Extract files from packets through the REST API

Published: 2025-02-06

This guide provides instructions for extracting files (also known as file carving) through the ExtraHop REST API Explorer and through a Python script.

Before you begin

- For sensors and the ExtraHop console, you must have a valid API key to make changes through the REST API and complete the procedures below. (See Generate an API key.)
- For RevealX 360, you must have valid REST API credentials to make changes through the REST API and complete the procedures below. (See Create REST API credentials.)

Extract files through the REST API Explorer

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

   The URL is the hostname or IP address of your sensor or console, followed by `/api/v1/explore/`. For example, if your hostname is seattle-eda, the URL is `https://seattle-eda/api/v1/explore/`.
2. Enter your REST API credentials.
   - For sensors and the ExtraHop console, click **Enter API Key** and then paste or type your API key into the **API Key** field.
   - For RevealX 360, click **Enter API Credentials** and then paste or type the ID and secret of your API credentials into the **ID** and **Secret** fields.
3. Click **Authorize** and then click **Close**.
4. Click **Packet Search** and then click **POST /packets/search**.
5. Click **Try it out**.

   The JSON schema is automatically added to the **body** text box.
6. In the **body** text box, specify search parameters for the packets you want to extract files from.

   For example, the following parameters retrieve files from packets that were sent to or from the IP address 10.10.10.10 over the last 30 minutes:

   ```
   {
       "from": "-30m",
       "output": "extract",
       "ip1": "10.10.10.10"
   }
   ```
7. Click **Send Request**.

   When the request completes, the Server Response section appears. If the request was successful, a 200 status code is displayed.
8. Next to the 200 status code, click **Download file**.

Retrieve and run the example Python script

The ExtraHop GitHub repository contains an example Python script that extracts files from packets through the REST API.

1. Go to the ExtraHop code-examples GitHub repository and download the `extract_files/extract_files.py` file to your local machine.

2. In a text editor, open the `extract_files.py` file and replace the configuration variables with information from your environment.
   a) For sensors and the ExtraHop console, specify the following configuration variables:
      - **HOST:** The IP address or hostname of the sensor or console.
      - **API_KEY:** The API key.

   b) For Reveal(x) 360, specify the following configuration variables:
      - **HOST:** The hostname of the Reveal(x) 360 API. This hostname is displayed in the Reveal(x) 360 API Access page under API Endpoint. The hostname does not include the /oauth2/token.
      - **ID:** The ID of the Reveal(x) 360 REST API credentials.
      - **SECRET:** The secret of the Reveal(x) 360 REST API credentials.

   c) For all systems, specify the **SEARCH** configuration variable for the packets you want to extract files from.
3. Run the following command:

   ```
   python3 extract_files.py
   ```

   If the system is successful, the files are saved to a `.zip` file named `extracted_files.zip`.

   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)
   ```