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 <u>Event Hook: Splunk</u> in the HYPR documentation.

Sending events to Crowdstrike

Crowstrike supports receiving data over an HTTP Event Data Collector, which can be found in their <u>online documentation</u>. As part of the event collector configuration, you must also <u>configure</u> <u>a parser</u> 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

≡	💉 Data connectors 🏻 Parse	rs 🛛		Q	Search		↓		*	C
	Data connectio	ns Dashboard	Parsers	Alerts (2)	Detection exclusions	Fleet management	Data setting:			
F	Parsers									
S	earch by name $\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \$	Type V Status V Cl	ear all ΰ					dd new	parser	2
P	Parser name	♀ Data source	1	Туре	Status	∧ Associate	d connectors			

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

Create new parser						
Parser name						
MyHYPRParser						
Blank template V						
Cancel	Create					

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

arser script ①	Generate parser ₊∳₊	Test data Failed: 3 + Add test Run tests 🖽
<pre>1 /* 2 # Log Parsing Template 2 # Log Parsing Template 3 This template implements Crowdstrike's Parsing Standard for Log nor 4 Reference: https://falcon.ics/accowdstrike.com/documentation/page/u05f69c9/ crowdstrike-parsing-standard 6 https://falcon.eu-l.crowdstrike.com/documentation/page/u05f69c9/ crowdstrike-parsing-standard 7 */ 8 9 //</pre>	malization. cg/ malization. cg/ malization. cg/ malization. cg/ tic.co/docs/reference/ r.crowdstrike.com/docs/ developer.crowdstrike. reference/ccs/ developer.crowdstrike. reference/ccs/ code malization. for, "message": "User iscard") /syntax-regex.html) [exampleSDID@32473 vent.outcome] observer.type	Show only failed tests 2018-10-15T12:51:40+00:00 [INF0] This is an example log entry. id=123 fruit=banana ! Failed to validate event against data schema. Please see the "Schema violations" tat for more information. and 1 more error 2018-10-15T12:52:42+01:30 [ERROR] Here is an error log entry. class=c.o.StringUtil fruit=pineapple ! Failed to validate event against data schema. Please see the "Schema violations" tat for more information. and 1 more error 2018-10-15T12:53:12+01:00 [INF0] User logged in. user_id=1831923 protocol=http ! Failed to validate event against data schema. Please see the "Schema violations" tat for more information. and 1 more error 2018-10-15T12:53:12+01:00 [INF0] User logged in. user_id=1831923 protocol=http ! Failed to validate event against data schema. Please see the "Schema violations" tat for more information. and 1 more error

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

Ξ 💉 Data connectors Parsers > MyHYPRParser □	Q Search		🔺 🛛 🖉 🛓 🕓
Edit parser			
Parser script ①	Generate parser +++	Test data Failed: 3 + Add	test 🛛 Run tests 🖭
1		Show only failed tests	
		2018-10-15T12:51:40+00:00 [INFO] This is an exa id=123 fruit=banana	mple log entry.
		! Failed to validate event against data schema. Please see the for more information and 1 more error	"Schema violations" tab
		2018-10-15T12:52:42+01:30 [ERROR] Here is an er class=c.o.StringUtil fruit=pineapple	ror log entry.
		! Failed to validate event against data schema. Please see the for more information. and 1 more error	"Schema violations" tab
		2018-10-15T12:53:12+01:00 [INFO] User logged in protocol=http	.user_id=1831923
		! Failed to validate event against data schema. Please see the for more information. and 1 more error	"Schema violations" tab
Fields to Cps.version Vendor ecs.version event.dataset event.kind event.module eve	ent.outcome observer.type		
tag:			
Cancel Save and exit			

