REST API を使用した Cisco APIC の設定

REST API を使用した APIC クラスタの拡大

クラスタは、実際のサイズを目標サイズに合わせます。目標サイズが実際のサイズよりも大きい場合、クラスタ サイズが拡大します。

手順

ステップ 1 APIC クラスタのサイズを拡大するために目標のクラスタ サイズを設定します。

例：

```
POST
https://<IP address>/api/node/mo/uni/controller.xml
<infraClusterPol name='default' size=3/>
```

ステップ 2 クラスタに追加する APIC コントローラを物理的に接続します。

REST API を使用した APIC クラスタの縮小

コントローラを削除してクラスタサイズを縮小するには、次の手順を使用します。クラスタ サイズの縮小の詳細については、Cisco APIC クラスタの縮小 を参照してください。

（注） Cisco APIC リリース 6.0(2) 以降、デコミッション操作を強制できるようにするために、API コマンドに2つの追加プロパティが追加されました。新しいオブジェクトプロパティは次のとおりです。

- `infraClusterPol:shrink`
  - `false`：（デフォルト）ターゲット クラスタ サイズ（`infraClusterPol:size`）が現在の運用クラスタ サイズより小さい場合、以前のリリースと同様に、削除する APIC を手動で廃止する必要があります。
  - `true`：ターゲット クラスタ サイズが現在の運用クラスタ サイズよりも小さい場合、クラスタ縮小デコミッションがトリガーされます。削除される APIC については、コントローラ ID 番号が最も大きい APIC から自動的にデコミッションされます。
- `infraWiNode:force`
  - `false`：（デフォルト）クラスタが異常な場合、またはアップグレード状態である場合には、デコミッションが適切でない可能性があるため、それ以外の場合にのみデコミッションを続行します。
  - `true`：クラスタの状態に関係なく、デコミッションを続行します。

次に、クラスタを 3 つの APIC コントローラから 1 つのコントローラに縮小する例を示します。ターゲット サイズを 1 にするには、APIC3 と APIC2 をこの順序で廃止する必要があります。

手順

ステップ 1 APIC クラスタのサイズを縮小するため、目標のクラスタ サイズを設定します。

`shrink='true'` を使用してクラスタ サイズを縮小すると、削除される APIC は自動的にデコミッションされます。それ以外の場合は、手動でデコミッションする必要があります。

例：

Cisco APIC リリース 6.0(1) 以前：

```
POST
https://<IP address>/api/node/mo/uni/controller.xml
<infraClusterPol name='default' size=1 />
```

次の手順に示すように、削除する APIC を手動でデコミッションする必要があります。

REST API を使用した Cisco APIC の設定

例：

Cisco APIC リリース 6.0(2) 以降で「shrink」プロパティを使用する場合：

```
POST
https://<IP address>/api/node/mo/uni/controller.xml
<infraClusterPol name='default' size=1 shrink='true' />
```

`shrink='true'` を使用すると、次の手順をスキップできます。削除する APIC は自動的にデコミッションされます。

```
POST
https://<IP address>/api/node/mo/uni/controller.xml
<infraClusterPol name='default' size=1 shrink='false' />
```

`shrink='false'` を使用する場合には、次の手順に示すように、削除する APIC を手動でデコミッションする必要があります。

ステップ 2 クラスタ縮小のための APIC1 上の APIC3 の解放

例：

Cisco APIC リリース 6.0(1) 以前：

```
POST
https://<IP address>/api/node/mo/topology/pod-1/node-1/av.xml
<infraWiNode id=3 adminSt='out-of-service'/>
```

デコミッションは、クラスターが正常な状態にある場合にのみ続行されます。

例：

Cisco APIC リリース 6.0(2) 以降で「force」プロパティを使用する場合：

```
POST
https://<IP address>/api/node/mo/topology/pod-1/node-1/av.xml
<infraWiNode id=3 adminSt='out-of-service' force='true' />
```

`force='true'` の場合、クラスタの状態に関係なくデコミッションが進行します。

ステップ 3 クラスタ縮小のための APIC1 上の APIC2 の解放

例：

Cisco APIC リリース 6.0(1) 以前：

```
POST
https://<IP address>/api/node/mo/topology/pod-1/node-1/av.xml
<infraWiNode id=2 adminSt='out-of-service'/>
```

