

Troubleshoot APIC NXOS Style CLI Commands

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Background Information](#)

[NGINX](#)

[The Decoy Service](#)

[Troubleshoot the NXOS Style CLI](#)

[Trace the CLI Command through Logs](#)

[Run the CLI Command](#)

[Check decoy.log for CMD Execution](#)

[Check access.log for API Calls](#)

[Check nginx.bin.log](#)

[Re-run the Individual API Calls](#)

[Via icurl](#)

[Via moquery](#)

[Re-run the CLI Command via Python](#)

[Trace Object Modification via APIC CLI](#)

[Object Creation](#)

[Object Deletion](#)

[References & Useful links](#)

Introduction

This document describes steps to debug commands run from the APIC CLI.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- ACI REST API
- ACI Object Model

The reader must have a previous knowledge on how the works, also how the DME process logs their messages.

These documents explain in more detail about the ACI APIC and Object model:

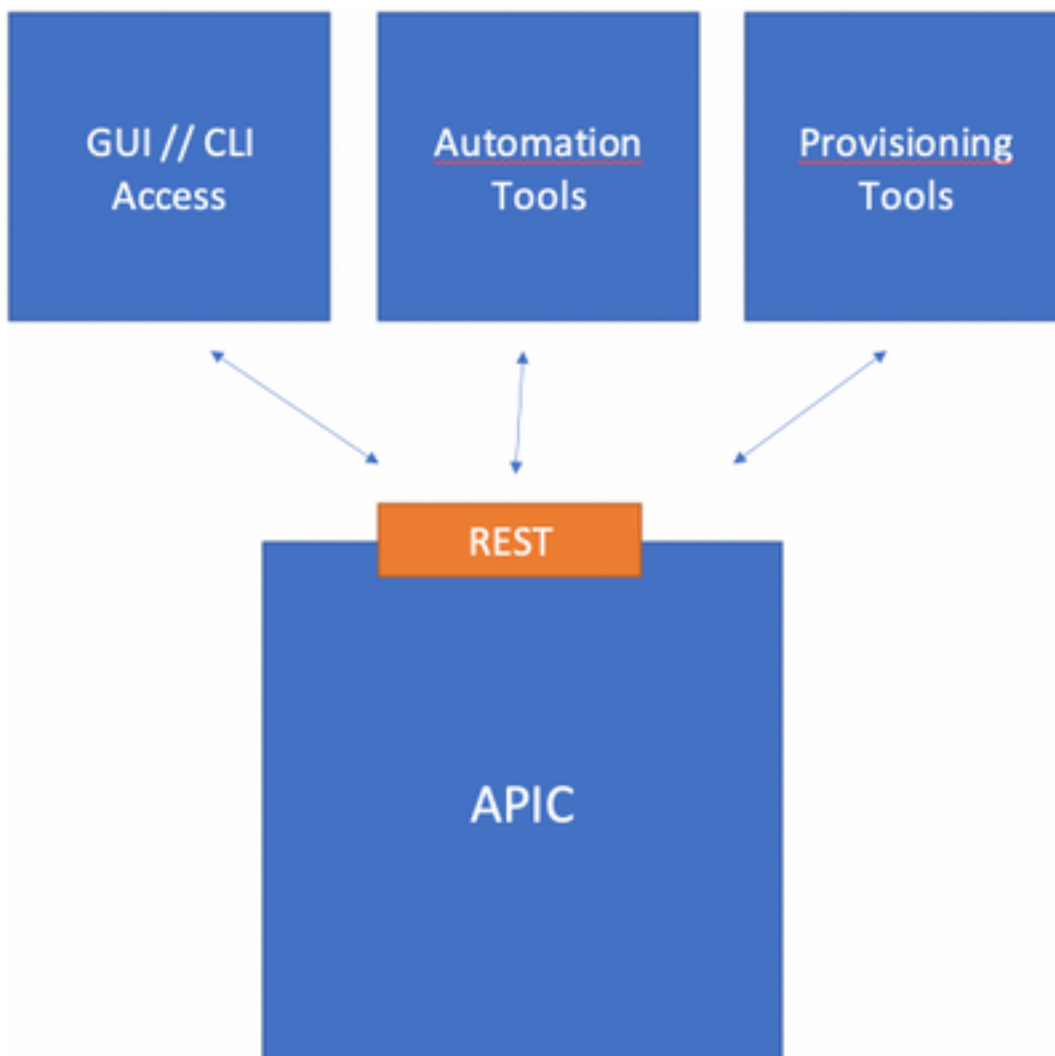
<https://developer.cisco.com/docs/aci/>

Background Information

The Cisco Application policy Controller contains a single Northbound API which is used for Policy management.

Every policy based interaction with an APIC boils down to an HTTP/S API Request/Response interaction. This holds true for the GUI, custom python scripts and the CLI Commands that exist on both the APICs and Switches.

This Image summarizes the way users and tools interact with the APICs API



All **show** commands run on an APIC invoke the **NXOS Style CLI**. These commands trigger a series of python scripts which build out the API requests required to gather the information requested. When a response is received, it is parsed with python then delivered to the user in a pretty format.

NGINX

NGINX is an open source web server. Each APIC runs its own NGINX process which serves the RESTful API. On an APIC, NGINX includes logging via both the `/var/log/dme/log/nginx.bin.log`

and **/var/log/dme/log/access.log** files.

The **nginx.bin.log** file shows the details of all API requests and DME interactions.

The **access.log** file logs every API request handled by NGINX.

As all API requests are HTTP requests, standard HTTP response codes can be referenced for NGINX API Call handling:

- **200** code is OK
- **4XX** codes are Client Errors
- **5XX** Codes are Server Errors, which is the APIC in this case

The Decoy Service

The Decoy service serves a special API call via a python module hosted off of NGINX.

This service:

1. Accepts the requested CLI command
2. Identifies the python scripts required to build out the REST API calls
3. Sends the API calls to the RESTful API
