Solucionar problemas de comandos CLI do estilo APIC NXOS

Contents

Introduction

Este documento descreve as etapas para depurar comandos executados a partir da CLI do APIC.

Prerequisites

Requirements

A Cisco recomenda que você tenha conhecimento destes tópicos:

- API REST DA ACI
- Modelo de objeto da ACI

O leitor deve ter um conhecimento anterior sobre como o funciona, bem como sobre como o processo DME registra suas mensagens.

Estes documentos explicam com mais detalhes sobre o ACI APIC e o modelo de objeto:

https://developer.cisco.com/docs/aci/

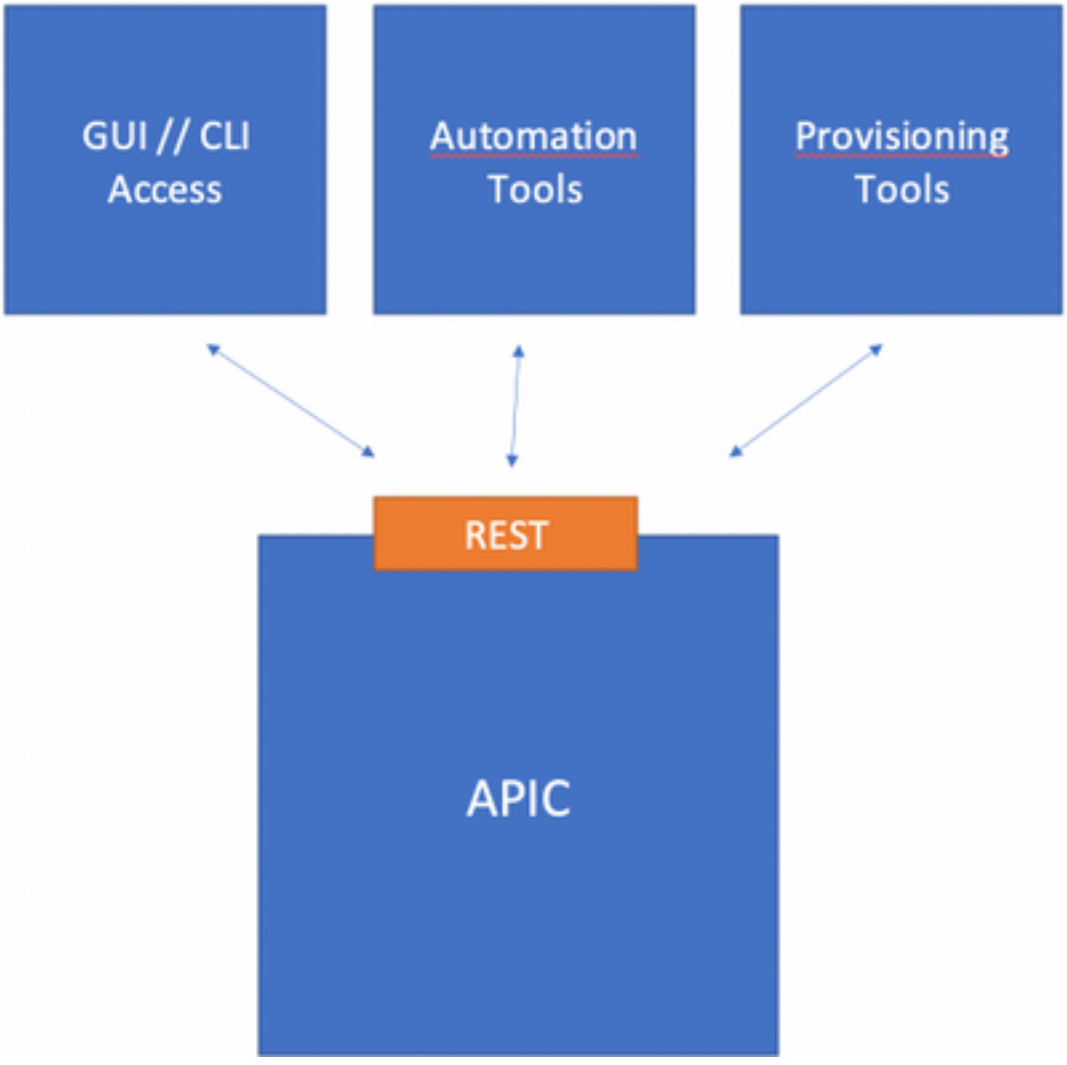
https://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/policy-model-guide/b-Cisco-ACI-Policy-Model-Guide.html

Informações de Apoio

O Cisco Application Policy Controller contém uma única API ascendente que é usada para o gerenciamento de políticas.

Toda interação baseada em políticas com um APIC se resume a uma interação de solicitação/resposta de API HTTP/S. Isso se aplica à GUI, aos scripts python personalizados e aos comandos CLI que existem nos APICs e Switches.

Esta imagem resume a forma como usuários e ferramentas interagem com a API do APICs



Todos os comandos **show** executados em um APIC chamam a CLI do **estilo NXOS.** Esses comandos acionam uma série de scripts python que criam as solicitações de API necessárias para reunir as informações solicitadas. Quando uma resposta é recebida, ela é analisada com python e então entregue ao usuário em um formato bonito.

NGINX

NGINX é um servidor web de código aberto. Cada APIC executa seu próprio processo NGINX, que serve à API RESTful. Em um APIC, o NGINX inclui registro através dos arquivos

/var/log/dme/log/nginx.bin.log e **/var/log/dme/log/access.log**.

O arquivo **nginx.bin.log** mostra os detalhes de todas as solicitações de API e interações DME.

O arquivo **access.log** registra cada solicitação de API tratada pelo NGINX.