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)
Your screen will now look like this:
                     🚍 💉 Data connectors 🛛 Parsers > MyHYPRParser 🔾
                                                                                                                                                                                                                                                               Q Search
                                                                                                                                                                                                                                                                                                                                                                                                                                                              🌲 🛛 🗶
                               Edit parser
                                                                                                                                                                                                                                         Generate parser +++
                                    Parser script ①
                                                                                                                                                                                                                                                                                                                                                                                                                                      + Add test
                                                                                                                                                                                                                                                                                                                                                                                                                                                                             Run tests 🗐
                                                                                                                                                                                                                                                                                                        Test data Failed:
                                                                                                                                                                                                                                                                                                          $ Show only failed tests
                                                       // #region NORMALIZATION
                                     38
                                     39
                                                        1-
                                     40
41
                                                        ********* Normalize additional fields from the JSON payload to CPS fields
                                                                                                                                                                                                                                                                                                        2018-10-15T12:51:40+00:00 [INFO] This is an example log entry.
                                                        ****
                                                                                                                                                                                                                                                                                                         id=123 fruit=banana
                                                                                                                                                                                            ********
                                     42
                                     43
44
45
46
47
48
                                                        // Core event fields
                                                                                                                                                                                                                                                                                                         ! Failed to validate event against data schema. Please see the "Schema violations" tab
                                                       // core control c
                                                                                                                                                                                                                                                                                                        for more information.
                                                                                                                                                                                                                                                                                                        2018-10-15T12:52:42+01:30 [ERROR] Here is an error log entry.
                                                        // Map additional vendor-specific fields for further context
                                                         // Map additional vendor-specific fields for further context
// Vendor.eventId := rename(Vendor.detail.data.id)
| Vendor.eventDataType := rename(Vendor.detail.data.type)
| Vendor.subName := rename(Vendor.detail.data.subName)
| Vendor.loggedTyme := rename(Vendor.detail.data.ventLoggedBy)
| Vendor.loggedTyme := rename(Vendor.detail.data.loggedTyme.imUTC)
| Vendor.loggedTyme := rename(Vendor.detail.data.toggedTyme.TUTC)
                                                                                                                                                                                                                                                                                                        class=c.o.StringUtil fruit=pineapple
                                     49
                                    50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
                                                                                                                                                                                                                                                                                                        ! Failed to validate event against data schema. Please see the "Schema violations" tab
                                                                                                                                                                                                                                                                                                         for more information.
                                                        Vendor.loggedTime := rename(Vendor.detail.data.loggedTimeTinTC)
Vendor.tenantId := rename(Vendor.detail.data.tenantId)
Vendor.traceId := rename(Vendor.detail.data.teraceId)
Vendor.traceId := rename(Vendor.detail.data.teraceId)
Vendor.rokpId := rename(Vendor.detail.data.teraceId)
Vendor.rokpId := rename(Vendor.detail.data.teraceId)
Vendor.rokpId := rename(Vendor.detail.data.teracId)
Vendor.traceId := rename(Vendor.detail.data.teracId)
Vendor.traceId := rename(Vendor.detail.data.teracId)
Vendor.teracId := rename(Vendor.detail.data.teraCid)
Vendor.teraceId := rename(Vendor.detail.data.teraCid)
Vendor.teraceId := rename(Vendor.detail.data.teraCid)
