

CIMC configureren via Ansible Rest API Module

Inhoud

[Inleiding](#)

[Voorwaarden](#)

[Vereisten](#)

[Gebruikte componenten](#)

[CIMC API - Overzicht](#)

[Configuraties](#)

[1. Zoek de klasse of het DN van het CIMC Managed Object \(MO\)](#)

[1 bis. Gebruik de API om in te loggen op de CIMC en de cookie-informatie op te halen](#)

[1 ter. De API-querymethode configResolveDN gebruiken om alle beheerde objectgegevens \(MO\) op te halen](#)

[Voorbeeld 1: Query de klasse en DN van de tijdzone](#)

[Voorbeeld 2: Query de klasse en DN van de hostnaam](#)

[2. CIMC beheren via REST API](#)

[Gebruik de API-methode configResolveClass om informatie op te halen](#)

[API-methode configConfMo gebruiken om de configuratie te wijzigen](#)

[3. Voorbeeld van een CIMC-automatiseringsconfiguratie](#)

[Gerelateerde informatie](#)

Inleiding

In dit document wordt beschreven hoe u de Cisco Integrated Management Controller (CIMC) configureert via de module Ansible REST API.

Voorwaarden

Vereisten

Cisco raadt kennis van de volgende onderwerpen aan:

- UCS CIMC
- API
- bedachtzaam

Gebruikte componenten

- UCS C220-M4, versie 4.1 (2f)
- Een client met Postman en Ansible versie 2.14.5

De informatie in dit document is gebaseerd op de apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een

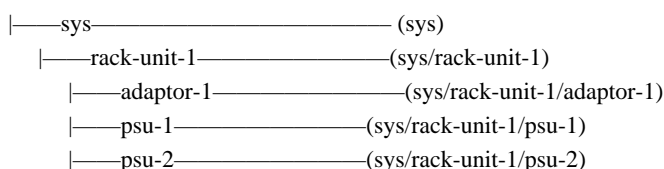
opgeschoonde (standaard)configuratie. Als uw netwerk live is, moet u zorgen dat u de potentiële impact van elke opdracht begrijpt.

CIMC API - Overzicht

Alle fysieke en logische componenten waaruit Cisco UCS bestaat, zijn vertegenwoordigd in een hiërarchisch managementinformatiemodel (MIM), ook wel MIT genoemd. Elk knooppunt in de structuur vertegenwoordigt een beheerd object (MO) of een groep objecten die de beheerdersstatus en de operationele status bevat.

De hiërarchische structuur begint bovenaan (sys) en bevat bovenliggende en onderliggende knooppunten. Elke node in deze structuur is een beheerd object en elk object in Cisco UCS heeft een unieke DN (Distinguished Name) die het object en de plaats in de structuur beschrijft. Beheerde objecten zijn abstracties van de Cisco UCS-bronnen, zoals CPU's, DIMM's, adapterkaarten, ventilatoren en voedingseenheden.

Illustratie van de CIMC MIM-structuur:



Objectnaam:

- DN: Met de onderscheidende naam kunt u een doelobject ondubbelzinnig identificeren.
- RN: De relatieve naam identificeert een object binnen de context van het bovenliggende object.

Bijvoorbeeld deze onderscheidende naam:

```
<dn = "sys/rack-unit-1/adaptor-1/host-eth-eth2"/>
```

Het bestaat uit 4 relatieve namen:

```
topSystem MO: rn="sys"
computeRackUnit MO: rn ="rack-unit-1"
adaptorUnit MO: rn="adaptor-<id>"
adaptorHostEthIf MO: rn="host-eth-<id>"
```

API gebruikt in dit artikel:

- Authenticatie: aaaLogin. Initiële methode voor het inloggen. Gebruik de aaaLogin methode om een geldige cookie te krijgen.
- Query: configResolveDn. Hiermee worden objecten opgehaald door DN.
- Configuratie: configConfMo. De methode configConfMo wordt gebruikt om een of meer eigenschappen in een beheerd object (MO) te configureren. De te configureren MO wordt uniek geïdentificeerd door een DN (Distinguished Name).