Como todas as solicitações de API são solicitações HTTP, os códigos de resposta HTTP padrão podem ser referenciados para tratamento de chamadas de API NGINX:

- **200** código está OK
- Os códigos **4XX** são erros de cliente
- Os códigos **5XX** são erros de servidor, que é o APIC neste caso

O serviço de chamariz

O serviço Decoy serve uma chamada de API especial através de um módulo python hospedado fora do NGINX.

Este serviço:

1. Aceita o comando CLI solicitado
2. Identifica os scripts python necessários para criar as chamadas da API REST
3. Envia as chamadas de API para a API RESTful
4. Aceita a resposta
5. Formata a resposta
6. Atende a resposta formatada ao usuário.

Este serviço inclui estes arquivos de log:

- /var/log/dme/log/decoy.log
- /var/log/dme/log/decoy.error.log
- /var/log/dme/log/decoy_server.log

O arquivo decoy.log registra os comandos CLI que foram executados.

O arquivo **access.log** do nginx usa o formato "**POST /decoy/exec/cmd.cli HTTP/1.1**" junto com o código HTTP associado à solicitação. . O arquivo registra as chamadas da API REST do comando.

> **Note**: Os registros nginx e decoy mencionados são coletados no Techsupport do APIC 3of3.

Solucionar problemas da CLI do estilo NXOS

Rastrear o comando CLI através de registros

Executar o comando CLI

Para este exemplo, o comando "show controller" é emitido através do APIC CLI:

```
APIC-1# show controller
Fabric Name          : ACI-POD1
Operational Size     : 2
Cluster Size         : 3
Time Difference      : 0
Fabric Security Mode : PERMISSIVE

 ID    Pod   Address          In-Band IPv4     In-Band IPv6               OOB IPv4         OOB
IPv6                        Version             Flags  Serial Number     Health
 ----  ----  --------------  --------------  ------------------------  ---------------  -----
------------------------  ------------------  -----  ----------------  ------------------
 1*    1     10.0.0.1         0.0.0.0        fc00::1                   192.168.1.1
4.2(6h)             crva-  XXXXXXXXXX fully-fit
 2     1     10.0.0.2         0.0.0.0          fc00::1                   192.168.1.1
        4.2(6h)             crva-  XXXXXXXXXX       fully-fit
Flags - c:Commissioned | r:Registered | v:Valid Certificate | a:Approved | f/s:Failover
fail/success
(*)Current (~)Standby (+)AS
```

O comando retorna uma saída formatada.

Verifique decoy.log para execução de CMD

O **arquivo decoy.log** pode ser verificado para encontrar o CMD "**show controller**" que foi chamado:

```
APIC-1# tail /var/log/dme/log/decoy.log
...
...||AUTH COOKIE="XXXXXX"||...
...||CLI: {"option": "server", "loglevel": ["disable"], "cols": 171, "mode": [["exec"]], "port":
51719, "cli": ["show", "controller"]}||...
...||port: 51719||...
...||Mode: [[u'exec']]||...
...||Command: [u'show', u'controller']||...
...||CommandCompleter: add exec||...
...||CommandCompleter: add show||...
...||CommandCompleter: add controller||...
...||last tokens: ['show', 'controller']||...
...||modeCmd: Mode: exec, fulltree: False teminal context is : {'mode-module': 'yaci._cfgroot',
'module': 'show._controllers', 'inherited': False}||.
...||('_%CMD_TERM%_', {'mode-module': 'yaci._cfgroot', 'module': 'show._controllers',
'inherited': False}, {'prompt': '# ', 'mode': [[u'exec']]}, None)||...
...||terminal command module: {"mode-module": "yaci._cfgroot", "module": "show._controllers",
"inherited": false}|...
```

O CLI envia isso como um dicionário; podemos vê-lo na linha realçada. A versão formatada seria semelhante a esta:

```
{
   "option": "server",
   "loglevel": [
       "disable" <-- Can be modified to debug the interaction further.
   ],
   "cols": 171,
   "mode": [     <-- Command ran by admin
       [
           "exec"
       ]
   ],
   "port": 51719, <-- Random TCP port from session.
   "cli": [      <-- Actual command
       "show",
```

```
        "controller"
    ]
}
```

A linha com '_%CMD_TERM%_' mostra o comando executado, juntamente com o módulo usado, neste exemplo:

```
('_%CMD_TERM%_', {'mode-module': 'yaci._cfgroot', 'module': 'show._controllers', 'inherited':
False}, {'prompt': '# ', 'mode': [[u'exec']]}, None)
```

O módulo CLI deve convertê-lo em chamadas da API REST.

Verificar access.log para Chamadas de API

O arquivo **access.log** exibe as Chamadas de API recebidas após o decoy processar o CMD:

```
APIC-1# tail /var/log/dme/log/access.log
...
127.0.0.1 - - [24/May/2021:18:43:12 +0000] "POST /decoy/exec/cmd.cli HTTP/1.1" 200 0 "-"
"python-requests/2.7.0..."
127.0.0.1 - - [24/May/2021:18:43:19 +0000] "GET /api/mo/topology/pod-1/node-1/sys.xml HTTP/1.1"
200 1273 "-" "python-requests/2.7.0..."
127.0.0.1 - - [24/May/2021:18:43:19 +0000] "GET /api/mo/uni/fabsslcomm/ifmcertnode-1.xml
HTTP/1.1" 200 2391 "-" "python-requests/2.7.0..."
127.0.0.1 - - [24/May/2021:18:43:19 +0000] "GET /api/mo/uni/fabsslcomm/ifmcertnode-2.xml
HTTP/1.1" 200 2508 "-" "python-requests/2.7.0..."
127.0.0.1 - - [24/May/2021:18:43:19 +0000] "GET /api/mo/topology/pod-1/node-2/sys.xml HTTP/1.1"
200 1265 "-" "python-requests/2.7.0..."
```

Verifique nginx.bin.log

O *arquivo nginx.bin.log* inclui detalhes adicionais sobre todas as Solicitações de API tratadas e deve incluir informações sobre o payload recebido de vários DMEs antes de ser enviado de volta ao solicitante.

