Extrahieren Sie die Geräteliste über die REST-API

Veröffentlicht: 2024-03-20

Die ExtraHop REST API ermöglicht es Ihnen, die Liste der Geräte zu extrahieren, die von Sensor oder Konsole. Durch Extrahieren der Liste mit einem REST-API-Skript können Sie die Liste in einem Format exportieren, das von Drittanbieteranwendungen gelesen werden kann, z. B. einer Configuration Management Datenbank (CMDB). In diesem Thema zeigen wir Methoden zum Extrahieren einer Liste sowohl mit dem cURL-Befehl als auch mit einem Python-Skript.

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.)

Rufen Sie die Geräteliste mit dem cURL-Befehl ab

Die Geräteliste enthält alle Gerätemetadaten wie MAC-Adressen und Geräte-IDs. Sie können die Geräteliste jedoch mit einem JSON-Parser filtern, um die spezifischen Informationen zu extrahieren, die Sie exportieren möchten. In diesem Beispiel wird die Geräteliste abgerufen und dann mit dem jq-Parser gefiltert, um nur den Anzeigenamen der einzelnen Gerät zu extrahieren.

Hinweis: Das folgende Verfahren ist nicht mit der Reveal (x) 360 REST-API kompatibel. Informationen zum Abrufen der Geräteliste von Reveal (x) 360 finden Sie unter Rufen Sie die Geräteliste von Reveal (x) 360 mit dem Befehl cURL ab.

Bevor Sie beginnen

- Das cURL-Tool muss auf Ihrem Computer installiert sein.
- Der jq-Parser muss auf Ihrem Computer installiert sein. Weitere Informationen finden Sie unter https://stedolan.github.io/jq/.

Öffnen Sie eine Terminalanwendung und führen Sie den folgenden Befehl aus, wobei `YOUR_KEY` ist die API für Ihr Benutzerkonto, `HOSTNAME` ist der Hostname Ihres Sensor oder Ihrer Konsole und `MAX_DEVICES` ist eine Zahl, die groß genug ist, um mehr als die Gesamtzahl der von Ihrem System erkannten Geräte zu sein:

```
curl -s -X POST "https://HOSTNAME/api/v1/devices/search" --header
 "accept: application/json" --header "Authorization: ExtraHop
 apikey=YOUR_KEY" --header "Content-Type: application/json" -d
 "{ \"active_from\": 1, \"active_until\": 0, \"limit\": MAX_DEVICES}" |
 jq -r '.[] | .display_name'
```

Hinweis: Wenn der Befehl keine Ergebnisse zurückgibt, stellen Sie sicher, dass Ihrem ExtraHop-System wurde ein vertrauenswürdiges Zertifikat hinzugefügt. Alternativ können Sie das hinzufügen `--insecure` Option zum Abrufen der Geräteliste von einem ExtraHop-System ohne vertrauenswürdiges Zertifikat; diese Methode ist jedoch nicht sicher und wird nicht empfohlen.

Hinweis: Sie können das anhängen `select(.analysis == "LEVEL")` Option zum Filtern der Ergebnisse nach Analyseebene. Beispielsweise schränkt der folgende Befehl

die Ergebnisse so ein, dass sie nur Geräte enthalten, die für die erweiterte Analyse ausgewählt wurden:

```
curl -s -X POST "https://HOSTNAME/api/v1/devices/search" --
header "accept: application/json" --header "Authorization:
 ExtraHop apikey=YOUR_KEY" --header "Content-Type: application/
json" -d "{ \"active_from\": 1, \"active_until\": 0, \"limit\":
 1000000000}" | jq -r '.[] | select(.analysis == "advanced")
 | .display_name'
```

Hinweis: Sie können das anhängen `select(.critical == BOOLEAN)` Option zum Filtern von Ergebnissen nach dem kritischen Feld. Beispielsweise schränkt der folgende Befehl die Ergebnisse so ein, dass nur Geräte berücksichtigt werden, die vom ExtraHop-System als kritisch eingestuft wurden:

```
curl -s -X POST "https://HOSTNAME/api/v1/devices/search" --
header "accept: application/json" --header "Authorization:
 ExtraHop apikey=YOUR_KEY" --header "Content-Type: application/
json" -d "{ \"active_from\": 1, \"active_until\": 0, \"limit
\": 1000000000}" | jq -r '.[] | select(.critical == true)
 | .display_name'
```