Opmerking:

Veel querymethoden bevatten een inHierarchical-argument dat een Booleaanse waarde accepteert (true/ja of false/nee). Wanneer dit argument wordt ingesteld op true, retourneert de methode alle onderliggende objecten binnen de hiërarchie.

Configuraties

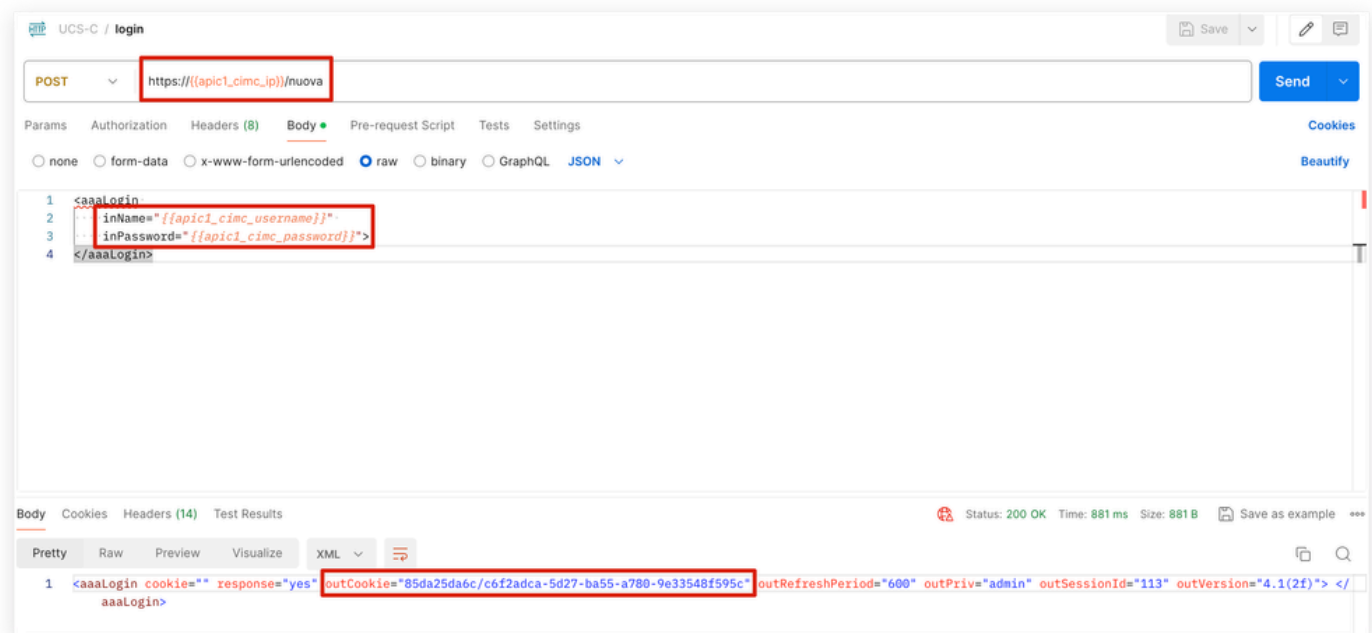
1. Zoek de klasse of het DN van het CIMC Managed Object (MO)

Om de configuratie van CIMC via de API te automatiseren, is het essentieel om de specifieke klasse- of DN-informatie (Distinguished Name) te bepalen die is gekoppeld aan de beheerde objecten (MO's) die u wilt configureren.

1 bis. Gebruik de API om in te loggen op de CIMC en de cookie-informatie op te halen

Stuur een POST aanvraag naar https://{{apic_cimc_ip}}/nuova en specificeer de aaaLogin methode. Voer een gebruikersnaam en wachtwoord in.

Kopieer de cookie uit de API-reactie.



