Cisco ACI Multi-Site REST API Configuration Guide, Release 2.0(1)

First Published: 2018-10-24

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com go trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2018 Cisco Systems, Inc. All rights reserved.

CONTENTS

PREFACE

Preface xi

Audience xi

Document Conventions xi

Related Documentation xiii

Documentation Feedback xiv

Obtaining Documentation and Submitting a Service Request xiv

CHAPTER 1

New and Changed Information 1

New and Changed Information 1

CHAPTER 2

Using the REST API 3

About the REST API 3

Object Naming 3

Composing REST API Requests 3

User Roles and Authorization 4

Search for Policies by displayName 4

Accessing In-Browser Developer Tools to View REST API Calls 5

Using In-Browser Developer Tools to Work with REST API Calls 6

CHAPTER 3

Data Structure for REST Schemas 11

About Data Structure of the REST Schemas 11

Schema 11

Template 11

Name 12

displayName 12

tenantID 13

Reference Entities 13
ref 13
anpRef and vrfRef 13
Reference Generation 14
Local References 14
References from Another Schema 14
Application Profile 15
EPG 16
Bridge Domains 18
Contract 19
Filter 19
Site 20
External Authentication 22
Providers 22
Login Domains 23
Remote User Login 23
Backup 24
Auditing 24

CHAPTER 4 REST Examples 29
REST API Examples 29
APIs for Consistency Checker 30
About Consistency Checker 30
Download of File of APIC's Response (Raw JSON) 30
Trigger an APIC Consistency Service for a Schema 30
Trigger an APIC Consistency Service for a Template 31
Read the Consistency Status for a Schema 31
Read the Consistency Status for a Template 32
Read the Consistency Report for a Schema 32
Read the Consistency Report for a Template 38
Read the Consistency Report for a Template for a Specific Site 43
Get the Heatmap Summary 46
APIs for Site Local Properties 50
List Domains 50

List L3Outs 51
List L3Outs Route Control Panels 52
List Security Domains 52
List L4-L7 Devices 53
List L4-L7 Device Interfaces 54
List L4-L7 Redirect Policies 54
APIs For Local Authentication 55
User Login 55
User Logout 56
Refresh Token 56
Using Tokens on Authentication on APIs 57
Authenticating and Maintaining an API Session 57
Login to Authenticate and Maintain an API Session 57
Refresh to Authenticate and Maintain an API Session 58
APIs For External Authentication 59
About External Authentication 59
APIs For RADIUS Configurations 59
Create a New RADIUS Provider 59
Read RADIUS Provider Information 60
List Providers 60
Update an Existing RADIUS Provider 62
Delete a RADIUS Provider 62
APIs for TACACS + Configurations 63
Create a New TACACS Provider 63
Read TACACS Provider Information 63
List TACACS Providers 64
Update an Existing TACAS Provider 64
Delete a TACACS Provider 65
APIs For Login Domain 65
Create a New Login Domain 65
List all Active Login Domains 66
List Login Domains 67
Modify an Existing Login Domain 68
Delete Login Domain 68

APIs For Remote User Login 69
Remote User Login 69
APIs For Labels 70
Create a New Label 70
Read Label Information 70
Update a Label 71
Delete Labels 71
List Labels 71
APIs For Roles 72
Read Role Information 72
List Roles 73
APIs For Users 74
Create a New User 74
Reset Password for an Existing User 74
Read User Information 75
Update an Existing User 76
Delete a User 77
List Users 77
List Roles That May Be Associated With Users 78
Retrieve Roles Associated With a User 79
Retrieve Permissions Associated With a User 79
APIs For Fabric Connectivity 80
Get Fabric Connectivity for a Site 80
Get Fabric Connectivity for Multiple Sites 89
Get Fabric Connectivity Multipod Information 91
Get Fabric Connectivity Status 91
Update Fabric Connectivity 97
Refresh Fabric Connectivity for a Site 103
Fabric Leaf Switches And Physical Interfaces 105
Get Nodes 105
Get Physical Interfaces Details 105
APIs For Tenants 107
Create a Tenant 107
Read Tenant Information 107

Update an Existing Tenant 108
Delete a Tenant 109
List Tenants 109
List Sites Associated with a Tenant 111
List Users Associated with a Tenant 111
List Schemas Associated with a Tenant 112
List Schemas Associated with all Tenants 113
Create a New Tenant Network Mapping 114
Get Existing Tenant Network Mappings 115
APIs For All Schemas 115
Create a New Schema 115
Read a Schema 168
Schema Health Summary 170
Schema Health Sync 172
Delete Schema Health Information 174
Retrieve Policy States 174
Retrieve References 176
Retrieving Schema Health and Faults 178
Update an Existing Schema 183
Delete a Schema 186
List Schema Identities 186
List Sites Associated to a Schema 188
List Tenants That Can be Associated With a Schema 189
List All Tenants Associated to Schemas 190
List Sites That Can Be Associated With a Schema 191
List All Sites Associated to Schemas 192
Retrieve Service Graph Node Types 192
Create Service Graph Node Types 193
APIs For Import 193
Import a Tenant from APIC 193
APIs For Sites 204
Create a New Site 204
Read Site Information 204
Read Site Information That Contains Health Faults 205

Update an Existing Site 206
Delete a Site 207
List Sites 207
List Labels 208
APIs For Template 209
Execute a Template 209
Undeploy a Template 209
APIs For Entities 210
Login Domains 210
List L3Outs 211
APIs For Platforms 211
Read Node Information 211
Download Troubleshooting Report 212
Expected HTTP Response Code Examples 212
APIs For Audit 213
About Auditing 213
Filter Audit Records 213
Paginate Audit Records 217
About Login Domains Auditing 219
List or Filter Login Domains Audit Records 219
About Users Auditing 220
List or Filter User Audit Records 220
About Providers Auditing 221
List or Filter Provider Audit Records 222
List or Filter RADIUS Provider Audit Records 223
List or Filter TACACS Provider Audit Records 224
Find Users Associated with Tenant Audit Records 225
List or Filter Audit Records for Tenants 226
Find Users Associated with Schema Audit Records 229
List or Filter Audit Records for Schemas 230
Find Users Associated with Fabric-Connectivity Audit Records 240
List or Filter Audit Records for Fabric-Connectivity 240
APIs For Backups 240
About Backup 240

Create a New Backup 241
List Backups 241
Download a Backup 243
Upload a Backup 243

Preface

This preface includes the following sections:

- Audience, on page xi
- Document Conventions, on page xi
- Related Documentation, on page xiii
- Documentation Feedback, on page xiv
- Obtaining Documentation and Submitting a Service Request, on page xiv

Audience

This guide is intended primarily for data center administrators with responsibilities and expertise in one or more of the following:

- Virtual machine installation and administration
- Server administration
- Switch and network administration

Document Conventions

Command descriptions use the following conventions:

Convention	Description
bold	Bold text indicates the commands and keywords that you enter literally as shown.
Italic	Italic text indicates arguments for which the user supplies the values.
[x]	Square brackets enclose an optional element (keyword or argument).
[x | y]	Square brackets enclosing keywords or arguments separated by a vertical bar indicate an optional choice.
{x | y}	Braces enclosing keywords or arguments separated by a vertical bar indicate a required choice.

Convention	Description
[x {y | z}]	Nested set of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.
`variable`	Indicates a variable for which you supply values, in context where italics cannot be used.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.

Examples use the following conventions:

Convention	Description
`screen font`	Terminal sessions and information the switch displays are in screen font.
`boldface screen font`	Information you must enter is in boldface screen font.
italic screen font	Arguments for which you supply values are in italic screen font.
< >	Nonprinting characters, such as passwords, are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

This document uses the following conventions:

Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.

Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.

Warning IMPORTANT SAFETY INSTRUCTIONS

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device.

SAVE THESE INSTRUCTIONS

Related Documentation

Application Policy Infrastructure Controller (APIC) Documentation

The following companion guides provide documentation for APIC:

- *Cisco APIC Getting Started Guide*
- *Cisco APIC Basic Configuration Guide*
- *Cisco ACI Fundamentals*
- *Cisco APIC Layer 2 Networking Configuration Guide*
- *Cisco APIC Layer 3 Networking Configuration Guide*
- *Cisco APIC NX-OS Style Command-Line Interface Configuration Guide*
- *Cisco APIC REST API Configuration Guide*
- *Cisco APIC Layer 4 to Layer 7 Services Deployment Guide*
- *Cisco ACI Virtualization Guide*
- *Cisco Application Centric Infrastructure Best Practices Guide*

All these documents are available at the following URL: http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html

Cisco Application Centric Infrastructure (ACI) Documentation

The broader ACI documentation is available at the following URL: http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html.

Cisco Application Centric Infrastructure (ACI) Simulator Documentation

The Cisco ACI Simulator documentation is available at http://www.cisco.com/c/en/us/support/cloud-systems-management/application-centric-infrastructure-simulator/tsd-products-support-series-home.html.

Cisco Nexus 9000 Series Switches Documentation

The Cisco Nexus 9000 Series Switches documentation is available at http://www.cisco.com/c/en/us/support/switches/nexus-9000-series-switches/tsd-products-support-series-home.html.

Cisco Application Virtual Switch Documentation

The Cisco Application Virtual Switch (AVS) documentation is available at http://www.cisco.com/c/en/us/support/switches/application-virtual-switch/tsd-products-support-series-home.html.

Cisco Application Centric Infrastructure (ACI) Integration with OpenStack Documentation

Cisco ACI integration with OpenStack documentation is available at http://www.cisco.com/c/en/us/support/cloud-systems-management/application-policy-infrastructure-controller-apic/tsd-products-support-series-home.html.

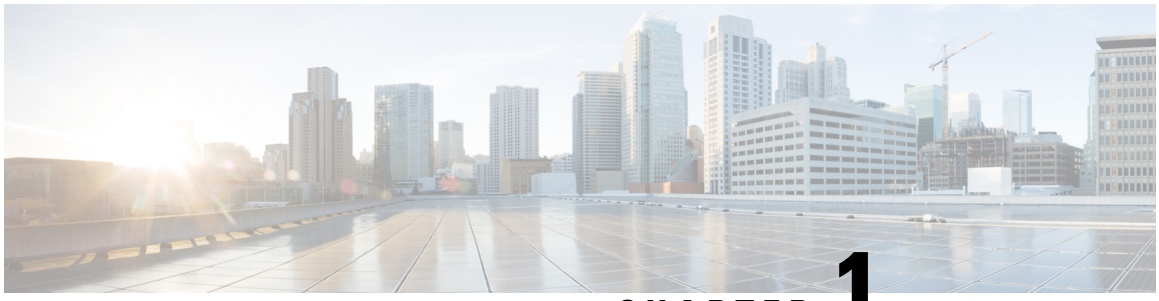
Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to apic-docfeedback@cisco.com. We appreciate your feedback.

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, using the Cisco Bug Search Tool (BST), submitting a service request, and gathering additional information, see *What's New in Cisco Product Documentation* at: http://www.cisco.com/c/en/us/td/docs/general/whatsnew/whatsnew.html

Subscribe to *What's New in Cisco Product Documentation*, which lists all new and revised Cisco technical documentation as an RSS feed and delivers content directly to your desktop using a reader application. The RSS feeds are a free service.



CHAPTER 1

New and Changed Information

This chapter contains the following sections:

- New and Changed Information, on page 1

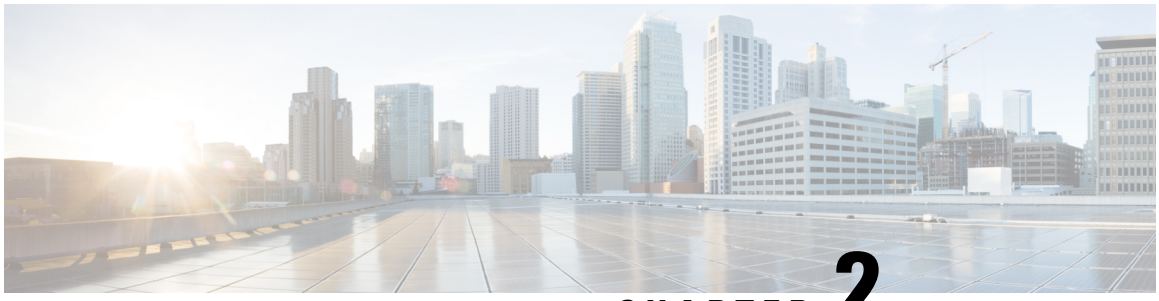
New and Changed Information

The following table provides an overview of the significant changes to the organization and features in this guide up to this current release. The table does not provide an exhaustive list of all changes made to the guide.

Table 1: New Features and Changed Behavior in Cisco ACI Multi-Site Release 2.0(1)

Cisco ACI Multi-Site Release	Description	Where Documented
2.0(1)	Using browser developer tools for API call inspection.	Accessing In-Browser Developer Tools to View REST API Calls, on page 5

New and Changed Information



CHAPTER 2

Using the REST API

This chapter contains the following sections:

- About the REST API, on page 3
- Object Naming, on page 3
- Composing REST API Requests, on page 3
- User Roles and Authorization, on page 4
- Search for Policies by `displayName`, on page 4
- Accessing In-Browser Developer Tools to View REST API Calls, on page 5
- Using In-Browser Developer Tools to Work with REST API Calls, on page 6

About the REST API

The Cisco ACI Multi-Site REST API is a programmatic interface that uses REST architecture. The API accepts and returns HTTP (not enabled by default) or HTTPS messages that contain JavaScript Object Notation (JSON) documents. You can use any programming language to generate the messages and the JSON documents that contain the API methods.

Standard REST methods are supported on the API, which includes POST, GET, PUT, and DELETE operations through HTTP. The PUT methods are idempotent, meaning that there is no additional effect if they are called more than once with the same input parameters. The GET method is nullipotent, meaning that it can be called zero or more times without making any changes (or that it is a read-only operation).

Payloads to and from the REST interface must be encapsulated through JSON encoding. Containment is defined by creating child attributes.

Object Naming

Resources are accessed by supplying the database identifier.

Composing REST API Requests

Read and Write Operations

```
/api/v1/tenants/<id>
```

`v1` is the first version of the APIs.

Read Operations

After the object payloads are properly encoded as JSON, they can be used in create, read, update, or delete operations on the REST API.

Because the REST API is HTTP-based, defining the URI to access a certain resource type is important. The first two sections of the request URI simply define the protocol. Next in the request URI is the literal string /api, indicating that the API is invoked.

Write Operations

POST = create

PUT = update

If an object does not exist, it is created. If an object exists, it is updated to reflect any changes between its existing state and desired state.

Create and update operations use the same syntax as read operations. The create or update operation must target a specific entity. The payload of the POST and PUT operations contains JSON encoded data.

User Roles and Authorization

The API supports user-specific authorization based on the role of a user. A user can be associated with specific roles for access based on their function. While the admin can perform all operations, REST endpoints can be restricted based on a user's role. For more information, see the *Cisco ACI Multi-Site Configuration Guide*.

Search for Policies by `displayName`

GET

```
GET /api/v1/schemas/search-policies?type=:type&tenant-id=:tenantId&query=:query
```

For example:

```
/api/v1/schemas/search-policies?type=vrf&tenant-id=T1&query=vrf
/api/v1/schemas/search-policies?type=contract&query=contra
/api/v1/schemas/search-policies?type=serviceGraph&tenant-id=T1&query=SG
```

Searchable policy types are as follows:

- `vrf`
- `bd`
- `contract`
- `filter`
- `serviceGraph`

You must follow these guidelines when using searchable Policy Types:

- TenantId is mandatory for the following policy types: VRF, BD, Filter, and serviceGraph.

- TenantId is not required and is not used for Policy type Contracts.
- Query must be three or more characters (in other cases, a BadRequest will be returned).
- This API may return 100 records.

REQUEST

```
<empty>
```

RESPONSE

```
{
  "policies": [
    {
      "vrfRef": "/schemas/58dad0988000004201faa03c/templates/template1/vrfs/UntitledVRF1",
      "policyName": "Untitled VRF 1",
      "schemaName": "Schema",
      "templateName": "Template 1"
    },
    {
      "vrfRef": "/schemas/58dae4b2800000e500b79f1c/templates/template1/vrfs/UntitledVRF1",
      "policyName": "Untitled VRF 1",
      "schemaName": "Schema2",
      "templateName": "Template 1"
    }
  ]
}
```

Accessing In-Browser Developer Tools to View REST API Calls

Cisco ACI Multi-SiteOrchestrator is fully RESTful, as such each action performed in the GUI triggers one or more REST API calls to retrieve the fabric information to be displayed or to make changes to the fabric. You can view these API calls using the built-in developer tools in your browser of choice. Inspecting API calls associated with one or more UI actions may be useful for becoming familiar with the specific call syntax and contents.

This section describes how to access developer tools in your browser.

Step 1 Open the built-in browser developer tools.

a) If you are using Google Chrome, right click in the main browser window and choose **Inspect**.
b) If you are using Mozilla Firefox, right click in the main browser window and choose **Inspect Element**.
c) If you are using Microsoft Edge, right click in the main browser window and choose **Inspect Element**.
d) If you are using Apple Safari, you must first enable the Develop menu.

   From the top menu bar, choose **Safari** > **Preferences**. Then select the **Advanced** tab and check the **Show Develop menu in menu bar** checkbox.

   Finally, open the developer tools pane by choosing **Develop** > **Show Web Inspector** from the top menu bar.

A developer tools panel will open in the main browser window alongside the current page.

Step 2 In the developer tools panel that opens, select the **Network** tab.

Step 3 Navigate to your Multi-Site Orchestrator and log in.

Step 4 Perform an action in the Multi-Site GUI to trigger one or more API calls.

You will be able to see the calls as they are being made by the browser to the Multi-Site Orchestrator.

Note You may want to clear the list before performing an action by clicking **Clear** button in the **Network** tab's menu bar.

Step 5 (Optional) Filter the list of API calls to view the relevant ones.

You can filter the list to display only the REST API calls by clicking the **XHR** button in the **Network** tab's menu bar.

Step 6 (Optional) Sort the list of API calls to group similar ones.

You can click on the column title to sort the network calls based on the property.

Step 7 Select one of the listed API calls to inspect its content.

When you select one of the calls, you can view its details in the pane that opens:

- **Headers** tab shows the general, request, and response headers for the selected API call. You can view the call's request method, such as `GET` or `POST`, the remote address, authorization token, and the cookies.
- **Preview** tab shows the XML text of the call sent to the server.
- **Response** tab shows the XML text of the response received from the server.
- **Cookies** tab shows the cookie information.

What to do next

For in-depth information on each browser's developer tools and their capabilities, see their respective documentation:

- Google Chrome: https://developers.google.com/web/tools/chrome-devtools/#network
- Mozilla Firefox: https://developer.mozilla.org/en-US/docs/Tools/Network_Monitor
- Microsoft Edge: https://docs.microsoft.com/en-us/microsoft-edge/devtools-guide/network
- Apple Safari: https://support.apple.com/guide/safari-developer/network-tab-dev1f3525e58/mac

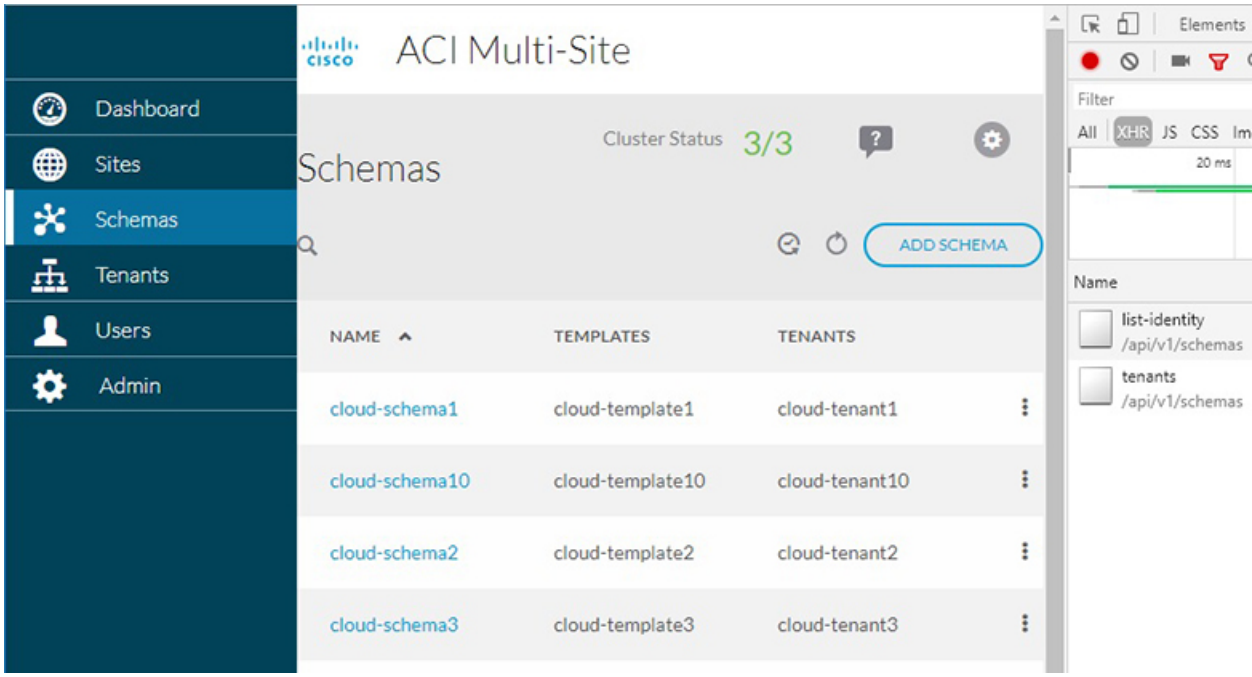
Using In-Browser Developer Tools to Work with REST API Calls

This section illustrates some simple examples of using browser tools to view, modify, and re-send REST API calls. The examples presented here use Google Chrome, however the steps are similar for other browsers. Accessing each browser's developer tools is described in Accessing In-Browser Developer Tools to View REST API Calls, on page 5.

When you perform an action in the Cisco ACI Multi-Site Orchestrator GUI with DevTools open, you can see a list of the API calls made by the GUI to the Orchestrator. From there on, you can select each individual call to inspect its request and response contents.

For example, if you click on the **Schemas** view in the left-hand navigation sidebar of the GUI, two calls are made to `/api/v1/schemas` – one to get a list of schemas and another to get a list of tenants for each schema:

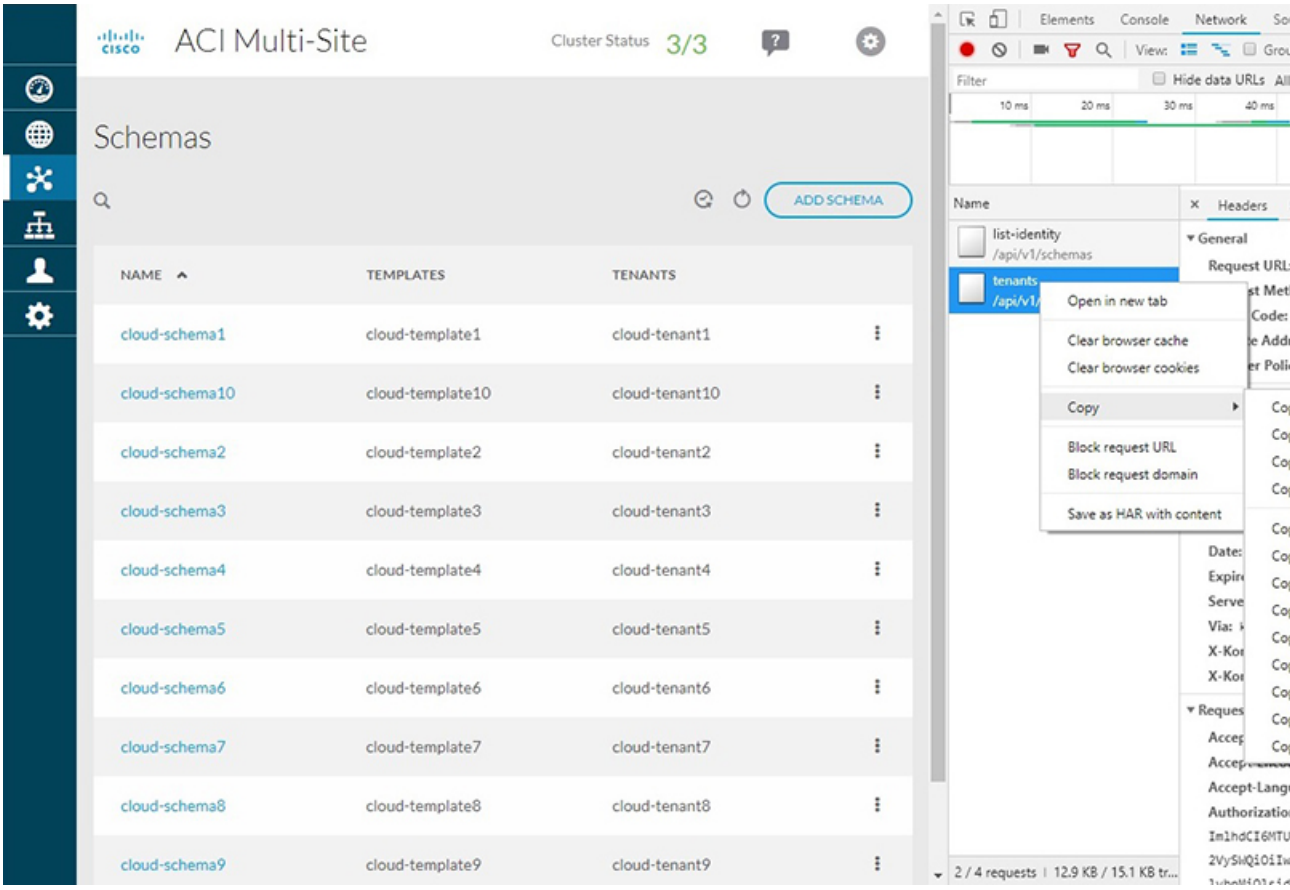
Figure 1: Schemas API Calls



If you then select one of the calls, for example `tenants`, you can view the call details, such as the request method, request URL, authentication token, and so on. The response tab provides the exact XML string received in response to the call.

If you right-click on any of the listed API calls, you can choose to copy the call in any of the available formats:

Figure 2: Copy API Call



For example, you can choose to copy the request as cURL, which you can then execute in 3rd party scripts:

```
curl "https://172.31.187.59/api/v1/schemas/tenants" -H "Authorization: Bearer
 -H "DNT: 1" -H "Accept-Encoding: gzip, deflate, br" -H "Accept-Language: en-US,en;q=0.9"
-H "User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/69.0.3497.100 Safari/537.36" -H "Content-Type: application/json" -H "Accept:
 application/json" -H "Referer: https://172.31.187.59/schemas" -H "Connection: keep-alive"
 --compressed --insecure
```

Another example would be the `users` API which is called when you select the **Users** view. The GUI retrieves the list of users using the following corresponding cURL request:

```
curl "https://172.31.187.59/api/v1/users" -H "Authorization: Bearer
 -H "DNT: 1" -H "Accept-Encoding: gzip, deflate, br" -H "Accept-Language: en-US,en;q=0.9"
-H "User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/69.0.3497.100 Safari/537.36" -H "Content-Type: application/json" -H "Accept:
 application/json" -H "Referer: https://172.31.187.59/users" -H "Connection: keep-alive"
--compressed --insecure
```

And the request returns a JSON-formatted XML message similar to the following:

```
{
    "users": [{
            "id": "59f17fe7120000da00bf0d11",
            "username": "max-usermgr",
            "password": "******",
```

```
            "firstName": "firstName",
            "lastName": "lastName",
            "emailAddress": "email@domain.com",
            "phoneNumber": "7474747477",
            "accountStatus": "active",
            "needsPasswordUpdate": false,
            "roles": [{
                    "roleId": "0000ffff0000000000000035"
                }
            ],
            "domainId": "0000ffff0000000000000090"
        }, {
            "id": "59f17f9e0e00001701000326",
            "username": "max-schemamgr",
            "password": "******",
            "firstName": "firstName",
            "lastName": "lastName",
            "emailAddress": "email@domain.com",
            "phoneNumber": "34343442234",
            "accountStatus": "active",
            "needsPasswordUpdate": false,
            "roles": [{
                    "roleId": "0000ffff0000000000000033"
                }
            ],
            "domainId": "0000ffff0000000000000090"
        }, {
            "id": "59f17f6b120000eb00bf0d10",
            "username": "max-sitemgr",
            "password": "******",
            "firstName": "firstName",
            "lastName": "lastName",
            "emailAddress": "email@domain.com",
            "phoneNumber": "3838833838",
            "accountStatus": "active",
            "needsPasswordUpdate": false,
            "roles": [{
                    "roleId": "0000ffff0000000000000032"
                }
            ],
            "domainId": "0000ffff0000000000000090"
        }
    ]
}
```


Modifying Examples and Making Changes to the Fabric

In addition to being able to retrieve information about your Cisco ACI Multi-Site environment using the REST API, you can use it to make changes.

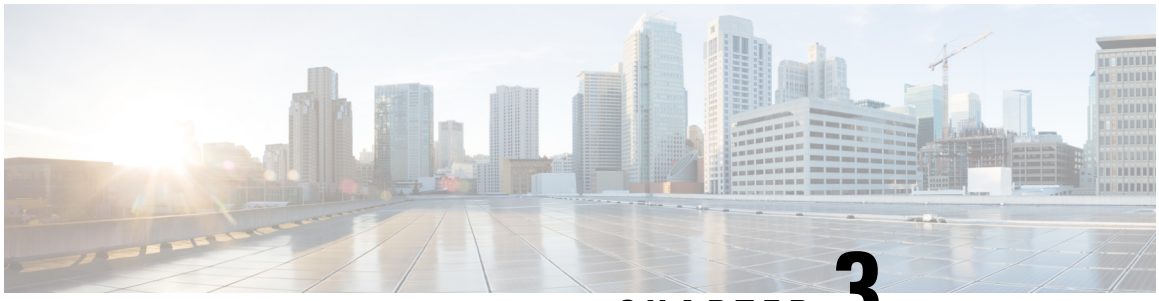
For instance, if instead of viewing the users, you create one in the GUI, you can then copy the call as an example, modify it, and then use the modified call in your 3rd party script to create or update user accounts. Below is a sample cURL command to create a user along with the XML POST contents:

```
curl "https://172.31.187.59/api/v1/users" -H "Origin: https://172.31.187.59" -H
"Accept-Encoding: gzip, deflate, br" -H "Accept-Language: en-US,en;q=0.9" -H "Authorization:
 Bearer
 -H "Content-Type: application/json" -H "Accept: application/json" -H "Referer:
https://172.31.187.59/users/create" -H "User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64;
x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/69.0.3497.100 Safari/537.36" -H
"Connection: keep-alive" -H "DNT: 1" --data-binary
```

```
'{[illegible]}'
 --compressed --insecure
```

And this is the contents of the `POST` call to create a user:

```
{
    "id": "5bb7be240f00008206d3f137",
    "username": "newUserName",
    "password": "******",
    "firstName": "firstName",
    "lastName": "lastName",
    "emailAddress": "email@domain.com",
    "phoneNumber": "1234567890",
    "accountStatus": "active",
    "needsPasswordUpdate": true,
    "roles": [{
            "roleId": "0000ffff0000000000000031"
        }, {
            "roleId": "0000ffff0000000000000032"
        }
    ],
    "domainId": "0000ffff0000000000000090"
}
```



CHAPTER 3

Data Structure for REST Schemas

This chapter contains the following sections:

- About Data Structure of the REST Schemas, on page 11
- External Authentication, on page 22

About Data Structure of the REST Schemas

This section describes how the REST Schemas are structured in Cisco ACI Multi-Site using JSON. The data structure is consistent across the Multi-Site. The top-level entities in a Schema are templates. The templates contain smaller entities such as application profiles (anps), bridge domains (bds), virtual routing and forwarding (vrfs) and such. These entities are also composed from other smaller structures. A few high-level entities or concepts are described in this section.

Schema

A schema is a container for templates. When a site is chosen, it automatically gets added to the schema, and it maps to the template in that schema. The high-level attributes of a schema are displayed in the following example API snippet:

```
{
  "id": "583c7c482501002501061985",
  "displayName": "Schema 1",
  "templates": [ ],
  "sites": [ ],
}
```

Template

A template is the detailed structure that contains policies that the user wants to push to Cisco APIC. Templates exist in the context of a Schema and not on their own. A user can define the policy of a new template as desired. If there are multiple templates, the user can choose a template from which to inherit the policy. The high-level attributes of a template are as follows:

- name
- displayName
- tenantId

- anps
- vrfs
- bds
- contracts
- filters

The key structures inside templates are as follows:

- anps (application network profiles)
- vrfs
- bds
- contracts
- filters

When an entity name is pluralized (such as vrfs, bds), it means that an array of that sub-entity is supported under its parent entity, and you can have multiple entries of the entity present in the template.

The high-level attributes of a template are displayed in the following example API snippet:

```
{
      "name": "template1",
      "displayName": "Template 1",
      "tenantId": "592c7c482501002501061578",
      "anps": [ ],
      "vrfs": [ ],
      "bds": [ ],
      "contracts": [ ],
      "filters": [ ],
}
```

Name

The name field is for entities that have a name. This is the name that is mapped to the names of policies in Cisco APIC. These names are restrictive and are determined based on restrictions placed by Cisco APIC. The following is an example of a Name API snippet:

```
"name": "template1",
```

displayName

The displayName field contains a user-friendly name that supports spaces and special characters in the displayName. The displayName will be visible in the Cisco ACI Multi-Site UI, and it is generally editable in Multi-Site.

The following is an example of a displayName API snippet:

```
"displayName": "Template 1",
```

tenantID

The tenantID is an association with tenants.

The following is an example of a tenantID API snippet:

```
"tenantId": "592c7c482501002501061578",
```

Reference Entities

ref

A ref field for a policy is optional. However, to reference entities in the system that you create, you must specify the reference in these relations. Users must specify references in order to provide relationships between policies.

anpRef and vrfRef

The anpRef identifies the application profile (anp), for which it is a sub-attribute in the system. The anpRef is an identifier for anp. The anpRef sub-attribute is typically used in situations where other entities have to be referred within a particular entity. For policies within a template, a ref field is consistently generated. A user does not specify the value of the anpRef sub-attribute. The system generates an anpRef for each and every policy.

Providing an input as anpRef is not a requirement. However, if you define a policy within a particular Schema, and you want to reference it immediately, you can provide the anpRef as input.

The following example displays an anpRef in the anps structure:

```
"anps": [
        {
          "name": "anp1",
          "displayName": "Anp 1",
          "anpRef": "/schemas/583c7c482501002501061985/templates/template1/anps/anp1",
          "epgs": [ <->]
      }
],
```

The following example displays a VRF reference:

```
"vrfs": [
        {
          "name": "vrf1",
          "displayName": "VRF 1",
          "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1"
        }
```

Reference Generation

References are generated for each and every policy. A reference is generated on the first save. However, if you define a policy within the Schema, and you want to reference it right away, you can specify a reference without having to wait for it to be generated.

In a JSON example, the left side data shows what the user provides, such as local references. After an operation concludes and the data persists, the system provides the data that is displayed on the right side. For example, after persistence, the system generates the prefix for a reference, and it is displayed on the right side as a system-generated reference.

Local References

In some cases, the reference that is supplied as input to the post operation, does not contain the Schema ID (as system generated references contain). A local reference means that an entity within this Schema is referenced by the path of that entity. In such a case, the ID is not used as a reference, and the reference does not contain Schema ID.

For example, you create a contract, and you want to reference the contract within the EPG immediately. The system will not yet have generated a reference for the contract, so the generated 'ref' is not available. However, you can still ensure the relationship by referencing the contract to the path of an entity that already exists in the system.

Once it persists, the system makes a local reference an absolute reference. The system prepends the Schema ID even though it is a local reference that is available in the same Schema, and thereby fully populates the reference.

The following is an example of a local contract reference API snippet:

```
{
   "name": "epg1",
   "displayName": "EPG 2",
   "contractRelationships":
  {
     "contractRef": "contractRef": "/templates/template1/contracts/contract1",
     "relationshipType":  "consumer"
    }
 },
```

References from Another Schema

It is possible to associate an entity (for example, a provider contract) from another Schema. You can specify the contract reference by prepending it with an ID contained in an absolute reference from another Schema. This references another policy in another Schema.

The following is an API snippet example of a reference from another Schema:

```
{
   "name": "epg1",
   "displayName": "EPG 2",
   "contractRelationships":
  {
     "contractRef":
"/schemas/590ca1811f000062006eef23/templates/template1/contracts/contract2",
     "relationshipType":  "provider"
    }
 },
```

Note The attributes for the **epg** > **contractRelationships** > **relationshipType** are provider and consumer.

Application Profile

The application profile is the anp entity. This contains epgs as a sub-entity.

The following is an example application profile entity API snippet:

```
"anps": [
        {
          "name": "anp1",
          "displayName": "Anp 1",
          "anpRef": "/schemas/583c7c482501002501061985/templates/template1/anps/anp1",
          "epgs": [
            {
              "name": "epg1",
              "displayName": "EPG 1",
              "epgRef":
"/schemas/583c7c482501002501061985/templates/template1/anps/anp1/epgs/epg1",
              "contractRelationships": [
                {
                  "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
                  "relationshipType": "provider"
                }
              ],
              "subnets": [ {
                    "ip": "192.168.132.1/24",
                    "description": "Subnet for IT department",
                    "scope" : "private"
                    "shared": true
                }, {
                    "ip": "192.168.133.1/24",
                    "description": "Subnet for sales department",
                    "scope" : "public"
                    "shared": false
                }],
              "uSegEpg" : true,
              "uSegAttrs" : [ {
                        "name" : "IP1",
                        "displayName" : "...",
                        "description" : "... ",
                        "type" : "ip",
                        "fvSubnet" : true,
                        "value" : "0.0.0.0"
                      }, {
                       "name" : "IP2",
                        "description" : "... ",
                        "type" : "ip",
                        "fvSubnet" : false,
                        "value" : "192.168.132.1"
                      }, {
                       "name" : "MAC",
                       "description" : "...",
                       "type" : "mac",
                        "value" : "aa:bb:cc:dd:ee:ff"
                      }, {
                       "name" : "VM Name",
```

```
                    "description" : "...",
                    "type" : "vm-name",
                      "operator" : "equals | contains | startsWith | endsWith",
                    "value" : "..."
                   }, {
                    "name" : "VM Data Center",
                    "description" : "... (optional)",
                      "type" : "rootContName",
                      "operator" : "...",
                      "value" : "..."
                  }, {
                      "name" : "VM Hypervisor Identifier",
                      "description" : "... (optional)",
                      "type" : "hv",
                      "operator" : "...",
                      "value" : "..."
                  }, {
                      "name" : "VmOperatingSystem",
                      "description" : "...",
                      "type" : "guest-os",
                      "operator" : "...",
                      "value" : "..."
                  }, {
                      "name" : "VM Tag",
                      "description" : "... ",
                      "type" : "tag",
                      "operator" : "...",
                      "category" : "...",
                      "value" : "..."
                  }, {
                      "name" : "Vm Identifier",
                      "description" : "... ",
                      "type" : "vm",
                      "operator" : "...",
                      "value" : "..."
                  }, {
                      "name" : "Vm Vmm Domain",
                      "description" : "...",
                      "type" : "domain",
                      "operator" : "...",
                      "value" : "..."
                  }, {
                      "name" : "Vm Vnic Dn",
                      "description" : "... (optional)",
                      "type" : "vnic",
                      "operator" : "...",
                      "value" : "..."
                  } ],
            "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1"
          },
          ]
    }
  ],
```

EPG

One EPG can be related to multiple contracts through provider or consumer contract relationships. EPGs are related to contracts and bridge domains (bds).

The following is an example of an epg API snippet:

```
"epgs": [
```

```
            {
              "name": "epg1",
              "displayName": "EPG 1",
              "epgRef":
"/schemas/583c7c482501002501061985/templates/template1/anps/anp1/epgs/epg1",
              "contractRelationships": [
                {
                  "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
                  "relationshipType": "provider|consumer"
                }
              ],
              "subnets": [ {
                    "ip": "192.168.132.1/24",
                    "description": "Subnet for IT department",
                    "scope" : "private"
                    "shared": true
                  }, {
                    "ip": "192.168.133.1/24",
                    "description": "Subnet for sales department",
                    "scope" : "public"
                    "shared": false
                  }],
              "uSegEpg" : true,
              "uSegAttrs" : [ {
                        "name" : "IP1",
                        "displayName" : "...",
                        "description" : "... ",
                        "type" : "ip",
                        "fvSubnet" : true,
                        "value" : "0.0.0.0"
                       }, {
                        "name" : "IP2",
                         "description" : "... ",
                         "type" : "ip",
                         "fvSubnet" : false,
                         "value" : "192.168.132.1"
                       }, {
                        "name" : "MAC",
                        "description" : "...",
                        "type" : "mac",
                         "value" : "aa:bb:cc:dd:ee:ff"
                       }, {
                        "name" : "VM Name",
                        "description" : "...",
                        "type" : "vm-name",
                          "operator" : "equals | contains | startsWith | endsWith",
                        "value" : "..."
                       }, {
                        "name" : "VM Data Center",
                        "description" : "... (optional)",
                          "type" : "rootContName",
                          "operator" : "...",
                          "value" : "..."
                      }, {
                          "name" : "VM Hypervisor Identifier",
                          "description" : "... (optional)",
                          "type" : "hv",
                          "operator" : "...",
                          "value" : "..."
                      }, {
                          "name" : "VmOperatingSystem",
                          "description" : "...",
                          "type" : "guest-os",
```

```
                    "operator" : "...",
                    "value" : "..."
                }, {
                    "name" : "VM Tag",
                    "description" : "... ",
                    "type" : "tag",
                    "operator" : "...",
                    "category" : "...",
                    "value" : "..."
                }, {
                    "name" : "Vm Identifier",
                    "description" : "... ",
                    "type" : "vm",
                    "operator" : "...",
                    "value" : "..."
                }, {
                    "name" : "Vm Vmm Domain",
                    "description" : "...",
                    "type" : "domain",
                    "operator" : "...",
                    "value" : "..."
                }, {
                    "name" : "Vm Vnic Dn",
                    "description" : "... (optional)",
                    "type" : "vnic",
                    "operator" : "...",
                    "value" : "..."
                } ],
            "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1"
          },
        ]
```

Note The attributes for **VM Name** > **uSegAttrs** > **operator** are "equals", "contain", "startsWith", "endsWith".

Bridge Domains

Bridge Domains (bds) have attributes such as multicast flooding.

The following is an example of a bridge domain API snippet:

```
"bds": [
        {
          "name": "bd1",
          "displayName": "BD 1",
          "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
          "l3UnknownMulticastFlooding": "opt-flood",
          "intersiteBumTrafficAllow": true,
          "multiDestinationFlooding": "encap-flood",
          "l2UnknownUnicast": "flood",
          "l2Stretch": true,
          "subnets": [ ... ],
          "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1"
        }
      ],
```

Note

- The attributes for **bds** > **l3UnknownMulticastFlooding** are "flood", "opt-flood".
- The attributes for **bds** > **multiDestinationFlooding** are "bd-flood", "drop", "encap-flood".
- The attributes for **bds** > **l2UnknownUnicast** are "flood"and "proxy".

Contract

The following is an example of a contract API snippet:

```
"contracts": [
        {
          "name": "contract1",
          "displayName": "Contract 1",
          "filterRelationships": [
            {
              "filterRef": "/templates/template1/filters/filter1",
              "directives": [
                "log|none"
              ]
            }
          ],
          "scope": "global|tenant|context|application-profile"
        },
        {
          "name": "contract2",
          "displayName": "Contract 2",
          "scope": "global|tenant|context|application-profile"
        }
      ],
```

Note The attributes for **contracts** > **scope** are "global", "tenant", "context", "application-profile".

Filter

The following is an example of a filter API snippet:

```
"filters": [
        {
          "name": "filter1",
          "displayName": "Filter 1",
          "description": "",
          "entries": [
            {
              "name": "filterEntry11",
              "displayName": "Filter Entry 11",
              "description": "",
              "etherType": "ip",
              "ipProtocol": "icmp"
            },
            {
              "name": "filterEntry12",
```

```
              "displayName": "Filter Entry 12",
              "description": "",
              "etherType": "ip",
              "ipProtocol": "udp",
              "matchOnlyFragments": false,
              "sourceFrom": "dns",
              "sourceTo": "http",
              "destinationFrom": "dns",
              "destinationTo": "80"
            }
          ]
        },
```

the following is an example of attributes for a filter:

```
{
          "name": "filter2",
          "displayName": "Filter 2",
          "description": "",
          "entries": [
            {
              "name": "filterEntry21",
              "displayName": "Filter Entry 21",
              "description": "",
              "etherType": "arp|fcoe|ip|mac_security|mpls_ucast|trill|unspecified",
              "arpFlag": "request|reply|unspecified",
              "ipProtocol":
"eigrp|egp|icmp|icmpv6|igmp|igp|l2tp|ospfigp|pim|tcp|udp|unspecified",
              "matchOnlyFragments": false,
              "stateful": false,
            "sourceFrom": "dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",

              "sourceTo": "dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
              "destinationFrom":
"dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
              "destinationTo":
"dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
              "tcpSessionRules": [
                "acknowledgement|established|finish|synchronize|reset|unspecified",
                "acknowledgement|established|finish|synchronize|reset|unspecified"
              ]
```

Site

Each site is associated with a specific template. The policies from specific templates are pushed to sites based on the site-to-template association.

The siteId is an internal identification pointing to a site defined in Cisco ACI Multi-Site.

Some attributes are site-specific attributes and they are defined for a specific site. For example, site-specific attributes for entities in a template.

