HYPR Integration with SIEMs

Sending HYPR event data to to a SIEM

Overview

HYPR Event Hooks integrates with virtually any SIEM to stream event data in real time from HYPR to the SIEM. All HYPR software services generate detailed events revealing insights into the state of services and transactions. While the HYPR events can be viewed in the HYPR Control Center administrative interface, it is best to ingest the data into a purpose-built SIEM for long-term storage and data mining.

Technology

HYPR Event Hooks utilize web hook technology to stream HYPR events to an external SIEM. Simply configure an HTTP Event Collector (HEC) in the target SIEM to receive the HYPR events. Each time HYPR generates an event, it is immediately added to the event stream

Configuration

HYPR supports Splunk and Datadog out of the box, but any SIEM that supports web hooks can be configured as a target for HYPR event streaming.

Configuring HYPR Event Hooks involves setting up the SIEM to receive events with a HEC and configuring HYPR with the SIEM target information.

Sending events to Splunk

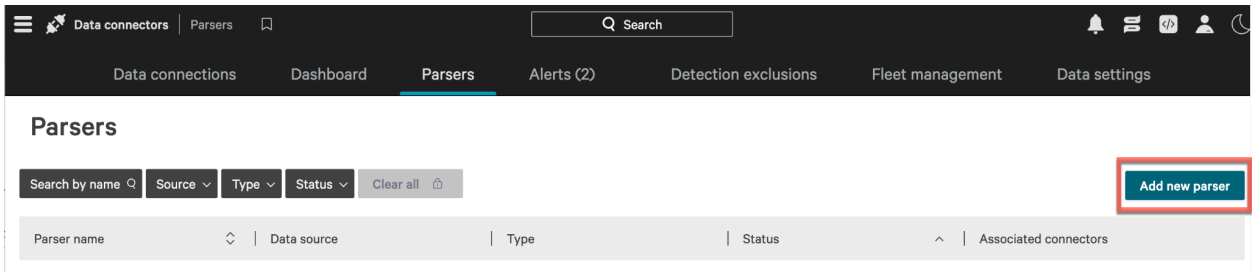
HYPR has a built-in integration with Splunk. The HYPR product documentation describes the steps to set up both Splunk and HYPR. See Event Hook: Splunk in the HYPR documentation.

Sending events to Crowdstrike

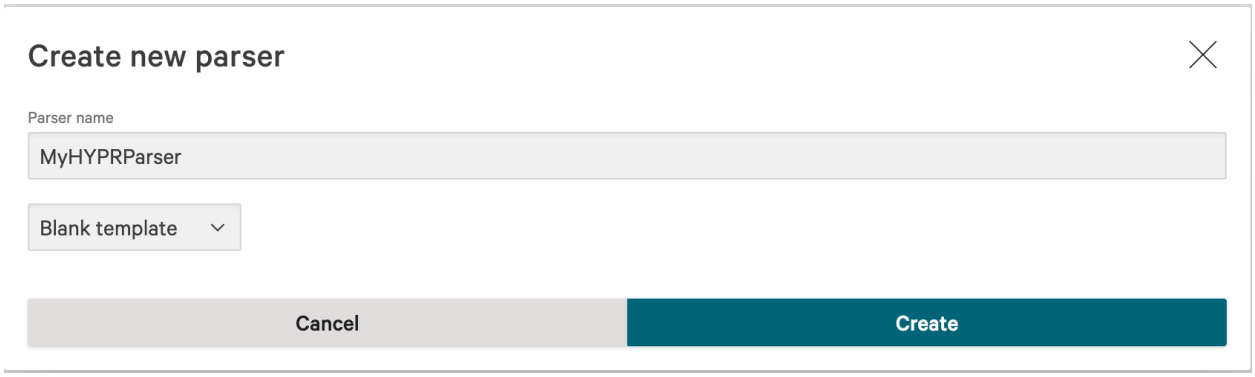
Crowstrike supports receiving data over an HTTP Event Data Collector, which can be found in their online documentation. As part of the event collector configuration, you must also configure a parser to transform the HYPR data into a common format for Crowdstrike.

Configure the Crowdstrike parser

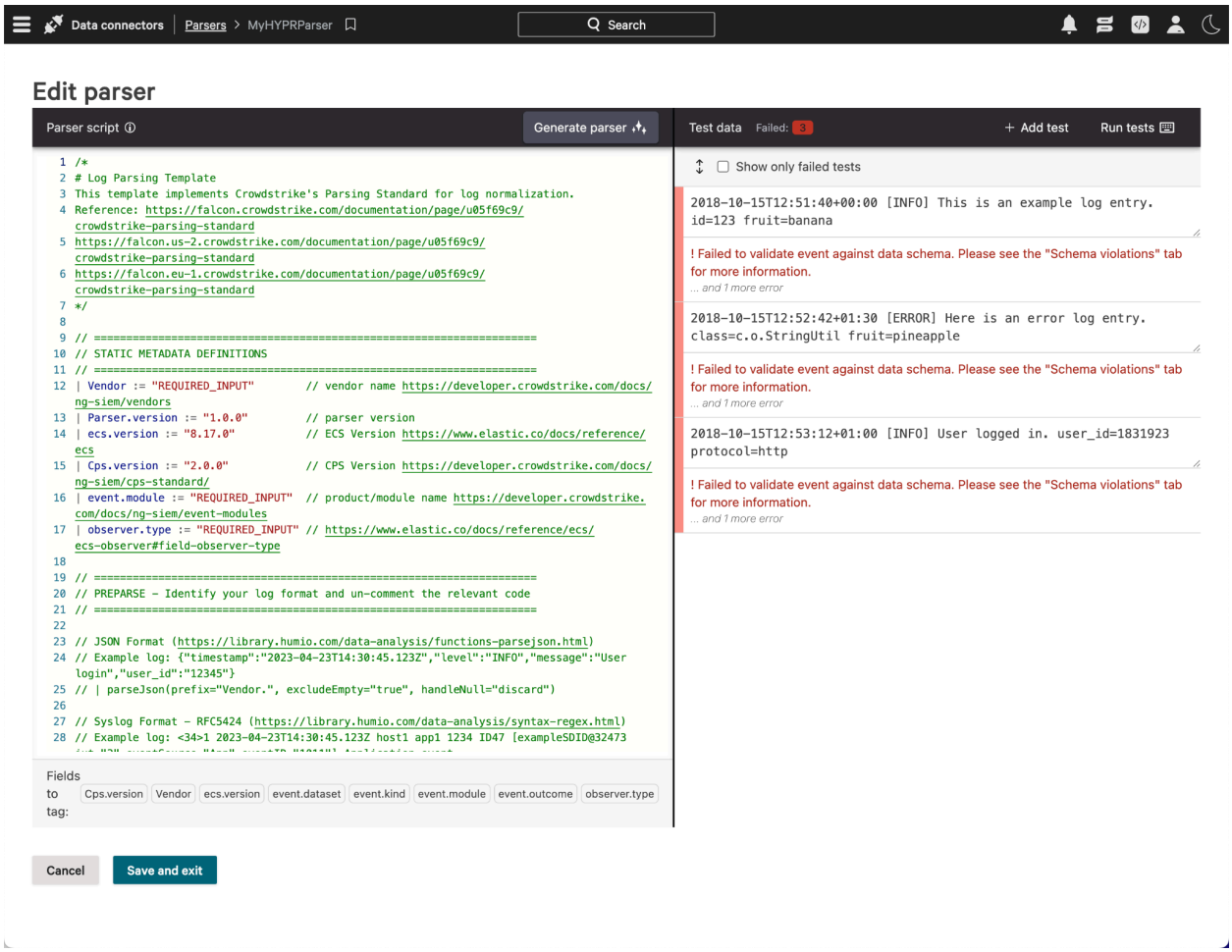
1. In the Falcon console, go to Next-Gen SIEM > Log Management > Data onboarding > Parsers. Click on the **Add new parser** button



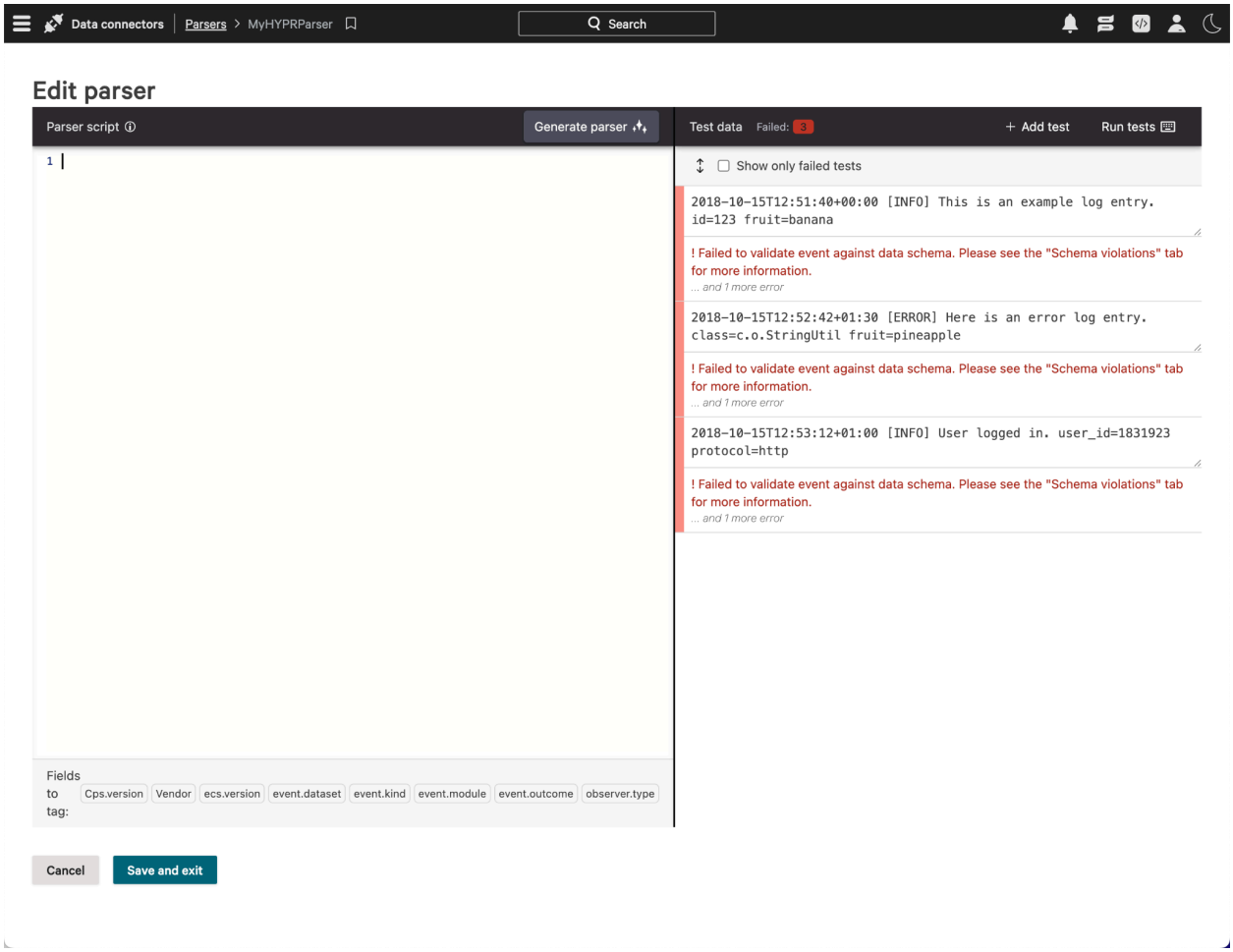
2. In the dialog box, give the parser a name and select “Blank template” in the drop-down. Click the **Create** button.