Hinweis: Sie können das anhängen `select(.cloud_instance_name != null)` Option zum Filtern von Ergebnissen nach dem Feld Cloud-Instanzname. Beispielsweise schränkt der folgende Befehl die Ergebnisse so ein, dass sie nur Geräte mit einem Cloud-Instanznamen enthalten:

```
curl -s -X POST "https://HOSTNAME/api/v1/devices/search" --
header "accept: application/json" --header "Authorization:
 ExtraHop apikey=YOUR_KEY" --header "Content-Type: application/
json" -d "{ \"active_from\": 1, \"active_until\": 0, \"limit
\": 1000000000}" | jq -r '.[] | select(.cloud_instance_name !=
 null) | .cloud_instance_name'
```

Rufen Sie die Geräteliste von Reveal (x) 360 mit dem Befehl cURL ab

Die Geräteliste enthält alle Geräte-Metadaten wie MAC-Adressen und Geräte-IDs. Sie können die Geräteliste jedoch mit einem JSON-Parser filtern, um die spezifischen Informationen zu extrahieren, die Sie exportieren möchten. In diesem Beispiel wird die Geräteliste abgerufen und dann mit dem jq-Parser gefiltert, um nur den Anzeigenamen jedes Gerät zu extrahieren.

Hinweis: Das folgende Verfahren ist nur mit der Reveal (x) 360-REST-API kompatibel. Informationen zum Abrufen der Geräteliste von Sensoren und ECA-VMs finden Sie unter Rufen Sie die Geräteliste mit dem cURL-Befehl ab.

Bevor Sie beginnen

- Das cURL-Tool muss auf Ihrem Computer installiert sein.
- Der JQ-Parser muss auf Ihrem Computer installiert sein. Weitere Informationen finden Sie unter https://stedolan.github.io/jq/.

1. Öffnen Sie eine Terminal-Anwendung und führen Sie den folgenden Befehl aus, wobei `REVEAL_X_360_REST_API` ist der Hostname der Reveal (x) 360-API. Dieser Hostname wird angezeigt in Reveal (x) 360 auf dem API-Zugriff Seite unter API-Endpunkt. Der Hostname beinhaltet nicht `/oauth2/token`:

```
HOST="https://REVEAL_X_360_REST_API"
```

2. Führen Sie den folgenden Befehl aus, wobei `YOUR_ID` ist die ID der REST-API-Anmeldeinformationen:

```
ID="YOUR_ID"
```

3. Führen Sie den folgenden Befehl aus, wobei `YOUR_SECRET` ist das Geheimnis der REST-API-Anmeldeinformationen:

```
SECRET="YOUR_SECRET"
```

4. Führen Sie den folgenden Befehl aus:

```
AUTH=$(printf "$ID:$SECRET" | base64 --wrap=0)
```

5. Führen Sie den folgenden Befehl aus:

```
ACCESS_TOKEN=$(curl -s \
    -H "Authorization: Basic ${AUTH}" \
    -H "Content-Type: application/x-www-form-urlencoded" \
    --request POST \
     ${HOST}/oauth2/token \
    -d "grant_type=client_credentials" \
    | jq -r '.access_token')
```

6. Führen Sie den folgenden Befehl aus, wobei `MAX_DEVICES` ist eine Zahl, die groß genug ist, um mehr als die Gesamtzahl der von Ihrem System erkannten Geräte zu sein:

```
curl -s -X GET -H "Authorization: Bearer ${ACCESS_TOKEN}"  "$HOST/api/
v1/devices?active_from=1&active_until=0&limit=MAX_DEVICES" | jq -r '.[]
 | .display_name'
```