No mesmo arquivo, depois que **decoy/exec/cmd.cli** é chamado, há várias chamadas de API registradas:

```
admin@APIC-1:log> cat nginx.bin.log| grep "ifmcertnode|sys.xml"
17567||2021-05-24T18:43:19.817094383+00:00||nginx||DBG4||||Request received
/api/mo/topology/pod-1/node-1/sys.xml||../common/src/rest/./Rest.cc||67   bico 11.322
17567||2021-05-24T18:43:19.817184335+00:00||nginx||DBG4||||httpmethod=1; from 127.0.0.1;
url=/api/mo/topology/pod-1/node-1/sys.xml; url options=||../common/src/rest/./Request.cc||133

17567||2021-05-24T18:43:19.817409193+00:00||nginx||DBG4||||Request received
/api/mo/topology/pod-1/node-2/sys.xml||../common/src/rest/./Rest.cc||67
17567||2021-05-24T18:43:19.817466216+00:00||nginx||DBG4||||httpmethod=1; from 127.0.0.1;
url=/api/mo/topology/pod-1/node-2/sys.xml; url options=||../common/src/rest/./Request.cc||133

17567||2021-05-24T18:43:19.817589102+00:00||nginx||DBG4||||Request received
/api/mo/uni/fabsslcomm/ifmcertnode-1.xml||../common/src/rest/./Rest.cc||67
17567||2021-05-24T18:43:19.817641070+00:00||nginx||DBG4||||httpmethod=1; from 127.0.0.1;
url=/api/mo/uni/fabsslcomm/ifmcertnode-1.xml; url options=||../common/src/rest/./Request.cc||133

17567||2021-05-24T18:43:19.819268449+00:00||nginx||DBG4||||Request received
/api/mo/uni/fabsslcomm/ifmcertnode-2.xml||../common/src/rest/./Rest.cc||67
17567||2021-05-24T18:43:19.819340589+00:00||nginx||DBG4||||httpmethod=1; from 127.0.0.1;
url=/api/mo/uni/fabsslcomm/ifmcertnode-2.xml; url options=||../common/src/rest/./Request.cc||133
```

...

Executar novamente as chamadas de API individuais

Os access.logs contêm uma lista de exatamente quais chamadas de API foram enviadas para processamento.

O objetivo desta etapa é executar novamente cada chamada de API para determinar:

1. Há algum problema com a chamada API em si?
2. Há algo diferente entre uma chamada de API bem-sucedida e uma falha?

Por exemplo, uma das chamadas de API geradas a partir do comando "show controllers" é:

```
127.0.0.1 - - [24/May/2021:18:43:19 +0000] "GET /api/mo/topology/pod-1/node-1/sys.xml HTTP/1.1"
200 1273 "-" "python-requests/2.7.0..."
```

Via icurl

Esta solicitação de API pode ser executada novamente com o uso de icurl em um APIC:

```
icurl 'http://localhost:7777/



        '
```

Note: A porta 7777 é usada especificamente para permitir que o APIC faça consultas.

Com a chamada específica como exemplo:

```
APIC-1# bash
admin@APIC-1:~> icurl 'http://localhost:7777/api/mo/topology/pod-1/node-1/sys.xml'

<?xml version="1.0" encoding="UTF-8"?>
<imdata totalCount="1">
<topSystem address="10.0.0.1" bootstrapState="none" childAction="" dn="topology/pod-1/node-
1/sys" ... />
</imdata>
```

A quantidade e a complexidade das chamadas de API necessárias para cada comando CLI do NXOS podem ser muito diferentes. s. Em todos os casos, as consultas podem ser reproduzidas via icurl para validação de solicitação e resposta.

Via moquery

A partir das chamadas de API anteriores mostradas, o módulo analisa a saída do comando ‘show controller’ com as informações dos MOs solicitados.

```
admin@APIC-1:~> moquery -d topology/pod-1/node-1/sys -o xml
...
 <topSystem address="10.0.0.1" dn="topology/pod-1/node-1/sys" fabricDomain="ACI-POD1"
 id="1" inbMgmtAddr="0.0.0.0" inbMgmtAddr6=""
```

```
 oobMgmtAddr="192.168.1.1" oobMgmtAddr6="" podId="1" serial="..."
 state="in-service" version="4.2(6h)"/>


admin@APIC-1:~> moquery -d topology/pod-1/node-2/sys -o xml
...
 <topSystem address="10.0.0.2" dn="topology/pod-1/node-2/sys" fabricDomain="ACI-POD1"
 id="2" inbMgmtAddr="0.0.0.0" inbMgmtAddr6=""
 oobMgmtAddr="192.168.1.2" oobMgmtAddr6="" podId="1" serial="..."
 state="in-service" version="4.2(6h)"/>


admin@APIC-1:~> moquery -d uni/fabsslcomm/ifmcertnode-1  -o xml
...
 <pkiFabricNodeSSLCertificate nodeId="1" serialNumber="..." subject="/serialNumber=PID:APIC-
SERVER-M2 SN:..." .../>


admin@APIC-1:~> moquery -d uni/fabsslcomm/ifmcertnode-2  -o xml
...
 <pkiFabricNodeSSLCertificate nodeId="2" serialNumber="..." subject="/serialNumber=PID:APIC-
SERVER-M2 SN:..." .../>
```

Executar novamente o comando CLI via Python

Como mencionado, todos os comandos "show" são na verdade uma chamada à API REST para uma série de scripts python.