Site-specific attributes are as follows:

- epgs have domainAssociations and staticPorts
- bds have subnets
- contracts have serviceGraphTemplate

The following is an example of a site API snippet:

```
"sites": [
    {
      "siteId": "58202f7066e6e10001c41236",
      "templateName": "template1",
      "bds": [
        {
          "bdRef": "/templates/template1/bds/bd1",
          "subnets": [ ... ]
        }
      ]
    },
    {
      "siteId": "69202f7066e6e10001c41247",
      "templateName": "template1",
      "anps": [
        {
          "anpRef": "/templates/template1/anps/anp1",
          "epgs": [
            {
              "epgRef": "/templates/template1/anps/anp1/epgs/epg1",
              "domainAssociations": [
                {
                  "dn": "uni/vmmp-VMware/dom-TestVMMDomain-DoNotDelete",
                  "domainType": "vmmDomain",
                  "deployImmediacy": "On Demand|Immediate",
                  "resolutionImmediacy": "On Demand|Immediate|Pre-provision",
                  "allowMicroSegmentation": true,
                  "allowPromiscuous": true,
                  "macChanges": false,
                  "allowForgedTransmits": true,
                  "delimiter": "delimiter"
                },
                {
                  "dn": "uni/phys-testPhysicalDomain-DoNotDelete",
                  "domainType": "physicalDomain",
                  "deployImmediacy": "On Demand|Immediate",
                  "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
                },
                {
                  "dn": "uni/l2dom-TestL2Domain-DoNotDelete",
                  "domainType": "l2ExtDomain",
                  "deployImmediacy": "On Demand|Immediate",
                  "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
                },
                {
                  "dn": "uni/l3dom-TestL3Domain-DoNotDelete",
                  "domainType": "l3ExtDomain",
                  "deployImmediacy": "On Demand|Immediate",
                  "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
                },
                {
                  "dn": "uni/fc-TestFibreChannelDomain",
                  "domainType": "fibreChannelDomain",
                  "deployImmediacy": "On Demand|Immediate",
                  "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
                }
              ],
              "staticPorts": [
                {
                    "type" : "port",
                    "path" : "topology/pod-1/paths-101/pathep[eth1/11]",
                    "portEncapVlan": 10,
                    "microSegVlan" : 8,
```

```
                    "deploymentImmediacy" : "immediate|lazy",
                    "mode" : "regular|native|untagged"
                }, {
                    "type" : "VPC",
                    "path" : "Topology/pod-1/protpaths-101-102/pathep[NFS-Storage]",
                    "portEncapVlan": 10,
                    "microSegVlan" : 8,
                    "deploymentImmediacy" : "Immediate",
                    "mode" : "Access"
                }, {
                    "type" : "DPC",
                    "path" : "...",
                    "portEncapVlan": 9,
                    "microSegVlan" : 8,
                    "deploymentImmediacy" : "OnDemand",
                    "mode" : "AccessUntagged"
                }
              ],
              "staticLeafs": [
                {
                    "path" : "topology/pod-1/node-201",
                    "portEncapVlan": 10
                }
              ]
            }
          ]
        }
      ],
```

External Authentication

Starting with Cisco ACI Multi-Site release 1.1(1), APIs for external authentication are supported.

Providers

Starting with Release 1.1(1), two types of providers are supported, and they are RADIUS and TACACS+ providers. They have similar data structures.

The attributes for a RADIUS provider are displayed in this API snippet example:

```
{
    "host": "10.23.237.168",
    "description": "ACS RADIUS Provider",
    "sharedSecret": "ins3965!",
    "port": 1812,
    "protocol": "PAP",
    "timeoutInSeconds": 10,
    "retries":5
}
```

The attributes for a TACACS+ provider are displayed in this example:

```
{
    "host": "10.23.237.168",
    "description": "Tacacs Provider - 1",
    "sharedSecret": "ins3965!",
    "port": 49,
    "protocol": "PAP",
```

```
    "timeoutInSeconds": 10,
    "retries":5
}
```

Note The values for the protocol field are as follows:

- CHAP
- PAP

Login Domains

End users can specify a login domain when they log in to the Multi-Site user interface.

The realm field displays the possible provider values. Currently, they can be radius or tacacs.

The following is an example of a login domain API snippet:

```
{
    "domains": [
        {
            "id": "59d6d0398d0000d2009eb2c4",
            "name": "EngineeringGroup",
            "description": "Used for RADIUS Auth Login",
            "realm": "radius",
            "status": "active",
            "providerAssociations": []
        },
        {
            "id": "59d6cfb58d0000bd009eb2c3",
            "name": "VM-TACACS",
            "description": "Used for Tacacs Auth Login",
            "realm": "tacacs",
            "status": "active",
            "providerAssociations": []
        }
    ]
}
```

Remote User Login

Remote user login is a POST API login endpoint. You can create and read remote user login domains with these APIs. Remote user login APIs do not require the authorization header to be set.

The domainID parameter value is used to login after a domain is defined. Defining a domainID is an optional field. For example, the domainID parameter is not specified when the default domain is Local.

The following is an example of a remote user login API snippet:

```
{
"username": "bob",
"password": "we1come!",
"domainId":"59d6d0398d0000d2009eb2c4",
```

```
}
```

Backup

Starting with Cisco ACI Multi-Site release 1.1(1), APIs for backup are supported.

The data structure represents what you require to backup your system. It has fields such as ID, size, timestamps, description etc.

The following is an example of a backup snippet:

```
HTTP/1.1 201
Location: /api/v1/backups/20171116221404
{
    "id": "20171116221404",
    "sizeKB": 18.44,
    "timestamp": "2017-11-16T22:14:04",
    "metadata": {
        "description": "test-description",
        "name": "Check_20171116221404"
    }
}
```

Auditing

Starting with Cisco ACI Multi-Site release 1.1(1), APIs for auditing are supported.

Auditing in the API is exposed through the audit-records endpoint. Through this endpoint, APIs to create, read, filter, and paginate records are available.

Filter Audit Records by **type** are as follows:

Module/Sub-module	Types
Tenants	• tenant • tenant-site • tenant-user

Module/Sub-module	Types
Schemas	• schema • template • schema-site • anp • bd • bd-vrf • epg • epg-bd • epg-contract • external-epg • external-epg-contract • contract • contract-filter • external-epg • filter • vrf
Sites	• site • site-label
Fabric Connectivity	• fabric-connectivity • infra-bgp-config • infra-ospf-policy • infra-site • infra-pod • infra-node • infra-port
Users	• user • user-role • user-domain
Authentication	• user-auth

Module/Sub-module	Types
Login Domains	• domain • domain-provider
Providers	• radius • tacacs
(not module specific)	attribute

The events that are available in auditing APIs are as follows:

- created
- deleted
- updated
- added
- removed
- associated
- disassociated
- deployed
- undeployed
- downloaded
- uploaded
- restored
- logged-in
- login-failed
- logged-out
- login-failed

The following is an example of an auditing API snippet:

```
HTTP/1.1 200
{
  "auditRecords": [
    {
      "id": "59b5e11f1a0100270319b05e",
      "type": "tenant",
      "key": "59b5e11f3400004900c0ea97",
      "name": "Tenant 001",
      "event": "created",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "Admin User",
```

```
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-09-10T01:04:31.701"
    },
    {
      "id": "59b5e11f1a0100270319b05e",
      "type": "tenant",
      "key": "59b5e11f3400004900c0ea97",
      "name": "Tenant 001",
      "event": "updated",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "Admin User",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-09-10T01:04:35.501"
    }
  ]
}
```

The data structure from a schema auditing perspective is as follows: `schema/:id/audit-records/`. The context of a site is applicable when we examine site local attributes from a schema auditing perspective.

The following is an example of a schema auditing API snippet:

```
HTTP/1.1 200

{
  "auditRecords": [
    {
      "id": "59ef87743a000022069baf60",
      "type": "schema",
      "key": "59ef87743800008e06881fe5",
      "name": "Test Schema",
      "event": "created",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
```

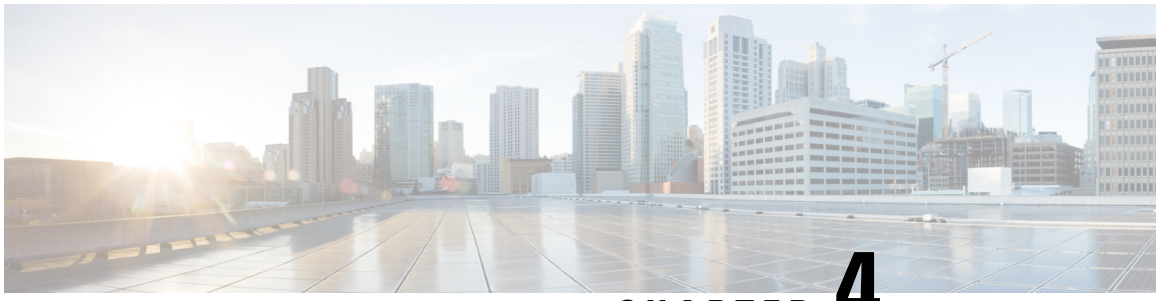
Tenant audit records have a specific usage where type = attribute. The name of the attribute is provided at the end of the string, for example description is provided in the key attribute.

Another specific usage is the following values applied to attributes: oldValue and newValue.

The following is an example of a tenant auditing API snippet:

```
HTTP/1.1 200
{
```

```
 "auditRecords": [
    {
      "id": "59db14881e0000f00071f40e",
      "type": "tenant",
      "key": "59db147b2500000b01c90429",
      "name": "Tenant 001",
      "event": "updated",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-09T06:17:44.141"
    },
  ]
}
```



CHAPTER 4

REST Examples

This chapter contains the following sections:

- REST API Examples , on page 29
- APIs for Consistency Checker, on page 30
- APIs for Site Local Properties, on page 50
- APIs For Local Authentication, on page 55
- APIs For External Authentication, on page 59
- APIs For Labels, on page 70
- APIs For Roles, on page 72
- APIs For Users, on page 74
- APIs For Fabric Connectivity, on page 80
- Fabric Leaf Switches And Physical Interfaces, on page 105
- APIs For Tenants, on page 107
- APIs For All Schemas, on page 115
- APIs For Import, on page 193
- APIs For Sites, on page 204
- APIs For Template, on page 209
- APIs For Entities, on page 210
- APIs For Platforms, on page 211
- Expected HTTP Response Code Examples, on page 212
- APIs For Audit, on page 213
- APIs For Backups, on page 240

REST API Examples

Note The word Schema is alternately known as Profile in this guide.

For the examples in this guide, if an example does not specify a release number in the heading, it is applicable to release 1.0(1) and release 1.0(2).

APIs for Consistency Checker

About Consistency Checker

Starting with Cisco ACI Multi-Site, release 1.2(1), the APIs for Consistency Checker are supported. Therefore, the Consistency Checker examples are applicable starting with release 1.2(1) only.

The Multi-Site solution enables extending the policy domain end-to-end across fabrics, where you can create policies and push them to all sites or selected sites. Consistency Checker is used to verify that those policies have been deployed correctly and consistently across those sites.

The following APIs are used for the Consistency Checker:

- Status APIs — Provides the current status of a deployment of a policy across multiple sites.
- Report APIs — Provides a full report, including the status, of what succeeded or failed in the deployment of a policy across multiple sites.

Download of File of APIC's Response (Raw JSON)

This is an example of how to download a file of the APIC's response, in raw JSON format.

You should have permissions to manage tenantSchemas and schemas.

GET:

```
GET /api/v1/consistency/schema/:schemaId/template/:templateName/site/:siteId/download
```

Trigger an APIC Consistency Service for a Schema

This is an example of an asynchronous POST request that triggers Consistency Checker for each template of the schema. The responses are saved in consistency collection.

You should have permissions to manage tenantSchemas and schemas.

POST:

```
POST /api/v1/consistency/schema-trigger
```

Request:

```
{
    "schemaId": "5a610997240000a20075b9f7"
}
```

Response:

```
{
    "result": "ok"
}
```

Trigger an APIC Consistency Service for a Template

The following is an example of an asynchronous POST request that triggers Consistency Checker on each APIC the template is deployed to. The responses are saved in consistency collection.

You should have permissions to manage tenantSchemas and schemas.

POST:

```
POST  /api/v1/consistency/trigger
```

Request:

```
{
    "schemaId": "5a610997240000a20075b9f7",
    "templateName": "Template1"
}
```

Response:

```
{
    "result": "ok"
}
```

Read the Consistency Status for a Schema

The following is an example of how to fetch consistency reports for all templates of the schema.

You should have permissions to manage tenantSchemas and schemas.

GET:

```
GET /api/v1/consistency/schema/:schemaId/status
```

Request:

```
<empty>
```

Response:

```
{
    "consistencyStatus": [
        {
            "schemaId": "5a78b5ef200000dd006f2f99",
            "templateName": "Template1",
            "siteId": "5a6b8d975c000045010f1c17",
            "finalResult": {
                "status": "PASSED"
            },
            "timestamp": "2018-02-07T10:24:11.288"
        },
        {
            "schemaId": "5a78b5ef200000dd006f2f99",
            "templateName": "Template1",
            "siteId": "5a6b8d975c000045010f1c18",
            "finalResult": {
                "status": "PASSED"
            },
```

```
            "timestamp": "2018-02-07T10:24:11.288"
        }
    ]
}
```

Read the Consistency Status for a Template

The following is an example of how to fetch consistency reports for all of the sites that the template is deployed to.

You should have permissions to manage tenantSchemas and schemas.

GET:

```
GET  /api/v1/consistency/schema/:schemaId/template/:templateName/status
```

Request:

```
<empty>
```

Response:

```
{
    "consistencyStatus": [
        {
            "schemaId": "5a78b5ef200000dd006f2f99",
            "templateName": "Template1",
            "siteId": "5a6b8d975c000045010f1c17",
            "finalResult": {
                "status": "PASSED"
            },
            "timestamp": "2018-02-07T10:24:11.288"
        },
        {
            "schemaId": "5a78b5ef200000dd006f2f99",
            "templateName": "Template1",
            "siteId": "5a6b8d975c000045010f1c18",
            "finalResult": {
                "status": "PASSED"
            },
            "timestamp": "2018-02-07T10:24:11.288"
        }
    ]
}
```

Read the Consistency Report for a Schema

The following is an example of how to fetch consistency reports for all templates of the schema.

You should have permissions to manage tenantSchemas and schemas.

GET:

```
GET  /api/v1/consistency/schema/:schemaId/report
```

Request:

```
<empty>
```

Note In the following response example, long single lines of text are broken up with the ^ character to improve readability.

Response:


```
{
    "consistencyReport": [
        {
            "id": "5a793a998e00009000ac7564",
            "schemaId": "5a78b5ef200000dd006f2f99",
            "templateName": "Template1",
            "siteId": "5a6b8d975c000045010f1c17",
            "apicRawData": "{\"Final result\":\"PASSED\",\"Translation Report\": ^
            {\"bds\":[{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
            [{\"Local site-2\":{\"Spines\":\"?\",\"ctxDef/BdDef\":{\"encapOrPcTag\": ^
            \"3080219/15958055\",\"pcTagStatus\":\"OK\"},\"logical ctx/Bd\": ^
            {\"encapOrPcTag\":\"3080219/15958055\",\"pcTagStatus\":\"OK\"},\"msc\": ^
            {\"encapOrPcTag\":\"3080219/15958055\",\"pcTagStatus\":\"OK\"}}, ^
            \"Remote site-1\":{\"Spines\":\"?\",\"ctxDef/BdDef\":{\"encapOrPcTag\": ^
            \"2654245/16482249\",\"pcTagStatus\":\"OK\"},\"logical ctx/Bd\": ^
            {\"encapOrPcTag\":\"2654245/16482249\",\"pcTagStatus\":\"OK\"},\"msc\": ^
            {\"encapOrPcTag\":\"2654245/16482249\",\"pcTagStatus\":\"OK\"}}}], ^
            \"common\":{\"Context\": ^
            \"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"},\"dn\": ^
            \"uni/tn-CC-Test-Tenant-DoNotDelete/BD-BD-EPG-ONLY\",\"mscRef\": ^
            \"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/bds/BD-EPG-ONLY\"}], ^
            \"epgs\":[{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
            [{\"Local site-2\":{\"Spines\":\"?\",\"ctxDef/epgDef\":{\"encapOrPcTag\": ^
            \"3080219/16386\",\"pcTagStatus\":\"OK\"},\"logical ctx/epg\": ^
            {\"encapOrPcTag\":\"3080219/16386\",\"pcTagStatus\":\"OK\"},\"msc\": ^
            {\"encapOrPcTag\":\"3080219/16386\",\"pcTagStatus\":\"OK\"}}, ^
            \"Remote site-1\":{\"Spines\":\"?\",\"ctxDef/epgDef\":{\"encapOrPcTag\": ^
            \"2654245/32770\",\"pcTagStatus\":\"OK\"},\"logical ctx/epg\": ^
            {\"encapOrPcTag\":\"2654245/32770\",\"pcTagStatus\":\"OK\"},\"msc\": ^
            {\"encapOrPcTag\":\"2654245/32770\",\"pcTagStatus\":\"OK\"}}}],\"common\": ^
            {\"Context\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"},\"dn\": ^
            \"uni/tn-CC-Test-Tenant-DoNotDelete/ap-ANP-EPG-ONLY/epg-EPG-ONLY\", ^
            \"mscRef\":\"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/anps/ ^
            ANP-EPG-ONLY/epgs/EPG-ONLY\"},{\"Overall Status\":\"NOT OK\", ^
            \"Sites Report\":[{\"Local site-2\":{\"Spines\":\"?\",\"ctxDef/epgDef\": ^
            {\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\":\"OK\"}, ^
            \"logical ctx/epg\":{\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\": ^
            \"OK\"},\"msc\":{\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\": ^
            \"OK\"}},\"Remote site-1\":{\"Spines\":\"?\",\"ctxDef/epgDef\": ^
            {\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\":\"OK\"}, ^
            \"logical ctx/epg\":{\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\": ^
            \"OK\"},\"msc\":{\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\": ^
            \"OK\"}}}],\"common\":{\"Context\": ^
            \"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"},\"dn\": ^
            \"uni/tn-CC-Test-Tenant-DoNotDelete/ap-ANP-EPG-ONLY ^
            /epg-UntitledEPG1Consumer\",\"mscRef\": ^
            \"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/anps ^
            /ANP-EPG-ONLY/epgs/UntitledEPG1Consumer\"}],\"vrfs\": ^
            [{\"Overall Status\":\"NOT OK\",\"Sites Report\":[{\"Local site-2\": ^
            {\"Spines\":\"?\",\"ctx Def\":{\"encapOrPcTag\":\"3080219\", ^
            \"pcTagStatus\":\"OK\"},\"logical ctx\":{\"encapOrPcTag\":\"3080219\", ^
```

```
\"pcTagStatus\":\"OK\"},\"msc\":{\"encapOrPcTag\":\"3080219\", ^
\"pcTagStatus\":\"OK\"}},\"Remote site-1\":{\"Spines\":\"?\",\"ctx Def\": ^
{\"encapOrPcTag\":\"2654245\",\"pcTagStatus\":\"OK\"},\"logical ctx\": ^
{\"encapOrPcTag\":\"2654245\",\"pcTagStatus\":\"OK\"},\"msc\": ^
{\"encapOrPcTag\":\"2654245\",\"pcTagStatus\":\"OK\"}}}],\"common\": ^
{\"Context\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"}, ^
\"dn\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\",\"mscRef\": ^
\"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/vrfs/ ^
VRF-EPG-ONLY\"}]}}",
"finalResult": {
    "status": "PASSED"
},
"policies": {
    "bds": [
        {
            "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
             Template1/bds/BD-EPG-ONLY",
            "sites": [
                {
                    "apic": true,
                    "siteId": "5a6b8d975c000045010f1c17",
                    "siteName": "Site2",
                    "isLocal": true,
                    "switch": true
                },
                {
                    "apic": true,
                    "siteId": "5a6b8e7a5c0000a2010f1c18",
                    "siteName": "Test",
                    "isLocal": false,
                    "switch": true
                }
            ]
        }
    ],
    "epgs": [
        {
            "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
             Template1/anps/ANP-EPG-ONLY/epgs/EPG-ONLY",
            "sites": [
                {
                    "apic": true,
                    "siteId": "5a6b8d975c000045010f1c17",
                    "siteName": "Site2",
                    "isLocal": true,
                    "switch": true
                },
                {
                    "apic": true,
                    "siteId": "5a6b8e7a5c0000a2010f1c18",
                    "siteName": "Test",
                    "isLocal": false,
                    "switch": true
                }
            ]
        },
        {
            "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
             Template1/anps/ANP-EPG-ONLY/epgs/UntitledEPG1Consumer",
            "sites": [
                {
                    "apic": true,
                    "siteId": "5a6b8d975c000045010f1c17",
                    "siteName": "Site2",
```

```
                            "isLocal": true,
                            "switch": true
                        },
                        {
                            "apic": true,
                            "siteId": "5a6b8e7a5c0000a2010f1c18",
                            "siteName": "Test",
                            "isLocal": false,
                            "switch": true
                        }
                    ]
                }
            ],
            "vrfs": [
                {
                    "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates ^
                     /Template1/vrfs/VRF-EPG-ONLY",
                    "sites": [
                        {
                            "apic": true,
                            "siteId": "5a6b8d975c000045010f1c17",
                            "siteName": "Site2",
                            "isLocal": true,
                            "switch": true
                        },
                        {
                            "apic": true,
                            "siteId": "5a6b8e7a5c0000a2010f1c18",
                            "siteName": "Test",
                            "isLocal": false,
                            "switch": true
                        }
                    ]
                }
            ],
            "externalEpgs": []
        },
        "timestamp": "2018-02-07T10:24:11.288"
    },
    {
        "id": "5a793a998e00008e00ac7563",
        "schemaId": "5a78b5ef200000dd006f2f99",
        "templateName": "Template1",
        "siteId": "5a6b8e7a5c0000a2010f1c18",
        "apicRawData": "{\"Final result\":\"PASSED\",\"Translation Report\": ^
         {\"bds\":[{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
         [{\"Local site-1\":{\"Spines\":\"?\",\"ctxDef/BdDef\":{\"encapOrPcTag\": ^
         \"2654245/16482249\",\"pcTagStatus\":\"OK\"},\"logical ctx/Bd\": ^
         {\"encapOrPcTag\":\"2654245/16482249\",\"pcTagStatus\":\"OK\"},\"msc\": ^
         {\"encapOrPcTag\":\"2654245/16482249\",\"pcTagStatus\":\"OK\"}}, ^
         \"Remote site-2\":{\"Spines\":\"?\",\"ctxDef/BdDef\":{\"encapOrPcTag\": ^
         \"3080219/15958055\",\"pcTagStatus\":\"OK\"},\"logical ctx/Bd\": ^
         {\"encapOrPcTag\":\"3080219/15958055\",\"pcTagStatus\":\"OK\"},\"msc\": ^
         {\"encapOrPcTag\":\"3080219/15958055\",\"pcTagStatus\":\"OK\"}}}], ^
         \"common\":{\"Context\": ^
         \"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"},\"dn\": ^
         \"uni/tn-CC-Test-Tenant-DoNotDelete/BD-BD-EPG-ONLY\",\"mscRef\": ^
         \"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/bds/ ^
         BD-EPG-ONLY\"}],\"epgs\":[{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
         [{\"Local site-1\":{\"Spines\":\"?\",\"ctxDef/epgDef\":{\"encapOrPcTag\": ^
         \"2654245/32770\",\"pcTagStatus\":\"OK\"},\"logical ctx/epg\": ^
         {\"encapOrPcTag\":\"2654245/32770\",\"pcTagStatus\":\"OK\"},\"msc\": ^
         {\"encapOrPcTag\":\"2654245/32770\",\"pcTagStatus\":\"OK\"}}, ^
         \"Remote site-2\":{\"Spines\":\"?\",\"ctxDef/epgDef\":{\"encapOrPcTag\": ^
```

```
 \"3080219/16386\",\"pcTagStatus\":\"OK\"},\"logical ctx/epg\": ^
 {\"encapOrPcTag\":\"3080219/16386\",\"pcTagStatus\":\"OK\"},\"msc\": ^
 {\"encapOrPcTag\":\"3080219/16386\",\"pcTagStatus\":\"OK\"}}}],\"common\": ^
 {\"Context\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"}, ^
 \"dn\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ap-ANP-EPG-ONLY/epg-EPG-ONLY\", ^
 \"mscRef\":\"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/anps ^
 /ANP-EPG-ONLY/epgs/EPG-ONLY\"},{\"Overall Status\":\"NOT OK\", ^
 \"Sites Report\":[{\"Local site-1\":{\"Spines\":\"?\",\"ctxDef/epgDef\": ^
 {\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\":\"OK\"}, ^
 \"logical ctx/epg\":{\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\": ^
 \"OK\"},\"msc\":{\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\": ^
 \"OK\"}},\"Remote site-2\":{\"Spines\":\"?\",\"ctxDef/epgDef\": ^
 {\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\":\"OK\"}, ^
 \"logical ctx/epg\":{\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\": ^
 \"OK\"},\"msc\":{\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\": ^
 \"OK\"}}}],\"common\":{\"Context\": ^
 \"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"},\"dn\": ^
 \"uni/tn-CC-Test-Tenant-DoNotDelete/ap-ANP-EPG-ONLY/ ^
 epg-UntitledEPG1Consumer\",\"mscRef\":\"/schemas/5a78b5ef200000dd006f2f99/ ^
 templates/Template1/anps/ANP-EPG-ONLY/epgs/UntitledEPG1Consumer\"}], ^
 \"vrfs\":[{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
 [{\"Local site-1\":{\"Spines\":\"?\",\"ctx Def\":{\"encapOrPcTag\": ^
 \"2654245\",\"pcTagStatus\":\"OK\"},\"logical ctx\":{\"encapOrPcTag\": ^
 \"2654245\",\"pcTagStatus\":\"OK\"},\"msc\":{\"encapOrPcTag\":\"2654245\", ^
 \"pcTagStatus\":\"OK\"}},\"Remote site-2\":{\"Spines\":\"?\",\"ctx Def\": ^
 {\"encapOrPcTag\":\"3080219\",\"pcTagStatus\":\"OK\"},\"logical ctx\": ^
 {\"encapOrPcTag\":\"3080219\",\"pcTagStatus\":\"OK\"},\"msc\": ^
 {\"encapOrPcTag\":\"3080219\",\"pcTagStatus\":\"OK\"}}}],\"common\": ^
 {\"Context\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"}, ^
 \"dn\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\",\"mscRef\": ^
 \"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/vrfs/ ^
 VRF-EPG-ONLY\"}]}}",
"finalResult": {
    "status": "PASSED"
},
"policies": {
    "bds": [
        {
            "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
             Template1/bds/BD-EPG-ONLY",
            "sites": [
                {
                    "apic": true,
                    "siteId": "5a6b8e7a5c0000a2010f1c18",
                    "siteName": "Test",
                    "isLocal": true,
                    "switch": true
                },
                {
                    "apic": true,
                    "siteId": "5a6b8d975c000045010f1c17",
                    "siteName": "Site2",
                    "isLocal": false,
                    "switch": true
                }
            ]
        }
    ],
    "epgs": [
        {
            "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
             Template1/anps/ANP-EPG-ONLY/epgs/EPG-ONLY",
            "sites": [
                {
```

```
                                "apic": true,
                                "siteId": "5a6b8e7a5c0000a2010f1c18",
                                "siteName": "Test",
                                "isLocal": true,
                                "switch": true
                            },
                            {
                                "apic": true,
                                "siteId": "5a6b8d975c000045010f1c17",
                                "siteName": "Site2",
                                "isLocal": false,
                                "switch": true
                            }
                        ]
                    },
                    {
                        "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
                         Template1/anps/ANP-EPG-ONLY/epgs/UntitledEPG1Consumer",
                        "sites": [
                            {
                                "apic": true,
                                "siteId": "5a6b8e7a5c0000a2010f1c18",
                                "siteName": "Test",
                                "isLocal": true,
                                "switch": true
                            },
                            {
                                "apic": true,
                                "siteId": "5a6b8d975c000045010f1c17",
                                "siteName": "Site2",
                                "isLocal": false,
                                "switch": true
                            }
                        ]
                    }
                ],
                "vrfs": [
                    {
                        "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
                         Template1/vrfs/VRF-EPG-ONLY",
                        "sites": [
                            {
                                "apic": true,
                                "siteId": "5a6b8e7a5c0000a2010f1c18",
                                "siteName": "Test",
                                "isLocal": true,
                                "switch": true
                            },
                            {
                                "apic": true,
                                "siteId": "5a6b8d975c000045010f1c17",
                                "siteName": "Site2",
                                "isLocal": false,
                                "switch": true
                            }
                        ]
                    }
                ],
                "externalEpgs": []
            },
            "timestamp": "2018-02-07T10:24:11.29"
        }
]
```

```
}
```

Read the Consistency Report for a Template

The following is an example of how to fetch a consistency report for a template for all sites that it is deployed to.

You should have permissions to manage tenantSchemas and schemas.

GET:

```
GET  /api/v1/consistency/schema/:schemaId/template/:templateName/report
```

Request:

```
<empty>
```

Note In the following response example, long single lines of text are broken up with the ^ character to improve readability.

Response:

```
{
    "consistencyReport": [
        {
            "id": "5a793a998e00009000ac7564",
            "schemaId": "5a78b5ef200000dd006f2f99",
            "templateName": "Template1",
            "siteId": "5a6b8d975c000045010f1c17",
            "apicRawData": "{\"Final result\":\"PASSED\",\"Translation Report\": ^
            {\"bds\":[{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
            [{\"Local site-2\":{\"Spines\":\"?\",\"ctxDef/BdDef\":{\"encapOrPcTag\": ^
            \"3080219/15958055\",\"pcTagStatus\":\"OK\"},\"logical ctx/Bd\": ^
            {\"encapOrPcTag\":\"3080219/15958055\",\"pcTagStatus\":\"OK\"},\"msc\": ^
            {\"encapOrPcTag\":\"3080219/15958055\",\"pcTagStatus\":\"OK\"}}, ^
            \"Remote site-1\":{\"Spines\":\"?\",\"ctxDef/BdDef\":{\"encapOrPcTag\": ^
            \"2654245/16482249\",\"pcTagStatus\":\"OK\"},\"logical ctx/Bd\": ^
            {\"encapOrPcTag\":\"2654245/16482249\",\"pcTagStatus\":\"OK\"},\"msc\": ^
            {\"encapOrPcTag\":\"2654245/16482249\",\"pcTagStatus\":\"OK\"}}}], ^
            \"common\":{\"Context\": ^
            \"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"},\"dn\": ^
            \"uni/tn-CC-Test-Tenant-DoNotDelete/BD-BD-EPG-ONLY\",\"mscRef\": ^
            \"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/bds/BD-EPG-ONLY\"}], ^
            \"epgs\":[{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
            [{\"Local site-2\":{\"Spines\":\"?\",\"ctxDef/epgDef\":{\"encapOrPcTag\": ^
            \"3080219/16386\",\"pcTagStatus\":\"OK\"},\"logical ctx/epg\": ^
            {\"encapOrPcTag\":\"3080219/16386\",\"pcTagStatus\":\"OK\"},\"msc\": ^
            {\"encapOrPcTag\":\"3080219/16386\",\"pcTagStatus\":\"OK\"}}, ^
            \"Remote site-1\":{\"Spines\":\"?\",\"ctxDef/epgDef\":{\"encapOrPcTag\": ^
            \"2654245/32770\",\"pcTagStatus\":\"OK\"},\"logical ctx/epg\": ^
            {\"encapOrPcTag\":\"2654245/32770\",\"pcTagStatus\":\"OK\"},\"msc\": ^
            {\"encapOrPcTag\":\"2654245/32770\",\"pcTagStatus\":\"OK\"}}}],\"common\": ^
            {\"Context\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"},\"dn\": ^
            \"uni/tn-CC-Test-Tenant-DoNotDelete/ap-ANP-EPG-ONLY/epg-EPG-ONLY\", ^
            \"mscRef\":\"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/anps/ ^
            ANP-EPG-ONLY/epgs/EPG-ONLY\"},{\"Overall Status\":\"NOT OK\", ^
```

```
\"Sites Report\":[{\"Local site-2\":{\"Spines\":\"?\",\"ctxDef/epgDef\": ^
{\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\":\"OK\"}, ^
\"logical ctx/epg\":{\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\": ^
\"OK\"},\"msc\":{\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\": ^
\"OK\"}},\"Remote site-1\":{\"Spines\":\"?\",\"ctxDef/epgDef\": ^
{\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\":\"OK\"}, ^
\"logical ctx/epg\":{\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\": ^
\"OK\"},\"msc\":{\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\": ^
\"OK\"}}}],\"common\":{\"Context\": ^
\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"},\"dn\": ^
\"uni/tn-CC-Test-Tenant-DoNotDelete/ap-ANP-EPG-ONLY ^
/epg-UntitledEPG1Consumer\",\"mscRef\": ^
\"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/anps ^
/ANP-EPG-ONLY/epgs/UntitledEPG1Consumer\"}],\"vrfs\": ^
[{\"Overall Status\":\"NOT OK\",\"Sites Report\":[{\"Local site-2\": ^
{\"Spines\":\"?\",\"ctx Def\":{\"encapOrPcTag\":\"3080219\", ^
\"pcTagStatus\":\"OK\"},\"logical ctx\":{\"encapOrPcTag\":\"3080219\", ^
\"pcTagStatus\":\"OK\"},\"msc\":{\"encapOrPcTag\":\"3080219\", ^
\"pcTagStatus\":\"OK\"}},\"Remote site-1\":{\"Spines\":\"?\",\"ctx Def\": ^
{\"encapOrPcTag\":\"2654245\",\"pcTagStatus\":\"OK\"},\"logical ctx\": ^
{\"encapOrPcTag\":\"2654245\",\"pcTagStatus\":\"OK\"},\"msc\": ^
{\"encapOrPcTag\":\"2654245\",\"pcTagStatus\":\"OK\"}}}],\"common\": ^
{\"Context\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"}, ^
\"dn\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\",\"mscRef\": ^
\"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/vrfs/ ^
VRF-EPG-ONLY\"}]}}",
"finalResult": {
    "status": "PASSED"
},
"policies": {
    "bds": [
        {
            "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
             Template1/bds/BD-EPG-ONLY",
            "sites": [
                {
                    "apic": true,
                    "siteId": "5a6b8d975c000045010f1c17",
                    "siteName": "Site2",
                    "isLocal": true,
                    "switch": true
                },
                {
                    "apic": true,
                    "siteId": "5a6b8e7a5c0000a2010f1c18",
                    "siteName": "Test",
                    "isLocal": false,
                    "switch": true
                }
            ]
        }
    ],
    "epgs": [
        {
            "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
             Template1/anps/ANP-EPG-ONLY/epgs/EPG-ONLY",
            "sites": [
                {
                    "apic": true,
                    "siteId": "5a6b8d975c000045010f1c17",
                    "siteName": "Site2",
                    "isLocal": true,
                    "switch": true
                },
```

```
                        {
                            "apic": true,
                            "siteId": "5a6b8e7a5c0000a2010f1c18",
                            "siteName": "Test",
                            "isLocal": false,
                            "switch": true
                        }
                    ]
                },
                {
                    "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
                     Template1/anps/ANP-EPG-ONLY/epgs/UntitledEPG1Consumer",
                    "sites": [
                        {
                            "apic": true,
                            "siteId": "5a6b8d975c000045010f1c17",
                            "siteName": "Site2",
                            "isLocal": true,
                            "switch": true
                        },
                        {
                            "apic": true,
                            "siteId": "5a6b8e7a5c0000a2010f1c18",
                            "siteName": "Test",
                            "isLocal": false,
                            "switch": true
                        }
                    ]
                }
            ],
            "vrfs": [
                {
                    "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates ^
                     /Template1/vrfs/VRF-EPG-ONLY",
                    "sites": [
                        {
                            "apic": true,
                            "siteId": "5a6b8d975c000045010f1c17",
                            "siteName": "Site2",
                            "isLocal": true,
                            "switch": true
                        },
                        {
                            "apic": true,
                            "siteId": "5a6b8e7a5c0000a2010f1c18",
                            "siteName": "Test",
                            "isLocal": false,
                            "switch": true
                        }
                    ]
                }
            ],
            "externalEpgs": []
        },
        "timestamp": "2018-02-07T10:24:11.288"
    },
    {
        "id": "5a793a998e00008e00ac7563",
        "schemaId": "5a78b5ef200000dd006f2f99",
        "templateName": "Template1",
        "siteId": "5a6b8e7a5c0000a2010f1c18",
        "apicRawData": "{\"Final result\":\"PASSED\",\"Translation Report\": ^
         {\"bds\":[{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
         [{\"Local site-1\":{\"Spines\":\"?\",\"ctxDef/BdDef\":{\"encapOrPcTag\": ^
```

```
\"2654245/16482249\",\"pcTagStatus\":\"OK\"},\"logical ctx/Bd\": ^
{\"encapOrPcTag\":\"2654245/16482249\",\"pcTagStatus\":\"OK\"},\"msc\": ^
{\"encapOrPcTag\":\"2654245/16482249\",\"pcTagStatus\":\"OK\"}}, ^
\"Remote site-2\":{\"Spines\":\"?\",\"ctxDef/BdDef\":{\"encapOrPcTag\": ^
\"3080219/15958055\",\"pcTagStatus\":\"OK\"},\"logical ctx/Bd\": ^
{\"encapOrPcTag\":\"3080219/15958055\",\"pcTagStatus\":\"OK\"},\"msc\": ^
{\"encapOrPcTag\":\"3080219/15958055\",\"pcTagStatus\":\"OK\"}}}], ^
\"common\":{\"Context\": ^
\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"},\"dn\": ^
\"uni/tn-CC-Test-Tenant-DoNotDelete/BD-BD-EPG-ONLY\",\"mscRef\": ^
\"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/bds/ ^
BD-EPG-ONLY\"}],\"epgs\":[{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
[{\"Local site-1\":{\"Spines\":\"?\",\"ctxDef/epgDef\":{\"encapOrPcTag\": ^
\"2654245/32770\",\"pcTagStatus\":\"OK\"},\"logical ctx/epg\": ^
{\"encapOrPcTag\":\"2654245/32770\",\"pcTagStatus\":\"OK\"},\"msc\": ^
{\"encapOrPcTag\":\"2654245/32770\",\"pcTagStatus\":\"OK\"}}, ^
\"Remote site-2\":{\"Spines\":\"?\",\"ctxDef/epgDef\":{\"encapOrPcTag\": ^
\"3080219/16386\",\"pcTagStatus\":\"OK\"},\"logical ctx/epg\": ^
{\"encapOrPcTag\":\"3080219/16386\",\"pcTagStatus\":\"OK\"},\"msc\": ^
{\"encapOrPcTag\":\"3080219/16386\",\"pcTagStatus\":\"OK\"}}}],\"common\": ^
{\"Context\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"}, ^
\"dn\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ap-ANP-EPG-ONLY/epg-EPG-ONLY\", ^
\"mscRef\":\"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/anps ^
/ANP-EPG-ONLY/epgs/EPG-ONLY\"},{\"Overall Status\":\"NOT OK\", ^
\"Sites Report\":[{\"Local site-1\":{\"Spines\":\"?\",\"ctxDef/epgDef\": ^
{\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\":\"OK\"}, ^
\"logical ctx/epg\":{\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\": ^
\"OK\"},\"msc\":{\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\": ^
\"OK\"}},\"Remote site-2\":{\"Spines\":\"?\",\"ctxDef/epgDef\": ^
{\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\":\"OK\"}, ^
\"logical ctx/epg\":{\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\": ^
\"OK\"},\"msc\":{\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\": ^
\"OK\"}}}],\"common\":{\"Context\": ^
\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"},\"dn\": ^
\"uni/tn-CC-Test-Tenant-DoNotDelete/ap-ANP-EPG-ONLY/ ^
epg-UntitledEPG1Consumer\",\"mscRef\":\"/schemas/5a78b5ef200000dd006f2f99/ ^
templates/Template1/anps/ANP-EPG-ONLY/epgs/UntitledEPG1Consumer\"}], ^
\"vrfs\":[{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
[{\"Local site-1\":{\"Spines\":\"?\",\"ctx Def\":{\"encapOrPcTag\": ^
\"2654245\",\"pcTagStatus\":\"OK\"},\"logical ctx\":{\"encapOrPcTag\": ^
\"2654245\",\"pcTagStatus\":\"OK\"},\"msc\":{\"encapOrPcTag\":\"2654245\", ^
\"pcTagStatus\":\"OK\"}},\"Remote site-2\":{\"Spines\":\"?\",\"ctx Def\": ^
{\"encapOrPcTag\":\"3080219\",\"pcTagStatus\":\"OK\"},\"logical ctx\": ^
{\"encapOrPcTag\":\"3080219\",\"pcTagStatus\":\"OK\"},\"msc\": ^
{\"encapOrPcTag\":\"3080219\",\"pcTagStatus\":\"OK\"}}}],\"common\": ^
{\"Context\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"}, ^
\"dn\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\",\"mscRef\": ^
\"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/vrfs/ ^
VRF-EPG-ONLY\"}]}}",
"finalResult": {
    "status": "PASSED"
},
"policies": {
    "bds": [
        {
            "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
             Template1/bds/BD-EPG-ONLY",
            "sites": [
                {
                    "apic": true,
                    "siteId": "5a6b8e7a5c0000a2010f1c18",
                    "siteName": "Test",
                    "isLocal": true,
                    "switch": true
```

```
                },
                {
                    "apic": true,
                    "siteId": "5a6b8d975c000045010f1c17",
                    "siteName": "Site2",
                    "isLocal": false,
                    "switch": true
                }
            ]
        }
    ],
    "epgs": [
        {
            "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
             Template1/anps/ANP-EPG-ONLY/epgs/EPG-ONLY",
            "sites": [
                {
                    "apic": true,
                    "siteId": "5a6b8e7a5c0000a2010f1c18",
                    "siteName": "Test",
                    "isLocal": true,
                    "switch": true
                },
                {
                    "apic": true,
                    "siteId": "5a6b8d975c000045010f1c17",
                    "siteName": "Site2",
                    "isLocal": false,
                    "switch": true
                }
            ]
        },
        {
            "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
             Template1/anps/ANP-EPG-ONLY/epgs/UntitledEPG1Consumer",
            "sites": [
                {
                    "apic": true,
                    "siteId": "5a6b8e7a5c0000a2010f1c18",
                    "siteName": "Test",
                    "isLocal": true,
                    "switch": true
                },
                {
                    "apic": true,
                    "siteId": "5a6b8d975c000045010f1c17",
                    "siteName": "Site2",
                    "isLocal": false,
                    "switch": true
                }
            ]
        }
    ],
    "vrfs": [
        {
            "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/ ^
             Template1/vrfs/VRF-EPG-ONLY",
            "sites": [
                {
                    "apic": true,
                    "siteId": "5a6b8e7a5c0000a2010f1c18",
                    "siteName": "Test",
                    "isLocal": true,
                    "switch": true
```

```
                                },
                                {
                                    "apic": true,
                                    "siteId": "5a6b8d975c000045010f1c17",
                                    "siteName": "Site2",
                                    "isLocal": false,
                                    "switch": true
                                }
                            ]
                        }
                    ],
                    "externalEpgs": []
                },
                "timestamp": "2018-02-07T10:24:11.29"
            }
    ]
}
```

Read the Consistency Report for a Template for a Specific Site

The following is an example of how to fetch a consistency report for a template for the requested site.

You should have permissions to manage tenantSchemas and schemas.

GET:

```
GET /api/v1/consistency/schema/:schemaId/template/:templateName/site/:siteId/report
```

Request:

```
<empty>
```

Note In the following response example, long single lines of text are broken up with the ^ character to improve readability.

Response:

```
{
            "id": "5a793a998e00008e00ac7563",
            "schemaId": "5a78b5ef200000dd006f2f99",
            "templateName": "Template1",
            "siteId": "5a6b8e7a5c0000a2010f1c18",
            "apicRawData": "{\"Final result\":\"PASSED\",\"Translation Report\": ^
             {\"bds\":[{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
             [{\"Local site-1\":{\"Spines\":\"?\",\"ctxDef/BdDef\":{\"encapOrPcTag\": ^
             \"2654245/16482249\",\"pcTagStatus\":\"OK\"},\"logical ctx/Bd\": ^
             {\"encapOrPcTag\":\"2654245/16482249\",\"pcTagStatus\":\"OK\"},\"msc\": ^
             {\"encapOrPcTag\":\"2654245/16482249\",\"pcTagStatus\":\"OK\"}}, ^
             \"Remote site-2\":{\"Spines\":\"?\",\"ctxDef/BdDef\":{\"encapOrPcTag\": ^
             \"3080219/15958055\",\"pcTagStatus\":\"OK\"},\"logical ctx/Bd\": ^
             {\"encapOrPcTag\":\"3080219/15958055\",\"pcTagStatus\":\"OK\"},\"msc\": ^
             {\"encapOrPcTag\":\"3080219/15958055\",\"pcTagStatus\":\"OK\"}}}], ^
             \"common\":{\"Context\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ ^
             ctx-VRF-EPG-ONLY\"},\"dn\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ ^
             BD-BD-EPG-ONLY\",\"mscRef\":\"/schemas/5a78b5ef200000dd006f2f99/templates/ ^
             Template1/bds/BD-EPG-ONLY\"}],\"epgs\":[{\"Overall Status\":\"NOT OK\", ^
```

```
\"Sites Report\":[{\"Local site-1\":{\"Spines\":\"?\",\"ctxDef/epgDef\": ^
{\"encapOrPcTag\":\"2654245/32770\",\"pcTagStatus\":\"OK\"}, ^
\"logical ctx/epg\":{\"encapOrPcTag\":\"2654245/32770\",\"pcTagStatus\": ^
\"OK\"},\"msc\":{\"encapOrPcTag\":\"2654245/32770\",\"pcTagStatus\": ^
\"OK\"}},\"Remote site-2\":{\"Spines\":\"?\",\"ctxDef/epgDef\": ^
{\"encapOrPcTag\":\"3080219/16386\",\"pcTagStatus\":\"OK\"}, ^
\"logical ctx/epg\":{\"encapOrPcTag\":\"3080219/16386\",\"pcTagStatus\": ^
\"OK\"},\"msc\":{\"encapOrPcTag\":\"3080219/16386\",\"pcTagStatus\": ^
\"OK\"}}}],\"common\":{\"Context\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ ^
ctx-VRF-EPG-ONLY\"},\"dn\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ ^
ap-ANP-EPG-ONLY/epg-EPG-ONLY\",\"mscRef\": ^
\"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/anps/ANP-EPG-ONLY/ ^
epgs/EPG-ONLY\"},{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
[{\"Local site-1\":{\"Spines\":\"?\",\"ctxDef/epgDef\":{\"encapOrPcTag\": ^
\"2654245/32771\",\"pcTagStatus\":\"OK\"},\"logical ctx/epg\": ^
{\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\":\"OK\"},\"msc\": ^
{\"encapOrPcTag\":\"2654245/32771\",\"pcTagStatus\":\"OK\"}}, ^
\"Remote site-2\":{\"Spines\":\"?\",\"ctxDef/epgDef\":{\"encapOrPcTag\": ^
\"3080219/16387\",\"pcTagStatus\":\"OK\"},\"logical ctx/epg\": ^
{\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\":\"OK\"},\"msc\": ^
{\"encapOrPcTag\":\"3080219/16387\",\"pcTagStatus\":\"OK\"}}}],\"common\": ^
{\"Context\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"}, ^
\"dn\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ap-ANP-EPG-ONLY/ ^
epg-UntitledEPG1Consumer\",\"mscRef\":\"/schemas/5a78b5ef200000dd006f2f99/ ^
templates/Template1/anps/ANP-EPG-ONLY/epgs/UntitledEPG1Consumer\"}], ^
\"vrfs\":[{\"Overall Status\":\"NOT OK\",\"Sites Report\": ^
[{\"Local site-1\":{\"Spines\":\"?\",\"ctx Def\":{\"encapOrPcTag\": ^
\"2654245\",\"pcTagStatus\":\"OK\"},\"logical ctx\":{\"encapOrPcTag\": ^
\"2654245\",\"pcTagStatus\":\"OK\"},\"msc\":{\"encapOrPcTag\":\"2654245\", ^
\"pcTagStatus\":\"OK\"}},\"Remote site-2\":{\"Spines\":\"?\",\"ctx Def\": ^
{\"encapOrPcTag\":\"3080219\",\"pcTagStatus\":\"OK\"},\"logical ctx\": ^
{\"encapOrPcTag\":\"3080219\",\"pcTagStatus\":\"OK\"},\"msc\": ^
{\"encapOrPcTag\":\"3080219\",\"pcTagStatus\":\"OK\"}}}],\"common\": ^
{\"Context\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\"}, ^
\"dn\":\"uni/tn-CC-Test-Tenant-DoNotDelete/ctx-VRF-EPG-ONLY\",\"mscRef\": ^
\"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/vrfs/ ^
VRF-EPG-ONLY\"}]}}",
"finalResult": {
    "status": "PASSED"
},
"policies": {
    "bds": [
        {
            "mscRef": "/schemas/5a78b5ef200000dd006f2f99/templates/Template1/
             bds/BD-EPG-ONLY",
            "sites": [
                {
                    "apic": true,
                    "siteId": "5a6b8e7a5c0000a2010f1c18",
                    "siteName": "Test",
                    "isLocal": true,
                    "switch": true
                },
                {
                    "apic": true,
                    "siteId": "5a6b8d975c000045010f1c17",
                    "siteName": "Site2",
                    "isLocal": false,
                    "switch": true
                }
            ]
        }
    ],
    "epgs": [
```

```
                    {
                        "mscRef":
"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/anps/ANP-EPG-ONLY/epgs/EPG-ONLY",
                        "sites": [
                            {
                                "apic": true,
                                "siteId": "5a6b8e7a5c0000a2010f1c18",
                                "siteName": "Test",
                                "isLocal": true,
                                "switch": true
                            },
                            {
                                "apic": true,
                                "siteId": "5a6b8d975c000045010f1c17",
                                "siteName": "Site2",
                                "isLocal": false,
                                "switch": true
                            }
                        ]
                    },
                    {
                        "mscRef":
"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/anps/ANP-EPG-ONLY/epgs/UntitledEPG1Consumer",

                        "sites": [
                            {
                                "apic": true,
                                "siteId": "5a6b8e7a5c0000a2010f1c18",
                                "siteName": "Test",
                                "isLocal": true,
                                "switch": true
                            },
                            {
                                "apic": true,
                                "siteId": "5a6b8d975c000045010f1c17",
                                "siteName": "Site2",
                                "isLocal": false,
                                "switch": true
                            }
                        ]
                    }
                ],
                "vrfs": [
                    {
                        "mscRef":
"/schemas/5a78b5ef200000dd006f2f99/templates/Template1/vrfs/VRF-EPG-ONLY",
                        "sites": [
                            {
                                "apic": true,
                                "siteId": "5a6b8e7a5c0000a2010f1c18",
                                "siteName": "Test",
                                "isLocal": true,
                                "switch": true
                            },
                            {
                                "apic": true,
                                "siteId": "5a6b8d975c000045010f1c17",
                                "siteName": "Site2",
                                "isLocal": false,
                                "switch": true
                            }
                        ]
                    }
                ],
```

```
                "externalEpgs": []
            },
            "timestamp": "2018-02-07T10:24:11.29"
        }
```

Get the Heatmap Summary

The following is an example of how to fetch a consistency heatmap for all templates of the schema.