3. You will be brought to a parser edit GUI, which comes pre-populated with some parser code. It looks like this:



4. Select all the code on the left and deleted it



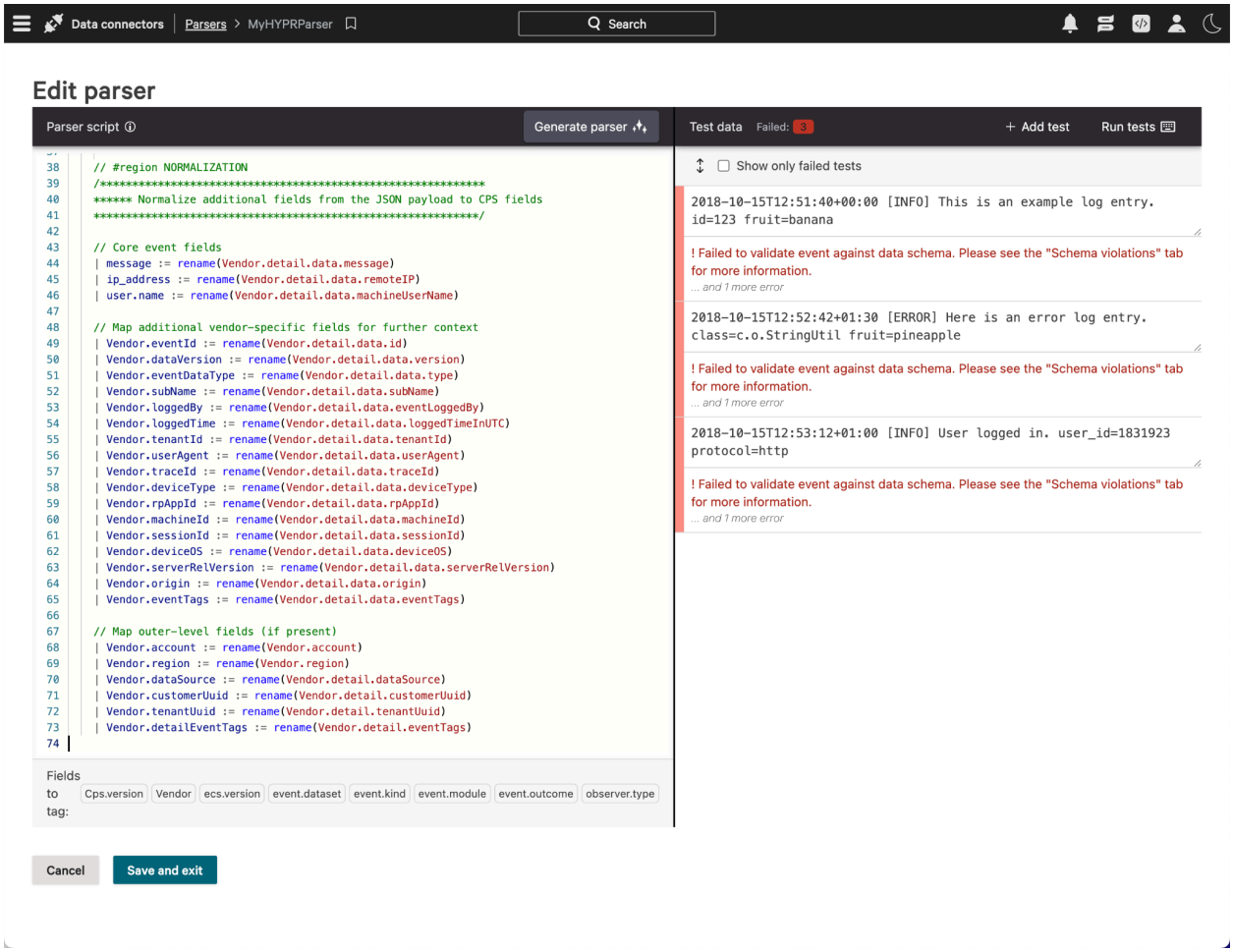
5. Next copy/paste this code into the Parser script text box:

```
      // HYPR Parser

    // #region PREPARSE
    /*************************************************************
    ****** Parse JSON payload with prefix "Vendor."
    ****** This flattens the JSON so that all keys are prefixed with "Vendor."
    *************************************************************/
  | parseJson(prefix="Vendor.", handleNull="discard", excludeEmpty=true)

    // Attempt to parse the timestamp from detail.data.eventTimeInUTC (milliseconds)
    // or, if not present, from Vendor.time using an explicit ISO8601 format.
    | case {
        Vendor.detail.data.eventTimeInUTC = *
            | parseTimestamp(field="Vendor.detail.data.eventTimeInUTC",
format="milliseconds");
        Vendor.time = *
            | parseTimestamp(field="Vendor.time", format="yyyy-MM-dd'T'HH:mm:ssX");
        *;
```

```
}
// #endregion

// #region METADATA
/*************************************************************
****** Static Metadata Definitions (required fields)
*************************************************************/
| Parser.version := "1.0.0"
| Vendor := "hypr"
| event.kind := "event"
| event.module := "fido2"
| ecs.version := "8.11.0"
| Cps.version := "1.0.0"
| event.dataset := "fido2.registration"
| event.category[0] := "authentication"
| event.type[0] := "user"
// #endregion

 // #endregion

// #region NORMALIZATION
/*************************************************************
****** Normalize additional fields from the JSON payload to CPS fields
*************************************************************/

// Core event fields
| message := rename(Vendor.detail.data.message)
| ip_address := rename(Vendor.detail.data.remoteIP)
| user.name := rename(Vendor.detail.data.machineUserName)

// Map additional vendor-specific fields for further context
| Vendor.eventId := rename(Vendor.detail.data.id)
| Vendor.dataVersion := rename(Vendor.detail.data.version)
| Vendor.eventDataType := rename(Vendor.detail.data.type)
| Vendor.subName := rename(Vendor.detail.data.subName)
| Vendor.loggedBy := rename(Vendor.detail.data.eventLoggedBy)
| Vendor.loggedTime := rename(Vendor.detail.data.loggedTimeInUTC)
| Vendor.tenantId := rename(Vendor.detail.data.tenantId)
| Vendor.userAgent := rename(Vendor.detail.data.userAgent)
| Vendor.traceId := rename(Vendor.detail.data.traceId)
| Vendor.deviceType := rename(Vendor.detail.data.deviceType)
| Vendor.rpAppId := rename(Vendor.detail.data.rpAppId)
| Vendor.machineId := rename(Vendor.detail.data.machineId)
| Vendor.sessionId := rename(Vendor.detail.data.sessionId)
| Vendor.deviceOS := rename(Vendor.detail.data.deviceOS)
| Vendor.serverRelVersion := rename(Vendor.detail.data.serverRelVersion)
| Vendor.origin := rename(Vendor.detail.data.origin)
| Vendor.eventTags := rename(Vendor.detail.data.eventTags)

// Map outer-level fields (if present)
| Vendor.account := rename(Vendor.account)
| Vendor.region := rename(Vendor.region)
| Vendor.dataSource := rename(Vendor.detail.dataSource)
| Vendor.customerUuid := rename(Vendor.detail.customerUuid)
```

```
  | Vendor.tenantUuid := rename(Vendor.detail.tenantUuid)
  | Vendor.detailEventTags := rename(Vendor.detail.eventTags)
```

6. Your screen will now look like this:

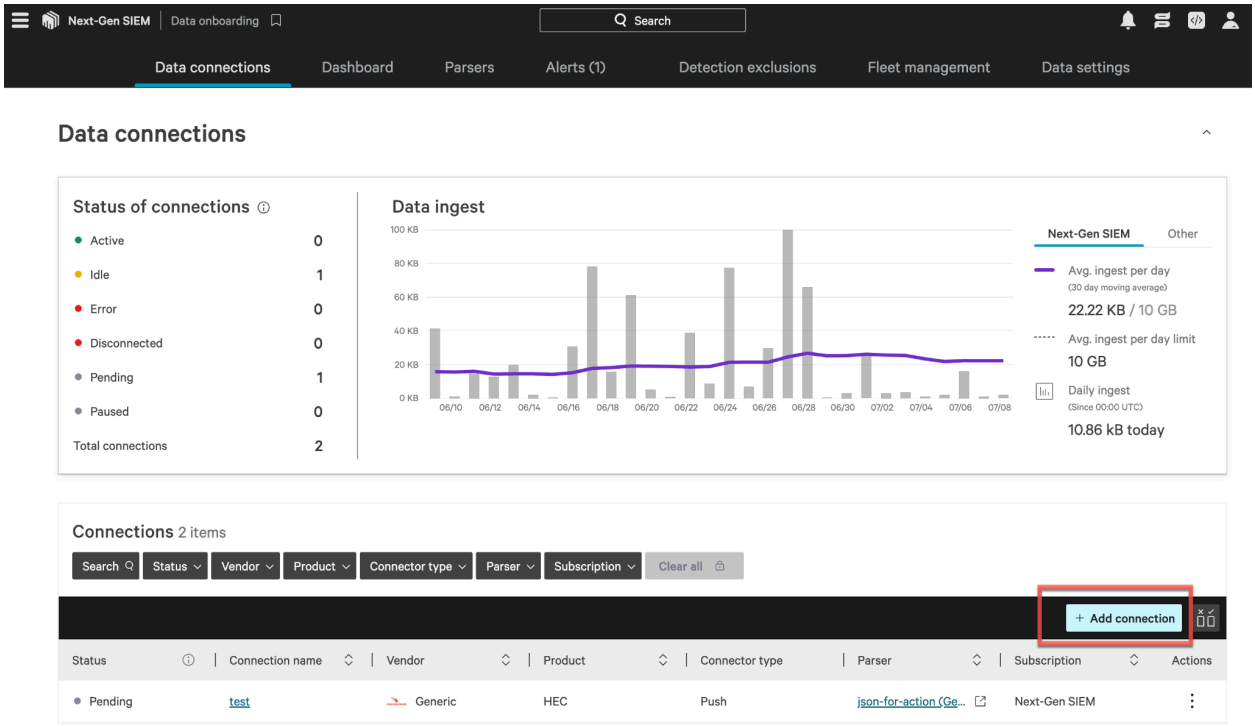


7. Click the **Save and exit** button

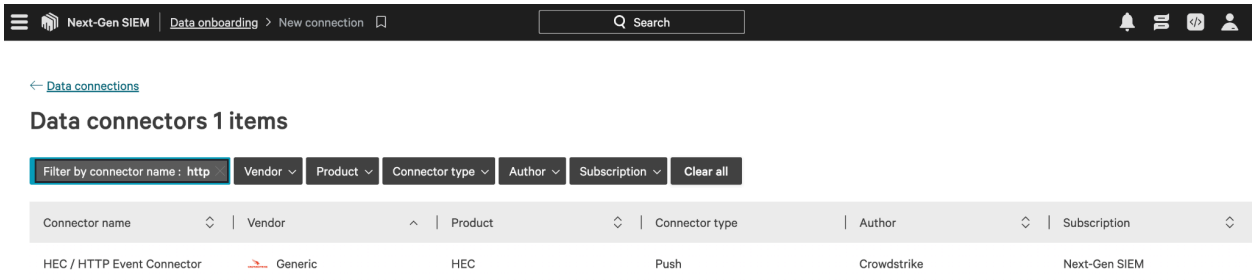
Your parser has now been saved for use during setup of the data connector in the next section.

