

# Stappen naar RMA op de RCM-gebaseerde AIO-server in CNDP-implementatie

## Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[Ken de RCM IP-camera](#)

[Back-upprocedure](#)

[Configuratie back-up](#)

[Procedure vooraf](#)

[Voorcontroles op AIO](#)

[Voorcontroles voorbeeld van uitvoer](#)

[Uitvoeringsprocedure](#)

[Stappen om op RCM uit te voeren voordat u het AIO-knooppunt sluit](#)

[Stap 3 om het knooppunt van Kubernetes uit te voeren voordat u het AIO-knooppunt sluit](#)

[Onderhoudsprocedure voor servers](#)

[Kubernetes herstelprocedure](#)

[Maatregelen om het knooppunt van Kubernetes te implementeren via Power-over-AIO knooppunt](#)

[RCM-herstelprocedure](#)

[Stappen om uit te voeren op CEE en RCM Ops-centers om de toepassing te herstellen](#)

[Verificatieprocedure](#)

## Inleiding

Dit document beschrijft de gedetailleerde procedure voor Return Material Authorization (RMA) voor de op Redundancy Configuration Manager (RCM) gebaseerde All-in-One (AIO) server in Cloud Native Deployment Platform (CNDP) voor hardwareproblemen of onderhoudsgerelateerde activiteiten.

## Voorwaarden

### Vereisten

Cisco raadt kennis van de volgende onderwerpen aan:

- RCM
- Kubernetes

### Gebruikte componenten

De informatie in dit document is gebaseerd op de RCM versie - rcm.2021.02.1.i18

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u zorgen dat u de potentiële impact van elke opdracht begrijpt.

## Ken de RCM IP-camera

Dit document legt het RCM-ontwerp uit dat bestaat uit twee AIO-knooppunten met twee RCM-glasvezelknooppunten en één RCM-CEE voor elk AIO-knooppunt.

Het in dit artikel bedoelde RCM AIO-knooppunt voor de RMA is AIO-1 (AI0301), dat beide RCM-knooppunten in de PRIMAIRE staat bevat.

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

## Back-upprocedure

### Configuratie back-up

Om te beginnen, verzamel de configuratie backup van de in werking stellen-instelling van RCM-ingangen die op het doel-AIO-knooppunt draaien.

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

Verzamel de in werking stellen-configuratie van RCM CEE-operatoren die op het doel-AIO-knooppunt draaien.

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

## Procedure vooraf

### Voorcontroles op AIO

Verzamel de opdrachtoutput van beide AIO-knooppunten en controleer of alle pods in de actieve status zijn.

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

### Voorcontroles voorbeeld van uitvoer

Merk op dat de twee RCM-schermen en één RCM-CEE-operatoren op de AIO-1-knooppunt actief zijn

```
cloud-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-aio-1-master-1:~$
```

Meld u aan bij zowel het RCM-OpenCenter van AIO-1 en controleer de status.

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

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

Herhaal de zelfde stappen op het AIO-2 knooppunt waar de andere twee RCM-operatoren

overeenkomen met het AIO-1 knooppunt aanwezig is.

```
cloud-user@up0300-aio-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-aio-2-master-1:~$
```

Meld u aan bij zowel het RCM-OpenCenter van AIO-2 en controleer de 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#
```

## Uitvoeringsprocedure

### Stappen om op RCM uit te voeren voordat u het AIO-knooppunt sluit

1. Aangezien beide RCM's op AIO-1 MASTER zijn, kunt u ze migreren naar BACKUP.

a. Om dat te doen, moet u de **rcm migraeren primaire** opdracht op de actieve RCMs uitvoeren voordat u de AIO-1 server uitschakelt.

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

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

b. Controleer of de status nu BACKUP is op AIO-1.

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

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

c. Controleer of de status nu MASTER is op AIO-2 en zorg ervoor dat het MASTER is.

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

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

d. RCM-shutdown uitvoeren op zowel rm0301 als 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. We moeten ook de CEE-ops sluiten die op de AIO-1-opdrachten worden uitgevoerd.

```
[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
```

Wacht een paar minuten en controleer het systeem op 0,0%.

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

3. Controleer of er geen peulen voor RCM- en CEE-naamruimtes zijn, behalve voor documentatie, "smart agent", "ops-center-rcm" en "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
```

## Stap 3 om het knooppunt van Kubernetes uit te voeren voordat u het AIO-knooppunt sluit

Pomp het knooppunt Kubernetes af zodat de poten en services die hiermee gepaard gaan, zorgvuldig worden afgesloten. De planner zou dit Kubernetes knooppunt niet langer selecteren en pods uit dat knooppunt verwijderen. Eén knooppunt tegelijk afvoeren.