GET:

```
GET  /api/v1/consistency/summary
```

Request:

```
<empty>
```

Response:

```
{
   "schemas":[
      {
         "schemaId":"5a70b1001200005200f83a05",
         "displayName":"L3OutSchema",
         "templates":[
            {
               "templateName":"Template1",
               "templateDisplayName":"Template 1",
               "sites":[
                  {
                     "siteId":"590b49d4a86500a865bd176c",
                     "status":{
                        "status":"UNVERIFIED",
                        "result":"NO RESULT"
                     }
                  },
                  {
                     "siteId":"590b4a42a86500ba65bd176e",
                     "status":{
                        "status":"UNVERIFIED",
                        "result":"NO RESULT"
                     }
                  }
               ]
            }
         ]
      },
      {
         "schemaId":"5a66d54a0f00004c01e4fe60",
         "displayName":"Msite_PBR_Threshold",
         "templates":[
            {
               "templateName":"Schema_T1",
               "templateDisplayName":"Schema_T1",
               "sites":[
                  {
                     "siteId":"590b49d4a86500a865bd176c",
                     "status":{
                        "status":"UNVERIFIED",
                        "result":"NO RESULT"
```

```
                        }
                     },
                     {
                        "siteId":"590b4a42a86500ba65bd176e",
                        "status":{
                           "status":"UNVERIFIED",
                           "result":"NO RESULT"
                        }
                     }
                  ]
               }
            ]
         },
         {
            "schemaId":"5a612e4f1100003000f8bd2b",
            "displayName":"MSite_PBR_Schema",
            "templates":[
               {
                  "templateName":"Template1",
                  "templateDisplayName":"Template 1",
                  "sites":[
                     {
                        "siteId":"590b49d4a86500a865bd176c",
                        "status":{
                           "status":"IN PROGRESS",
                           "result":"NO RESULT"
                        }
                     },
                     {
                        "siteId":"590b4a42a86500ba65bd176e",
                        "status":{
                           "status":"IN PROGRESS",
                           "result":"NO RESULT"
                        }
                     }
                  ]
               }
            ]
         },
         {
            "schemaId":"5a25da1d0e000029003b9948",
            "displayName":"Schema_L4L7",
            "templates":[
               {
                  "templateName":"Template1",
                  "templateDisplayName":"Template 1",
                  "sites":[
                     {
                        "siteId":"590b49d4a86500a865bd176c",
                        "status":{
                           "status":"INIT",
                           "result":"NO RESULT",
                           "description":"Template Template1 should be deployed to more than
1 site"
                        }
                     }
                  ]
               }
            ]
         },
         {
            "schemaId":"5a6a6c001100008802f8bd2c",
            "displayName":"Usecase1",
            "templates":[
```

```
        {
            "templateName":"template0",
            "templateDisplayName":"template0",
            "sites":[
                {
                    "siteId":"590b49d4a86500a865bd176c",
                    "status":{
                        "status":"UNVERIFIED",
                        "result":"NO RESULT"
                    }
                },
                {
                    "siteId":"590b4a42a86500ba65bd176e",
                    "status":{
                        "status":"UNVERIFIED",
                        "result":"NO RESULT"
                    }
                }
            ]
        }
    ]
},
{
    "schemaId":"5a6789680e00000e0117391f",
    "displayName":"Scale Schema",
    "templates":[
        {
            "templateName":"Stretch_all_Sites",
            "templateDisplayName":"Stretch_all_Sites",
            "sites":[
                {
                    "siteId":"590b49d4a86500a865bd176c",
                    "status":{
                        "status":"UNVERIFIED",
                        "result":"NO RESULT"
                    }
                },
                {
                    "siteId":"590b4a42a86500ba65bd176e",
                    "status":{
                        "status":"UNVERIFIED",
                        "result":"NO RESULT"
                    }
                }
            ]
        },
        {
            "templateName":"ProvBDEPG_localSite2",
            "templateDisplayName":"ProvBDEPG_localSite2",
            "sites":[
                {
                    "siteId":"590b4a42a86500ba65bd176e",
                    "status":{
                        "status":"UNVERIFIED",
                        "result":"NO RESULT"
                    }
                }
            ]
        },
        {
            "templateName":"L3outTemplate",
            "templateDisplayName":"L3outTemplate",
            "sites":[
                {
```

```
                    "siteId":"590b49d4a86500a865bd176c",
                    "status":{
                       "status":"UNVERIFIED",
                       "result":"NO RESULT"
                    }
                 },
                 {
                    "siteId":"590b4a42a86500ba65bd176e",
                    "status":{
                       "status":"UNVERIFIED",
                       "result":"NO RESULT"
                    }
                 }
              ]
           },
           {
              "templateName":"ConBDEPG_localSite1",
              "templateDisplayName":"ConBDEPG_localSite1",
              "sites":[
                 {
                    "siteId":"590b49d4a86500a865bd176c",
                    "status":{
                       "status":"UNVERIFIED",
                       "result":"NO RESULT"
                    }
                 }
              ]
           },
           {
              "templateName":"VRF_ServiceBDStretch",
              "templateDisplayName":"VRF_ServiceBDStretch",
              "sites":[
                 {
                    "siteId":"590b49d4a86500a865bd176c",
                    "status":{
                       "status":"UNVERIFIED",
                       "result":"NO RESULT"
                    }
                 },
                 {
                    "siteId":"590b4a42a86500ba65bd176e",
                    "status":{
                       "status":"UNVERIFIED",
                       "result":"NO RESULT"
                    }
                 }
              ]
           }
        ]
     },
     {
        "schemaId":"5a7cb35e0e0000b204173922",
        "displayName":"test_schema",
        "templates":[
           {
              "templateName":"Template2",
              "templateDisplayName":"Template 2",
              "sites":[
                 {
                    "siteId":"590b49d4a86500a865bd176c",
                    "status":{
                       "status":"UNVERIFIED",
                       "result":"NO RESULT"
                    }
```

```
                    },
                    {
                       "siteId":"590b4a42a86500ba65bd176e",
                       "status":{
                          "status":"UNVERIFIED",
                          "result":"NO RESULT"
                       }
                    }
                 ]
              },
              {
                 "templateName":"Template1",
                 "templateDisplayName":"Template 1",
                 "sites":[
                    {
                       "siteId":"590b49d4a86500a865bd176c",
                       "status":{
                          "status":"NOT DEPLOYED",
                          "result":"NO RESULT"
                       }
                    },
                    {
                       "siteId":"590b4a42a86500ba65bd176e",
                       "status":{
                          "status":"NOT DEPLOYED",
                          "result":"NO RESULT"
                       }
                    }
                 ]
              }
           ]
        }
     ]
}
```


APIs for Site Local Properties

List Domains

The following is an example of how to list domains.

GET:

```
GET /api/v1/sites/<siteId>/aci/domains?type=<domain_type>
```

Where domain_type is one of the following:

- vmm
- physical
- l2Ext
- l3Ext
- fibreChannel

For example:

```
GET /api/v1/sites/58202f7066e6e10001c41236/aci/domains?type=vmm
```

Request:

```
<empty>
```

Response:

```
{
  "domains": [
    {
      "name": "cSphere",
      "dn": "uni/vmmp-VMware/dom-cSphere"
    },
    {
      "name": "Fab1-DVS1",
      "dn": "uni/vmmp-VMware/dom-Fab1-DVS1"
    },
    {
      "name": "TestDomain-DoNotDelete",
      "dn": "uni/vmmp-VMware/dom-TestDomain-DoNotDelete"
    }
  ]
}
```

List L3Outs

The following is an example of how to list L3Outs.

GET:

```
GET /api/v1/sites/<id>/aci/tenants/<idOrName>/l3-outs?includeCommon=<true|false>
```

For example:

```
GET
/api/v1/sites/58202f7066e6e10001c41236/aci/tenants/597995d13300006700fdf7b8/l3-outs?includeCommon=false
```

Request:

```
<empty>
```

Response:

```
{
  "l3Outs": [
    {
      "name": "TestL3Out-DoNotDelete",
      "dn": "uni/tn-TestTenant-DoNotDelete/out-TestL3Out-DoNotDelete",
      "networkInstanceProfiles": [
        {
          "name": "TestNIP-1-DoNotDelete",
          "dn":
"uni/tn-TestTenant-DoNotDelete/out-TestL3Out-DoNotDelete/instP-TestNIP-1-DoNotDelete"
        },
        {
          "name": "TestNIP-2-DoNotDelete",
          "dn":
"uni/tn-TestTenant-DoNotDelete/out-TestL3Out-DoNotDelete/instP-TestNIP-2-DoNotDelete"
```

```
        }
      ]
    },
    {
      "name": "TestL3Out2",
      "dn": "uni/tn-TestTenant-DoNotDelete/out-TestL3Out2",
      "networkInstanceProfiles": [
        {
          "name": "N1",
          "dn": "uni/tn-TestTenant-DoNotDelete/out-TestL3Out2/Instp-N1"
        }
      ]
    }
  ]
}
```

List L3Outs Route Control Panels

The following is an example of how to list L3Outs route control panels.

GET:

```
GET
/api/v1/sites/<id>/aci/tenants/<name>/l3-out/<l3OutName>/route-ctrl-profiles?includeCommon=<true|false>
```

For example:

```
GET
/api/v1/sites/5aa85b065c000067008cce69/aci/tenants/Suresh_Tenant2/l3-out/L3Out2/route-ctrl-profiles?includeCommon=false
```

Request:

```
<empty>
```

Response:

```
{
    "listL3OutRouteCtrlProfileDns": [
        "uni/tn-Suresh_Tenant2/out-L3Out2/prof-RMP1",
        "uni/tn-Suresh_Tenant2/out-L3Out2/prof-default-import"
    ]
}
```

List Security Domains

The following is an example of how to list security domains.

GET:

```
GET /api/v1/sites/<id>/aci/security-domains
```

For example:

```
GET /api/v1/sites/58dfeae2fd0000fd00cee617/aci/security-domains
```

Request:

```
<empty>
```

Response:

```
{
  "securityDomains": [
    {
      "name": "TestSecurityDomain",
      "dn": "uni/userext/domain-TestSecurityDomain"
    },
    {
      "name": "common",
      "dn": "uni/userext/domain-common"
    },
    {
      "name": "all",
      "dn": "uni/userext/domain-all"
    },
    {
      "name": "mgmt",
      "dn": "uni/userext/domain-mgmt"
    },
    {
      "name": "Foo",
      "dn": "uni/userext/domain-Foo"
    },
    {
      "name": "fgdfg",
      "dn": "uni/userext/domain-fgdfg"
    }
  ]
}
```

List L4-L7 Devices

The following is an example of how to list L4-L7 devices.

GET:

```
GET /api/v1/sites/<id>/aci/tenants/<name>/devices?deviceType=<deviceType>
```

Where **deviceType** values are one of the following:

- **FW** for Firewall
- **ADC** for Load Balancer
- **OTHERS** for IPS/IDS

For example:

```
GET /api/v1/sites/5a00a8656f00006f0044f160/aci/tenants/BigC/devices?deviceType=ADC
```

Request:

```
<empty>
```

Response:

```
{
    "devices": [
        {
            "name": "Load_Balancer_A10_Networks",
            "dn": "uni/tn-BigC/lDevVip-Load_Balancer_A10_Networks",
            "deviceType": "ADC"
        },
        {
            "name": "ADC2",
            "dn": "uni/tn-BigC/lDevVip-ADC2",
            "deviceType": "ADC"
        }
    ]
}
```

List L4-L7 Device Interfaces

The following is an example of how to list L4-L7 device interfaces.

GET:

```
GET /api/v1/sites/<id>/aci/tenants/<name>/devices/<deviceName>/interfaces
```

For example:

```
GET /api/v1/sites/5a00a8656f00006f0044f160/aci/tenants/BigC/devices/A101/interfaces
```

Request:

```
<empty>
```

Response:

```
{
    "interfaces": [
        {
            "name": "L3",
            "dn": "uni/tn-BigC/lDevVip-ADC2/lIf-L3"
        },
        {
            "name": "L4",
            "dn": "uni/tn-BigC/lDevVip-ADC2/lIf-L4"
        },
        {
            "name": "L2",
            "dn": "uni/tn-BigC/lDevVip-ADC2/lIf-L2"
        },
        {
            "name": "L1",
            "dn": "uni/tn-BigC/lDevVip-ADC2/lIf-L1"
        }
    ]
}
```

List L4-L7 Redirect Policies

The following is an example of how to list L4-L7 redirect policies.

GET:

```
GET /api/v1/sites/<id>/aci/tenants/<name>/redirect-policies
```

For example:

```
GET /api/v1/sites/5a131dc44300004300627e41/aci/tenants/BigC/redirect-policies
```

Request:

```
<empty>
```

Response:

```
{
    "redirectPolicies": [
        {
            "name": "pbr1",
            "dn": "uni/tn-BigC/svcCont/svcRedirectPol-pbr1"
        },
        {
            "name": "pbr2",
            "dn": "uni/tn-BigC/svcCont/svcRedirectPol-pbr2"
        }
    ]
}
```

APIs For Local Authentication

User Login

The following is an example of how a user can login.

POST:

```
POST https://<host>:<port>/api/v1/auth/login
```

Request:

```
{
 "username": "bob",
 "password": "welcome"
}
```

Response:

```
HTTP/1.1 201 (Created)      -> Valid user credentials
{
  "userId": "592717d22000008b01db2749",
  "token":
"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.ey...fQ.QpfcqZdo-g-O0RGUmMZCVspIL7rYHjKJH0srX7Tl3wA",

  "permissions": [
    "view-sites",
    "manage-sites",
    "view-tenants",
```

```
    "manage-tenants"
  ]
}

HTTP/1.1 401(Unauthorized)   -> Invalid user credentials
{
  "code": 401,
  "message": "Invalid user credentials"
}
```

Note

- This is the only API in Cisco ACI Multi-Site that does not require the Authorization header to be set.
- The token in the above example has been trimmed for readability.

User Logout

The following is an example of how a user can logout.

DELETE:

```
DELETE https://<host>:<port>/api/v1/auth/logout
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 204
```

Refresh Token

The following is an example of how to refresh a token.

GET:

```
GET https://<host>:<port>/api/v1/auth/refresh-token
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "userId": "592717d22000008b01db2749",
  "token":
"eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.ey...fQ.YVGLUtVZYfrMa-Bpf7gk281dZ4uPg20pk1QF6ffH63w",

  "permissions": [
    "view-sites",
    "manage-sites",
    "view-tenants",
```

```
    "manage-tenants"
  ]
}
```

Note Tokens in Cisco ACI Multi-Site expire every 20 minutes. For longer interactions, this API provides an ability to obtain new tokens for subsequent interactions.

Using Tokens on Authentication on APIs

Once the user token has been returned from the initial login API POST, this token must be inserted into a Header key called “Authorization” with the value of “Bearer [Token_ID]” for all subsequent API calls.

The following is an example of how to use tokens on authentication on APIs. In this example, it is to get a list of tenants.

GET:

```
GET https://<host>:<port>/api/v1/tenants -H 'Authorization': 'Bearer {token}'
```

Request:

```
curl http://localhost/api/v1/tenants -H 'Authorization': 'Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.ey...fQ.QpfcqZdo-g-O0RGUmMZCVspIL7rYHjKJH0srX7Tl3wA'
```

Authenticating and Maintaining an API Session

The following is an example of how to authenticate and maintain an API session.

- Before you access the API, you must first log in with the username and password of a configured user.
- When a login message is accepted, the API returns a data structure that includes a token that represents the session, and it also returns the permissions of the user.
- The default timeout period is 20 minutes.

Login: Sent as a post message, this method logs in the user, and opens the session. A user object, with the name and password attributes and the response, contains a session token. A set of permissions given to this user are also returned as a part of the response body.

Refresh: Tokens in Multi-Site expire every 20 minutes. For longer interactions, this API provides an ability to obtain new tokens for subsequent interactions.

Login to Authenticate and Maintain an API Session

The following example shows how to login to authenticate and maintain an API session.

POST:

```
POST https://<host>:<port>/api/v1/auth/login
```

Request:

```
{
  "username": "georgewa",
```

```
  "password": "paSSword1”
}
```

Response:

Note This is part of the response after a successful login.

```
{
  "userId": "0000ffff0000000000000020",
  "needsPasswordUpdate": false,
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJleHAiOjE1MDA2NTgyODMsImlhd
CI6MTUwMDY1NzA4MywiaXNzIjoiZTc3Zjc5OTBlYjA4NDhkZWEwZDk0YWY0NmMzMWZmNDYiLCJ1c2
VySWQiOiIwMDAwZmZmZjAwMDAwMDAwMDAwMDAwMjAiLCJ1c2VybmFtZSI6ImFkbWluIiwic
GVybWlzc2lvbnMiOlsidmlldy1zaXRlcyIsIm1hbmFnZS1zaXRlcyIsInZpZXctdGVuYW50
cyIsIm1hbmFnZS10ZW5hbnRzIiwidmlldy10ZW5hbnQtc2NoZW1hcyIsIm1hbmFnZS10ZW5
hbnQtc2NoZW1hcyIsInZpZXctc2NoZW1hcyIsIm1hbmFnZS1zY2hlbWFzIiwidmlldy11c2
VycyIsIm1hbmFnZS11c2VycyIsInZpZXctcm9sZXMiLCJtYW5hZ2Utcm9sZXMiLCJ2aWV3
LWxhYmVscyIsIm1hbmFnZS1sYWJlbHMiLCJwbGF0Zm9ybS1sb2dzIl19.KU5I4qynMKvj
2ye0f3yzEKdABkE6uQNjGXj6Z3bbbb",
  "permissions": [
    "view-sites",
[TRUNCATED] ...
  ]
}
Note: User has to copy above token, in the header of all the requests user need to access
from now onward.
Where the header key is ‘Authorization’
And value is ‘Bearer {token}’
============================End=============================
```

Refresh to Authenticate and Maintain an API Session

The following example shows how to refresh to authenticate and maintain an API session.

Tokens in Multi-Site expire every 20 minutes. For longer interactions, this API provides an ability to obtain new tokens for subsequent interactions.

GET:

```
GET https://<host>:<port>/api/v1/auth/refresh-token
-H 'Authorization': 'Bearer {token}'
```

Note Replace {token} with your current token.

Response:

```
{
  "userId": "0000ffff0000000000000020",
  "needsPasswordUpdate": false,
  "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJleHAiOjE1MDA2NTgyODMsImlhd
CI6MTUwMDY1NzA4MywiaXNzIjoiZTc3Zjc5OTBlYjA4NDhkZWEwZDk0YWY0NmMzMWZmNDYiLCJ1c2
VySWQiOiIwMDAwZmZmZjAwMDAwMDAwMDAwMDAwMjAiLCJ1c2VybmFtZSI6ImFkbWluIiwicGVybWlzc
2lvbnMiOlsidmlldy1zaXRlcyIsIm1hbmFnZS1zaXRlcyIsInZpZXctdGVuYW50cyIsIm1hbmFnZS10Z
W5hbnRzIiwidmlldy10ZW5hbnQtc2NoZW1hcyIsIm1hbmFnZS10ZW5hbnQtc2NoZW1hcyIsInZpZXctc2
NoZW1hcyIsIm1hbmFnZS1zY2hlbWFzIiwidmlldy11c2VycyIsIm1hbmFnZS11c2VycyIsInZpZXctcm9s
ZXMiLCJtYW5hZ2Utcm9sZXMiLCJ2aWV3LWxhYmVscyIsIm1hbmFnZS1sYWJlbHMiLCJwbGF0Zm9ybS1sb
```

```
2dzIl19.KU5I4qynMKvj2ye0f3yzEKdABkE6uQNjGXj6Z3bbbb",
  "permissions": [
    "view-sites",
[TRUNCATED] ...
  ]
}
```

The following is an example to get a list of tenants.

```
GET https://<host>:<port>/api/v1/tenants
-H 'Authorization': 'Bearer {token}'

{
  "tenants": [
    {
      "id": "0000ffff0000000000000010",
      "name": "common",
      "displayName": "common",
      "siteAssociations": [
        {
          "siteId": "596e52244800007400840196",
          "securityDomains": []
        },
        {
          "siteId": "596e522a4800007700840198",
          "securityDomains": []
        }
      ],
      "userAssociations": [
        {
          "userId": "0000ffff0000000000000020"
        }
      ],
      "description": "Common tenant for use with all other tenants"
    }
  ]
}
```

APIs For External Authentication

About External Authentication

Starting with Cisco ACI Multi-Site, release 1.1(1), the APIs for external authentication are supported. Therefore, the external authentication examples are applicable starting with release 1.1(1) only.

APIs For RADIUS Configurations

Create a New RADIUS Provider

The following is an example of how to create a new RADIUS provider.

POST:

```
POST https://<host>:<port>/api/v1/auth/providers/radius
```

Request:

```
{
    "host": "10.23.237.168",
    "description": "ACS RADIUS Provider",
    "sharedSecret": "ins3965!",
    "port": 1812,
    "protocol": "PAP",
    "timeoutInSeconds": 10,
    "retries":5
}
```

Response:

```
HTTP/1.1 201
Location: /api/v1/auth/providers/radius/59cc163c1f00005400ab4ffd
{
    "id": "59cc163c1f00005400ab4ffd",
    "host": "10.23.237.168",
    "description": "Tacacs Provider - 1",
    "sharedSecret": "******",
    "port": 1812,
    "protocol": "PAP",
    "timeoutInSeconds": 10,
    "retries": 5
}
```

Read RADIUS Provider Information

The following is an example of how to read RADIUS provider information.

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/radius/59cc163c1f00005400ab4ffd
```

Request:

```
<empty >
```

Response:

```
HTTP/1.1 200
{
    "id": "59cc163c1f00005400ab4ffd",
    "host": "10.23.237.168",
    "description": "ACS RADIUS Provider",
    "sharedSecret": "******",
    "port": 1812,
    "protocol": "PAP",
    "timeoutInSeconds": 10,
    "retries": 5
}
```

List Providers

The following is an example of how to list RADIUS providers.

GET:

```
GET https://<host>:<port>/api/v1/providers/radius
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
    "radiusProviders": [
        {
            "id": "59cc163c1f00005400ab4ffd",
            "host": "10.23.237.168",
            "description": "RADIUS",
            "sharedSecret": "******",
            "port": 1812,
            "protocol": "PAP",
            "timeoutInSeconds": 10,
            "retries": 5
        },
        {
            "id": "59cc19942100004a0077f85e",
            "host": "10.23.237.169",
            "description": "ACS RADIUS Provider",
            "sharedSecret": "******",
            "port": 1812,
            "protocol": "PAP",
            "timeoutInSeconds": 10,
            "retries": 5
        }
    ]
}
```

The following is an example of how to list all providers.

GET:

```
GET https://<host>:<port>/api/v1/providers
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
    "providers": [
        {
            "id": "59cc30942200005e0076c633",
            "host": "10.23.237.101",
            "providerType": "radius",
            "description": "RADIUS",
            "port": 1812,
            "protocol": "PAP",
            "timeoutInSeconds": 10,
            "retries": 5
```

```
        },
        {
            "id": "59cc3157220000780076c634",
            "host": "10.23.237.188",
            "providerType": "radius",
            "description": "RADIUS",
            "port": 1812,
            "protocol": "PAP",
            "timeoutInSeconds": 10,
            "retries": 5
        }
    ]
}
```

Update an Existing RADIUS Provider

The following is an example of how to update an existing RADIUS provider.

PUT:

```
PUT https://<host>:<port>/api/v1/auth/providers/radius/59cc163c1f00005400ab4ffd
```

Request:

```
{
    "id": "59cc163c1f00005400ab4ffd",
    "host": "10.23.237.168",
    "description": "ACS RADIUS Provider",
    "sharedSecret": "ins3965!",
    "port": 1812,
    "protocol": "PAP",
    "timeoutInSeconds": 10,
    "retries": 3
}
```

Response:

```
HTTP/1.1 200
{
    "id": "59cc163c1f00005400ab4ffd",
    "host": "10.23.237.168",
    "description": "ACS RADIUS Provider",
    "sharedSecret": "******",
    "port": 1812,
    "protocol": "PAP",
    "timeoutInSeconds": 10,
    "retries": 3
}
```

Delete a RADIUS Provider

The following is an example of how to delete a RADIUS provider.

DELETE:

```
DELETE https://<host>:<port>/api/v1/auth/providers/radius/59cc163c1f00005400ab4ffd
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 204
```

APIs for TACACS + Configurations

Create a New TACACS Provider

The following is an example of how to create a new TACACS provider.

POST:

```
POST https://<host>:<port>/api/v1/auth/providers/tacacs
```

Request:

```
{
    "host": "10.23.237.168",
    "description": "Tacacs Provider - 1",
    "sharedSecret": "ins3965!",
    "port": 49,
    "protocol": "PAP",
    "timeoutInSeconds": 10,
    "retries":5
}
```

Response:

```
HTTP/1.1 201
Location: /api/v1/auth/providers/tacacs/59cc163c1f00005400ab4ffd
{
    "id": "59cc163c1f00005400ab4ffd",
    "host": "10.23.237.168",
    "description": "Tacacs Provider - 1",
    "sharedSecret": "******",
    "port": 49,
    "protocol": "PAP",
    "timeoutInSeconds": 10,
    "retries": 5
}
```

Read TACACS Provider Information

The following is an example of how to read TACACS provider information.

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/tacacs/59cc163c1f00005400ab4ffd
```

Request:

```
<empty >
```

Response:

```
HTTP/1.1 200
{
    "id": "59cc163c1f00005400ab4ffd",
    "host": "10.23.237.168",
    "description": "Tacacs Provider - 1",
    "sharedSecret": "******",
    "port": 49,
    "protocol": "PAP",
    "timeoutInSeconds": 10,
    "retries": 5
}
```

List TACACS Providers

The following is an example of how to list TACACS providers.

GET:

```
GET https://<host>:<port>/api/v1/providers/tacacs
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
    "tacacsProviders": [
        {
            "id": "59cc163c1f00005400ab4ffd",
            "host": "10.23.237.168",
            "description": "Tacacs Provider - 1",
            "sharedSecret": "******",
            "port": 49,
            "protocol": "PAP",
            "timeoutInSeconds": 10,
            "retries": 5
        },
        {
            "id": "59cc19942100004a0077f85e",
            "host": "10.23.237.169",
            "description": "Tacacs Provider - 1",
            "sharedSecret": "******",
            "port": 49,
            "protocol": "PAP",
            "timeoutInSeconds": 10,
            "retries": 5
        }
    ]
}
```

Update an Existing TACAS Provider

The following is an example of how to update an existing TACACS provider.

PUT:

```
PUT https://<host>:<port>/api/v1/auth/providers/tacacs/59cc163c1f00005400ab4ffd
```

Request:

```
{
    "id": "59cc163c1f00005400ab4ffd",
    "host": "10.23.237.168",
    "description": "Tacacs Provider - 1",
    "sharedSecret": "ins3965!",
    "port": 49,
    "protocol": "PAP",
    "timeoutInSeconds": 10,
    "retries": 3
}
```

Response:

```
HTTP/1.1 200
{
    "id": "59cc163c1f00005400ab4ffd",
    "host": "10.23.237.168",
    "description": "Tacacs Provider - 1",
    "sharedSecret": "******",
    "port": 49,
    "protocol": "PAP",
    "timeoutInSeconds": 10,
    "retries": 3
}
```

Delete a TACACS Provider

The following is an example of how to delete a TACACS provider.

DELETE:

```
DELETE https://<host>:<port>/api/v1/auth/providers/tacacs/59cc163c1f00005400ab4ffd
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 204
```

APIs For Login Domain

Create a New Login Domain

The following is an example of how to create a new login domain.

POST:

```
POST https://<host>:<port>/api/v1/auth/domains
```

Request:

```
{
    "name" : "EngineeringGroup",
    "description" : "Used for RADIUS Auth Login",
    "realm" : "radius",
    "status": "active",
    "providerAssociations": [
    {
      "providerId": "59d5b4de8d0000ba00909f62",
      "priority" : 1
    }
  ]
}
```

Response:

```
HTTP/1.1 201
{
    "id": "59d6d0398d0000d2009eb2c4",
    "name": "EngineeringGroup",
    "description": "Used for RADIUS Auth Login",
    "realm": "radius",
    "status": "active",
    "providerAssociations": [
        {
            "providerId": "59d5b4de8d0000ba00909f62",
            "priority": 1
        }
    ]
}
```

List all Active Login Domains

The following is an example of how to list all active login domains.

Note

- This is an unauthorized APIC for initial login.

GET:

```
GET https://<host>:<port>/api/v1/auth/login-domains
```

Request:

```
{
 {
    "domains": [
        {
            "id": "59d6d0398d0000d2009eb2c4",
            "name": "EngineeringGroup",
            "description": "Used for RADIUS Auth Login",
            "realm": "radius",
            "status": "active",
            "providerAssociations": []
        },
        {
            "id": "59d6cfb58d0000bd009eb2c3",
            "name": "VM-TACACS",
            "description": "Used for Tacacs Auth Login",
```

```
            "realm": "tacacs",
            "status": "active",
            "providerAssociations": []
        }
    ]
}
}
```

Note

- The token in the above example has been truncated for readability.

List Login Domains

The following is an example of how to list all login domains.

GET:

```
GET https://<host>:<port>/api/v1/auth/domains
```

Request:

```
{
 {
    "domains": [
        {
            "id": "59d6d0398d0000d2009eb2c4",
            "name": "EngineeringGroup",
            "description": "Used for RADIUS Auth Login",
            "realm": "radius",
            "status": "active",
            "providerAssociations": [
                {
                    "providerId": "59d5b4de8d0000ba00909f62",
                    "priority": 1
                },
                {
                    "providerId": "59d6d0988d0000cb009eb2c5",
                    "priority": 2
                }
            ]
        },
        {
            "id": "59d6cfb58d0000bd009eb2c3",
            "name": "VM-TACACS",
            "description": "Used for Tacacs Auth Login",
            "realm": "tacacs",
            "status": "active",
            "providerAssociations": [
                {
                    "providerId": "59d5b4de8d0000ba00909f62",
                    "priority": 1
                }
            ]
        }
    ]
}
```

The following is an example of how to list a login domain detail.

POST:

```
POST https://<host>:<port>/api/v1/auth/domains/59d2c88f9100000105b849e0
```

Request:

```
{
    "id": "59d6d0398d0000d2009eb2c4",
    "name": "EngineeringGroup",
    "description": "Used for RADIUS Auth Login",
    "realm": "radius",
    "status": "active",
    "providerAssociations": [
        {
            "providerId": "59d5b4de8d0000ba00909f62",
            "priority": 1
        },
        {
            "providerId": "59d6d0988d0000cb009eb2c5",
            "priority": 2
        }
    ]
}
```

Modify an Existing Login Domain

The following is an example of how to modify an existing login domain.

PUT:

```
PUT https://<host>:<port>/api/v1/auth/domains/59d6d0398d0000d2009eb2c4
```

Request:

```
{
    "id": "59d6d0398d0000d2009eb2c4",
    "name": "EngineeringGroup",
    "description": "Used for RADIUS Auth Login",
    "realm": "radius",
    "status": "active",
    "providerAssociations": [
        {
            "providerId": "59d5b4de8d0000ba00909f62",
            "priority": 1
        },
        {
            "providerId": "59d6d0988d0000cb009eb2c5",
            "priority": 2
        }
    ]
}
```

Delete Login Domain

The following is an example of how to delete a login domain.

DELETE:

```
DELETE https://<host>:<port>/api/v1/domains/59d6d0398d0000d2009eb2c4
```

Request:

```
<empty>
```

APIs For Remote User Login

Remote User Login

The following is an example of how to read and create remote user login domains.

POST:

```
POST https://<host>:<port>/api/v1/auth/login
```

Request:

```
{
 "username": "bob",
 "password": "we1come!",
 "domainId":"59d5b5978d0000d000909f65",
}
```

Response:

```
HTTP/1.1 201 (Created)      -> Valid user credentials
{
    "userId": "0000ffff0000000000000020",
    "needsPasswordUpdate": false,
    "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9....e-g308aqI8",
    "permissions": [
        "view-sites",
        "manage-sites",
        "view-tenants",
        "manage-tenants",
        "view-tenant-schemas",
        "manage-tenant-schemas",
        "view-schemas",
        "manage-schemas",
        "view-users",
        "manage-users",
        "view-roles",
        "manage-roles"
    ]
}

HTTP/1.1 401(Unauthorized)   -> Invalid user credentials
{
  "code": 401,
  "message": "Invalid user credentials"
}

HTTP/1.1 500(UnsupportedAuthType)   -> Authentication domain not supported
{
  "code": 500,
  "message": "Authentication domain not supported"
}
```

Note

- This API does not require the Authorization header to be set.
- The token in the above example has been truncated for readability.

APIs For Labels

Create a New Label

The following is an example of how to create a new label.

POST:

```
POST https://<host>:<port>/api/v1/labels
```

Request:

```
{
  "name" : "devops",
  "type" : "site"
}
```

Response:

```
HTTP/1.1 201
Location: /api/v1/labels/58202f7066e6e10001c41236
{
  "id": "58202f7066e6e10001c41236",
  "name" : "devops",
  "type" : "site"
}
```

Read Label Information

The following is an example of how to read label information.

GET:

```
GET https://<host>:<port>/api/v1/labels/58202f7066e6e10001c41236
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "id": "58202f7066e6e10001c41236",
  "name" : "devops",
```

```
  "type" : "site"
}
```

Update a Label

The following is an example of how to update a label.

PUT:

```
PUT https://<host>:<port>/api/v1/labels/58202f7066e6e10001c41236
```

Request:

```
{
    "id": "58202f7066e6e10001c41236",
    "name" : "test",
    "type" : "site"
}
```

Response:

```
HTTP/1.1 200
{
  "id": "58202f7066e6e10001c41236",
  "name" : "test",
  "type" : "site"
}
```

Delete Labels

The following is an example of how to delete a label.

DELETE:

```
DELETE https://<host>:<port>/api/v1/labels/58202f7066e6e10001c41236
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 204
```

List Labels

The following is an example of how to list labels.

GET:

```
GET https://<host>:<port>/api/v1/labels
```

Note Filtering, pagination, and sorting is currently not supported.

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "labels": [
    {
      "id": "58202f7066e6e10001c41236",
      "name" : "devops",
      "type" : "site"
    },
    {
      "id": "58202fc466e6e10001c41237",
      "name" : "prod",
      "type" : "site"
    }
  ]
}
```

APIs For Roles

Read Role Information

The following is an example of how to read role information.

GET:

```
GET https://<host>:<port>/api/v1/role/591f58c1220000e406fa80b5
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "id": "591f58c1220000e406fa80b5",
  "name": "role1",
  "displayName": "Role 1",
  "description": "Description for Role 1",
  "permissions": [
    "view-sites",
    "manage-sites",
    "view-tenants",
```

```
    "manage-tenants",
    "view-tenant-schemas",
    "view-schemas",
    "manage-schemas",
    "view-users",
    "manage-users",
    "view-roles",
    "manage-roles"
  ]
}
```

List Roles

The following is an example of how to list roles.

GET:

```
GET https://<host>:<port>/api/v1/roles
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "roles": [
    {
      "id": "591f58c1220000e406fa80b5",
      "name": "role1",
      "displayName": "Role 1",
      "description": "Description for Role 1",
      "permissions": [
        "view-sites",
        "manage-sites",
        "view-tenants",
        "manage-tenants",
        "view-tenant-schemas",
        "view-schemas",
        "manage-schemas",
        "view-users",
        "manage-users",
        "view-roles",
        "manage-roles"
      ]
    },
    {
      "id": "591f5903220000f106fa80b6",
      "name": "role2",
      "displayName": "Role 2",
      "description": "Description for Role 2",
      "permissions": [
        "view-sites",
        "manage-sites",
        "view-tenants",
        "manage-tenants"
      ]
    }
  ]
```

```
}
```

APIs For Users

Create a New User

The following is an example of how to create a new user.

POST:

```
POST https://<host>:<port>/api/v1/users
```

Request:

```
{
 "username": "user1",
 "password": "Cisco@123!",
 "firstName": "Foo",
 "lastName": "Bar",
 "emailAddress": "noreply@cisco.com",
 "phoneNumber": "408-527-0000",
 "accountStatus": "active",
  "roles": [
    {
      "roleId": "591f5903220000f106fa80b6"
    }
  ]
```

Response:

```
HTTP/1.1 201
Location: /api/v1/users/58202f7066e6e10001c41236
{
  "id": "58202f7066e6e10001c41236",
  "username": "user1",
  "password": "******",
  "firstName": "Foo",
  "lastName": "Bar",
  "emailAddress": "noreply@cisco.com",
  "phoneNumber": "408-527-0000",
  "accountStatus": "active",
  "roles": [
    {
      "roleId": "591f5903220000f106fa80b6"
    }
  ]
}
```

Reset Password for an Existing User

The following is an example of how to reset the password for an existing user.

PUT:

```
PUT https://<host>:<port>/api/v1/users/58202f7066e6e10001c41236/reset-password
```

Request:

```
{
  "currentPassword": "Cisco@123!",
   "newPassword":    "Cisco@456!"
}
```

Response:

```
HTTP/1.1 200 OK
```

Note If a user password is system generated or generated by a user-manager in Multi-Site, then upon login, that user (including the admin) will be required to reset their own password.

The minimum requirements for the password field are as follows:

- Minimum 6 characters must be used
- At least one letter must be included
- At least one number must be included
- At least 1 special character must be used. But space and * are not supported.

Read User Information

The following is an example of how to read user information.

GET:

```
GET https://<host>:<port>/api/v1/users/58202f7066e6e10001c41236
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  id": "58202f7066e6e10001c41236",
  "username": "user1",
  "password": "******",
  "firstName": "Foo",
  "lastName": "Bar",
  "emailAddress": "noreply@cisco.com",
  "phoneNumber": "408-527-0000",
```

```
  "accountStatus": "active",
  "roles": [
    {
      "roleId": "591f5903220000f106fa80b6"
    }
  ]
}
```

Update an Existing User

The following is an example of how to update an existing user.

PUT:

```
PUT https://<host>:<port>/api/v1/users/58202f7066e6e10001c41236
```

Request:

```
{
  id": "58202f7066e6e10001c41236",
  "username": "user1",
  "password": "******",
  "firstName": "Foo",
  "lastName": "Bar",
  "emailAddress": "noreply@cisco.com",
  "phoneNumber": "408-527-0000",
  "accountStatus": "active",
  "roles": [
    {
      "roleId": "591f58c1220000e406fa80b5"
    },
    {
      "roleId": "591f5903220000f106fa80b6"
    }
  ]
}
```

Response:

```
HTTP/1.1 200
{
  id": "58202f7066e6e10001c41236",
  "username": "user1",
  "password": "******",
  "firstName": "Foo",
  "lastName": "Bar",
  "emailAddress": "noreply@cisco.com",
  "phoneNumber": "408-527-0000",
  "accountStatus": "active",
  "roles": [
    {
      "roleId": "591f58c1220000e406fa80b5"
    },
    {
      "roleId": "591f5903220000f106fa80b6"
    }
  ]
```

```
}
```

Delete a User

The following is an example of how to delete a user.

DELETE:

```
DELETE https://<host>:<port>/api/v1/users/58202f7066e6e10001c41236
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 204
```

List Users

The following is an example of how to list users.

GET:

```
GET https://<host>:<port>/api/v1/users
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "users": [
    {
      "id": "58202f7066e6e10001c41236",
      "username": "user1",
      "password": "",
      "firstName": "Foo",
      "lastName": "Bar",
      "emailAddress": "noreply@cisco.com",
      "phoneNumber": "408-527-000",
      "accountStatus": "active",
      "roles": [
        {
          "roleId": "591f5903220000f106fa80b6"
        }
      ]
    },
    {
      "id": "58202fc466e6e10001c41237",
```

```
      "username": "user2",
      "password": "",
      "firstName": "Foo2",
      "lastName": "Bar2",
      "emailAddress": "noreply@cisco.com",
      "phoneNumber": "408-527-0000",
      "accountStatus": "Inactive",
      "roles": []
    }
  ]
}
```

List Roles That May Be Associated With Users

The following is an example of how to list roles that may be associated with users.

GET:

```
GET https://<host>:<port>/api/v1/users/allowed-roles
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "roles": [
    {
      "id": "591f58c1220000e406fa80b5",
      "displayName": "Role 1",
      "permissions": [
        "view-sites",
        "manage-sites",
        "view-tenants",
        "manage-tenants",
        "view-tenant-schemas",
        "view-schemas",
        "manage-schemas",
        "view-users",
        "manage-users",
        "view-roles",
        "manage-roles"
      ]
    },
    {
      "id": "591f5903220000f106fa80b6",
      "displayName": "Role 2",
      "permissions": [
        "view-sites",
        "manage-sites",
        "view-tenants",
        "manage-tenants"
      ]
    }
  ]
```

```
}
```

Retrieve Roles Associated With a User

The following is an example of how to retrieve roles associated with a user.

GET:

```
GET https://<host>:<port>/api/v1/users/58202f7066e6e10001c41236/roles
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "roles": [
    {
      "id": "591f58c1220000e406fa80b5",
      "displayName": "Role 1",
      "permissions": [
        "view-sites",
        "manage-sites",
        "view-tenants",
        "manage-tenants",
        "view-tenant-schemas",
        "view-schemas",
        "manage-schemas",
        "view-users",
        "manage-users",
        "view-roles",
        "manage-roles"
      ]
    }
  ]
}
```

Retrieve Permissions Associated With a User

The following is an example of how to retrieve permissions associated with a user.

GET:

```
GET https://<host>:<port>/api/v1/users/58202f7066e6e10001c41236/permissions
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "permissions": [
    "view-sites",
    "manage-sites",
    "view-tenants",
    "manage-tenants",
    "view-tenant-schemas",
    "view-schemas",
    "manage-schemas",
    "view-users",
    "manage-users",
    "view-roles",
    "manage-roles"
  ]
}
```

APIs For Fabric Connectivity

Get Fabric Connectivity for a Site

Example for ACI Multi-Site Release 1.2(1)

The following is an example of how to get fabric connectivity for a site.

GET:

```
GET /api/v1/sites/id/fabric-connectivity
```

For example:

```
GET /api/v1/sites/5aeccccb35000050001977bc/fabric-connectivity
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200

{
  "id" : "5aeccccb35000050001977bc",
  "apicSiteId" : 2,
  "fabricId" : 2,
  "msiteEnabled" : true,
  "msiteDataPlaneMulticastTep" : "12.12.12.1",
  "bgpAsn" : 1234,
  "bgpPassword" : "",
  "ospfAreaId" : "0.0.0.8",
  "ospfAreaType" : "nssa",
  "externalRoutedDomain" : "uni/l3dom-MultisiteL3Domain1-DoNotDelete",
  "ospfPolicies" : [ {
    "name" : "common/default",
    "networkType" : "unspecified",
    "priority" : 1,
```

```
    "interfaceCost" : 0,
    "interfaceControls" : [ ],
    "helloInterval" : 10,
    "deadInterval" : 40,
    "retransmitInterval" : 5,
    "transmitDelay" : 1
  }, {
    "name" : "msc-ospf-policy-default",
    "networkType" : "point-to-point",
    "priority" : 1,
    "interfaceCost" : 0,
    "interfaceControls" : [ ],
    "helloInterval" : 10,
    "deadInterval" : 40,
    "retransmitInterval" : 5,
    "transmitDelay" : 1
  } ],
  "pods" : [ {
    "podId" : 1,
    "name" : "pod-1",
    "msiteDataPlaneUnicastTep" : "32.32.32.1",
    "spines" : [ {
      "nodeId" : 211,
      "name" : "spine11",
      "ports" : [ {
        "portId" : "1/29",
        "ipAddress" : "20.23.12.1/30",
        "mtu" : "inherit",
        "routingPolicy" : "msc-ospf-policy-default"
      } ],
      "bgpPeeringEnabled" : true,
      "msiteControlPlaneTep" : "22.22.22.21",
      "routeReflectorEnabled" : true
    } ]
  } ],
  "status" : {
    "state" : "success"
  }
}
```

The following is an example of how to include fabric connectivity status for a site.

GET:

```
GET /api/v1/sites/fabric-connectivity?include=status
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200
{
  ...