Configure the Crowdstrike data connector

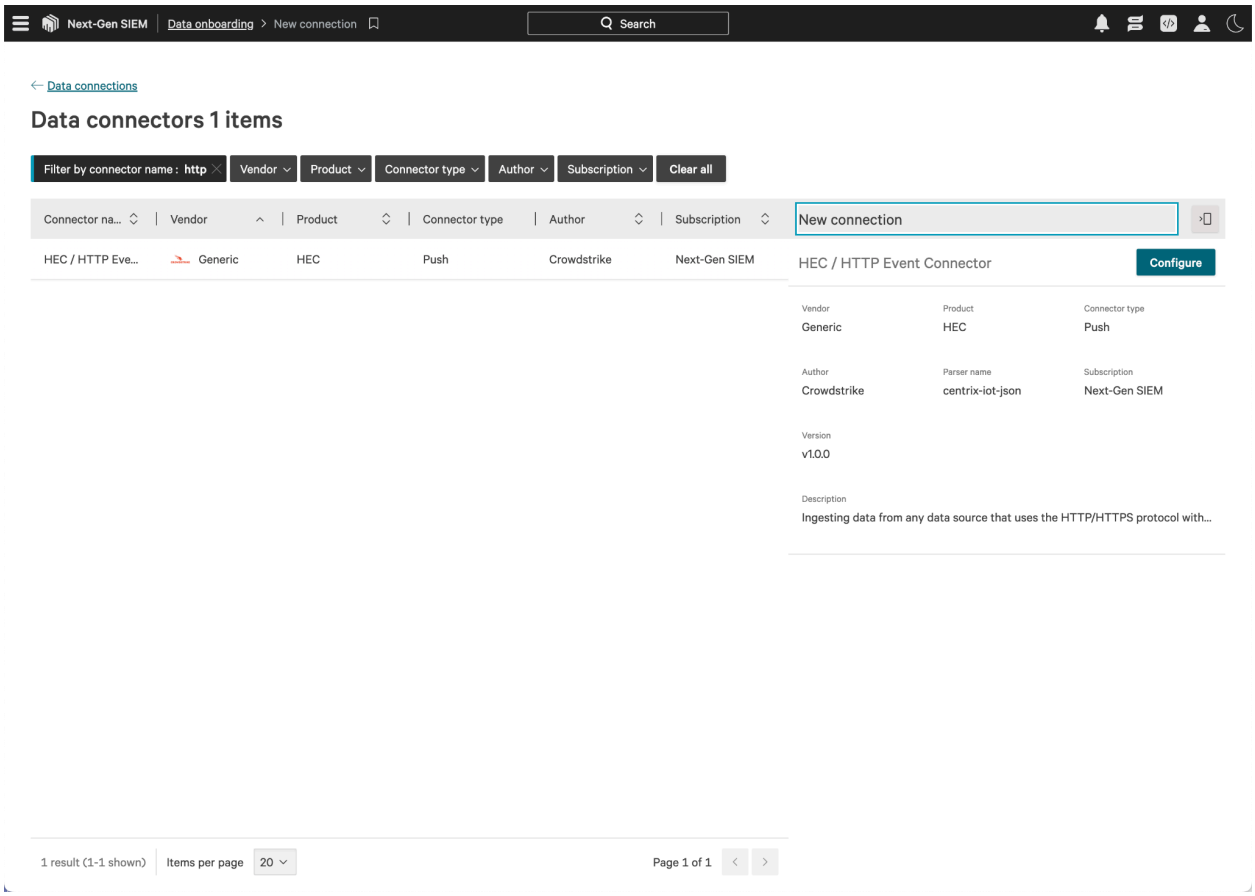
1. In the Falcon console, go to Next-Gen SIEM > Log Management > Data onboarding > Data connections. Click on the **+Add connection** button



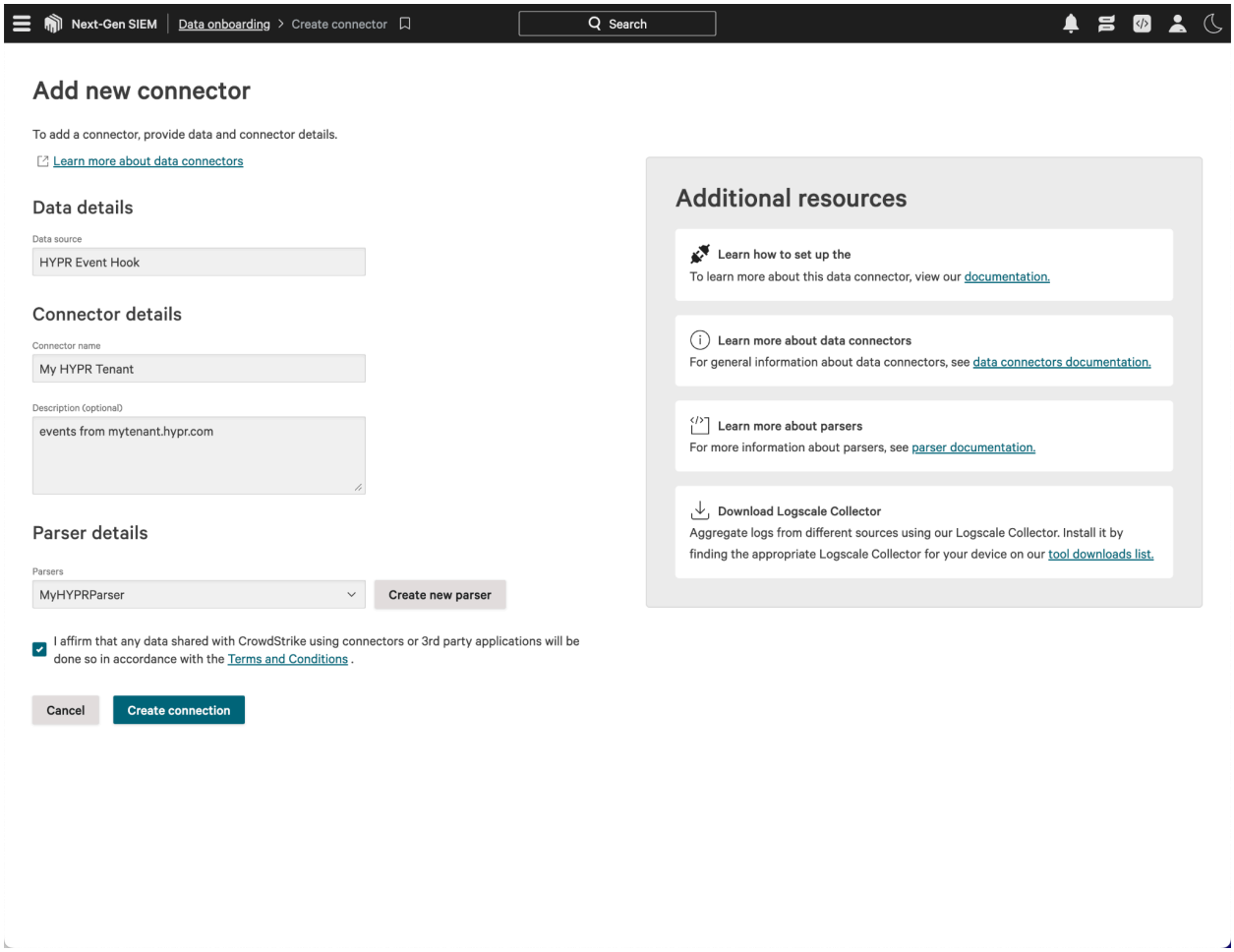
2. In the Data Connectors page, filter or sort by Connector name, Vendor, Product, Connector Type, Author, or Subscription to find and select the HEC/HTTP Event Data Connector.



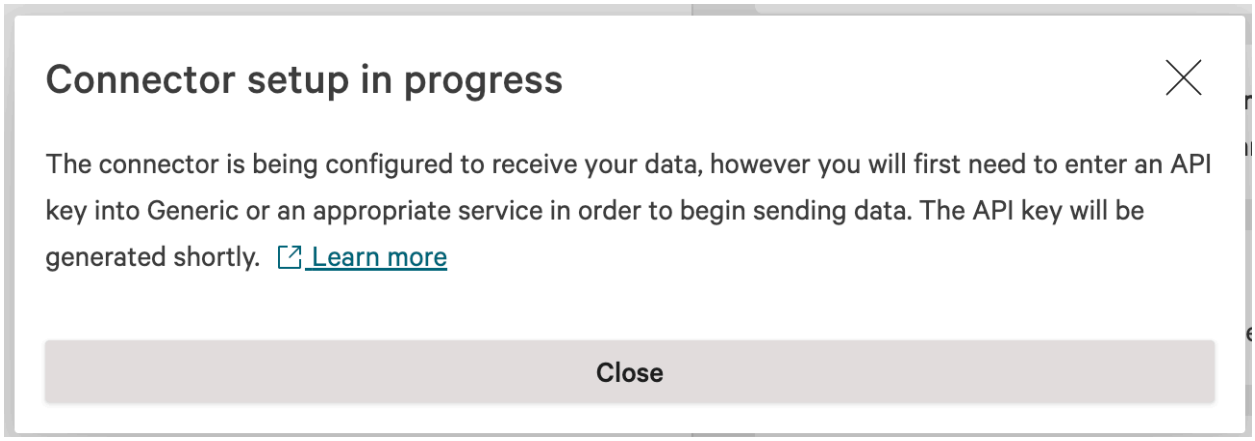
3. In the New connection dialog, review connector metadata, version, and description. Click Configure.



4. In the Add new connector page, enter or select these details:
    - Data source: Enter a name for the data source to display on the connection's Details page.
    - Connector name: Enter a name to identify the connector. This name displays in the Connections list.
    - Description: Optional. Enter a description of the connector.
    - Parsers: Select a parser to use for this connection. In the Parsers dropdown menu, search for an existing parser that aligns with the data source. If such a parser does not exist, you need to create a custom parser. To create a custom parser, click Create new parser. For more info, see Add a new parser. For custom parser requirements, see Understanding the CrowdStrike Parsing Standard.

