Create custom devices through the REST API

Published: 2024-11-02

You can create custom devices through the REST API that track network traffic across multiple IP addresses and ports. For example, you might want to add a custom device for each branch office. If you create the devices through a script, you can read the list of devices from a CSV file. In this topic, we will demonstrate methods for both the REST API and the ExtraHop REST API Explorer.

Before you begin

- You must log in to the sensor with an account that has system and access administration privileges to generate an API key.
- You must have a valid API key to make changes through the REST API and complete the procedures below. (See Generate an API key.)
- Familiarize yourself with the ExtraHop REST API Guide to learn how to navigate the ExtraHop REST API Explorer.

Create a custom device through the REST API Explorer

You can create a custom device and associate the custom device with a list of IP addresses or CIDR blocks through the **POST /customdevices** operation.

1. In a browser, navigate to the REST API Explorer.

   The URL is the hostname or IP address of your sensor, followed by `/api/v1/explore/`. For example, if your hostname is seattle-eda, the URL is `https://seattle-eda/api/v1/explore/`.
2. Click **Custom Device**, and then click **POST /customdevices**.
3. In the body field, specify properties for the custom device that you want to create.

   For example, the following body matches the custom device to the CIDR blocks 192.168.0.0/26, 192.168.0.64/27, 192.168.0.96/30, and 192.168.0.100/32:

```
{
  "description": "The location of our office in Washington",
  "name": "Seattle",
  "criteria": [
    {
      "ipaddr": "192.168.0.0/26"
    },
    {
      "ipaddr": "192.168.0.64/27"
    },
    {
      "ipaddr": "192.168.0.96/30"
    },
    {
      "ipaddr": "192.168.0.100/32"
    }
  ]
}
```

Retrieve and run the example Python script

The ExtraHop GitHub repository contains an example Python script that creates custom devices by reading criteria from a CSV file.

1. Go to the ExtraHop code-examples GitHub repository and download the `create_custom_devices/create_custom_devices.py` file to your local machine.
2. Create a CSV file with rows that contain the following columns in the specified order:

Name	ID	Description	IP address or CIDR block

Tip: The `create_custom_devices` directory contains an example CSV file named `device_list.csv`.

The script does not accept a header row in the CSV file. There is no limit to the number of columns in the table; each column after the first four specifies an additional IP address for the device. The first four columns are required for each row.

3. In a text editor, open the `create_custom_devices.py` file and replace the following configuration variables with information from your environment:
   - **HOST:** The IP address or hostname of the sensor.
   - **APIKEY:** The API key.
   - **CSV_FILE:** The path of the CSV file relative to the location of the script file.
4. Run the following command:

```
python3 create_custom_devices.py
```

Note: If the script returns an error message that the TLS certificate verification failed, make sure that a trusted certificate has been added to your sensor or console. Alternatively, you can add the `verify=False` option to bypass certificate verification. However, this method is not secure and is not recommended. The following code sends an HTTP GET request without certificate verification:

```
requests.get(url, headers=headers, verify=False)
```