Recorded Future for Google SecOps SOAR

Recorded Future, Inc

Version 12.0, 2026

Table of Contents

Chapter 1. Overview

Recorded Future has partnered with Google to deliver robust threat intelligence directly into Google SecOps SOAR (formerly Chronicle SOAR / Siemplify). The integration enriches indicators, manages alerts, automates investigation playbooks, and provides access to Recorded Future's sandbox and detection rule capabilities — all from within the SOAR platform.

1.1. Features

IOC Enrichment

Query Recorded Future for real-time intelligence on IPs, domains, file hashes, URLs, and CVEs. Each enrichment action returns a risk score, risk band label, and evidence details. When the **Include Links** parameter is enabled, actions also return related MITRE ATT&CK techniques and linked entities. Results can automatically mark SOAR entities as suspicious or malicious based on a configurable threshold.

Collective Insights

When enabled on enrichment actions, detections are contributed back to Recorded Future Collective Insights. This improves the quality of intelligence for all Recorded Future customers while preserving anonymity.

Classic Alerts

Ingest Recorded Future Classic Alerts into SOAR cases via the Classic Alerts Connector. Alerts are created from Recorded Future alerting rules (custom or IGL). The `Get Alert Details` and `Update Alert` actions allow analysts to fetch full alert context and update alerts from within a SOAR playbook.

Playbook Alerts

Ingest Recorded Future Playbook Alerts into SOAR cases. Playbook Alerts are curated alerts covering categories such as domain abuse, cyber vulnerabilities, identity exposures, and malware reports. Two connectors are available: one for initial ingestion and one that tracks lifecycle changes (priority increases, new assessments, entity additions, and reopened alerts).

Sandbox

Submit files and URLs to the Recorded Future Sandbox for dynamic analysis. `Search Hash Malware Intelligence` queries existing sandbox reports for file hashes already analyzed in the Recorded Future platform.

Detection Rules

Search for and retrieve Recorded Future detection rules (YARA, Sigma, and Snort) produced by the Insikt Group. `Search Detection Rules` supports filtering by type, title,

associated entity, and date. `Fetch Detection Rule` retrieves a specific rule by its Recorded Future document ID.

Analyst Notes

Add analyst notes directly to Recorded Future entities from within a SOAR playbook using `Add Analyst Note`. Notes are associated with the scope entities on the SOAR case and support topic categorization.

Lists

Create and manage Recorded Future lists from SOAR playbooks. Actions cover the full list lifecycle: create, fetch, add entities, remove entities, and retrieve status.

Entity Utilities

`Entity Lookup` retrieves Recorded Future entity details by Recorded Future ID. `Entity Match` resolves a free-text name to a Recorded Future entity ID, with optional type filtering.

1.2. Requirements

- A Recorded Future API token
- Google SecOps SOAR instance
- HTTPS connectivity to `api.recordedfuture.com`
- For `Detonate File` with GCP Bucket source: a configured Google Cloud Storage bucket accessible from the SOAR server

1.3. API Connectivity

The integration communicates exclusively with the Recorded Future API at `https://api.recordedfuture.com`. Outbound proxies are supported — configure proxy settings on each connector instance.

If the Sandbox API is configured, the integration also communicates with `https://sandbox.recordedfuture.com`. The sandbox URL and key are configured separately on the integration configuration screen.

Chapter 2. Configuration

2.1. Integration Setup

Install the integration from Google SecOps **Content Hub** under the **Response Integrations** section. Locate the integration listed as **RecordedFutureIntelligence** and install the **Community** edition.

The Recorded Future integration is listed as the **Community** edition in Content Hub. Install this edition to get access to the latest features and support. Do **not** install the Google Certified version — it is no longer supported and will be removed in a future release.