Devido a isso, um usuário pode chamar manualmente o script python que executa o cmd. O objetivo aqui é ver se Python fornece mais informações sobre por que um CMD específico tem problemas.

Para qualquer comando "**show**" que tenha um problema, chame o comando via Python para comparar um APIC bem-sucedido com um com falha:

```
apic1# ${PYTHON} -m pyclient.remote exec terminal ${COLUMNS} <some show command>
```

O exemplo executa:

```
apic1# ${PYTHON} -m pyclient.remote exec terminal ${COLUMNS} show switch
apic1# ${PYTHON} -m pyclient.remote exec terminal ${COLUMNS} show controller
```

Exemplo onde o script direto Python fornece informações adicionais de depuração em uma falha:

```
a-apic1# ${PYTHON} -m pyclient.remote exec terminal ${COLUMNS} show switch

Process Process-2:
Traceback (most recent call last):
File "/usr/lib64/python2.7/multiprocessing/process.py", line 267, in _bootstrap
self.run()
File "/usr/lib64/python2.7/multiprocessing/process.py", line 114, in run
self._target(*self._args, **self._kwargs)
File "/controller/yaci/execmode/show/_switch_nodes.py", line 75, in _systemQuery
mo = ctx.moDir.lookupByDn(dn)
File "/controller/ishell/cobra/mit/access.py", line 80, in lookupByDn
mos = self.query(dnQuery)
File "/controller/yaci/pyclient/local.py", line 36, in query
```

```
raise e
QueryError: Unable to deliver the message, Resolve timeout from (type/num/svc/shard) =
switch:205:4:0

ID Pod Address In-Band IPv4 In-Band IPv6 OOB IPv4 OOB IPv6 Version Flags Serial Number Name
---- ---- --------------- --------------- ------------------------ -------------- ------------
------------ ----------------- ----- ---------------- -----------------
101 1 10.0.40.65 192.168.2.231 :: 192.168.1.101 :: n9000-14.2(6o) aliv XXXXXXXXXXX leaf101
...
205 ...



206 2 10.0.156.65 192.168.2.236 :: 192.168.1.106 :: n9000-14.2(6l) aliv XXXXXXXXXXX leaf206
...

Flags - a:Active | l/s:Leaf/Spine | v:Valid Certificate | i:In-Service
```

Rastrear modificação de objeto via APIC CLI

Criação de objeto

O APIC utiliza **configure terminal** para modificar o MIT de uma forma semelhante ao NXOS.

Os resultados dessas operações podem ser vistos no arquivo **decoy.log**.

Exemplo de criação de VRF:

```
APIC-1# configure terminal
APIC-1(config)# tenant TestTn
APIC-1(config-tenant)# vrf context Test-CTX-1
APIC-1(config-tenant-vrf)#
```

Saída de **decoy.log**:

```
'18279||2021-05-25 22:52:11,877||decoy||INFO||AUTH
COOKIE="eyJhbGciOiJSUzI1NiIsImtpZCI6ImgycDE1NmdjYXIxMG50emx3YmR3MDFjNnV5dGEycXQzIiwidHlwIjoiand0
In0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVz
ZXJuYW1lIjoiYWRtaW4iLCJ1c2VyaWQiOjE1Mzc0LCJ1c2VyZmxhZ3MiOjQsImlhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIx
MjgyMjY2LCJzZXNzaW9uaWQiOiJUWW9pZGgyMFRIeUc5YldMUllhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVta
CbamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd281eaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM5lUmFxhZgYI2Lx8gg0P6sLoUydcShKKcUNgRmGWw
O64LH7rMEpzyCTapJBXdkzUhJ-zm98fmOy1oGGTesBteSWP_ksH14Xq411k1ebJ83sV4tL6-
FJLhcPNIKwqYJ87fqUWwZFZb5tY4JUJxrSnahKfwyidNXt5m8LCIc8pt-xbBtVihAFkAYBoXKI-
OYBrwg"||/mgmt/opt/controller/decoy/decoy/_app.py||32'
'18279||2021-05-25 22:52:11,878||decoy||INFO||CLI: {"option": "server", "loglevel": ["disable"],
"cols": 100, "mode": [["exec"], ["configure", "terminal"]], "port": 60003, "cli": ["tenant",
"TestTn"]}||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||88'
'18279||2021-05-25 22:52:11,878||decoy||INFO||port:
60003||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||107'
'18279||2021-05-25 22:52:11,879||decoy||INFO||Mode: [[u'exec'], [u'configure',
u'terminal']]||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||120'
'18279||2021-05-25 22:52:11,879||decoy||INFO||Command: [u'tenant',
u'TestTn']||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||121'
'18279||2021-05-25 22:52:11,879||decoy||DEBUG||CommandCompleter: add
exec||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'18279||2021-05-25 22:52:11,880||decoy||DEBUG||CommandCompleter: add
configure||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
```

'18279||2021-05-25 22:52:11,880||decoy||DEBUG||OptionKeyword: add
terminal||/mgmt/opt/controller/yaci/yaci/_completer.py||1036'
'11818||2021-05-25 22:52:11,910||decoy||DEBUG||CommandCompleter: add
tenant||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'11818||2021-05-25 22:52:11,911||decoy||DEBUG||ArgCompleter: add
TestTn||/mgmt/opt/controller/yaci/yaci/_completer.py||531'
'11818||2021-05-25 22:52:11,912||decoy||INFO||last tokens: ['tenant',
u'TestTn']||/mgmt/opt/controller/yaci/yaci/_ctx.py||617'
'11818||2021-05-25 22:52:11,912||decoy||INFO||TID: 98||CLI Command: 'tenant
TestTn'||/mgmt/opt/controller/yaci/yaci/_transaction.py||67'

