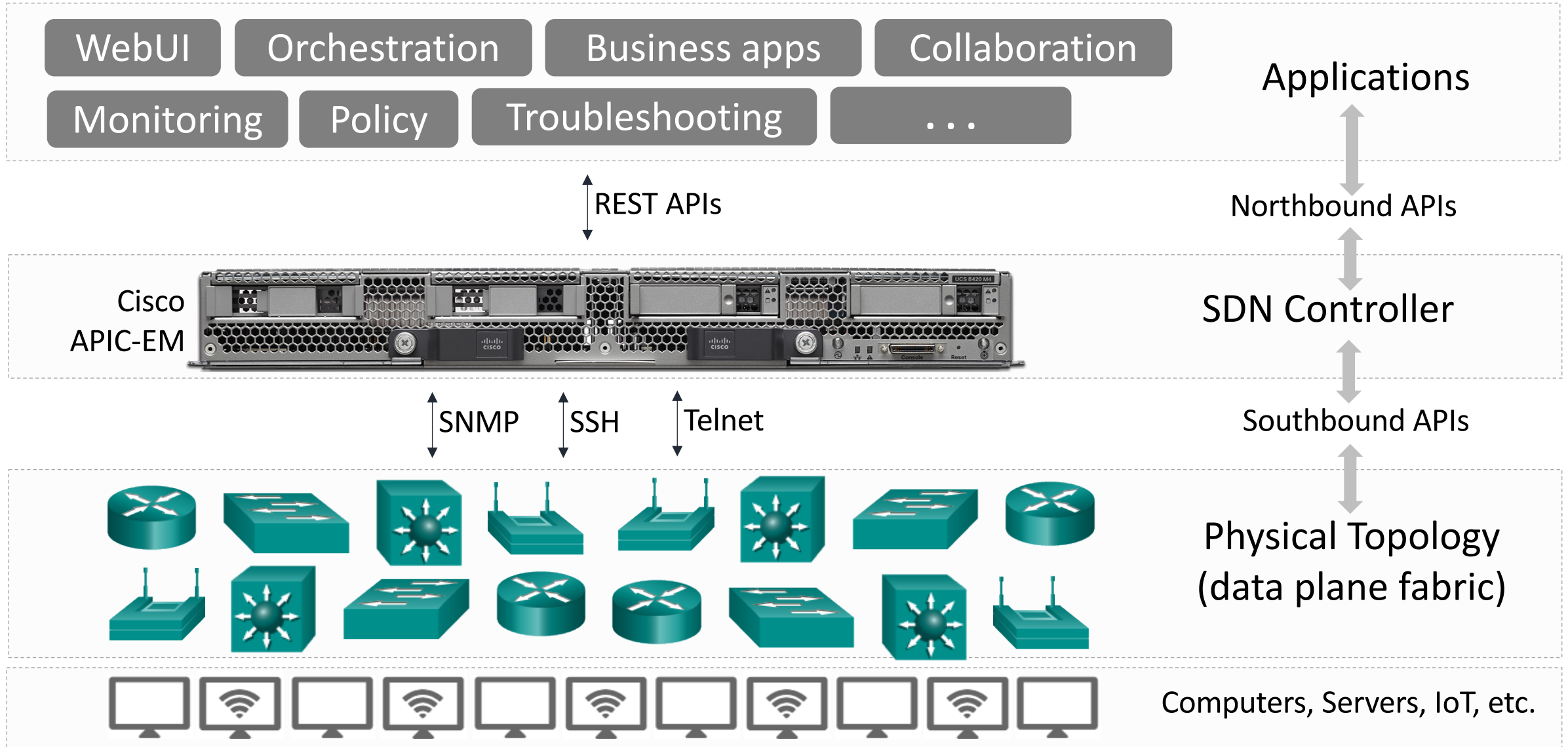


The APIC-EM

Cisco SDN Model with APIC-EM



What is the APIC-EM?



