

Laden Sie STIX-Dateien über die REST-API hoch

Veröffentlicht: 2025-03-28

Veröffentlicht: 2025-03-28

Mithilfe von Bedrohungssammlungen kann Ihr ExtraHop-System verdächtige IP-Adressen, Hostnamen und URIs identifizieren, die in Ihrer Netzwerkaktivität gefunden wurden. Obwohl von Extrahop kuratierte Bedrohungssammlungen standardmäßig aktiviert sind, können Sie auch eine benutzerdefinierte Bedrohungssammlung aus kostenlosen oder kommerziellen Quellen hochladen.

Bevor Sie beginnen

 **Wichtig:** STIX-Datei-Uploads sind jetzt veraltet und werden voraussichtlich im März 2025 entfernt.

- Für RevealX 360 benötigen Sie gültige REST-API-Anmeldeinformationen, um Änderungen über die REST-API vorzunehmen und die folgenden Verfahren durchzuführen. (siehe [REST-API-Anmeldeinformationen erstellen](#).)
- Für Sensoren und die ExtraHop-Konsole benötigen Sie einen gültigen API-Schlüssel, um Änderungen über die REST-API vorzunehmen und die folgenden Verfahren durchzuführen. (siehe [Generieren Sie einen API-Schlüssel](#).)
- Machen Sie sich vertraut mit [Bedrohungsinformationen](#).


Bedrohungssammlungen müssen für alle verbundenen Geräte hinzugefügt und aktualisiert werden Sensoren und Konsolen. Und da diese Quellen häufig aktualisiert werden, bietet die REST-API die Möglichkeit, Aktualisierungen für Bedrohungssammlungen für alle zu automatisieren. Sensoren und Konsolen.

Benutzerdefinierte Bedrohungssammlungen müssen in Structured Threat Information Expression (STIX) als komprimierte TAR-Dateien wie .TGZ oder TAR.GZ formatiert werden. ExtraHop-Systeme unterstützen derzeit das Hochladen der STIX-Dateiversionen 1.0 - 1.2.

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

Das ExtraHop GitHub-Repository enthält ein Beispiel-Python-Skript, das alle STIX-Dateien in einem bestimmten Verzeichnis in eine Liste von hochlädt Sensoren und Konsolen. Zunächst liest das Skript eine CSV-Datei durch, die die URLs und API-Schlüssel für jedes System enthält. Für jedes System ruft das Skript eine Liste aller Bedrohungssammlungen ab, die sich bereits auf dem System befinden. Das Skript verarbeitet dann jede STIX im Verzeichnis für jedes System.

Wenn der Name der Datei mit dem Namen einer Bedrohungssammlung auf dem System übereinstimmt, überschreibt das Skript die Bedrohungssammlung mit dem Dateiinhalt. Wenn es keine Namen der Bedrohungssammlung gibt, die dem Dateinamen entsprechen, lädt das Skript die Datei hoch, um eine neue Bedrohungssammlung zu erstellen.

 **Hinweis** Das folgende Verfahren ist nicht mit der RevealX 360-REST-API kompatibel. Informationen zum Hochladen von STIX-Dateien auf RevealX 360 finden Sie unter [Rufen Sie das Python-Beispielskript für RevealX 360 ab und führen Sie es aus](#).

1. Gehe zum [ExtraHop Codebeispiele GitHub-Repository](#) und laden Sie das herunter `upload_stix/upload_stix.py` Datei auf Ihrem lokalen Computer.
2. Erstellen Sie eine CSV-Datei mit Zeilen, die die folgenden Spalten in der angegebenen Reihenfolge enthalten:

Hostname des Systems

API-Schlüssel



Hinweis Das `upload_stix` Verzeichnis enthält eine Beispiel-CSV-Datei mit dem Namen `systems.csv`.

- Öffnen Sie in einem Texteditor den `upload_stix.py` archivieren und ersetzen Sie die folgenden Konfigurationsvariablen durch Informationen aus Ihrer Umgebung:
 - **SYSTEM_LISTE:** Der Pfad der CSV-Datei mit den HTTPS-URLs und API-Schlüsseln der Systeme
 - **STIX_DIR:** Der Pfad des Verzeichnisses, das die STIX-Dateien enthält
- Führen Sie den folgenden Befehl aus:

```
python3 upload_stix.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)
```

Rufen Sie das Python-Beispielskript für RevealX 360 ab und führen Sie es aus

Das ExtraHop GitHub-Repository enthält ein Python-Skript, das alle STIX-Dateien in einem bestimmten Verzeichnis auf RevealX 360 hochlädt.

Wenn der Name der Datei mit dem Namen einer Bedrohungssammlung auf RevealX 360 übereinstimmt, überschreibt das Skript die Bedrohungssammlung mit dem Dateiinhalt. Wenn es keine Namen der Bedrohungssammlung gibt, die dem Dateinamen entsprechen, lädt das Skript die Datei hoch, um eine neue Bedrohungssammlung zu erstellen.



Hinweis Das folgende Verfahren ist nur mit der RevealX 360 REST-API kompatibel. Informationen zum Hochladen von STIX-Dateien auf Sensoren und die ExtraHop-Konsole finden Sie unter [Rufen Sie das Python-Beispielskript ab und führen Sie es aus](#).

- Gehe zum [ExtraHop Codebeispiele GitHub-Repository](#) [🔗](#) und laden Sie das herunter `upload_stix/upload_stix_rx360.py` Datei auf Ihrem lokalen Computer.
- Öffnen Sie in einem Texteditor den `create_device_groups.py` archivieren und ersetzen Sie die folgenden Konfigurationsvariablen durch Informationen aus Ihrer Umgebung:
 - **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.
 - **STIX_DIR:** Der Pfad des Verzeichnisses, das die STIX-Dateien enthält
- Führen Sie den folgenden Befehl aus:

```
python3 upload_stix_rx360.py
```