'11818||2021-05-25 22:52:14,331||decoy||DEBUG||[commit]: <?xml version="1.0" **encoding="UTF-8"?>**

status='created,modified'></fvTenant>||/mgmt/opt/controller/yaci/yaci/_ctx.py||1024'

**'11818||2021-05-25 22:52:14,349||decoy||INFO||CLIENT-HEADERS: {'APIC-Client': u'tenant TestTn',
'Cookie': 'APIC-
cookie=eyJhbGciOiJSUzI1NiIsImtpZCI6ImgycDE1NmdjYXIxMG50emx3YmR3MDFjNnV5dGEycXQzIiwidHlwIjoiand0In0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVzZXJuYW1lIjoiYWRtaW4iLCJ1c2VyaWQiOjE1Mzc0LCJ1c2VyZmxhZ3MiOjQsImlhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIxMjgyMjY2LCJzZXNzaW9uaWQiOiJUWW9pZGgyMFRIeUc5YldMUllhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaCbamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd281eaSQPPWiT-ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM5lUmFxhZgYI2Lx8gg0P6sLoUydcShKKcUNgRmGWwO64LH7rMEpzyCTapJBXdkzUhJ-zm98fmOy1oGGTesBteSWP_ksH14Xq411k1ebJ83sV4tL6-FJLhcPNIKwqYJ87fqUWwZFZb5tY4JUJxrSnahKfwyidNXt5m8LCIc8pt-xbBtVihAFkAYBoXKI-OYBrwg',
'Client_Name': 'APIC-CLI', 'Request-Tag':
'tag0'}||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||31'
'11818||2021-05-25 22:52:14,350||decoy||INFO||Starting new HTTP connection (1):
127.0.0.1||/mgmt/opt/controller/decoy/decoy-env/lib/python2.7/site-
packages/requests/packages/urllib3/connectionpool.py||203'
'11818||2021-05-25 22:52:14,381||decoy||DEBUG||" POST /api/mo/uni/tn-TestTn.xml HTTP/1.1" 200
70||/mgmt/opt/controller/decoy/decoy-env/lib/python2.7/site-
packages/requests/packages/urllib3/connectionpool.py||383'**

**'18279||2021-05-25 22:52:14,383||decoy||DEBUG||(None, {}, {'prompt': '(config-tenant)# ',
'mode': [[u'exec'], [u'configure', u'terminal'], ['tenant', u'TestTn']]},
None)||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||144'**

**'18280||2021-05-25 22:52:48,569||decoy||INFO||AUTH
COOKIE="eyJhbGciOiJSUzI1NiIsImtpZCI6ImgycDE1NmdjYXIxMG50emx3YmR3MDFjNnV5dGEycXQzIiwidHlwIjoiand0In0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVzZXJuYW1lIjoiYWRtaW4iLCJ1c2VyaWQiOjE1Mzc0LCJ1c2VyZmxhZ3MiOjQsImlhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIxMjgyMjY2LCJzZXNzaW9uaWQiOiJUWW9pZGgyMFRIeUc5YldMUllhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaCbamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd281eaSQPPWiT-ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM5lUmFxhZgYI2Lx8gg0P6sLoUydcShKKcUNgRmGWwO64LH7rMEpzyCTapJBXdkzUhJ-zm98fmOy1oGGTesBteSWP_ksH14Xq411k1ebJ83sV4tL6-FJLhcPNIKwqYJ87fqUWwZFZb5tY4JUJxrSnahKfwyidNXt5m8LCIc8pt-xbBtVihAFkAYBoXKI-OYBrwg"||/mgmt/opt/controller/decoy/decoy/_app.py||32'
'18280||2021-05-25 22:52:48,571||decoy||INFO||CLI: {"option": "server", "loglevel": ["disable"],
"cols": 100, "mode": [["exec"], ["configure", "terminal"], ["tenant", "TestTn"]], "port": 60821,
"cli": ["vrf", "context", "Test-CTX-
1"]}||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||88'**

'18280||2021-05-25 22:52:48,571||decoy||INFO||port:

60821||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||107'
'18280||2021-05-25 22:52:48,572||decoy||INFO||Mode: [[u'exec'], [u'configure', u'terminal'],
[u'tenant', u'TestTn']]||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||120'
'18280||2021-05-25 22:52:48,572||decoy||INFO||Command: [u'vrf', u'context', u' Test-CTX-
1']||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||121'

'18280||2021-05-25 22:52:48,572||decoy||DEBUG||CommandCompleter: add
exec||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'18280||2021-05-25 22:52:48,572||decoy||DEBUG||CommandCompleter: add
configure||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'18280||2021-05-25 22:52:48,573||decoy||DEBUG||OptionKeyword: add
terminal||/mgmt/opt/controller/yaci/yaci/_completer.py||1036'
'18280||2021-05-25 22:52:48,574||decoy||DEBUG||CommandCompleter: add
tenant||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'18280||2021-05-25 22:52:48,574||decoy||DEBUG||ArgCompleter: add
TestTn||/mgmt/opt/controller/yaci/yaci/_completer.py||531'
'13482||2021-05-25 22:52:48,599||decoy||DEBUG||CommandCompleter: add
vrf||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'13482||2021-05-25 22:52:48,600||decoy||DEBUG||Keyword: add
context||/mgmt/opt/controller/yaci/yaci/_completer.py||983'
'13482||2021-05-25 22:52:48,601||decoy||DEBUG||ArgCompleter: add Test-CTX-
1||/mgmt/opt/controller/yaci/yaci/_completer.py||531'
'13482||2021-05-25 22:52:48,602||decoy||INFO||last tokens: ['vrf', 'context', u'Test-CTX-
1']||/mgmt/opt/controller/yaci/yaci/_ctx.py||617'
'13482||2021-05-25 22:52:48,602||decoy||INFO||TID: 105||CLI Command: 'vrf context Test-CTX-
1'||/mgmt/opt/controller/yaci/yaci/_transaction.py||67'