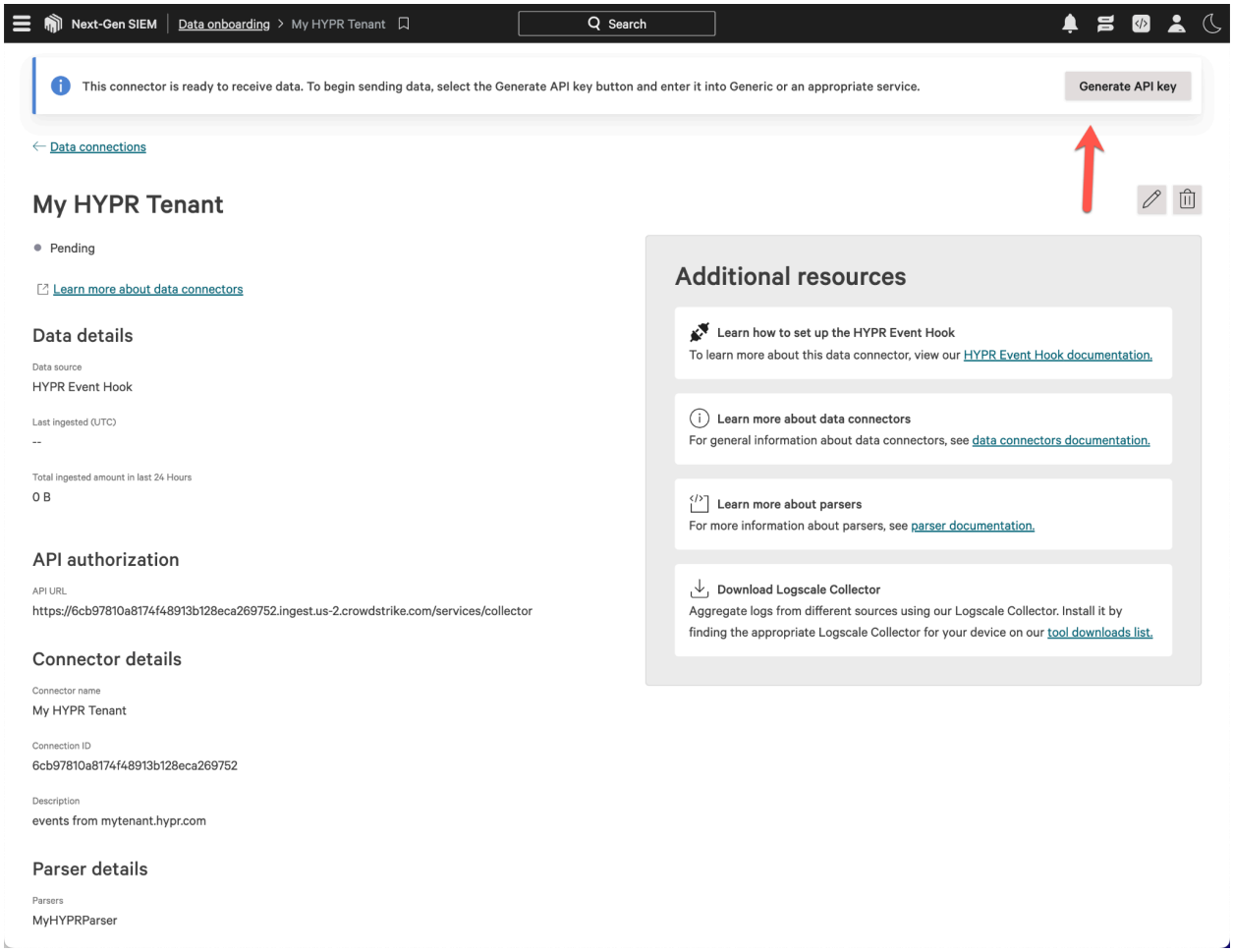


5. Click the Terms and Conditions box, then click the **Create connection**.
6. A banner message appears in the Falcon console when your API key and API URL are ready to be generated. Click the Close button.

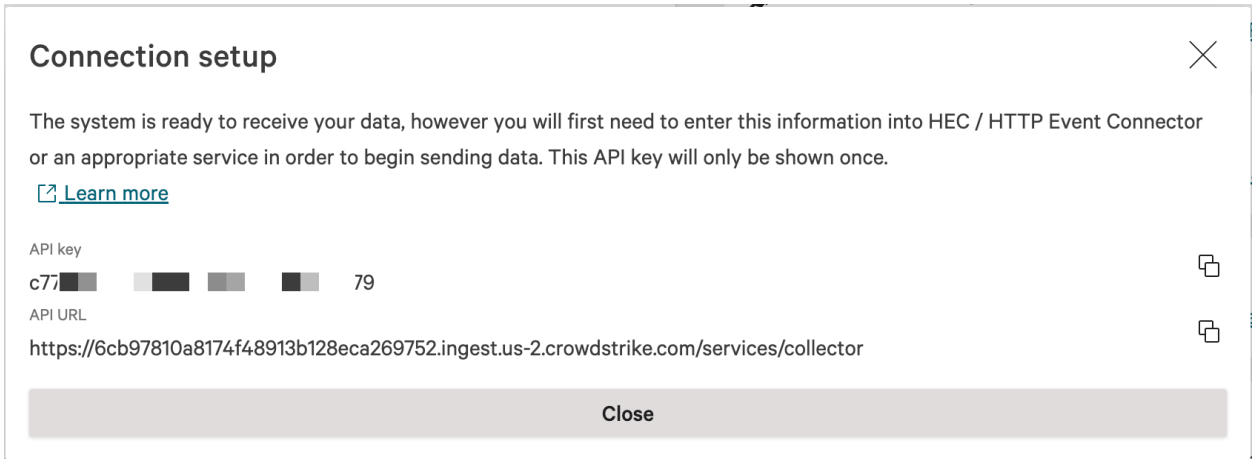


7. You will be returned to the configuration screen for the data connector you just created. At the top of the page, you will see a message that you need to generate an API key.

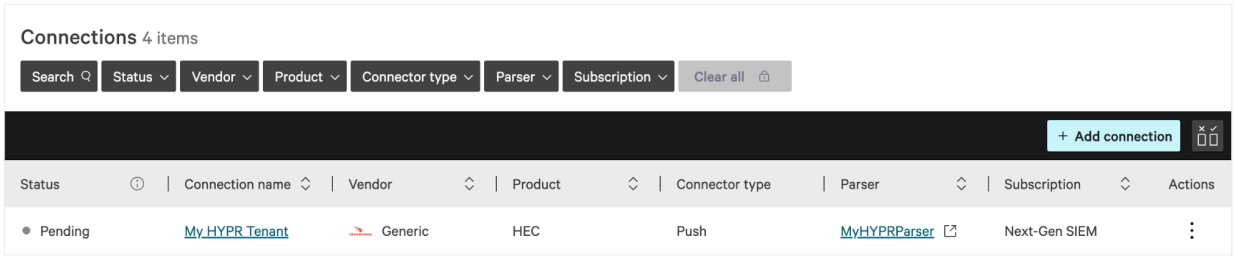
Click the **Generate API key** button.



8. You will see a dialog box with the API key and API URL. Copy and safely store the API key and API URL to use during connector configuration. Click the **Close** button.



Return to the Next-Gen SIEM > Log Management > Data onboarding > Data connections screen to view your new data connector.



The status will remain “Pending” until HYPR has been configured to send data to this connector.

This completes the Crowdstrike data connection configuration.

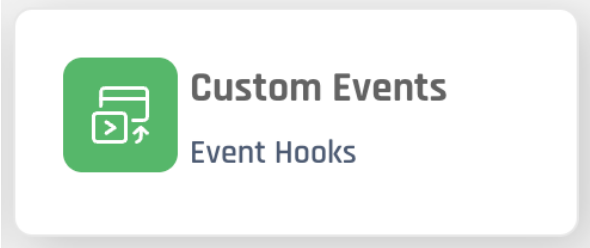
Configuring Event Hook in HYPR

Use the AP key and API URL from the previous section to configure the JSON for the HYPR event hook.

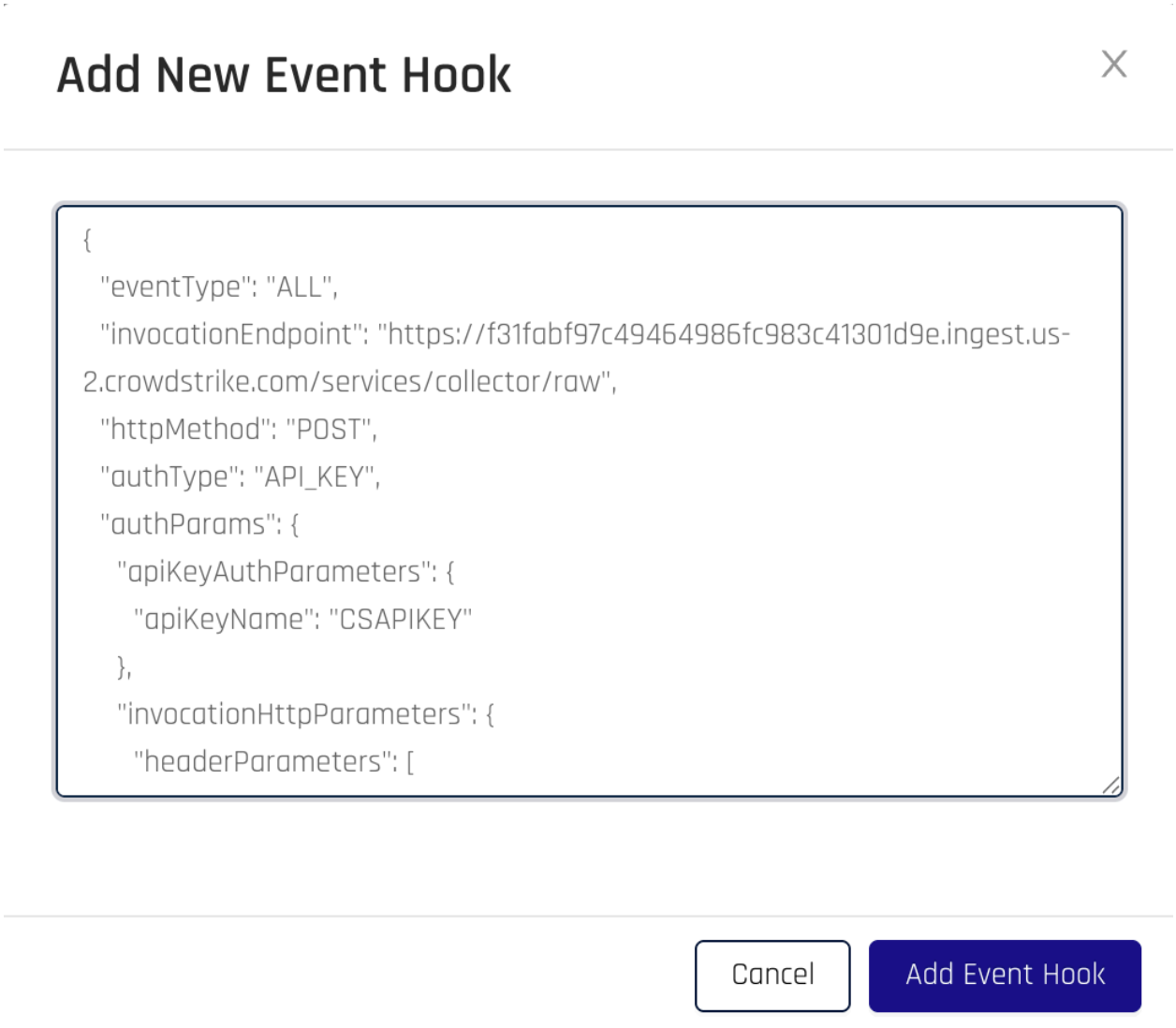
Follow the procedure for configuring a custom event hook in the Event Hook: Custom page in the HYPR documentation. Copy the below JSON data into a text editor and add your path, port and token.

```
{
  "name": "Crowdstrike Event Hook",
  "eventType": "ALL",
  "invocationEndpoint": "<API URL>/raw",
  "httpMethod": "POST",
  "authType": "API_KEY",
  "authParams": {
    "apiKeyAuthParameters": {
      "apiKeyName": "Authorization",
      "apiKeyValue": "Bearer XXXXX"
    },
    "invocationHttpParameters": {
      "headerParameters": [
        {
          "key": "Content-Type",
          "value": "application/json",
          "isValueSecret": false
        }
      ]

    }
  }
}
```

In the HYPR Control Center, navigate to Integrations. Click the **Add new integration** button and choose Custom Events.



