Rubrik Security Cloud:
Google SecOps SIEM Integration

User Guide

Version: 1.0

Table of Contents

Overview

This document contains instructions for ingesting Rubrik Security Cloud data into a Google SecOps SIEM through webhooks, viewing dashboards, creating correlation rules and search queries.

Google SecOps SIEM Platform

Google SecOps SIEM is a cybersecurity telemetry platform for threat hunting and threat intelligence and is part of the Google Cloud Platform. This platform stores log events it receives in either the original raw log or in a structured Unified Data Model (UDM) log. The Unified Data Model (UDM) defines the schema for parsing and Configuration Based Normalizers (CBN) describe how to log data that is transformed to the UDM schema.

Rubrik Security Cloud Platform

Rubrik Security Cloud is a cloud-native SaaS platform that provides a unified control plane for data management and protection across hybrid and multi-cloud environments. It centralizes visibility, policy enforcement, compliance, and threat detection through a single, API-driven interface.

Google SecOps Integration for Rubrik Security Cloud

This integration aims to enable seamless ingestion, parsing, and visualization of Rubrik Security Cloud data within Google SecOps SIEM. This integration will allow Google SecOps SIEM to receive real-time audit, threat hunt, threat monitor, anomaly and ransomware and security violation data from Rubrik Security Cloud using HTTP webhook, enriching the SIEM's threat detection and response capabilities.

As part of this integration, we will configure a webhook feed in Google SecOps for ingesting Rubrik Security Cloud detections, develop a parser to standardize this data for Google SecOps's use, and create dashboards, search queries and correlation rules to visualize the Rubrik Security Cloud insights within Google SecOps.

Prerequisites

Google SecOps

- Google SecOps SIEM instance and permissions to create a new feed.
- API Key credentials of the Google Cloud Platform on which Google SecOps SIEM instance is hosted.

Rubrik Security Cloud

- Access to a Rubrik Security Cloud instance to ingest data into Google SecOps SIEM.

Release Notes

v1.0

- Provided the parser that processes data ingested from the Rubrik Security Cloud platform and converts it into the Google SecOps UDM data model.
- Provided Rubrik Security Cloud dashboard for visualization which contains below panels:
    - Top 10 Logins by username
    - Login success/failure
    - Number of Anomaly detection alerts over time
    - Threat Monitoring matches over time
    - Number of Events by class
- Provided below detection rules:
    - Login failures for user exists
    - Login failures for same source ip and user not exists
    - Login failures for same user and user not exists
    - Ransomware presence detected
    - Threat Monitoring Matches
    - Critical Anomaly
    - Critical Severity Data Violation
- Provided below search queries:
    - Search Failed Login Attempts
    - Search Security Violations
    - Search Threat Monitoring
    - Search Anomaly Detection
    - Search Threat Hunting
    - Critical Severity Events
    - Warning Severity Events

  - Failed Threat Monitoring Events
  - Succeeded Threat Monitoring Events
  - Failed Threat Hunt Events
- Rubrik Security Cloud integration, we have introduced support for log parsing under the new **RUBRIK_SECURITY_CLOUD** label.

Ingest data from Rubrik Security Cloud into Google SecOps SIEM

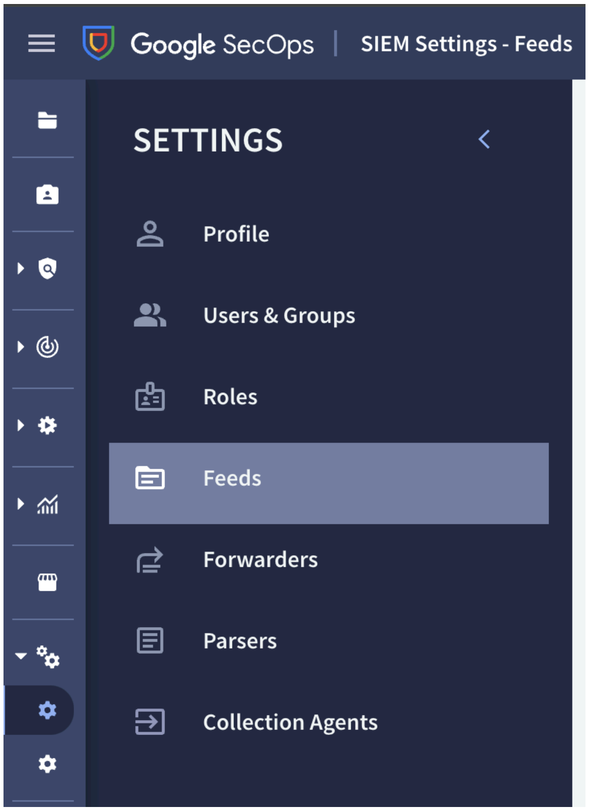
Complete the following steps to ingest data from Rubrik Security Cloud into Google SecOps SIEM:

1. Configure an HTTPS webhook feed in Google SecOps SIEM to receive detection data.
2. Configure a connection to Google SecOps SIEM and specify webhook parameters from Rubrik Security Cloud Instance.