4. Accepts the response
5. Formats the response
6. Serves the formatted response to the user.

This service includes these log files:

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

The decoy.log file registers the CLI Commands that were run.

The **access.log** file from nginx uses the “**POST /decoy/exec/cmd.cli HTTP/1.1**” format along with the HTTP code associated to the request. . The file logs the REST API calls from the command.

Note: The nginx and decoy logs mentioned are collected within the APIC 3of3 Techsupport.

Troubleshoot the NXOS Style CLI

Trace the CLI Command through Logs

Run the CLI Command

For this example, the “show controller” is issued via the 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- XXXXXXXXXXXX fully-fit
2       1    10.0.0.2        0.0.0.0          fc00::1          192.168.1.1
          4.2(6h)          crva- XXXXXXXXXXXX          fully-fit
Flags - c:Commissioned | r:Registered | v:Valid Certificate | a:Approved | f/s:Failover
fail/success
(*)Current (~)Standby (+)AS
```

The command returns a pretty formatted output.

Check decoy.log for CMD Execution

The **decoy.log** file can be checked to find the **"show controller"** CMD that was invoked:

```
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 terminal 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}||...
```

The CLI sends this as a dictionary, we can see it from the highlighted line. The formatted version would look like this:

```
{
  "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"
  ]
}
```

```
}
```

The line with `'_%CMD_TERM%_'` shows the command executed, along with the module used, in this example:

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

The CLI module must then translate it to REST API calls.

Check access.log for API Calls

The `access.log` file displays the API Calls received after decoy processes the 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..."
```

Check nginx.bin.log

The `nginx.bin.log` file includes additional details on all API Requests handled and must include information about the payload received from various DMEs before it is sent back to the requester.

In the same file, after `decoy/exec/cmd.cli` is called, there are several API calls logged:

```
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
...
```

Re-run the Individual API Calls

The access.logs contain a list of exactly which API calls were sent for processing.

The goal of this step is to re-run each API call to determine:

1. Is there an issue with the API call itself?
2. Is there something different about a successful API call vs a failed one?