The Cisco Application Policy Infrastructure Controller Enterprise Module (APIC-EM):

- A Software-Defined Networking (SDN) controller for enterprise networks
- A virtual, software-only, or physical appliance
- Creates an intelligent, open, programmable network with open APIs
- Can transform business-intent policies into dynamic network configuration
- Provides a single point for network-wide automation and control

APIC-EM – Log in

Virtualized APIC-EM Controllers are available in several DevNet Sandboxes:

Always On, NetAcad instances

- For NetAcad users only
- <https://DevNetSBX-NetAcad-APICEM-1.cisco.com>
- <https://DevNetSBX-NetAcad-APICEM-2.cisco.com>
- <https://DevNetSBX-NetAcad-APICEM-3.cisco.com>
- **User/PW:** Ask your instructor

Always on, public instance

- For to all DevNet users
- <https://SandBoxAPICEM.cisco.com>
- **User:** *devnetuser* **PW:** *Cisco123!*



APIC-EM – Log in

A virtualized APIC-EM Controller is available in a DevNet Sandbox:

Always on, NetAcad only instances

- For NetAcad users in specific regions.
- <https://DevNetSBX-NetAcad-APICEM-1.cisco.com>
- <https://DevNetSBX-NetAcad-APICEM-2.cisco.com>
- <https://DevNetSBX-NetAcad-APICEM-3.cisco.com>
- **User/PW:** Ask your instructor

Always on, public instance

- For all DevNet users
- <https://SandBoxAPICEM.cisco.com>
- **User:** *devnetuser* **PW:** *Cisco123!*