Configure a WebHook Feed in Google SecOps

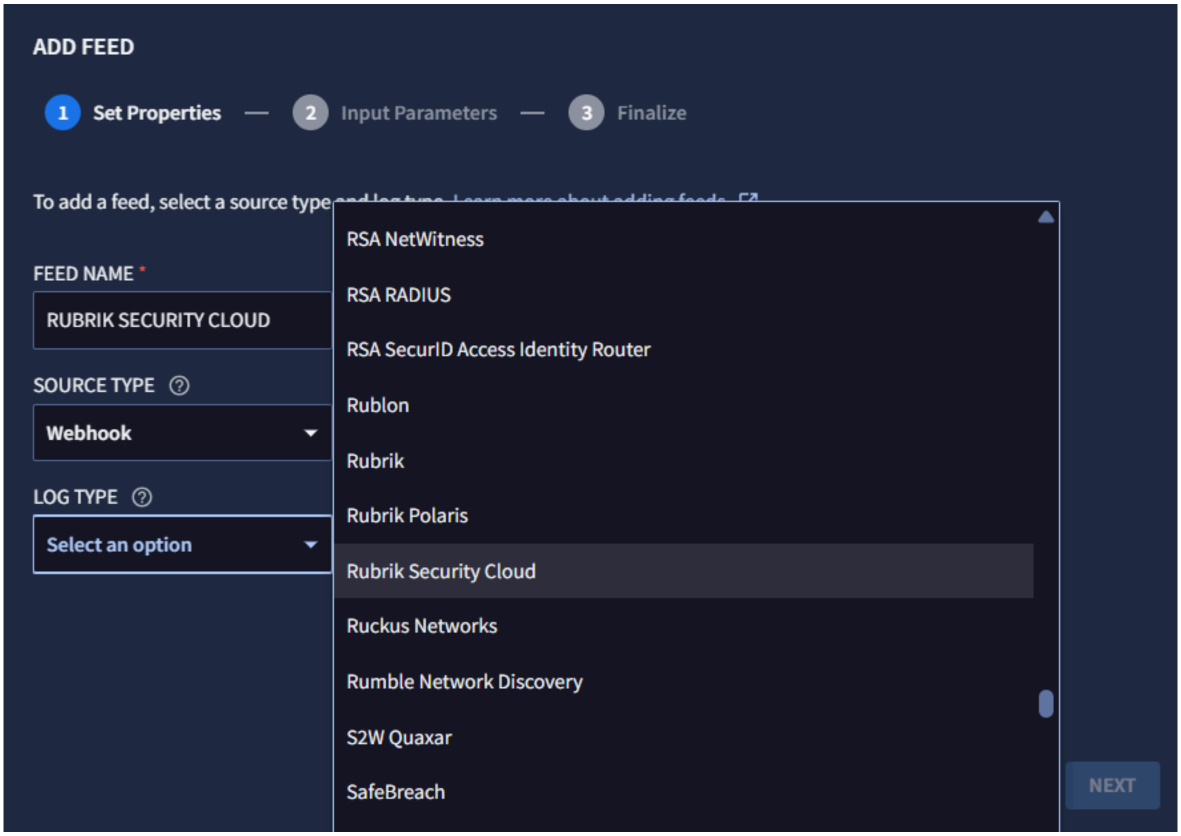
Step 1: Create an HTTPS webhook feed

1. In **Google SecOps**, go to **SIEM Settings > Feeds**.



Google SecOps: SIEM Settings

2. Click **Add New**.



ADD FEED

1 Set Properties — 2 Input Parameters — 3 Finalize

To add a feed, select a source type

FEED NAME *

RUBRIK SECURITY CLOUD

SOURCE TYPE

Webhook

LOG TYPE

Select an option

Google SecOps: Add Feed

3. Type a **Feed Name**.
4. Select **Webhook** from the **Source Type** list
5. Select **Rubrik Security Cloud** from the Log type list and then click **Next**.
6. Select the default input parameter values and then click **Next**.
7. Click **Submit**.
8. Click **Generate Secret Key** and then copy and store the key, which can only be viewed once.
9. From the **Details** tab, copy and save the field endpoint URL from the **Endpoint Information** field.
10. Click **Done.**

For more information, see Setup Feed of type - HTTPS Webhook.

Step 2: Create an API key for the webhook feed

An API key for the Google Cloud Platform is required to ingest events into Google SecOps SIEM through webhooks.

1. From the Google Cloud console, go to the Credentials page.
2. Click Create Credentials, and then select API Key.

3. Click Copy to copy the generated API key.

For more information, see Create API Keys in GCP and restrict access.

