UCS Director
Sample Workflow – APIC API Command

Background

This workflow has been built to take user input and perform an action on an Application Centric Infrastructure (ACI) Application Policy Infrastructure Controller (APIC) using the controller's native API. This is accomplished with a Custom Task that sets up a secure connection and performs the API call. It is designed to work with XML and JSON formatted requests.

Custom Task – APIC API Command

Pre-Requisites

HTTP access enabled on APIC
User credentials to perform APIC call(s) on the APIC

Summary

Sends an ACI API command to an APIC controller.

Description

Authenticates with the ACI Application Policy Infrastructure Controller (APIC) and then sends the API command to the APIC.

Methods:

Post – Send XML or JSON to APIC.
Get – Retrieves information from the APIC

Notes:

Requires HTTP to be enabled on APIC.
When using Post the URI extension (.json/.xml) must match the format of the "APIC API XML-JSON".

Category

APIC Tasks

Inputs

APIC IP	IP address of the APIC
APIC Login	User with sufficient credentials
APIC Password	Password for the above user
Method	The HTTP method (Post/Get)
APIC URI	The URI for the API call (this is the URL without the HTTP://<IP>)
APIC API XML-JSON	API Parameters for the requested URI up to 2048 characters

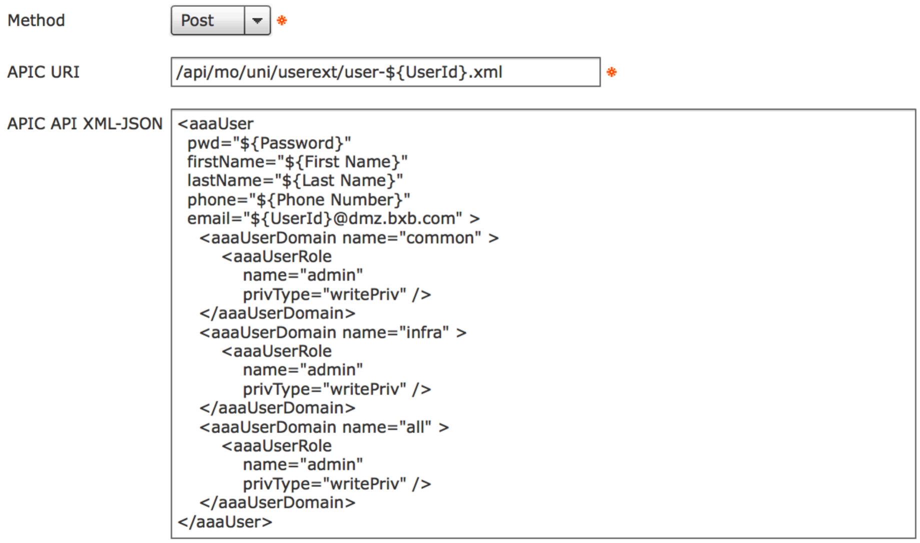
Outputs

Response	The response returned for the call

Examples

Post – XML

Note the suffix on the URI matches the XML content below.



Method: Post *

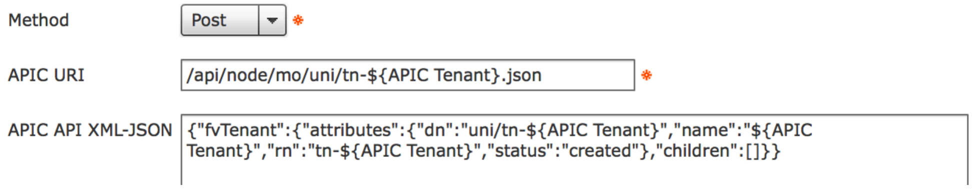
APIC URI: /api/mo/uni/userext/user-${UserId}.xml *

APIC API XML-JSON:

```
<aaaUser
 pwd="${Password}"
 firstName="${First Name}"
 lastName="${Last Name}"
 phone="${Phone Number}"
 email="${UserId}@dmz.bxb.com" >
   <aaaUserDomain name="common" >
     <aaaUserRole
        name="admin"
        privType="writePriv" />
   </aaaUserDomain>
   <aaaUserDomain name="infra" >
     <aaaUserRole
        name="admin"
        privType="writePriv" />
   </aaaUserDomain>
   <aaaUserDomain name="all" >
     <aaaUserRole
        name="admin"
        privType="writePriv" />
   </aaaUserDomain>
</aaaUser>
```

Post – JSON

Note the suffix on the URI matches the XML content below.