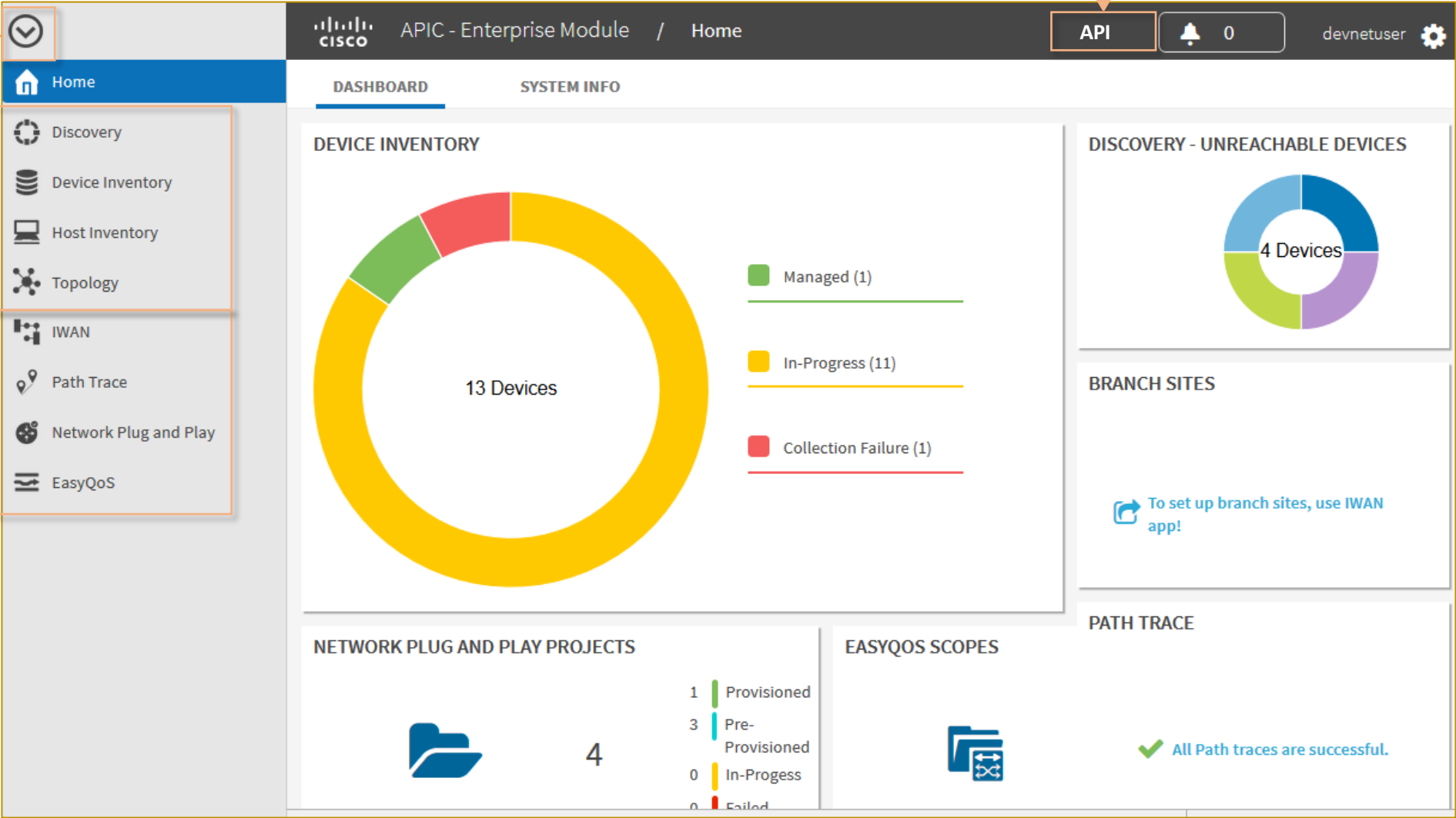
APIC-EM Home Page

Expand Navigation Bar

Services

Applications

API documentation

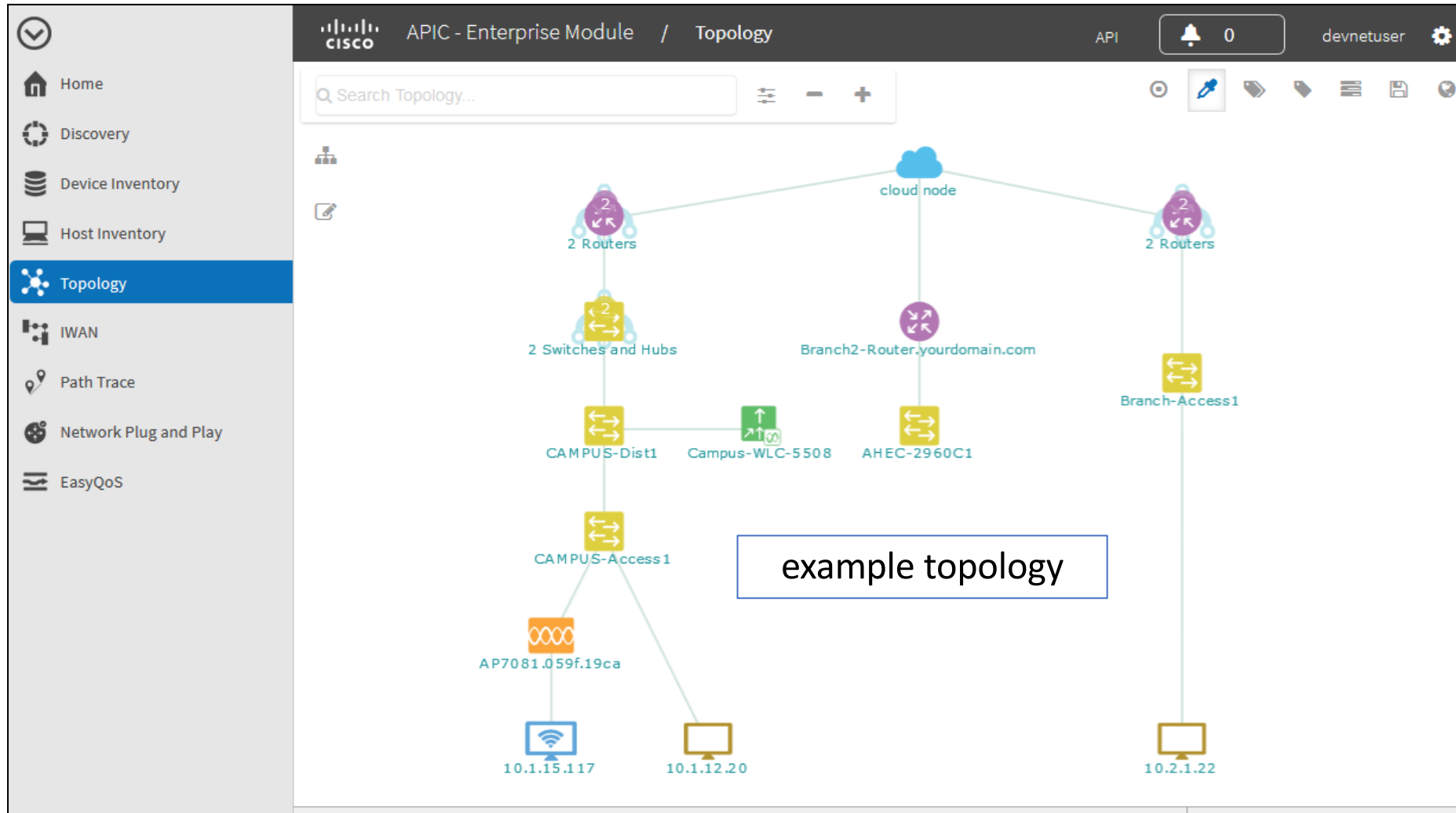


APIC-EM Applications

- **Network Plug-and-Play (PnP)**
Provides a unified approach to provision enterprise networks comprised of Cisco routers, switches, and wireless access points with a near-zero-touch deployment experience.
- **Easy QoS**
Provides a simple way to classify and assign application priority.
- **Intelligent WAN (IWAN)**
Simplifies WAN deployments by providing an intuitive, policy-based interface that helps IT abstract network complexity and design for business intent.