Configure a Webhook in Rubrik Security Cloud Instance

Prerequisites

- Users must have access to a Rubrik Security Cloud instance, and the required permissions to configure a Google SecOps webhook.

Configuration

- Login to Rubrik Security Cloud Instance.
- Go to **Settings** → **Integrations** → **Google SecOps**
- Click on the **Connect** button.
- Select either **SIEM** or **SIEM + SOAR** as the integration type.
- Fill the **Integration Name**. ex. GoogleSecOps.
- Click on **Next**.
- Select the **Service Account** to be used for the integration.
- Click on **Next**.
- Fill the **Endpoint** of Google SecOps through which events will be ingested.
- Fill the **API Key** generated in the GCP project associated with the SecOps instance.
- Fill the **Secret Key** which is generated while configuring the WebHook Feed in Google SecOps.
- Click on **Next**.
- Recommended **Event types** and **Severity Levels** are preselected.
- (If configuring a SIEM integration) Click on **Connect**.
- (If configuring a SIEM+SOAR integration) Click on **Next**.
- (If configuring a SIEM+SOAR integration) Steps for completing the SOAR Setup are shown.
- (If configuring a SIEM+SOAR integration) Click on **Connect**.
- Click on **Done**.

Parameters

When configuring a webhook in Rubrik Security Cloud, certain fields need to be populated. The details for each are mentioned as follows:

Name	**Description**	**Required**
Endpoint URL	The webhook endpoint where the data would be sent. This would be generated	Yes

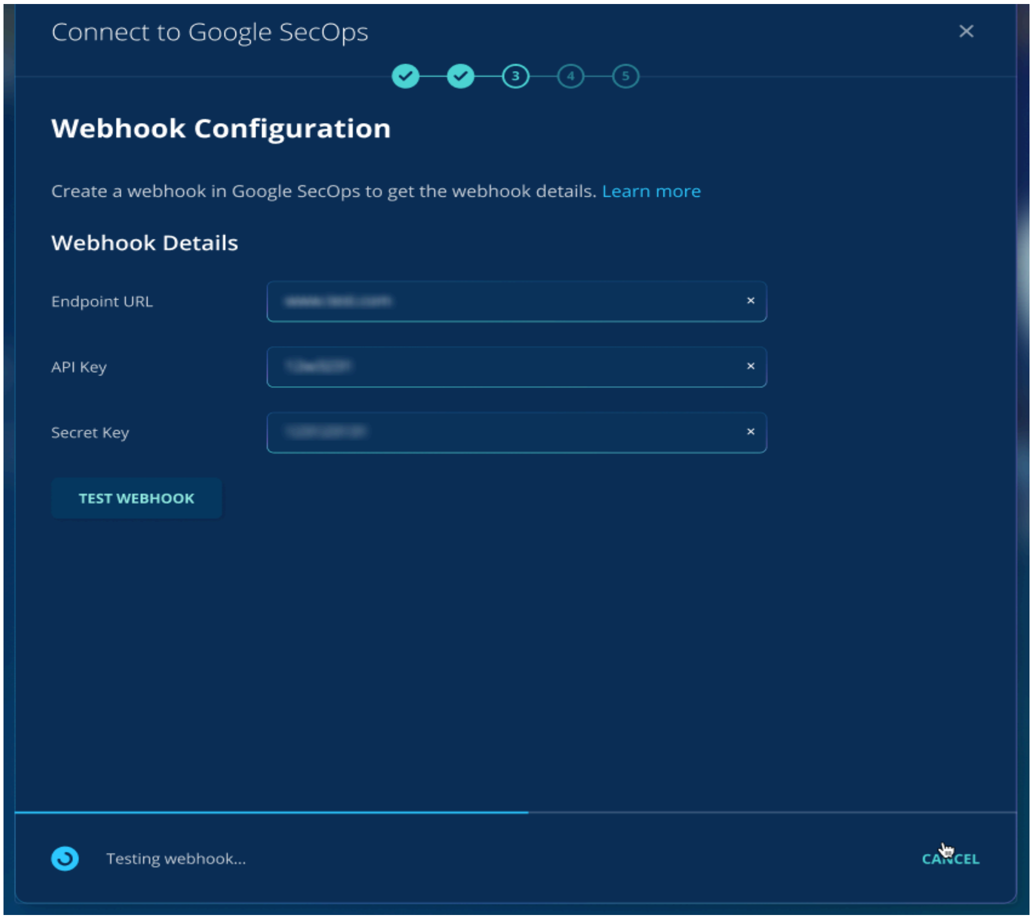
	during Configure a WebHook Feed in Google SecOps.	
API Key	API key generated in the GCP instance that hosts the SecOps instance.	Yes
Secret Key	The secret key which is generated during Configure a WebHook Feed in Google SecOps.	Yes