For example, one of the API calls generated from the "show controllers" command is:

```
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

This API Request can be re-run with the use icurl on an APIC:

```
icurl 'http://localhost:7777/<API-CALL-from-ACCESS.log>'
```

Note: Port 7777 is specifically used to allow the APIC to query itself.

With the specific call as an example:

```
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>
```

The amount and complexity of API calls required for each NXOS CLI command can differ greatly. s. In all cases, the queries can be re-played via icurl for request and response validation.

Via moquery

From the previous API calls shown, the module parses the output for the 'show controller' command with the information from the MOs requested.

```
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:..." .../>
```

Re-run the CLI Command via Python

As mentioned, all the "show" commands are actually a REST API call to a series of python scripts.

Due to this, a user can technically manually invoke the python script which runs the cmd. The goal here is to see if Python gives more information on why a specific CMD has issues.

For any "show" command that has an issue, invoke the command via Python to compare a successful vs a failed APIC:

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

Example runs:

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

Example where the direct Python script gives additional debug information on a failure:

```
a-apicl# ${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 XXXXXXXXXXXX leaf101
...
205                                     ...<empty>...
al-v          XXXXXXXXXXXX          leaf205
206 2 10.0.156.65 192.168.2.236 :: 192.168.1.106 :: n9000-14.2(6l) aliv XXXXXXXXXXXX leaf206
...
```

Trace Object Modification via APIC CLI

Object Creation

The APIC utilizes **configure terminal** to modify the MIT in an NXOS-Like fashion.

The results from these operations can be seen in the **decoy.log** file.

VRF Creation Example:

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

Output from decoy.log:

```
'18279| |2021-05-25 22:52:11,877| |decoy| |INFO| |AUTH
COOKIE="eyJhbGciOiJSUzI1NiIsImtpZCI6ImgyYmR3MDFjNnV5dGEycXQzIiwidHlwIjoianD0
In0.eyJyYmFjIjpbeyJkb2lhaW4iOiJhbGwiLCJyb2xlcl1iOiJEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVz
ZXJ1YWllIjoiYWRtaW4iLCJlc2VyaWQiOiJlMzc0LCJlc2VyZmxhZ3MiOiJQsImldhdi6MTYyMTI4MTY2NiwiZXhwIjoxNjIx
MjgyMjY2LCJzZXNzaW9uaWQiOiJUUW9pZGgyMFRlUc5YldMU1lhb0FnPT0ifQ.ksnCeOxnrNQeuaQnmpauUG_eja70nVta
CbamxFaBl1LkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd28leaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM5lUmFxbZgYI2Lx8gg0P6sLoUydcShKKcUNGmGWw
064LH7rMEpzyCTapJBXdkzUhJ-zm98fmOy1oGGTesBteSWP_ksH14Xq411klebJ83sV4tL6-
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"?>
<fvTenant name='TestTn'
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=eyJhbGciOiJSUzI1NiIsImtpZCI6ImgyYmR3MDFjNnV5dGEycXQzIiwidHlwIjoianD0I
```