Copy the JSON into the dialog box and click the **Add Event Hook** button.



At this point new HYPR events will be sent to Crowdstrike.

Sending events to Cribl

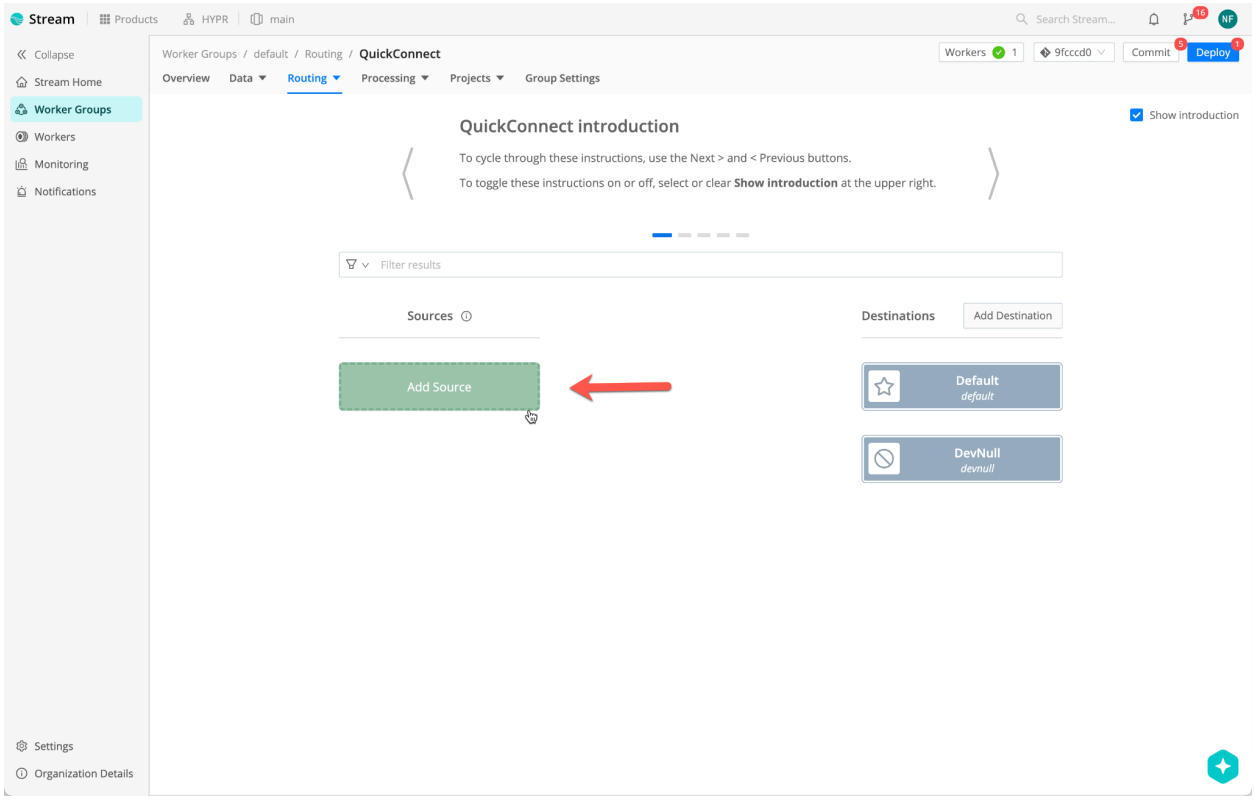
HYPR can be configured to send events to Cribl using HYPR's custom event hook (Event Hook: Custom) functionality. In this case, Cribl is configured to receive data over HTTP/S and HYPR is configured to send event data to the Cribl endpoint.

see https://docs.cribl.io/stream/sources-raw-http/
see free trial at https://cribl.io/try-cribl/

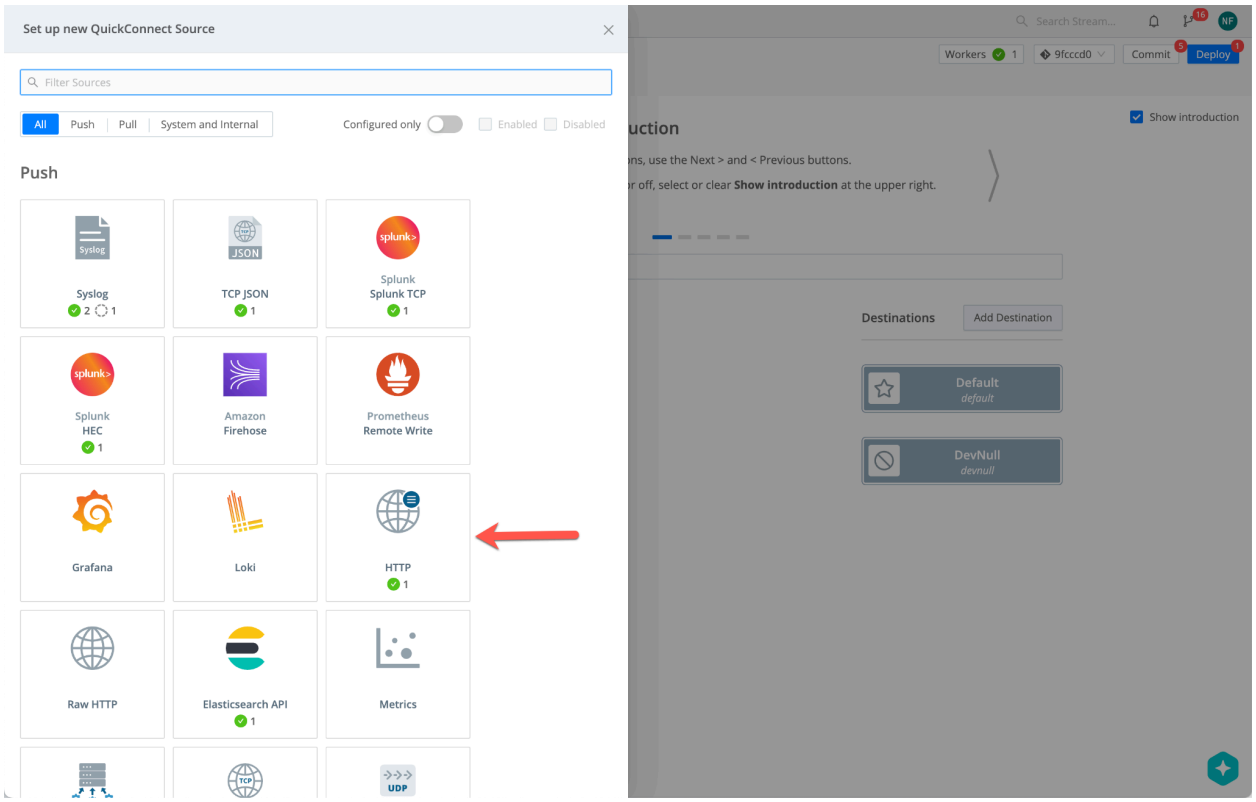
Setting up Cribl

Before HYPR can be configured, you must first set up Cribl to receive data over HTTP/S. Cribl provides documentation for this procedure in their product documentation. This section provides some example screenshots for configuring Cribl.

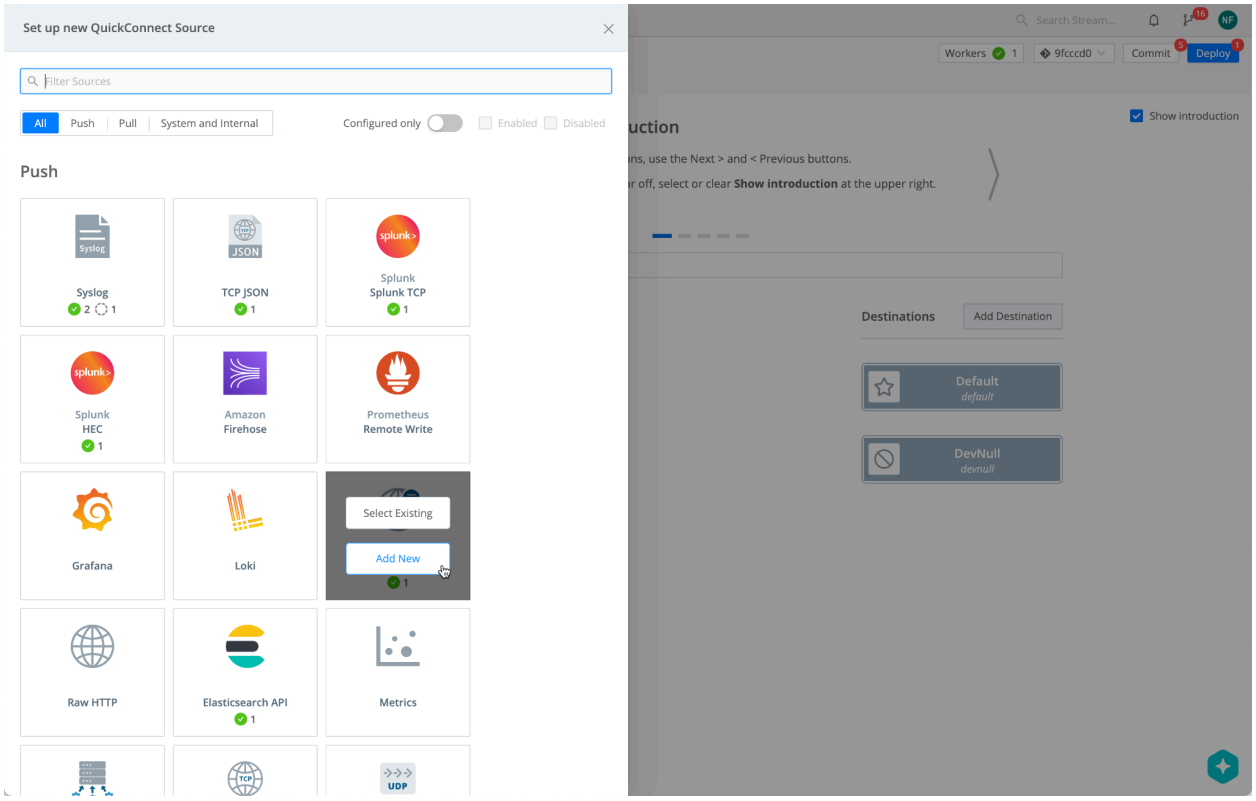
Login to your Cribl tenant and click Worker Groups in the left-hand menu and then click on your desired worker group (you may only have one called "default"). To configure via QuickConnect, navigate to **Routing** > **QuickConnect** (Stream).