- The webhook can authenticate the connection to ingest data into SecOps:
  - **Enable Authentication with Headers (Recommended for security):** Add the API key and secret key in custom headers:

    X-goog-api-key = API_KEY

    X-Webhook-Access-Key = SECRET

Reference: Setup Webhook in Client Applications | SecOps



Connect to Google SecOps

Webhook Configuration

Create a webhook in Google SecOps to get the webhook details. Learn more

Webhook Details

Endpoint URL: [redacted]

API Key: [redacted]

Secret Key: [redacted]

Rubrik Security Cloud: Webhook Configuration

View Parsed Logs in Google SecOps SIEM

Complete the following steps to view ingested, parsed logs in Google SecOps SIEM:

1. Log in to Google SecOps:
    a. Open a web browser and navigate to the Google SecOps instance URL. For example: https://{test}.backstory.chronicle.security/
    b. Replace test with your actual Google SecOps instance name.
2. Access SIEM Search:
    a. From the top left corner of the Google SecOps console, select the "Investigation" option.
    b. Within the Investigation section, choose "SIEM Search".
3. Filter Events by Log Type:
    a. In the SIEM Search interface, locate the "Search" section.
    b. Apply a filter for the metadata field "log_type". Set the filter value to `metadata.log_type = "RUBRIK_SECURITY_CLOUD"`
4. Specify the time interval to search.
5. Click **Run Search**.
6. View Rubrik Security Cloud Events.

The search might take several minutes to complete depending on the data and time interval. Upon completion, UDM events are displayed on the Events tab. You can open UDM events to view mapped fields and the raw log data.

Mappings

UDM Field Name	Raw Log Field Name	Mapping Logic
metadata.vendor_name		This field is set to “RUBRIK”
metadata.product_name	source	If the source is present and not empty, set metadata.product_name to the value of the source field; otherwise, set it to "RUBRIK SECURITY CLOUD".
metadata.event_type		The system assigns event types based on class and attributes: Login with target → USER_LOGIN, Class is ThreatHunt with principal → SCAN_HOST, any class with principal → STATUS_UPDATE, otherwise → GENERIC_EVENT.
metadata.event_timestamp	timestamp	
metadata.product_event_type	custom_details.eventName	
metadata.product_log_id	custom_details.id	
metadata.url_back_to_product	custom_details.url	
principal.ip	custom_details.ipAddress, summary.ip_address	If custom_details.ipAddress does not exist or is empty then it will take ip_address from summary if it is present.
principal.resource.product_object_id	custom_details.clusterId	
principal.resource.name	custom_details.clusterName	
principal.resource.resource_type		This field is set to “CLUSTER”.
principal.administrative_domain	custom_details.customerID	
principal.domain.name	summary.domain	

principal.user.user_display_name	summary.user_name	If class is other than “Login” then only it will map to principal.user.user_display_name
principal.user.email_addresses	summary.user_email	If class is other than “Login” then only it will map to principal.user.email_addresses
target.location.name	custom_details.location	
target.resource.product_object_id	custom_details.objectId	
target.resource.name	custom_details.objectName, summary.object_name, summary.resource_name	If custom_details.objectName does not exist or empty then it will take object_name from summary if it is present. And if summary.object_name does not exist then it will take summary.resource_name.
target.resource.resource_subtype	custom_details.objectType, component	
target.resource.resource_type		Set to “UNSPECIFIED”
target.file.size	custom_details.logicalSizeInBytes	
target.user.userid	custom_details.auditUserId	
target.user.user_display_name	custom_details.auditUserName, summary.user_name	If class is “Login” and custom_details.auditUserName does not exist then only it will be mapped.
target.user.email_addresses	summary.user_email	If class is “Login” then only it will map to target.user.email_addresses
target.user.group_identifiers	summary.groups	
target.labels["client_id"]	summay.client_id	
target.labels["client_name"]	summay.client_name	
network.sent_bytes	custom_details.dataTransferredInBytes	
security_result.action	custom_details.status	If class is Login and

		custom_details.status is “Failure” then it will be set to “BLOCK” else if custom_details.status is “Failure” then it will be set to “FAIL” and otherwise it will be set to “UNKNOWN_ACTION”.
security_result.action_details	custom_details.status	
security_result.category	class	Classify security events into categories: ThreatMonitoring as "SOFTWARE_MALICIOUS", SecurityViolation as "POLICY_VIOLATION", failed Login as "AUTH_VIOLATION", and default to "UNKNOWN_CATEGORY" for all other events.
security_result.category_details	class	
security_result.severity	severity, summary.severity	If severity does not exist or is empty then it will take severity from summary if it is present.
security_result.severity_details	severity, summary.severity	If severity does not exist or is empty then it will take severity from summary if it is present.
security_result.description	custom_details.errorMessage	
security_result.summary	summary	
security_result.rule_name	summary.hunt_name	
security_result.rule_type	summary.hunt_type	
security_result.first_discovered_time	summary.hunt_date	
security_result.confidence_details	summary.confidence	
security_result.threat_feed_name	summary.threat_feed_type	
security_result.detection_fields["series_id"]	custom_details.seriesId, custom_details.seriesID	
security_result.detection_fields["error_id"]	custom_details.errorId	