Method: Post *

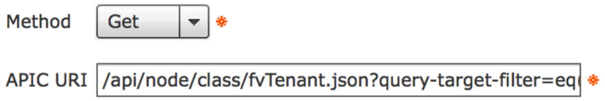
APIC URI: /api/node/mo/uni/tn-${APIC Tenant}.json *

APIC API XML-JSON:

```
{"fvTenant":{"attributes":{"dn":"uni/tn-${APIC Tenant}","name":"${APIC
Tenant}","rn":"tn-${APIC Tenant}","status":"created"},"children":[]}}
```

Get

There is no JSON/XML content required.



Method: Get *

APIC URI: /api/node/class/fvTenant.json?query-target-filter=eq *

Note – If changing from Get → Post, change the method to **Post**, click **Next**, then **Back** to re-nable the “APIC API XML-JSON” field.

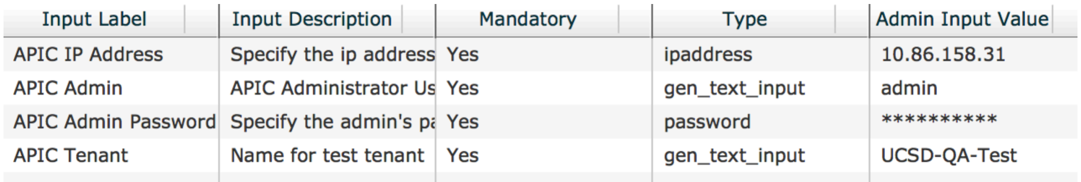
Sample Workflow

The sample “APIC API Command” workflow will create the tenant, verify it is found, then delete it.

Inputs

For simplicity, the workflow can be localized to any environment by setting the first three values below for their environment. The “APIC Tenant” can be modified to a unique value.

The “APIC Admin” user must have the ability to create a tenant.



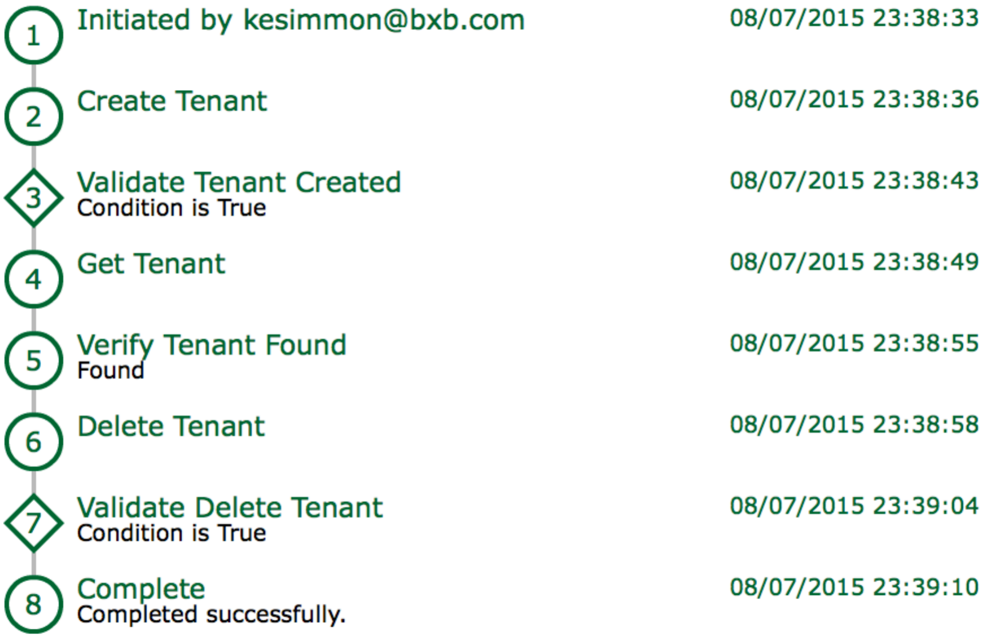
Input Label	Input Description	Mandatory	Type	Admin Input Value
APIC IP Address	Specify the ip address	Yes	ipaddress	10.86.158.31
APIC Admin	APIC Administrator Us	Yes	gen_text_input	admin
APIC Admin Password	Specify the admin's pa	Yes	password	**********
APIC Tenant	Name for test tenant	Yes	gen_text_input	UCSD-QA-Test

Workflow

Verifies add/delete/view functions with the APIC controller.

Create Tenant – Creates a tenant with the name in "APIC Tenant".
Get Tenant– Verifies the tenant was created and can be seen.
Delete Tenant – Removes the tenant from the APIC controller.