- **Path Trace**
Greatly eases and accelerates the task of connection monitoring and troubleshooting.

More [information](#) on the Cisco APIC-EM.

APIC-EM Topology Page





Programming the APIC-EM REST API

REST APIs

What is so great about REST*?



Easy to use:

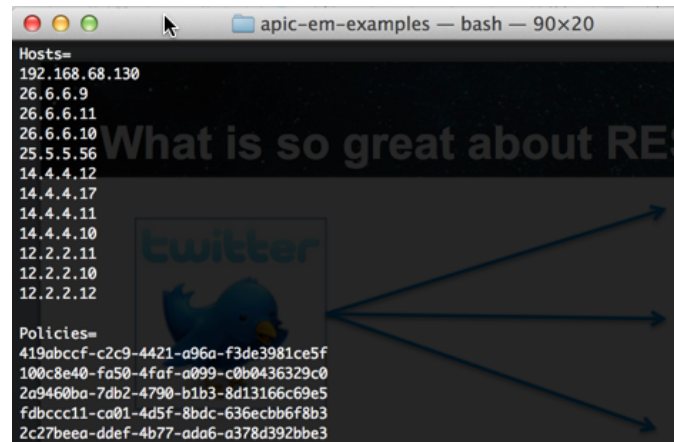
- In mobile apps
- In console apps
- In web apps



Cisco APIC-EM REST APIs

- Hosts
- Devices
- Users
- + more

How does this work?



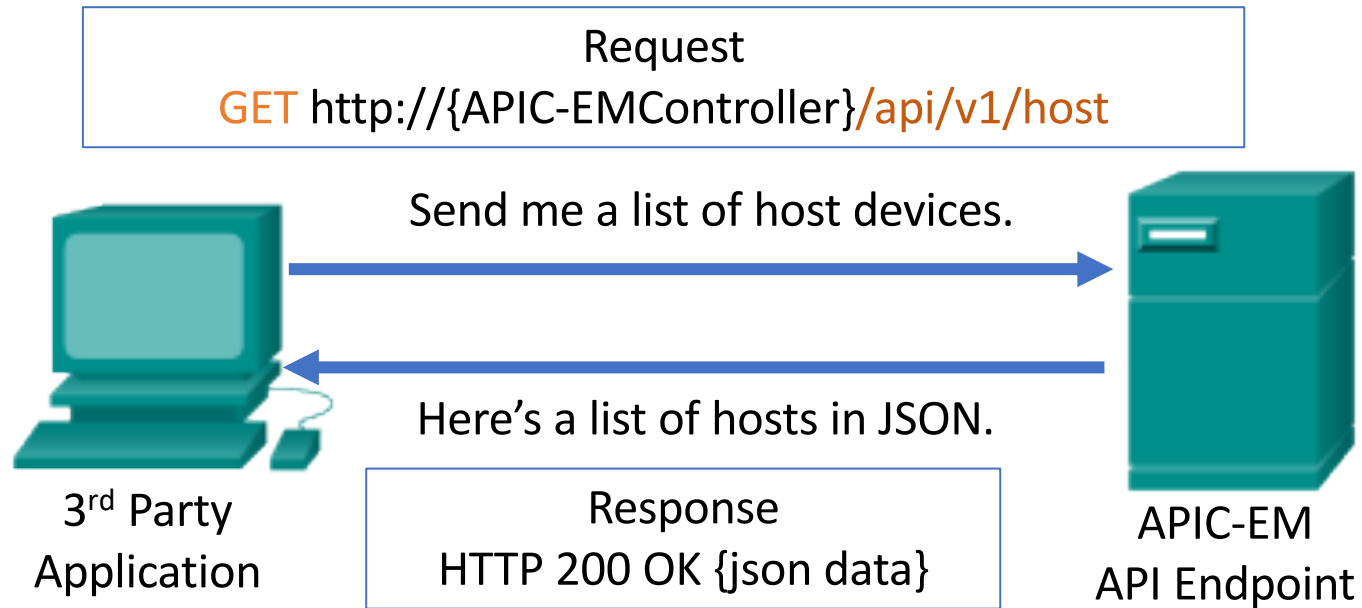
*representational state transfer (REST).