security_result.detection_fields["error_code"]	custom_details.errorCode	
security_result.detection_fields["error_remedy"]	custom_details.errorRemedy	
security_result.detection_fields["error_reason"]	custom_details.errorReason	
security_result.detection_fields["remedy"]	summary.remedy	
security_result.detection_fields["obj_succeeded"]	summary.obj_succeeded	
security_result.detection_fields["obj_partially_succeeded"]	summary.obj_partially_succeeded	
security_result.detection_fields["obj_failed"]	summary.obj_failed	
security_result.detection_fields["object_matches"]	summary.object_matches	
security_result.detection_fields["file_matches"]	summary.file_matches	
security_result.detection_fields['hash_tf_version']	summary.hash_tf_version	
security_result.detection_fields["num_hash_matches"]	summary.num_hash_matches	
security_result.detection_fields["num_yara_rule_matches"]	summary.num_yara_rule_matches	
security_result.detection_fields["yara_tf_version"]	summary.yara_tf_version	
security_result.detection_fields["num_files_with_matches"]	summary.num_files_with_matches	
security_result.detection_fields["risk_name"]	summary.risk_name	
security_result.detection_fields["remediation_type"]	summary.remediation_type	
security_result.detection_fields["access_type"]	summary.access_type	

security_result.detection_fields["policy_name"]	summary.policy_name	
security_result.detection_fields["failed_document_count"]	summary.failed_document_count	
security_result.detection_fields["document_count"]	summary.document_count	
security_result.detection_fields["skipped_document_count"]	summary.skipped_document_count	
security_result.detection_fields["successful_document_count"]	summary.successful_document_count	
security_result.detection_fields["identity_name"]	summary.identity_name	
security_result.detection_fields["num_of_violating_identities"]	summary.num_of_violating_identities	
security_result.detection_fields["total_accessible_files_at_risk_count"]	summary.total_accessible_files_at_risk_count	
security_result.detection_fields["detection_time"]	summary.detection_time	
security_result.detection_fields["files_created_count"]	summary.files_created_count	
security_result.detection_fields["files_modified_count"]	summary.files_modified_count	
security_result.detection_fields["files_removed_count"]	summary.files_removed_count	
security_result.detection_fields["files_suspicious_count"]	summary.files_suspicious_count	
security_result.detection_fields["strain_name"]	summary.strain_name	
security_result.detection_fields["encryption_level"]	summary.encryption_level	
security_result.detection_fields["vm_count"]	summary.vm_count	
security_result.detection_fields["directories_unsnoozed"]	summary.directories_unsnoozed	

security_result.detection_fields["directories_snoozed"]	summary.directories_snoozed	
security_result.detection_fields["reason"]	summary.reason or summary.failure_reason	
security_result.detection_fields["action_date"]	summary.action_date	
security_result.detection_fields["{sub_field_name}"]	custom_details.eventInfo	All subfields of custom_details.eventInfo which are of type string or numeric will be added as key-value pairs.
additional.fields["user_note"]	custom_details.userNote	
additional.fields["effective_throughput"]	custom_details.effectiveThroughput	
additional.fields["is_polaris_audit"]	custom_details.isPolarisAudit	
additional.fields["org_id"]	custom_details.orgID	
additional.fields["group"]	group	
additional.fields["type"]	custom_details.type	
additional.fields["snappable_type"]	summary.snappable_type	
additional.fields["snappable_name"]	summary.snappable_name	
additional.fields["snapshot_date"]	summary.snapshot_date	
additional.fields["label_name"]	summary.label_name	
additional.fields["node"]	summary.node	
additional.fields["{sub_field_name}"]	custom_details.auditInfo	All subfields of custom_details.auditInfo which are of type string or numeric will be added as key-value pairs.
extensions.auth.type		Set to “UNSPECIFIED” if the class is Login.

Note: summary.{field} refers to the field extracted from the summary data.

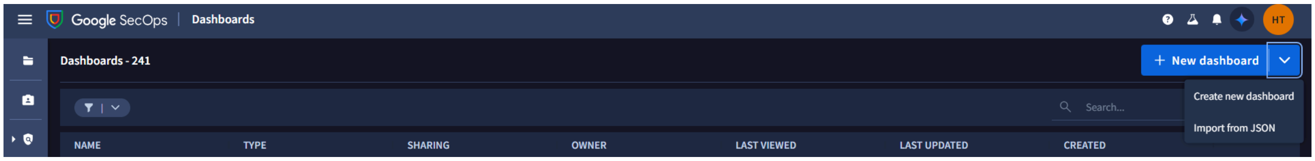
Dashboards

You can visualize and analyze ingested data by creating dashboards in Google SecOps SIEM.

