Suchen Sie über die REST-API nach einem Gerät

Veröffentlicht: 2024-04-10

Sie können alle erkannten Geräte auf Ihrem durchsuchen Sensor oder Konsole indem Sie Ihre Kriterien (wie IP-Adresse oder Discovery-ID) angeben und dann die Geräteliste und die zugehörigen Metadaten in ein Dateiformat exportieren, das mit einer Drittanbieteranwendung wie Microsoft Excel oder einem beliebigen CSV-Reader lesbar ist. Beispielsweise möchten Sie möglicherweise die IP-Adressen aller VMware-Geräte in Ihrem Netzwerk anzeigen und exportieren.

Sie können Gerätesuchabfragen testen, bevor Sie sie in ein Skript integrieren, indem Sie die Abfragen im ExtraHop REST API Explorer ausführen. Dieses Handbuch enthält Methoden sowohl für den REST API Explorer als auch ein Python-Beispielskript.

Bevor Sie beginnen

- Für Sensoren und ECA-VMs benötigen Sie einen gültigen API-Schlüssel, um Änderungen über die REST-API vornehmen und die folgenden Verfahren ausführen zu können. (siehe Generieren Sie einen API-Schlüssel.)
- Für Reveal (x) 360 benötigen Sie gültige REST-API-Anmeldeinformationen, um Änderungen über die REST-API vornehmen und die folgenden Verfahren ausführen zu können. (siehe REST-API-Anmeldeinformationen erstellen.)

Suchen Sie über den REST API Explorer nach einem Gerät

1. Navigieren Sie in einem Browser zum REST API Explorer.

   Die URL ist der Hostname oder die IP-Adresse Ihres Sensor oder Konsole, gefolgt von `/api/v1/explore/`. Wenn Ihr Hostname beispielsweise seattle-eda ist, lautet die URL `https://seattle-eda/api/v1/explore/`.
2. Klicken Sie **API-Schlüssel eingeben** und fügen Sie dann Ihren API-Schlüssel ein oder geben Sie ihn in das API-Schlüssel Feld.
3. klicken **Autorisieren** und klicken Sie dann **Schliessen**.
4. klicken **Gerät** um Geräteoperationen anzuzeigen.
5. klicken **POST /Geräte/Suche**.
6. klicken **Probiere es aus**.

   Das JSON-Schema wird automatisch zum Textfeld für den Body-Parameter hinzugefügt.
7. Geben Sie in das Textfeld Ihre Suchkriterien ein.
   Die folgenden Suchkriterien geben ein Gerät mit der IP-Adresse 10.10.10.200 zurück:

```
{
  "filter": {
    "field": "ipaddr",
    "operand": "10.10.10.200",
    "operator": "="
  }
}
```

   Weitere Informationen zu Gerätesuchfiltern finden Sie unter Operandenwerte für die Gerätesuche.

Rufen Sie das Python-Beispielskript ab und führen Sie es aus

Das ExtraHop GitHub-Repository enthält ein Python-Skript, das anhand der IP-Adresse nach einer Liste von Geräten sucht. Das Skript gibt dann die ExtraHop Discovery-ID für jede IP-Adresse aus.

1. Gehe zum ExtraHop Codebeispiele GitHub-Repository und laden Sie das herunter `search_device/search_device.py` Datei auf Ihrem lokalen Computer.
2. Öffnen Sie in einem Texteditor den `search_device.py` archivieren und ersetzen Sie die Konfigurationsvariablen durch Informationen aus Ihrer Umgebung.
   - Geben Sie für Sensoren und ECA-VMs die folgenden Konfigurationsvariablen an:
     - **GASTGEBER:** Die IP-Adresse oder der Hostname des Sensor oder der ECA-VM.
     - **API-SCHLÜSSEL:** Der API-Schlüssel.
     - **IP_ADDR_LIST:** Eine Reihe von IP-Adressen.
   - Geben Sie für Reveal (x) 360 die folgenden Konfigurationsvariablen an:
     - **GASTGEBER:** Der Hostname der Reveal (x) 360-API. Dieser Hostname wird auf der Reveal (x) 360 API Access-Seite unter API-Endpunkt angezeigt. Der Hostname beinhaltet nicht `/oauth2/token`.
     - **ID:** Die ID der Reveal (x) 360-REST-API-Anmeldeinformationen.
     - **GEHEIM:** Das Geheimnis der Reveal (x) 360 REST-API-Anmeldeinformationen.
     - **IP_ADDR_LIST:** Eine Reihe von IP-Adressen.
3. Führen Sie den folgenden Befehl aus:

```
python3 search_device.py
```

Hinweis: Wenn das Skript eine Fehlermeldung zurückgibt, dass die SSL-Zertifikatsüberprüfung fehlgeschlagen ist, stellen Sie sicher, dass Ihrem Sensor oder Ihrer Konsole wurde ein vertrauenswürdiges Zertifikat hinzugefügt. Alternativ können Sie das hinzufügen `verify=False` Option zur Umgehung der Zertifikatsüberprüfung. Diese Methode ist jedoch nicht sicher und wird nicht empfohlen. Der folgende Code sendet eine HTTP GET-Anfrage ohne Zertifikatsüberprüfung:

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