Als alternatief kunt u curl gebruiken om de cookie-informatie op te halen.

```
curl -k -d "
```

```
" https://apic_cimc_ip/nuova
```

1 ter. De API-querymethode configResolveDN gebruiken om alle beheerde objectgegevens (MO) op te halen

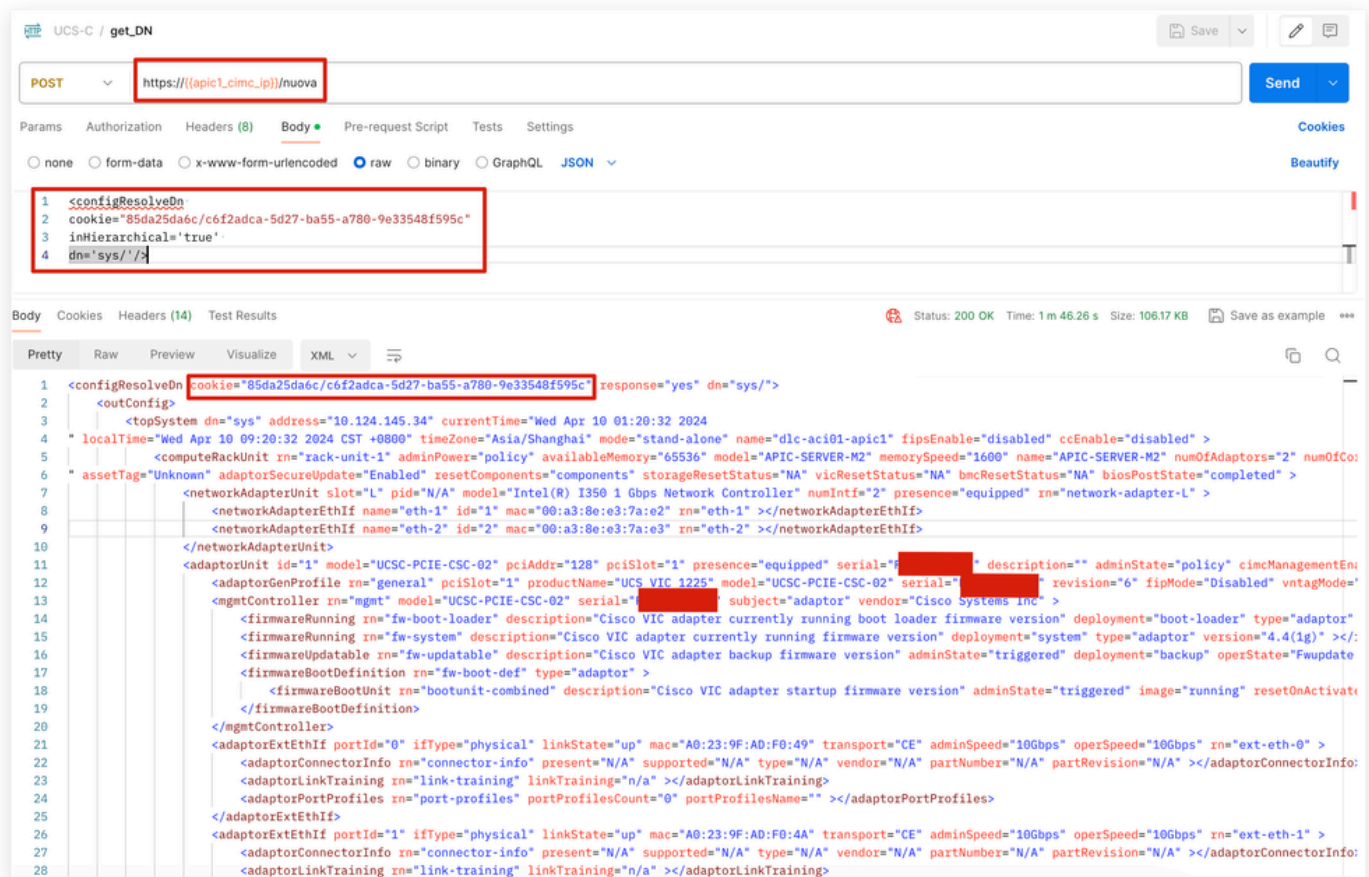
Wanneer u configResolveDN gebruikt met inHierarchical="true" en dn="sys/", wordt alle informatie over beheerde objecten (MO) opgehaald van CIMC.

configResolveDn: de methode configResolveDn haalt één beheerd object op voor een opgegeven DN.

inHierarchical=true: als deze optie is ingesteld op true, worden alle onderliggende knooppuntgegevens geretourneerd. Deze combinatie maakt het mogelijk alle knooppunten en subnodes MO-informatie van CIMC op te halen.

dn="sys/": Dit is de bovenste wortel van MIT.