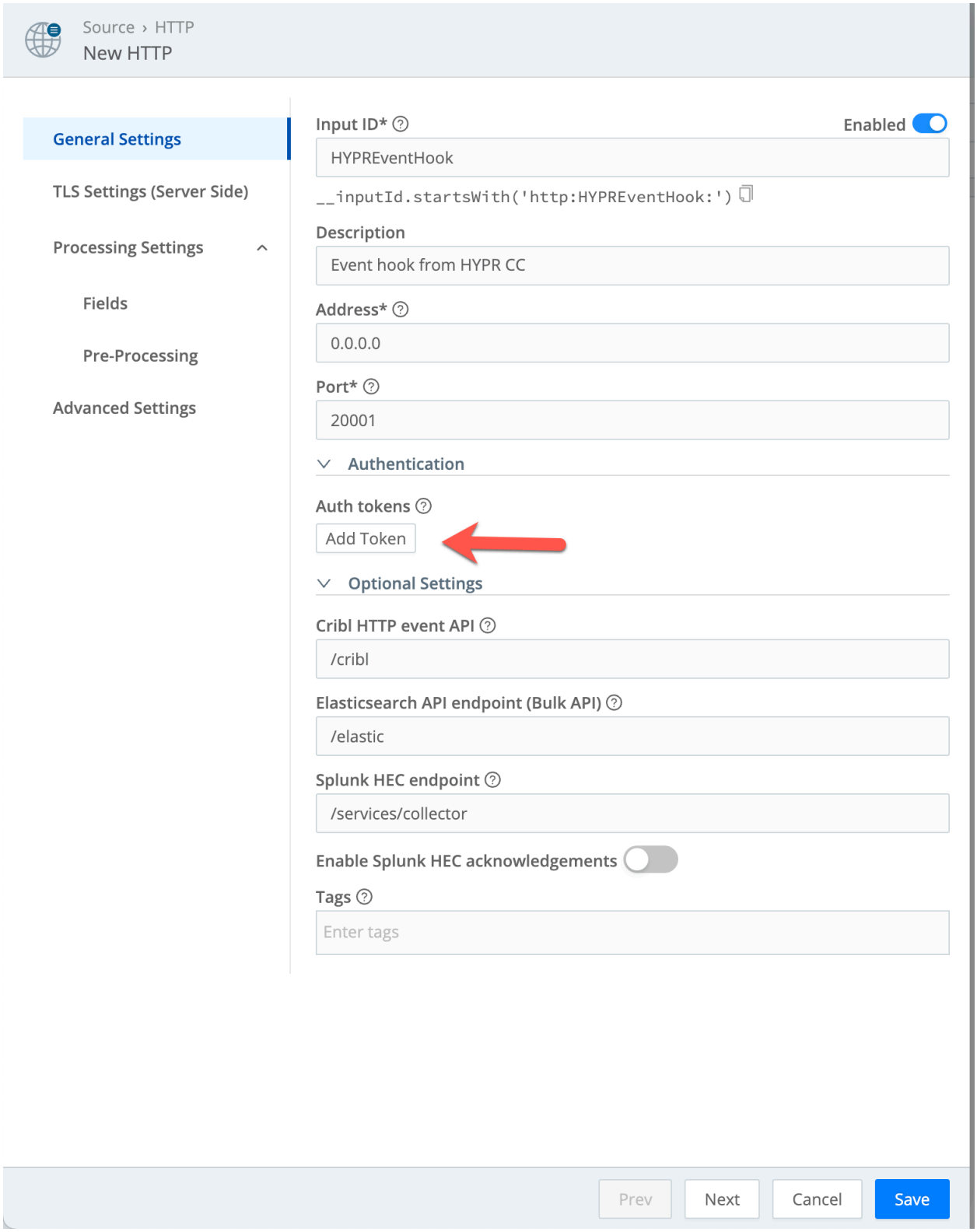
Click **Add Source** and you will be presented with a list of available sources.



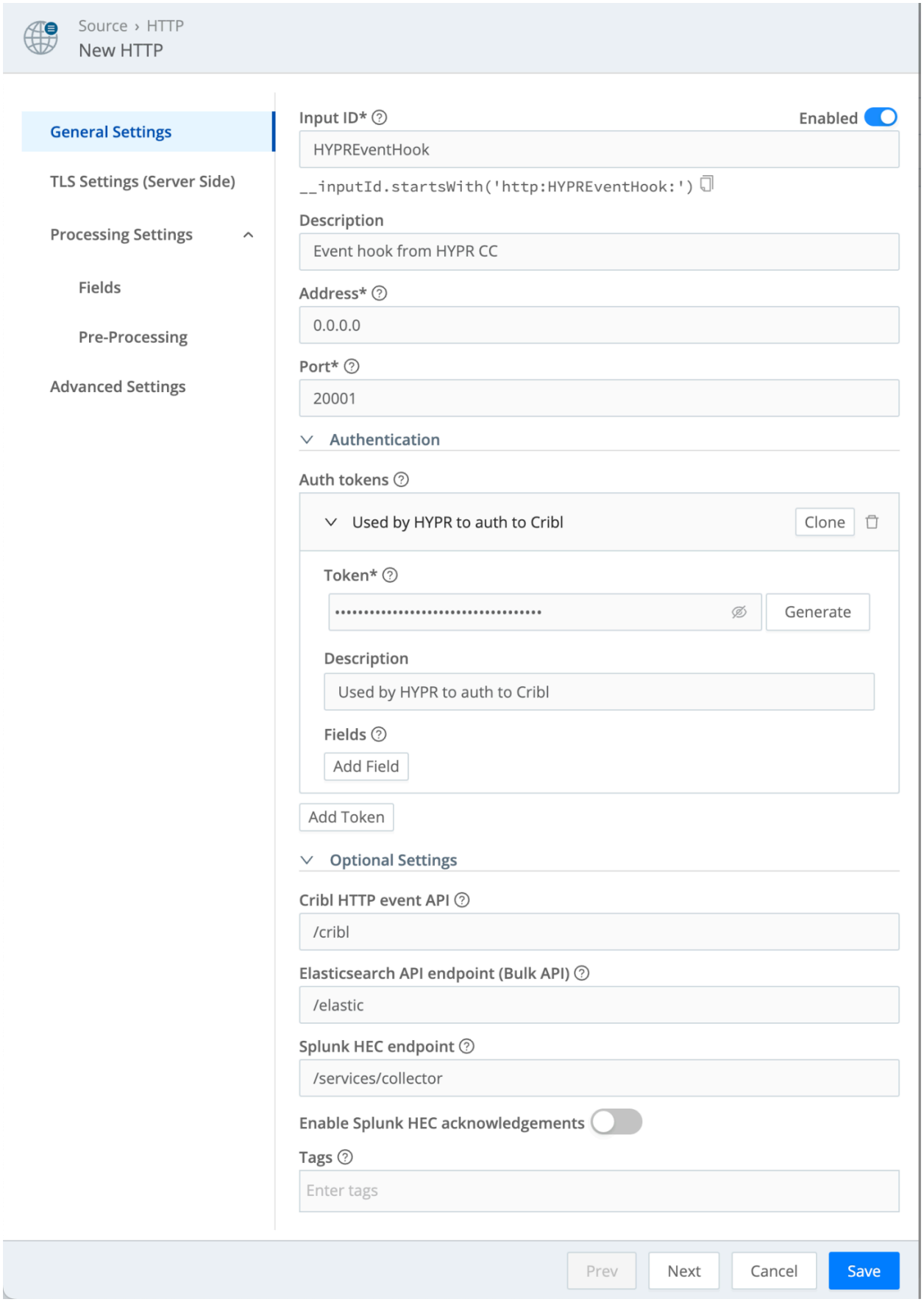
Locate the HTTP source, click it and select Add New



On the General Settings tab, give it a Name, Description and Port. Click on Add Token button to create an access token.



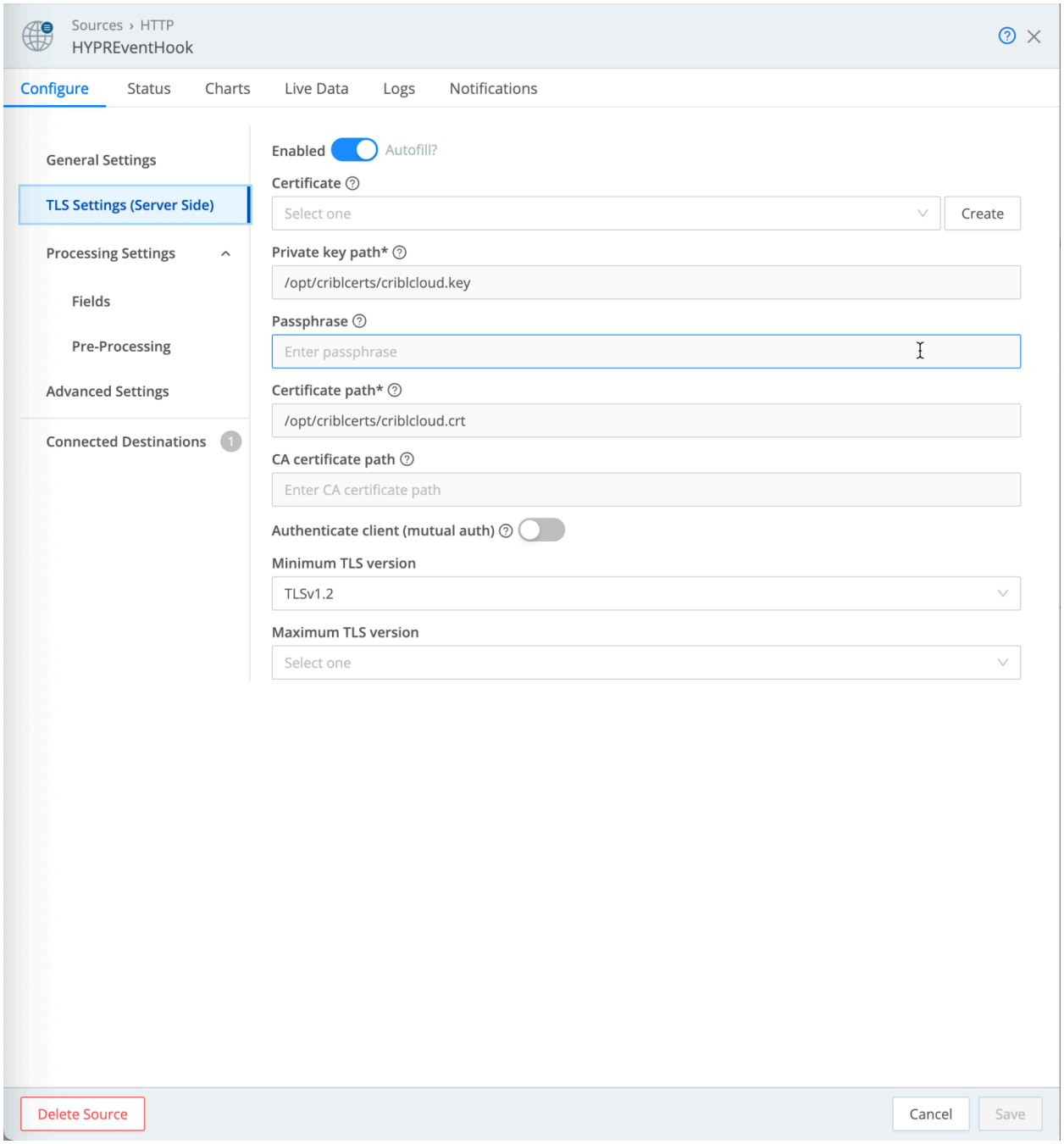
Generate a token and record it in a secure location such as a password manager. You can also enter a description of the token.