'13482||2021-05-25 22:52:51,393||decoy||DEBUG||[commit]: <?xml version="1.0" encoding="UTF-8"?>

status='created,modified' name='

Test-CTX-1'></fvCtx>||/mgmt/opt/controller/yaci/yaci/_ctx.py||1024'

'13482||2021-05-25 22:52:51,405||decoy||INFO||CLIENT-HEADERS: {'APIC-Client': u'tenant TestTn;
vrf context Test-CTX-1', 'Cookie': 'APIC-
cookie=eyJhbGciOiJSUzI1NiIsImtpZCI6ImgycDE1NmdjYXIxMG50emx3YmR3MDFjNnV5dGEycXQzIiwidHlwIjoiand0I
n0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVzZ
XJuYW1lIjoiYWRtaW4iLCJ1c2VyaWQiOjE1Mzc0LCJ1c2VyZmxhZ3MiOjQsImlhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIxM
jgyMjY2LCJzZXNzaW9uaWQiOiJUWW9pZGgyMFRIeUc5YldMUllhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaC
bamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd281eaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM5lUmFxhZgYI2Lx8gg0P6sLoUydcShKKcUNgRmGWw
O64LH7rMEpzyCTapJBXdkzUhJ-zm98fmOy1oGGTesBteSWP_ksH14Xq411k1ebJ83sV4tL6-
FJLhcPNIKwqYJ87fqUWwZFZb5tY4JUJxrSnahKfwyidNXt5m8LCIc8pt-xbBtVihAFkAYBoXKI-OYBrwg',
'Client_Name': 'APIC-CLI', 'Request-Tag':
'tag0'}||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||31'
'13482||2021-05-25 22:52:51,407||decoy||INFO||Starting new HTTP connection (1):
127.0.0.1||/mgmt/opt/controller/decoy/decoy-env/lib/python2.7/site-
packages/requests/packages/urllib3/connectionpool.py||203'
'13482||2021-05-25 22:52:51,452||decoy||DEBUG||" POST /api/mo/uni/tn-TestTn/ctx-Test-CTX-1.xml
HTTP/1.1" 200 70||/mgmt/opt/controller/decoy/decoy-env/lib/python2.7/site-
packages/requests/packages/urllib3/connectionpool.py||383'

```
'18280||2021-05-25 22:52:51,455||decoy||DEBUG||(None, {}, {'prompt': '(config-tenant-vrf)# ',
'mode': [[u'exec'], [u'configure', u'terminal'], [u'tenant', u'TestTn'], ['vrf', 'context',
u'Test-CTX-1']]}, None)||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||144'
```

Observe os objetos fvTenant e fvCtx definidos com status="created,modified".Exclusão de ObjetoPodemos revisar as operações REST envolvidas quando qualquer MO é excluído.Exemplo de exclusão de VRF:

```
APIC-1# configure terminal
APIC-1(config)# tenant TestTn
APIC-1(config-tenant)# no vrf context Test-CTX-1
APIC-1(config-tenant-vrf)#
```

Saída de decoy.log:

```
'18279||2021-05-25 23:32:40,009||decoy||INFO||AUTH
COOKIE="eyJhbGciOiJSUzI1NiIsImtpZCI6ImgycDE1NmdjYXIxMG50emx3YmR3MDFjNnV5dGEycXQzIiwidHlwIjoiand0
In0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVz
ZXJuYW1lIjoiYWRtaW4iLCJ1c2VyaWQiOjE1Mzc0LCJ1c2VyZmxhZ3MiOjQsImlhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIx
MjgyMjY2LCJzZXNzaW9uaWQiOiJUWW9pZGgyMFRIeUc5Y1dMU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVta
CbamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd281eaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM51UmFxhZgYI2Lx8gg0P6sLoUydcShKKcUNgRmGWw
O64LH7rMEpzyCTapJBXdkzUhJ-zm98fmOy1oGGTesBteSWP_ksH14Xq411k1ebJ83sV4tL6-
FJLhcPNIKwqYJ87fqUWwZFZb5tY4JUJxrSnahKfwyidNXt5m8LCIc8pt-xbBtVihAFkAYBoXKI-
OYBrwg"||/mgmt/opt/controller/decoy/decoy/_app.py||32'
'18279||2021-05-25 23:32:40,010||decoy||INFO||CLI: {"option": "server", "loglevel": ["disable"],
"cols": 100, "mode": [["exec"], ["configure", "terminal"]], "port": 32789, "cli": ["tenant",
"TestTn"]}||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||88'
'18279||2021-05-25 23:32:40,011||decoy||INFO||port:
32789||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||107'
'18279||2021-05-25 23:32:40,012||decoy||INFO||Mode: [[u'exec'], [u'configure',
u'terminal']]||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||120'
'18279||2021-05-25 23:32:40,012||decoy||INFO||Command: [u'tenant',
u'TestTn']||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||121'
'18279||2021-05-25 23:32:40,012||decoy||DEBUG||CommandCompleter: add
exec||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'18279||2021-05-25 23:32:40,013||decoy||DEBUG||CommandCompleter: add
configure||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'18279||2021-05-25 23:32:40,013||decoy||DEBUG||OptionKeyword: add
terminal||/mgmt/opt/controller/yaci/yaci/_completer.py||1036'
'27163||2021-05-25 23:32:40,042||decoy||DEBUG||CommandCompleter: add
tenant||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'27163||2021-05-25 23:32:40,043||decoy||DEBUG||ArgCompleter: add
TestTn||/mgmt/opt/controller/yaci/yaci/_completer.py||531'
'27163||2021-05-25 23:32:40,043||decoy||INFO||last tokens: ['tenant',
u'TestTn']||/mgmt/opt/controller/yaci/yaci/_ctx.py||617'
'27163||2021-05-25 23:32:40,044||decoy||INFO||TID: 106||CLI Command: 'tenant
TestTn'||/mgmt/opt/controller/yaci/yaci/_transaction.py||67'

'27163||2021-05-25 23:32:42,467||decoy||DEBUG||[commit]: <?xml version="1.0" encoding="UTF-8"?>
<fvTenant name='TestTn'
status='created,modified'></fvTenant>||/mgmt/opt/controller/yaci/yaci/_ctx.py||1024'
'27163||2021-05-25 23:32:42,485||decoy||INFO||CLIENT-HEADERS: {'APIC-Client': u'tenant TestTn',
'Cookie': 'APIC-
cookie=eyJhbGciOiJSUzI1NiIsImtpZCI6ImgycDE1NmdjYXIxMG50emx3YmR3MDFjNnV5dGEycXQzIiwidHlwIjoiand0I
n0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVzZ
XJuYW1lIjoiYWRtaW4iLCJ1c2VyaWQiOjE1Mzc0LCJ1c2VyZmxhZ3MiOjQsImlhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIxM
jgyMjY2LCJzZXNzaW9uaWQiOiJUWW9pZGgyMFRIeUc5Y1dMU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaC
bamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd281eaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM51UmFxhZgYI2Lx8gg0P6sLoUydcShKKcUNgRmGWw
O64LH7rMEpzyCTapJBXdkzUhJ-zm98fmOy1oGGTesBteSWP_ksH14Xq411k1ebJ83sV4tL6-
FJLhcPNIKwqYJ87fqUWwZFZb5tY4JUJxrSnahKfwyidNXt5m8LCIc8pt-xbBtVihAFkAYBoXKI-OYBrwg',
```

