

# Abfragen von Datensätzen über die REST-API

Veröffentlicht: 2024-09-26

Mit der ExtraHop REST API können Sie nach Datensätzen abfragen, die in einem Recordstore gespeichert sind. Durch Abfragen von Datensätzen mit einem REST-API-Skript können Sie Datensätze in eine Drittanbieteranwendung wie Microsoft Excel importieren. Wenn Ihre Abfrage mit mehr als der maximalen Anzahl von Datensätzen übereinstimmt, die von der REST-API zurückgegeben werden, können Sie das Skript außerdem so konfigurieren, dass es rekursiv nach den verbleibenden Datensätzen fragt. In diesem Thema zeigen wir Methoden zum Abfragen von Datensätzen sowohl über den ExtraHop REST API Explorer als auch über ein Python-Skript.

## Bevor Sie beginnen

- Sie müssen sich anmelden bei Sensor oder Konsole mit einem Konto, das über volle Schreibrechte verfügt, um einen API-Schlüssel zu generieren.
- Sie benötigen einen gültigen API-Schlüssel, um Änderungen über die REST-API vornehmen und die folgenden Verfahren ausführen zu können. (siehe [Generieren Sie einen API-Schlüssel](#).)
- Machen Sie sich vertraut mit dem [ExtraHop REST API-Leitfaden](#) um zu erfahren, wie Sie im ExtraHop REST API Explorer navigieren.

## Abfragen von Datensätzen über den REST API Explorer

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. Geben Sie Ihre REST-API-Anmeldeinformationen Anmeldeinformationen.

- Für Sensoren und ECA-VMs klicken Sie auf **API-Schlüssel eingeben** und fügen Sie dann Ihren API-Schlüssel ein oder geben Sie ihn in das **API-Schlüssel** Feld.
- Klicken Sie für RevealX 360 auf **Geben Sie die API-Anmeldeinformationen ein** und fügen Sie dann die ID und das Geheimnis Ihrer API-Anmeldeinformationen ein oder geben Sie sie in das **ID** und **Geheim** Felder.

3. Klicken Sie **Autorisieren** und klicken Sie dann **Schliessen**.

4. Klicken Sie **Protokoll aufzeichnen** und klicken Sie dann **POST /Datensätze/Suche**.

5. Klicken Sie **Probieren es aus**.

Das JSON-Schema wird automatisch dem hinzugefügt Körper Parameter-Textfeld.

6. Geben Sie im Textfeld Textkörper Felder für Ihre Datensatzabfrage an.

Mit den folgenden Feldern werden beispielsweise Datensätze der letzten 30 Minuten abgerufen, die eine IP-Adresse, einen Domännennamen oder eine URI enthalten, die gemäß folgenden Angaben als verdächtig eingestuft wurden [Bedrohungsinformationen](#):

```
{
  "from": "-30m",
  "filter": {
    "field": "ex.isSuspicious",
    "operator": "=",
    "operand": {
      "type": "boolean",
      "value": "true"
    }
  }
}
```

Eine vollständige Liste der gültigen Felder finden Sie im Abschnitt Körperparameter unter **POST / Datensätze/Suche** im REST API Explorer.

## Python-Skriptbeispiele

Die folgenden Python-Skripte fragen nach Datensätzen ab, die eine IP-Adresse, einen Domainnamen oder eine URI enthalten, die laut Bedrohungsinformationen als verdächtig eingestuft wurden. Die Skripts schreiben dann die angegebenen Datensatzfelder in eine CSV-Datei, die in einem Tabellenkalkulationsprogramm angezeigt werden kann.

 **Hinweis** Weitere Informationen zu Bedrohungsinformationen mit ExtraHop finden Sie unter [Bedrohungsinformationen](#) und [Laden Sie STIX-Dateien über die REST-API hoch](#).

### Ruft das Python-Beispielskript für einen ExtraHop-Recordstore ab und führt es aus

Das ExtraHop GitHub-Repository enthält ein Python-Beispielskript, das Datensätze aus einem ExtraHop-Recordstore abrufen.


#### Bevor Sie beginnen

Dieses Beispielskript unterstützt RevealX 360 nicht. Hinweise zu einem Beispielskript, das RevealX 360 unterstützt, finden Sie unter [Rufen Sie das Python-Beispielskript für einen Drittanbieter- oder Cloud-Recordstore ab und führen Sie es aus](#).

 **Wichtig:** Wenn die Abfrage mit mehr als der maximalen Anzahl von Datensätzen übereinstimmt, die gleichzeitig abgerufen werden können, ruft das Skript die verbleibenden Datensätze ab, indem es mit der POST /records/cursor-Operation einen Cursor an den Sensor oder die Konsole sendet. Diese Operation ist nur mit ExtraHop Recordstore gültig. Wenn Sie einen Drittanbieter- oder Cloud-Recordstore konfiguriert haben, finden Sie unter [Rufen Sie das Python-Beispielskript für einen Drittanbieter- oder Cloud-Recordstore ab und führen Sie es aus](#).