  "sites" : [ {
    "id" : "58dd7707120000370089450f",
    ...
 "status" : {
  "state" : "success"
 }
```

```
  }, {
    "id" : "58dd77071200003600894510",
 ...
    "status" : {
  "state" : "in-progress"
 }
  }, {
    "id" : "58dd77071200003600894520",
 ...
    "status" : {
  "state" : "failure",
  "message" : "Failed to sync with site 58dd77071200003600894520."
 }
  } ]
}
```

The following is an example of how to obtain fabric connectivity health faults.

GET:

```
GET /api/v1/sites/fabric-connectivity?include=health,faults
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200
{
  ...

  "sites" : [ {
    "id" : "58dd77071200003700894501",
    ...
    "pods" : [ {
      "podId" : 1,
       "name" : "pod-1",
       ...
       "spines" : [ {
         "nodeId" : 201,
         "name" : "spine1",
         ...
         "health" : { "score": 100 }
   "faults" : [ {
      "code":"F0103",
               "message": "...",
               "type": "minor"
             } ]
       } ]
    } ]
  } ]
}
```

Example for ACI Multi-Site Release 1.0(2)

The following is an example of how to get fabric connectivity for a site.

GET:

```
GET /api/v1/sites/fabric-connectivity
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200HTTP/1.1 200 {
"controlPlaneBgpConfig" : {
    "peeringType" : "full-mesh | route-reflector",
    "ttl" : 16,
    "keepAliveInterval" : 60,
    "holdInterval" : 180,
    "staleInterval" : 300,
    "gracefulRestartEnabled" : true,
    "maxAsLimit" : 0
  },
  "ospfPolicies" : [ {
    "name" : "msc-ospf-policy-default",
    "networkType" : "broadcast | point-to-point | unspecified",
    "priority" : 1,
    "interfaceCost" : 1,
   "interfaceControls" : [ "advertise-subnet", "mtu-ignore", "bfd", "passive-participation"
 ], // Array of strings.. values are optional.. those are all possible values
    "helloInterval" : 10,
    "deadInterval" : 40,
    "retransmitInterval" : 5,
    "transmitDelay" : 1
  } ],
  "sites" : [ {
    "id" : "5935b0b633000088017723a4",
    "apicSiteId" : 1,
    "fabricId" : 1,
    "msiteEnabled" : true,
    "msiteDataPlaneMulticastTep" : "11.30.1.1",
    "bgpAsn" : 1234,
    "bgpPassword" : "",
    "ospfAreaId" : "57",      // Number or in IP address format
    "ospfAreaType" : "nssa | regular | stub",
    "ospfAreaCost" : 1,
    "externalRoutedDomain" : "uni/l3dom-MultisiteL3Domain1-DoNotDelete",
    "externalSubnetsToImport" : [ "20.23.10.0/16", "20.23.12.0/16"],
    "pods" : [ {
      "podId" : 1,
      "name" : "pod-1",
      "msiteDataPlaneUnicastTep" : "11.20.1.1",
      "spines" : [ {
        "nodeId" : 201,
        "name" : "spine1",
        "ports" : [ {
          "portId" : "1/29",
          "ipAddress" : "20.23.10.1/24",
          "mtu" : "inherit",      // value is 'inherit' or number from range 576 to 9000

          "routingPolicy" : "msc-ospf-policy-default"
        } ],
        "bgpPeeringEnabled" : true,
        "msiteControlPlaneTep" : "11.11.11.11",
        "routeReflectorEnabled" : true,
        "health" : {
          "score" : 93
        },
        "faults" : [ {
```

```
          "type" : "minor",
          "message" : "This fault occurs when a ntp configuration on a controller has
problems",
          "code" : "F1700"
        }, {
          "type" : "minor",
          "message" : "This fault occurs when an End Point Group is incompletely or
incorrectly configured.",
          "code" : "F0467"
        } ]
      } ],
      "health" : {
        "score" : 87
      },
      "faults" : [ {
        "type" : "major",
        "message" : "This fault occurs when vpc interface goes down while peer interface
is also down.",
        "code" : "F1296"
      }, { "type" : "minor",
        "message" : "This fault occurs when psu is shut down",
        "code" : "F1451"
      } ]
    } ],
    "status" : {
      "state" : "success"
    },
    "health" : {
      "score" : 87
    },
    "faults" : [ {
      "type" : "major",
      "message" : "This fault occurs when a physical interface on a controller is in the
link-down state.",
      "code" : "F0103"
    }, {
      "type" : "major",
      "message" : "Task or FSM failure on object of class dhcpPool",
      "code" : "F608160"
    } ]
  } ]
}
```

The following is an example of how to include fabric connectivity status for a site.

GET:

```
GET /api/v1/sites/fabric-connectivity?include=status
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200
{
  ...

  "sites" : [ {
    "id" : "58dd7707120000370089450f",
    ...
```

```
    "status" : {
        "state" : "success"
    }
  }, {
    "id" : "58dd77071200003600894510",
    ...
    "status" : {
        "state" : "in-progress"
    }
  }, {
    "id" : "58dd77071200003600894520",
    ...
    "status" : {
        "state" : "failure",
        "message" : "Failed to sync with site 58dd77071200003600894520."
    }
  } ]
}
```

The following is an example of how to obtain fabric connectivity health faults.

GET:

```
GET /api/v1/sites/fabric-connectivity?include=health,faults
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200HTTP/1.1 200
{
  ...

  "sites" : [ {
    "id" : "58dd770712000370089450f",
    ...
    "pods" : [ {
        "podId" : 1,
        "name" : "pod-1",
        ...
        "spines" : [ {
            "nodeId" : 201,
            "name" : "spine1",
            ...
            "health" : { "score": 100 }
            "faults" : [ {
               "code":"F0103",
               "message": "...",
               "type": "minor"
             } ]
        } ]
    } ]
  } ]
}
```

Example for ACI Multi-Site Release 1.0(1)

The following is an example of how to get fabric connectivity for a site.

GET:

```
GET https://<host>:<port>/api/v1/sites/fabric-connectivity
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200HTTP/1.1 200 {
"controlPlaneBgpConfig" : {
    "peeringType" : "full-mesh | route-reflector",
    "ttl" : 16,
    "keepAliveInterval" : 60,
    "holdInterval" : 180,
    "staleInterval" : 300,
    "gracefulRestartEnabled" : true,
    "maxAsLimit" : 0
  },
  "ospfPolicies" : [ {
    "name" : "msc-ospf-policy-default",
    "networkType" : "broadcast | point-to-point | unspecified",
    "priority" : 1,
    "interfaceCost" : 1,
   "interfaceControls" : [ "advertise-subnet", "mtu-ignore", "bfd", "passive-participation"
 ], // Array of strings.. values are optional.. those are all possible values
    "helloInterval" : 10,
    "deadInterval" : 40,
    "retransmitInterval" : 5,
    "transmitDelay" : 1
  } ],
  "sites" : [ {
    "id" : "5935b0b633000088017723a4",
    "apicSiteId" : 1,
    "fabricId" : 1,
    "msiteEnabled" : true,
    "msiteDataPlaneMulticastTep" : "11.30.1.1",
    "bgpAsn" : 1234,
    "bgpCommunity" : "extended:as2-nn4:4:15",
    "bgpPassword" : "",
    "ospfAreaId" : "57",      // Number or in IP address format
    "ospfAreaType" : "nssa | regular | stub",
    "ospfAreaCost" : 1,
    "externalRoutedDomain" : "uni/l3dom-MultisiteL3Domain1-DoNotDelete",
    "externalSubnetsToImport" : [ "20.23.10.0/16", "20.23.12.0/16"],
    "pods" : [ {
      "podId" : 1,
      "name" : "pod-1",
      "msiteDataPlaneUnicastTep" : "11.20.1.1",
      "spines" : [ {
        "nodeId" : 201,
        "name" : "spine1",
        "ports" : [ {
          "portId" : "1/29",
          "ipAddress" : "20.23.10.1/24",
          "mtu": "inherit",      // value is 'inherit' or number from range 576 to 9000

          "routingPolicy" : "msc-ospf-policy-default"
        } ],
        "bgpPeeringEnabled" : true,
        "msiteControlPlaneTep" : "11.11.11.11",
```

```
        "routeReflectorEnabled" : true,
        "health" : {
          "score" : 93
        },
        "faults" : [ {
          "type" : "minor",
          "message" : "This fault occurs when a ntp configuration on a controller has
problems",
          "code" : "F1700"
        }, {
          "type" : "minor",
          "message" : "This fault occurs when an End Point Group is incompletely or
incorrectly configured.",
          "code" : "F0467"
        } ]
      } ],
      "health" : {
        "score" : 87
      },
      "faults" : [ {
        "type" : "major",
        "message" : "This fault occurs when vpc interface goes down while peer interface
is also down.",
        "code" : "F1296"
      }, { "type" : "minor",
        "message" : "This fault occurs when psu is shut down",
        "code" : "F1451"
      } ]
    } ],
    "status" : {
      "state" : "success"
    },
    "health" : {
      "score" : 87
    },
    "faults" : [ {
      "type" : "major",
      "message" : "This fault occurs when a physical interface on a controller is in the
link-down state.",
      "code" : "F0103"
    }, {
      "type" : "major",
      "message" : "Task or FSM failure on object of class dhcpPool",
      "code" : "F608160"
    } ]
  } ]
}
```

The following is an example of how to include fabric connectivity status for a site.

GET:

```
GET https://<host>:<port>/api/v1/sites/fabric-connectivity?include=status
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200
{
```

```
  ...

  "sites" : [ {
    "id" : "58dd77071200003700894501f",
    ...
    "status" : {
        "state" : "success"
    }
  }, {
    "id" : "58dd77071200003600894510",
    ...
    "status" : {
        "state" : "in-progress"
    }
  }, {
    "id" : "58dd77071200003600894520",
    ...
    "status" : {
        "state" : "failure",
        "message" : "Failed to sync with site 58dd77071200003600894520."
    }
  } ]
}
```

The following is an example of how to obtain fabric connectivity health faults.

GET:

```
GET https://<host>:<port>/api/v1/sites/fabric-connectivity?include=health,faults
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200HTTP/1.1 200
{
  ...

  "sites" : [ {
    "id" : "58dd77071200003700894501f",
    ...
    "pods" : [ {
        "podId" : 1,
        "name" : "pod-1",
        ...
        "spines" : [ {
            "nodeId" : 201,
            "name" : "spine1",
            ...
            "health" : { "score": 100 }
            "faults" : [ {
               "code":"F0103",
               "message": "...",
               "type": "minor"
             } ]
        } ]
    } ]
  } ]
}
```

Get Fabric Connectivity for Multiple Sites

Example for ACI Multi-Site Release 1.2(1)

The following is an example of how to include fabric connectivity status for multiple sites.

GET:

```
GET /api/v1/sites/fabric-connectivity
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200
{
  "controlPlaneBgpConfig" : {
    "peeringType" : "full-mesh",
    "ttl" : 16,
    "keepAliveInterval" : 60,
    "holdInterval" : 180,
    "staleInterval" : 300,
    "gracefulRestartEnabled" : true,
    "maxAsLimit" : 0
  },
  "sites" : [ {
    "id" : "5ae37e2736000036007aa08e",
    "apicSiteId" : 1,
    "fabricId" : 1,
    "msiteEnabled" : false,
    "msiteDataPlaneMulticastTep" : "11.11.11.1",
    "bgpAsn" : 1234,
    "bgpPassword" : "",
    "ospfAreaId" : "0.0.0.8",
    "ospfAreaType" : "nssa",
    "ospfPolicies" : [ {
      "name" : "common/default",
      "networkType" : "unspecified",
      "priority" : 1,
      "interfaceCost" : 0,
      "interfaceControls" : [ ],
      "helloInterval" : 10,
      "deadInterval" : 40,
      "retransmitInterval" : 5,
      "transmitDelay" : 1
    }, {
      "name" : "msc-ospf-policy-default",
      "networkType" : "point-to-point",
      "priority" : 1,
      "interfaceCost" : 0,
      "interfaceControls" : [ ],
      "helloInterval" : 10,
      "deadInterval" : 40,
      "retransmitInterval" : 5,
      "transmitDelay" : 1
    } ],
    "pods" : [ {
      "podId" : 1,
      "name" : "pod-1",
```

```
      "msiteDataPlaneUnicastTep" : "31.31.31.1",
      "spines" : [ {
        "nodeId" : 201,
        "name" : "spine1",
        "ports" : [ {
          "portId" : "1/29",
          "ipAddress" : "20.23.10.1/30",
          "mtu" : "inherit",
          "routingPolicy" : "default"
        } ],
        "bgpPeeringEnabled" : true,
        "msiteControlPlaneTep" : "21.21.21.1",
        "routeReflectorEnabled" : true
      } ]
    } ]
  }, {
    "id" : "5ae37e2836000034007aa08f",
    "apicSiteId" : 2,
    "fabricId" : 2,
    "msiteEnabled" : false,
    "msiteDataPlaneMulticastTep" : "12.12.12.1",
    "bgpAsn" : 1234,
    "bgpPassword" : "",
    "ospfAreaId" : "0.0.0.8",
    "ospfAreaType" : "nssa",
    "ospfPolicies" : [ {
      "name" : "common/default",
      "networkType" : "unspecified",
      "priority" : 1,
      "interfaceCost" : 0,
      "interfaceControls" : [ ],
      "helloInterval" : 10,
      "deadInterval" : 40,
      "retransmitInterval" : 5,
      "transmitDelay" : 1
    }, {
      "name" : "msc-ospf-policy-default",
      "networkType" : "point-to-point",
      "priority" : 1,
      "interfaceCost" : 0,
      "interfaceControls" : [ ],
      "helloInterval" : 10,
      "deadInterval" : 40,
      "retransmitInterval" : 5,
      "transmitDelay" : 1
    } ],
    "pods" : [ {
      "podId" : 1,
      "name" : "pod-1",
      "msiteDataPlaneUnicastTep" : "32.32.32.1",
      "spines" : [ {
        "nodeId" : 211,
        "name" : "spine11",
        "ports" : [ {
          "portId" : "1/29",
          "ipAddress" : "20.23.12.1/30",
          "mtu" : "inherit",
          "routingPolicy" : "msc-ospf-policy-default"
        } ],
        "bgpPeeringEnabled" : true,
        "msiteControlPlaneTep" : "22.22.22.21",
        "routeReflectorEnabled" : true
      } ]
    } ]
```

```
  } ]
}
```

Get Fabric Connectivity Multipod Information

The following is an example of how to get fabric connectivity Multipod information.

GET:

```
GET /api/v1/sites/id/fabric-connnectivity/mpod-info
```

For example:

```
GET /api/v1/sites/5a5507e1340000360022f3ca/fabric-connectivity/mpod-info
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200
{
"id" : "5a5507e1340000360022f3ca",
  "pods" : [ {
    "podId" : 1,
    "name" : "pod-1",
    "mpodDataPlaneUnicastTep" : "200.0.0.1",
    "spines" : [ {
      "nodeId" : 211,
      "name" : "spine11",
      "mpodControlPlaneTep" : 201.0.0.1
    } ]
  },
  {
    "podId" : 2,
    "name" : "pod-2",
    "mpodDataPlaneUnicastTep" : "200.0.0.2",
    "spines" : [ {
      "nodeId" : 212,
      "name" : "spine12",
      "mpodControlPlaneTep" : 201.0.0.2
    } ]
  } ]
}
```

Get Fabric Connectivity Status

This API will provide fabric connectivity status for all sites. The user interface can derive the combined status from the status from individual sites. Look at the API to get the combined-status for a site.

The user interface should tell exactly which `bgp-session/tunnel` is in a `down` state, including local-node and peer-node info, when the combined-status is down for a site. Hover over the site to get this information.

The following is an example of how to get fabric connectivity status.

GET:

```
GET /api/v1/sites/fabric-connectivity-status
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200

{
  "sites" : [
    {
      "siteId": "59a06d4f33000033003ddc01",
      "siteName": "site1",
      "error": "",          <--  When 'error' is empty or doesn't exist, derive combined
                                 connectivity-status for this site from all individual
                                 status under msiteBgpSessionsStatus,
                                 msiteUnicastTunnelsStatus & msiteMulticastTunnelsStatus
                                 under this site

      "msiteBgpSessionsStatus": [
        {
          "podId": 1,
          "nodeId": 101,
          "nodeName": "spine11",
          "peerSiteId": "59a06d4f33000033003ddc02",
          "peerSiteName": "site2",
          "peerPodId": 1,
          "peerNodeId": 201,
          "peerNodeName": "spine21",
          "status": "up | down"
        },
        {
          "podId": 1,
          "nodeId": 101,
          "nodeName": "spine11",
          "peerSiteId": "59a06d4f33000033003ddc02",
          "peerSiteName": "site2",
          "peerPodId": 1,
          "peerNodeId": 202,
          "peerNodeName": "spine22",
          "status": "up | down"
        },
        {
          "podId": 1,
          "nodeId": 102,
          "nodeName": "spine12",
          "peerSiteId": "59a06d4f33000033003ddc02",
          "peerSiteName": "site2",
          "peerPodId": 1,
          "peerNodeId": 201,
          "peerNodeName": "spine21",
          "status": "up | down"
        },
        {
          "podId": 1,
          "nodeId": 102,
          "nodeName": "spine12",
          "peerSiteId": "59a06d4f33000033003ddc02",
          "peerSiteName": "site2",
          "peerPodId": 1,
```

```
      "peerNodeId": 202,
      "peerNodeName": "spine22",
      "status": "up | down"
    },
    {
      "podId": 1,
      "nodeId": 101,
      "nodeName": "spine11",
      "peerSiteId": "59a06d4f33000033003ddc03",
      "peerSiteName": "site3",
      "peerPodId": 1,
      "peerNodeId": 301,
      "peerNodeName": "spine31",
      "status": "up | down"
    },
    {
      "podId": 1,
      "nodeId": 101,
      "nodeName": "spine11",
      "peerSiteId": "59a06d4f33000033003ddc03",
      "peerSiteName": "site3",
      "peerPodId": 1,
      "peerNodeId": 302,
      "peerNodeName": "spine32",
      "status": "up | down"
    },
    {
      "podId": 1,
      "nodeId": 102,
      "nodeName": "spine12",
      "peerSiteId": "59a06d4f33000033003ddc03",
      "peerSiteName": "site3",
      "peerPodId": 1,
      "peerNodeId": 301,
      "peerNodeName": "spine31",
      "status": "up | down"
    },
    {
      "podId": 1,
      "nodeId": 102,
      "nodeName": "spine12",
      "peerSiteId": "59a06d4f33000033003ddc03",
      "peerSiteName": "site3",
      "peerPodId": 1,
      "peerNodeId": 302,
      "peerNodeName": "spine32",
      "status": "up | down"
    }
  ],
  "msiteVxlanUnicastTunnelsStatus": [
    {
      "podId": 1,
      "nodeId": 101,
      "nodeName": "spine11",
      "peerSiteId": "59a06d4f33000033003ddc02",
      "peerSiteName": "site2",
      "peerPodId": 1,
      "peerNodeId": 201,
      "peerNodeName": "spine21",
      "status": "up | down"
    },
    {
      "podId": 1,
      "nodeId": 101,
```

```
    "nodeName": "spine11",
    "peerSiteId": "59a06d4f33000033003ddc02",
    "peerSiteName": "site2",
    "peerPodId": 1,
    "peerNodeId": 202,
    "peerNodeName": "spine22",
    "status": "up | down"
  },
  {
    "podId": 1,
    "nodeId": 102,
    "nodeName": "spine12",
    "peerSiteId": "59a06d4f33000033003ddc02",
    "peerSiteName": "site2",
    "peerPodId": 1,
    "peerNodeId": 201,
    "peerNodeName": "spine21",
    "status": "up | down"
  },
  {
    "podId": 1,
    "nodeId": 102,
    "nodeName": "spine12",
    "peerSiteId": "59a06d4f33000033003ddc02",
    "peerSiteName": "site2",
    "peerPodId": 1,
    "peerNodeId": 202,
    "peerNodeName": "spine22",
    "status": "up | down"
  },
  {
    "podId": 1,
    "nodeId": 101,
    "nodeName": "spine11",
    "peerSiteId": "59a06d4f33000033003ddc03",
    "peerSiteName": "site3",
    "peerPodId": 1,
    "peerNodeId": 301,
    "peerNodeName": "spine31",
    "status": "up | down"
  },
  {
    "podId": 1,
    "nodeId": 101,
    "nodeName": "spine11",
    "peerSiteId": "59a06d4f33000033003ddc03",
    "peerSiteName": "site3",
    "peerPodId": 1,
    "peerNodeId": 302,
    "peerNodeName": "spine32",
    "status": "up | down"
  },
  {
    "podId": 1,
    "nodeId": 102,
    "nodeName": "spine12",
    "peerSiteId": "59a06d4f33000033003ddc03",
    "peerSiteName": "site3",
    "peerPodId": 1,
    "peerNodeId": 301,
    "peerNodeName": "spine31",
    "status": "up | down"
  },
  {
```

```
      "podId": 1,
      "nodeId": 102,
      "nodeName": "spine12",
      "peerSiteId": "59a06d4f33000033003ddc03",
      "peerSiteName": "site3",
      "peerPodId": 1,
      "peerNodeId": 302,
      "peerNodeName": "spine32",
      "status": "up | down"
    }
  ],
  "msiteVxlanMulticastTunnelsStatus": [
    {
      "podId": 1,
      "nodeId": 101,
      "nodeName": "spine11",
      "peerSiteId": "59a06d4f33000033003ddc02",
      "peerSiteName": "site2",
      "peerPodId": 1,
      "peerNodeId": 201,
      "peerNodeName": "spine21",
      "status": "up | down"
    },
    {
      "podId": 1,
      "nodeId": 101,
      "nodeName": "spine11",
      "peerSiteId": "59a06d4f33000033003ddc02",
      "peerSiteName": "site2",
      "peerPodId": 1,
      "peerNodeId": 202,
      "peerNodeName": "spine22",
      "status": "up | down"
    },
    {
      "podId": 1,
      "nodeId": 102,
      "nodeName": "spine12",
      "peerSiteId": "59a06d4f33000033003ddc02",
      "peerSiteName": "site2",
      "peerPodId": 1,
      "peerNodeId": 201,
      "peerNodeName": "spine21",
      "status": "up | down"
    },
    {
      "podId": 1,
      "nodeId": 102,
      "nodeName": "spine12",
      "peerSiteId": "59a06d4f33000033003ddc02",
      "peerSiteName": "site2",
      "peerPodId": 1,
      "peerNodeId": 202,
      "peerNodeName": "spine22",
      "status": "up | down"
    },
    {
      "podId": 1,
      "nodeId": 101,
      "nodeName": "spine11",
      "peerSiteId": "59a06d4f33000033003ddc03",
      "peerSiteName": "site3",
      "peerPodId": 1,
      "peerNodeId": 301,
```

```
            "peerNodeName": "spine31",
            "status": "up | down"
          },
          {
            "podId": 1,
            "nodeId": 101,
            "nodeName": "spine11",
            "peerSiteId": "59a06d4f33000033003ddc03",
            "peerSiteName": "site3",
            "peerPodId": 1,
            "peerNodeId": 302,
            "peerNodeName": "spine32",
            "status": "up | down"
          },
          {
            "podId": 1,
            "nodeId": 102,
            "nodeName": "spine12",
            "peerSiteId": "59a06d4f33000033003ddc03",
            "peerSiteName": "site3",
            "peerPodId": 1,
            "peerNodeId": 301,
            "peerNodeName": "spine31",
            "status": "up | down"
          },
          {
            "podId": 1,
            "nodeId": 102,
            "nodeName": "spine12",
            "peerSiteId": "59a06d4f33000033003ddc03",
            "peerSiteName": "site3",
            "peerPodId": 1,
            "peerNodeId": 302,
            "peerNodeName": "spine32",
            "status": "up | down"
          }
        ]
      },
      {
        "siteId": "59a06d4f33000033003ddc02",
        "siteName": "site2",
        "error" : "site not reachable", <-- When this is non-empty, individual bgp-session
                                           or tunnel statuses would be empty list and
                                           make combined status to be down and show this
                                           error in summary/details page
        "msiteBgpSessionsStatus": [ ],
        "msiteVxlanUnicastTunnelsStatus": [],
        "msiteVxlanMulticastTunnelsStatus": []
      },
      {
        "siteId": "59a06d4f33000033003ddc03",
        "siteName": "site3",
                                             <-- 'error' won't exist or empty when MSC able

                                             to fetch bgp or tunnel statues from
                                             APICs/sites
        "msiteBgpSessionsStatus": [........],
        "msiteVxlanUnicastTunnelsStatus": [........],
        "msiteVxlanMulticastTunnelsStatus": [.......]
      }
    ]
```

Update Fabric Connectivity

Example for ACI Multi-Site Release 1.2(1)

The following is an example of how to get an update for fabric connectivity.

PUT:

```
PUT /api/v1/sites/fabric-connectivity
```

Request:

```
{
  "controlPlaneBgpConfig" : {
    "peeringType" : "full-mesh | route-reflector",
    "ttl" : 16,
    "keepAliveInterval" : 60,
    "holdInterval" : 180,
    "staleInterval" : 300,
    "gracefulRestartEnabled" : true,
    "maxAsLimit" : 0
  },
  "sites" : [ {
    "id" : "5ae3743e4a00003500bc19ad",
    "apicSiteId" : 1,
    "fabricId" : 1,
    "msiteEnabled" : true,
    "msiteDataPlaneMulticastTep" : "11.11.11.1",
    "bgpAsn" : 1234,
    "bgpPassword" : "",
    "ospfAreaId" : "8",
            // Number or in IP address format
    "ospfAreaType" : "nssa | regular | stub",
    "externalRoutedDomain" : "uni/l3dom-MultisiteL3Domain1-DoNotDelete",

    "ospfPolicies" : [ {
      "name" : "msc-ospf-policy-default",
      "networkType" : "broadcast | point-to-point | unspecified",
      "priority" : 1,
      "interfaceCost" : 0,
     "interfaceControls" : [ "advertise-subnet", "mtu-ignore", "bfd", "passive-participation"
 ],     // Values are optional
      "helloInterval" : 10,
      "deadInterval" : 40,
      "retransmitInterval" : 5,
      "transmitDelay" : 1
    }, {
      "name" : "common/default",
      "networkType" : "unspecified",
      "priority" : 1,
      "interfaceCost" : 0,
      "interfaceControls" : [ ],
      "helloInterval" : 10,
      "deadInterval" : 40,
      "retransmitInterval" : 5,
      "transmitDelay" : 1
    } ],
    "pods" : [ {
      "podId" : 1,
      "name" : "pod-1",
      "msiteDataPlaneUnicastTep" : "31.31.31.1",
      "spines" : [ {
```

```
        "nodeId" : 201,
        "name" : "spine1",
        "ports" : [ {
          "portId" : "1/29",
          "ipAddress" : "20.23.10.1/30",
          "mtu" : "inherit",                                          // value is 'inherit' or
number from range 576 to 9000
          "routingPolicy" : "msc-ospf-policy-default"
        } ],
        "bgpPeeringEnabled" : true,
        "msiteControlPlaneTep" : "21.21.21.1",
        "routeReflectorEnabled" : true
      } ]
    } ]
  }, {
    "id" : "5ae3743e4a00004b00bc19ae",
    "apicSiteId" : 2,
    "fabricId" : 2,
    "msiteEnabled" : true,
    "msiteDataPlaneMulticastTep" : "12.12.12.1",
    "bgpAsn" : 1234,
    "bgpPassword" : "",
    "ospfAreaId" : "8",
    "ospfAreaType" : "nssa",
    "externalRoutedDomain" : "uni/l3dom-MultisiteL3Domain2-DoNotDelete",

    "ospfPolicies" : [ {
      "name" : "msc-ospf-policy-default",
      "networkType" : "point-to-point",
      "priority" : 1,
      "interfaceCost" : 0,
      "interfaceControls" : [ ],
      "helloInterval" : 10,
      "deadInterval" : 40,
      "retransmitInterval" : 5,
      "transmitDelay" : 1
    }, {
      "name" : "msc-ospf-policy-default",
      "networkType" : "unspecified",
      "priority" : 1,
      "interfaceCost" : 0,
      "interfaceControls" : [ ],
      "helloInterval" : 10,
      "deadInterval" : 40,
      "retransmitInterval" : 5,
      "transmitDelay" : 1
    } ],
    "pods" : [ {
      "podId" : 1,
      "name" : "pod-1",
      "msiteDataPlaneUnicastTep" : "32.32.32.1",
      "spines" : [ {
        "nodeId" : 211,
        "name" : "spine11",
        "ports" : [ {
          "portId" : "1/29",
          "ipAddress" : "20.23.12.1/30",
          "mtu" : "inherit",
          "routingPolicy" : "msc-ospf-policy-default"
        } ],
        "bgpPeeringEnabled" : true,
        "msiteControlPlaneTep" : "22.22.22.21",
        "routeReflectorEnabled" : true
      } ]
```

```
    } ]
  } ]
}
```

Response:

```
HTTP/1.1 200
{
  "controlPlaneBgpConfig" : {
    "peeringType" : "full-mesh",
    "ttl" : 16,
    "keepAliveInterval" : 60,
    "holdInterval" : 180,
    "staleInterval" : 300,
    "gracefulRestartEnabled" : true,
    "maxAsLimit" : 0
  },
  "sites" : [ {
    "id" : "5ae378544c00004c007e29da",
    "apicSiteId" : 1,
    "fabricId" : 1,
    "msiteEnabled" : true,
    "msiteDataPlaneMulticastTep" : "11.11.11.1",
    "bgpAsn" : 1234,
    "bgpPassword" : "",
    "ospfAreaId" : "8",
    "ospfAreaType" : "nssa",
    "externalRoutedDomain" : "uni/l3dom-MultisiteL3Domain1-DoNotDelete",

    "ospfPolicies" : [ {
      "name" : "common/default",
      "networkType" : "unspecified",
      "priority" : 1,
      "interfaceCost" : 0,
      "interfaceControls" : [ ],
      "helloInterval" : 10,
      "deadInterval" : 40,
      "retransmitInterval" : 5,
      "transmitDelay" : 1
    }, {
      "name" : "msc-ospf-policy-default",
      "networkType" : "point-to-point",
      "priority" : 1,
      "interfaceCost" : 0,
      "interfaceControls" : [ ],
      "helloInterval" : 10,
      "deadInterval" : 40,
      "retransmitInterval" : 5,
      "transmitDelay" : 1
    } ],
    "pods" : [ {
      "podId" : 1,
      "name" : "pod-1",
      "msiteDataPlaneUnicastTep" : "31.31.31.1",
      "spines" : [ {
        "nodeId" : 201,
        "name" : "spine1",
        "ports" : [ {
          "portId" : "1/29",
          "ipAddress" : "20.23.10.1/30",
          "mtu" : "inherit",
          "routingPolicy" : "common/default"
        } ],
```

```
      "bgpPeeringEnabled" : true,
      "msiteControlPlaneTep" : "21.21.21.1",
      "routeReflectorEnabled" : true
    } ]
  } ],
  "status" : {
    "state" : "success"                                    // Deploy status for site

  }
}, {
  "id" : "5ae378544c00004c007e29db",
  "apicSiteId" : 2,
  "fabricId" : 2,
  "msiteEnabled" : true,
  "msiteDataPlaneMulticastTep" : "12.12.12.1",
  "bgpAsn" : 1234,
  "bgpPassword" : "",
  "ospfAreaId" : "8",
  "ospfAreaType" : "nssa",
  "externalRoutedDomain" : "uni/l3dom-MultisiteL3Domain2-DoNotDelete",
  "ospfPolicies" : [ {
    "name" : "common/default",
    "networkType" : "unspecified",
    "priority" : 1,
    "interfaceCost" : 0,
    "interfaceControls" : [ ],
    "helloInterval" : 10,
    "deadInterval" : 40,
    "retransmitInterval" : 5,
    "transmitDelay" : 1
  }, {
    "name" : "msc-ospf-policy-default",
    "networkType" : "point-to-point",
    "priority" : 1,
    "interfaceCost" : 0,
    "interfaceControls" : [ ],
    "helloInterval" : 10,
    "deadInterval" : 40,
    "retransmitInterval" : 5,
    "transmitDelay" : 1
  } ],
  "pods" : [ {
    "podId" : 1,
    "name" : "pod-1",
    "msiteDataPlaneUnicastTep" : "32.32.32.1",
    "spines" : [ {
      "nodeId" : 211,
      "name" : "spine11",
      "ports" : [ {
        "portId" : "1/29",
        "ipAddress" : "20.23.12.1/30",
        "mtu" : "inherit",
        "routingPolicy" : "msc-ospf-policy-default"
      } ],
      "bgpPeeringEnabled" : true,
      "msiteControlPlaneTep" : "22.22.22.21",
      "routeReflectorEnabled" : true
    } ]
  } ],
  "status" : {
    "state" : "success"
  }
} ]
```

```
}
```

Example for ACI Multi-Site Release 1.0(2)

The following is an example of how to get an update for fabric connectivity.

PUT:

```
PUT /api/v1/sites/fabric-connectivity
```

Request:

```
{
"controlPlaneBgpConfig" : {
    "peeringType" : "full-mesh | route-reflector",
    "ttl" : 16,
    "keepAliveInterval" : 60,
    "holdInterval" : 180,
    "staleInterval" : 300,
    "gracefulRestartEnabled" : true,
    "maxAsLimit" : 0
  },
  "ospfPolicies" : [ {
    "name" : "msc-ospf-policy-default",
    "networkType" : "broadcast | point-to-point | unspecified",
    "priority" : 1,
    "interfaceCost" : 1,
   "interfaceControls" : [ "advertise-subnet", "mtu-ignore", "bfd", "passive-participation"
 ], //values are optional
    "helloInterval" : 10,
    "deadInterval" : 40,
    "retransmitInterval" : 5,
    "transmitDelay" : 1
  } ],
  "sites" : [ {
    "id" : "5935b0b633000088017723a4",
    "apicSiteId" : 1,
    "fabricId" : 1,
    "msiteEnabled" : true,
    "msiteDataPlaneMulticastTep" : "11.30.1.1",
    "bgpAsn" : 1234,
    "bgpPassword" : "",
    "ospfAreaId" : "57",      // Number or in IP address format
    "ospfAreaType" : "nssa | regular | stub",
    "ospfAreaCost" : 1,
    "externalRoutedDomain" : "uni/l3dom-MultisiteL3Domain1-DoNotDelete",
    "externalSubnetsToImport" : [ "20.23.10.0/16", "20.23.12.0/16"],
    "pods" : [ {
      "podId" : 1,
      "name" : "pod-1",
      "msiteDataPlaneUnicastTep" : "11.20.1.1",
      "spines" : [ {
        "nodeId" : 201,
        "name" : "spine1",
        "ports" : [ {
          "portId" : "1/29",
          "ipAddress" : "20.23.10.1/24",
          "mtu" : "inherit",      // value is 'inherit' or number from range 576 to 9000

          "routingPolicy" : "msc-ospf-policy-default"
        } ],
        "bgpPeeringEnabled" : true,
        "msiteControlPlaneTep" : "11.11.11.11",
```

```
        "routeReflectorEnabled" : true,
          } ],
    } ],
  } ]
}
```

Response:

```
{
"controlPlaneBgpConfig" : {
    "peeringType" : "full-mesh | route-reflector",
    "ttl" : 16,
    "keepAliveInterval" : 60,
    "holdInterval" : 180,
    "staleInterval" : 300,
    "gracefulRestartEnabled" : true,
    "maxAsLimit" : 0
  },
  "ospfPolicies" : [ {
    "name" : "msc-ospf-policy-default",
    "networkType" : "broadcast | point-to-point | unspecified",
    "priority" : 1,
    "interfaceCost" : 1,
   "interfaceControls" : [ "advertise-subnet", "mtu-ignore", "bfd", "passive-participation"
 ], // Array of strings.. values are optional.. those are all possible values
    "helloInterval" : 10,
    "deadInterval" : 40,
    "retransmitInterval" : 5,
    "transmitDelay" : 1
  } ],
  "sites" : [ {
    "id" : "5935b0b633000088017723a4",
    "apicSiteId" : 1,
    "fabricId" : 1,
    "msiteEnabled" : true,
    "msiteDataPlaneMulticastTep" : "11.30.1.1",
    "bgpAsn" : 1234,
    "bgpPassword" : "",
    "ospfAreaId" : "57",      // Number or in IP address format
    "ospfAreaType" : "nssa | regular | stub",
    "ospfAreaCost" : 1,
    "externalRoutedDomain" : "uni/l3dom-MultisiteL3Domain1-DoNotDelete",
    "externalSubnetsToImport" : [ "20.23.10.0/16", "20.23.12.0/16"],
    "pods" : [ {
      "podId" : 1,
      "name" : "pod-1",
      "msiteDataPlaneUnicastTep" : "11.20.1.1",
      "spines" : [ {
        "nodeId" : 201,
        "name" : "spine1",
        "ports" : [ {
          "portId" : "1/29",
          "ipAddress" : "20.23.10.1/24",
          "mtu" : "inherit",      // value is 'inherit' or number from range 576 to 9000

          "routingPolicy" : "msc-ospf-policy-default"
        } ],
        "bgpPeeringEnabled" : true,
        "msiteControlPlaneTep" : "11.11.11.11",
        "routeReflectorEnabled" : true,
      } ],
    } ],
  } ]
```

```
}
```

Refresh Fabric Connectivity for a Site

Example for ACI Multi-Site Release 1.2(1)

The following is an example of how to refresh fabric connectivity for a site.

PUT:

```
PUT /api/v1/sites/id/fabric-connnectivity/refresh?removeDecommissiondNodes=yes|no
```

Example:

```
PUT /api/v1/sites/5aeccccb35000050001977bc/fabric-connectivity/refresh
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200

{
  "id" : "5aeccccb35000050001977bc",
  "apicSiteId" : 2,
  "fabricId" : 2,
  "msiteEnabled" : true,
  "msiteDataPlaneMulticastTep" : "12.12.12.1",
  "bgpAsn" : 1234,
  "bgpPassword" : "",
  "ospfAreaId" : "0.0.0.8",
  "ospfAreaType" : "nssa",
  "externalRoutedDomain" : "uni/l3dom-MultisiteL3Domain1-DoNotDelete",
  "ospfPolicies" : [ {
    "name" : "common/default",
    "networkType" : "unspecified",
    "priority" : 1,
    "interfaceCost" : 0,
    "interfaceControls" : [ ],
    "helloInterval" : 10,
    "deadInterval" : 40,
    "retransmitInterval" : 5,
    "transmitDelay" : 1
  }, {
    "name" : "msc-ospf-policy-default",
    "networkType" : "point-to-point",
    "priority" : 1,
    "interfaceCost" : 0,
    "interfaceControls" : [ ],
    "helloInterval" : 10,
    "deadInterval" : 40,
    "retransmitInterval" : 5,
    "transmitDelay" : 1
  } ],
  "pods" : [ {
    "podId" : 1,
    "name" : "pod-1",
    "msiteDataPlaneUnicastTep" : "32.32.32.1",
```

```
    "spines" : [ {
      "nodeId" : 211,
      "name" : "spine11",
      "ports" : [ {
        "portId" : "1/29",
        "ipAddress" : "20.23.12.1/30",
        "mtu" : "inherit",
        "routingPolicy" : "msc-ospf-policy-default"
      } ],
      "bgpPeeringEnabled" : true,
      "msiteControlPlaneTep" : "22.22.22.21",
      "routeReflectorEnabled" : true
    } ]
  } ],
  "status" : {
    "state" : "success"
  }
}
```

Example for ACI Multi-Site Release 1.0(2)

The following is an example of how to refresh fabric connectivity for a site.

PUT:

```
PUT /api/v1/sites/id/fabric-connnectivity/refresh
```

Example:

```
PUT /api/v1/sites/58dd77071200003600894510/fabric-connectivity/refresh
```

Request:

```
   <empty>
```

Response:

```
HTTP/1.1 200
{
      "id" : "5935b0b633000088017723a4",
    "apicSiteId" : 1,
    "fabricId" : 1,
    "msiteEnabled" : true,
    "msiteDataPlaneMulticastTep" : "11.30.1.1",
    "bgpAsn" : 1234,
    "bgpPassword" : "",
    "ospfAreaId" : "57",      // Number or in IP address format
    "ospfAreaType" : "nssa | regular | stub",
    "ospfAreaCost" : 1,
    "externalRoutedDomain" : "uni/l3dom-MultisiteL3Domain1-DoNotDelete",
    "externalSubnetsToImport" : [ "20.23.10.0/16", "20.23.12.0/16"],
    "pods" : [ {
      "podId" : 1,
      "name" : "pod-1",
      "msiteDataPlaneUnicastTep" : "11.20.1.1",
      "spines" : [ {
        "nodeId" : 201,
        "name" : "spine1",
        "ports" : [ {
          "portId" : "1/29",
          "ipAddress" : "20.23.10.1/24",
          "mtu" : "inherit",      // value is 'inherit' or number from range 576 to 9000
```

```
            "routingPolicy" : "msc-ospf-policy-default"
          } ],
          "bgpPeeringEnabled" : true,
          "msiteControlPlaneTep" : "11.11.11.11",
          "routeReflectorEnabled" : true,
        } ]
}
```

Fabric Leaf Switches And Physical Interfaces

Get Nodes

The following is an example of how to get nodes.

GET:

```
GET https://<host>:<port>/api/v1/aci/sites/:id/nodes?type=<leaf,spine>
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
    "nodes" : [ {
        "dn" : "topology/pod-1/node-101",
        "name" : "leaf1",
        "type" : "leaf"
    }, {
        "dn" : "topology/pod-1/node-102",
        "name" : "leaf2",
        "type" : "leaf"
    } ]
}
```

Get Physical Interfaces Details

The following is an example of how to get physical interfaces details.

GET:

```
GET https://<host>:<port>/api/v1/aci/sites/:id/nodes/interfaces?node=<node_dn>&type=physical
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200HTTP/1.1 200
```

```
{
  "interfaces": [
    {
      "dn": "topology/pod-1/paths-101/pathep-[eth1/1]",
      "name": "eth1/1",
      "type": "physical"
    },
    {
      "dn": "topology/pod-1/paths-101/pathep-[eth1/10]",
      "name": "eth1/10",
      "type": "physical"
    }
 ]
}
```

The following is an example of how to get physical interfaces, direct port channel type.

GET:

```
GET https://<host>:<port>/api/v1/aci/sites/:id/nodes/interfaces?type=dpc
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200HTTP/1.1 200
{
  "interfaces": [
    {
      "dn": "topology/pod-1/paths-101/pathep-[Switch101_1-ports-3-4_PolGrp]",
      "name": "Switch101_1-ports-3-4_PolGrp",
      "type": "dpc"
    }
  ]
}
```

The following is an example of how to get physical interfaces, virtual port channel type.

GET:

```
GET https://<host>:<port>/api/v1/aci/sites/:id/nodes/interfaces?type=vpc
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200HTTP/1.1 200
{
  "interfaces": [
    {
      "dn": "topology/pod-1/protpaths-101-102/pathep-[Switch101-102_1-ports-5_PolGrp]",
      "name": "Switch101-102_1-ports-5_PolGrp",
      "type": "vpc"
    }
  ]
}
```

APIs For Tenants

Create a Tenant

The following is an example of how to create a Tenant.

POST:

```
POST https://<host>:<port>/api/v1/tenants
```

Request:

```
{
    "displayName": "Tenant 2",
    "name": "Tenant2",
    "description": "Description for Tenant 2",
    "siteAssociations": [
        {
        "siteId": "58dfeae2fd0000fd00cee617",
        "securityDomains": ["TestSecurityDomain", "domain-mgmt"]
        }
    ],
    "userAssociations": [
        { "userId": "58202f7066e6e10001c41236" }
        { "userId": "58202fc466e6e10001c41237" }
    ]
}
```

Response:

```
HTTP/1.1 201
Location: /api/v1/tenants/58202f7066e6e10001c41236
{
  "id": "58ed806aa10000bb0044da1e",
  "name": "Tenant2",
  "displayName": "Tenant 2",
  "description": "Description for Tenant 2",
  "siteAssociations": [
    {
      "siteId": "58dfeae2fd0000fd00cee617",
      "securityDomains": [
        "TestSecurityDomain",
        "mgmt"
      ]
    }
  ],
  "userAssociations": [
    { "userId": "58202f7066e6e10001c41236" }
    { "userId": "58202fc466e6e10001c41237" }
  ]
}
```

Read Tenant Information

The following is an example of how to read Tenant information.

GET:

```
GET https://<host>:<port>/api/v1/tenants/58202f7066e6e10001c41236
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "id": "58ed806aa10000bb0044da1e",
  "name": "Tenant2",
  "displayName": "Tenant 2",
  "description": "Description for Tenant 2",
  "siteAssociations": [
    {
      "siteId": "58dfeae2fd0000fd00cee617",
      "securityDomains": [
        "TestSecurityDomain",
        "mgmt"
      ]
    }
  ],
  "userAssociations": [
    { "userId": "58202f7066e6e10001c41236" }
    { "userId": "58202fc466e6e10001c41237" }
  ]
}
```

Update an Existing Tenant

The following is an example of how to update an existing Tenant.

PUT:

```
PUT https://<host>:<port>/api/v1/tenants/58202f7066e6e10001c41236
```

Request:

```
{
  {
  "id": "58ed806aa10000bb0044da1e",
  "name": "Tenant2",
  "displayName": "Tenant 2 (Updated)",
  "description": "Description for Tenant 2",
  "siteAssociations": [
    {
      "siteId": "58dfeae2fd0000fd00cee617",
      "securityDomains": [
        "TestSecurityDomain",
        "mgmt"
      ]
    }
  ],
  "userAssociations": [
    { "userId": "58202f7066e6e10001c41236" }
    { "userId": "58202fc466e6e10001c41237" }
    { "userId": "58202fc466e6e10001c41238" }
  ]
}
```

Note "id" is optional, other attributes are mandatory even if unchanged.

Response:

```
HTTP/1.1 200
{
    "id": "58ed806aa10000bb0044da1e",
    "displayName": "Tenant 2 (Updated)",
    "name": "Tenant2",
    "description": "Description for Tenant 2",
    "siteAssociations": [
        {
        "siteId": "58dfeae2fd0000fd00cee617",
        "securityDomains": ["TestSecurityDomain", "mgmt"]
        }
    ],
    "userAssociations": [
      { "userId": "58202f7066e6e10001c41236" }
      { "userId": "58202fc466e6e10001c41237" }
      { "userId": "58202fc466e6e10001c41238" }
    ]
}
```

Delete a Tenant

The following is an example of how to delete a Tenant.

PUT:

```
DELETE https://<host>:<port>/api/v1/tenants/58202f7066e6e10001c41236
```

Request:

```
 <empty>
```

Response:

```
HTTP/1.1 204
```

List Tenants

The following is an example of how to list Tenants.

PUT:

```
PUT https://<host>:<port>/api/v1/tenants/58202f7066e6e10001c41236
```

Request:

```
<empty>
```

Note Filtering, pagination, and sorting is currently not supported.

Response:

```
HTTP/1.1 200
{
  "tenants": [
    {
      "id": "58ed7f22a10000a10044da19",
      "name": "test",
      "displayName": "test",
      "description": "Description for test",
      "siteAssociations": [
        {
          "siteId": "58dfeae2fd0000fd00cee617",
          "securityDomains": []
        }
      ],
      "userAssociations": [
      { "userId": "58202f7066e6e10001c41236" }
      { "userId": "58202fc466e6e10001c41237" }
      { "userId": "58202fc466e6e10001c41238" }
     ]
    },
    {
      "id": "58ed7f46a10000a50044da1a",
      "name": "Tenant1",
      "displayName": "Tenant 1",
      "description": "Description for Tenant 1",
      "siteAssociations": [
        {
          "siteId": "58dfeae2fd0000fd00cee617",
          "securityDomains": []
        }
      ],
      "userAssociations": [
      { "userId": "58202f7066e6e10001c41236" }
     ]
    },
    {
      "id": "58ed806aa10000bb0044da1e",
      "name": "Tenant2",
      "displayName": "Tenant 2 (Updated)",
      "description": "Description for Tenant 2",
      "siteAssociations": [
        {
          "siteId": "58dfeae2fd0000fd00cee617",
          "securityDomains": [
            "TestSecurityDomain",
            "mgmt"
          ]
        }
      ],
      "userAssociations": [
      { "userId": "58202f7066e6e10001c41236" }
     ]
    }
  ]
}
```

List Sites Associated with a Tenant

The following is an example of how to list sites associated with a Tenant.

GET:

```
GET /api/v1/tenants/allowed-sites
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "sites": [
    {
      "id": "591b6e5172040072044258ea",
      "name": "site1",
      "urls": [
        "https://10.23.236.232"
      ],
      "username": "admin",
      "password": "******",
      "siteId": "1"
    },
    {
      "id": "591b71353f00002700a2e8b1",
      "name": "site2",
      "urls": [
        "https://10.23.236.233"
      ],
      "username": "admin",
      "password": "******",
      "siteId": "2"
    }
  ]
}
```

List Users Associated with a Tenant

The following is an example of how to list users associated with a Tenant.

GET:

```
GET /api/v1/tenants/allowed-users
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "users": [
```

```
    {
      "id": "0000ffff0000000000000020",
      "username": "admin",
      "password": "******",
      "firstName": "Admin",
      "lastName": "Admin",
      "emailAddress": "admin@msc.com",
      "phoneNumber": "(888) 123-4567",
      "accountStatus": "active",
      "roles": [
        {
          "roleId": "0000ffff0000000000000031"
        }
      ]
    },
    {
      "id": "593b12e62000004b00f13b21",
      "username": "bob",
      "password": "******",
      "firstName": "Bob",
      "lastName": "Smith",
      "emailAddress": "bob@gmail.com",
      "phoneNumber": "(888) 123-7890",
      "accountStatus": "active",
      "roles": [
        {
          "roleId": "0000ffff0000000000000034"
        }
      ]
    },
    {
      "id": "593b12f62000005200f13b22",
      "username": "russ",
      "password": "******",
      "firstName": "Russ",
      "lastName": "King",
      "emailAddress": "russ@gmail.com",
      "phoneNumber": "(800) 123-4545",
      "accountStatus": "active",
      "roles": [
        {
          "roleId": "0000ffff0000000000000033"
        },
        {
          "roleId": "0000ffff0000000000000035"
        },
        {
          "roleId": "0000ffff0000000000000032"
        }
      ]
    }
  ]
}
```


List Schemas Associated with a Tenant

The following is an example of how to list Schemas associated with a Tenant.

GET:

```
GET /api/v1/tenants/:id/schemas
```

Request:

```
<empty>
```

Response:

```
{
  "schemas": [
    {
      "id": "594170153200003000a8b4fc",
      "displayName": "Schema 1",
      "templates": [],
      "sites": []
    },
    {
      "id": "594170153200003200a8b4fd",
      "displayName": "Schema 2",
      "templates": [],
      "sites": []
    },
    {
      "id": "594170153200003300a8b4ff",
      "displayName": "Schema 4",
      "templates": [],
      "sites": []
    }
  ]
}
```

List Schemas Associated with all Tenants

The following is an example of how to list Schemas associated with all Tenants.

GET:

```
GET /api/v1/tenants/schemas
```

Request:

```
<empty>
```

Response:

```
{
  "tenants": {
    "59416f953200004c008cc048": [
      {
        "id": "59416f9532000030008cc04a",
        "displayName": "Schema 1",
        "templates": [],
        "sites": []
      },
      {
        "id": "59416f9532000033008cc04b",
        "displayName": "Schema 2",
        "templates": [],
        "sites": []
      },
```

```
      {
        "id": "59416f9532000033008cc04d",
        "displayName": "Schema 4",
        "templates": [],
        "sites": []
      }
    ],
    "59416f9532000036008cc049": [
      {
        "id": "59416f9532000052008cc04c",
        "displayName": "Schema 3",
        "templates": [],
        "sites": []
      },
      {
        "id": "59416f9532000033008cc04d",
        "displayName": "Schema 4",
        "templates": [],
        "sites": []
      }
    ]
  }
}
```

Create a New Tenant Network Mapping

The following is an example of how to create a new Tenant network mapping.

PUT:

```
PUT /api/v1/tenants/:id/infra
```

Request:

```
{
  "externalEpgMappings": [
    {
      "siteMappings": [
        {
          "siteId": "59726e7a490000ea095b4cc8",
          "mapping": "uni/tn-Pepsi/out-L3Out-1/instP-Network"
        }
      ]
    }
  ]
}
```

Response:

```
HTTP/1.1 201
{
  "externalEpgMappings": [
    {
      "siteMappings": [
        {
          "siteId": "59726e7a490000ea095b4cc8",
          "mapping": "uni/tn-Pepsi/out-L3Out-1/instP-Network"
        }
      ]
    }
```

```
  ]
}
```

Get Existing Tenant Network Mappings

The following is an example of how to list existing Tenant network mappings.

GET:

```
GET /api/v1/tenants/:id/infra
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 201
{
  "externalEpgMappings": [
    {
      "siteMappings": [
        {
          "siteId": "59726e7a490000ea095b4cc8",
          "mapping": "uni/tn-Pepsi/out-L3Out-1/instP-Network"
        }
      ]
    }
  ]
}
```

APIs For All Schemas

Create a New Schema

Example for ACI Multi-Site Release 1.2(1)

Starting with ACI Multi-Site release 1.2(1), you will use the information in the request for this API call for certain service graph functions:

- At the template level, use information in the **serviceGraphRelationship** area in the request to associate a contract with a service graph. The **serviceGraphRelationship** area is highlighted in the example request below.
- At the template level, use information in the first **serviceGraphs** area in the request to provide information on the service graph template, such as:
  - The name of the service graph (in this example request, **SG1**)
  - Information on and number of service nodes in this service graph (in this example request, there are two nodes, **node1** and **node2**)

  The first **serviceGraphs** area is highlighted in the example request below.

- At the site local level, go to the **contracts** area in the request and use the information in the **serviceGraphRelationship** area to attach this service node and the cluster interfaces. The **serviceGraph** and **serviceNode** information in the **contracts** area is highlighted in the example request below.
- Finally, use information in the second **serviceGraphs** area in the request to add devices. The second **serviceGraphs** area is highlighted in the example request below.

For the Services feature, a Schema should have the following:

- A service graph template — You must have a service graph template before you can begin adding devices.
- L4-L7 devices — You fetch the devices from site local and attach these devices to the service graph template. Use the information in List L4-L7 Devices, on page 53 to have Multi-Site issue a query to APIC to get information about the devices.
- L4-L7 device interfaces — All L4-L7 devices have device interfaces. When you query (select) a device, you are then able to get the device interface for that device. List L4-L7 Device Interfaces, on page 54 describes how to gather this information, where the *devices* part of the GET query is a device that was provided from the output generated from the information in List L4-L7 Devices, on page 53.
- Redirect policies — After you pick a device interface, you would then pick the redirect policy, where Multi-Site issues a query to APIC to get all the redirect policies available. See List L4-L7 Redirect Policies, on page 54 for more information.

The order of the devices, device interfaces and redirect policies depends on the service graph. When you create a service graph, for each site, you will attach devices. When you want to deploy a service graph to a contract, you will need to add cluster interfaces and redirect policies. Only if the service graph template is associated with a contract can you add redirect policies and bridge domains to it. If you try to add redirect policies and you don't have a service graph template, then this will result in an error.

The following topics provide more information for service graphs from a schema level:

- Retrieve Service Graph Node Types, on page 192
- Create Service Graph Node Types, on page 193

Note For Multi-Site Service Integration use cases, see the *Cisco ACI Multi-Site Fundamentals Guide*.

The following is an example of how to create a new Schema.

POST:

```
POST  https://<host>:<port>/api/v1/schemas
```

Request:

```
{
  "displayName": "Schema 1",
  "templates": [
    {
      "name": "template1",
      "displayName": "Template 1",
      "tenantId": "592c7c482501002501061578",
      "anps": [
        {
```

```
"name": "anp1",
"displayName": "Anp 1",
"epgs": [
  {
    "name": "epg1",
    "displayName": "EPG 1",
    "contractRelationships": [
      {
        "contractRef": "/templates/template1/contracts/contract1",
        "relationshipType": "provider|consumer"
      }
    ],
    "subnets": [ {
          "ip": "192.168.132.1/24",
          "description": "Subnet for IT department",
          "scope" : "private"
          "shared": true
      }, {
          "ip": "192.168.133.1/24",
          "description": "Subnet for sales department",
          "scope" : "public"
          "shared": false
      }],
    "uSegEpg" : true,
    "uSegAttrs" : [ {
              "name" : "IP1",
              "displayName" : "...",
              "description" : "...",
              "type" : "ip",
              "fvSubnet" : true,
              "value" : "0.0.0.0"
           }, {
              "name" : "IP2",
              "displayName" : "...",
              "description" : "...",
              "type" : "ip",
              "fvSubnet" : false,
              "value" : "192.168.132.1"
           }, {
              "name" : "MAC",
              "displayName" : "...",
              "description" : "...",
              "type" : "mac",
              "value" : "aa:bb:cc:dd:ee:ff"
           }, {
              "name" : "VM Name",
              "displayName" : "...",
              "description" : "...",
              "type" : "vm-name",
              "operator" : "equals | contains | startsWith | endsWith",
              "value" : "..."
           }, {
              "name" : "VM Data Center",
              "displayName" : "...",
              "description" : "...",
              "type" : "rootContName",
              "operator" : "...",
              "value" : "..."
          }, {
              "name" : "VM Hypervisor Identifier",
              "displayName" : "...",
              "description" : "...",
              "type" : "hv",
              "operator" : "...",
```

```
                    "value" : "..."
                }, {
                    "name" : "VmOperatingSystem",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "guest-os",
                    "operator" : "...",
                    "value" : "..."
                }, {
                    "name" : "VM Tag",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "tag",
                    "operator" : "...",
                    "category" : "...",
                    "value" : "..."
                }, {
                    "name" : "Vm Identifier",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "vm",
                    "operator" : "...",
                    "value" : "..."
                }, {
                    "name" : "Vm Vmm Domain",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "domain",
                    "operator" : "...",
                    "value" : "..."
                }, {
                    "name" : "Vm Vnic Dn",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "vnic",
                    "operator" : "...",
                    "value" : "..."
                } ],
              "bdRef": "/templates/template1/bds/bd1"
            },
            {
              "name": "epg2",
              "displayName": "EPG 2",
              "contractRelationships": [
                {
                  "contractRef": "/templates/template1/contracts/contract2",
                  "relationshipType": "provider|consumer"
                }
              ],
              "bdRef": "templates/template1/bds/bd1"
            }
          ]
        },
        {
            "displayName": "AP 10",
            "name": "AP10",
            "anpRef": "/templates/Template1/anps/AP10",
            "epgs": [{
                "displayName": "EPG 10",
                "name": "EPG10",
                "epgRef": "/templates/Template1/anps/AP10/epgs/EPG10",
                "bdRef": "/templates/Template1/bds/BD10",
                "uSegEpg": false,
                "contractRelationships": [{
```

```
                "contractRef": "/templates/Template1/contracts/Contract34",
                "relationshipType": "provider"
            }],
            "subnets": [],
            "uSegAttrs": [],
            "intraEpg": "unenforced",
            "proxyArp": false
        }
],
"vrfs": [
  {
    "name": "vrf1",
    "displayName": "VRF 1"
  }
],
"bds": [
  {
    "name": "bd1",
    "displayName": "BD 1",
    "l3UnknownMulticastFlooding": "flood|opt-flood",
    "intersiteBumTrafficAllow": true,
    "multiDestinationFlooding": "bd-flood|drop|encap-flood",
    "l2UnknownUnicast": "flood|proxy",
    "l2Stretch": true,
    "subnets": [ ... ],
    "vrfRef": "templates/template1/vrfs/vrf1"
  }
],
"contracts": [
  {
    "name": "contract1",
    "displayName": "Contract 1",
    "filters": "80",
    "filterRelationships": [
      {
        "filterRef": "/templates/template1/filters/filter1",
        "directives": [
          "log|none"
        ]
      }
    ],
    "scope": "global|tenant|context|application-profile"
  },
  {
    "name": "contract2",
    "displayName": "Contract 2",
    "filters": "80",
    "scope": "global|tenant|context|application-profile"
  },
  {
      "displayName": "Contract 34",
      "name": "Contract34",
      "contractRef": "/templates/Template1/contracts/Contract34",
      "scope": "global",
      "filterRelationships": [],
      "filterType": "oneWay",
      "filterRelationshipsProviderToConsumer": [{
          "filterRef": "/templates/Template1/filters/Filter4",
          "directives": ["log"]
      }],
      "filterRelationshipsConsumerToProvider": [{
          "filterRef": "/templates/Template1/filters/Filter3",
          "directives": ["log"]
      }],
```

```
            "serviceGraphRelationship": {
                "serviceGraphRef": "/templates/Template1/serviceGraphs/SG1",
                "serviceNodesRelationship": [
                  {
                    "serviceNodeRef":
"/templates/Template1/serviceGraphs/SG1/serviceNodes/node1",
                    "consumerConnector": {
                        "connectorType": "general",
                        "bdRef": "/templates/Template1/bds/BD1"
                    },
                    "providerConnector": {
                        "connectorType": "general",
                        "bdRef": "/templates/Template1/bds/BD1"
                    }
                  }
                ]
            }
        }
      ],
      "filters": [
        {
          "name": "filter1",
          "displayName": "Filter 1",
          "description": "",
          "entries": [
            {
              "name": "filterEntry11",
              "displayName": "Filter Entry 11",
              "description": "",
              "etherType": "ip",
              "ipProtocol": "icmp"
            },
            {
              "name": "filterEntry12",
              "displayName": "Filter Entry 12",
              "description": "",
              "etherType": "ip",
              "ipProtocol": "udp",
              "matchOnlyFragments": false,
              "sourceFrom": "dns",
              "sourceTo": "http",
              "destinationFrom": "dns",
              "destinationTo": "80"
            }
          ]
        },
        {
          "name": "filter2",
          "displayName": "Filter 2",
          "description": "",
          "entries": [
            {
              "name": "filterEntry21",
              "displayName": "Filter Entry 21",
              "description": "",
              "etherType": "arp|fcoe|ip|mac_security|mpls_ucast|trill|unspecified",
              "arpFlag": "request|reply|unspecified",
              "ipProtocol":
"eigrp|egp|icmp|icmpv6|igmp|igp|l2tp|ospfigp|pim|tcp|udp|unspecified",
              "matchOnlyFragments": false,
              "stateful": false,
            "sourceFrom": "dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",

              "sourceTo": "dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
```

```
              "destinationFrom":
"dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
              "destinationTo":
"dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
              "tcpSessionRules": [
                "acknowledgement|established|finish|synchronize|reset|unspecified",
                "acknowledgement|established|finish|synchronize|reset|unspecified"
              ]
            }
          ]
        },{
            "displayName": "Filter 3",
            "name": "Filter3",
            "filterRef": "/templates/Template1/filters/Filter3",
            "entries": [{
                "etherType": "ip",
                "tcpSessionRules": ["unspecified"],
                "sourceFrom": "unspecified",
                "destinationTo": "unspecified",
                "arpFlag": "unspecified",
                "stateful": false,
                "displayName": "e1",
                "matchOnlyFragments": false,
                "sourceTo": "unspecified",
                "destinationFrom": "unspecified",
                "description": "",
                "ipProtocol": "unspecified",
                "name": "e1"
            }]
        }, {
            "displayName": "Filter 4",
            "name": "Filter4",
            "filterRef": "/templates/Template1/filters/Filter4",
            "entries": [{
                "etherType": "ip",
                "tcpSessionRules": ["unspecified"],
                "sourceFrom": "unspecified",
                "destinationTo": "unspecified",
                "arpFlag": "unspecified",
                "stateful": false,
                "displayName": "e4",
                "matchOnlyFragments": false,
                "sourceTo": "unspecified",
                "destinationFrom": "unspecified",
                "description": "",
                "ipProtocol": "unspecified",
                "name": "e4"
            }]
        }
      ],
      "externalEpgs": [
        {
          "name": "Test-extEPG",
          "displayName": "Test-extEPG",
          "subnets": [
            {
              "ip": "2.2.2.2/24"
            }
          ],
          "contractRelationships": [
            {
              "relationshipType": "provider",
              "contractRef": "/templates/Template1/contracts/contract1"
            }
```

```
          ],
          "externalEpgRef": "/templates/Template1/externalEpgs/Test-extEPG"
        }
      ],
      "serviceGraphs": [{
          "name": "SG1",
          "displayName": "SG1",
          "serviceGraphRef": "/templates/Template1/serviceGraphs/SG1",
          "description":"",
          "serviceNodes": [
            {
              "name": "node1",
            "serviceNodeRef": "/templates/Template1/serviceGraphs/SG1/serviceNodes/node1",

              "serviceNodeTypeId": "0000ffff0000000000000051",
              "index": 1
            },
            {
              "name": "node2",
            "serviceNodeRef": "/templates/Template1/serviceGraphs/SG1/serviceNodes/node2",

              "serviceNodeTypeId": "0000ffff0000000000000051",
              "index": 2
            }
          ]
        }
  ],
  "sites": [
    {
      "siteId": "58202f7066e6e10001c41236",
      "templateName": "template1",
      "bds": [
        {
          "bdRef": "/templates/template1/bds/bd1",
          "subnets": [ ... ]
        }
      ]
    },
    {
      "siteId": "69202f7066e6e10001c41247",
      "templateName": "template1",
      "anps": [
        {
          "anpRef": "/templates/template1/anps/anp1",
          "epgs": [
            {
              "epgRef": "/templates/template1/anps/anp1/epgs/epg1",
              "domainAssociations": [
                {
                  "dn": "uni/vmmp-VMware/dom-TestVMMDomain-DoNotDelete",
                  "domainType": "vmmDomain",
                  "deployImmediacy": "On Demand|Immediate",
                  "resolutionImmediacy": "On Demand|Immediate|Pre-provision",
                  "allowMicroSegmentation": true,
                  "allowPromiscuous": true,
                  "macChanges": false,
                  "allowForgedTransmits": true,
                  "delimiter": "delimiter",
                  "portEncapVlan": {
                    "vlanType": "vlan",
                    "vlan": 400
                  },
                  "deployImmediacy": "lazy",
                  "microSegVlan": {
```

```
                    "vlanType": "vlan",
                    "vlan": 450
                  },
                  "vlanEncapMode": "static"
                },
                {
                  "dn": "uni/phys-testPhysicalDomain-DoNotDelete",
                  "domainType": "physicalDomain",
                  "deployImmediacy": "On Demand|Immediate",
                  "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
                },
                {
                  "dn": "uni/l2dom-TestL2Domain-DoNotDelete",
                  "domainType": "l2ExtDomain",
                  "deployImmediacy": "On Demand|Immediate",
                  "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
                },
                {
                  "dn": "uni/l3dom-TestL3Domain-DoNotDelete",
                  "domainType": "l3ExtDomain",
                  "deployImmediacy": "On Demand|Immediate",
                  "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
                },
                {
                  "dn": "uni/fc-TestFibreChannelDomain",
                  "domainType": "fibreChannelDomain",
                  "deployImmediacy": "On Demand|Immediate",
                  "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
                }
              ],
              "staticPorts": [
                {
                    "type" : "port",
                    "path" : "topology/pod-1/paths-101/pathep[eth1/11]",
                    "portEncapVlan": 10,
                    "microSegVlan" : 8,
                    "deploymentImmediacy" : "immediate|lazy",
                    "mode" : "regular|native|untagged"
                }, {
                    "type" : "VPC",
                    "path" : "Topology/pod-1/protpaths-101-102/pathep[NFS-Storage]",
                    "portEncapVlan": 10,
                    "microSegVlan" : 8,
                    "deploymentImmediacy" : "Immediate",
                    "mode" : "Access"
                }, {
                    "type" : "DPC",
                    "path" : "...",
                    "portEncapVlan": 9,
                    "microSegVlan" : 8,
                    "deploymentImmediacy" : "OnDemand",
                    "mode" : "AccessUntagged"
                }
              ],
              "staticLeafs": [
                {
                    "path" : "topology/pod-1/node-201",
                    "portEncapVlan": 10
                }
              ]
            }
          ]
        }
    ],
```

```
        "bds": [
          {
            "bdRef": "/templates/template1/bds/bd1",
            "subnets": [ (see POST /templates) ]
          }
        ],
        "contracts": [
          {
            "contractRef": "/templates/template1/contracts/contract34",
            "serviceGraphRelationship": {
              "serviceGraphRef": "/templates/Template1/serviceGraphs/SG1",
              "serviceNodesRelationship": [
                {
                  "serviceNodeRef": "/templates/Template1/serviceGraphs/SG1/serviceNodes/node1",

                  "providerConnector": {
                    "clusterInterface": {
                      "dn": "/dn/CI1"
                    },
                    "redirectPolicy": {
                      "dn": "/dn/policy1"
                    }
                  },
                  "consumerConnector": {
                    "clusterInterface": {
                      "dn": "/dn/CI1"
                    },
                    "redirectPolicy": {
                      "dn": "/dn/policy1"
                    }
                  }
                }
              ]
            }
          }
        ],
        "externalEpgs": [
          {
            "externalEpgRef": "/templates/Template1/externalEpgs/Test-extEPG",
            "l3outDn": "uni/tn-Tenant/out-TestL3Out-site2",
            "subnets": [],
            "vrfRef": "templates/Template1/vrfs/vrf1",
            "faults": []
          }
        ],
        "serviceGraphs": [
        {
            "serviceGraphRef": "/templates/Template1/serviceGraphs/SG1",
            "serviceNodes": [
              {
                "serviceNodeRef": "/templates/Template1/serviceGraphs/SG1/serviceNodes/node1",

                "device": {
                  "dn": "/tn/Device1"
                }
              },
              {
              "serviceNodeRef": "/templates/Template1/serviceGraphs/SG1/serviceNodes/node2",

                "device": {
                  "dn": "/tn/Device2"
                }
              }
```

```
            ]
        }
      ]
    ]
}
```

Response:

```
HTTP/1.1 201
Location: /api/v1/schemas/583c7c482501002501061985
{
  "id": "583c7c482501002501061985",
  "displayName": "Schema 1",
  "templates": [
    {
      "name": "template1",
      "displayName": "Template 1",
      "tenantId": "592c7c48250100250106157 8",
      "anps": [
        {
          "name": "anp1",
          "displayName": "Anp 1",
          "anpRef": "/schemas/583c7c482501002501061985/templates/template1/anps/anp1",
          "epgs": [
            {
              "name": "epg1",
              "displayName": "EPG 1",
              "epgRef":
"/schemas/583c7c482501002501061985/templates/template1/anps/anp1/epgs/epg1",
              "contractRelationships": [
                {
                  "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
                  "relationshipType": "provider|consumer"
                }
              ],
              "subnets": [ {
                    "ip": "192.168.132.1/24",
                    "description": "Subnet for IT department",
                    "scope" : "private"
                    "shared": true
                }, {
                    "ip": "192.168.133.1/24",
                    "description": "Subnet for sales department",
                    "scope" : "public"
                    "shared": false
                }],
              "uSegEpg" : true,
              "uSegAttrs" : [ {
                        "name" : "IP1",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "ip",
                        "fvSubnet" : true,
                        "value" : "0.0.0.0"
                      }, {
                        "name" : "IP2",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "ip",
                        "fvSubnet" : false,
                        "value" : "192.168.132.1"
                      }, {
```

```
                "name" : "MAC",
                "displayName" : "...",
                "description" : "...",
                "type" : "mac",
                "value" : "aa:bb:cc:dd:ee:ff"
             }, {
                "name" : "VM Name",
                "displayName" : "...",
                "description" : "...",
                "type" : "vm-name",
                "operator" : "equals | contains | startsWith | endsWith",
                "value" : "..."
             }, {
                "name" : "VM Data Center",
                "displayName" : "...",
                "description" : "...",
                "type" : "rootContName",
                "operator" : "...",
                "value" : "..."
            }, {
                "name" : "VM Hypervisor Identifier",
                "displayName" : "...",
                "description" : "...",
                "type" : "hv",
                "operator" : "...",
                "value" : "..."
            }, {
                "name" : "VmOperatingSystem",
                "displayName" : "...",
                "description" : "...",
                "type" : "guest-os",
                "operator" : "...",
                "value" : "..."
            }, {
                "name" : "VM Tag",
                "displayName" : "...",
                "description" : "...",
                "type" : "tag",
                "operator" : "...",
                "category" : "...",
                "value" : "..."
            }, {
                "name" : "Vm Identifier",
                "displayName" : "...",
                "description" : "...",
                "type" : "vm",
                "operator" : "...",
                "value" : "..."
            }, {
                "name" : "Vm Vmm Domain",
                "displayName" : "...",
                "description" : "...",
                "type" : "domain",
                "operator" : "...",
                "value" : "..."
            }, {
                "name" : "Vm Vnic Dn",
                "displayName" : "...",
                "description" : "...",
                "type" : "vnic",
                "operator" : "...",
                "value" : "..."
            } ],
        "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1"
```

```
            },
            {
              "name": "epg2",
              "displayName": "EPG 2",
              "epgRef":
"/schema/583c7c482501002501061985/template/template1/anp/anp1/epg/epg2",
              "contractRelationships": [
                {
                  "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract2",
                  "relationshipType": "provider|consumer"
                }
              ],
              "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1"
            }
          ]
        }
      ],
      "vrfs": [
        {
          "name": "vrf1",
          "displayName": "VRF 1",
          "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1"
        }
      ],
      "bds": [
        {
          "name": "bd1",
          "displayName": "BD 1",
          "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
          "l3UnknownMulticastFlooding": "flood|opt-flood",
          "intersiteBumTrafficAllow": true,
          "multiDestinationFlooding": "bd-flood|drop|encap-flood",
          "l2UnknownUnicast": "flood|proxy",
          "l2Stretch": true,
          "subnets": [ ... ],
          "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1"
        }
      ],
      "contracts": [
        {
          "name": "contract1",
          "displayName": "Contract 1",
          "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
          "filters": "80",
          "filterRelationships": [
            {
              "filterRef":
"/schemas/583c7c482501002501061985/templates/template1/filters/filter1",
              "directives": [
                "log"
              ]
            }
          ],
          "scope": "global|tenant|context|application-profile"
        },
        {
          "name": "contract2",
          "displayName": "Contract 2",
          "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract2",
          "filters": "80",
          "filterRelationships": [],
```

```
            "scope": "global|tenant|context|anp"
          },
          {
              "displayName": "Contract 34",
              "name": "Contract34",
              "contractRef": "/templates/Template1/contracts/Contract34",
              "scope": "global",
              "filterRelationships": [],
              "filterType": "oneWay",
              "filterRelationshipsProviderToConsumer": [{
                  "filterRef": "/templates/Template1/filters/Filter4",
                  "directives": ["log"]
              }],
              "filterRelationshipsConsumerToProvider": [{
                  "filterRef": "/templates/Template1/filters/Filter3",
                  "directives": ["log"]
              }],
              "serviceGraphRelationship": {
                  "serviceGraphRef": "/templates/Template1/serviceGraphs/SG1",
                  "serviceNodesRelationship": [
                   {
                      "serviceNodeRef":
"/templates/Template1/serviceGraphs/SG1/serviceNodes/node1",
                      "consumerConnector": {
                          "connectorType": "general",
                          "bdRef": "/templates/Template1/bds/BD1"
                      },
                      "providerConnector": {
                          "connectorType": "general",
                          "bdRef": "/templates/Template1/bds/BD1"
                      }
                   }
                  ]
              }
          }
        ],
        "filters": [
          {
            "name": "filter1",
            "displayName": "Filter 1",
            "description": "",
            "filterRef":
"/schemas/583c7c482501002501061985/templates/template1/filters/filter1",
            "entries": [
              {
                "name": "filterEntry11",
                "displayName": "Filter Entry 11",
                "description": "",
                "etherType": "ip",
                "ipProtocol": "icmp"
              },
              {
                "name": "filterEntry12",
                "displayName": "Filter Entry 12",
                "description": "",
                "etherType": "ip",
                "ipProtocol": "udp",
                "matchOnlyFragments": false,
                "sourceFrom": "dns",
                "sourceTo": "http",
                "destinationFrom": "dns",
                "destinationTo": "http"
              }
            ]
```

```
        },
        {
          "name": "filter2",
          "displayName": "Filter 2",
          "description": "",
          "filterRef":
"/schemas/583c7c482501002501061985/templates/template1/filters/filter2",
          "entries": [...]
        }
      ],
      "externalEpgs": [
        {
          "name": "Test-extEPG",
          "displayName": "Test-extEPG",
          "subnets": [
            {
              "ip": "2.2.2.2/24"
            }
          ],
          "contractRelationships": [
            {
              "relationshipType": "provider",
              "contractRef":
"/schemas/583c7c482501002501061985/templates/Template1/contracts/contract1"
            }
          ],
          "externalEpgRef":
"/schemas/583c7c482501002501061985/templates/Template1/externalEpgs/Test-extEPG"
        }
      ]
    }
  ],
  "serviceGraphs": [
  {
        "serviceGraphRef": "/templates/Template1/serviceGraphs/SG1",
        "serviceNodes": [
        {
            "serviceNodeRef": "/templates/Template1/serviceGraphs/SG1/serviceNodes/node1",

              "device": {
                "dn": "/tn/Device1"
              }
        },
        {
            "serviceNodeRef": "/templates/Template1/serviceGraphs/SG1/serviceNodes/node2",

              "device": {
                "dn": "/tn/Device2"
              }
         }
        ]
    }
  ]
,
  "sites": [
    {
      "siteId": "58202f7066e6e10001c41236",
      "templateName": "template1",
      "bds": [
        {
          "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
          "subnets": [ ... ]
        }
      ]
```

```
    },
    {
      "siteId": "69202f7066e6e10001c41247",
      "templateName": "template1",
      "anps": [
        {
          "anpRef": "/schemas/583c7c482501002501061985/templates/template1/anps/anp1",
          "epgs": [
            {
              "epgRef":
"/schemas/583c7c482501002501061985/templates/template1/anps/anp1/epgs/epg1",
              "domainAssociations": [
                {
                  "dn": "uni/vmmp-VMware/dom-TestVMMDomain-DoNotDelete",
                  "domainType": "vmmDomain",
                  "deployImmediacy": "On Demand",
                  "resolutionImmediacy": "Immediate",
                  "allowMicroSegmentation": true,
                  "allowPromiscuous": true,
                  "macChanges": false,
                  "allowForgedTransmits": true,
                  "delimiter": "delimiter",
                  "microSegVlan": {
                    "vlanType": "vlan",
                    "vlan": 450
                  },
                  "portEncapVlan": {
                    "vlanType": "vlan",
                    "vlan": 400
                  },
                  "vlanEncapMode": "static"
                },
                {
                  "dn": "uni/phys-testPhysicalDomain-DoNotDelete",
                  "domainType": "physicalDomain",
                  "deployImmediacy": "On Demand",
                  "resolutionImmediacy": "Immediate"
                },
                {
                  "dn": "uni/l2dom-TestL2Domain-DoNotDelete",
                  "domainType": "l2ExtDomain",
                  "deployImmediacy": "On Demand",
                  "resolutionImmediacy": "Immediate"
                },
                {
                  "dn": "uni/l3dom-TestL3Domain-DoNotDelete",
                  "domainType": "l3ExtDomain",
                  "deployImmediacy": "On Demand",
                  "resolutionImmediacy": "Immediate"
                },
                {
                  "dn": "uni/fc-TestFibreChannelDomain",
                  "domainType": "fibreChannelDomain",
                  "deployImmediacy": "On Demand",
                  "resolutionImmediacy": "Immediate"
                }
              ],
             "staticPorts": [
                 {
                     "type" : "port",
                     "path" : "topology/pod-1/paths-101/pathep[eth1/11]",
                     "portEncapVlan": 10,
                     "microSegVlan" : 8,
                     "deploymentImmediacy" : "immediate|lazy",
```

```
                    "mode" : "regular|native|untagged"
                }, {
                    "type" : "VPC",
                    "path" : "Topology/pod-1/protpaths-101-102/pathep[NFS-Storage]",
                    "portEncapVlan": 10,
                    "microSegVlan" : 8,
                    "deploymentImmediacy" : "Immediate",
                    "mode" : "Access"
                }, {
                    "type" : "DPC",
                    "path" : "...",
                    "portEncapVlan": 9,
                    "microSegVlan" : 8,
                    "deploymentImmediacy" : "OnDemand",
                    "mode" : "AccessUntagged"
                }
              ],
              "staticLeafs": [
                {
                    "path" : "topology/pod-1/node-201",
                    "portEncapVlan": 10
                }

              ]
            }
          ]
        }
      ],
      "bds": [
        {
          "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
          "subnet": [ (see POST /templates) ]
        }
      ],
      "contracts": [
        {
          "contractRef": "/templates/template1/contracts/contract34",
          "serviceGraphRelationship": {
            "serviceGraphRef": "/templates/Template1/serviceGraphs/SG1",
            "serviceNodesRelationship": [
              {
              "serviceNodeRef": "/templates/Template1/serviceGraphs/SG1/serviceNodes/node1",

                "providerConnector": {
                  "clusterInterface": {
                    "dn": "/dn/CI1"
                  },
                  "redirectPolicy": {
                    "dn": "/dn/policy1"
                  }
                },
                "consumerConnector": {
                  "clusterInterface": {
                    "dn": "/dn/CI1"
                  },
                  "redirectPolicy": {
                    "dn": "/dn/policy1"
                  }
                }
              }
            ]
          }
        }
      ],
```

```
      "externalEpgs": [
        {
          "externalEpgRef":
"/schemas/59a4824425000043017c65ea/templates/Template1/externalEpgs/Test-extEPG",
          "l3outDn": "uni/tn-Tenant/out-TestL3Out-site2",
          "subnets": [],
          "vrfRef": "/schemas/59a4824425000043017c65ea/templates/Template1/vrfs/vrf1",
          "faults": []
        }
      ],
      "serviceGraphs": [
       {
          "serviceGraphRef": "/templates/Template1/serviceGraphs/SG1",
          "serviceNodes": [
           {
            "serviceNodeRef": "/templates/Template1/serviceGraphs/SG1/serviceNodes/node1",

              "device": {
                "dn": "/tn/Device1"
              }
            },
            {
            "serviceNodeRef": "/templates/Template1/serviceGraphs/SG1/serviceNodes/node2",

              "device": {
                "dn": "/tn/Device2"
             }
            }
          ]
        }
      ]
    }
  ]
}
```

Example for ACI Multi-Site Release 1.1(1)

The following is an example of how to create a new Schema .

POST:

```
POST  https://<host>:<port>/api/v1/schemas
```

Request:

```
{
  "displayName": "Schema 1",
  "templates": [
    {
      "name": "template1",
      "displayName": "Template 1",
      "tenantId": "592c7c482501002501061578",
      "anps": [
        {
          "name": "anp1",
          "displayName": "Anp 1",
          "epgs": [
            {
              "name": "epg1",
              "displayName": "EPG 1",
              "contractRelationships": [
                {
```

```
    "contractRef": "/templates/template1/contracts/contract1",
    "relationshipType": "provider|consumer"
  }
],
"subnets": [ {
      "ip": "192.168.132.1/24",
      "description": "Subnet for IT department",
      "scope" : "private"
      "shared": true
  }, {
      "ip": "192.168.133.1/24",
      "description": "Subnet for sales department",
      "scope" : "public"
      "shared": false
  }],
"uSegEpg" : true,
"uSegAttrs" : [ {
          "name" : "IP1",
          "displayName" : "...",
          "description" : "...",
          "type" : "ip",
          "fvSubnet" : true,
          "value" : "0.0.0.0"
       }, {
          "name" : "IP2",
          "displayName" : "...",
          "description" : "...",
          "type" : "ip",
          "fvSubnet" : false,
          "value" : "192.168.132.1"
       }, {
          "name" : "MAC",
          "displayName" : "...",
          "description" : "...",
          "type" : "mac",
          "value" : "aa:bb:cc:dd:ee:ff"
       }, {
          "name" : "VM Name",
          "displayName" : "...",
          "description" : "...",
          "type" : "vm-name",
          "operator" : "equals | contains | startsWith | endsWith",
          "value" : "..."
       }, {
          "name" : "VM Data Center",
          "displayName" : "...",
          "description" : "...",
          "type" : "rootContName",
          "operator" : "...",
          "value" : "..."
      }, {
          "name" : "VM Hypervisor Identifier",
          "displayName" : "...",
          "description" : "...",
          "type" : "hv",
          "operator" : "...",
          "value" : "..."
      }, {
          "name" : "VmOperatingSystem",
          "displayName" : "...",
          "description" : "...",
          "type" : "guest-os",
          "operator" : "...",
          "value" : "..."
```

```
                }, {
                    "name" : "VM Tag",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "tag",
                    "operator" : "...",
                    "category" : "...",
                    "value" : "..."
                }, {
                    "name" : "Vm Identifier",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "vm",
                    "operator" : "...",
                    "value" : "..."
                }, {
                    "name" : "Vm Vmm Domain",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "domain",
                    "operator" : "...",
                    "value" : "..."
                }, {
                    "name" : "Vm Vnic Dn",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "vnic",
                    "operator" : "...",
                    "value" : "..."
                } ],
            "bdRef": "/templates/template1/bds/bd1"
        },
        {
          "name": "epg2",
          "displayName": "EPG 2",
          "contractRelationships": [
            {
              "contractRef": "/templates/template1/contracts/contract2",
              "relationshipType": "provider|consumer"
            }
          ],
          "bdRef": "templates/template1/bds/bd1"
        }
      ]
    },
    {
        "displayName": "AP 10",
        "name": "AP10",
        "anpRef": "/templates/Template1/anps/AP10",
        "epgs": [{
            "displayName": "EPG 10",
            "name": "EPG10",
            "epgRef": "/templates/Template1/anps/AP10/epgs/EPG10",
            "bdRef": "/templates/Template1/bds/BD10",
            "uSegEpg": false,
            "contractRelationships": [{
                "contractRef": "/templates/Template1/contracts/Contract34",
                "relationshipType": "provider"
            }],
            "subnets": [],
            "uSegAttrs": [],
            "intraEpg": "unenforced",
            "proxyArp": false
        }
```

```
],
"vrfs": [
  {
    "name": "vrf1",
    "displayName": "VRF 1"
  }
],
"bds": [
  {
    "name": "bd1",
    "displayName": "BD 1",
    "l3UnknownMulticastFlooding": "flood|opt-flood",
    "intersiteBumTrafficAllow": true,
    "multiDestinationFlooding": "bd-flood|drop|encap-flood",
    "l2UnknownUnicast": "flood|proxy",
    "l2Stretch": true,
    "subnets": [ ... ],
    "vrfRef": "templates/template1/vrfs/vrf1"
  }
],
"contracts": [
  {
    "name": "contract1",
    "displayName": "Contract 1",
    "filters": "80",
    "filterRelationships": [
      {
        "filterRef": "/templates/template1/filters/filter1",
        "directives": [
          "log|none"
        ]
      }
    ],
    "scope": "global|tenant|context|application-profile"
  },
  {
    "name": "contract2",
    "displayName": "Contract 2",
    "filters": "80",
    "scope": "global|tenant|context|application-profile"
  },
  {
      "displayName": "Contract 34",
      "name": "Contract34",
      "contractRef": "/templates/Template1/contracts/Contract34",
      "scope": "global",
      "filterRelationships": [],
      "filterRelationshipsProviderToConsumer": [{
          "filterRef": "/templates/Template1/filters/Filter4",
          "directives": ["log"]
      }],
      "filterRelationshipsConsumerToProvider": [{
          "filterRef": "/templates/Template1/filters/Filter3",
          "directives": ["log"]
      }],
      "filterType": "oneWay"
  }
],
"filters": [
  {
    "name": "filter1",
    "displayName": "Filter 1",
    "description": "",
    "entries": [
```

```
                {
                  "name": "filterEntry11",
                  "displayName": "Filter Entry 11",
                  "description": "",
                  "etherType": "ip",
                  "ipProtocol": "icmp"
                },
                {
                  "name": "filterEntry12",
                  "displayName": "Filter Entry 12",
                  "description": "",
                  "etherType": "ip",
                  "ipProtocol": "udp",
                  "matchOnlyFragments": false,
                  "sourceFrom": "dns",
                  "sourceTo": "http",
                  "destinationFrom": "dns",
                  "destinationTo": "80"
                }
              ]
            },
            {
              "name": "filter2",
              "displayName": "Filter 2",
              "description": "",
              "entries": [
                {
                  "name": "filterEntry21",
                  "displayName": "Filter Entry 21",
                  "description": "",
                  "etherType": "arp|fcoe|ip|mac_security|mpls_ucast|trill|unspecified",
                  "arpFlag": "request|reply|unspecified",
                  "ipProtocol":
"eigrp|egp|icmp|icmpv6|igmp|igp|l2tp|ospfigp|pim|tcp|udp|unspecified",
                  "matchOnlyFragments": false,
                  "stateful": false,
                 "sourceFrom": "dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",

                  "sourceTo": "dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
                  "destinationFrom":
"dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
                  "destinationTo":
"dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
                  "tcpSessionRules": [
                    "acknowledgement|established|finish|synchronize|reset|unspecified",
                    "acknowledgement|established|finish|synchronize|reset|unspecified"
                  ]
                }
              ]
            },{
                "displayName": "Filter 3",
                "name": "Filter3",
                "filterRef": "/templates/Template1/filters/Filter3",
                "entries": [{
                    "etherType": "ip",
                    "tcpSessionRules": ["unspecified"],
                    "sourceFrom": "unspecified",
                    "destinationTo": "unspecified",
                    "arpFlag": "unspecified",
                    "stateful": false,
                    "displayName": "e1",
                    "matchOnlyFragments": false,
                    "sourceTo": "unspecified",
                    "destinationFrom": "unspecified",
```

```
                "description": "",
                "ipProtocol": "unspecified",
                "name": "e1"
            }]
        }, {
            "displayName": "Filter 4",
            "name": "Filter4",
            "filterRef": "/templates/Template1/filters/Filter4",
            "entries": [{
                "etherType": "ip",
                "tcpSessionRules": ["unspecified"],
                "sourceFrom": "unspecified",
                "destinationTo": "unspecified",
                "arpFlag": "unspecified",
                "stateful": false,
                "displayName": "e4",
                "matchOnlyFragments": false,
                "sourceTo": "unspecified",
                "destinationFrom": "unspecified",
                "description": "",
                "ipProtocol": "unspecified",
                "name": "e4"
            }]
        }
      ],
      "externalEpgs": [
        {
          "name": "Test-extEPG",
          "displayName": "Test-extEPG",
          "subnets": [
            {
              "ip": "2.2.2.2/24"
            }
          ],
          "contractRelationships": [
            {
              "relationshipType": "provider",
              "contractRef": "/templates/Template1/contracts/contract1"
            }
          ],
          "externalEpgRef": "/templates/Template1/externalEpgs/Test-extEPG"
        }
      ]
    }
  ],
  "sites": [
    {
      "siteId": "58202f7066e6e10001c41236",
      "templateName": "template1",
      "bds": [
        {
          "bdRef": "/templates/template1/bds/bd1",
          "subnets": [ ... ]
        }
      ]
    },
    {
      "siteId": "69202f7066e6e10001c41247",
      "templateName": "template1",
      "anps": [
        {
          "anpRef": "/templates/template1/anps/anp1",
          "epgs": [
            {
```

```
"epgRef": "/templates/template1/anps/anp1/epgs/epg1",
"domainAssociations": [
  {
    "dn": "uni/vmmp-VMware/dom-TestVMMDomain-DoNotDelete",
    "domainType": "vmmDomain",
    "deployImmediacy": "On Demand|Immediate",
    "resolutionImmediacy": "On Demand|Immediate|Pre-provision",
    "allowMicroSegmentation": true,
    "allowPromiscuous": true,
    "macChanges": false,
    "allowForgedTransmits": true,
    "delimiter": "delimiter",
    "portEncapVlan": {
      "vlanType": "vlan",
      "vlan": 400
    },
    "deployImmediacy": "lazy",
    "microSegVlan": {
      "vlanType": "vlan",
      "vlan": 450
    },
    "vlanEncapMode": "static"
  },
  {
    "dn": "uni/phys-testPhysicalDomain-DoNotDelete",
    "domainType": "physicalDomain",
    "deployImmediacy": "On Demand|Immediate",
    "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
  },
  {
    "dn": "uni/l2dom-TestL2Domain-DoNotDelete",
    "domainType": "l2ExtDomain",
    "deployImmediacy": "On Demand|Immediate",
    "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
  },
  {
    "dn": "uni/l3dom-TestL3Domain-DoNotDelete",
    "domainType": "l3ExtDomain",
    "deployImmediacy": "On Demand|Immediate",
    "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
  },
  {
    "dn": "uni/fc-TestFibreChannelDomain",
    "domainType": "fibreChannelDomain",
    "deployImmediacy": "On Demand|Immediate",
    "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
  }
],
"staticPorts": [
  {
      "type" : "port",
      "path" : "topology/pod-1/paths-101/pathep[eth1/11]",
      "portEncapVlan": 10,
      "microSegVlan" : 8,
      "deploymentImmediacy" : "immediate|lazy",
      "mode" : "regular|native|untagged"
  }, {
      "type" : "VPC",
      "path" : "Topology/pod-1/protpaths-101-102/pathep[NFS-Storage]",
      "portEncapVlan": 10,
      "microSegVlan" : 8,
      "deploymentImmediacy" : "Immediate",
      "mode" : "Access"
  }, {
```