Aanmelden bij SMI Cluster Manager.

```
cloud-user@bot-deployer-cm-primary:~$ kubectl get svc -n smi-cm
NAME                                TYPE           CLUSTER-IP      EXTERNAL-IP
PORT(S)                             AGE
cluster-files-offline-smi-cluster-deployer ClusterIP      10.102.108.177  <none>
8080/TCP                             78d
iso-host-cluster-files-smi-cluster-deployer ClusterIP      10.102.255.174  192.168.0.102
80/TCP                                 78d
iso-host-ops-center-smi-cluster-deployer ClusterIP      10.102.58.99    192.168.0.100
3001/TCP                               78d
netconf-ops-center-smi-cluster-deployer ClusterIP      10.102.108.194  10.244.110.193
3022/TCP,22/TCP                       78d
ops-center-smi-cluster-deployer      ClusterIP      10.102.156.123  <none>
8008/TCP,2024/TCP,2022/TCP,7681/TCP,3000/TCP,3001/TCP 78d
squid-proxy-node-port                NodePort       10.102.73.130   <none>
3128:31677/TCP                       78d
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  -
```

### Het hoofdknooppunt leegpompen:

[bot-deployer-cm-primary] SMI Cluster Deployer# clusters up0300-aio-1 nodes master-1 actions sync drain remove-node true

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

message accepted

### Merk het master-1 knooppunt op in onderhoudsmodus:

[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

### Start Cluster sync en controleer de logbestanden voor de sync-actie:

[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

### Monsteruitvoer voor clustersync-logbestanden:

[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    ignored=0
cmts-worker2           : ok=1    changed=0    unreachable=0    failed=0    skipped=2
rescued=0    ignored=0
cmts-worker3           : ok=1    changed=0    unreachable=0    failed=0    skipped=2
rescued=0    ignored=0
master1                : ok=1    changed=0    unreachable=0    failed=0    skipped=2
rescued=0    ignored=0
master2                : ok=1    changed=0    unreachable=0    failed=0    skipped=2
rescued=0    ignored=0
master3                : ok=1    changed=0    unreachable=0    failed=0    skipped=2
rescued=0    ignored=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

```

## Onderhoudsprocedure voor servers

Schakel de server uit via CIMC. Ga verder met de hardware-gerelateerde onderhoudsactiviteit zoals gedefinieerd in de Hardware MoP en zorg ervoor dat alle gezondheidscontroles worden doorgegeven nadat de server is ingeschakeld.

Opmerking: Dit artikel heeft geen betrekking op de hardware- of onderhoudsactiviteit MoP voor de server, aangezien deze verschilt van de probleemverklaring

## Kubernetes herstelprocedure

### Maatregelen om het knooppunt van Kubernetes te implementeren via Power-over-AIO knooppunt

## Aanmelden bij SMI Cluster Manager:

```
cloud-user@bot-deployer-cm-primary:~$ kubectl get svc -n smi-cm
NAME                                TYPE           CLUSTER-IP      EXTERNAL-IP
PORT(S)                             AGE
cluster-files-offline-smi-cluster-deployer ClusterIP      10.102.108.177  <none>
8080/TCP                             78d
iso-host-cluster-files-smi-cluster-deployer ClusterIP      10.102.255.174  192.168.0.102
80/TCP                                 78d
iso-host-ops-center-smi-cluster-deployer ClusterIP      10.102.58.99    192.168.0.100
3001/TCP                               78d
netconf-ops-center-smi-cluster-deployer ClusterIP      10.102.108.194  10.244.110.193
3022/TCP,22/TCP                       78d
ops-center-smi-cluster-deployer      ClusterIP      10.102.156.123  <none>
8008/TCP,2024/TCP,2022/TCP,7681/TCP,3000/TCP,3001/TCP 78d
squid-proxy-node-port                NodePort      10.102.73.130   <none>
3128:31677/TCP                       78d
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  -
```

## Schakel de onderhoudsvlag uit om de master-1 weer in cluster te voegen.

```
[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
```

## Herstel de basisknooppunten en -services met clustersync-actie.

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

## Controleer de logbestanden voor de sync-actie.

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

## Controleer de clusterstatus van de AIO-1-master.



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

Uitvoer van monster:

```
[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
```

## RCM-herstelprocedure

### Stappen om uit te voeren op CEE en RCM Ops-centers om de toepassing te herstellen

Update CEE opscenter en RCM opscenter in actieve modus.

Configureer de actieve modus voor 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
```

Wacht een paar minuten en controleer of het systeem 100,0% is.

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

Configureer de **actieve** modus voor 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>
```

Wacht een paar minuten en controleer of het systeem 100,0% is.

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

Configureer de **actieve** modus voor 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>
```

Wacht een paar minuten en controleer of het systeem 100,0% is.

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

## Verificatieprocedure

Controleer of de pods helemaal UP zijn en de status **Running** op beide AIO-knooppunten met deze opdrachten.

on AIO nodes:

```
kubect1 get ns
```

```
kubect1 get pods -A -o wide
```

on RCM ops-centers:

```
rcm show-status
```