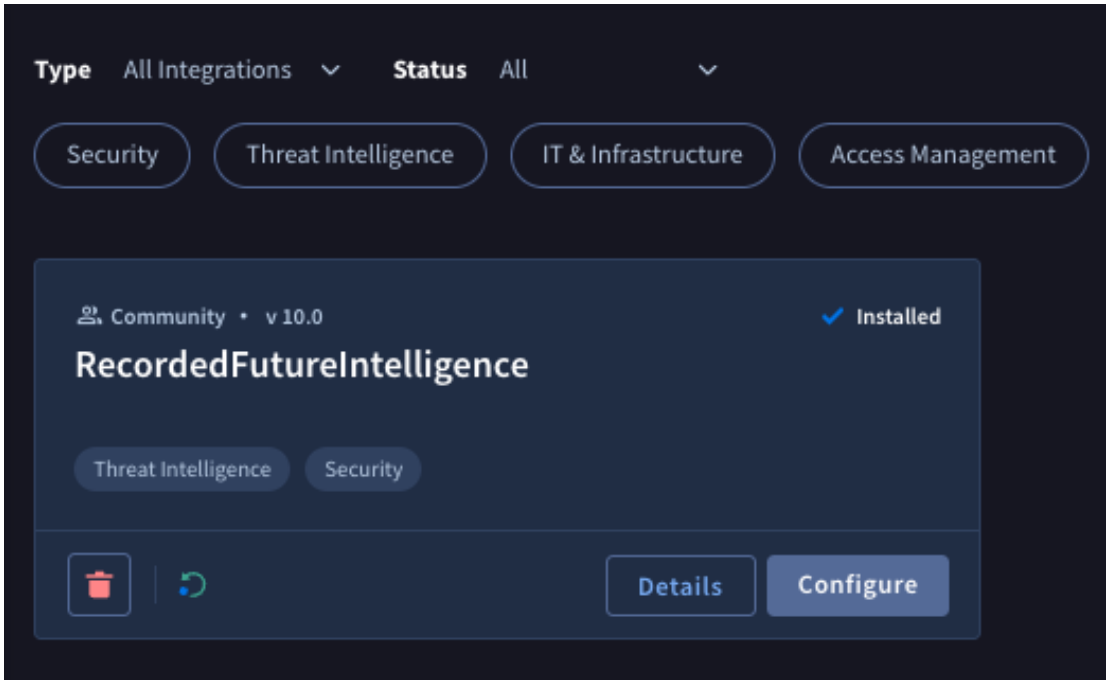

Figure 1. Content Hub — search for RecordedFutureIntelligence and select the Community edition

Once installed, open the integration configuration screen and provide the following:

Parameter	Required	Description
API URL	Yes	Root URL for the Recorded Future API. Default: `https://api.recordedfuture.com`.
API Key	Yes	Your Recorded Future API key.
Sandbox API URL	No	Root URL for the Recorded Future Sandbox API. Default: `https://sandbox.recordedfuture.com`. Required only if using the `Detonate File` or `Detonate URL` actions.

Parameter	Required	Description
Sandbox API Key	No	Your Recorded Future Sandbox API key. Required only if using `Detonate File` or `Detonate URL` actions.
Collective Insights	Yes	INtegration toggle for Collective Insights contribution. When enabled at the integration level, individual enrichment actions can also contribute detections back to Recorded Future. Default: `true`
Verify SSL	No	Verify the SSL certificate when connecting to the Recorded Future API. Default: enabled. Disable only on private/on-premise setups when instructed.