Vendor.teraceId := rename(Vendor.detail.data.teraCid)
Vendor.origin := rename(Vendor.detail.data.origin)
Vendor.origin := rename(Vendor.detail.data.login)
                                                                                                                                                                                                                                                                                                        2018-10-15T12:53:12+01:00 [INFO] User logged in. user_id=1831923
                                                                                                                                                                                                                                                                                                        protocol=http
                                                                                                                                                                                                                                                                                                         ! Failed to validate event against data schema. Please see the "Schema violations" tab
                                                                                                                                                                                                                                                                                                       for more information.
... and 1 more error
                                                         | Vendor.eventTags := rename(Vendor.detail.data.eventTags)
                                                       // Map outer-level fields (if present)
                                                         | Vendor.account := rename(Vendor.account)
| Vendor.region := rename(Vendor.region)
                                     69
                                    70
71
72
73
74
                                                          Vendor.dataSource := rename(Vendor.detail.dataSource)
                                                        Vendor.customerUuid := rename(Vendor.detail.customerUuid)
Vendor.tenantUuid := rename(Vendor.detail.tenantUuid)
Vendor.detailEventTags := rename(Vendor.detail.eventTags)
                                    Fields
                                                   Cps.version Vendor ecs.version event.dataset event.kind event.module event.outcome observer.type
                                     to
                                    tag:
                                   Cancel Save and exit
```

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

Status of	connections 🛈	Dat	a ingest				
 Active 		0					Next-Gen SIEM Oth
Idle		80 KB					 Avg. ingest per day (30 day moving average)
Error		60 KB		- 11			22.22 KB / 10 GB
Disconnect	ted	0 40 KB	L	- 1 I.			Avg. ingest per day lin
Pending		1 20 KB					Daily ingest
Paused		0	06/10 06/12 06	/14 06/16 06/18 06/	20 06/22 06/24 06/26 06/28 0	5/30 07/02 07/04 07/06 07/08	(Since 00:00 UTC)
Total connecti	ions	2					10.00 KB LODBY

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.

■ Mext-Gen SIEM Data onboarding > New connection □	Q Search]	ļ F	5 🐶 👗
Data connectors 1 items				
Filter by connector name : http 🔀 Vendor 🗸 Product 🗸 Connector type 🗸	Author \sim Subscription \sim Clear all			
Connector name	Connector type	Author		\$
		/ Witter	V Cabbonpton	Ý
HEC / HTTP Event Connector 🕒 Generic HEC	Push	Crowdstrike	Next-Gen SIEM	

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

Next-Gen SIEM	<u>Data onboarding</u> >	New connection]	Q Search	1			🌲 🛛 🗶 🔇
- Data connections								
ata connec	tors 1 item	IS						
Filter by connector na	me: http $ imes$ Vendo	or ~ Product ~	Connector type 🗸 Author	or V Subscription V	Clear all			
Connector na 🗘	Vendor ^	Product		Author 🗘	Subscription	New connection		·[]
HEC / HTTP Eve	🗻 Generic	HEC	Push	Crowdstrike	Next-Gen SIEM	HEC / HTTP Eve	nt Connector	Configure
						Vendor Generic	Product HEC	Connector type Push
						Author Crowdstrike	Parser name centrix-iot-json	Subscription Next-Gen SIEM
						Version v1.0.0		
						Description Ingesting data from a	any data source that uses th	e HTTP/HTTPS protocol with
result (1-1 shown)	Items per page 20	0 ~			Page 1 of 1 < >			

- 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 <u>Add a new parser</u>. For custom parser requirements, see <u>Understanding the CrowdStrike Parsing Standard</u>.

	A	Additional resources
		Learn how to set up the To learn more about this data connector, view our <u>documentation</u> .
		Learn how to set up the To learn more about this data connector, view our <u>documentation</u> .
		To learn more about this data connector, view our <u>documentation</u> .
		(i) Learn more about data connectors
		For general information about data connectors, see <u>data connectors documentation.</u>
		Learn more about parsers
		For more information about parsers, see <u>parser documentation.</u>
		Aggregate logs from different sources using our Logscale Collector. Install it by
		finding the appropriate Logscale Collector for your device on our <u>tool downloads list.</u>
Create new parser		
	Create new parser	Create new parser

- 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.

Connector setup in progress	$\boldsymbol{<}$
The connector is being configured to receive your data, however you will first need to enter an A key into Generic or an appropriate service in order to begin sending data. The API key will be generated shortly. C Learn more	¥ΡΙ
Close	

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.

1 This connector is ready to receive data. To begin sending data, select the Generate API key bu	utton and enter it into Generic or an appropriate service. Generate API key
← <u>Data connections</u>	1
My HYPR Tenant	
Pending	
2 Learn more about data connectors	Additional resources
Data details	Learn how to set up the HYPR Event Hook
Data source	To learn more about this data connector, view our <u>HYPR Event Hook documentation.</u>
HYPR Event Hook	
Last ingasted (UTC)	i Learn more about data connectors
-	For general information about data connectors, see data connectors documentation.
Total ingested amount in last 24 Hours	
0 B	$[]^{\prime}$ Learn more about parsers
	For more information about parsers, see parser documentation.
API authorization	
API URL	🕗 Download Logscale Collector
https://6cb97810a8174f48913b128eca269752.ingest.us-2.crowdstrike.com/services/collector	Aggregate logs from different sources using our Logscale Collector. Install it by
Connector detaile	finding the appropriate Logscale Collector for your device on our tool downloads list.
Connector details	
Connector name My HYPR Tenant	
, Anaratica ID	
6cb97810a8174f48913b128eca269752	
Description	
events from mytenant.hypr.com	
Descendent 1	
Parser details	
Parsars MvHVDDDarcar	

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.

Connection setup	\times
The system is ready to receive your data, however you will first need to enter this information into HEC / HTTP Event Co or an appropriate service in order to begin sending data. This API key will only be shown once. [3] Learn more	nnector
API key c77 7 79 79	Ъ
API URL https://6cb97810a8174f48913b128eca269752.ingest.us-2.crowdstrike.com/services/collector	G
Close	

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

Search Q Status Vendor Vendor Connector type Parser Subscription Clear all									
							+ Add	connecti	on ČĆ
Status	③ Connection name \$\$	Vendor	≎ Product		Parser	\$	Subscription	\$	Actions
Pending	My HYPR Tenant	🏊 Generic	HEC	Push	<u>MyHYPRParser</u>	12	Next-Gen SIEM		:

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 <u>Event Hook: Custom</u> page in the HYPR documentation. Copy the below JSON data into a text editor and add your path, port and token.