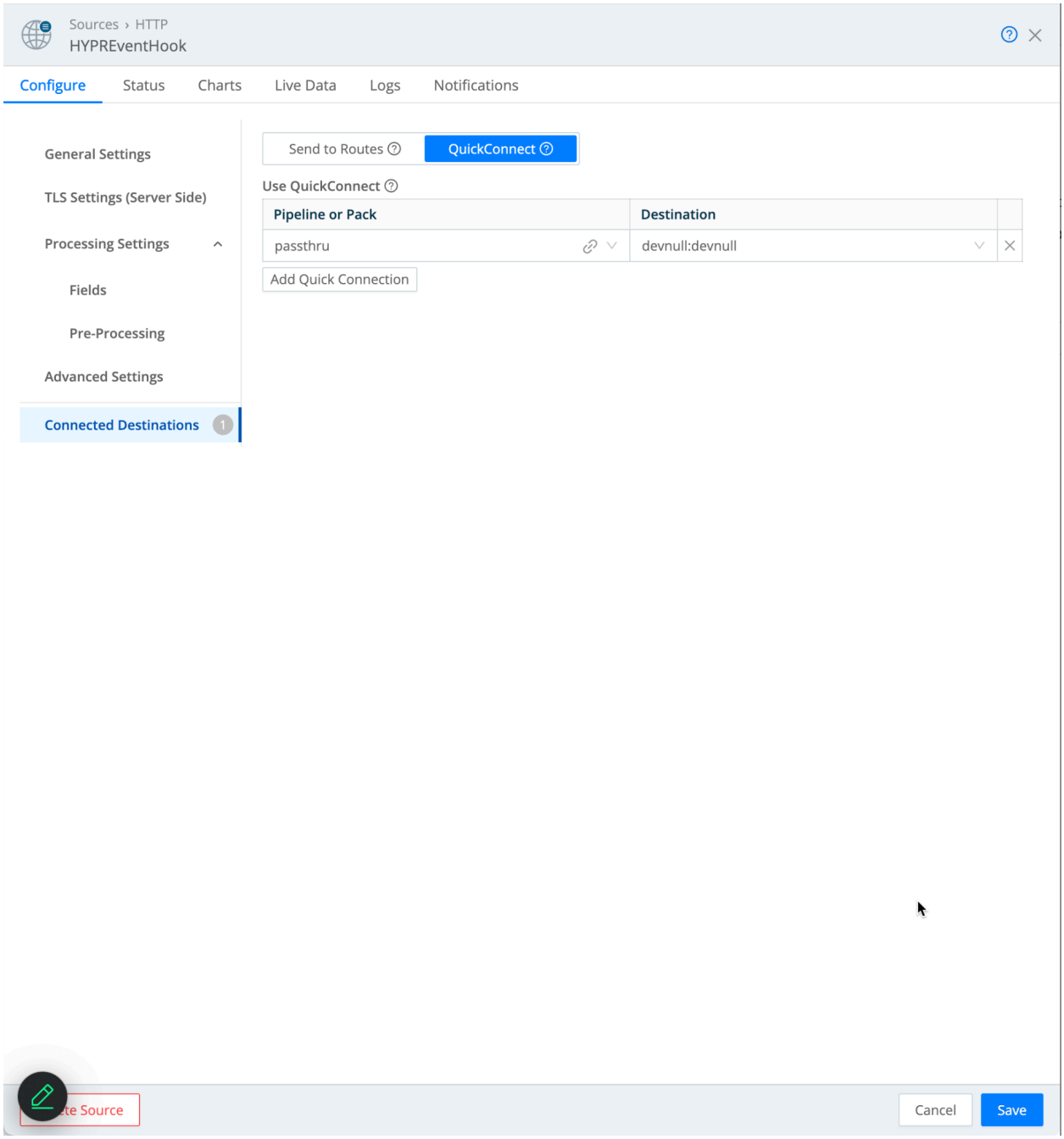
Next select the TLS settings tab. You can use the default Cribl server certificate or configure your own. In this example, we use the default Cribl certificate. Enter the following values in the form

Name	**Value**
Private key path	/opt/criblcerts/criblcloud.key
Certificate path	/opt/criblcerts/criblcloud.crt

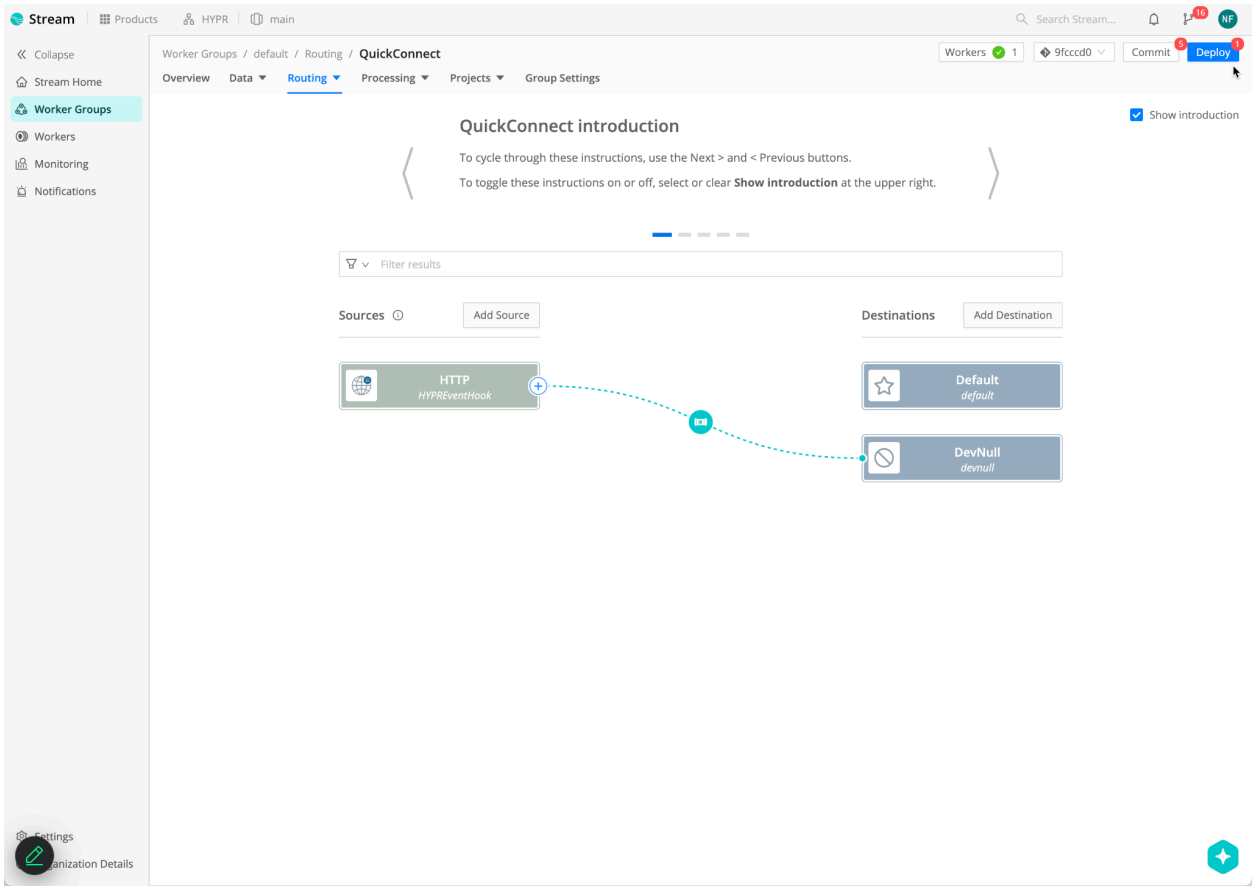
Leave the remaining values as default.



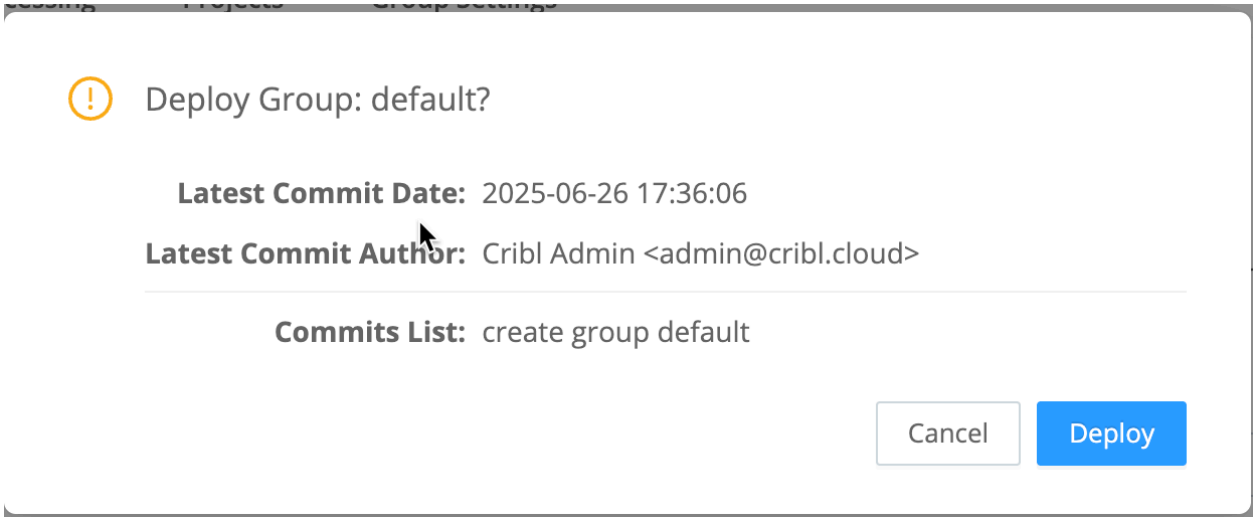
Next click on the Connected Destinations tab. In this example, we create a “passthu” pipeline to devnull.



