

Task 8: API (Application Programming Interface) SL1

QE: Andrés Felipe Vargas

Objectives:

1. The goal of this task is to get familiar with API EM7 and the available resources
2. Review and understand the following sections
 1. URI Formatting, Request and Response (Result set), API settings, HTTP Status Codes, Response/Request Headers, Response Formats, Filters, Options (Extended fetch)
3. Understand the API response (Http code, Header, and Body)
4. Use Options in the request (e.g.: Extended_fetch, hide_filterinfo, limit, offset, etc)
5. Perform API requests (http methods) to different available resources
 1. Read, Create, Update sl1 resources – Get, Post, Put (eg.Dynamic Apps, devices, templates, device groups, etc)

Applications:

1. Tool: Postman, others

Chapter 1 Introduction sections

1. What is the SL1 API?

This API allows other external systems to access data in SL1, access to entities in SL1 (Tickets, devices, collected data) using standard HTTPS request/response protocols.

These SL1 appliances gives access to the API:

- All-in-one appliances
- Administration portals
- Database Servers

2. Accessing the API

All communication with the API is managed by HTTPS requests.

The following is what a request must include:

- Valid SL1 login credentials: The credentials included in the HTTP request are validated by the stored in the system.

- A Resource URI: Is the URI for the resource (entity) that is getting requested.
- An HTTP Method: Is the action that is going to be performed on the resource.
- An Accept Header: Specifies the format that should be used for the response. The API supports application/xml and application/json formats.
- The base URI of the API: Is the full address of the main API index. This base includes information about the appliance used to access the API.
 - The base URI of the API for DB Servers, Administration Portals and All-in-one appliances is:
 - <https://<ip-address or hostname of appliance>/api>