Hinweis: Sie können das anhängen `select(.analysis == "LEVEL")` Option zum Filtern der Ergebnisse nach Analyseebene. Beispielsweise schränkt der folgende Befehl die Ergebnisse so ein, dass nur Geräte berücksichtigt werden, die für die erweiterte Analyse ausgewählt wurden:

```
curl -s -X GET -H "Authorization: Bearer
 ${ACCESS_TOKEN}"  "$HOST/api/v1/devices?
active_from=1&active_until=0&limit=10000000000" | jq -r '.[] |
 select(.analysis == "advanced") | .display_name'
```

Hinweis: Sie können das anhängen `select(.critical == BOOLEAN)` Option zum Filtern der Ergebnisse nach dem kritischen Feld. Mit dem folgenden Befehl werden die Ergebnisse beispielsweise so begrenzt , dass sie nur Geräte enthalten, die vom ExtraHop-System als kritisch eingestuft wurden:

```
curl -s -X GET -H "Authorization: Bearer
 ${ACCESS_TOKEN}"  "$HOST/api/v1/devices?
active_from=1&active_until=0&limit=10000000000"  | jq -r '.[] |
 select(.critical == true) | .display_name'
```

Hinweis: Sie können das anhängen `select(.cloud_instance_name != null)` Option zum Filtern der Ergebnisse nach dem Feld mit dem Namen der Cloud-Instanz. Beispielsweise schränkt der folgende Befehl die Ergebnisse so ein, dass sie nur Geräte mit einem Cloud-Instanznamen enthalten:

```
curl -s -X GET -H "Authorization: Bearer
 ${ACCESS_TOKEN}"  "$HOST/api/v1/devices?
active_from=1&active_until=0&limit=10000000000"  | jq -r '.[] |
 select(.cloud_instance_name != null) | .cloud_instance_name'
```

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

Das ExtraHop GitHub-Repository enthält ein Python-Skript, das die Geräteliste einschließlich aller Geräte-Metadaten extrahiert und die Liste in eine CSV-Datei im selben Verzeichnis wie das Skript schreibt.

1. Gehe zum ExtraHop Codebeispiele GitHub-Repository und laden Sie das herunter `extract_device_list/extract_device_list.py` Datei auf Ihrem lokalen Computer.
2. Öffnen Sie in einem Texteditor den `extract_device_list.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.
     - **CSV_DATEI:** Die Datei, die die Liste der Gerätegruppen enthält.
     - **DATEINAME:** Die Datei, in die die Ausgabe geschrieben wird
     - **GRENZE:** Die maximale Anzahl von Geräten, die mit jeder GET-Anfrage abgerufen werden sollen
     - **SAVEL 2:** Ruft übergeordnete L2-Geräte ab. Diese Variable ist nur gültig, wenn Sie das ExtraHop-System aktiviert haben, Geräte anhand der IP-Adresse zu erkennen.
     - **NUR FÜR FORTGESCHRITTENE:** Ruft nur Geräte ab, die derzeit einer erweiterten Analyse unterzogen werden
     - **NUR HOHER WERT:** Ruft nur Geräte ab, die als hoher Wert eingestuft werden
   - 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 enthält nicht die `/oauth2/token`.
     - **ID:** Die ID der Reveal (x) 360-REST-API-Anmeldeinformationen.
     - **GEHEIM:** Das Geheimnis der Reveal (x) 360 REST-API-Anmeldeinformationen.
     - **CSV_DATEI:** Die Datei, die die Liste der Gerätegruppen enthält.
     - **DATEINAME:** Die Datei, in die die Ausgabe geschrieben wird
     - **GRENZE:** Die maximale Anzahl von Geräten, die mit jeder GET-Anfrage abgerufen werden sollen
     - **SAVEL 2:** Ruft übergeordnete L2-Geräte ab. Diese Variable ist nur gültig, wenn Sie das ExtraHop-System aktiviert haben, Geräte anhand der IP-Adresse zu erkennen.
     - **NUR FÜR FORTGESCHRITTENE:** Ruft nur Geräte ab, die derzeit einer erweiterten Analyse unterzogen werden
     - **NUR HOHER WERT:** Ruft nur Geräte ab, die als hoher Wert eingestuft werden
3. Führen Sie den folgenden Befehl aus:

```
python3 extract_device_list.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)
```