デコミッションは、クラスターが正常な状態にある場合にのみ続行されます。

例：

Cisco APIC リリース 6.0(2) 以降で「force」プロパティを使用する場合：

```
POST
https://<IP address>/api/node/mo/topology/pod-1/node-1/av.xml
<infraWiNode id=2 adminSt='out-of-service' force='false' />
```

`force='false'` の場合、クラスターが正常な状態にある場合にのみデコミッションが続行されます。

稼動クラスタのサイズが縮小するのは、最後のアプライアンスが解放されたときで、管理サイズを変更したときではありません。各コントローラを解放した後、そのコントローラの動作状態が未登録になり、すでにクラスタ内で稼動していないことを確認します。

（注） デコミッションされた APIC コントローラがファブリックからすぐに削除されない場合、再検出される可能性があり、問題が発生する可能性があります。その場合、コントローラを削除するために APIC クラスタのサイズ縮小 の説明に従います。

REST API を使用してアクティブ APIC とスタンバイ APIC を切り替える

REST API を使用してアクティブな APIC とスタンバイ APIC を切り替えるには、この手順を使用します。

手順

アクティブ APIC とスタンバイ APIC を切り替えます。

```
URL for POST: https://ip
address/api/node/mo/topology/pod-initiator_pod_id/node-initiator_id/av.xml
Body: <infraWiNode id=outgoing_apic_id targetMbSn=backup-serial-number/>
where initiator_id = id of an active APIC other than the APIC being replaced.
pod-initiator_pod_id = pod ID of the active APIC
backup-serial-number = serial number of standby APIC
```

例：

```
https://ip address/api/node/mo/topology/pod-1/node-1/av.xml
<infraWiNode id=2 targetMbSn=FCH1750V00Q/>
```

REST API を使用した未登録スイッチの登録

この手順を使用して、REST API を使用して **[ファブリック メンバーシップ (Fabric Membership)]** 作業ウィンドウの **[保留中ノードの登録 (Nodes Pending Registration)]** タブからスイッチを登録します。

（注） この手順は、「REST API を使用したディスカバリ前のスイッチの追加」と同じです。コードを適用すると、システムはノードが存在するかどうかを判断し、存在しない場合はそのノードを追加します。ノードが存在しない場合、システムにより登録されます。

手順

スイッチ説明を追加します。

例：

```
POST
https://<IP address>/api/policymgr/mo/uni.xml

<!-- /api/policymgr/mo/uni.xml -->
<polUni>
<ctrlrInst>
    <fabricNodeIdentPol>
        <fabricNodeIdentP  nodeType="tier-2-leaf" podId="1" serial="XXXXXXXXX"
        name="tier-2-leaf-leaf1" nodeId="101"/>

    </fabricNodeIdentPol>
</ctrlrInst>
</polUni>
```

REST API を使用したディスカバリ前のスイッチの追加

この手順を使用して、REST API を使用して **[ファブリック メンバーシップ (Fabric Membership)]** 作業ウィンドウの **[保留中ノードの登録 (Nodes Pending Registration)]** タブにスイッチを追加します。

（注） この手順は、「REST API を使用した未登録スイッチの登録」と同じです。コードを適用すると、システムはノードが存在するかどうかを判断し、存在しない場合はそのノードを追加します。ノードが存在しない場合、システムにより登録されます。

手順

スイッチ説明を追加します。

例：

```
POST
https://<IP address>/api/policymgr/mo/uni.xml

<!-- /api/policymgr/mo/uni.xml -->
<polUni>
<ctrlrInst>
    <fabricNodeIdentPol>
        <fabricNodeIdentP  nodeType="tier-2-leaf" podId="1" serial="XXXXXXXXX"
        name="tier-2-leaf1" nodeId="101"/>
```

```
    </fabricNodeIdentPol>
</ctrlrInst>
</polUni>
```

REST API を使用して、メンテナンス モードにスイッチを削除

REST API を使用して、メンテナンス モードにスイッチを削除するのにには、次の手順を使用します。

手順

メンテナンス モードにスイッチを削除します。

例：

```
POST
https://<IP address>/api/node/mo/uni/fabric/outofsvc.xml

<fabricOOServicePol
    descr=""
    dn=""
    name="default"
    nameAlias=""
    ownerKey=""
    ownerTag="">
  <fabricRsDecommissionNode
      debug="yes"
      dn=""
      removeFromController="no"
      tDn="topology/pod-1/node-102"/>
</fabricOOServicePol>
```