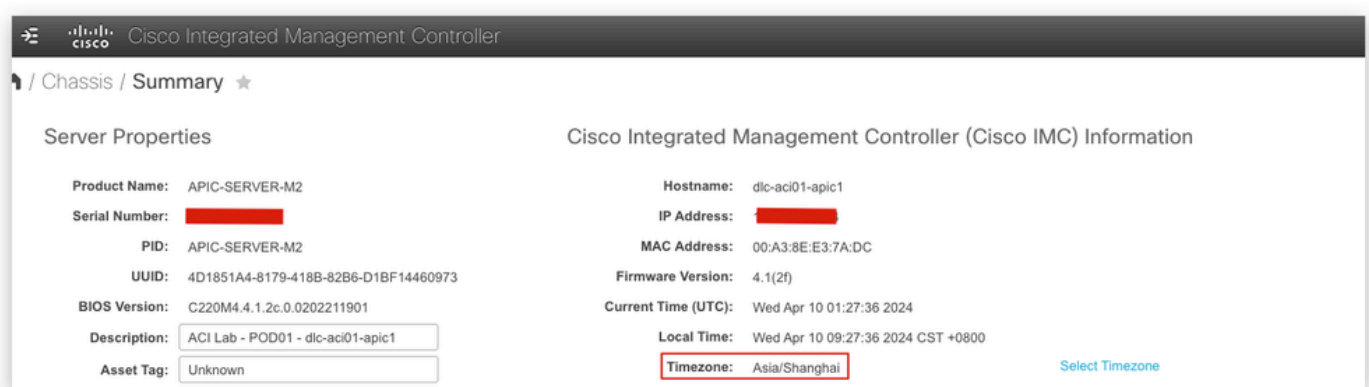
API-respons:



Kopieer de uitvoerings reactie van Postman naar een teksteditor zoals Kladblok, PyCharm of Visual Studio Code voor verder zoeken van de klasse en DN op basis van MO.

Voorbeeld 1: Query de klasse en DN van de tijdzone

De tijdzone die is geconfigureerd in de huidige CIMC GUI is "Azië/Shanghai".



Zoek naar 'Azië/Shanghai' uit de resultaten die door Postman in stap 1b zijn geretourneerd. De tijdzone is "Azië / Shanghai", de klasse is "topSystem", en de DN is "sys /".

<#root>

```

<configResolveDn cookie="85da25da6c/c6f2adca-5d27-ba55-a780-9e33548f595c" response="yes" dn="sys/">
  <outConfig>