'Client_Name': 'APIC-CLI', 'Request-Tag':
'tag0'}||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||31'
'27163||2021-05-25 23:32:42,487||decoy||INFO||Starting new HTTP connection (1):
127.0.0.1||/mgmt/opt/controller/decoy/decoy-env/lib/python2.7/site-
packages/requests/packages/urllib3/connectionpool.py||203'
'27163||2021-05-25 23:32:42,517||decoy||DEBUG||"POST /api/mo/uni/tn-TestTn.xml HTTP/1.1" 200
70||/mgmt/opt/controller/decoy/decoy-env/lib/python2.7/site-
packages/requests/packages/urllib3/connectionpool.py||383'
'18279||2021-05-25 23:32:42,520||decoy||DEBUG||(None, {}, {'prompt': '(config-tenant)# ',
'mode': [[u'exec'], [u'configure', u'terminal'], ['tenant', u'TestTn']]},
None)||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||144'

'18280||2021-05-25 23:32:53,702||decoy||INFO||AUTH
COOKIE="eyJhbGciOiJSUzI1NiIsImtpZCI6ImgycDE1NmdjYXIxMG50emx3YmR3MDFjNnV5dGEycXQzIiwidH1wIjoiand0
In0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVz
ZXJuYW1lIjoiYWRtaW4iLCJ1c2VyaWQiOjE1Mzc0LCJ1c2VyZmxhZ3MiOjQsImlhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIx
MjgyMjY2LCJzZXNzaW9uaWQiOiJUWW9pZGgyMFRIeUc5Y1dMU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVta
CbamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd281eaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM5lUmFxhZgYI2Lx8gg0P6sLoUydcShKKcUNgRmGWw
O64LH7rMEpzyCTapJBXdkzUhJ-zm98fmOy1oGGTesBteSWP_ksH14Xq411k1ebJ83sV4tL6-
FJLhcPNIKwqYJ87fqUWwZFZb5tY4JUJxrSnahKfwyidNXt5m8LCIc8pt-xbBtVihAFkAYBoXKI-
OYBrwg"||/mgmt/opt/controller/decoy/decoy/_app.py||32'
'18280||2021-05-25 23:32:53,704||decoy||INFO||CLI: {"option": "server", "loglevel": ["disable"],
"cols": 100, "mode": [["exec"], ["configure", "terminal"], ["tenant", "TestTn"]], "port": 32805,
"cli": ["no", "vrf", "context", "Test-CTX-
1"]}||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||88'
'18280||2021-05-25 23:32:53,704||decoy||INFO||port:
32805||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||107'
'18280||2021-05-25 23:32:53,705||decoy||INFO||Mode: [[u'exec'], [u'configure', u'terminal'],
[u'tenant', u'TestTn']]||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||120'
'18280||2021-05-25 23:32:53,705||decoy||INFO||Command: [u'no', u'vrf', u'context', u'Test-CTX-
1']||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||121'
'18280||2021-05-25 23:32:53,705||decoy||DEBUG||CommandCompleter: add
exec||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'18280||2021-05-25 23:32:53,706||decoy||DEBUG||CommandCompleter: add
configure||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'18280||2021-05-25 23:32:53,706||decoy||DEBUG||OptionKeyword: add
terminal||/mgmt/opt/controller/yaci/yaci/_completer.py||1036'
'18280||2021-05-25 23:32:53,708||decoy||DEBUG||CommandCompleter: add
tenant||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'18280||2021-05-25 23:32:53,708||decoy||DEBUG||ArgCompleter: add
TestTn||/mgmt/opt/controller/yaci/yaci/_completer.py||531'
'27771||2021-05-25 23:32:53,736||decoy||DEBUG||CommandCompleter: add
vrf||/mgmt/opt/controller/yaci/yaci/_completer.py||255'
'27771||2021-05-25 23:32:53,737||decoy||DEBUG||Keyword: add
context||/mgmt/opt/controller/yaci/yaci/_completer.py||983'
'27771||2021-05-25 23:32:53,738||decoy||DEBUG||ArgCompleter: add Test-CTX-
1||/mgmt/opt/controller/yaci/yaci/_completer.py||531'
'27771||2021-05-25 23:32:53,738||decoy||INFO||last tokens: ['no', 'vrf', 'context', u'Test-CTX-
1']||/mgmt/opt/controller/yaci/yaci/_ctx.py||617'
'27771||2021-05-25 23:32:53,738||decoy||INFO||TID: 113||CLI Command: 'no vrf context Test-CTX-