REST APIs

- Use HTTP protocol methods and transport
- API **endpoints** exist as server processes that are accessed through URIs
- Webpages present data and functionality in human-machine interaction driven by a user.
- APIs present data and functionality in machine-machine interactions driven by software.

Directory of Public APIs: <https://www.programmableweb.com/apis/directory>

How does this work?



Anatomy of a REST Request

REST requests require the following elements (requirements may differ depending on the API):

Method

- GET (retrieve), POST (create), PUT (update), DELETE (remove)

URL

- Example: `http://{APIC-EMController}/api/v1/host`

Authentication

- Basic HTTP, OAuth, none, Custom

Custom Headers

- HTTP Headers
- Example: `Content-Type: application/json`

Request Body

- JSON or XML containing data needed to complete request

What is in the Response?

HTTP Status Codes

- <http://www.w3.org/Protocols/HTTP/HTRESP.html>
- 200 OK
- 201 Created
- 401, 403 Authorization error
- 404 Resource not found
- 500 Internal Error

Headers

Body

- JSON
- XML



Example output of a HTTP response in the Postman application

JSON and XML

JSON

```
1 {
2   "response": {
3     "request": {
4       "sourceIP": "10.1.15.117",
5       "destIP": "10.2.1.22",
6       "periodicRefresh": false,
7       "id": "feb8f5c6-56d1-45ec-9a49-bd4afac5c887",
8       "status": "COMPLETED",
9       "createTime": 1506693815419,
10      "lastUpdateTime": 1506693823127
11    },
12    "lastUpdate": "Fri Sep 29 14:03:43 UTC 2017",
13    "networkElementsInfo": [
14      {
15        "id": "48cdeb9b-b412-491e-a80c-7ec5bbe98167",
16        "type": "wireless",
17        "ip": "10.1.15.117",
18        "linkInformationSource": "Switched"
19      },
20      {
21        "id": "cd6d9b24-839b-4d58-adfe-3fdf781e1782",
22        "name": "AP7081.059f.19ca",
23        "type": "Unified AP",
24        "ip": "10.1.14.3",
25        "role": "ACCESS",
26        "linkInformationSource": "Switched",
27        "tunnels": [
28          "CAPWAP Tunnel"
29        ]
30      },
31      {
32        "id": "5b5ea8da-8c23-486a-b95e-7429684d25fc",
33        "name": "CAMPUS-Access1",
34        "type": "Switches and Hubs",
35        "ip": "10.1.12.1",
36        "ingressInterface": {
37          "physicalInterface": {
38            "id": "dd2c47ea-ad19-4a1e-ad0e-82d9deefd61b",
39            "name": "GigabitEthernet1/0/26"
40          }
41        },
42        "egressInterface": {
43          "physicalInterface": {
44            "id": "38c72319-855e-43bc-8458-94f695d435b6",
45            "name": "GigabitEthernet1/0/1"
46          }
47        }
48      }
49    ]
50  }
51 }
```