```

```
<topSystem
```

```
dn="sys"
```

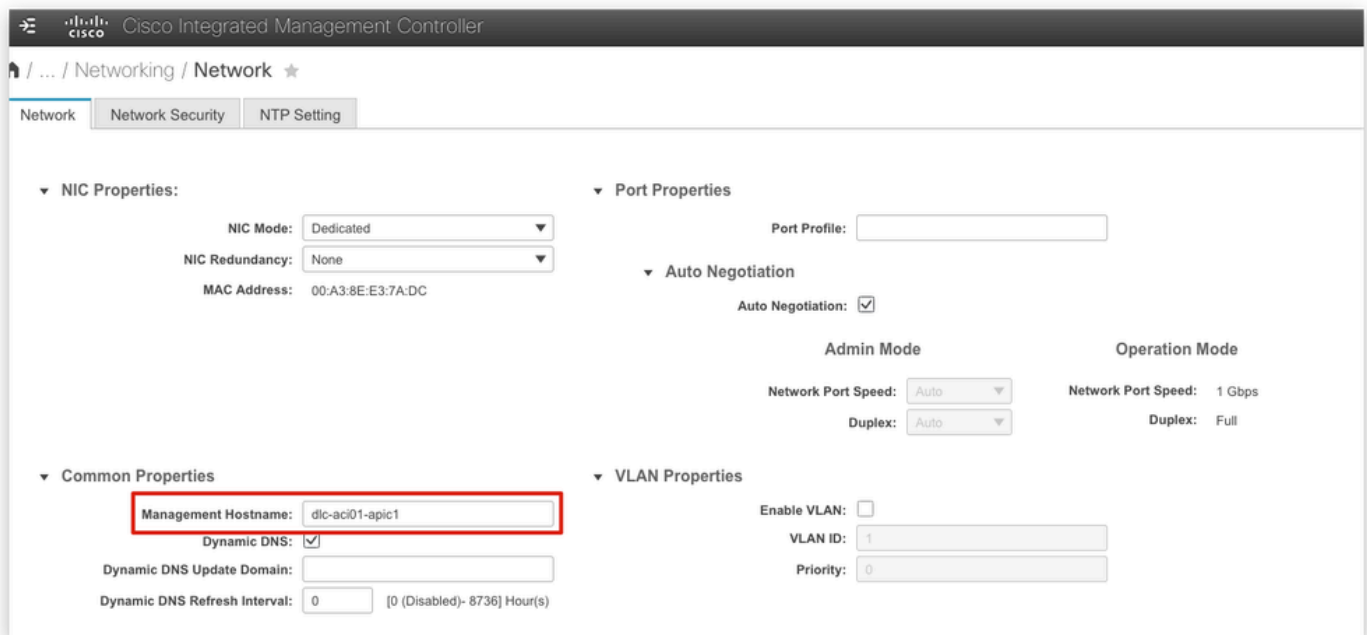
```
address="a.b.c.d" currentTime="Wed Apr 10 01:05:12 2024  
" localTime="Wed Apr 10 09:05:12 2024 CST +0800"
```

```
timeZone="Asia/Shanghai"
```

```
mode="stand-alone" name="dlc-aci01-apic1" fipsEnable="disabled" ccEnable="disabled" >
```

Voorbeeld 2: Query de klasse en DN van de hostnaam

De hostnaam die is geconfigureerd in de huidige CIMC GUI is "dlc-aci01-apic1".



Zoek naar "dlc-aci01-apic1" in de resultaten geretourneerd door Postman. De hostnaam is "dlc-aci01-apic1", de klasse is "mgmtIf" en de rn is "if-1".

```
<#root>
```

```
<mgmtIf rn="if-1" description="Management Interface Network Settings" id="1" extEnabled="yes" extIp="a.  
ifType="physical" mac="00:A3:8E:E3:7A:DC"
```

```
hostname="dlc-aci01-apic1"
```

```
dhcpEnable="no" dnsUsingDhcp="no" ddnsEnable="yes" ddnsDomain=""  
dnsPreferred="a.b.c.z" dnsAlternate="0.0.0.0" ddnsRefreshInterval="0" nicMode="dedicated" vicSlot="0" n  
vlanEnable="no" vlanId="1" vlanPriority="0" portProfile="" v6extEnabled="no" v6extIp="::" v6extGw="::"  
v6SlaacIp="::" v6dhcpEnable="no" v6dnsUsingDhcp="no" v6dnsPreferred="::" v6dnsAlternate="::" subject="b  
adminNetSpeed="auto" adminDuplex="auto" operNetSpeed="1Gbps" operDuplex="full" >
```

Vervolgens query CIMC visore van https://CIMC_IP/visore.html, de hostnaam "dlc-aci01-apic1" komt overeen met de DN= `sys/rack-unit-1/mgmt/if-1`.

