployer-cm-primary] SMI Cluster Deployer# clusters up0100-aio-1 nodes master-1 actions sync logs
```

Verifique o status do cluster do mestre AIO-1.

```
[bot-deployer-cm-primary] SMI Cluster Deployer# clusters up0300-aio-1 actions k8s cluster-status
```

Saída de exemplo:

```
[installer-] SMI Cluster Deployer# clusters kali-stacked actions k8s cluster-status  
pods-desired-count 67  
pods-ready-count 67  
pods-desired-are-ready true  
etcd-healthy true  
all-ok true
```

Procedimento de restauração do RCM

Etapas para executar nos centros de operações CEE e RCM para restaurar o aplicativo

Atualize o opscenter CEE e o opscenter RCM no modo de execução.

Configure o modo de execução para rce301.

```
[up0300-aio-1/rce301] cee# config  
Entering configuration mode terminal  
[up0300-aio-1/rce301] cee(config)# system mode running  
[up0300-aio-1/rce301] cee(config)# commit comment <CRNUMBER>  
[up0300-aio-1/rce301] cee(config)# exit
```

Aguarde alguns minutos e verifique se o sistema está em 100,0%.

```
[up0300-aio-1/rce301] cee# show system
```

Configure o modo running para rm0301.

```
[up0300-aio-2/rm0301] rcm# config
Entering configuration mode terminal
[up0300-aio-2/rm0301] rcm(config)# system mode running
[up0300-aio-1/rce301] rcm(config)# commit comment <CRNUMBER>
```

Aguarde alguns minutos e verifique se o sistema está em 100,0%.

```
[up0300-aio-1/rm0301] cee# show system
```

Configure o modo running para rm0303.

```
[up0300-aio-2/rm0303] rcm# config
Entering configuration mode terminal
[up0300-aio-2/rm0303] rcm(config)# system mode running
[up0300-aio-1/rce303] rcm(config)# commit comment <CRNUMBER>
```

Aguarde alguns minutos e verifique se o sistema está em 100,0%.

```
[up0300-aio-1/rm0303] cee# show system
```

Procedimento de verificação

Verifique se todos os pods estão nos estados UP e Running em ambos os nós AIO com esses comandos.

```
on AIO nodes:
kubect1 get ns
kubect1 get pods -A -o wide
```

```
on RCM ops-centers:
rcm show-status
```

Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.