1. Gehe zum [ExtraHop Codebeispiele GitHub-Repository](#) und laden Sie das herunter `query_records_explore/query_records_explore.py` Datei auf Ihrem lokalen Computer.
2. Öffnen Sie in einem Texteditor den `query_records_explore.py` archivieren und ersetzen Sie die folgenden Konfigurationsvariablen durch Informationen aus Ihrer Umgebung:
  - **GASTGEBER:** Die IP-Adresse oder der Hostname des Sensor oder der Konsole. Beachten Sie, dass dieser Hostname nicht der Hostname des verbundenen ExtraHop-Recordstores ist, auf dem die Datensätze gespeichert sind.
  - **EIN HÜNENSCHLÜSSEL:** Der API-Schlüssel.
  - **DATEINAME:** Die Datei, in die die Ausgabe geschrieben wird.
  - **ZEITLIMIT:** Wenn die Datensatzabfrage mit mehr als 100 Datensätzen übereinstimmt, die Zeitspanne nach der ersten Abfrage, bis die verbleibenden Datensätze aus dem System abgerufen werden können.
  - **ABFRAGE:** Die Parameter für die Datensatzabfrage.
  - **SPALTEN:** Die Datensatzfelder, die in die CSV-Ausgabedatei geschrieben werden.
3. Führen Sie den folgenden Befehl aus:

```
python3 query_records_explore.py
```

 **Hinweis** Wenn das Skript eine Fehlermeldung zurückgibt, dass die TLS-Zertifikatsüberprüfung fehlgeschlagen ist, stellen Sie sicher, dass [Ihrem Sensor oder Ihrer Konsole wurde ein vertrauenswürdigen Zertifikat hinzugefügt](#). Alternativ können Sie das `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)
```

## Rufen Sie das Python-Beispielskript für einen Drittanbieter- oder Cloud-Recordstore ab und führen Sie es aus

Das ExtraHop GitHub-Repository enthält ein Python-Beispielskript, das Datensätze aus Drittanbieter- und Cloud-Recordstores abrufen.



**Hinweis** Wenn die Abfrage mit mehr als der maximalen Anzahl von Datensätzen übereinstimmt, die gleichzeitig abgerufen werden können, ruft das Skript die verbleibenden Datensätze ab, indem es zusätzliche Anfragen mit dem `offset` Parameter. Der Offset-Parameter überspringt eine bestimmte Anzahl von Datensätzen in einer Abfrage.

1. Gehe zum [ExtraHop Codebeispiele GitHub-Repository](#) und laden Sie das herunter `query_records_third_party/query_records_third_party.py` Datei auf Ihrem lokalen Computer.
2. Öffnen Sie in einem Texteditor den `query_records_third_party.py` archivieren und ersetzen Sie die folgenden 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.
    - **DATEINAME**: Die Datei, in die die Ausgabe geschrieben wird.
    - **ZEITLIMIT**: Wenn die Datensatzabfrage mit mehr als 100 Datensätzen übereinstimmt, die Zeitspanne nach der ersten Abfrage, bis die verbleibenden Datensätze aus dem System abgerufen werden können.
    - **ABFRAGE**: Die Parameter für die Datensatzabfrage.
    - **SPALTEN**: Die Datensatzfelder, die in die CSV-Ausgabedatei geschrieben werden.
  - Geben Sie für RevealX 360 die folgenden Konfigurationsvariablen an:
    - **GASTGEBER**: Der Hostname der RevealX 360-API. Dieser Hostname wird auf der RevealX 360 API Access-Seite unter API-Endpunkt angezeigt . Der Hostname beinhaltet nicht `/oauth2/token`.
    - **ID**: Die ID der RevealX 360 REST-API-Anmeldeinformationen.
    - **GEHEIM**: Das Geheimnis der RevealX 360 REST-API-Anmeldeinformationen.
    - **DATEINAME**: Die Datei, in die die Ausgabe geschrieben wird.
    - **ZEITLIMIT**: Wenn die Datensatzabfrage mit mehr als 100 Datensätzen übereinstimmt, die Zeitspanne nach der ersten Abfrage, bis die verbleibenden Datensätze aus dem System abgerufen werden können.
    - **ABFRAGE**: Die Parameter für die Datensatzabfrage.
    - **SPALTEN**: Die Datensatzfelder, die in die CSV-Ausgabedatei geschrieben werden.
3. Führen Sie den folgenden Befehl aus:

```
python3 query_records_third_party.py
```



**Hinweis** Wenn das Skript eine Fehlermeldung zurückgibt, dass die TLS-Zertifikatsüberprüfung fehlgeschlagen ist, stellen Sie sicher, dass **Ihrem Sensor oder Ihrer Konsole wurde ein vertrauenswürdigen 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)
```