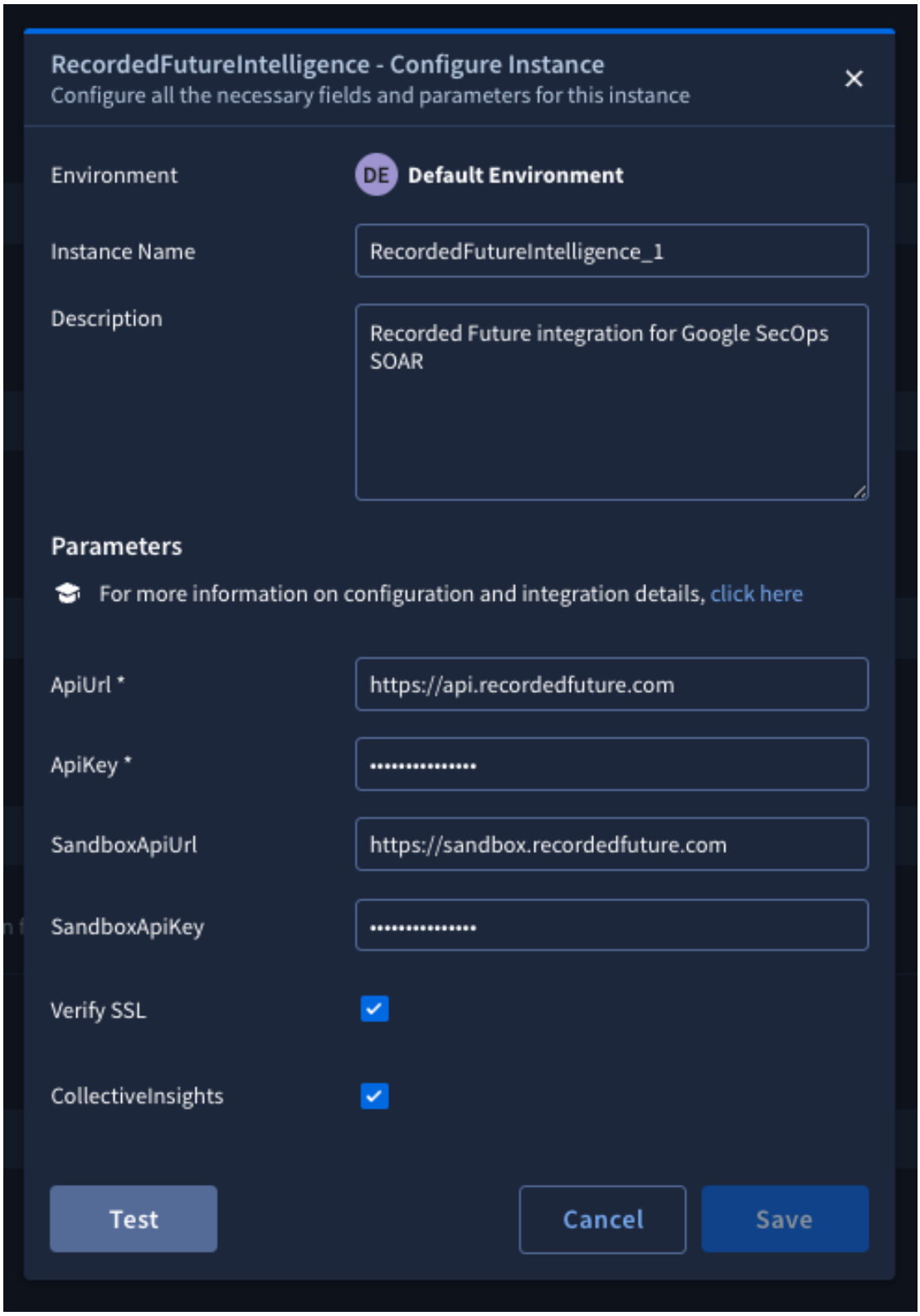

Figure 2. Integration configuration — enter API credentials and optional Sandbox API settings

Click **Test** to verify connectivity. A successful response confirms that the API key is valid and the SOAR server can reach the Recorded Future API.

The **Ping** action (`Test Connectivity`) makes a lightweight API call and returns success or failure. Use it to validate credentials after initial setup or after an API key rotation.