Filter

Class or DN: inHierarchical

Property: Op: Val1: Val2:

[Display XML of last query](#)

Total objects shown: 1

| mgmtIf | |
|---------------|---|
| dn | sys/rack-unit-1/mgmt/if-1 < > |
| description | Management Interface Network Settings |
| id | 1 |
| extEnabled | yes |
| extIp | |
| extMask | 255.255.255.0 |
| extGw | |
| ifType | physical |
| mac | 00:A3:8E:E3:7A:DC |
| hostname | dlc-aci01-apic1 |
| dhcpEnable | no |
| dnsUsingDhcp | no |
| ddnsEnable | yes |

2. CIMC beheren via REST API

- In stap 1 hebt u al de klasse en de DN (Distinguished Name) geïdentificeerd die overeenkomen met het beheerde object (MO).
- U kunt de Ansible `community.general.imc_rest` module gebruiken om CIMC via API te beheren. Detailinformatie verwijst naar: [imc_rest module Documentatie](#)

Gebruik de API-methode `configResolveClass` om informatie op te halen

`configResolveClass`: de methode haalt beheerde objecten in een bepaalde klasse op. Als `inHierarchical=true`, bevatten de resultaten kinderen. Als u bijvoorbeeld de firmwareversie opvraagt, gebruikt u de API-methode `configResolveClass` en geeft u de `classID` van de MO op.

Uitvoer van Ansible-scriptinhoud:

```
<#root>
```

```
- name: IMC login and check
community.general.imc_rest:
  hostname: '{{ imc_hostname }}'
  username: '{{ imc_username }}'
  password: '{{ imc_password }}'
  validate_certs: false # only do this when you trust the network!
```

content: |

```
<  
configResolveClass  
inHierarchical='false'  
classId='firmwareRunning'  
>
```

API-methode configConfMo gebruiken om de configuratie te wijzigen

Als u de configuratie van MO wilt wijzigen met behulp van de CIMC API, gebruikt u de methode configConfMo. Deze methode is ontworpen om de instellingen van een specifieke MO te configureren of aan te passen. Bij het aanroepen van configConfMo is het belangrijk om de exacte klasse- of DN-informatie te verstrekken van de MO die u wilt wijzigen.

Filter

Class or DN: inHierarchical

Property: Op: Val1: Val2:

[Display XML of last query](#)

Total objects shown: 1

| computeRackUnit ? | |
|-----------------------------------|--------------------------------------|
| dn | sys/rack-unit-1 < > |
| adminPower | policy |
| availableMemory | 65536 |
| model | APIC-SERVER-M2 |
| memorySpeed | 1600 |
| name | APIC-SERVER-M2 |
| numOfAdaptors | 2 |
| numOfCores | 12 |
| numOfCoresEnabled | 12 |
| numOfCpus | 2 |
| numOfEthHostIfs | 2 |
| numOfFcHostIfs | 2 |
| numOfThreads | 12 |
| operPower | on |
| originalUuid | 4D1851A4-8179-418B-82B6-D1BF14460973 |
| presence | equipped |
| serverId | 1 |
| serial | FCH2113V2WF |
| totalMemory | 65536 |
| usrLbl | ACI Lab - POD01 - dlc-aci01-apic1 |
| uuid | 4D1851A4-8179-418B-82B6-D1BF14460973 |

Uitvoer van Ansible-scriptinhoud:

<#root>

```
- name: change CIMC description
community.general.imc_rest:
  hostname: '{{ imc_hostname }}'
  username: '{{ imc_username }}'
  password: '{{ imc_password }}'
  validate_certs: false
```

content: |

<

```
computeRackUnit dn="sys/rack-unit-1" usrLbl="new_lab_CIMC_description"  
/>
```

examples:

3. Voorbeeld van een CIMC-automatiseringsconfiguratie

Cisco APIC is Cisco ACI controller software geïnstalleerd op de UCS C220 serie. De workflow illustreert het geautomatiseerde proces van het opnieuw in beeld brengen van APIC-software.