REST API を使用した操作モードへのスイッチの挿入

REST API を使用して操作モードにスイッチを挿入するには、次の手順を使用します。

手順

操作モードにスイッチを挿入します。

例：

```
POST
https://<IP address>/api/node/mo/uni/fabric/outofsvc.xml

<fabricOOServicePol
    descr=""
    dn=""
    name="default"
    nameAlias=""
    ownerKey=""
    ownerTag="">
  <fabricRsDecommissionNode
      debug="yes"
      dn=""
      removeFromController="no"
      tDn="topology/pod-1/node-102"
      status="deleted"/>
</fabricOOServicePol>
```

REST API を使用したリモート ロケーションの設定

この手順では、REST API を使用してリモート ロケーションを作成する方法について説明します。

```
<fileRemotePath name="local" host="host or ip" protocol="ftp|scp|sftp" remotePath="path
 to folder" userName="uname" userPasswd="pwd" />
```

REST API を使用したオンデマンド テクニカル サポート ファイルの送信

手順

ステップ 1 REST API を使用して次の例のような XML を POST 送信し、テクニカル サポート ファイルのリモート宛先を設定します。

例：

```
<fileRemotePath userName="" remotePort="22" remotePath="" protocol="sftp" name="ToSupport"
 host="192.168.200.2"
dn="uni/fabric/path-ToSupport" descr="">

<fileRsARemoteHostToEpg tDn="uni/tn-mgmt/mgmtp-default/oob-default"/>

</fileRemotePath>
```

ステップ 2 REST API を使用して次のような XML を POST 送信し、オンデマンドのテクニカル サポート ファイルを生成します。

例：

```
<dbgexpTechSupOnD upgradeLogs="no" startTime="unspecified" name="Tech_Support_9-20-16"
  exportToController="no" endTime="unspecified" dn="uni/fabric/tsod-Tech_Support_9-20-16"
 descr=""
  compression="gzip" category="forwarding" adminSt="untriggered">
    <dbgexpRsExportDest tDn="uni/fabric/path-ToSupport"/>
    <dbgexpRsTsSrc tDn="topology/pod-1/node-102/sys"/>
    <dbgexpRsTsSrc tDn="topology/pod-1/node-103/sys"/>
    <dbgexpRsTsSrc tDn="topology/pod-1/node-101/sys"/>
    <dbgexpRsData tDn="uni/fabric/tscont"/>
</dbgexpTechSupOnD>
<fabricFuncP>
    <fabricCtrlrPGrp name="default">
        <fabricRsApplTechSupOnDemand tnDbgexpTechSupOnDName=" Tech_Support_9-20-16"/>
    </fabricCtrlrPGrp>
</fabricFuncP>
```

REST API を使用したスイッチ インベントリの検索

このセクションでは、REST API を使用してスイッチのモデルとシリアル番号を見つける方法について説明します

手順

次のようにスイッチ インベントリを見つけます。

例：

```
GET
https://192.0.20.123/api/node/mo/topology/pod-1.json?query-target=children&target-subtree-class=fabricNode
```

次の応答が返されます：

```
response:
  {
     "totalCount":"8",
     "imdata":
     [{
         "fabricNode":{
           "attributes":{
              "adSt":"on",
              "childAction":"",
              "delayedHeartbeat":"no",
              "dn":"topology/pod-1/node-103",
              "fabricSt":"active",
              "id":"103",
              "lcOwn":"local",
              "modTs":"2016-10-08T14:49:35.665+00:00",
              "model":"N9K-C9396PX",
              "monPolDn":"uni/fabric/monfab-default",
              "name":"leaf3",
              "nameAlias":"",
              "role":"leaf",
              "serial":"TEP-1-103",
```

REST API を使用した Cisco APIC の設定


```
          "status":"","uid":"0",
          "vendor":"Cisco Systems, Inc",
          "version":""}
         }
},{
    "fabricNode":{
      "attributes":{
        "adSt":"on",
        "childAction":"",
        "delayedHeartbeat":"no",
        "dn":"topology/pod-1/node-105",
        "fabricSt":"active",
        "id":"105",
        "lcOwn":"local",
        "modTs":"2016-10-08T14:47:52.011+00:00",
        "model":"N9K-C9508",
        "monPolDn":"uni/fabric/monfab-default",
        "name":"spine2",
        "nameAlias":"",
        "role":"spine",
        "serial":"TEP-1-105","status":"",
        "uid":"0",
        "vendor":"Cisco Systems, Inc",
        "version":""
      ...
  [TRUNCATED]
      ...
}
```

REST API を使用した Cisco APIC の設定

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。