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

▼<APIFeatures elemtype="list">
  <link URI="/api/access_lock" description="Get/Delete Access Locks" elemtype="href"/>
  <link URI="/api/account" description="Get/Update/Add/Delete User Accounts" elemtype="href"/>
  <link URI="/api/account_policy" description="Get/Update/Add/Delete User Policies" elemtype="href"/>
  <link URI="/api/alert" description="Add Alerts" elemtype="href"/>
  <link URI="/api/appliance" description="Get/Update EM7 Appliances" elemtype="href"/>
  <link URI="/api/asset" description="Get/Update/Add/Delete Asset Records" elemtype="href"/>
  <link URI="/api/cleared_event" description="View Cleared Events" elemtype="href"/>
  <link URI="/api/collection_label" description="Get Collection Label" elemtype="href"/>
  <link URI="/api/collection_label_group" description="Get Collection Label Groups" elemtype="href"/>
  <link URI="/api/collector_group" description="Get/Update/Add/Delete Collector Groups" elemtype="href"/>
  <link URI="/api/contacts" description="Get/Update/Add/Delete External Contacts" elemtype="href"/>
  <link URI="/api/credential" description="Get/Update/Add Credentials" elemtype="href"/>
  <link URI="/api/custom_attribute" description="Get/Update/Add/Delete Custom Attributes for various reso" elemtype="href"/>
  <link URI="/api/dashboard" description="Get/Update/Delete Dashboards" elemtype="href"/>
  <link URI="/api/data_performance" description="Get collected rollup performance data across devices" elemtype="href"/>
  <link URI="/api/data_performance_raw" description="Get collected raw performance data across devices" elemtype="href"/>
  <link URI="/api/device" description="Get/Update/Add/Delete Devices and Get Collected Data" elemtype="href"/>
  <link URI="/api/device_category" description="Get Device Categories" elemtype="href"/>
  <link URI="/api/device_class" description="Get Device Classes" elemtype="href"/>
  <link URI="/api/device_group" description="Get/Update/Add/Delete Device Groups" elemtype="href"/>
  <link URI="/api/device_template" description="Get/Update/Add/Delete Device Templates" elemtype="href"/>
  <link URI="/api/discovery_session" description="Get/Update/Add/Delete Device Discovery Sessions" elemtype="href"/>
  <link URI="/api/discovery_session_active" description="View/Start/Stop Active Device Discovery Sessions" elemtype="href"/>
  <link URI="/api/dynamic_app" description="Get Dynamic Application Resources" elemtype="href"/>
  <link URI="/api/environment_aligned_silo_library" description="ScienceLogic Library alignment with Exec" elemtype="href"/>
  <link URI="/api/event" description="View/Update/Clear Events" elemtype="href"/>

```

- The base URI of the API for Integration Servers is:
 - <https://<ip-address or hostname of appliance>>

The following is what a response must include:

- An HTTP Status Code: Is the result of the request.
- SL1 Specific Status Headers: Is the additional information about the result of a request.
- XML or JSON data: Is the format that can be specified in the request and the information about the requested resource.

3. API Settings

The SL1 API settings can be configured in the following path:

System > Settings > API

REST API Settings

Authentication/Access Control

Internal Request Account	[em7admin]	▼
X-EM7-run-as Header Support	[Enabled (Admin only)]	▼

Ticket Resource Behavior

Logging	[Normal (Ticket and System Logs)]	▼
X-EM7-suppress-logging Header Support	[Disabled]	▼
Send Notification	[Only if X-EM7-send-notification:1 is sent]	▼

Chapter 2

1. HTTP Methods

Are used to perform operations on API resources and can be used in the requests. On each resource can be used certain of the four methods to make requests.

GET

The response to a GET includes information about the resource requested.

POST

Updates or creates a new resource.

PUT

Adds or replace a resource. Using this method, the entire information of any resource is replaced.

DELETE

Remove resources if the method is allowed to be requested.

2. HTTP Status Codes

The SL1 API uses standard HTTP status codes to display the results of a request.

200 OK. Valid transaction and executed normally.

201 Created. A new resource was created.

202 Accepted. The request was accepted for processing.

204 No Content Returned. Request successful without content. Typical response when a file is uploaded via a PUT request.

301 Moved Permanently. The request URI does not match a sub-resource associated with the main resource.

400 Bad Request. Bad syntax or missing required fields.

401 Unauthorized. Invalid credentials provided for authentication.

403 Forbidden. Valid authentication credentials but the user has not permissions to access the resource.

404 Not Found. There is no resource at the URI specified in the request.

405 Method Not Allowed. The method used in the request is not permitted with the specified resource.

3. Request Headers.

The following headers can be used when making a request:

- **X-em7-beautify-response.**
- **X-em7-run-as**
- **X-em7-suppress-logging**
- **X-em7-send-notification.**

4. Response Headers

Every response from the API includes headers that gives additional information about the result of a request, listed below:

- **X-EM7-Implemented-methods.**
- **X-EM7-Applicable-resources.**
- **X-EM7-authenticated-user.**
- **X-EM7-status-code.**
- **X-EM7-status-message.**
- **X-EM7-Last-updated.**

5. Response formats

The responses are displayed as XML or JSON formats.

Accept headers:

accept: application/json, */*. the API will respond with JSON as the default response format

accept: application/xml, */*. The API will respond in XML format.

The request must include this accept header, if not is going to be a "406 Method Not Acceptable" status code.

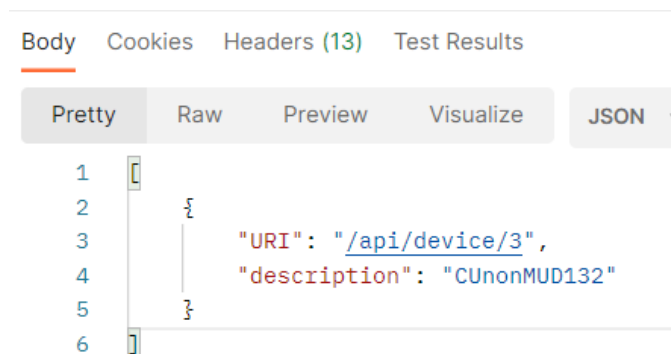
Chapter 3

1. Constructing URIs Using a searchspec

A searchspec indicates the filters and options that can be added to a resource URI to limit or change the response. These filters and options can be added as below indicates:

<resource uri>?<option 1>&<option 2>&<filter 1>&<filter 2>

http://192.168.10.130/api/device?hide_filterinfo=1&limit=1&filter.0._id.eq=3



Any number of options and filters can be added to the URI after the question mark (?), delimited by ampersands (&).

1.1 Filters

The results can be filtered using any of the fields contained in the "fields" section of the searchspec.

The following operators can be added before the equals sign (=) to perform different filters:

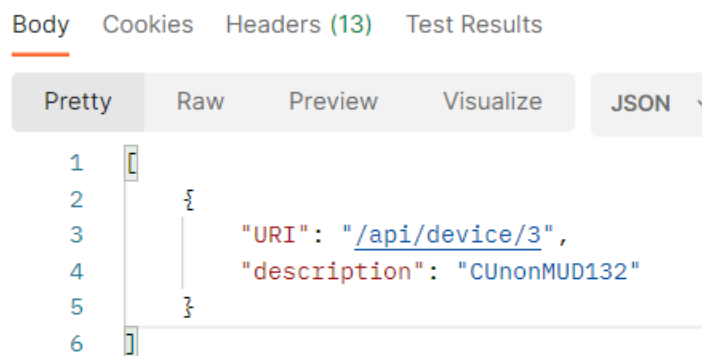
- .min -> Equivalent to a "greater than or equal to" operation.
- .max. -> Equivalent to a "less than or equal to" operation.
- .contains. -> The field contains the specified value as a sub-string.
- .begins_with -> The field begins with the specified value as a sub-string.
- .ends_with. -> The field ends with the specified value as a sub-string.
- .isnull
- .in

1.2 Options

The resource index has a set of options for adding to a request URI to limit or change the response.

Options that are available on the resources:

- **hide_filterinfo.** The response will have only the result set if these options is set as 1 in the URI.
- http://192.168.10.130/api/device?hide_filterinfo=1&limit=1&filter.0.id.eq=7



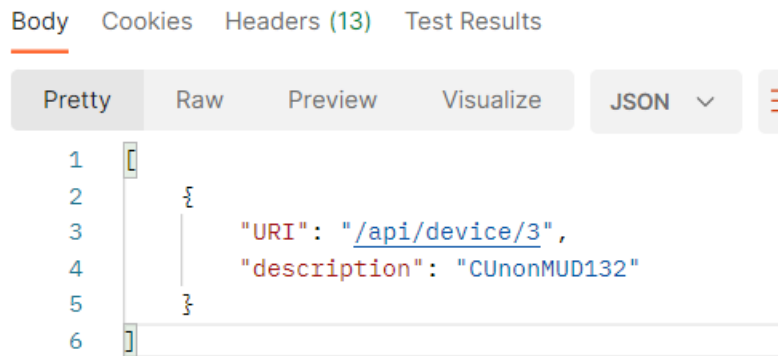
- **hide_filterinfo** set as 0 we are going to get the searchspec information
- http://192.168.10.130/api/device?hide_filterinfo=0&limit=1&filter.0.id.eq=7