```
                    "type" : "DPC",
                    "path" : "...",
                    "portEncapVlan": 9,
                    "microSegVlan" : 8,
                    "deploymentImmediacy" : "OnDemand",
                    "mode" : "AccessUntagged"
                  }
                ],
                "staticLeafs": [
                  {
                    "path" : "topology/pod-1/node-201",
                    "portEncapVlan": 10
                  }
                ]
              }
            ]
          }
        ],
        "bds": [
          {
            "bdRef": "/templates/template1/bds/bd1",
            "subnets": [ (see POST /templates) ]
          }
        ],
        "contracts": [
          {
            "contractRef": "/templates/template1/contracts/contract1",
            "serviceGraphTemplate": {
              "name": "SGT1"
            }
          }
        ],
        "externalEpgs": [
          {
            "externalEpgRef": "/templates/Template1/externalEpgs/Test-extEPG",
            "l3outDn": "uni/tn-Tenant/out-TestL3Out-site2",
            "subnets": [],
            "vrfRef": "templates/Template1/vrfs/vrf1",
            "faults": []
          }
        ]
      }
    ]
}
```

Response:

```
HTTP/1.1 201
Location: /api/v1/schemas/583c7c482501002501061985
{
  "id": "583c7c482501002501061985",
  "displayName": "Schema 1",
  "templates": [
    {
      "name": "template1",
      "displayName": "Template 1",
      "tenantId": "592c7c4825010025010615 78",
      "anps": [
        {
          "name": "anp1",
          "displayName": "Anp 1",
          "anpRef": "/schemas/583c7c482501002501061985/templates/template1/anps/anp1",
          "epgs": [
```

```
            {
              "name": "epg1",
              "displayName": "EPG 1",
              "epgRef":
"/schemas/583c7c4825010025010619 85/templates/template1/anps/anp1/epgs/epg1",
              "contractRelationships": [
                {
                  "contractRef":
"/schemas/583c7c4825010025010619 85/templates/template1/contracts/contract1",
                  "relationshipType": "provider|consumer"
                }
              ],
              "subnets": [ {
                    "ip": "192.168.132.1/24",
                    "description": "Subnet for IT department",
                    "scope" : "private"
                    "shared": true
                  }, {
                    "ip": "192.168.133.1/24",
                    "description": "Subnet for sales department",
                    "scope" : "public"
                    "shared": false
                  }],
              "uSegEpg" : true,
              "uSegAttrs" : [ {
                        "name" : "IP1",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "ip",
                        "fvSubnet" : true,
                        "value" : "0.0.0.0"
                      }, {
                        "name" : "IP2",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "ip",
                        "fvSubnet" : false,
                        "value" : "192.168.132.1"
                      }, {
                        "name" : "MAC",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "mac",
                        "value" : "aa:bb:cc:dd:ee:ff"
                      }, {
                        "name" : "VM Name",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "vm-name",
                        "operator" : "equals | contains | startsWith | endsWith",
                        "value" : "..."
                      }, {
                        "name" : "VM Data Center",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "rootContName",
                        "operator" : "...",
                        "value" : "..."
                      }, {
                        "name" : "VM Hypervisor Identifier",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "hv",
                        "operator" : "...",
```

```
                        "value" : "..."
                    }, {
                        "name" : "VmOperatingSystem",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "guest-os",
                        "operator" : "...",
                        "value" : "..."
                    }, {
                        "name" : "VM Tag",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "tag",
                        "operator" : "...",
                        "category" : "...",
                        "value" : "..."
                    }, {
                        "name" : "Vm Identifier",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "vm",
                        "operator" : "...",
                        "value" : "..."
                    }, {
                        "name" : "Vm Vmm Domain",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "domain",
                        "operator" : "...",
                        "value" : "..."
                    }, {
                        "name" : "Vm Vnic Dn",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "vnic",
                        "operator" : "...",
                        "value" : "..."
                    } ],
                  "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1"
                },
                {
                  "name": "epg2",
                  "displayName": "EPG 2",
                  "epgRef":
"/schema/583c7c482501002501061985/template/template1/anp/anp1/epg/epg2",
                  "contractRelationships": [
                    {
                      "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract2",
                      "relationshipType": "provider|consumer"
                    }
                  ],
                  "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1"
                }
              ]
            }
          ],
          "vrfs": [
            {
              "name": "vrf1",
              "displayName": "VRF 1",
              "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1"
            }
          ],
```

```
      "bds": [
        {
          "name": "bd1",
          "displayName": "BD 1",
          "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
          "l3UnknownMulticastFlooding": "flood|opt-flood",
          "intersiteBumTrafficAllow": true,
          "multiDestinationFlooding": "bd-flood|drop|encap-flood",
          "l2UnknownUnicast": "flood|proxy",
          "l2Stretch": true,
          "subnets": [ ... ],
          "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1"
        }
      ],
      "contracts": [
        {
          "name": "contract1",
          "displayName": "Contract 1",
          "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
          "filters": "80",
          "filterRelationships": [
            {
              "filterRef":
"/schemas/583c7c482501002501061985/templates/template1/filters/filter1",
              "directives": [
                "log"
              ]
            }
          ],
          "scope": "global|tenant|context|application-profile"
        },
        {
          "name": "contract2",
          "displayName": "Contract 2",
          "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract2",
          "filters": "80",
          "filterRelationships": [],
          "scope": "global|tenant|context|anp"
        }
      ],
      "filters": [
        {
          "name": "filter1",
          "displayName": "Filter 1",
          "description": "",
          "filterRef":
"/schemas/583c7c482501002501061985/templates/template1/filters/filter1",
          "entries": [
            {
              "name": "filterEntry11",
              "displayName": "Filter Entry 11",
              "description": "",
              "etherType": "ip",
              "ipProtocol": "icmp"
            },
            {
              "name": "filterEntry12",
              "displayName": "Filter Entry 12",
              "description": "",
              "etherType": "ip",
              "ipProtocol": "udp",
              "matchOnlyFragments": false,
```

```
                "sourceFrom": "dns",
                "sourceTo": "http",
                "destinationFrom": "dns",
                "destinationTo": "http"
              }
            ]
          },
          {
            "name": "filter2",
            "displayName": "Filter 2",
            "description": "",
            "filterRef":
"/schemas/583c7c482501002501061985/templates/template1/filters/filter2",
            "entries": [...]
          }
        ],
        "externalEpgs": [
          {
            "name": "Test-extEPG",
            "displayName": "Test-extEPG",
            "subnets": [
              {
                "ip": "2.2.2.2/24"
              }
            ],
            "contractRelationships": [
              {
                "relationshipType": "provider",
                "contractRef":
"/schemas/583c7c482501002501061985/templates/Template1/contracts/contract1"
              }
            ],
            "externalEpgRef":
"/schemas/583c7c482501002501061985/templates/Template1/externalEpgs/Test-extEPG"
          }
        ]
      }
    ],
    "sites": [
      {
        "siteId": "58202f7066e6e10001c41236",
        "templateName": "template1",
        "bds": [
          {
            "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
            "subnets": [ ... ]
          }
        ]
      },
      {
        "siteId": "69202f7066e6e10001c41247",
        "templateName": "template1",
        "anps": [
          {
            "anpRef": "/schemas/583c7c482501002501061985/templates/template1/anps/anp1",
            "epgs": [
              {
                "epgRef":
"/schemas/583c7c482501002501061985/templates/template1/anps/anp1/epgs/epg1",
                "domainAssociations": [
                  {
                    "dn": "uni/vmmp-VMware/dom-TestVMMDomain-DoNotDelete",
                    "domainType": "vmmDomain",
                    "deployImmediacy": "On Demand",
```

```
            "resolutionImmediacy": "Immediate",
            "allowMicroSegmentation": true,
            "allowPromiscuous": true,
            "macChanges": false,
            "allowForgedTransmits": true,
            "delimiter": "delimiter",
            "microSegVlan": {
              "vlanType": "vlan",
              "vlan": 450
            },
            "portEncapVlan": {
              "vlanType": "vlan",
              "vlan": 400
            },
            "vlanEncapMode": "static"
          },
          {
            "dn": "uni/phys-testPhysicalDomain-DoNotDelete",
            "domainType": "physicalDomain",
            "deployImmediacy": "On Demand",
            "resolutionImmediacy": "Immediate"
          },
          {
            "dn": "uni/l2dom-TestL2Domain-DoNotDelete",
            "domainType": "l2ExtDomain",
            "deployImmediacy": "On Demand",
            "resolutionImmediacy": "Immediate"
          },
          {
            "dn": "uni/l3dom-TestL3Domain-DoNotDelete",
            "domainType": "l3ExtDomain",
            "deployImmediacy": "On Demand",
            "resolutionImmediacy": "Immediate"
          },
          {
            "dn": "uni/fc-TestFibreChannelDomain",
            "domainType": "fibreChannelDomain",
            "deployImmediacy": "On Demand",
            "resolutionImmediacy": "Immediate"
          }
        ],
      "staticPorts": [
          {
              "type" : "port",
              "path" : "topology/pod-1/paths-101/pathep[eth1/11]",
              "portEncapVlan": 10,
              "microSegVlan" : 8,
              "deploymentImmediacy" : "immediate|lazy",
              "mode" : "regular|native|untagged"
          }, {
              "type" : "VPC",
              "path" : "Topology/pod-1/protpaths-101-102/pathep[NFS-Storage]",
              "portEncapVlan": 10,
              "microSegVlan" : 8,
              "deploymentImmediacy" : "Immediate",
              "mode" : "Access"
          }, {
              "type" : "DPC",
              "path" : "...",
              "portEncapVlan": 9,
              "microSegVlan" : 8,
              "deploymentImmediacy" : "OnDemand",
              "mode" : "AccessUntagged"
          }
```

```
                ],
                "staticLeafs": [
                  {
                      "path" : "topology/pod-1/node-201",
                      "portEncapVlan": 10
                  }

                ]
              }
            ]
          }
        ],
        "bds": [
          {
            "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
            "subnet": [ (see POST /templates) ]
          }
        ],
        "contracts": [
          {
            "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
            "serviceGraphTemplate": {
              "name": "SGT1"
            }
          }
        ],
        "externalEpgs": [
          {
            "externalEpgRef":
"/schemas/59a4824425000043017c65ea/templates/Template1/externalEpgs/Test-extEPG",
            "l3outDn": "uni/tn-Tenant/out-TestL3Out-site2",
            "subnets": [],
            "vrfRef": "/schemas/59a4824425000043017c65ea/templates/Template1/vrfs/vrf1",
            "faults": []
          }
        ]
      }
    ]
}
```

Example for ACI Multi-Site Release 1.0(2)

The following is an example of how to create a new Schema .

POST:

```
POST  https://<host>:<port>/api/v1/schemas
```

Request:

```
{
  "displayName": "Schema 1",
  "templates": [
    {
      "name": "template1",
      "displayName": "Template 1",
      "tenantId": "592c7c4825010025010615 78",
      "anps": [
        {
          "name": "anp1",
          "displayName": "Anp 1",
          "epgs": [
```

```
        {
          "name": "epg1",
          "displayName": "EPG 1",
          "contractRelationships": [
            {
              "contractRef": "/templates/template1/contracts/contract1",
              "relationshipType": "provider|consumer"
            }
          ],
"subnets": [ {
            "ip": "192.168.132.1/24",
            "description": "Subnet for IT department",
            "scope" : "private"
            "shared": true
         }, {
            "ip": "192.168.133.1/24",
            "description": "Subnet for sales department",
            "scope" : "public"
            "shared": false
        }],
"uSegEpg" : true,
"uSegAttrs" : [ {
    "name" : "IP1",
 "displayName" : "...",
    "description" : "...",
    "type" : "ip",
    "fvSubnet" : true,
    "value" : "0.0.0.0"
  }, {
     "name" : "IP2",
     "displayName" : "...",
    "description" : "...",
     "type" : "ip",
     "fvSubnet" : false,
       "value" : "192.168.132.1"
  }, {
     "name" : "MAC",
     "displayName" : "...",
    "description" : "...",
     "type" : "mac",
       "value" : "aa:bb:cc:dd:ee:ff"
  }, {
     "name" : "VM Name",
      "displayName" : "...",
    "description" : "...",
     "type" : "vm-name",
    "operator" : "equals | contains | startsWith | endsWith",
    "value" : "..."
  }, {
     "name" : "VM Data Center",
     "displayName" : "...",
    "description" : "...",
     "type" : "rootContName",
     "operator" : "...",
     "value" : "..."
  }, {
     "name" : "VM Hypervisor Identifier",
    "displayName" : "...",
    "description" : "...",
    "type" : "hv",
    "operator" : "...",
    "value" : "..."
  }, {
    "name" : "VmOperatingSystem",
```

```
    "displayName" : "...",
    "description" : "...",
    "type" : "guest-os",
    "operator" : "...",
    "value" : "..."
  }, {
    "name" : "VM Tag",
    "displayName" : "...",
    "description" : "...",
    "type" : "tag",
    "operator" : "...",
    "category" : "...",
    "value" : "..."
  }, {
    "name" : "Vm Identifier",
    "displayName" : "...",
    "description" : "...",
    "type" : "vm",
    "operator" : "...",
    "value" : "..."
  }, {
    "name" : "Vm Vmm Domain",
    "displayName" : "...",
    "description" : "...",
    "type" : "domain",
    "operator" : "...",
    "value" : "..."
  }, {
    "name" : "Vm Vnic Dn",
    "displayName" : "...",
    "description" : "...",
    "type" : "vnic",
    "operator" : "...",
    "value" : "..."
  } ],
          "bdRef": "/templates/template1/bds/bd1"
        },
        {
          "name": "epg2",
          "displayName": "EPG 2",
          "contractRelationships": [
            {
              "contractRef": "/templates/template1/contracts/contract2",
              "relationshipType": "provider|consumer"
            }
          ],
          "bdRef": "templates/template1/bds/bd1"
        }
      ]
    }
  ],
  "vrfs": [
    {
      "name": "vrf1",
      "displayName": "VRF 1"
    }
  ],
  "bds": [
    {
      "name": "bd1",
      "displayName": "BD 1",
      "l3UnknownMulticastFlooding": "flood|opt-flood",
      "intersiteBumTrafficAllow": true,
      "multiDestinationFlooding": "bd-flood|drop|encap-flood",
```

```
      "l2UnknownUnicast": "flood|proxy",
      "l2Stretch": true,
      "subnets": [ ... ],
      "vrfRef": "templates/template1/vrfs/vrf1"
    }
  ],
  "contracts": [
    {
      "name": "contract1",
      "displayName": "Contract 1",
      "filters": "80",
      "filterRelationships": [
        {
          "filterRef": "/templates/template1/filters/filter1",
          "directives": [
            "log|none"
          ]
        }
      ],
      "scope": "global|tenant|context|application-profile"
    },
    {
      "name": "contract2",
      "displayName": "Contract 2",
      "filters": "80",
      "scope": "global|tenant|context|application-profile"
    }
  ],
  "filters": [
    {
      "name": "filter1",
      "displayName": "Filter 1",
      "description": "",
      "entries": [
        {
          "name": "filterEntry11",
          "displayName": "Filter Entry 11",
          "description": "",
          "etherType": "ip",
          "ipProtocol": "icmp"
        },
        {
          "name": "filterEntry12",
          "displayName": "Filter Entry 12",
          "description": "",
          "etherType": "ip",
          "ipProtocol": "udp",
          "matchOnlyFragments": false,
          "sourceFrom": "dns",
          "sourceTo": "http",
          "destinationFrom": "dns",
          "destinationTo": "80"
        }
      ]
    },
    {
      "name": "filter2",
      "displayName": "Filter 2",
      "description": "",
      "entries": [
        {
          "name": "filterEntry21",
          "displayName": "Filter Entry 21",
          "description": "",
```

```
                "etherType": "arp|fcoe|ip|mac_security|mpls_ucast|trill|unspecified",
                "arpFlag": "request|reply|unspecified",
                "ipProtocol":
"eigrp|egp|icmp|icmpv6|igmp|igp|l2tp|ospfigp|pim|tcp|udp|unspecified",
                "matchOnlyFragments": false,
                "stateful": false,
              "sourceFrom": "dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",

                "sourceTo": "dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
                "destinationFrom":
"dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
                "destinationTo":
"dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
                "tcpSessionRules": [
                  "acknowledgement|established|finish|synchronize|reset|unspecified",
                  "acknowledgement|established|finish|synchronize|reset|unspecified"
                ]
              }
            ]
          }
        ],
        "externalEpgs": [
          {
            "name": "Test-extEPG",
            "displayName": "Test-extEPG",
            "subnets": [
              {
                "ip": "2.2.2.2/24"
              }
            ],
            "contractRelationships": [
              {
                "relationshipType": "provider",
                "contractRef": "/templates/Template1/contracts/contract1"
              }
            ],
            "externalEpgRef": "/templates/Template1/externalEpgs/Test-extEPG"
          }
        ]
      }
    ],
    "sites": [
      {
        "siteId": "58202f7066e6e10001c41236",
        "templateName": "template1",
        "bds": [
          {
            "bdRef": "/templates/template1/bds/bd1",
            "subnets": [ ... ]
          }
        ]
      },
      {
        "siteId": "69202f7066e6e10001c41247",
        "templateName": "template1",
        "anps": [
          {
            "anpRef": "/templates/template1/anps/anp1",
            "epgs": [
              {
                "epgRef": "/templates/template1/anps/anp1/epgs/epg1",
                "domainAssociations": [
                  {
                    "dn": "uni/vmmp-VMware/dom-TestVMMDomain-DoNotDelete",
```

```
                "domainType": "vmmDomain",
                "deployImmediacy": "On Demand|Immediate",
                "resolutionImmediacy": "On Demand|Immediate|Pre-provision",
                "allowMicroSegmentation": true,
                "allowPromiscuous": true,
                "macChanges": false,
                "allowForgedTransmits": true,
                "delimiter": "delimiter",
 "portEncapVlan": {
                  "vlanType": "vlan",
                  "vlan": 400
                },
                "deployImmediacy": "lazy",
                "microSegVlan": {
                  "vlanType": "vlan",
                  "vlan": 450
                },
 "vlanEncapMode": "static"
              },
              {
                "dn": "uni/phys-testPhysicalDomain-DoNotDelete",
                "domainType": "physicalDomain",
                "deployImmediacy": "On Demand|Immediate",
                "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
              },
              {
                "dn": "uni/l2dom-TestL2Domain-DoNotDelete",
                "domainType": "l2ExtDomain",
                "deployImmediacy": "On Demand|Immediate",
                "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
              },
              {
                "dn": "uni/l3dom-TestL3Domain-DoNotDelete",
                "domainType": "l3ExtDomain",
                "deployImmediacy": "On Demand|Immediate",
                "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
              },
              {
                "dn": "uni/fc-TestFibreChannelDomain",
                "domainType": "fibreChannelDomain",
                "deployImmediacy": "On Demand|Immediate",
                "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
              }
            ],
            "staticPorts": [
{
   "type" : "port",
      "path" : "topology/pod-1/paths-101/pathep[eth1/11]",
      "portEncapVlan": 10,
      "microSegVlan" : 8,
         "deploymentImmediacy" : "immediate|lazy",
      "mode" : "regular|native|untagged"
  }, {
      "type" : "VPC",
      "path" : "Topology/pod-1/protpaths-101-102/pathep[NFS-Storage]",
      "portEncapVlan": 10,
      "microSegVlan" : 8,
      "deploymentImmediacy" : "Immediate",
      "mode" : "Access"
  }, {
      "type" : "DPC",
      "path" : "...",
      "portEncapVlan": 9,
      "microSegVlan" : 8,
```

```
            "deploymentImmediacy" : "OnDemand",
            "mode" : "AccessUntagged"
          }
                ],
       "staticLeafs": [
      {
            "path" : "topology/pod-1/node-201",
            "portEncapVlan": 10
          }
       ]
              }
            ]
          }
        ],
        "bds": [
          {
            "bdRef": "/templates/template1/bds/bd1",
            "subnets": [ (see POST /templates) ]
          }
        ],
        "contracts": [
          {
            "contractRef": "/templates/template1/contracts/contract1",
            "serviceGraphTemplate": {
              "name": "SGT1"
            }
          }
        ],
        "externalEpgs": [
          {
            "externalEpgRef": "/templates/Template1/externalEpgs/Test-extEPG",
            "l3outDn": "uni/tn-Tenant/out-TestL3Out-site2",
            "subnets": [],
            "vrfRef": "templates/Template1/vrfs/vrf1",
            "faults": []
          }
        ]
      }
    ]
}
```

Response:

```
HTTP/1.1 201
Location: /api/v1/schemas/583c7c482501002501061985
{
  "id": "583c7c482501002501061985",
  "displayName": "Schema 1",
  "templates": [
    {
      "name": "template1",
      "displayName": "Template 1",
      "tenantId": "592c7c4825010025010615 78",
      "anps": [
        {
          "name": "anp1",
          "displayName": "Anp 1",
          "anpRef": "/schemas/583c7c482501002501061985/templates/template1/anps/anp1",
          "epgs": [
            {
              "name": "epg1",
              "displayName": "EPG 1",
              "epgRef":
"/schemas/583c7c482501002501061985/templates/template1/anps/anp1/epgs/epg1",
              "contractRelationships": [
```

```
                {
                  "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
                  "relationshipType": "provider|consumer"
                }
              ],
              "subnets": [ {
                "ip": "192.168.132.1/24",
                "description": "Subnet for IT department",
                "scope" : "private"
                "shared": true
              }, {
                "ip": "192.168.133.1/24",
                "description": "Subnet for sales department",
                "scope" : "public"
                "shared": false
            }],
              "uSegEpg" : true,
     "uSegAttrs" : [ {
         "name" : "IP1",
      "displayName" : "...",
         "description" : "...",
         "type" : "ip",
         "fvSubnet" : true,
         "value" : "0.0.0.0"
       }, {
          "name" : "IP2",
          "displayName" : "...",
         "description" : "...",
          "type" : "ip",
          "fvSubnet" : false,
            "value" : "192.168.132.1"
       }, {
          "name" : "MAC",
          "displayName" : "...",
         "description" : "...",
          "type" : "mac",
            "value" : "aa:bb:cc:dd:ee:ff"
       }, {
          "name" : "VM Name",
           "displayName" : "...",
         "description" : "...",
          "type" : "vm-name",
         "operator" : "equals | contains | startsWith | endsWith",
         "value" : "..."
       }, {
          "name" : "VM Data Center",
          "displayName" : "...",
         "description" : "...",
          "type" : "rootContName",
          "operator" : "...",
          "value" : "..."
       }, {
          "name" : "VM Hypervisor Identifier",
         "displayName" : "...",
         "description" : "...",
         "type" : "hv",
         "operator" : "...",
         "value" : "..."
       }, {
         "name" : "VmOperatingSystem",
         "displayName" : "...",
         "description" : "...",
         "type" : "guest-os",
```

```
            "operator" : "...",
            "value" : "..."
          }, {
            "name" : "VM Tag",
            "displayName" : "...",
            "description" : "...",
            "type" : "tag",
            "operator" : "...",
            "category" : "...",
            "value" : "..."
          }, {
            "name" : "Vm Identifier",
            "displayName" : "...",
            "description" : "...",
            "type" : "vm",
            "operator" : "...",
            "value" : "..."
          }, {
            "name" : "Vm Vmm Domain",
            "displayName" : "...",
            "description" : "...",
            "type" : "domain",
            "operator" : "...",
            "value" : "..."
          }, {
            "name" : "Vm Vnic Dn",
            "displayName" : "...",
            "description" : "...",
            "type" : "vnic",
            "operator" : "...",
            "value" : "..."
          } ],
                  "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1"
                },
                {
                  "name": "epg2",
                  "displayName": "EPG 2",
                  "epgRef":
"/schema/583c7c482501002501061985/template/template1/anp/anp1/epg/epg2",
                  "contractRelationships": [
                    {
                      "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract2",
                      "relationshipType": "provider|consumer"
                    }
                  ],
                  "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1"
                }
              ]
            }
          ],
          "vrfs": [
            {
              "name": "vrf1",
              "displayName": "VRF 1",
              "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1"
            }
          ],
          "bds": [
            {
              "name": "bd1",
              "displayName": "BD 1",
              "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
              "l3UnknownMulticastFlooding": "flood|opt-flood",
```

```
            "intersiteBumTrafficAllow": true,
            "multiDestinationFlooding": "bd-flood|drop|encap-flood",
            "l2UnknownUnicast": "flood|proxy",
            "l2Stretch": true,
            "subnets": [ ... ],
            "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1"
          }
        ],
        "contracts": [
          {
            "name": "contract1",
            "displayName": "Contract 1",
            "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
            "filters": "80",
            "filterRelationships": [
              {
                "filterRef":
"/schemas/583c7c482501002501061985/templates/template1/filters/filter1",
                "directives": [
                  "log"
                ]
              }
            ],
            "scope": "global|tenant|context|application-profile"
          },
          {
            "name": "contract2",
            "displayName": "Contract 2",
            "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract2",
            "filters": "80",
            "filterRelationships": [],
            "scope": "global|tenant|context|anp"
          }
        ],
        "filters": [
          {
            "name": "filter1",
            "displayName": "Filter 1",
            "description": "",
            "filterRef":
"/schemas/583c7c482501002501061985/templates/template1/filters/filter1",
            "entries": [
              {
                "name": "filterEntry11",
                "displayName": "Filter Entry 11",
                "description": "",
                "etherType": "ip",
                "ipProtocol": "icmp"
              },
              {
                "name": "filterEntry12",
                "displayName": "Filter Entry 12",
                "description": "",
                "etherType": "ip",
                "ipProtocol": "udp",
                "matchOnlyFragments": false,
                "sourceFrom": "dns",
                "sourceTo": "http",
                "destinationFrom": "dns",
                "destinationTo": "http"
              }
            ]
```

```
        },
        {
          "name": "filter2",
          "displayName": "Filter 2",
          "description": "",
          "filterRef":
"/schemas/583c7c482501002501061985/templates/template1/filters/filter2",
          "entries": [...]
        }
      ],
      "externalEpgs": [
        {
          "name": "Test-extEPG",
          "displayName": "Test-extEPG",
          "subnets": [
            {
              "ip": "2.2.2.2/24"
            }
          ],
          "contractRelationships": [
            {
              "relationshipType": "provider",
              "contractRef":
"/schemas/583c7c482501002501061985/templates/Template1/contracts/contract1"
            }
          ],
          "externalEpgRef":
"/schemas/583c7c482501002501061985/templates/Template1/externalEpgs/Test-extEPG"
        }
      ]
    }
  ],
  "sites": [
    {
      "siteId": "58202f7066e6e10001c41236",
      "templateName": "template1",
      "bds": [
        {
          "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
          "subnets": [ ... ]
        }
      ]
    },
    {
      "siteId": "69202f7066e6e10001c41247",
      "templateName": "template1",
      "anps": [
        {
          "anpRef": "/schemas/583c7c482501002501061985/templates/template1/anps/anp1",
          "epgs": [
            {
              "epgRef":
"/schemas/583c7c482501002501061985/templates/template1/anps/anp1/epgs/epg1",
              "domainAssociations": [
                {
                  "dn": "uni/vmmp-VMware/dom-TestVMMDomain-DoNotDelete",
                  "domainType": "vmmDomain",
                  "deployImmediacy": "On Demand",
                  "resolutionImmediacy": "Immediate",
                  "allowMicroSegmentation": true,
                  "allowPromiscuous": true,
                  "macChanges": false,
                  "allowForgedTransmits": true,
                  "delimiter": "delimiter",
```

```
                    "microSegVlan": {
                      "vlanType": "vlan",
                      "vlan": 450
                    },
                    "portEncapVlan": {
                      "vlanType": "vlan",
                      "vlan": 400
                    },
                    "vlanEncapMode": "static"
                  },
                  {
                    "dn": "uni/phys-testPhysicalDomain-DoNotDelete",
                    "domainType": "physicalDomain",
                    "deployImmediacy": "On Demand",
                    "resolutionImmediacy": "Immediate"
                  },
                  {
                    "dn": "uni/l2dom-TestL2Domain-DoNotDelete",
                    "domainType": "l2ExtDomain",
                    "deployImmediacy": "On Demand",
                    "resolutionImmediacy": "Immediate"
                  },
                  {
                    "dn": "uni/l3dom-TestL3Domain-DoNotDelete",
                    "domainType": "l3ExtDomain",
                    "deployImmediacy": "On Demand",
                    "resolutionImmediacy": "Immediate"
                  },
                  {
                    "dn": "uni/fc-TestFibreChannelDomain",
                    "domainType": "fibreChannelDomain",
                    "deployImmediacy": "On Demand",
                    "resolutionImmediacy": "Immediate"
                  }
                ],
"staticPorts": [
  {
     "type" : "port",
        "path" : "topology/pod-1/paths-101/pathep[eth1/11]",
        "portEncapVlan": 10,
        "microSegVlan" : 8,
           "deploymentImmediacy" : "immediate|lazy",
        "mode" : "regular|native|untagged"
    }, {
        "type" : "VPC",
        "path" : "Topology/pod-1/protpaths-101-102/pathep[NFS-Storage]",
        "portEncapVlan": 10,
        "microSegVlan" : 8,
        "deploymentImmediacy" : "Immediate",
        "mode" : "Access"
    }, {
        "type" : "DPC",
        "path" : "...",
        "portEncapVlan": 9,
        "microSegVlan" : 8,
        "deployment Immediacy" : "OnDemand",
        "mode" : "AccessUntagged"
    }
            ],
  "staticLeafs": [
 {
        "path" : "topology/pod-1/node-201",
        "portEncapVlan": 10
    }
```

```
      ]
             }
           ]
        }
      ],
      "bds": [
        {
          "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
          "subnet": [ (see POST /templates) ]
        }
      ],
      "contracts": [
        {
          "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
          "serviceGraphTemplate": {
            "name": "SGT1"
          }
        }
      ],
      "externalEpgs": [
        {
          "externalEpgRef":
"/schemas/59a4824425000043017c65ea/templates/Template1/externalEpgs/Test-extEPG",
          "l3outDn": "uni/tn-Tenant/out-TestL3Out-site2",
          "subnets": [],
          "vrfRef": "/schemas/59a4824425000043017c65ea/templates/Template1/vrfs/vrf1",
          "faults": []
        }
      ]
    }
  ]
}
```

Example for ACI Multi-Site Release 1.0(1)

The following is an example of how to create a new Schema .

POST:

```
POST  https://<host>:<port>/api/v1/schemas
```

Request:

```
{
  "displayName": "Schema 1",
  "templates": [
    {
      "name": "template1",
      "displayName": "Template 1",
      "tenantId": "592c7c4825010025010615 78",
      "anps": [
        {
          "name": "anp1",
          "displayName": "Anp 1",
          "epgs": [
            {
              "name": "epg1",
              "displayName": "EPG 1",
              "contractRelationships": [
                {
```

```
      "contractRef": "/templates/template1/contracts/contract1",
      "relationshipType": "provider|consumer"
    }
  ],
  "subnets": [ {
        "ip": "192.168.132.1/24",
        "description": "Subnet for IT department",
        "scope" : "private"
        "shared": true
    }, {
        "ip": "192.168.133.1/24",
        "description": "Subnet for sales department",
        "scope" : "public"
        "shared": false
    }],
  "uSegEpg" : true,
  "uSegAttrs" : [ {
            "name" : "IP1",
            "displayName" : "...",
            "description" : "...",
            "type" : "ip",
            "fvSubnet" : true,
            "value" : "0.0.0.0"
         }, {
            "name" : "IP2",
            "displayName" : "...",
            "description" : "...",
            "type" : "ip",
            "fvSubnet" : false,
            "value" : "192.168.132.1"
         }, {
            "name" : "MAC",
            "displayName" : "...",
            "description" : "...",
            "type" : "mac",
            "value" : "aa:bb:cc:dd:ee:ff"
         }, {
            "name" : "VM Name",
            "displayName" : "...",
            "description" : "...",
            "type" : "vm-name",
            "operator" : "equals | contains | startsWith | endsWith",
            "value" : "..."
         }, {
            "name" : "VM Data Center",
            "displayName" : "...",
            "description" : "...",
            "type" : "rootContName",
            "operator" : "...",
            "value" : "..."
        }, {
            "name" : "VM Hypervisor Identifier",
            "displayName" : "...",
            "description" : "...",
            "type" : "hv",
            "operator" : "...",
            "value" : "..."
        }, {
            "name" : "VmOperatingSystem",
            "displayName" : "...",
            "description" : "...",
            "type" : "guest-os",
            "operator" : "...",
            "value" : "..."
```

```
                }, {
                    "name" : "VM Tag",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "tag",
                    "operator" : "...",
                    "category" : "...",
                    "value" : "..."
                }, {
                    "name" : "Vm Identifier",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "vm",
                    "operator" : "...",
                    "value" : "..."
                }, {
                    "name" : "Vm Vmm Domain",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "domain",
                    "operator" : "...",
                    "value" : "..."
                }, {
                    "name" : "Vm Vnic Dn",
                    "displayName" : "...",
                    "description" : "...",
                    "type" : "vnic",
                    "operator" : "...",
                    "value" : "..."
                } ],
            "bdRef": "/templates/template1/bds/bd1"
          },
          {
            "name": "epg2",
            "displayName": "EPG 2",
            "contractRelationships": [
              {
                "contractRef": "/templates/template1/contracts/contract2",
                "relationshipType": "provider|consumer"
              }
            ],
            "bdRef": "templates/template1/bds/bd1"
          }
        ]
      }
    ],
    "vrfs": [
      {
        "name": "vrf1",
        "displayName": "VRF 1"
      }
    ],
    "bds": [
      {
        "name": "bd1",
        "displayName": "BD 1",
        "l3UnknownMulticastFlooding": "flood|opt-flood",
        "intersiteBumTrafficAllow": true,
        "multiDestinationFlooding": "bd-flood|drop|encap-flood",
        "l2UnknownUnicast": "flood|proxy",
        "l2Stretch": true,
        "subnets": [ ... ],
        "vrfRef": "templates/template1/vrfs/vrf1"
      }
```

```
        ],
        "contracts": [
          {
            "name": "contract1",
            "displayName": "Contract 1",
            "filters": "80",
            "filterRelationships": [
              {
                "filterRef": "/templates/template1/filters/filter1",
                "directives": [
                  "log|none"
                ]
              }
            ],
            "scope": "global|tenant|context|application-profile"
          },
          {
            "name": "contract2",
            "displayName": "Contract 2",
            "filters": "80",
            "scope": "global|tenant|context|application-profile"
          }
        ],
        "filters": [
          {
            "name": "filter1",
            "displayName": "Filter 1",
            "description": "",
            "entries": [
              {
                "name": "filterEntry11",
                "displayName": "Filter Entry 11",
                "description": "",
                "etherType": "ip",
                "ipProtocol": "icmp"
              },
              {
                "name": "filterEntry12",
                "displayName": "Filter Entry 12",
                "description": "",
                "etherType": "ip",
                "ipProtocol": "udp",
                "matchOnlyFragments": false,
                "sourceFrom": "dns",
                "sourceTo": "http",
                "destinationFrom": "dns",
                "destinationTo": "80"
              }
            ]
          },
          {
            "name": "filter2",
            "displayName": "Filter 2",
            "description": "",
            "entries": [
              {
                "name": "filterEntry21",
                "displayName": "Filter Entry 21",
                "description": "",
                "etherType": "arp|fcoe|ip|mac_security|mpls_ucast|trill|unspecified",
                "arpFlag": "request|reply|unspecified",
                "ipProtocol":
    "eigrp|egp|icmp|icmpv6|igmp|igp|l2tp|ospfigp|pim|tcp|udp|unspecified",
                "matchOnlyFragments": false,
```

```
              "stateful": false,
            "sourceFrom": "dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",

              "sourceTo": "dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
              "destinationFrom":
"dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
              "destinationTo":
"dns|ftp-data|http|https|pop3|rtsp|smtp|unspecified|[0..65535]",
              "tcpSessionRules": [
                "acknowledgement|established|finish|synchronize|reset|unspecified",
                "acknowledgement|established|finish|synchronize|reset|unspecified"
              ]
            }
          ]
        }
      ]
    }
  ],
  "sites": [
    {
      "siteId": "58202f7066e6e10001c41236",
      "templateName": "template1",
      "bds": [
        {
          "bdRef": "/templates/template1/bds/bd1",
          "subnets": [ ... ]
        }
      ]
    },
    {
      "siteId": "69202f7066e6e10001c41247",
      "templateName": "template1",
      "anps": [
        {
          "anpRef": "/templates/template1/anps/anp1",
          "epgs": [
            {
              "epgRef": "/templates/template1/anps/anp1/epgs/epg1",
              "domainAssociations": [
                {
                  "dn": "uni/vmmp-VMware/dom-TestVMMDomain-DoNotDelete",
                  "domainType": "vmmDomain",
                  "deployImmediacy": "On Demand|Immediate",
                  "resolutionImmediacy": "On Demand|Immediate|Pre-provision",
                  "allowMicroSegmentation": true,
                  "allowPromiscuous": true,
                  "macChanges": false,
                  "allowForgedTransmits": true,
                  "delimiter": "delimiter",
                  "portEncapVlan": {
                    "vlanType": "vlan",
                    "vlan": 400
                  },
                  "deployImmediacy": "lazy",
                  "microSegVlan": {
                    "vlanType": "vlan",
                    "vlan": 450
                  },
                  "vlanEncapMode": "static"
                },
                {
                  "dn": "uni/phys-testPhysicalDomain-DoNotDelete",
                  "domainType": "physicalDomain",
                  "deployImmediacy": "On Demand|Immediate",
```

```
            "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
          },
          {
            "dn": "uni/l2dom-TestL2Domain-DoNotDelete",
            "domainType": "l2ExtDomain",
            "deployImmediacy": "On Demand|Immediate",
            "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
          },
          {
            "dn": "uni/l3dom-TestL3Domain-DoNotDelete",
            "domainType": "l3ExtDomain",
            "deployImmediacy": "On Demand|Immediate",
            "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
          },
          {
            "dn": "uni/fc-TestFibreChannelDomain",
            "domainType": "fibreChannelDomain",
            "deployImmediacy": "On Demand|Immediate",
            "resolutionImmediacy": "On Demand|Immediate|Pre-provision"
          }
        ],
        "staticPorts": [
          {
              "type" : "port",
              "path" : "topology/pod-1/paths-101/pathep[eth1/11]",
              "portEncapVlan": 10,
              "microSegVlan" : 8,
              "deploymentImmediacy" : "immediate|lazy",
              "mode" : "regular|native|untagged"
          }, {
              "type" : "VPC",
              "path" : "Topology/pod-1/protpaths-101-102/pathep[NFS-Storage]",
              "portEncapVlan": 10,
              "microSegVlan" : 8,
              "deploymentImmediacy" : "Immediate",
              "mode" : "Access"
          }, {
              "type" : "DPC",
              "path" : "...",
              "portEncapVlan": 9,
              "microSegVlan" : 8,
              "deploymentImmediacy" : "OnDemand",
              "mode" : "AccessUntagged"
          }
        ],
        "staticLeafs": [
          {
              "path" : "topology/pod-1/node-201",
              "portEncapVlan": 10
          }
        ]
      }
    ]
  }
],
"bds": [
  {
    "bdRef": "/templates/template1/bds/bd1",
    "subnets": [ (see POST /templates) ]
  }
],
"contracts": [
  {
    "contractRef": "/templates/template1/contracts/contract1",
```

```
          "serviceGraphTemplate": {
            "name": "SGT1"
          }
        }
      ]
    }
  ]
}
```

Response:

```
HTTP/1.1 201
Location: /api/v1/schemas/583c7c482501002501061985
{
  "id": "583c7c482501002501061985",
  "displayName": "Schema 1",
  "templates": [
    {
      "name": "template1",
      "displayName": "Template 1",
      "tenantId": "592c7c48250100250106157 8",
      "anps": [
        {
          "name": "anp1",
          "displayName": "Anp 1",
          "anpRef": "/schemas/583c7c482501002501061985/templates/template1/anps/anp1",
          "epgs": [
            {
              "name": "epg1",
              "displayName": "EPG 1",
              "epgRef":
"/schemas/583c7c482501002501061985/templates/template1/anps/anp1/epgs/epg1",
              "contractRelationships": [
                {
                  "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
                  "relationshipType": "provider|consumer"
                }
              ],
              "subnets": [ {
                    "ip": "192.168.132.1/24",
                    "description": "Subnet for IT department",
                    "scope" : "private"
                    "shared": true
                }, {
                    "ip": "192.168.133.1/24",
                    "description": "Subnet for sales department",
                    "scope" : "public"
                    "shared": false
                }],
              "uSegEpg" : true,
              "uSegAttrs" : [ {
                        "name" : "IP1",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "ip",
                        "fvSubnet" : true,
                        "value" : "0.0.0.0"
                      }, {
                        "name" : "IP2",
                        "displayName" : "...",
                        "description" : "...",
                        "type" : "ip",
                        "fvSubnet" : false,
```

```
            "value" : "192.168.132.1"
        }, {
            "name" : "MAC",
            "displayName" : "...",
            "description" : "...",
            "type" : "mac",
            "value" : "aa:bb:cc:dd:ee:ff"
        }, {
            "name" : "VM Name",
            "displayName" : "...",
            "description" : "...",
            "type" : "vm-name",
            "operator" : "equals | contains | startsWith | endsWith",
            "value" : "..."
        }, {
            "name" : "VM Data Center",
            "displayName" : "...",
            "description" : "...",
            "type" : "rootContName",
            "operator" : "...",
            "value" : "..."
        }, {
            "name" : "VM Hypervisor Identifier",
            "displayName" : "...",
            "description" : "...",
            "type" : "hv",
            "operator" : "...",
            "value" : "..."
        }, {
            "name" : "VmOperatingSystem",
            "displayName" : "...",
            "description" : "...",
            "type" : "guest-os",
            "operator" : "...",
            "value" : "..."
        }, {
            "name" : "VM Tag",
            "displayName" : "...",
            "description" : "...",
            "type" : "tag",
            "operator" : "...",
            "category" : "...",
            "value" : "..."
        }, {
            "name" : "Vm Identifier",
            "displayName" : "...",
            "description" : "...",
            "type" : "vm",
            "operator" : "...",
            "value" : "..."
        }, {
            "name" : "Vm Vmm Domain",
            "displayName" : "...",
            "description" : "...",
            "type" : "domain",
            "operator" : "...",
            "value" : "..."
        }, {
            "name" : "Vm Vnic Dn",
            "displayName" : "...",
            "description" : "...",
            "type" : "vnic",
            "operator" : "...",
            "value" : "..."
```

```
                    } ],
                  "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1"
                },
                {
                  "name": "epg2",
                  "displayName": "EPG 2",
                  "epgRef":
"/schema/583c7c482501002501061985/template/template1/anp/anp1/epg/epg2",
                  "contractRelationships": [
                    {
                      "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract2",
                      "relationshipType": "provider|consumer"
                    }
                  ],
                  "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1"
                }
              ]
            }
          ],
          "vrfs": [
            {
              "name": "vrf1",
              "displayName": "VRF 1",
              "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1"
            }
          ],
          "bds": [
            {
              "name": "bd1",
              "displayName": "BD 1",
              "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
              "l3UnknownMulticastFlooding": "flood|opt-flood",
              "intersiteBumTrafficAllow": true,
              "multiDestinationFlooding": "bd-flood|drop|encap-flood",
              "l2UnknownUnicast": "flood|proxy",
              "l2Stretch": true,
              "subnets": [ ... ],
              "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1"
            }
          ],
          "contracts": [
            {
              "name": "contract1",
              "displayName": "Contract 1",
              "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
              "filters": "80",
              "filterRelationships": [
                {
                  "filterRef":
"/schemas/583c7c482501002501061985/templates/template1/filters/filter1",
                  "directives": [
                    "log"
                  ]
                }
              ],
              "scope": "global|tenant|context|application-profile"
            },
            {
              "name": "contract2",
              "displayName": "Contract 2",
              "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract2",
```

```
            "filters": "80",
            "filterRelationships": [],
            "scope": "global|tenant|context|anp"
          }
        ],

        "filters": [
          {
            "name": "filter1",
            "displayName": "Filter 1",
            "description": "",
            "filterRef":
"/schemas/583c7c482501002501061985/templates/template1/filters/filter1",
            "entries": [
              {
                "name": "filterEntry11",
                "displayName": "Filter Entry 11",
                "description": "",
                "etherType": "ip",
                "ipProtocol": "icmp"
              },
              {
                "name": "filterEntry12",
                "displayName": "Filter Entry 12",
                "description": "",
                "etherType": "ip",
                "ipProtocol": "udp",
                "matchOnlyFragments": false,
                "sourceFrom": "dns",
                "sourceTo": "http",
                "destinationFrom": "dns",
                "destinationTo": "http"
              }
            ]
          },
          {
            "name": "filter2",
            "displayName": "Filter 2",
            "description": "",
            "filterRef":
"/schemas/583c7c482501002501061985/templates/template1/filters/filter2",
            "entries": [...]
          }
        ]
      }
    ],
    "sites": [
      {
        "siteId": "58202f7066e6e10001c41236",
        "templateName": "template1",
        "bds": [
          {
            "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
            "subnets": [ ... ]
          }
        ]
      },
      {
        "siteId": "69202f7066e6e10001c41247",
        "templateName": "template1",
        "anps": [
          {
            "anpRef": "/schemas/583c7c482501002501061985/templates/template1/anps/anp1",
            "epgs": [
```