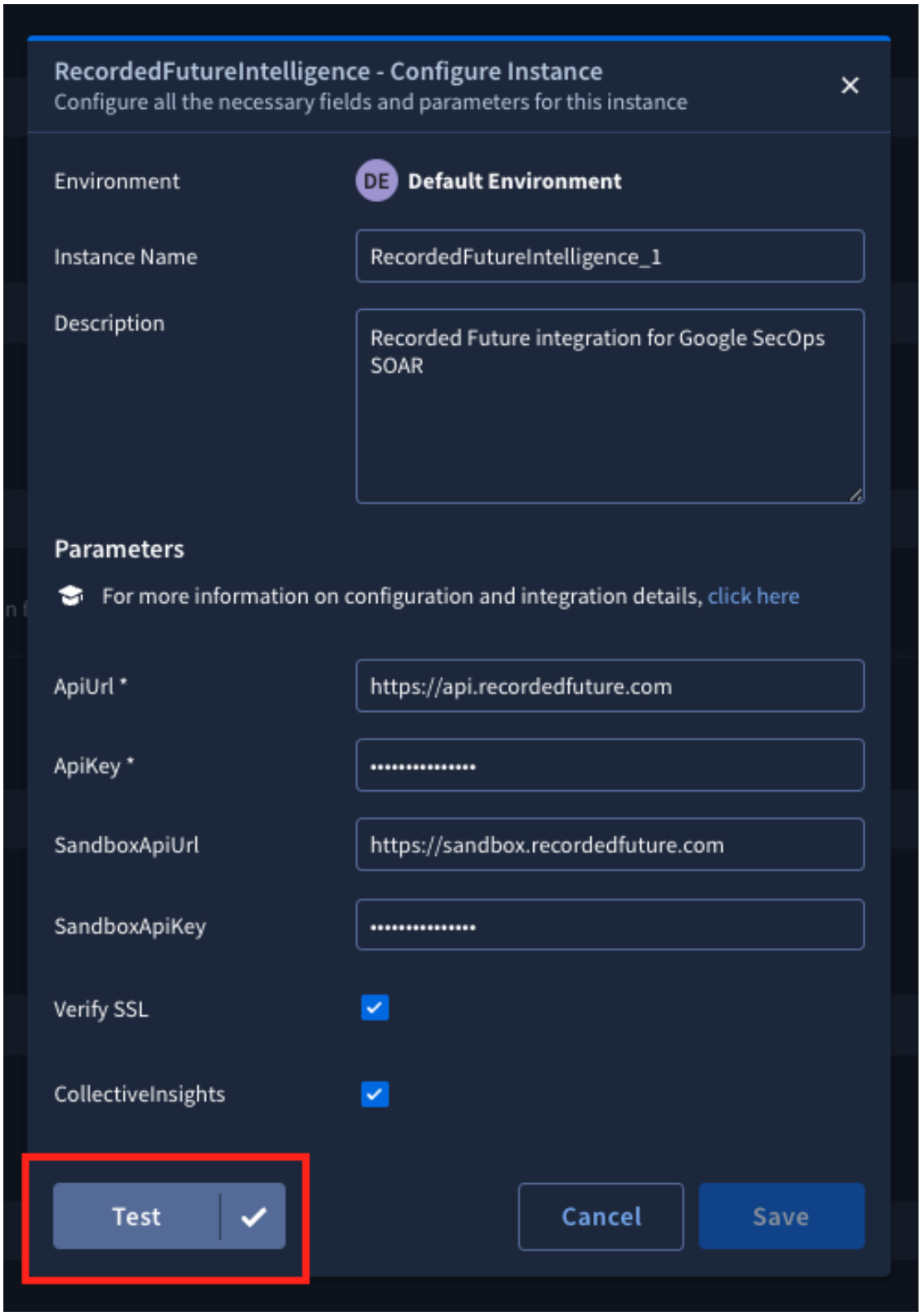

Figure 3. Ping action — successful connectivity test result

To contribute detections to Recorded Future's Collective Insights program, both the integration-level **Collective Insights** setting AND the action-level **Enable Collective Insights** parameter must be enabled. Additionally, the SOAR case source must not originate from Recorded Future (i.e., alerts ingested from Recorded Future connectors are excluded from Collective Insights contribution).

2.2. Proxy Configuration

If the SOAR server reaches the internet through an outbound proxy, configure the proxy at the connector level:

Parameter	Required	Description
Proxy Server Address	No	Full address of the proxy server (e.g. `http://proxy.example.com:8080`).
Proxy Username	No	Username for proxy authentication. Leave blank if the proxy does not require authentication.
Proxy Password	No	Password for proxy authentication. Stored as a secret.

2.3. Connector Setup

Connectors run on a polling schedule and ingest Recorded Future alerts as SOAR cases. Configure each connector independently. All three connectors share a common set of base parameters in addition to their specific settings documented in the Connectors chapter.