Import Rubrik Security Cloud Dashboard into Google SecOps SIEM

Complete the following steps to import a dashboard:

1. Download the dashboard **.json** file for Rubrik Security Cloud from the following GitHub repository.
2. Go to **Dashboards & Reports** > **Dashboards,** Select **New Dashboard** > **Import from JSON**.



3. Click on **Upload Dashboard files** dialog, browse and select the appropriate JSON file.
4. Click **Edit** to update the name, description, and the dashboard access you're importing.
5. Click **Import** to import the dashboard.

See Import Dashboards into Google SecOps for more information.

Rubrik Security Cloud Dashboard

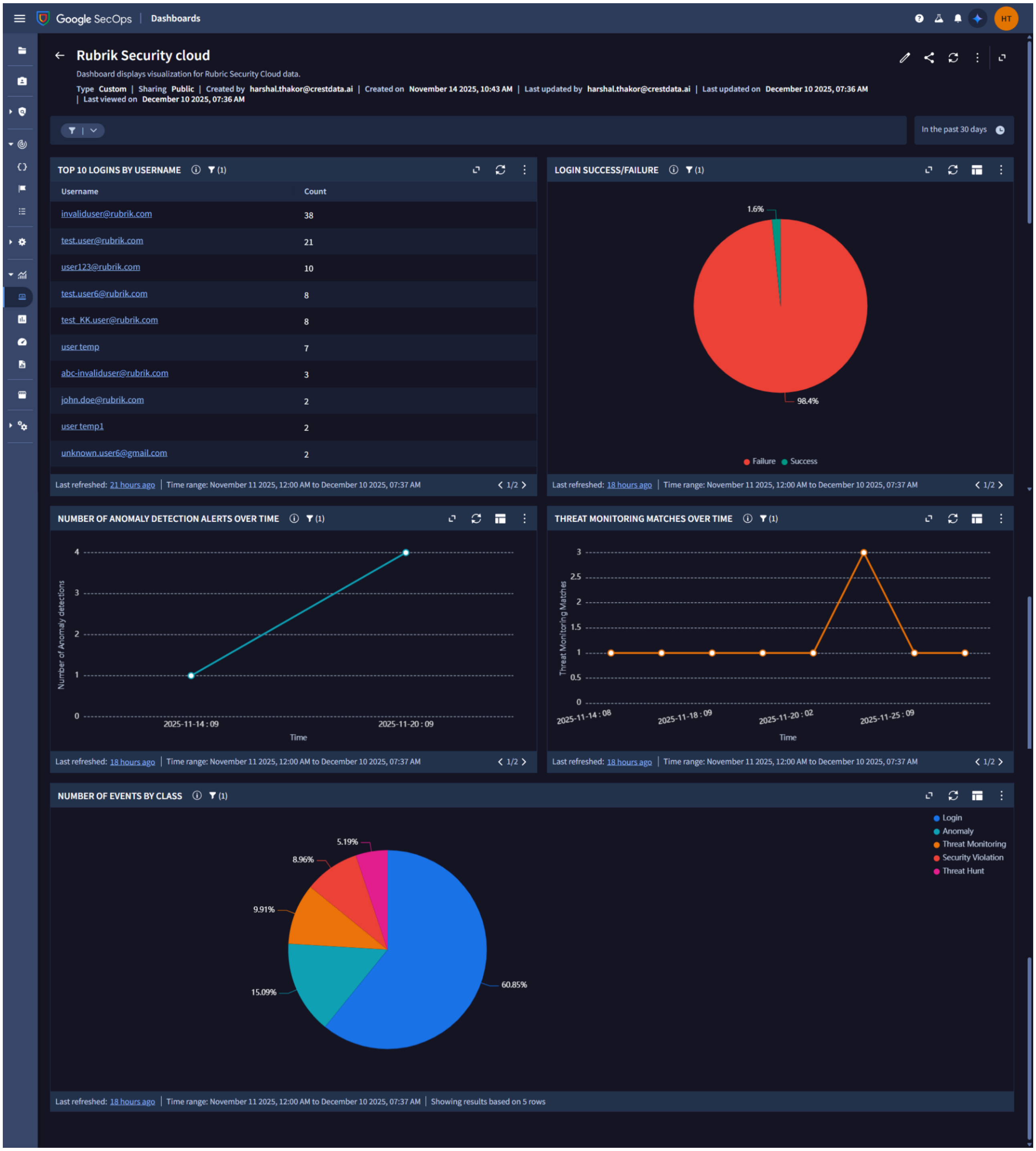
The Rubrik Security Cloud dashboard consists of the following panels:

Panel	Description	Type	Default Sorting
Top 10 Logins by username	Displays the top 10 number user Login attempts by username within the specified time range.	Table	The Count is in descending order
Login success/failure	Displays the number of login succeeded and failed within the specified time range	Pie chart	-
Number of Anomaly detection alerts over time	Displays the number of anomaly detection alerts over the time period.	Line chart	-
Threat Monitoring matches over time	Displays the number of threat monitoring matches alerts over the time period.	Line chart	-
Number of Events by class	Displays the total number of events by class name within the specified time range.	Pie chart	-

Available Filters

Name	Type	Description	Default
Time	Selector	A customized set of time periods that are available to filter data on the	Past 1 day

		dashboard.	