```
n0.eyJyYmFjIjpbeyJkb2lhaW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVzZ
XJuYWllIjoIYWRtaW4iLCJlc2VyaWQiOjE1Mzc0LCJlc2VyZmxhZ3MiOjQsImh0LnVzZm9uYydcShkKcUNGRmGWw
jgyMjY2LCJzZXNzaW9uaWQiOiJUUW9pZGgyMFRlEuc5YldMU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaC
bamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd28leaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHJuvXpPctDsBYi9Ym5lUmFhxZgYI2Lx8gg0P6sLoUydcShkKcUNGRmGWw
064LH7rMEpyCTapJBXdKzUhJ-zm98fmOyloGGTebTebSWP_ksH14Xq411klebJ83sV4tL6-
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="eyJhbGciOiJSUzI1NiIsImtpZCI6ImgyYmFjIjpbeyJkb2lhaW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVz
Zm9uYydcShkKcUNGRmGWwZm9uYydcShkKcUNGRmGWwZm9uYydcShkKcUNGRmGWwZm9uYydcShkKcUNGRmGWwZm9uYydcShkKcUNGRmGWwZm9uYydcShkKcUNGRmGWw
MjY2LCJzZXNzaW9uaWQiOiJUUW9pZGgyMFRlEuc5YldMU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaC
bamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd28leaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHJuvXpPctDsBYi9Ym5lUmFhxZgYI2Lx8gg0P6sLoUydcShkKcUNGRmGWw
064LH7rMEpyCTapJBXdKzUhJ-zm98fmOyloGGTebTebSWP_ksH14Xq411klebJ83sV4tL6-
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"?>
<fvCtx 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=eyJhbGciOiJSUzI1NiIsImtpZCI6ImgyYDE1NmMjYXlMG50emx3YmR3MDFjNnV5dGEycXQzIiwidHlwIjoian0I
n0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVYI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVzZ
XJuYwllIjoieYWRtaW4iLCJlc2VyaWQiOjE1Mzc0LCJlc2VyZmxhZ3MiOjQsImldCI6MTYyMTI4MTY2NiwiZXhwIjojNjIxM
jgyMjY2LjZzZXNzaW9uaWQiOiJUUW9pZGgyMFRlUc5YldMU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaC
bamxFab1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd28leaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM5lUmFhxZgYI2Lx8gg0P6sLoUydcShKKcUNGmGWw
064LH7rMEpzyCTapJBXdkzUhJ-zm98fmOyloGGTesBteSWP_ksH14Xq411klebJ83sV4tL6-
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 the fvTenant and fvCtx objects both set with **status="created,modified"**.

Object Deletion

We can review the REST operations involved when any MO gets deleted.

VRF Deletion example:

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

Output from **decoy.log**:

```
'18279||2021-05-25 23:32:40,009||decoy||INFO||AUTH
COOKIE="eyJhbGciOiJSUzI1NiIsImtpZCI6ImgyYDE1NmMjYXlMG50emx3YmR3MDFjNnV5dGEycXQzIiwidHlwIjoian0I
In0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVYI6MX1dLCJpc3MiOiJBQ0kgQVBJQyIsInVzZ
XJuYwllIjoieYWRtaW4iLCJlc2VyaWQiOjE1Mzc0LCJlc2VyZmxhZ3MiOjQsImldCI6MTYyMTI4MTY2NiwiZXhwIjojNjIxM
jgyMjY2LjZzZXNzaW9uaWQiOiJUUW9pZGgyMFRlUc5YldMU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaC
bamxFab1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd28leaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM5lUmFhxZgYI2Lx8gg0P6sLoUydcShKKcUNGmGWw
064LH7rMEpzyCTapJBXdkzUhJ-zm98fmOyloGGTesBteSWP_ksH14Xq411klebJ83sV4tL6-
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=eyJhbGciOiJSUzI1NiIsImtpZCI6ImgyYUc5YldmU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaC  
n0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgVVBjQyIsInVzZ  
XJuYVllIjoiYWRtaW4iLCJlc2VyaWQiOiJlMzc0LCJlc2VyZmxhZ3MiOiJQsImldhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIxM  
jgyMjY2LCJzZXNzaW9uaWQiOiJUUWw9pZGgyMFRlUc5YldmU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaC  
bamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VrauKwW8Fztd281leaSQPPWiT-  
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFUHVjvXpPctDsBYi9Ym5lUmFhxZgYI2Lx8gg0P6sLoUydcSHkKcUNGmGWw  
064LH7rMEpzyCTapJBXdkzUhJ-zm98fmOyloGGTesBteSWP_ksH14Xq411klebJ83sV4tL6-  
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="eyJhbGciOiJSUzI1NiIsImtpZCI6ImgyYUc5YldmU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaC  
In0.eyJyYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOjEsInJvbGVzVyI6MX1dLCJpc3MiOiJBQ0kgVVBjQyIsInVzZ  
ZXJuYVllIjoiYWRtaW4iLCJlc2VyaWQiOiJlMzc0LCJlc2VyZmxhZ3MiOiJQsImldhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIxM  
jgyMjY2LCJzZXNzaW9uaWQiOiJUUWw9pZGgyMFRlUc5YldmU1lhb0FnPT0ifQ.ksnCeOxnrNQeuNaQnmpauUG_eja70nVtaC  
CbamxFaB1LLkMIqzJ_wk_GMN1h4eM1WLS41VrauKwW8Fztd281leaSQPPWiT-  
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFUHVjvXpPctDsBYi9Ym5lUmFhxZgYI2Lx8gg0P6sLoUydcSHkKcUNGmGWw  
064LH7rMEpzyCTapJBXdkzUhJ-zm98fmOyloGGTesBteSWP_ksH14Xq411klebJ83sV4tL6-  
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": [u'exec'], [u'configure', "terminal"], [u'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=eyJhbGciOiJSUzI1NiIsImtpZCI6ImgyDE1NmdjYXlMG50emx3YmR3MDFjNnV5dGEyXQZiIiwidHlwIjoiaW50eS9yYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOiJEsInjvbnV5IjE6MTdlcCJpc3MiOiJBQ0kgVXBjYQYsInVzZ
XJuYW1lIjoiYWRTaW4iLCJlc2VyaWQiOiJlMzC0LCJlc2VyZmxhZ3MiOiJQsImlhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIxM
jgyMjY2LCJzZXNzaW9uaWQiOiJUUVw9pZGgyMFRlEuc5YldMUL1hb0FnPT0ifQ.ksnCeOxnRnQeuNaQnmpauUG_eja70nVtaC
bamxFab1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd28leaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM5lUmFxxZgYI2Lx8gg0P6sLoUydcSHkKcUNGRmGWw
064LH7rMEpyCTapJBXdkzUhJ-zm98fmOy1oGGTesBteSWP_ksH14Xq411klebJ83sV4tL6-
FJLhcPNIKwqYJ87fqUWwZFz5tY4JUJxrSnahKfwyidNxt5m8LCIc8pt-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=eyJhbGciOiJSUzI1NiIsImtpZCI6ImgyDE1NmdjYXlMG50emx3YmR3MDFjNnV5dGEyXQZiIiwidHlwIjoiaW50eS9yYmFjIjpbeyJkb21haW4iOiJhbGwiLCJyb2xlc1IiOiJEsInjvbnV5IjE6MTdlcCJpc3MiOiJBQ0kgVXBjYQYsInVzZ
XJuYW1lIjoiYWRTaW4iLCJlc2VyaWQiOiJlMzC0LCJlc2VyZmxhZ3MiOiJQsImlhdCI6MTYyMTI4MTY2NiwiZXhwIjoxNjIxM
jgyMjY2LCJzZXNzaW9uaWQiOiJUUVw9pZGgyMFRlEuc5YldMUL1hb0FnPT0ifQ.ksnCeOxnRnQeuNaQnmpauUG_eja70nVtaC
bamxFab1LLkMIqzJ_wk_GMN1h4eM1WLS41VraukWw8Fztd28leaSQPPWiT-
ieCjWxim8Sw4spYS8XBrBBx62tot201TIEJ8mUFHUjvXpPctDsBYi9YM5lUmFxxZgYI2Lx8gg0P6sLoUydcSHkKcUNGRmGWw
064LH7rMEpyCTapJBXdkzUhJ-zm98fmOy1oGGTesBteSWP_ksH14Xq411klebJ83sV4tL6-
FJLhcPNIKwqYJ87fqUWwZFz5tY4JUJxrSnahKfwyidNxt5m8LCIc8pt-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 the fvCtx object set with **status="deleted"**.

References & Useful links

- [ACI Programmability](#)
- [Cisco ACI Policy Model Guide](#)
- [Cisco ACI Support for NGINX Rate Limit](#)