```
                {
                  "epgRef":
"/schemas/583c7c482501002501061985/templates/template1/anps/anp1/epgs/epg1",
                  "domainAssociations": [
                    {
                      "dn": "uni/vmmp-VMware/dom-TestVMMDomain-DoNotDelete",
                      "domainType": "vmmDomain",
                      "deployImmediacy": "On Demand",
                      "resolutionImmediacy": "Immediate",
                      "allowMicroSegmentation": true,
                      "allowPromiscuous": true,
                      "macChanges": false,
                      "allowForgedTransmits": true,
                      "delimiter": "delimiter",
                      "microSegVlan": {
                        "vlanType": "vlan",
                        "vlan": 450
                      },
                      "portEncapVlan": {
                        "vlanType": "vlan",
                        "vlan": 400
                      },
                      "vlanEncapMode": "static"
                    },
                    {
                      "dn": "uni/phys-testPhysicalDomain-DoNotDelete",
                      "domainType": "physicalDomain",
                      "deployImmediacy": "On Demand",
                      "resolutionImmediacy": "Immediate"
                    },
                    {
                      "dn": "uni/l2dom-TestL2Domain-DoNotDelete",
                      "domainType": "l2ExtDomain",
                      "deployImmediacy": "On Demand",
                      "resolutionImmediacy": "Immediate"
                    },
                    {
                      "dn": "uni/l3dom-TestL3Domain-DoNotDelete",
                      "domainType": "l3ExtDomain",
                      "deployImmediacy": "On Demand",
                      "resolutionImmediacy": "Immediate"
                    },
                    {
                      "dn": "uni/fc-TestFibreChannelDomain",
                      "domainType": "fibreChannelDomain",
                      "deployImmediacy": "On Demand",
                      "resolutionImmediacy": "Immediate"
                    }
                  ],
                 "staticPorts": [
                    {
                        "type" : "port",
                        "path" : "topology/pod-1/paths-101/pathep[eth1/11]",
                        "portEncapVlan": 10,
                        "microSegVlan" : 8,
                        "deploymentImmediacy" : "immediate|lazy",
                        "mode" : "regular|native|untagged"
                    }, {
                        "type" : "VPC",
                        "path" : "Topology/pod-1/protpaths-101-102/pathep[NFS-Storage]",
                        "portEncapVlan": 10,
                        "microSegVlan" : 8,
                        "deploymentImmediacy" : "Immediate",
                        "mode" : "Access"
```

```
                    }, {
                        "type" : "DPC",
                        "path" : "...",
                        "portEncapVlan": 9,
                        "microSegVlan" : 8,
                        "deploymentImmediacy" : "OnDemand",
                        "mode" : "AccessUntagged"
                    }
                  ],
                  "staticLeafs": [
                    {
                        "path" : "topology/pod-1/node-201",
                        "portEncapVlan": 10
                    }

                  ]
                }
              ]
            }
          ],
          "bds": [
            {
              "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
              "subnet": [ (see POST /templates) ]
            }
          ],
          "contracts": [
            {
              "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
              "serviceGraphTemplate": {
                "name": "SGT1"
              }
            }
          ]
        }
      ]
    }
```

Read a Schema

Example for ACI Multi-Site Release 1.0(2)

The following is an example for reading a Schema.

GET

```
GET https://<host>:<port>/api/v1/schemas/583c7c482501002501061985
```

REQUEST

```
GET
Request Body:
  <empty>
```

RESPONSE

```
HTTP/1.1 200
```

```
{
  "associations": [
    {
       "templateName": "template1",
  "bdAssociations" : [
   {
    "bdRef": "/schemas/583c7c482501002501061985/templates/template1/bds/bd1",
    "epgIdentities": [
     {
      "epgRef": "/schemas/583c7c482501002501061986/templates/template1/epgs/epg1",
      "schemaName": "Schema 1",
      "templateName": "Template 1"
      "displayName": "EPG1"
     }, ....
    ]
   }, ...
     ],
  "vrfAssociations" : [
   {
    "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1",
    "bdIdentities": [
     {
      "bdRef": "/schemas/583c7c482501002501061986/templates/template1/bds/bd1",
      "schemaName": "Schema 1",
      "templateName": "Template 1"
      "displayName": "BD 1"
     }, ....
    ],
                "externalEpgIdentities": [
                   {
                     "externalEpgRef":
"/schemas/583c7c482501002501061986/templates/template1/externalEpgs/Test-extEPG",
                     "schemaName": "schema1",
                     "templateName": "template1",
                     "displayName": "Test-extEPG"
                   }, ....
                ]
   }, ...
     ],
  "filterAssociations" : [
   {
    "filterRef": "/schemas/583c7c482501002501061985/templates/template1/filters/filter1",
    "contractIdentities": [
     {
      "contractRef":
"/schemas/583c7c482501002501061986/templates/template1/contracts/contract1",
      "schemaName": "Schema 1",
      "templateName": "Template 1"
      "displayName": "Contract 1"
     }, ....
    ]
   }, ...
     ],
  "contractAssociations" : [
   {
    "contractRef":
"/schemas/583c7c482501002501061985/templates/template1/contracts/contract1",
    "epgIdentities": [
     {
      "epgRef": "/schemas/583c7c482501002501061986/templates/template1/epgs/epg1",
      "schemaName": "Schema 1",
      "templateName": "Template 1"
      "displayName": "EPG1"
     }, ....
```

```
    ],
                "externalEpgIdentities": [
                  {
                    "externalEpgRef":
"/schemas/583c7c482501002501061986/templates/template1/externalEpgs/Test-extEPG",
                    "schemaName": "Demo-extEPG",
                    "templateName": "Provider",
                    "displayName": "Test-extEPG"
                  }, ....
                ]
   }, ...
     ],

 }, ....
  ]
}
```

Example for ACI Multi-Site Release 1.0(1)

The following is an example for reading a Schema.

GET

```
GET https://<host>:<port>/api/v1/schemas/583c7c482501002501061985
```

REQUEST

```
GET
Request Body:
  <empty>
```

RESPONSE

```
HTTP/1.1 200
{
  "id": "583c7c482501002501061985",
  "displayName": "Schema 1",
  ...
  << additional attributes >>
}
```

Schema Health Summary

The following is an example of a Schema health summary.

Note This API call returns a summary of the health of all Schemas to which the currently logged in user has access. The summary includes aggregated health of all policies in a Schema that has been pushed to a site, on a per template/site basis, as well as template/tenant identity information. This API is used for the Policy Heatmap on the dashboard page in the GUI .

Only Schemas that have been deployed will have health information.

GET

```
GET /api/v1/schemas/health-summary
```

REQUEST

```
  <empty>
```

RESPONSE


```
{
  "schemas": [
    {
      "id": "5969609c400000570068d69d",
      "schemaId": "5965bb96340000d800908add",
      "displayName": "Schema-Provider-AD",
      "templates": [
        {
          "name": "Template1",
          "displayName": "AD-Template-500"
        }
      ],
      "healthFaults": [
        {
          "siteId": "595e9796c00000c00000d632",
          "templateName": "Template1"
        }
      ]
    },
    {
      "id": "596e94253e00005f01347df0",
      "schemaId": "596e93c0860000e802274ac6",
      "displayName": "Common Schema",
      "templates": [
        {
          "name": "Simulator1CommonTenant",
          "displayName": "Simulator 1 Common Tenant"
        },
        {
          "name": "ifav87-site3",
          "displayName": "ifav87-site3"
        }
      ],
      "healthFaults": [
        {
          "siteId": "596d00644400008d00ca4ed0",
          "templateName": "Simulator1CommonTenant",
          "health": {
            "score": 100
          }
        },
        {
          "siteId": "596ff6e61f00007a002d1880",
          "templateName": "ifav87-site3",
          "health": {
            "score": 100
          }
        }
      ]
    },
    {
```

```
      "id": "5971314e2100007800076f36",
      "schemaId": "597131158500002301 44a54f",
      "displayName": "T2 Schema",
      "templates": [
        {
          "name": "Template1",
          "displayName": "Template 1"
        }
      ],
      "healthFaults": [
        {
          "siteId": "596d00644400008d00ca4ed0",
          "templateName": "Template1"
        }
      ]
    }
  ]
}
```

Schema Health Sync

The following is an example of how to sync health and fault information for all tenants in a schema from all the associated sites

GET

```
GET /api/v1/health/schemas/:schemaId/sync
```

REQUEST

```
  <empty>
```

RESPONSE

```
{
    "id": "5977ba963f0000de04760560",
    "schemaId": "5977ba913500006403c31911",
    "healthFaults": [
        {
            "siteId": "59726e7a490000ea095b4cc8",
            "templateName": "PespiTenant",
            "health": {
                "score": 100
            },
            "anps": [
                {
                    "anpRef": {
                        "schemaId": "5977ba913500006403c31911",
                        "templateName": "PespiTenant",
                        "anpName": "Pepsi-App"
                    },
                    "health": {
                        "score": 100
                    },
                    "faults": [],
                    "epgs": [
                        {
```

```
                            "epgRef": {
                                "schemaId": "5977ba913500006403c31911",
                                "templateName": "PespiTenant",
                                "anpName": "Pepsi-App",
                                "epgName": "App-EPG"
                            },
                            "health": {
                                "score": 100
                            },
                            "faults": []
                        }
                    ]
                }
            ],
            "vrfs": [
                {
                    "vrfRef": {
                        "schemaId": "5977ba913500006403c31911",
                        "templateName": "PespiTenant",
                        "vrfName": "TestVrf"
                    },
                    "health": {
                        "score": 100
                    },
                    "faults": []
                }
            ],
            "bds": [
                {
                    "bdRef": {
                        "schemaId": "5977ba913500006403c31911",
                        "templateName": "PespiTenant",
                        "bdName": "TestBD"
                    },
                    "health": {
                        "score": 100
                    },
                    "faults": []
                }
            ],
            "contracts": [
                {
                    "contractRef": {
                        "schemaId": "5977ba913500006403c31911",
                        "templateName": "PespiTenant",
                        "contractName": "TestContract"
                    },
                    "faults": []
                },
                {
                    "contractRef": {
                        "schemaId": "5977ba913500006403c31911",
                        "templateName": "PespiTenant",
                        "contractName": "TestContract2"
                    },
                    "faults": [
                        {
                            "type": "minor",
                            "message": "Configuration failed for Tenant Pepsi Contract
TestContract2 due to Filter Not Present",
                            "code": "F1228"
                        }
                    ]
                }
```

```
            ]
        }
    ]
}
```

Delete Schema Health Information

The following is an example of how to delete health and fault information for a schema.

GET

```
GET /api/v1/schemas/health-summary
```

REQUEST

```
  <empty>
```

RESPONSE

```
  <empty>
```

Retrieve Policy States

Example for ACI Multi-Site Release 1.2(1)

The following is an example that shows how to list policy states.

GET

```
GET  https://<host>:<port>/api/v1/:id/policy-states
```

REQUEST

```
  <empty>
```

RESPONSE

```
HTTP/1.1 200
{
  "policyStates": [
    {
      "siteId": "59631b3942000042001f1790",
      "templateName": "Template1",
      "anps": [...],
      "vrfs": [
        {
          "vrfRef": "/schemas/5963c0ff240000e900b34c55/templates/Template1/vrfs/VRF1",
          "apicState": "deleted"
        },
        {
          "vrfRef": "/schemas/5963c0ff240000e900b34c55/templates/Template1/vrfs/VRF2",
```

```
          "state": "created"
        }
      ],
      "bds": [
        {
          "bdRef": "/schemas/5963c0ff240000e900b34c55/templates/Template1/bds/BD10",
          "state": "modified"
        },
        {
          "bdRef": "/schemas/5963c0ff240000e900b34c55/templates/Template1/bds/BD11",
          "state": "deleted"
        },
        {
          "bdRef": "/schemas/5963c0ff240000e900b34c55/templates/Template1/bds/BD12",
          "state": "created"
        }
      ],
      "contracts": [...],
      "filters": [...],
      "externalEpgs": [
        {
          "externalEpgRef":
"/schemas/5963c0ff240000e900b34c55/templates/template1/externalEpgs/Test-extEPG",
          "state": "created"
        }
      ],
      "serviceGraphs": [{
            "serviceGraphRef":
"/schemas/5963c0ff240000e900b34c55/templates/Template1/serviceGraphs/SG",
          "state": "created"
        }]
    }
  ]
}
```

Example for ACI Multi-Site Release 1.0(2)

The following is an example that shows how to list policy states.

GET

```
GET  https://<host>:<port>/api/v1/:id/policy-states
```

REQUEST

```
  <empty>
```

RESPONSE

```
HTTP/1.1 200
{
  "policyStates": [
    {
      "siteId": "59631b3942000042001f1790",
      "templateName": "Template1",
      "anps": [...],
      "vrfs": [
        {
          "vrfRef": "/schemas/5963c0ff240000e900b34c55/templates/Template1/vrfs/VRF1",
          "apicState": "deleted"
```

```
        },
        {
          "vrfRef": "/schemas/5963c0ff240000e900b34c55/templates/Template1/vrfs/VRF2",
          "state": "created"
        }
      ],
      "bds": [
        {
          "bdRef": "/schemas/5963c0ff240000e900b34c55/templates/Template1/bds/BD10",
          "state": "modified"
        },
        {
          "bdRef": "/schemas/5963c0ff240000e900b34c55/templates/Template1/bds/BD11",
          "state": "deleted"
        },
        {
          "bdRef": "/schemas/5963c0ff240000e900b34c55/templates/Template1/bds/BD12",
          "state": "created"
        }
      ],
      "contracts": [...],
      "filters": [...],
      "externalEpgs": [
        {
          "externalEpgRef":
"/schemas/5963c0ff240000e900b34c55/templates/template1/externalEpgs/Test-extEPG",
          "state": "created"
        }
      ]
    }
  ]
}
```

Retrieve References

Example for ACI Multi-Site Release 1.2(1)

The following is an example that shows how to retrieve references.

GET

```
GET  https://<host>:<port>/api/v1/:id/references
```

REQUEST

```
  <empty>
```

RESPONSE

```
HTTP/1.1 200
{
  "references": {
    "vrfs": [
      {
        "identity": {
          "vrfRef": "/schemas/5963c0ff240000e900b34c55/templates/Template1/vrfs/VRF11",
          "schemaName": "Schema 11",
          "templateName": "Template 1",
```

```
          "displayName": "VRF 11"
        },
        "deployedOn": [
          "59631b3942000042001f1790"
        ]
      }
    ],
    "serviceGraphs": [{
            "identity": {
                "serviceGraphRef":
"/schemas/5963c0ff240000e900b34c55/templates/Template1/serviceGraphs/SG",
                "schemaName": "Schema 11",
                "templateName": "Template 1",
                "displayName": "SG"
            },
            "serviceNodes": [{
                "identity": {
                    "serviceNodeRef":
"/schemas/5963c0ff240000e900b34c55/templates/Template1/serviceGraphs/SG/serviceNodes/node1",

                    "serviceNodeTypeId": "0000ffff0000000000000051",
                    "index": 1,
                    "deviceDn": "/devicedn/lDevVip-FW2"
                }
            }],
            "deployedOn": []
        }]
  }
}
```

Example for ACI Multi-Site Release 1.0(2)

The following is an example that shows how to retrieve references.

GET

```
GET  https://<host>:<port>/api/v1/:id/references
```

REQUEST

```
  <empty>
```

RESPONSE

```
HTTP/1.1 200
{
  "references": {
    "vrfs": [
      {
        "identity": {
          "vrfRef": "/schemas/5963c0ff240000e900b34c55/templates/Template1/vrfs/VRF11",
          "schemaName": "Schema 11",
          "templateName": "Template 1",
          "displayName": "VRF 11"
        },
        "deployedOn": [
          "59631b3942000042001f1790"
        ]
      }
    ]
```

```
  }
}
```

Retrieving Schema Health and Faults

The following is an example for retrieving Schema health faults included.

Note Health and Fault information is sideloaded with Schema configuration in a separate internal structure called "HealthFaults". This structure closely resembles the template structure in terms of hierarchy of policies, and it includes health information for policies that can have health (ANP, EPG, VRF, BD) and faults for all policies that can have faults (ANP, EPG, VRF, BD, Contract) . The HealthFaults structures exist on a per template/site basis. Health and fault data will only exist for policies that have been pushed.

GET

```
GET /api/v1/schemas/583c7c482501002501061985?include=health,faults
```

REQUEST

```
  <empty>
```

RESPONSE

```
HTTP/1.1 200

{
    "id": "5971480c1b000067319d7b3b",
    "displayName": "Schema 1",
    "templates": [
        {
            "name": "Template1",
            "displayName": "Template 1",
            "tenantId": "5952ebc21e00007b16c1a022",
            "anps": [
                {
                    "name": "ANP1",
                    "displayName": "ANP1",
                    "anpRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/anps/ANP1",
                    "epgs": [
                        {
                            "name": "EPG1",
                            "displayName": "EPG1",
                            "epgRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/anps/ANP1/epgs/EPG1",
                            "contractRelationships": [],
                            "subnets": [],
                            "uSegEpg": false,
                            "uSegAttrs": [],
                            "intraEpg": "unenforced",
                            "proxyArp": false,
                            "bdRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/bds/BD1"
                        }
                    ]
```

```
                }
            ],
            "vrfs": [
                {
                    "name": "VRF1",
                    "displayName": "VRF1",
                    "vrfRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/vrfs/VRF1"
                }
            ],
            "bds": [
                {
                    "name": "BD1",
                    "displayName": "BD1",
                 "bdRef": "/schemas/5971480c1b000067319d7b3b/templates/Template1/bds/BD1",

                    "l2UnknownUnicast": "proxy",
                    "intersiteBumTrafficAllow": true,
                    "optimizeWanBandwidth": false,
                    "l2Stretch": true,
                    "subnets": [],
                    "vrfRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/vrfs/VRF1"
                }
            ],
            "contracts": [
                {
                    "name": "Contract1",
                    "displayName": "Contract1",
                    "contractRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/contracts/Contract1",
                    "filterRelationships": [],
                    "scope": "context"
                },
                {
                    "name": "Contract2",
                    "displayName": "Contract2",
                    "contractRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/contracts/Contract2",
                    "filterRelationships": [
                        {
                            "filterRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/filters/Filter1",
                            "directives": [
                                "none"
                            ]
                        },
                        {
                            "filterRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/filters/Filter2",
                            "directives": [
                                "none"
                            ]
                        }
                    ],
                    "scope": "context"
                }
            ],
            "filters": [
                {
                    "name": "Filter1",
                    "displayName": "Filter1",
                    "filterRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/filters/Filter1",
```

```
                    "entries": []
                },
                {
                    "name": "Filter2",
                    "displayName": "Filter2",
                    "filterRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/filters/Filter2",
                    "entries": [
                        {
                            "name": "Entry1",
                            "displayName": "Entry1",
                            "description": "",
                            "etherType": "ip",
                            "ipProtocol": "tcp",
                            "matchOnlyFragments": false,
                            "stateful": false,
                            "sourceFrom": "unspecified",
                            "sourceTo": "unspecified",
                            "destinationFrom": "unspecified",
                            "destinationTo": "unspecified",
                            "tcpSessionRules": [
                                "unspecified"
                            ]
                        }
                    ]
                }
            ]
        }
    ],
    "sites": [
        {
            "siteId": "594b77e84400004000752a18",
            "templateName": "Template1",
            "anps": [],
            "vrfs": [],
            "bds": [],
            "contracts": []
        },
        {
            "siteId": "594b77dc4400004400752a17",
            "templateName": "Template1",
            "anps": [],
            "vrfs": [],
            "bds": [],
            "contracts": []
        }
    ],
    "healthFaults": [
        {
            "siteId": "594b77e84400004000752a18",
            "templateName": "Template1",
            "health": {
                "score": 100
            },
            "anps": [
                {
                    "anpRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/anps/ANP1",
                    "health": {
                        "score": 100
                    },
                    "faults": [
                        {
                            "type": "minor",
```

```
                            "message": "Configuration failed for EPG EPG1 due to BD IDs Not
 Allocated,Context IDs Not Allocated",
                            "code": "F0523"
                        }
                    ],
                    "epgs": [
                        {
                            "epgRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/anps/ANP1/epgs/EPG1",
                            "health": {
                                "score": 100
                            },
                            "faults": [
                                {
                                    "type": "minor",
                                    "message": "Configuration failed for EPG EPG1 due to
BD IDs Not Allocated,Context IDs Not Allocated",
                                    "code": "F0523"
                                }
                            ]
                        }
                    ]
                }
            ],
            "vrfs": [
                {
                    "vrfRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/vrfs/VRF1",
                    "health": {
                        "score": 100
                    },
                    "faults": []
                }
            ],
            "bds": [
                {
                 "bdRef": "/schemas/5971480c1b000067319d7b3b/templates/Template1/bds/BD1",

                    "health": {
                        "score": 100
                    },
                    "faults": []
                }
            ],
            "contracts": [
                {
                    "contractRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/contracts/Contract1",
                    "faults": [
                        {
                            "type": "minor",
                            "message": "Configuration failed for Tenant DISA Contract
Contract1 due to Filter Not Present",
                            "code": "F1228"
                        }
                    ]
                },
                {
                    "contractRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/contracts/Contract2",
                    "faults": []
                }
            ]
        },
```

```
        {
            "siteId": "594b77dc4400004400752a17",
            "templateName": "Template1",
            "health": {
                "score": 100
            },
            "anps": [
                {
                    "anpRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/anps/ANP1",
                    "health": {
                        "score": 100
                    },
                    "faults": [
                        {
                            "type": "minor",
                           "message": "Configuration failed for EPG EPG1 due to BD IDs Not
 Allocated",
                            "code": "F0523"
                        }
                    ],
                    "epgs": [
                        {
                            "epgRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/anps/ANP1/epgs/EPG1",
                            "health": {
                                "score": 100
                            },
                            "faults": [
                                {
                                    "type": "minor",
                                    "message": "Configuration failed for EPG EPG1 due to
BD IDs Not Allocated",
                                    "code": "F0523"
                                }
                            ]
                        }
                    ]
                }
            ],
            "vrfs": [
                {
                    "vrfRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/vrfs/VRF1",
                    "health": {
                        "score": 100
                    },
                    "faults": []
                }
            ],
            "bds": [
                {
                 "bdRef": "/schemas/5971480c1b000067319d7b3b/templates/Template1/bds/BD1",

                    "health": {
                        "score": 100
                    },
                    "faults": []
                }
            ],
            "contracts": [
                {
                    "contractRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/contracts/Contract1",
```

```
                    "faults": [
                        {
                            "type": "minor",
                            "message": "Configuration failed for Tenant DISA Contract
Contract1 due to Filter Not Present",
                            "code": "F1228"
                        }
                    ]
                },
                {
                    "contractRef":
"/schemas/5971480c1b000067319d7b3b/templates/Template1/contracts/Contract2",
                    "faults": []
                }
            ]
        }
    ]
}
```

Update an Existing Schema

Example for ACI Multi-Site Release 1.0(2)

The following is an example to update an existing Schema.

PUT

```
PUT  https://<host>:<port>/api/v1/schemas/583c7c482501002501061985
```

REQUEST

```
{
  "id": "583c7c482501002501061985",
  "displayName": "Schema 1 (Updated)",
  "templates": [...],
  "sites": [
    {
      "siteId": "58202f7066e6e10001c41236",
      "templateName": "template1",
      "vrfs": [
        {
          "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1",
          "bds": [
            {
              "bdRef":
"/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1/bds/bd1",
              "subnet": "<subnet>"
            }
          ]
        }
      ]
    }
  ]
}
```

RESPONSE

```
HTTP/1.1 200
{
  "schemas": [
```

```
{
  "id": "594b78792d00008000d5fb6f",
  "displayName": "Saad_Stretched",
  "templates": [
    {
      "name": "Template1",
      "displayName": "Template 3",
      "tenantId": "594b77c82d00006000d5fb6c",
      "anps": [],
      "vrfs": [],
      "bds": [],
      "contracts": [],
      "filters": [],
      "externalEpgs": []
    },
    {
      "name": "Template2",
      "displayName": "Template 2safsdf",
      "tenantId": "594b77c82d00006000d5fb6c",
      "anps": [],
      "vrfs": [],
      "bds": [],
      "contracts": [],
      "filters": [],
      "externalEpgs": []
    }
  ],
  "sites": [
    {
      "siteId": "594b77dc4400004400752a17",
      "templateName": "Template1",
      "anps": [],
      "vrfs": [],
      "bds": [],
      "contracts": [],
      "externalEpgs": []
    },
    {
      "siteId": "594b77e84400004000752a18",
      "templateName": "Template1",
      "anps": [],
      "vrfs": [],
      "bds": [],
      "contracts": [],
      "externalEpgs": []
    },
    {
      "siteId": "594b77e84400004000752a18",
      "templateName": "Template2",
      "anps": [],
      "vrfs": [],
      "bds": [],
      "contracts": [],
      "externalEpgs": []
    }
  ]
},
{
  "id": "594b790a2d0000a900d5fb70",
  "displayName": "SharedService",
  "templates": [
    {
      "name": "Template1",
      "displayName": "Template 1",
```

```
            "tenantId": "594b782d2d00006f00d5fb6d",
            "anps": [],
            "vrfs": [],
            "bds": [],
            "contracts": [],
            "filters": [],
            "externalEpgs": []
          }
        ],
        "sites": [
          {
            "siteId": "594b77dc4400004400752a17",
            "templateName": "Template1",
            "anps": [],
            "vrfs": [],
            "bds": [],
            "contracts": [],
            "externalEpgs": []
          }
        ]
      }
    ]
}
```

Note

- "id" is optional, other attributes are mandatory even if unchanged.
- Many attributes in this example have been omitted for brevity, but they must be included in the PUT request. Refer to the full list of attributes in the POST example.
- The "ref" values for existing sub-entities must be specified.

Example for ACI Multi-Site Release 1.0(1)

The following is an example to updated an existing Schema.

PUT

```
PUT  https://<host>:<port>/api/v1/schemas/583c7c482501002501061985
```

REQUEST

```
{
  "id": "583c7c482501002501061985",
  "displayName": "Schema 1 (Updated)",
  "templates": [...],
  "sites": [
    {
      "siteId": "58202f7066e6e10001c41236",
      "templateName": "template1",
      "vrfs": [
        {
          "vrfRef": "/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1",
          "bds": [
            {
              "bdRef":
"/schemas/583c7c482501002501061985/templates/template1/vrfs/vrf1/bds/bd1",
              "subnet": "<subnet>"
            }
          ]
```

```
        }
      ]
    }
  ]
}
```

RESPONSE

```
HTTP/1.1 200
{
  "id": "583c7c4825010025010619 85",
  "displayName": "Schema 1 (Updated)",
  "templates": [...],
  "sites": [...]
```

Note

- "id" is optional, other attributes are mandatory even if unchanged.
- Many attributes in this example have been omitted for brevity, but they must be included in the PUT request. Refer to the full list of attributes in the POST example.
- The "ref" values for existing sub-entities must be specified.

Delete a Schema

The following is an example for deleting a Schema.

DELETE

```
DELETE  https://<host>:<port>/api/v1/schemas/583c7c4825010025010619 85
```

REQUEST

```
DELETE
Request Body:
  <empty>
```

RESPONSE

```
HTTP/1.1 204
```

List Schema Identities

The following is an example for listing Schema identities.

GET

```
GET  https://<host>:<port>/api/v1/schemas/list-identity
```

Note This provides a list of Schemas in the system with a few properties. Only template-tenant associations and template-site associations are returned. Policy configuration information in not included.

REQUEST

```
  <empty>
```

RESPONSE

```
HTTP/1.1 200
{
  "schemas": [
    {
      "id": "594b78792d00008000d5fb6f",
      "displayName": "Saad_Stretched",
      "templates": [
        {
          "name": "Template1",
          "displayName": "Template 3",
          "tenantId": "594b77c82d00006000d5fb6c",
          "anps": [],
          "vrfs": [],
          "bds": [],
          "contracts": [],
          "filters": []
        },
        {
          "name": "Template2",
          "displayName": "Template 2safsdf",
          "tenantId": "594b77c82d00006000d5fb6c",
          "anps": [],
          "vrfs": [],
          "bds": [],
          "contracts": [],
          "filters": []
        }
      ],
      "sites": [
        {
          "siteId": "594b77dc4400004400752a17",
          "templateName": "Template1",
          "anps": [],
          "vrfs": [],
          "bds": [],
          "contracts": []
        },
        {
          "siteId": "594b77e84400004000752a18",
          "templateName": "Template1",
          "anps": [],
          "vrfs": [],
          "bds": [],
          "contracts": []
        },
        {
          "siteId": "594b77e84400004000752a18",
          "templateName": "Template2",
          "anps": [],
```

```
          "vrfs": [],
          "bds": [],
          "contracts": []
        }
      ]
    },
    {
      "id": "594b790a2d0000a900d5fb70",
      "displayName": "SharedService",
      "templates": [
        {
          "name": "Template1",
          "displayName": "Template 1",
          "tenantId": "594b782d2d00006f00d5fb6d",
          "anps": [],
          "vrfs": [],
          "bds": [],
          "contracts": [],
          "filters": []
        }
      ],
      "sites": [
        {
          "siteId": "594b77dc4400004400752a17",
          "templateName": "Template1",
          "anps": [],
          "vrfs": [],
          "bds": [],
          "contracts": []
        }
      ]
    }
  ]
}
```

List Sites Associated to a Schema

The following is an example for listing sites associated to a Schema.

GET

```
GET https://<host>:<port>/api/v1/schemas/allowed-sites
```

REQUEST

```
GET
Request Body:
  <empty>
```

RESPONSE

```
HTTP/1.1 200
{
  "sites": [
    {
      "id": "591b6e5172040072044258ea",
      "name": "site1",
      "urls": [
        "https://10.23.236.232"
      ],
```

```
      "username": "admin",
      "password": "******",
      "siteId": "1"
    },
    {
      "id": "591b71353f00002700a2e8b1",
      "name": "site2",
      "urls": [
        "https://10.23.236.233"
      ],
      "username": "admin",
      "password": "******",
      "siteId": "2"
    }
  ]
}
```

List Tenants That Can be Associated With a Schema

The following is an example of how to list Tenants that can be associated with a Schema.

POST:

```
POST  https://<host>:<port>/api/v1/schemas/allowed-tenants
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "tenants": [
    {
      "id": "591b6da1200000dc03b47a10",
      "name": "TenantA",
      "displayName": "Tenant A",
      "description": "Tenant A",
      "siteAssociations": [
        {
          "siteId": "591b6e5172040072044258ea",
          "securityDomains": []
        }
      ],
      "userAssociations": [
        {
          "userId": "0000ffff0000000000000020"
        },
        {
          "userId": "591b562f210000d000823836"
        }
      ]
    },
    {
      "id": "591b6e35200000f303b47a11",
      "name": "TenantB",
      "displayName": "Tenant B",
      "description": "Tenant B",
```

```
      "siteAssociations": [
        {
          "siteId": "591b6e5172040072044258ea",
          "securityDomains": []
        },
        {
          "siteId": "591b71353f00002700a2e8b1",
          "securityDomains": []
        }
      ],
      "userAssociations": [
        {
          "userId": "0000ffff0000000000000020"
        }
      ]
    }
  ]
}
```

List All Tenants Associated to Schemas

The following is an example of how to list all Tenants associated to Schemas:

Endpoint:

```
https://<host>:<port>/api/v1/schemas/sites
```

Request:

```
  <empty>
```

Response:

```
 HTTP/1.1 200
{
  "tenants": [
    {
      "id": "0000ffff0000000000000010",
      "name": "common",
      "displayName": "common",
      "siteAssociations": [],
      "userAssociations": [
        {
          "userId": "0000ffff0000000000000020"
        }
      ],
      "description": "Common tenant for use with all other tenants"
    },
    {
      "id": "593b102322000088009ff785",
      "name": "TenantA",
      "displayName": "Tenant A",
      "siteAssociations": [
        {
          "siteId": "593b101169000069002de4c6",
          "securityDomains": [
            "TestSecurityDomain"
          ]
        }
      ],
      "userAssociations": [
```

```
        {
          "userId": "593b12e62000004b00f13b21"
        }
      ],
      "description": ""
    },
    {
      "id": "593b12cc2300002600eb63bd",
      "name": "TenantB",
      "displayName": "Tenant B",
      "siteAssociations": [
        {
          "siteId": "593b101169000069002de4c6",
          "securityDomains": []
        }
      ],
      "userAssociations": [
        {
          "userId": "593b12f62000005200f13b22"
        },
        {
          "userId": "593eba36210000140183e1a1"
        }
      ],
      "description": ""
    }
  ]
}
```

List Sites That Can Be Associated With a Schema

The following is an example of how to list Sites that can be associated with a Schema.

POST:

```
POST  https://<host>:<port>/api/v1/schemas/allowed-sites
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "sites": [
    {
      "id": "591b6e5172040072044258ea",
      "name": "site1",
      "urls": [
        "https://10.23.236.232"
      ],
      "username": "admin",
      "password": "******",
      "siteId": "1"
    },
    {
      "id": "591b71353f00002700a2e8b1",
      "name": "site2",
```

```
      "urls": [
        "https://10.23.236.233"
      ],
      "username": "admin",
      "password": "******",
      "siteId": "2"
    }
  ]
```

List All Sites Associated to Schemas

The following is an example of all Sites associated to Schemas.

POST:

```
POST  https://<host>:<port>/api/v1/schemas/sites
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "sites": [
    {
      "id": "591b6e5172040072044258ea",
      "name": "site1",
      "urls": [
        "https://10.23.236.232"
      ],
      "username": "admin",
      "password": "******",
      "siteId": "1"
    },
    {
      "id": "591b71353f00002700a2e8b1",
      "name": "site2",
      "urls": [
        "https://10.23.236.233"
      ],
      "username": "admin",
      "password": "******",
      "siteId": "2"
    }
  ]
}
```

Retrieve Service Graph Node Types

The following is an example of how to retrieve service graph node device types.

GET:

```
GET /api/v1/schemas/service-node-types
```

Request:

```
<empty>
```

Response:

```
{
    "serviceNodeTypes": [{
        "id": "0000ffff0000000000000051",
        "name": "firewall",
        "displayName": "Firewall"
    }, {
        "id": "0000ffff0000000000000052",
        "name": "load-balancer",
        "displayName": "Load Balancer"
    }, {
        "id": "0000ffff0000000000000053",
        "name": "other",
        "displayName": "Other"
    }]
}
```

Create Service Graph Node Types

The following is an example of how to create custom service graph node device types.

POST:

```
POST /api/v1/schemas/service-node-types
```

Request:

```
{
    "name":"IPS/IDS",
    "displayName":"IPS/IDS"
}
```

Response:

```
{
    "id":"5ac527982500006f0893c6da",
    "name":"IPS/IDS",
    "displayName":"IPS/IDS"
}
```

APIs For Import

Import a Tenant from APIC

Starting with ACI Multi-Site release 1.2(1), you will use the information in the response for this API call for certain service graph functions:

- Use the information in the **serviceGraphRelationship** area in the response for the association of a contract with a service graph. The **serviceGraphRelationship** area is highlighted in the example response below.
- Use the information in the first **serviceGraphs** area in the response for information on the service graph template, such as:
  - The name of the service graph (in this example response, **DemoSGT**)
  - Information on and number of service nodes in this service graph (in this example response, there are two nodes, **node1** and **node2**)

  The first **serviceGraphs** area is highlighted in the example response below.
- Use the information in the **serviceGraphTemplate** area in the response for contracts. The **serviceGraphTemplate** information is highlighted in the example response below.
- Finally, use information in the second **serviceGraphs** area in the response for devices. The second **serviceGraphs** area is highlighted in the example response below.

The following is an example showing how to import a tenant from APIC.

GET

```
GET  https://<host>:<port>/api/v1/import/sites/:id/tenants/:tenantId/templates/:templateName
```

An example:

```
/api/v1/import/sites/58b5f5ca390000390078e257/tenants/58b5f5ca390000390078e257/templates/template1
```

REQUEST

```
<empty>
```

RESPONSE

```
{
  "displayName": "",
  "templates": [
    {
      "name": "Template1",
      "displayName": "Template1",
      "tenantId": "5923588f800000e302e2a4fd",
      "anps": [
        {
          "name": "ANP1",
          "displayName": "ANP1",
          "anpRef": "/templates/Template1/anps/ANP1",
          "epgs": [
            {
              "name": "EPG1",
              "displayName": "EPG1",
              "epgRef": "/templates/Template1/anps/ANP1/epgs/EPG1",
              "contractRelationships": [
                {
                  "relationshipType": "provider",
                  "contractRef": "/templates/Template1/contracts/Contract1"
                }
              ],
              "subnets": [
                {
```

```
            "ip": "10.0.0.1/24",
            "scope": "private",
            "shared": false
          },
          {
            "ip": "5.5.5.5/21",
            "scope": "private",
            "shared": false
          }
        ],
        "uSegEpg": false,
        "uSegAttrs": [],
        "intraEpg": "unenforced",
        "bdRef": "/templates/Template1/bds/BD1"
      },
      {
        "name": "UntitledEPG1",
        "displayName": "UntitledEPG1",
        "epgRef": "/templates/Template1/anps/ANP1/epgs/UntitledEPG1",
        "contractRelationships": [],
        "subnets": [],
        "uSegEpg": false,
        "uSegAttrs": [],
        "intraEpg": "unenforced",
        "bdRef": "/templates/Template1/bds/BD1"
      }
    ]
  },
  {
    "name": "UntitledANP1",
    "displayName": "UntitledANP1",
    "anpRef": "/templates/Template1/anps/UntitledANP1",
    "epgs": [
      {
        "name": "EPG-Test",
        "displayName": "EPG-Test",
        "epgRef": "/templates/Template1/anps/UntitledANP1/epgs/EPG-Test",
        "contractRelationships": [],
        "subnets": [],
        "uSegEpg": false,
        "uSegAttrs": [],
        "intraEpg": "unenforced",
        "bdRef": "/templates/Template1/bds/BD1"
      }
    ]
  },
  {
    "name": "anp323",
    "displayName": "anp323",
    "anpRef": "/templates/Template1/anps/anp323",
    "epgs": [
      {
        "name": "sadfas",
        "displayName": "sadfas",
        "epgRef": "/templates/Template1/anps/anp323/epgs/sadfas",
        "contractRelationships": [],
        "subnets": [],
        "uSegEpg": false,
        "uSegAttrs": [],
        "intraEpg": "unenforced",
        "bdRef": "/templates/Template1/bds/BD1"
      }
    ]
  }
```

```
],
"vrfs": [
  {
    "name": "VRF1",
    "displayName": "VRF1",
    "vrfRef": "/templates/Template1/vrfs/VRF1"
  },
  {
    "name": "VRF2",
    "displayName": "VRF2",
    "vrfRef": "/templates/Template1/vrfs/VRF2"
  }
],
"bds": [
  {
    "name": "BD1",
    "displayName": "BD1",
    "bdRef": "/templates/Template1/bds/BD1",
    "l2UnknownUnicast": "proxy",
    "intersiteBumTrafficAllow": false,
    "l2Stretch": true,
    "subnets": [],
    "vrfRef": "/templates/Template1/vrfs/VRF1"
  }
],
"contracts": [
  {
    "name": "Contract1",
    "displayName": "Contract1",
    "contractRef": "/templates/Template1/contracts/Contract1",
    "filterRelationships": [
      {
        "filterRef": "/templates/Template1/filters/Filter1",
        "directives": []
      }
    ],
    "scope": "global",
    "subjectName": "subj1"
  },
  {
    "name": "Contract2",
    "displayName": "Contract2",
    "contractRef": "/templates/Template1/contracts/Contract2",
    "filterRelationships": [
      {
        "filterRef": "/templates/Template1/filters/Filter2",
        "directives": []
      },
      {
        "filterRef": "/templates/Template1/filters/Filter3",
        "directives": []
      }
    ],
    "scope": "global",
    "subjectName": "msc-subject"
  },
  {
    "name": "Contract3",
    "displayName": "Contract3",
    "contractRef": "/templates/Template1/contracts/Contract3",
    "filterRelationships": [],
    "scope": "context",
    "subjectName": "msc-subject"
  },
```

```
            {
              "name": "DemoContract",
              "displayName": "DemoContract",
              "contractRef": "/templates/Template1/contracts/DemoContract",
              "filterRelationships": [
                {
                  "filterRef": "/templates/Template1/filters/demofilter",
                  "directives": []
                }
              ],
              "scope": "context",
              "subjectName": "msc-subject",
              "filterType": "bothWay",
              "filterRelationshipsProviderToConsumer": [],
              "filterRelationshipsConsumerToProvider": [],
              "serviceGraphRelationship": {
                "serviceGraphRef": "/templates/Template1/serviceGraphs/DemoSGT",
                "serviceGraphContractRelationRef":
"/templates/Template1/contracts/DemoContract/serviceGraphs/DemoSGT",
                "serviceNodesRelationship": [
                  {
                    "serviceNodeRef":
"/templates/Template1/serviceGraphs/DemoSGT/serviceNodes/node1",
                    "providerConnector": {
                      "connectorType": "general",
                      "bdRef": "/templates/Template1/bds/DemoBD1",
                      "externalEpgRef": "/templates/Template1/externalEpgs/DemoExtEPG"
                    },
                    "consumerConnector": {
                      "connectorType": "general",
                      "bdRef": "/templates/Template1/bds/DemoBD5"
                    }
                  },
                  {
                    "serviceNodeRef":
"/templates/Template1/serviceGraphs/DemoSGT/serviceNodes/node2",
                    "providerConnector": {
                      "connectorType": "general",
                      "bdRef": "/templates/Template1/bds/DemoBD2"
                    },
                    "consumerConnector": {
                      "connectorType": "general",
                      "bdRef": "/templates/Template1/bds/DemoBD4"
                    }
                  }
                ]
            }
          }
        ],
        "filters": [
          {
            "name": "Filter1",
            "displayName": "Filter1",
            "filterRef": "/templates/Template1/filters/Filter1",
            "entries": []
          },
          {
            "name": "Filter2",
            "displayName": "Filter2",
            "filterRef": "/templates/Template1/filters/Filter2",
            "entries": [
              {
                "name": "Entry2",
```

```
                "displayName": "Entry2",
                "etherType": "ip",
                "arpFlag": "unspecified",
                "ipProtocol": "unspecified",
                "matchOnlyFragments": false,
                "stateful": false,
                "sourceFrom": "unspecified",
                "sourceTo": "unspecified",
                "destinationFrom": "unspecified",
                "destinationTo": "unspecified",
                "tcpSessionRules": [
                  "unspecified"
                ]
              }
            ]
          },
          {
            "name": "Filter3",
            "displayName": "Filter3",
            "filterRef": "/templates/Template1/filters/Filter3",
            "entries": []
          }
        ],
        "externalEpgs": [
          {
            "name": "Network",
            "displayName": "Network",
            "subnets": [],
            "contractRelationships": [
              {
                "relationshipType": "provider",
                "contractRef": "/templates/Template1/contracts/Contract1"
              },
              {
                "relationshipType": "consumer",
                "contractRef": "/templates/Template1/contracts/Contract2"
              },
              {
                "relationshipType": "consumer",
                "contractRef":
"/schemas/5983fcac220000d200b56034/templates/Template1/contracts/Contract1"
              }
            ],
            "vrfRef": "/templates/Template1/vrfs/VRF1",
            "externalEpgRef": "/templates/Template1/externalEpgs/Network"
          },
          {
            "name": "Network2",
            "displayName": "Network2",
            "subnets": [],
            "contractRelationships": [],
            "vrfRef": "/templates/Template1/vrfs/VRF1",
            "externalEpgRef": "/templates/Template1/externalEpgs/Network2"
          },
          {
            "name": "Network3",
            "displayName": "Network3",
            "subnets": [],
            "contractRelationships": [
              {
                "relationshipType": "consumer",
                "contractRef":
"/schemas/5983fcac220000d200b56034/templates/Template1/contracts/Contract1"
              }
```

```
            ],
            "vrfRef": "",
            "externalEpgRef": "/templates/Template1/externalEpgs/Network3"
          }
        ],
        "serviceGraphs": [
          {
            "name": "DemoSGT",
            "displayName": "DemoSGT",
            "serviceGraphRef": "/templates/Template1/serviceGraphs/DemoSGT",
            "serviceNodes": [
              {
                "name": "node1",
                "serviceNodeRef":
"/templates/Template1/serviceGraphs/DemoSGT/serviceNodes/node1",
                "serviceNodeTypeId": "0000ffff0000000000000051",
                "index": 1
              },
              {
                "name": "node2",
                "serviceNodeRef":
"/templates/Template1/serviceGraphs/DemoSGT/serviceNodes/node2",
                "serviceNodeTypeId": "0000ffff0000000000000052",
                "index": 2
              }
            ]
          }
        ]
      }
    ],
    "sites": [
      {
        "siteId": "591b427d820000a70055bbd5",
        "templateName": "Template1",
        "anps": [
          {
            "anpRef": "/templates/Template1/anps/ANP1",
            "epgs": [
              {
                "epgRef": "/templates/Template1/anps/ANP1/epgs/EPG1",
                "domainAssociations": [
                  {
                    "dn": "uni/phys-testPhysicalDomain-DoNotDelete",
                    "domainType": "physicalDomain",
                    "deployImmediacy": "lazy",
                    "resolutionImmediacy": "lazy"
                  },
                  {
                    "dn": "uni/l3dom-TestL3Domain-DoNotDelete",
                    "domainType": "l3ExtDomain",
                    "deployImmediacy": "lazy",
                    "resolutionImmediacy": "lazy"
                  },
                  {
                    "dn": "uni/fc-TestFibreChannelDomain",
                    "domainType": "fibreChannelDomain",
                    "deployImmediacy": "lazy",
                    "resolutionImmediacy": "lazy"
                  },
                  {
                    "dn": "uni/l2dom-TestL2Domain-DoNotDelete",
                    "domainType": "l2ExtDomain",
                    "deployImmediacy": "lazy",
                    "resolutionImmediacy": "lazy"
```

```
      }
    ],
    "staticPorts": [
      {
        "type": "port",
        "path": "topology/pod-1/paths-101/pathep-[eth1/1]",
        "portEncapVlan": 1234,
        "deploymentImmediacy": "lazy",
        "mode": "regular"
      }
    ],
    "staticLeafs": [
      {
        "path": "topology/pod-1/node-101",
        "portEncapVlan": 1234
      }
    ],
    "faults": []
  },
  {
    "epgRef": "/templates/Template1/anps/ANP1/epgs/UntitledEPG1",
    "domainAssociations": [
      {
        "dn": "uni/vmmp-VMware/dom-Fab1-DVS1",
        "domainType": "vmmDomain",
        "deployImmediacy": "lazy",
        "resolutionImmediacy": "immediate",
        "vlanEncapMode": "dynamic",
        "allowMicroSegmentation": true
      },
      {
        "dn": "uni/vmmp-VMware/dom-TestDomain",
        "domainType": "vmmDomain",
        "deployImmediacy": "lazy",
        "resolutionImmediacy": "immediate",
        "portEncapVlan": {
          "vlanType": "vlan",
          "vlan": 3
        },
        "vlanEncapMode": "static",
        "allowMicroSegmentation": false
      },
      {
        "dn": "uni/vmmp-VMware/dom-TestDomain-DoNotDelete",
        "domainType": "vmmDomain",
        "deployImmediacy": "lazy",
        "resolutionImmediacy": "immediate",
        "microSegVlan": {
          "vlanType": "vlan",
          "vlan": 2
        },
        "portEncapVlan": {
          "vlanType": "vlan",
          "vlan": 1
        },
        "vlanEncapMode": "static",
        "allowMicroSegmentation": true
      }
    ],
    "staticPorts": [
      {
        "type": "dpc",
        "path": "topology/pod-1/paths-101/pathep-[abc]",
        "portEncapVlan": 1234,
```