```
{
 "eventType": "ALL",
 "invocationEndpoint": "<API URL>/raw",
 "httpMethod": "POST",
 "authType": "API KEY",
 "authParams": {
    "apiKeyAuthParameters": {
      "apiKeyName": "CSAPIKEY"
   },
    "invocationHttpParameters": {
      "headerParameters": [
        {
          "key": "Authorization",
          "value": "Bearer <API key>",
          "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.

Add New Event Hook

Х

{	
"eventType": "ALL",	
"invocationEndpoint": "https://f31fabf97c49464986fc983c41301d9e.ingest.us-	
2.crowdstrike.com/services/collector/raw'',	
"httpMethod": "POST",	
"authType": "API_KEY",	
"authParams": {	
"apiKeyAuthParameters": {	
"apiKeyName": "CSAPIKEY"	
},	
"invocationHttpParameters": {	
"headerParameters": [

	Cancel	Add Event Hook
--	--------	----------------

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 (<u>Event Hook:</u> <u>Custom</u>) 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 <u>https://docs.cribl.io/stream/sources-raw-http/</u> see free trial at <u>https://cribl.io/try-cribl/</u>

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 <u>product documentation</u>. 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).

📚 Stream 🛛 🏢 Produc	ts 🖧 HYPR 🛛 🕕 main		Q Search Stream	Q 🕬 🐠
≪ Collapse	Worker Groups / default / Routing / QuickConnect		Workers 🥝 1 🔷 9fcccd0 🗸	Commit Deploy
습 Stream Home	Overview Data 🔻 Routing 🔻 Processing 👻 Projects 👻 Group Setti	ings		
Worker Groups Workers	QuickConnect i	ntroduction		Show introduction
ျက္က Monitoring	To cycle through these	instructions, use the Next > and < Previous buttons.	\	
실 Notifications	To toggle these instruct	ions on or off, select or clear Show introduction at the upper right.)	
	¥.		*	
	$\Psi \lor$ Filter results			
	Sources ①	Destinations	Add Destination	
	Add Source		Default	
	Add source @	Ш	default	
			DevNull	
			devnull	
Settings				
Organization Details				•

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.