XML

```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <response>
3   <request>
4     <sourceIP>10.1.15.117</sourceIP>
5     <destIP>10.2.1.22</destIP>
6     <periodicRefresh>false</periodicRefresh>
7     <id>feb8f5c6-56d1-45ec-9a49-bd4afac5c887</id>
8     <status>COMPLETED</status>
9     <createTime>1506693815419</createTime>
10    <lastUpdateTime>1506693823127</lastUpdateTime>
11  </request>
12  <lastUpdate>Fri Sep 29 14:03:43 UTC 2017</lastUpdate>
13  <networkElementsInfo>
14    <id>48cdeb9b-b412-491e-a80c-7ec5bbe98167</id>
15    <type>wireless</type>
16    <ip>10.1.15.117</ip>
17    <linkInformationSource>Switched</linkInformationSource>
18  </networkElementsInfo>
19  <networkElementsInfo>
20    <id>cd6d9b24-839b-4d58-adfe-3fdf781e1782</id>
21    <name>AP7081.059f.19ca</name>
22    <type>Unified AP</type>
23    <ip>10.1.14.3</ip>
24    <role>ACCESS</role>
25    <linkInformationSource>Switched</linkInformationSource>
26    <tunnels>CAPWAP Tunnel</tunnels>
27  </networkElementsInfo>
28  <networkElementsInfo>
29    <id>5b5ea8da-8c23-486a-b95e-7429684d25fc</id>
30    <name>CAMPUS-Access1</name>
31    <type>Switches and Hubs</type>
32    <ip>10.1.12.1</ip>
33    <ingressInterface>
34      <physicalInterface>
35        <id>dd2c47ea-ad19-4a1e-ad0e-82d9deefd61b</id>
36        <name>GigabitEthernet1/0/26</name>
37      </physicalInterface>
38    </ingressInterface>
39    <egressInterface>
40      <physicalInterface>
41        <id>38c72319-855e-43bc-8458-94f695d435b6</id>
42        <name>GigabitEthernet1/0/1</name>
43      </physicalInterface>
44    </egressInterface>
45  </networkElementsInfo>
46 </response>
47
```


What about authentication?

- **None:** The API resource is public and anybody can place the request.
 - **Basic HTTP:** The username and password are passed to the server in an encoded string.
 - **OAuth:** Open standard for HTTP authentication and session management. Creates an access token associated to a specific user that also specifies the user rights.
 - **Token:** A token is created and passed with each API call, but there is no session management and tracking of clients which simplifies interaction between the server and client.
- APIC-EM uses **Token** for authentication management. The APIC-EM calls this token a service ticket.

View the [APIC-EM Swagger Documentation](#)

CISCO APIC - Enterprise Module / Swagger

API 🔔 0 devnetuser ⚙️

Available APIs

- [File](#)
- [Flow Analysis](#)
- [Grouping](#)
- [IP Geolocation](#)
- [IP Pool Manager](#)
- [Identity-Manager](#)
- [Inventory](#)
- [Network Discovery](#)
- [Network Plug and Play](#)
- [PKI Broker Service](#)
- [Policy Administration](#)
- [Role Based Access Control](#)**
- [Scheduler](#)
- [Task](#)
- [Topology](#)
- [Visibility](#)

Role Based Access Control

APIC-EM Service API based on the Swagger™ 1.2 specification

[Terms of service](#)

[Cisco DevNet](#)

		Show/Hide	List Operations	Expand Operations	Raw
aaa : APIs to register and manage AAA Servers					
role : Role Description API					
ticket : Ticket Management API					
POST	/ticket				addTicket
POST	/ticket/attribute				createTicketAttribute
GET	/ticket/attribute/idletimeout				getIdleTimeout
GET	/ticket/attribute/sessiontimeout				getSessionTimeout
DELETE	/ticket/attribute/{attribute}				deleteTicketAttribute
DELETE	/ticket/{ticket}				deleteTicket
user : User Management API					

[BASE URL: https://sandboxapicem.cisco.com/api/v1/api-docs/rbac-service , API VERSION: 1.0]

I wish this page would...

POST /ticket Swagger

Try it out!

1. Click Model Schema
2. Click the yellow box under Model Schema
3. Enter the DevNet Sandbox APIC-EM credentials between the quotes.
4. Click the “Try it out !” button.
5. If successful, the ticket number will be in the response body JSON.

Swagger API

Parameter	Value	Description	Parameter Type	Data Type
user	<pre>{ "password": "Cisco123!", "username": "devnetuser" }</pre> <p>Parameter content type: application/json</p>	user	body	Model Model Schema <pre>{ "password": "", "username": "" }</pre> <p>Click to set as parameter value</p>

Error Status Codes

HTTP Status Code	Reason
200	This Request is OK
202	This Request is Accepted
403	This user is Forbidden Access to this Resource
401	Not Authorized Yet, Credentials to be supplied
404	No Resource Found

Try it out! [Hide Response](#)

Request URL

`https://sandboxapicem.cisco.com/api/v1/ticket`

Response Body

```
{
  "response": {
    "serviceTicket": "ST-9749-ACgWKTbXd37b0jZLQ4wv-cas",
    "idleTimeout": 1800,
    "sessionTimeout": 21600
  },
  "version": "1.0"
}
```

service ticket

response body JSON

Response Code

200