Google SecOps | Dashboards

Rubrik Security cloud

Dashboard displays visualization for Rubric Security Cloud data.

Type Custom | Sharing Public | Created by harshal.thakor@crestdata.ai | Created on November 14 2025, 10:43 AM | Last updated by harshal.thakor@crestdata.ai | Last updated on December 10 2025, 07:36 AM | Last viewed on December 10 2025, 07:36 AM

In the past 30 days

TOP 10 LOGINS BY USERNAME (1)

Username	Count
invaliduser@rubrik.com	38
test.user@rubrik.com	21
user123@rubrik.com	10
test.user6@rubrik.com	8
test_KK.user@rubrik.com	8
user temp	7
abc-invaliduser@rubrik.com	3
john.doe@rubrik.com	2
user temp1	2
unknown.user6@gmail.com	2

Last refreshed: 21 hours ago | Time range: November 11 2025, 12:00 AM to December 10 2025, 07:37 AM 1/2

LOGIN SUCCESS/FAILURE (1)

Last refreshed: 18 hours ago | Time range: November 11 2025, 12:00 AM to December 10 2025, 07:37 AM 1/2

NUMBER OF ANOMALY DETECTION ALERTS OVER TIME (1)

Last refreshed: 18 hours ago | Time range: November 11 2025, 12:00 AM to December 10 2025, 07:37 AM 1/2

THREAT MONITORING MATCHES OVER TIME (1)

Last refreshed: 18 hours ago | Time range: November 11 2025, 12:00 AM to December 10 2025, 07:37 AM 1/2

NUMBER OF EVENTS BY CLASS (1)

Last refreshed: 18 hours ago | Time range: November 11 2025, 12:00 AM to December 10 2025, 07:37 AM | Showing results based on 5 rows

Google SecOps: Rubrik Security Cloud Dashboard

Create Correlation Rules for Detections and Alerts

Correlation rules scan events ingested into the Google SecOps SIEM to generate detections and alerts for specified anomalies. Rubrik Security Cloud provides 7 rules that you can modify or copy to get started. You can find the Rubrik Security Cloud correlation rules in the following GitHub repository.

Create a new correlation rule

1. From Google SecOps SIEM, navigate to **Detections > Rules & Detections**.
2. From the **Rules Editor** tab, click **New**.
3. In the rule editor, clear all the contents, and then copy and paste the code from the GitHub repository.
4. Click **Save New Rule**.
5. To generate alerts from the correlation rule, click the three dots next to the rule name, and enable the **Alerting** option.

See Manage rules using Rules Editor for more information.

Here are some considerations while creating correlation rules for Rubrik Security Cloud:

- Events with the event name “PasswordLoginFailedUnknownUser” are treated as non-existing user login attempts.
- Events with the event name “PasswordLoginFailedKnownUser” are treated as existing user login attempts.

Create Search Queries

After collecting the data into Google SecOps in the form of events, the user can see the specific events from predefined search queries. You can find the Rubrik Security Cloud Search Queries in the following GitHub repository.

Create and Execute a new Search Query

1. From Google SecOps SIEM, navigate to **Investigation > SIEM Search**.
2. Go to **Search Manager,** click on **+** icon**.**
3. Copy and paste the Search Query in **UDM SEARCH,** Title in **Title** and Description in **Description** from the Search Queries listed below.
4. Click on **SAVE EDITS.**
5. Click on **LOAD SEARCH**.
6. Select **Time Range.**
7. Click on **Run Search** to execute the search query.

See Google SecOps: Saved Searches for more information.

Search Queries

Title	**Search Query**	**Description**

Rubrik Security Cloud - Search Failed Login Attempts	metadata.log_type = "RUBRIK_SECURITY_CLOUD"
	metadata.event_type = "USER_LOGIN"
	security_result.action_details = "Failure"
		List all events where the login attempt Failed.

Rubrik Security Cloud - Search Security Violations	metadata.log_type = "RUBRIK_SECURITY_CLOUD"
	security_result.category_details = "SecurityViolation"
		List all the events where class is "SecurityViolation".