Source > HTTP

New HTTP		
General Settings	Input ID* ⑦	Enabled 🔵
TLS Settings (Server Side)	inputId.startsWith('http:HYPREventHook:')	
Processing Settings	Description Event hook from HYPR CC	
Fields	Address* ⑦	
Pre-Processing	0.0.0.0	
Advanced Settings	Port* ⑦ 20001	
	 Authentication Auth tokens ⑦ Add Token Optional Settings Cribl HTTP event API ⑦ /cribl Elasticsearch API endpoint (Bulk API) ⑦ /elastic Splunk HEC endpoint ⑦ /services/collector Enable Splunk HEC acknowledgements ① Tags ⑦ Enter tags 	
	Prev Next Can	cel Save

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



Source > HTTP

eneral Settings	Input ID* ⑦	Enabled
	HYPREventHook	
S Settings (Server Side)	inputId.startsWith('http:HYPREventHook:')	J
cossing Cottings	Description	
ocessing settings	Event hook from HYPR CC	
Fields	Address* ⑦	
Pre-Processing	0.0.0.0	
The Processing	Port* ⑦	
vanced Settings	20001	
	✓ Authentication	
	Auth tokens ②	
	 Used by HYPR to auth to Cribi 	Cione
	Token* ⑦	
	•••••	Ø Generate
	Description	
	Used by HYPR to auth to Cribl	
	Fields ②	
	Add Field	
	Add Token	
	✓ Optional Settings	
	Cribl HTTP event API ⑦	
	/cribl	
	Elasticsearch API endpoint (Bulk API) ⑦	
	/elastic	
	Splunk HEC endpoint ⑦	
	/services/collector	
	Enable Splunk HEC acknowledgements	
	Tags ⑦	
	Enter tags	



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.

Sources > HTTP HYPREventHook			⑦ ×
Configure Status Charts	Live Data Logs Notifications		
General Settings TLS Settings (Server Side)	Enabled Autofill? Certificate @		Create
Processing Settings ^ Fields Pre-Processing Advanced Settings	Private key path* ⑦ /opt/criblcerts/criblcloud.key Passphrase ⑦ Enter passphrase Certificate path* ⑦ /opt/criblcerts/criblcloud.crt	I	
	CA certificate path ③ Enter CA certificate path Authenticate client (mutual auth) ③ Minimum TLS version TLSv1.2		v
	Maximum TLS version Select one		\vee
Delete Source		Cancel	Save

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

Sources > HTTP HYPREventHook					⊘ ×
Configure Status Charts	Live Data Logs	Notifications			
General Settings TLS Settings (Server Side) Processing Settings ^	Send to Routes ⑦ Use QuickConnect ⑦ Pipeline or Pack passthru	QuickConnect \odot	Destination devnull:devnull	v	×
Fields Pre-Processing Advanced Settings Connected Destinations	Add Quick Connection				
				ħ	
te Source				Cancel	Save

Click Save to save the configuration.

HYPR THE IDENTITY ASSURANCE COMPANY



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

(!)	Deploy Group: default	?		
	Latest Commit Date: Latest Commit Author:	2025-06-26 17:36:06 Cribl Admin <admin@cribl.clo< th=""><th>ud></th><th></th></admin@cribl.clo<>	ud>	
	Commits List:	create group default		
			Cancel	Deploy

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

http	HTTP	TLS https://default.main.focused-boyd-xbqkk6z.cribl.cloud:10080 Copy	k
:	A == C == = =		

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 <u>Event Hook: Custom</u> 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

Add New Event Hook

Х

{	
"name": "HYPR-CRIBL",	
"eventType": "ALL",	
"invocationEndpoint": "https:// <yourpath>.cribl.cloud:<yourport>/cribl/_bulk",</yourport></yourpath>	
"httpMethod": "POST",	
"authType": "API_KEY",	
"authParams": {	
"apiKeyAuthParameters": {	
"apiKeyName": "Authorization",	
"apiKeyValue": "Bearer <yourtoken>"</yourtoken>	
},	
"invocationHttpParameters": {	,
	1

	Cancel	Add Event Hook
L		

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