```

"options": {
  "hide_filterinfo": {
    "type": "boolean",
    "description": "Suppress filterspec and current filter info if 1 (true)",
    "default": "0"
  },
  "limit": {
    "type": "int",
    "description": "Number of records to retrieve",
    "default": "100"
  },
  "offset": {
    "type": "int",
    "description": "Specifies the index of the first returned resource within the entire result set",
    "default": "0"
  },
  "extended_fetch": {
    "type": "boolean",
    "description": "Fetch entire resource if 1 (true), or resource link only if 0 (false)",
    "default": "0"
  },
  "link_disp_field": {

```

- **limit.** The maximum number of resources that should be returned in the response.
- http://192.168.10.130/api/device/?hide_filterinfo=1&limit=1
- The limit is set as 1, there are two devices but only one is going to be on the response and is the first one.



```

Body  Cookies  Headers (13)  Test Results
Pretty  Raw  Preview  Visualize  JSON v
[
  {
    "URI": "/api/device/3",
    "description": "CUnonMUD132"
  }
]

```

-
- **Offset.** Determines which resource will be the first entry in the response list. Offset begins at zero for the first resource, one for the second and so on.
- http://192.168.10.130/api/device?hide_filterinfo=1&offset=1

Body Cookies Headers (13) Test Results

Pretty

Raw

Preview

Visualize

JSON



```
1 [
2   {
3     "URI": "/api/device/7",
4     "description": "AIOnonMUD130"
5   }
6 ]
```

SL1 API Requests

GET SNMP Credentials

GET



http://192.168.10.130/api/credential/snmp?hide_filterinfo=1

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

Query Params

	Key	Value	Description
<input checked="" type="checkbox"/>	hide_filterinfo	1	
	Key	Value	Description

Body Cookies Headers (13) Test Results

Status: 200 OK

Pretty

Raw

Preview

Visualize

JSON



```
1 [
2   {
3     "URI": "/api/credential/snmp/7",
4     "description": "LifeSize: Endpoint SNMP"
5   },
6   {
7     "URI": "/api/credential/snmp/11",
8     "description": "Cisco: CSP SNMP Port 161 Example"
9   },
10  {
11    "URI": "/api/credential/snmp/12",
12    "description": "Cisco: CSP SNMP Port 1610 Example"
13  },
14  {
15    "URI": "/api/credential/snmp/38",
16    "description": "Dell EMC: Isilon SNMPv2 Example"
17  },
18 ]
```


POST Device group

Device group name before making a POST request. "ACIVms"

Request

POST ▼ http://192.168.10.130/api/device_group/6

Params Authorization ● Headers (9) **Body ●** Pre-request Script Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL **JSON ▼**