Output



1 Initiated by kesimmon@bxb.com 08/07/2015 23:38:33
2 Create Tenant 08/07/2015 23:38:36
3 Validate Tenant Created 08/07/2015 23:38:43
Condition is True
4 Get Tenant 08/07/2015 23:38:49
5 Verify Tenant Found 08/07/2015 23:38:55
Found
6 Delete Tenant 08/07/2015 23:38:58
7 Validate Delete Tenant 08/07/2015 23:39:04
Condition is True
8 Complete 08/07/2015 23:39:10
Completed successfully.

ACI Specific

Enable HTTP

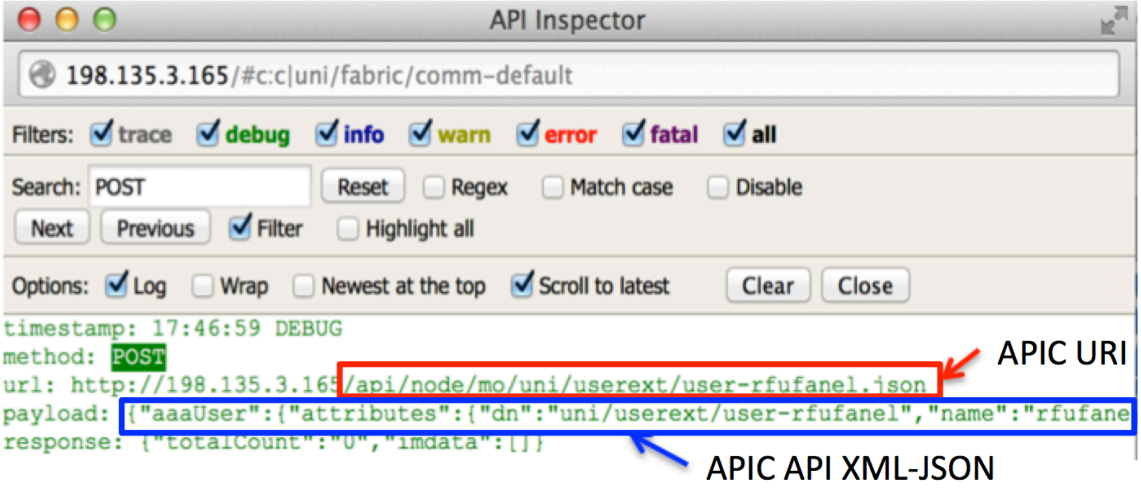
1. Login to the APIC as an administrator
2. Navigate to **Fabric → Fabric Policies → Pod Policies → Policies → Communication → default**
3. In the HTTP section set the **Admin State → Enabled**
4. Click **Submit**

API Syntax

The Cisco APIC REST API User Guide (http://www.cisco.com/c/en/us/td/docs/switches/datacenter/aci/apic/sw/1-x/api/rest/b_APIC_RESTful_API_User_Guide.html) has starter examples for using the APIC API with examples for XML and JSON.

The APIC has a built-in API Inspector that can watch actions performed in the APIC graphical user interface and record them in a separate window that can be filtered.

1. Log into the APIC
2. On the far right, click on the drop down menu with your userid
3. Select **Show API Inspector**
4. In the window that comes up, read/write operations to the GUI will be shown.
5. Perform the action of interest in the GUI.
6. In the Inspector window, set the **Search** to **POST** and click the **Filter** checkbox. This will now display the full URL as well as the Payload, which would go into the *APIC API XML-JSON* field in the UCS Director task. As shown in the example, the text can be substituted with UCS Director variables where appropriate.


API Inspector
198.135.3.165/#c:c|uni/fabric/comm-default
Filters: ☑ trace ☑ debug ☑ info ☑ warn ☑ error ☑ fatal ☑ all
Search: POST
Reset
□ Regex □ Match case □ Disable
Next
Previous
☑ Filter □ Highlight all
Options: ☑ Log □ Wrap □ Newest at the top ☑ Scroll to latest
Clear
Close

```
timestamp: 17:46:59 DEBUG
method: POST
url: http://198.135.3.165/api/node/mo/uni/userext/user-rfufanel.json
payload: {"aaaUser":{"attributes":{"dn":"uni/userext/user-rfufanel","name":"rfufane
response: {"totalCount":"0","imdata":[]}
```

APIC URI
APIC API XML-JSON

Last Validated

August 8, 2015
UCS Director 5.3.1.1
ACI APIC 1.1(1j)