1. Login to CIMC with pre-check
 - Retrieve firmware version
 - Retrieve faults
 - Retrieve TPM status
2. Update CIMC configurations
 - Update management hostname
 - Update Description
 - Update Timezone
 - Update ntp
 - Enable SOL
 - Update CIMC mapping vmedia
 - Update CIMC boot order to CIMC-map
 - Reboot CIMC
3. Ansible run shell expect to monitor installation status and enter iso link for APIC installation speed up
4. Retrieve CIMC post installation status
 - Update CIMC boot order back to HDD
 - Power-on host

Ansible module voorbeeld:



Opmerking: Het voorbeeld bevat alleen inhoudsinfo, de volledige zichtbare module verwijst naar `community.general.imc_rest` module van de officiële website van Ansible

<#root>

- name: Login to CIMC with pre-check
content: |

```
<
  configResolveClass
    inHierarchical='false'
  classId
  ='firmwareRunning'/>
```

```
<
  configResolveClass
    inHierarchical='false'
  classId
  ='faultInst'/>
```

```
<
  configResolveClass
    inHierarchical='false'
  classId
  ='equipmentTpm'/>
```

- name: IMC update CIMC infra info
content: |

<

mgmtIf

dn="sys/rack-unit-1/mgmt/if-1"

hostname="dlc-aci01-apic1"/>

<

computeRackUnit

dn="sys/rack-unit-1"

usrLbl="ACI Lab - POD01 - dlc-aci01-apic1"/>

<

topSystem

dn="sys"

timeZone="Asia/Shanghai"/>

<

commNtpProvider

dn="sys/svc-ext/ntp-svc"

ntpServer1="ntp.es1.cisco.com"/>

```
- name: Update CIMC configurations
  content: |
```

```
<
```

```
lsbootVMedia
```

```
dn="sys/rack-unit-1/boot-precision/vm-CIMC-map"
```

```
name="CIMC-map" type="VMEDIA" subtype="cimc-mapped-dvd" order="1" state="Enabled" />
```

```
<
```

```
commVMediaMap
```

```
volumeName="ACI-automation" map="www" remoteShare="http://a.b.c.d/Images/ACI/4/4.2/" remoteFile="aci-a
```

```
dn="sys/svc-ext/vmedia-svc/vmmap-ACI-automation"
```

```
>
```

```
<
```

```
computeRackUnit
```

```
dn="sys/rack-unit-1"
```

```
adminPower="hard-reset-immediate" />
```

```
# Ansible run shell expect to monitor installation status and enter iso link for APIC installation speed
```

```
- name: copy apic init script to
```

```
  template:
```

```
    src: "init.sh"
```

```
    dest: /tmp/init.sh
```

```
  delegate_to: localhost
```

```
- name: Make script executable
```

```
  file:
```

```
    path: /tmp/init.sh
```

```
    mode: "+x"
```

```
  delegate_to: localhost
```

```
  tags:
```

```
    - render
```

```
    - init
```

```
- name: Run the generated script
```

```
command: /tmp/init.sh
delegate_to: localhost
changed_when: no
tags:
  - script
```

```
- name: Retrieve CIMC post installation status
  content: |
```

<

lsbootVMedia

```
dn="sys/rack-unit-1/boot-precision/vm-CIMC-map"
name="CIMC-map" status='removed' />
```

<

commVMediaMap

```
dn="sys/svc-ext/vmedia-svc/vmmmap-ACI-automation"  
volumeName="ACI-automation" status='removed' >
```

<

lsbootStorage

```
dn="sys/rack-unit-1/boot-policy/storage-read-write"  
access="read-write" order="1" type="storage"/>
```

<

```
computeRackUnit dn="sys/rack-unit-1"  
adminPower="up" />
```

```
delegate_to: localhost  
tags:  
- retrieve_CIMC_status
```

Gerelateerde informatie

[Cisco UCS Rack-Mount Servers Cisco IMC XML API Programmeergids](#)

[community.general.imc_rest module – Beheer Cisco IMC hardware via de REST API](#)

[Referentie informatiemodel UCS Manager](#)

Over deze vertaling

Cisco heeft dit document vertaald via een combinatie van machine- en menselijke technologie om onze gebruikers wereldwijd ondersteuningscontent te bieden in hun eigen taal. Houd er rekening mee dat zelfs de beste machinevertaling niet net zo nauwkeurig is als die van een professionele vertaler. Cisco Systems, Inc. is niet aansprakelijk voor de nauwkeurigheid van deze vertalingen en raadt aan altijd het oorspronkelijke Engelstalige document ([link](#)) te raadplegen.