



API

- [Using Rest APIs](#) , on page 1

Using Rest APIs

You can use REST APIs to retrieve, add, or modify information on Cisco Spaces: Detect and Locate. The REST APIs are divided into five categories:

- **Active clients' location APIs:** APIs to retrieve client count and location data.
- **Clients location history APIs:** APIs to get a MAC address and the details for a given device.
- **Notifications APIs:** APIs for subscription-based notifications.
- **Map APIs:** APIs to upload, navigate the maps hierarchy, retrieve, and delete a map element.
- **Access point APIs:** APIs to get access point details.

API Key

To use REST APIs, you must generate an API Key. An API key is a Cisco-proprietary JSON Web Token (JWT) that is required in each HTTP request header to authenticate and authorize users.

You can generate an API Key from Cisco Spaces: Detect and Locate. Navigate to **Notifications > API Keys** and then click **Add**. You are prompted to configure the number of days after which the key should expire. Valid range is between 7 days and 365 days. After the key is generated, ensure that it is stored safely.

Figure 1: API Keys

Key	Create Time	Expire Time	User	Actions
.....FgOk	Apr 22nd, 2021 08:57:55 AM	Jul 1st, 2021 08:57:54 AM	user1@example.com	...
.....ew3g	Apr 22nd, 2021 08:55:41 AM	Apr 29th, 2021 08:55:40 AM	user2@example.com	...

The **API Keys** window shows the key names (only partially displayed), the date and time at which they were created, the date and time at which they are going to expire, and email IDs of the users who created the keys. To delete a key, click on the three dots icon in the **Actions** column and then click **Delete**. If you delete a key, the key is not revoked and you can still use it until its expiry date and time.

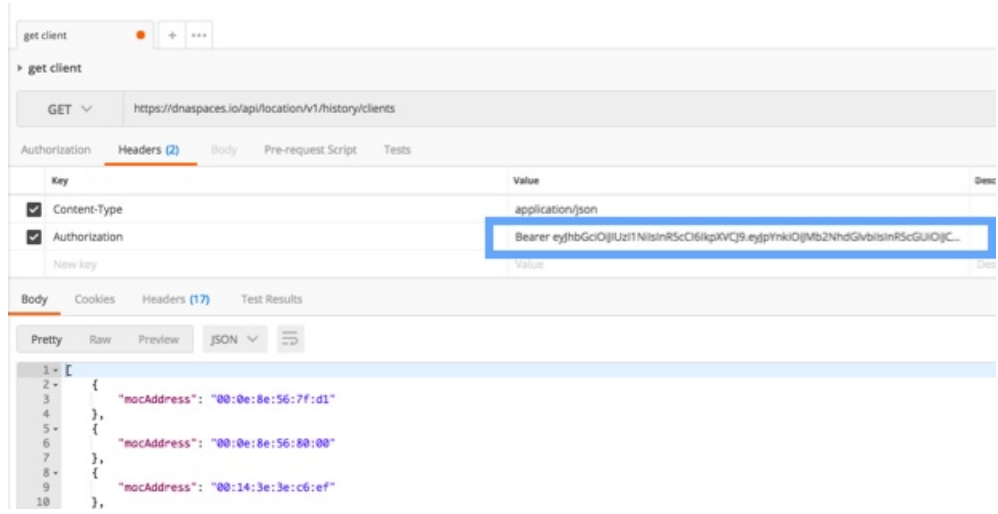


Note The API key is visible only at creation time, and hence must be stored securely. Cisco Spaces: Detect and Locate does not save the API key values. Each authenticated user can have up to five keys.

Figure 2: Copy the API Key

The following is an example from the POSTMAN client, where an API key has been used as an **Authorization header**.

Figure 3: API Keys



The screenshot displays a REST client interface for a GET request to `https://dnspaces.io/api/location/v1/history/clients`. The request headers are configured as follows:

Key	Value
Content-Type	application/json
Authorization	Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpbnkiOiJMb2NhZGlvbGlzInR5cGU6IjE2In0.eyJpbnkiOiJMb2NhZGlvbGlzInR5cGU6IjE2In0.

The response body is shown in JSON format, containing an array of three objects:

```
1 - [
2 - {
3 -   "macAddress": "00:0e:8e:56:7f:d1"
4 - },
5 - {
6 -   "macAddress": "00:0e:8e:56:80:00"
7 - },
8 - {
9 -   "macAddress": "00:14:3e:3e:c6:ef"
10 - },
```