```
            "deploymentImmediacy": "lazy",
            "mode": "regular"
          }
        ],
        "staticLeafs": [],
        "faults": []
      }
    ],
    "faults": []
  },
  {
    "anpRef": "/templates/Template1/anps/UntitledANP1",
    "epgs": [
      {
        "epgRef": "/templates/Template1/anps/UntitledANP1/epgs/EPG-Test",
        "domainAssociations": [
          {
            "dn": "uni/vmmp-VMware/dom-TestDomain-DoNotDelete",
            "domainType": "vmmDomain",
            "deployImmediacy": "lazy",
            "resolutionImmediacy": "immediate",
            "portEncapVlan": {
              "vlanType": "vlan",
              "vlan": 243
            },
            "vlanEncapMode": "static",
            "allowMicroSegmentation": false
          }
        ],
        "staticPorts": [],
        "staticLeafs": [],
        "faults": []
      }
    ],
    "faults": []
  },
  {
    "anpRef": "/templates/Template1/anps/anp323",
    "epgs": [
      {
        "epgRef": "/templates/Template1/anps/anp323/epgs/sadfas",
        "domainAssociations": [],
        "staticPorts": [
          {
            "type": "vpc",
            "path": "topology/pod-1/protpaths-101-102/pathep-[vpc1]",
            "portEncapVlan": 1234,
            "deploymentImmediacy": "lazy",
            "mode": "regular"
          }
        ],
        "staticLeafs": [],
        "faults": []
      }
    ],
    "faults": []
  }
],
"vrfs": [],
"bds": [
  {
    "bdRef": "/templates/Template1/bds/BD1",
    "subnets": [],
    "faults": []
```

```
        }
      ],
      "contracts": [
        {
          "contractRef": "/templates/Template1/contracts/Contract3",
          "faults": []
        },
        {
          "contractRef": "/templates/Template1/contracts/DemoContract",
          "serviceGraphRelationship": {
            "serviceGraphRef": "/templates/Template1/serviceGraphs/DemoSGT",
            "serviceNodesRelationship": [
              {
                "serviceNodeRef":
"/templates/Template1/serviceGraphs/DemoSGT/serviceNodes/node1",
                "providerConnector": {
                  "clusterInterface": {
                    "dn": "uni/tn-BigC/lDevVip-A101/lIf-L2"
                  }
                },
                "consumerConnector": {
                  "clusterInterface": {
                    "dn": "uni/tn-BigC/lDevVip-A101/lIf-L1"
                  },
                  "redirectPolicy": {
                    "dn": "uni/tn-BigC/svcCont/svcRedirectPol-pbr8"
                  }
                }
              },
              {
                "serviceNodeRef":
"/templates/Template1/serviceGraphs/DemoSGT/serviceNodes/node2",
                "providerConnector": {
                  "clusterInterface": {
                    "dn": "uni/tn-BigC/lDevVip-LB101/lIf-L4"
                  }
                },
                "consumerConnector": {
                  "clusterInterface": {
                    "dn": "uni/tn-BigC/lDevVip-LB101/lIf-L3"
                  }
                }
              }
            ]
          }
        }
      ],
      "externalEpgs": [
        {
          "externalEpgRef": "/templates/Template1/externalEpgs/Network",
          "l3outDn": "L3Out-1",
          "subnets": [],
          "faults": []
        },
        {
          "externalEpgRef": "/templates/Template1/externalEpgs/Network2",
          "l3outDn": "L3Out-1",
          "subnets": [],
          "faults": []
        },
        {
          "externalEpgRef": "/templates/Template1/externalEpgs/Network3",
          "l3outDn": "L3Out-2",
```

```
          "subnets": [],
          "faults": []
        }
      ],
      "serviceGraphs": [
        {
          "serviceGraphRef": "/templates/Template1/serviceGraphs/DemoSGT",
          "serviceNodes": [
            {
              "serviceNodeRef":
"/templates/Template1/serviceGraphs/DemoSGT/serviceNodes/node1",
              "device": {
                "dn": "uni/tn-BigC/lDevVip-A101"
              }
            },
            {
              "serviceNodeRef":
"/templates/Template1/serviceGraphs/DemoSGT/serviceNodes/node2",
              "device": {
                "dn": "uni/tn-BigC/lDevVip-LB101"
              }
            }
          ]
        }
      ]
    }
  ],
  "messages": {
    "contractMessages": [
      {
        "contractRef": "/templates/Template1/contracts/Contract1",
        "message": "Contract has multiple subjects. Only the first subject is imported."
      }
    ],
    "vrfMessages": [
      {
        "vrfRef": "/templates/Template1/vrfs/VRF1",
        "message": "VRF contains vzAny contracts which are not imported."
      },
      {
        "vrfRef": "/templates/Template1/vrfs/VRF2",
        "message": "VRF contains vzAny contracts which are not imported."
      }
    ],
    "externalEpgMessages": [
      {
        "externalEpgRef": "/templates/Template1/externalEpgs/Network",
        "messages": [
          "External EPG will contain unmanaged contract interface Contract1-Interface which
will become orphaned after template is deployed to source site. Please make sure to remove
it after deploying to site."
        ]
      },
      {
        "externalEpgRef": "/templates/Template1/externalEpgs/Network3",
        "messages": [
          "External EPG will contain unmanaged contract interface Contract1-Interface which
will become orphaned after template is deployed to source site. Please make sure to remove
it after deploying to site.",
          "Parent L3Out L3Out-2 is not associated to a VRF."
        ]
      }
    ]
  }
```

```
}
```

APIs For Sites

Create a New Site

The following is an example of how to create a new Site.

POST:

```
POST https://<host>:<port>/api/v1/sites
```

Request:

```
{
  "name": "APIC Site San Francisco",
  "urls": "http://16.105.47.201:8982/sf",
  "username": "admin",
  "password": "w3lcome3",
  "apicSiteId" : "1"
}
```

Response:

```
HTTP/1.1 201
Location: /api/v1/sites/58202f7066e6e10001c41236
{
  "id": "58202f7066e6e10001c41236",
  "name": "APIC Site San Francisco",
  "urls": "http://16.105.47.201:8982/sf",
  "username": "admin",
  "password": "w3lcome3",
  "siteId" : "1"
}
```

Read Site Information

The following is an example of how to get basic information about a new Site:

Endpoint:

```
https://<host>:<port>/api/v1/sites/58202f7066e6e10001c41236
```

Request:

```
  GET
Request Body:
  <empty>
```

Response:

```
  HTTP/1.1 200
{
  "id": "58202f7066e6e10001c41236",
  "name": "APIC Site San Francisco",
  "urls": [
```

```
    "http://16.105.47.201:8982/sf"
  ],
  "username": "admin",
  "password": "******",
  "apicSiteId": "1"
}
```

Read Site Information That Contains Health Faults

The following is an example of how to get basic information about a new Site that also contains health faults:

Endpoint:

```
https://<host>:<port>/api/v1/sites/58202f7066e6e10001c41236?include=health,faults,cluster-status
```

Request:

```
  GET
Request Body:
  <empty>
```

Response:

```
HTTP/1.1 200
{
  "id": "58202f7066e6e10001c41236",
  "name": "APIC Site San Francisco",
  "urls": [
    "http://16.105.47.201:8982/sf"
  ],
  "username": "admin",
  "password": "******",
  "apicSiteId": "1",
  "labels" : ["58202f7066e6e10001c41240", "58202f7066e6e10001c41241"],
  "health": {
    "score": 97
  },
  "clusterStatus": [
    {
      "controllerId": 1,
      "controllerName": "apic1",
      "ipAddress": "10.0.0.1",
      "adminState": "in-service",
      "operationalState": "available",
      "health": "fully-fit"
    },
    {
      "controllerId": 2,
      "controllerName": "apic2",
      "ipAddress": "10.0.0.2",
      "adminState": "in-service",
      "operationalState": "available",
      "health": "fully-fit"
    },
    {
      "controllerId": 3,
      "controllerName": "apic3",
      "ipAddress": "10.0.0.3",
      "adminState": "in-service",
```

```
      "operationalState": "available",
      "health": "fully-fit"
    }
  ],

  "faults": [
    {
      "count": 0,
      "type": "critical"
    },
    {
      "count": 2,
      "type": "major"
    },
    {
      "count": 736,
      "type": "minor"
    },
    {
      "count": 914,
      "type": "warning"
    }
  ]
}
```

Update an Existing Site

The following is an example for updating an existing Site.

Endpoint:

```
https://<host>:<port>/api/v1/sites/58202f7066e6e10001c41236
```

Request

```
PUT
  {
  "id": "58202f7066e6e10001c41236",
  "name": "APIC Site San Francisco (Updated)",
  "urls": "http://16.105.47.201:8982/sf",
  "username": "root",
  "password": "w3lcome3",
  "apicSiteId" : "1"
}
```

Response

```
HTTP/1.1 200
{
  "id": "58202f7066e6e10001c41236",
  "name": "APIC Site San Francisco (Updated)",
  "urls": "http://16.105.47.201:8982/sf",
  "username": "root",
  "password": "w3lcome3",
  "apicSiteId" : "1"
}
```

Delete a Site

The following is an example of how to delete a Site:

Endpoint:

```
https://<host>:<port>/api/v1/sites/58202f7066e6e10001c41236
```

Request:

```
  DELETE
Request Body:
  <empty>
```

Response:

```
 HTTP/1.1 204
Response Body:
  <empty>
```

List Sites

The following is an example of how to list all Sites:

Endpoint:

```
https://<host>:<port>/api/v1/sites
```

Request:

```
  GET
Request Body:
  <empty>
```

Response:

```
  HTTP/1.1 200
{
  "sites": [
    {
      "id": "58202f7066e6e10001c41236",
      "name": "APIC Site San Francisco",
      "urls": [
        "http://16.105.47.201:8982/sf"
      ],
      "username": "admin",
      "password": "******",
      "apicSiteId": "1"
    },
    {
      "id": "58202fc466e6e10001c41237",
      "name": "APIC Site San Jose",
      "urls": [
        "http://16.105.47.201:8982/sjc"
      ],
      "username": "root",
      "password": "******",
```

```
      "apicSiteId": "1"
    }
  ]
}
```

Note Filtering, pagination, and sorting is currently not supported.

List Labels

List Labels of Site Type Associated to a Site

The following is an example of how to list labels of site type associated to a site.

GET:

```
GET /api/v1/sites/allowed-labels
```

Request:

```
<empty>
```

Note Filtering, pagination, and sorting is currently not supported.

Response:

```
HTTP/1.1 200
{
  "labels": [
    {
      "id": "58202f7066e6e10001c41240",
      "name" : "devops",
      "type" : "site"
    },
    {
      "id": "58202fc466e6e10001c41241",
      "name" : "prod",
      "type" : "site"
    },
    {
      "id": "58202f7066e6e10001c41242",
      "name" : "prod2",
      "type" : "site"
    },
  ]
}
```

List Labels associated to a Site

The following is an example of how to list labels associated to a site.

GET:

```
GET /api/v1/sites/58202f7066e6e10001c41236/labels
```

Request:

```
<empty>
```

Note Filtering, pagination, and sorting is currently not supported.

Response:

```
HTTP/1.1 200
{
  "labels": [
    {
      "id": "58202f7066e6e10001c41240",
      "name" : "devops",
      "type" : "site"
    },
    {
      "id": "58202fc466e6e10001c41241",
      "name" : "prod",
      "type" : "site"
    }
  ]
}
```

APIs For Template

Execute a Template

The following is an example of how to execute a Template under a Schema to its associated sites.

GET:

```
api/v1/execute/schema/{schema-id}/template/{template-name}
```

Request:

Response:

```
HTTP/1.1 200
{"msg":"Successfully deployed"}
```

Undeploy a Template

The following is an example of how to undeploy a Template from its associated sites.

GET:

```
api/v1/execute/schema/{schema-id}/template/{template-name}?undeploy={site-id}
```

Request:

Response:

```
HTTP/1.1 200
{"msg":"Successfully deployed"}
```

APIs For Entities

Login Domains

The following is an example of how to list login domains.

List Domains:

```
GET https://<host>:<port>/api/v1/sites/:siteId/aci/domains?type=:type
```

Example:

```
GET https://<host>:<port>/api/v1/sites/58202f7066e6e10001c41236/aci/domains?type=vmm
```

Domain Types:

- vmm
- physical
- L2Ext
- l3Ext
- fibreChannel

Request:

```
{
  <empty>
}
```

Response:

```
{
  "domains": [
    {
      "name": "cSphere",
      "dn": "uni/vmmp-VMware/dom-cSphere"
    },
    {
      "name": "Fab1-DVS1",
      "dn": "uni/vmmp-VMware/dom-Fab1-DVS1"
    },
    {
      "name": "TestDomain-DoNotDelete",
      "dn": "uni/vmmp-VMware/dom-TestDomain-DoNotDelete"
    }
  ]
}
```

List L3Outs

The following is an example for listing L3Outs.

GET:

```
GET https://<host>:<port>/api/v1/sites/:id/aci/tenants/:name/l3-outs
```

Request

```
{
  <empty>
}
```

Response

```
{
  "l3Outs": [
    {
      "name": "TestL3Out-DoNotDelete",
      "dn": "uni/tn-TestTenant-DoNotDelete/out-TestL3Out-DoNotDelete",
      "networkInstanceProfiles": [
        {
          "name": "TestNIP-1-DoNotDelete",
          "dn":
"uni/tn-TestTenant-DoNotDelete/out-TestL3Out-DoNotDelete/instP-TestNIP-1-DoNotDelete"
        },
        {
          "name": "TestNIP-2-DoNotDelete",
          "dn":
"uni/tn-TestTenant-DoNotDelete/out-TestL3Out-DoNotDelete/instP-TestNIP-2-DoNotDelete"
        }
      ]
    },
    {
      "name": "TestL3Out2",
      "dn": "uni/tn-TestTenant-DoNotDelete/out-TestL3Out2",
      "networkInstanceProfiles": [
        {
          "name": "N1",
          "dn": "uni/tn-TestTenant-DoNotDelete/out-TestL3Out2/Instp-N1"
        }
      ]
    }
  ]
}
```

APIs For Platforms

Read Node Information

The following is an example of how to read node information.

GET:

```
GET /api/v1/platform/nodes
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
    "nodes": [
        {
            "id": "cfihme4upn2yfpgadx7xyv9u7",
            "state": "ready",
            "ip": "192.168.65.2",
            "hostname": "moby"
        }
    ]
}
```

Download Troubleshooting Report

The following is an example of how to download troubleshooting report.

GET:

```
GET /api/v1/platform/logs?include=db_backup,infra_logs
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
zip file named msc_report_<timestamp>.zip
```

Expected HTTP Response Code Examples

The following are examples of a few HTTP response codes:

400: Bad Request

```
{
  "code": 125,
  "info": {
    "obj.password": [
      "Missing path"
    ],
    "obj.username": [
      "Missing path"
    ],
    "obj.urls": [
      "Invalid urls"
    ]
  },
  "message": "Malformed body"
}
```

404: Entity not found or invalid endpoint

```
{
  "code": 404,
  "message": "Not Found"
}
```

409: Conflict - Duplicate entity/resource

```
{
  "code": 140,
  "message": "Duplicate Resource"
}
```

APIs For Audit

About Auditing

Starting with Cisco ACI Multi-Site, release 1.1(1), the API to create, read, and filter audit records is supported. Therefore, the auditing examples are applicable starting with release 1.1(1) only.

APIs listed in this section are granular and enable you to read and insert audit records across various Multi-Site entities.

Note The auditing APIs are accessible only to administrators and users with the Power User role.

Downloading Audit Records: When downloading audit logs, one of the following APIs are invoked:

GET:

```
GET https://<host>:<port>/api/v1/audit-records/download?format=csv
GET https://<host>:<port>//api/v1/audit-records/download?format=csv
```

Filter Audit Records

GET:

```
GET https://<host>:<port>/api/v1/audit-records?type=tenant
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "auditRecords": [
    {
      "id": "59b5e11f1a0100270319b05e",
      "type": "tenant",
```

```
      "key": "59b5e11f3400004900c0ea97",
      "name": "Tenant 001",
      "event": "created",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "Admin User",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-09-10T01:04:31.701"
    },
    {
      "id": "59b5e11f1a0100270319b05e",
      "type": "tenant",
      "key": "59b5e11f3400004900c0ea97",
      "name": "Tenant 001",
      "event": "updated",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "Admin User",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-09-10T01:04:35.501"
    }
  ]
}
```


Filter Audit Records by **container**

The following are examples of container-type:

- tenant
- schema
- site
- fabric-connectivity-site

The following are examples of container-key:

- <id>
- <ref>

GET:

```
GET
https://<host>:<port>/api/v1/audit-records?container-type=tenant&container-key=59b5e11f3400004900c0ea97
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "auditRecords": [
    {
      "id": "59b5e1561a0100300319b05f",
      "type": "tenant-site",
      "key": "59cbf2194500004500741bd3",
      "name": "Site 1",
      "event": "added",
      "container": {
        "type": "tenant",
        "key": "59b5e11f3400004900c0ea97",
        "name": "Tenant 001"
      },
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "Admin User",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-09-10T01:04:35.501"
    }
  ]
}
```

Filter Audit Records by **container** and **context**.

The following are examples of container-type:

Note (Values are the same as "type" values)

- tenant
- schema
- site

The following are examples of container-key:

- <id>
- <ref>

The following are examples of context-type:

- site
- template

The following are examples of context-key:

- <id>
- <ref>

The following are examples of context-key:

GET:

```
GET https://<host>:<port>/api/v1/audit-records?container-type=<type>
&container-key=<key>&context-key=<key>&context-type=<type>&context-key=<key>

GET https://<host>:<port>/api/v1/audit-records?container-type=tenant&container-key=
/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/
epgs/Web-EPG&context-type=site&context-key=/schemas/59ef87743800008e06881fe5/templates
/Template1/sites/59726e89490000ea095b4cc9
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "auditRecords": [
    {
            "id": "5a3aacc52100003d007f4c1a",
            "type": "attribute",
            "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG/subnets",
            "event": "added",
            "container": {
                "type": "epg",
                "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG",
                "name": "Web-EPG",
                "context": {
                    "type": "site",
                    "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/sites/59726e89490000ea095b4cc9",
                    "name": "New York"
                }
            },
            "newValue": "10.0.0.1/24",
            "user": {
                "userId": "0000ffff0000000000000020",
                "username": "admin",
                "firstName": "Admin",
                "lastName": "User",
                "domainId": "0000ffff0000000000000090",
                "domainName": "Local"
            },
            "timestamp": "2017-12-20T18:32:37"
        }
  ]
}
```

Filter Audit Records by **username**

GET:

```
GET https://<host>:<port>/api/v1/audit-records?username=admin
```

GET:

```
GET https://<host>:<port>/api/v1/audit-records?username=admin,bob,lisa
```

Filter Audit Records by **event**

GET:

```
GET https://<host>:<port>/api/v1/audit-records?event=updated
```

GET:

```
GET https://<host>:<port>//api/v1/audit-records?event=associated,added
```

Filter Audit Records by **timestamp**

GET:

```
GET https://<host>:<port>/api/v1/audit-records?timestamp=2017-09-10T01:04:31.701
```

Filter Audit Records by date/time **range**

GET:

```
GET
https://<host>:<port>/api/v1/audit-records?start=2017-09-10T01:04:31.701&end=2017-09-15T08:06:45.811
```

GET:

```
GET https://<host>:<port>/api/v1/audit-records?start=2017-09-10T07:00&end=2017-09-15T07:00
```

Combine multiple filters

GET:

```
GET https://<host>:<port>/api/v1/audit-records?username=admin&event=associated,added
```

```
GET
https://<host>:<port>/api/v1/audit-records?username=admin&start=2017-09-10T07:00&end=2017-09-15T07:00
```

Paginate Audit Records

The following is an example of how to paginate audit records.

Use the parameters offset and limit for this purpose. Both parameters are optional but defaults apply to every GET request that retrieves audit records. The default values are as follows:

- offset: 0
- limit: 20

Note The maximum value for the limit is 250.

The total number of records is available in the response header named "X-Total-Count".

The prev and next "Link" headers are populated when applicable.

GET:

```
GET https://<host>:<port>/api/v1/audit-records?offset=3&limit=3
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
Link: </api/v1/audit-records?offset=0&limit=3>; rel="prev",
</api/v1/audit-records?offset=6&limit=3>; rel="next"
X-Total-Count: 76
{
  "auditRecords": [
    {
      "id": "59ed6de03d00003800175a2c",
      "type": "site",
      "key": "59ed6de04900004c00fcd96b",
      "name": "Site 1",
      "event": "created",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-23T04:19:44.355"
    },
    {
      "id": "59ed6de03d00003800175a2d",
      "type": "attribute",
      "key": "/sites/59ed6de04900004c00fcd96b/name",
      "event": "created",
      "container": {
        "type": "site",
        "key": "59ed6de04900004c00fcd96b",
        "name": "Site 1"
      },
      "newValue": "Site 1",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-23T04:19:44.355"
    },
    {
      "id": "59ed6de03d00003800175a2e",
      "type": "attribute",
      "key": "/sites/59ed6de04900004c00fcd96b/username",
      "event": "created",
      "container": {
        "type": "site",
        "key": "59ed6de04900004c00fcd96b",
        "name": "Site 1"
      },
      "newValue": "admin",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
```

```
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-23T04:19:44.355"
    }
  ]
}
```

About Login Domains Auditing

Login Domains Auditing APIs are accessible by the following users with the following roles:

- admin
- Power User
- User and Role Manager

List or Filter Login Domains Audit Records

Login Domains Audit Records

The following examples provide a high-level audit view of Login Domains that are created, updated, or deleted.

Note Default limit and offset values apply. Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined.

GET:

```
GET https://<host>:<port>/api/v1/auth/domains/audit-records
```

GET:

```
GET https://<host>:<port>/api/v1/auth/domains/audit-records?offset=0&limit=3
```

GET:

```
GET
https://<host>:<port>/api/v1/auth/domains/audit-records?username=admin&event=created,updated
```

Single Login Domain Audit Records

The following examples provide a detailed audit view of a single Login Domain.

Note Default limit and offset values apply. Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined.

GET:

```
GET https://<host>:<port>/api/v1/auth/domains/:id/audit-records
```

GET:

```
GET https://<host>:<port>/api/v1/auth/domains/:id/audit-records?offset=0&limit=3
```

GET:

```
GET
https://<host>:<port>/api/v1/auth/domains/:id/audit-records?username=admin&event=created,updated
```

GET:

```
GET
https://<host>:<port>/api/v1/auth/domains/:id/audit-records?timestamp=2017-10-15T01:04:35.603
```

All Users Associated with Login Domain Audit Records

The following examples are for all users associated with Login Domain audit records.

GET:

```
GET https://<host>:<port>/api/v1/auth/domains/audit-records/users
```

GET:

```
GET https://<host>:<port>/api/v1/auth/domains/:id/audit-records/users
```

About Users Auditing

The Users Auditing APIs are accessible with the following roles:

- admin
- Power User
- User and Role Manager

List or Filter User Audit Records

User Audit Records

The following examples provide a high-level audit view of users that were created, updated, or deleted.

Note Default limit and offset values apply. Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined.

GET:

```
GET https://<host>:<port>/api/v1/users/audit-records
```

GET:

```
GET https://<host>:<port>/api/v1/users/audit-records?offset=0&limit=3
```

GET:

```
GET https://<host>:<port>/api/v1/users/audit-records?username=admin&event=created,updated
```

Single User Audit Records

The following examples provide a detailed audit view of a single user.

Note Default limit and offset values apply. Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined.

GET:

```
GET https://<host>:<port>/api/v1/users/:id/audit-records
```

GET:

```
GET https://<host>:<port>/api/v1/users/:id/audit-records?offset=0&limit=3
```

GET:

```
GET https://<host>:<port>/api/v1/users/:id/audit-records?username=admin&event=created,updated
```

GET:

```
GET https://<host>:<port>/api/v1/users/:id/audit-records?timestamp=2017-10-15T01:04:35.603
```

All Users Associated with User Audit Records

The following examples are for all users associated with user audit records.

GET:

```
GET https://<host>:<port>/api/v1/users/audit-records/users
```

GET:

```
GET https://<host>:<port>/api/v1/users/:id/audit-records/users
```

About Providers Auditing

The Providers Auditing APIs are accessible by the following users with the following roles:

- admin
- Power User
- User and Role Manager

List or Filter Provider Audit Records

Providers Audit Records

The following examples provide a high-level audit view of providers that were created, updated, or deleted.

Note Default limit and offset values apply. Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined.

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/audit-records
```

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/audit-records?offset=0&limit=3
```

GET:

```
GET
https://<host>:<port>/api/v1/auth/providers/audit-records?username=admin&event=created,updated
```

Single Provider Audit Records

The following examples provide a detailed audit view of a single provider.

Note Default limit and offset values apply. Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined.

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/:id/audit-records
```

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/:id/audit-records?offset=0&limit=3
```

GET:

```
GET
https://<host>:<port>/api/v1/auth/providers/:id/audit-records?username=admin&event=created,updated
```

GET:

```
GET
https://<host>:<port>/api/v1/auth/providers/:id/audit-records?timestamp=2017-10-15T01:04:35.603
```

All Users Associated with Provider Audit Records

The following examples are for all users associated with provider audit records.

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/audit-records/users
```

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/:id/audit-records/users
```

List or Filter RADIUS Provider Audit Records

RADIUS Providers Audit Records

The following examples provide a high-level audit view of RADIUS providers that were created, updated, or deleted.

Note Default limit and offset values apply. Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined.

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/radius/audit-records
```

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/radius/audit-records?offset=0&limit=3
```

GET:

```
GET
https://<host>:<port>/api/v1/auth/providers/radius/audit-records?username=admin&event=created,updated
```

Single RADIUS Provider Audit Records

The following examples provide a detailed audit view of a single RADIUS provider.

Note Default limit and offset values apply. Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined.

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/radius/:id/audit-records
```

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/radius/:id/audit-records?offset=0&limit=3
```

GET:

```
GET
https://<host>:<port>/api/v1/auth/providers/radius/:id/audit-records?username=admin&event=created,updated
```

GET:

```
GET
https://<host>:<port>/api/v1/auth/providers/radius/:id/audit-records?timestamp=2017-10-15T01:04:35.603
```

All Users Associated with RADIUS Provider Audit Records

The following examples are for all users associated with RADIUS provider audit records.

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/radius/audit-records/users
```

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/radius/:id/audit-records/users
```

List or Filter TACACS Provider Audit Records

TACACS Providers Audit Records

The following examples provide a high-level audit view of TACACS providers that were created, updated, or deleted.

Note Default limit and offset values apply. Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined.

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/tacacs/audit-records
```

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/tacacs/audit-records?offset=0&limit=3
```

GET:

```
GET
https://<host>:<port>/api/v1/auth/providers/tacacs/audit-records?username=admin&event=created,updated
```

Single TACACS Provider Audit Records

The following examples provide a detailed audit view of a single TACACS provider.

Note Default limit and offset values apply. Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined.

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/tacacs/:id/audit-records
```

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/tacacs/:id/audit-records?offset=0&limit=3
```

GET:

```
GET
https://<host>:<port>/api/v1/auth/providers/tacacs/:id/audit-records?username=admin&event=created,updated
```

GET:

```
GET
https://<host>:<port>/api/v1/auth/providers/tacacs/:id/audit-records?timestamp=2017-10-15T01:04:35.603
```

All Users Associated with TACACS Provider Audit Records

The following examples are for all users associated with TACACS provider audit records.

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/tacacs/audit-records/users
```

GET:

```
GET https://<host>:<port>/api/v1/auth/providers/tacacs/:id/audit-records/users
```

Find Users Associated with Tenant Audit Records

The following example shows how to find all users associated with Tenant audit records.

GET:

```
GET https://<host>:<port>/api/v1/tenants/audit-records/users
```

```
GET https://<host>:<port>/api/v1/tenants/:id/audit-records/users
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "auditUsers": [
    {
      "userId": "0000ffff0000000000000020",
      "username": "admin",
      "firstName": "Admin",
      "lastName": "User",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    },
    {
      "userId": "59ed6de03d00003800175a2e",
      "username": "bob",
      "firstName": "Bob",
      "lastName": "Smith",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    }
```

```
  ]
}
```

List or Filter Audit Records for Tenants

Note

- Default limit and offset values apply.
- Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined.

The following example shows how to list or filter audit records for all Tenants. These APIs provide a high-level audit view of tenants that were created, updated, or deleted.

GET:

```
GET https://<host>:<port>/api/v1/tenants/audit-records

GET https://<host>:<port>/api/v1/tenants/audit-records?offset=0&limit=3

GET https://<host>:<port>/api/v1/tenants/audit-records?username=admin&event=created,updated
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "auditRecords": [
    {
      "id": "59db14881e0000f00071f40e",
      "type": "tenant",
      "key": "59db147b2500000b01c90429",
      "name": "Tenant 001",
      "event": "updated",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-09T06:17:44.141"
    },
    {
      "id": "59db147c1e0000f00071f40c",
      "type": "tenant",
      "key": "59db147b2500000b01c90429",
      "name": "Tenant 001",
      "event": "created",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
```

```
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-09T06:17:31.873"
    },
    {
      "id": "59b5e11f1a0100270319b06a",
      "type": "tenant",
      "key": "59b5e11f3400004900c0ea98",
      "name": "Tenant 000",
      "event": "created",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-09-15T01:04:35.501"
    }
  ]
}
```

The following example shows how to list or filter a single Tenant's audit records.

GET:

```
GET https://<host>:<port>/api/v1/tenants/:id/audit-records

GET https://<host>:<port>/api/v1/tenants/:id/audit-records?offset=0&limit=4

GET
https://<host>:<port>/api/v1/tenants/:id/audit-records?username=admin&event=created,updated


GET https://<host>:<port>/api/v1/tenants/:id/audit-records?timestamp=2017-09-15T01:04:35.501
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "auditRecords": [
    {
      "id": "59db14881e0000f00071f40f",
      "type": "attribute",
      "key": "/tenants/59db147b2500000b01c90429/description",
      "event": "updated",
      "container": {
        "type": "tenant",
        "key": "59db147b2500000b01c90429",
        "name": "Tenant 001"
      },
      "oldValue": "",
      "newValue": "Description for Tenant 001",
      "user": {
```

```
      "userId": "0000ffff0000000000000020",
      "username": "admin",
      "firstName": "Admin",
      "lastName": "User",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    },
    "timestamp": "2017-10-09T06:17:44.141"
  },
  {
    "id": "59db14881e0000f00071f410",
    "type": "tenant-site",
    "key": "59d3f3254c00004c008172c5",
    "name": "Site 1",
    "event": "associated",
    "container": {
      "type": "tenant",
      "key": "59db147b2500000b01c90429",
      "name": "Tenant 001"
    },
    "user": {
      "userId": "0000ffff0000000000000020",
      "username": "admin",
      "firstName": "Admin",
      "lastName": "User",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    },
    "timestamp": "2017-10-09T06:17:44.141"
  },
  {
    "id": "59db14881e0000f00071f411",
    "type": "tenant-user",
    "key": "0000ffff0000000000000020",
    "name": "admin",
    "event": "associated",
    "container": {
      "type": "tenant",
      "key": "59db147b2500000b01c90429",
      "name": "Tenant 001"
    },
    "user": {
      "userId": "0000ffff0000000000000020",
      "username": "admin",
      "firstName": "Admin",
      "lastName": "User",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    },
    "timestamp": "2017-10-09T06:17:31.873"
  },
  {
    "id": "59db147c1e0000f00071f40d",
    "type": "attribute",
    "key": "/tenants/59db147b2500000b01c90429/displayName",
    "event": "created",
    "container": {
      "type": "tenant",
      "key": "59db147b2500000b01c90429",
      "name": "Tenant 001"
    },
    "oldValue": "",
    "newValue": "Tenant 001",
    "user": {
```

```
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-09T06:17:31.873"
    }
  ]
}
```

Find Users Associated with Schema Audit Records

The following example shows how to find all users associated with Schema audit records.

GET:

```
GET https://<host>:<port>/api/v1/schema/audit-records/users

GET https://<host>:<port>/api/v1/schema/:id/audit-records/users
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "auditUsers": [
    {
      "userId": "0000ffff0000000000000020",
      "username": "admin",
      "firstName": "Admin",
      "lastName": "User",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    },
    {
      "userId": "59ed6de03d00003800175a2e",
      "username": "bob",
      "firstName": "Bob",
      "lastName": "Smith",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    }
  ]
}
```

List or Filter Audit Records for Schemas

Note

- Default limit and offset values apply.
- Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined. This is a powerful API, and it allows you to filter records related to, for example, all ANPs in a template, a particular ANP, or all policies in a template.

The following example shows how to list or filter audit records for all Schemas. These APIs provide a high-level audit view of schemas that were created, updated, or deleted.

GET:

```
GET https://<host>:<port>/api/v1/schemas/audit-records

GET https://<host>:<port>/api/v1/schemas/audit-records?offset=0&limit=3

GET https://<host>:<port>/api/v1/schemas/audit-records?username=admin&event=created,updated
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
  "auditRecords": [
    {
      "id": "59ef87953a00001f069baf83",
      "type": "schema",
      "key": "59ef87743800008e06881fe5",
      "name": "Test Schema",
      "event": "updated",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:57.37"
    },
    {
      "id": "59ef87743a000022069baf60",
      "type": "schema",
      "key": "59ef87743800008e06881fe5",
      "name": "Test Schema",
      "event": "created",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
```

```
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    }
  ]
}
```

The following example shows how to list or filter a single Schema audit records. It shows audit records generated by a single schema.

GET:

```
GET https://<host>:<port>/api/v1/schemas/:id/audit-records

GET https://<host>:<port>/api/v1/schemas/:id/audit-records?offset=0&limit=4

GET
https://<host>:<port>/api/v1/schemas/:id/audit-records?username=admin&event=created,updated
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200

{
  "auditRecords": [
    {
      "id": "59ef87743a000022069baf60",
      "type": "schema",
      "key": "59ef87743800008e06881fe5",
      "name": "Test Schema",
      "event": "created",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf61",
      "type": "template",
      "key": "schemas/59ef87743800008e06881fe5/templates/Template1",
      "name": "Template 1",
      "event": "added",
      "container": {
        "type": "schema",
        "key": "59ef87743800008e06881fe5",
        "name": "Test Schema"
      },
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
```

```
      "firstName": "Admin",
      "lastName": "User",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    },
    "timestamp": "2017-10-24T18:33:24.584"
  },
  {
    "id": "59ef87743a000022069baf62",
    "type": "attribute",
    "key": "schemas/59ef87743800008e06881fe5/templates/Template1/displayName",
    "event": "created",
    "container": {
      "type": "template",
      "key": "schemas/59ef87743800008e06881fe5/templates/Template1",
      "name": "Template 1"
    },
    "newValue": "Template 1",
    "user": {
      "userId": "0000ffff0000000000000020",
      "username": "admin",
      "firstName": "Admin",
      "lastName": "User",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    },
    "timestamp": "2017-10-24T18:33:24.584"
  },
  {
    "id": "59ef87743a000022069baf63",
    "type": "anp",
    "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1",
    "name": "App-1",
    "event": "created",
    "container": {
      "type": "template",
      "key": "schemas/59ef87743800008e06881fe5/templates/Template1",
      "name": "Template 1"
    },
    "user": {
      "userId": "0000ffff0000000000000020",
      "username": "admin",
      "firstName": "Admin",
      "lastName": "User",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    },
    "timestamp": "2017-10-24T18:33:24.584"
  },
  {
    "id": "59ef87743a000022069baf64",
    "type": "attribute",
   "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/displayName",

    "event": "created",
    "container": {
      "type": "anp",
      "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1",
      "name": "App-1"
    },
    "newValue": "App-1",
    "user": {
      "userId": "0000ffff0000000000000020",
      "username": "admin",
```

```
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf65",
      "type": "epg",
     "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG",

      "name": "Web-EPG",
      "event": "created",
      "container": {
        "type": "template",
        "key": "schemas/59ef87743800008e06881fe5/templates/Template1",
        "name": "Template 1"
      },
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf66",
      "type": "attribute",
      "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG/displayName",

      "event": "created",
      "container": {
        "type": "epg",
        "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG",
        "name": "Web-EPG"
      },
      "newValue": "Web-EPG",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf67",
      "type": "attribute",
      "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG/uSegEpg",
      "event": "created",
      "container": {
        "type": "epg",
        "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG",
        "name": "Web-EPG"
```

```
      },
      "newValue": "false",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf68",
      "type": "attribute",
      "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG/intraEpg",
      "event": "created",
      "container": {
        "type": "epg",
        "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG",
        "name": "Web-EPG"
      },
      "newValue": "unenforced",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf69",
      "type": "attribute",
      "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG/proxyArp",
      "event": "created",
      "container": {
        "type": "epg",
        "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG",
        "name": "Web-EPG"
      },
      "newValue": "false",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf6a",
      "type": "attribute",
      "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG/bdRef",
      "event": "associated",
```

```
      "container": {
        "type": "epg",
        "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/Web-EPG",
        "name": "Web-EPG"
      },
      "newValue": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf6b",
      "type": "epg",
     "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/DB-EPG",

      "name": "DB-EPG",
      "event": "created",
      "container": {
        "type": "template",
        "key": "schemas/59ef87743800008e06881fe5/templates/Template1",
        "name": "Template 1"
      },
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf6c",
      "type": "attribute",
      "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/DB-EPG/displayName",

      "event": "created",
      "container": {
        "type": "epg",
       "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/DB-EPG",

        "name": "DB-EPG"
      },
      "newValue": "DB-EPG",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
```

```
      "id": "59ef87743a000022069baf6d",
      "type": "attribute",
      "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/DB-EPG/uSegEpg",
      "event": "created",
      "container": {
        "type": "epg",
       "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/DB-EPG",

        "name": "DB-EPG"
      },
      "newValue": "false",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf6e",
      "type": "attribute",
      "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/DB-EPG/intraEpg",
      "event": "created",
      "container": {
        "type": "epg",
       "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/DB-EPG",

        "name": "DB-EPG"
      },
      "newValue": "unenforced",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf6f",
      "type": "attribute",
      "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/DB-EPG/proxyArp",
      "event": "created",
      "container": {
        "type": "epg",
       "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/DB-EPG",

        "name": "DB-EPG"
      },
      "newValue": "false",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
```

```
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf70",
      "type": "attribute",
      "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/DB-EPG/bdRef",
      "event": "associated",
      "container": {
        "type": "epg",
       "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/anps/App-1/epgs/DB-EPG",

        "name": "DB-EPG"
      },
      "newValue": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf71",
      "type": "bd",
      "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1",
      "name": "BD1",
      "event": "created",
      "container": {
        "type": "template",
        "key": "schemas/59ef87743800008e06881fe5/templates/Template1",
        "name": "Template 1"
      },
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    },
    {
      "id": "59ef87743a000022069baf72",
      "type": "attribute",
      "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1/displayName",
      "event": "created",
      "container": {
        "type": "bd",
        "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1",
        "name": "BD1"
      },
      "newValue": "BD1",
      "user": {
        "userId": "0000ffff0000000000000020",
        "username": "admin",
        "firstName": "Admin",
        "lastName": "User",
```

```
          "domainId": "0000ffff0000000000000090",
          "domainName": "Local"
        },
        "timestamp": "2017-10-24T18:33:24.584"
      },
      {
        "id": "59ef87743a000022069baf73",
        "type": "attribute",
       "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1/l2UnknownUnicast",

        "event": "created",
        "container": {
          "type": "bd",
          "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1",
          "name": "BD1"
        },
        "newValue": "proxy",
        "user": {
          "userId": "0000ffff0000000000000020",
          "username": "admin",
          "firstName": "Admin",
          "lastName": "User",
          "domainId": "0000ffff0000000000000090",
          "domainName": "Local"
        },
        "timestamp": "2017-10-24T18:33:24.584"
      },
      {
        "id": "59ef87743a000022069baf74",
        "type": "attribute",
        "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1/intersiteBumTrafficAllow",
        "event": "created",
        "container": {
          "type": "bd",
          "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1",
          "name": "BD1"
        },
        "newValue": "true",
        "user": {
          "userId": "0000ffff0000000000000020",
          "username": "admin",
          "firstName": "Admin",
          "lastName": "User",
          "domainId": "0000ffff0000000000000090",
          "domainName": "Local"
        },
        "timestamp": "2017-10-24T18:33:24.584"
      },
      {
        "id": "59ef87743a000022069baf75",
        "type": "attribute",
        "key":
"/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1/optimizeWanBandwidth",
        "event": "created",
        "container": {
          "type": "bd",
          "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1",
          "name": "BD1"
        },
        "newValue": "true",
        "user": {
          "userId": "0000ffff0000000000000020",
          "username": "admin",
```

```
      "firstName": "Admin",
      "lastName": "User",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    },
    "timestamp": "2017-10-24T18:33:24.584"
  },
  {
    "id": "59ef87743a000022069baf76",
    "type": "attribute",
    "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1/l2Stretch",
    "event": "created",
    "container": {
      "type": "bd",
      "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1",
      "name": "BD1"
    },
    "newValue": "true",
    "user": {
      "userId": "0000ffff0000000000000020",
      "username": "admin",
      "firstName": "Admin",
      "lastName": "User",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    },
    "timestamp": "2017-10-24T18:33:24.584"
  },
  {
    "id": "59ef87743a000022069baf77",
    "type": "attribute",
    "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1/vrfRef",
    "event": "associated",
    "container": {
      "type": "bd",
      "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1",
      "name": "BD1"
    },
    "newValue": "/schemas/59ef87743800008e06881fe5/templates/Template1/vrfs/VRF1",
    "user": {
      "userId": "0000ffff0000000000000020",
      "username": "admin",
      "firstName": "Admin",
      "lastName": "User",
      "domainId": "0000ffff0000000000000090",
      "domainName": "Local"
    },
    "timestamp": "2017-10-24T18:33:24.584"
  },
  {
    "id": "59ef87743a000022069baf78",
    "type": "attribute",
    "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1/subnets",
    "event": "added",
    "container": {
      "type": "bd",
      "key": "/schemas/59ef87743800008e06881fe5/templates/Template1/bds/BD1",
      "name": "BD1"
    },
    "newValue": "10.0.0.1/24",
    "user": {
      "userId": "0000ffff0000000000000020",
      "username": "admin",
      "firstName": "Admin",
```

```
        "lastName": "User",
        "domainId": "0000ffff0000000000000090",
        "domainName": "Local"
      },
      "timestamp": "2017-10-24T18:33:24.584"
    }
  ]
}
```


Find Users Associated with Fabric-Connectivity Audit Records

The following example shows how to find all users associated with Fabric-Connectivity audit records.

GET:

```
GET /api/v1/sites/fabric-connectivity/audit-records/users
```

List or Filter Audit Records for Fabric-Connectivity

The following example shows how to list or filter audit records for fabric-connectivity. This example provides a high-level audit view of updates for fabric-connectivity.

Note

- Default limit and offset values apply.
- Multiple additional filters (such as username, event, timestamp, date range) may be applied. Filters may also be combined.

GET:

```
GET https://<host>:<port>/api/v1/sites/fabric-connectivity/audit-records

GET https://<host>:<port>/api/v1/sites/fabric-connectivity/audit-records?offset=0&limit=3

GET
https://<host>:<port>/api/v1/sites/fabric-connectivity/audit-records?username=admin&event=created,updated
```

Request:

APIs For Backups

About Backup

Starting with Cisco ACI Multi-Site, release 1.1(1), the APIs for backup are supported. Therefore, the backup examples are applicable starting with release 1.1(1) only.

Create a New Backup

The following is an example of how to create a new backup.

POST:

```
POST https://<host>:<port>/api/v1/backups
```

Request:

```
{
    "name": "testFile3",
    "description": "this is optional key value"
```

Response:

```
HTTP/1.1 201
Location: /api/v1/backups/20171116221404
{
    "id": "20171116221404",
    "sizeKB": 18.44,
    "timestamp": "2017-11-16T22:14:04",
    "metadata": {
        "description": "test-description",
        "name": "Check_20171116221404"
    }
}
```

List Backups

The following is an example of how to list backups.

GET:

```
GET https://<host>:<port>/api/v1/backups
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
    "backups": [
        {
            "id": "20171116221404",
            "sizeKB": 18.44,
            "timestamp": "2017-11-16T22:14:04",
            "metadata": {
                "description": "test-description",
                "name": "Check_20171116221404"
            }
        },
        {
            "id": "20171116221721",
            "sizeKB": 18.55,
            "timestamp": "2017-11-16T22:17:21",
```

```
            "metadata": {
                "description": "test-description",
                "name": "Check_20171116221721"
            }
        },
        {
            "id": "20171116221724",
            "sizeKB": 19.11,
            "timestamp": "2017-11-16T22:17:24",
            "metadata": {
                "description": "test-description",
                "name": "Check_20171116221724"
            }
        },
        {
            "id": "20171116224425",
            "sizeKB": 18.99,
            "timestamp": "2017-11-16T22:44:25",
            "metadata": {
                "description": "test-description",
                "name": "Check_20171116224425"
            }
        },
        {
            "id": "20171127071624",
            "sizeKB": 24.72,
            "timestamp": "2017-11-27T07:16:24",
            "metadata": {
                "name": "testFile3_20171127071624"
            }
        },
        {
            "id": "20171127071655",
            "sizeKB": 28.08,
            "timestamp": "2017-11-27T07:16:55",
            "metadata": {
                "description": "this is optional key value",
                "name": "testFile3_20171127071655"
            }
        }
    ]
}
```

The following is an example of how to list a backup.

GET:

```
GET https://<host>:<port>/api/v1/backups/:id
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
{
            "id": "20170926231729",
            "sizeKB": 1.91,
            "dateTime": "2017-09-26T23:17:29",
            "metadata": {
                "description": "Right before zip review"
```

```
            }
}
```

Download a Backup

The following is an example of how to download a backup.

GET:

```
GET https://<host>:<port>/api/v1/backups/:id/download
```

Request:

```
<empty>
```

Response:

```
HTTP/1.1 200
Binary output as it is a file
```

Upload a Backup

The following is an example of how to upload a backup.

GET:

```
GET https://<host>:<port>/api/v1/backups/:id/upload
```

Request:

```
form-data
key: name
value: valid tar.gz filename
```

Response:

```
HTTP/1.1 200
{
    "id": "20171127071655",
    "sizeKB": 28.08,
    "timestamp": "2017-11-27T07:16:55",
    "metadata": {
        "description": "this is optional key value",
        "name": "testFile3_20171127071655"
    }
}
```