Rubrik Security Cloud - Search Threat Monitoring	metadata.log_type = "RUBRIK_SECURITY_CLOUD"
	security_result.category_details = "ThreatMonitoring"
		List all the events where class is "ThreatMonitoring".

Rubrik Security Cloud - Search Anomaly Detection	metadata.log_type = "RUBRIK_SECURITY_CLOUD"
	security_result.category_details = "Anomaly"
		List all the events where class is "Anomaly".

Rubrik Security Cloud - Search Threat Hunting	metadata.log_type = "RUBRIK_SECURITY_CLOUD"
	security_result.category_details = "ThreatHunt"
		List all the events where class is "ThreatHunt".

Rubrik Security Cloud - Critical Severity Events	metadata.log_type = "RUBRIK_SECURITY_CLOUD"
	security_result.severity_details = "critical"
		List all the events with critical severity.

Rubrik Security Cloud - Warning Severity Events	metadata.log_type = "RUBRIK_SECURITY_CLOUD"
	security_result.severity_details = "warning"
		List all the events with warning severity.

Rubrik Security Cloud - Failed Threat Monitoring Events	metadata.log_type = "RUBRIK_SECURITY_CLOUD"
	security_result.category_details = "ThreatMonitoring"
	security_result.action_details = "Failure" or security_result.action_details = "TaskFailure"
		List events where class is “ThreatMonitoring” and status is “Failure” or "TaskFailure".

Rubrik Security Cloud - Succeeded Threat Monitoring Events	metadata.log_type = "RUBRIK_SECURITY_CLOUD"
	security_result.category_details = "ThreatMonitoring"
	security_result.action_details = "Success" or security_result.action_details = "TaskSuccess"
		List events where class is “ThreatMonitoring” and status “Success” or “TaskSuccess”.

Rubrik Security Cloud - Failed Threat Hunt Events	metadata.log_type = "RUBRIK_SECURITY_CLOUD"
	security_result.category_details = "ThreatHunt"
	security_result.action_details = "Failure"
		List events where class is “ThreatHunt” and status is “Failure”.

Limitations

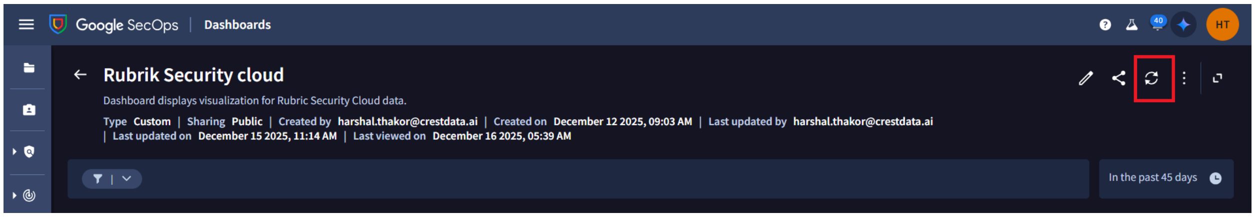
- It might take up to 30 minutes for the latest ingested data to be displayed in the dashboard.
- There can be a delay in creating detections when a live rule is running against ingested events.
- If a Rubrik Security Cloud log that is currently ingested by Google SecOps SIEM is updated and then ingested again, the log is considered a new event and is displayed as a new activity in the dashboard. This condition can result in inconsistent counts between Rubrik Security Cloud and the dashboard.
- In Google SecOps, raw log data is not directly accessible. This creates a limitation when building dashboards, as the visualization layer can only rely on processed or parsed data. As a result, the dashboards cannot always be fully populated with granular event details.
- All subfields of “custom_details.eventInfo” and “custom_details.auditInfo” that are of type string or numeric will be extracted by the parser. Subfields of any other data type will not be extracted by the parser.

Known Behaviours

- Newly ingested entities often take additional time to appear on the dashboard. This latency impacts real-time monitoring and reduces the effectiveness of dashboards for time-sensitive investigations. A support ticket was raised regarding this issue for further investigation and resolution.

Troubleshooting

- Events are successfully ingested but not displayed in search results:
    - The time range for the search is outside of the timestamp associated with the ingested event.
    - The ingested event might be a duplicate event with the same payload.
- Newly ingested events are not appearing on the dashboard:
    - This may be due to the dashboard not refreshing properly. Click the Refresh button in the top-right corner of the dashboard as shown in the screenshot below and check again.



Rubrik Security cloud

Dashboard displays visualization for Rubric Security Cloud data.

Type Custom | Sharing Public | Created by harshal.thakor@crestdata.ai | Created on December 12 2025, 09:03 AM | Last updated by harshal.thakor@crestdata.ai | Last updated on December 15 2025, 11:14 AM | Last viewed on December 16 2025, 05:39 AM

- The dashboard may be slow to load or unresponsive - This could be due to a problem with the data source being unavailable or having too much data, the query that is being used, or the way that the dashboard is being rendered.