```
1 {  
2   ...  
3   "name": "ACIVmsNewName"  
4   ...  
5 }
```

Response

Body Cookies Headers (13) Test Results 🌐 Status: 200 OK

Pretty Raw Preview Visualize JSON ▼ ⌵

```
1 {  
2   "guid": "72DF0CB657B582F976BD49CACE833DE8",  
3   "ppguid": "92BA2902165FFD866C22FA3D83DA30AE",  
4   "shared": "1",  
5   "name": "ACIVmsNewName",  
6   "icon": "_generic_cloud.png",  
7   "force_child_visibility": "1",  
8   "owner": "/api/account/1",  
9   "updated_by": "/api/account/1",  
10  "edit_date": "1689826483",  
}
```

Validation in the SL1 UI

Device Groups Groups Found [9]			
Device Group Name ▼		ID	
1. +	ACIVmsNewName	6	em7admin
2. +	AWS EBS Volumes	2	em7admin
3. +	AWS EC2 Instances	1	em7admin

Create a new device group using a POST request

Request

POST

▼

http://192.168.10.130/api/device_group

Params Authorization Headers (9) Body Pre-request Script Tests Settings

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON ▼

```
1 {
2   ... "guid": "923548359A882A6406D95E29183C857D",
3   ... "ppguid": "8B5E8402EEA09F06E8278BA889A1D510",
4   ... "shared": "1",
5   ... "name": "New device group Test",
6   ... "icon": "azure_managed_disk.png",
7   ... "force_child_visibility": "1",
8   ... "owner": "/api/account/1",
9   ... "updated_by": "/api/account/1",
10  ... "edit_date": "1688438211",
11  ... "devices": [],
12  ... "groups": [],
13  ... "visibility": [
14    ... "Notification/Automation"
15  ],
16  ... "parents": [],
17  ... "permission_keys": []
18  ...
19  ...
20 }
```

Response

Body Cookies Headers (14) Test Results

🌐 Status: 201 Created

Pretty Raw Preview Visualize JSON ▼

```
1 {
2   "guid": "923548359A882A6406D95E29183C857D",
3   "ppguid": "8B5E8402EEA09F06E8278BA889A1D510",
4   "shared": "1",
5   "name": "New device group Test",
6   "icon": "azure_managed_disk.png",
7   "force_child_visibility": "1",
8   "owner": "/api/account/1",
9   "updated_by": "/api/account/1",
10  "edit_date": "1688438211",
11  "devices": [],
12  "groups": [],
13  "visibility": [
14    "Notification/Automation"
15  ],
16  "parents": [],
17  "permission_keys": [],
18  "groupCount": 0,
19  "parentCount": 0,
20  "deviceCount": 0
21 }
```

Validation in the SL1 UI

Device Groups | Groups Found [10]

	Device Group Name ▾	ID	Owner
1. +	ACIVmsNewName	6	em7admin
2. +	AWS EBS Volumes	2	em7admin
3. +	AWS EC2 Instances	1	em7admin
4. +	LayerX Appliances	7	em7admin
5. +	Microsoft Azure: Storage Disks	5	em7admin
6. +	Microsoft Azure: Virtual Machines	4	em7admin
7. +	New device group Test	10	em7admin
8. +	Microsoft Azure: Storage Disks	6	em7admin

Remove a device group using the DELETE request

Request

DELETE ▾ http://192.168.10.130/api/device_group/10

Params **Authorization** ● Headers (7) Body Pre-request Script Tests Settings

Type Basic Auth ▾

The authorization header will be automatically generated when you send the request. Learn more about [authorization](#) ↗

ⓘ Heads up! These parameters hold sensitive data. To keep this data secure while working in a colla recommend using variables. Learn more about [variables](#) ↗

Username

Password ⚠

Response

Body

Cookies

Headers (13)

Test Results

Status: 200 OK

Pretty

Raw

Preview

Visualize

JSON

1

Validation in the SL1 UI

Device Groups Groups Found [9]				
	Device Group Name	ID	Owner	Access
1. +	ACIVmsNewName	6	em7admin	Shared
2. +	AWS EBS Volumes	2	em7admin	Private
3. +	AWS EC2 Instances	1	em7admin	Private
4. +	LayerX Appliances	7	em7admin	Private
5. +	Microsoft Azure: Storage Disks	5	em7admin	Private
6. +	Microsoft Azure: Virtual Machines	4	em7admin	Private
7. +	Nutanix Clusters	9	em7admin	Shared
8. +	Servers	3	em7admin	Shared
9. +	SL1 Appliances	8	em7admin	Private