1'||/mgmt/opt/controller/yaci/yaci/_transaction.py||67'
'27771||2021-05-25 23:32:53,740||decoy||INFO||CLIENT-HEADERS: {'APIC-Client': u'tenant TestTn;
no vrf context Test-CTX-1', 'Cookie': 'APIC-
cookie=eyJhbGciOiJSUzI1NiIsImtpZCI6ImgycDE1NmdjYXIxMG50emx3YmR3MDFjNnV5dGEycXQzIiwidH1wIjoiand0I
n0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVzZ
XJuYW1lIjoiYWRtaW4iLCJ1c2VyaWQiOjE1Mzc0LCJ1c2VyZmxhZ3MiOjQsImlhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIxM
jgyMjY2LCJzZXNzaW9uaWQiOiJUWW9pZGgyMFRIeUc5Y1dMU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaC
bamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd281eaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM5lUmFxhZgYI2Lx8gg0P6sLoUydcShKKcUNgRmGWw
O64LH7rMEpzyCTapJBXdkzUhJ-zm98fmOy1oGGTesBteSWP_ksH14Xq411k1ebJ83sV4tL6-
FJLhcPNIKwqYJ87fqUWwZFZb5tY4JUJxrSnahKfwyidNXt5m8LCIc8pt-xbBtVihAFkAYBoXKI-OYBrwg',
'Client_Name': 'APIC-CLI', 'Request-Tag':
'tag0'}||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||31'

```
'27771||2021-05-25 23:32:53,743||decoy||INFO||Starting new HTTP connection (1):
127.0.0.1||/mgmt/opt/controller/decoy/decoy-env/lib/python2.7/site-
packages/requests/packages/urllib3/connectionpool.py||203'
'27771||2021-05-25 23:32:53,750||decoy||DEBUG||"GET /api/mo/uni/tn-TestTn.xml?target-subtree-
class=l3extRsEctx&query-target-filter=eq(l3extRsEctx.tnFvCtxName,%22Test-CTX-1%22)&query-
target=subtree HTTP/1.1" 200 70||/mgmt/opt/controller/decoy/decoy-env/lib/python2.7/site-
packages/requests/packages/urllib3/connectionpool.py||383'
'27771||2021-05-25 23:32:53,753||decoy||DEBUG||dnListLen:
0||/mgmt/opt/controller/yaci/lib/utils/l3ext.py||97'

'27771||2021-05-25 23:32:56,375||decoy||DEBUG||[commit]: <?xml version="1.0" encoding="UTF-8"?>
<fvCtx status='deleted' name='Test-CTX-
1'></fvCtx>||/mgmt/opt/controller/yaci/yaci/_ctx.py||1024'
'27771||2021-05-25 23:32:56,386||decoy||INFO||CLIENT-HEADERS: {'APIC-Client': u'tenant TestTn;
no vrf context Test-CTX-1', 'Cookie': 'APIC-
cookie=eyJhbGciOiJSUzI1NiIsImtpZCI6ImgycDE1NmdjYXIxMG50emx3YmR3MDFjNnV5dGEycXQzIiwidHlwIjoiand0I
n0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVzZ
XJuYW1lIjoiYWRtaW4iLCJ1c2VyaWQiOjE1Mzc0LCJ1c2VyZmxhZ3MiOjQsImlhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIxM
jgyMjY2LCJzZXNzaW9uaWQiOiJUWW9pZGgyMFRIeUc5Y1dMU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaC
bamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd281eaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM5lUmFxhZgYI2Lx8gg0P6sLoUydcShKKcUNgRmGWw
O64LH7rMEpzyCTapJBXdkzUhJ-zm98fmOy1oGGTesBteSWP_ksH14Xq411k1ebJ83sV4tL6-
FJLhcPNIKwqYJ87fqUWwZFZb5tY4JUJxrSnahKfwyidNXt5m8LCIc8pt-xbBtVihAFkAYBoXKI-OYBrwg',
'Client_Name': 'APIC-CLI', 'Request-Tag':
'tag0'}||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||31'
'27771||2021-05-25 23:32:56,426||decoy||DEBUG||"POST /api/mo/uni/tn-TestTn/ctx-Test-CTX-1.xml
HTTP/1.1" 200 70||/mgmt/opt/controller/decoy/decoy-env/lib/python2.7/site-
packages/requests/packages/urllib3/connectionpool.py||383'
'18280||2021-05-25 23:32:56,428||decoy||DEBUG||(None, {}, {'prompt': '(config-tenant)# ',
'mode': [[u'exec'], [u'configure', u'terminal'], [u'tenant', u'TestTn']]},
None)||/mgmt/opt/controller/decoy/apps/execserver/execapp.py||144'
```

Observe o conjunto de objetos fvCtx com status="deleted".

Referências e links úteis

- Programabilidade da ACI
- Guia de modelos de política da Cisco ACI
- Suporte da Cisco ACI para limite de taxa do NGINX

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.