Click Save to save the configuration.

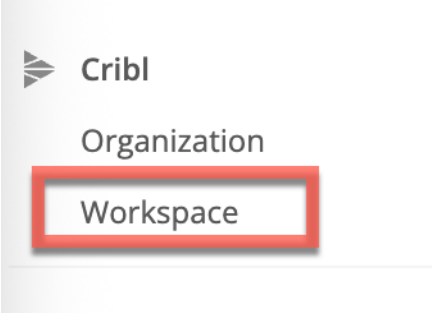


In the upper right corner, click the deploy button to push the changes out.

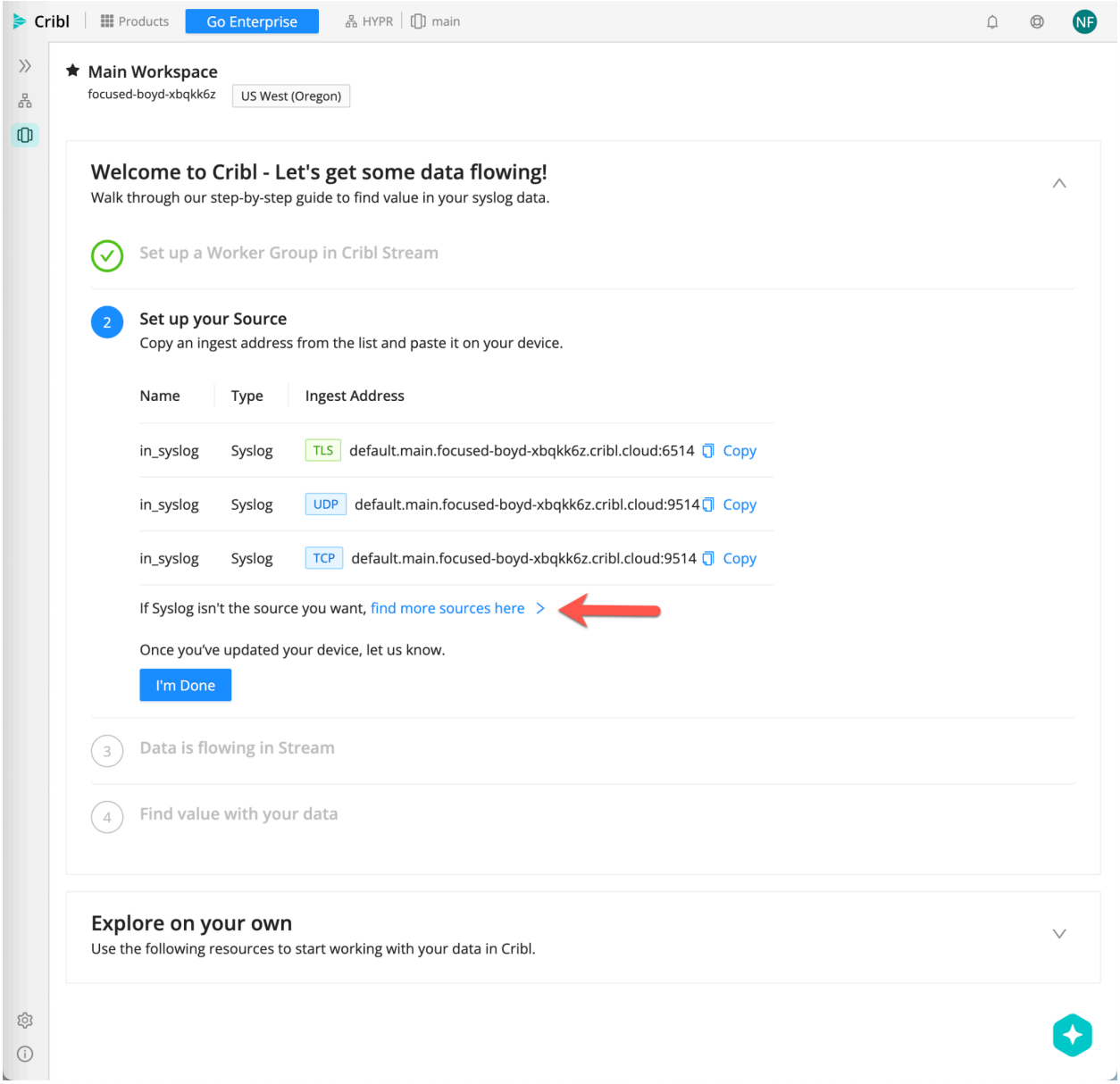


This completes the configuration.

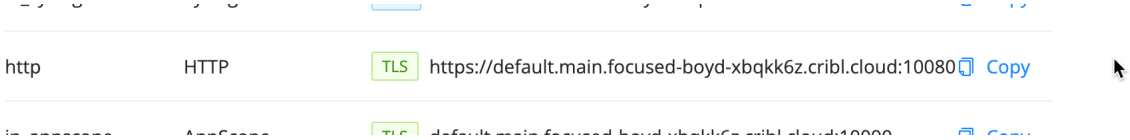
In order to configure HYPR, you will need to get the URL for the Cribl HTTP endpoint. At the top of the screen, click Products and then click Workspace



You will see a screen like this



Click “find more sources here”. Locate the HTTP URL and copy it



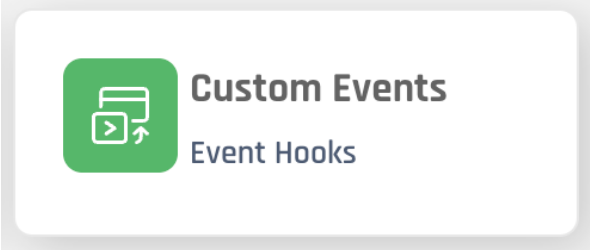
Put the URL in your notes and change the port value to the one you defined when creating the HTTP source (found in General Settings). You will need this URL and the access token you created to configure HYPR.

Configuring Event Hook in HYPR

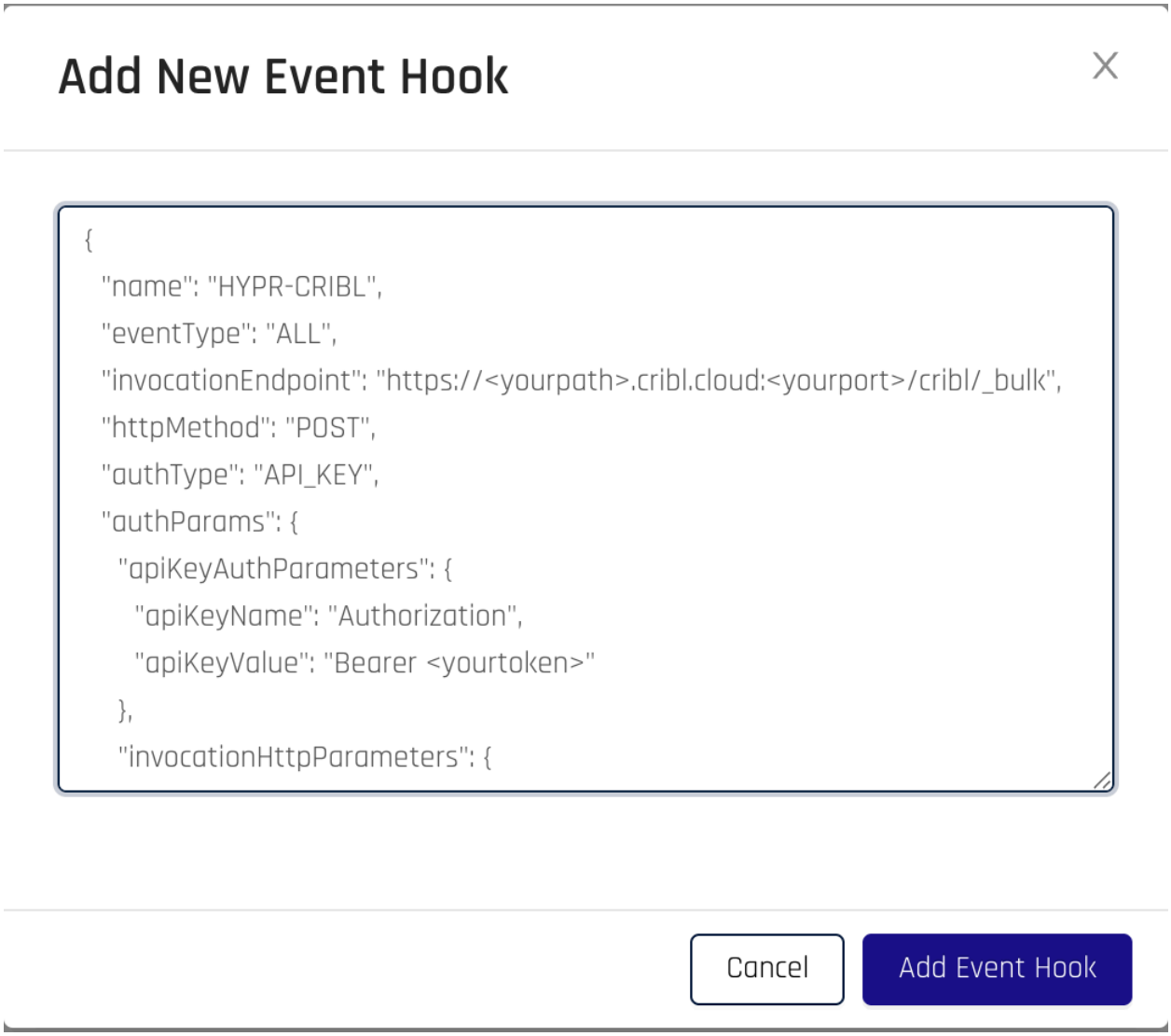
Follow the procedure for configuring a custom event hook in the Event Hook: Custom page in the HYPR documentation. Copy the below JSON data into a text editor and add your path, port and token.

```
{
  "name": "HYPR-CRIBL",
  "eventType": "ALL",
  "invocationEndpoint":
"https://<yourpath>.cribl.cloud:<yourport>/cribl/_bulk",
  "httpMethod": "POST",
  "authType": "API_KEY",
  "authParams": {
    "apiKeyAuthParameters": {
      "apiKeyName": "Authorization",
      "apiKeyValue": "Bearer <yourtoken>"
    },
    "invocationHttpParameters": {
      "headerParameters": [
        {
          "key": "Content-Type",
          "value": "application/json",
          "isValueSecret": false
        }
      ]
    }
  }
}
```

In the HYPR Control Center, navigate to Integrations. Click the **Add new integration** button and choose Custom Events.



Copy the JSON into the dialog box and click the **Add Event Hook** button



At this point new HYPR events will be sent to Cribl.