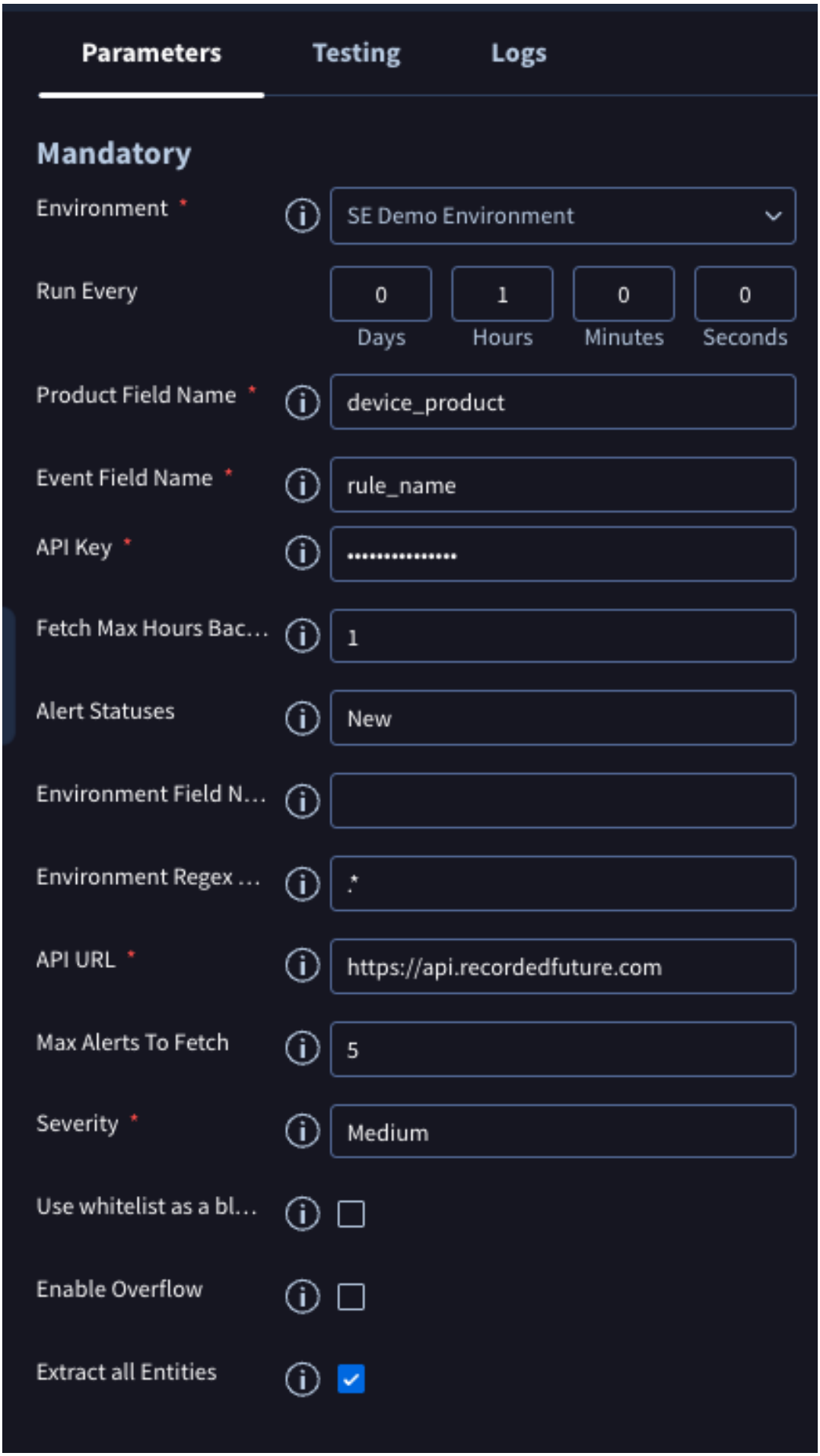

Figure 4. Connector configuration screen — common parameters shared across all connectors

Common connector parameters:

Parameter	Required	Description
API URL	Yes	Root URL for the Recorded Future API. Default: `https://api.recordedfuture.com`.
API Key	Yes	Your Recorded Future API key.
Fetch Max Hours Backwards	No	How far back (in hours) to look for alerts on the first run or after a gap. Default: `1`.
Max Alerts To Fetch	No	Maximum number of alerts to ingest per connector polling cycle. Default: `100`.
Enable Overflow	No	Use the Google SecOps overflow mechanism to deduplicate alerts that arrive in rapid succession. Default: disabled.
Verify SSL	No	Verify the SSL certificate for the connection to the Recorded Future API. Default: disabled.
Proxy Server Address	No	Address of an outbound proxy server, if required.
Proxy Username / Password	No	Credentials for proxy authentication.

2.4. Alert View Widget

We highly recommend adding a classic alert widget to the default alert view. This enables analysts triaging Recorded Future alerts to get a quick overview and leverage AI insights.

Navigate to **Settings ¬ SOAR Settings ¬ Case Data ¬ Views ¬ Default Alert View**. In the **General** tab, drag an HTML widget into the Default Alert View. We recommend placing this widget at the top of the view.

Manually copy and paste the following fields:

Field	Value
Widget Title	`Recorded Future Alert Overview`
Widget Description	`This widget displays details of the Recorded Future Alert such as the alert name, AI Insights, and link to the Recorded Future Portal.`
Widget Width	`100%`

Field	Value
Widget Height	`325 px`

Paste the HTML from `recorded_future_classic_alert.html` into the box under **HTML Code**. The full HTML is provided in the Appendix of this document.

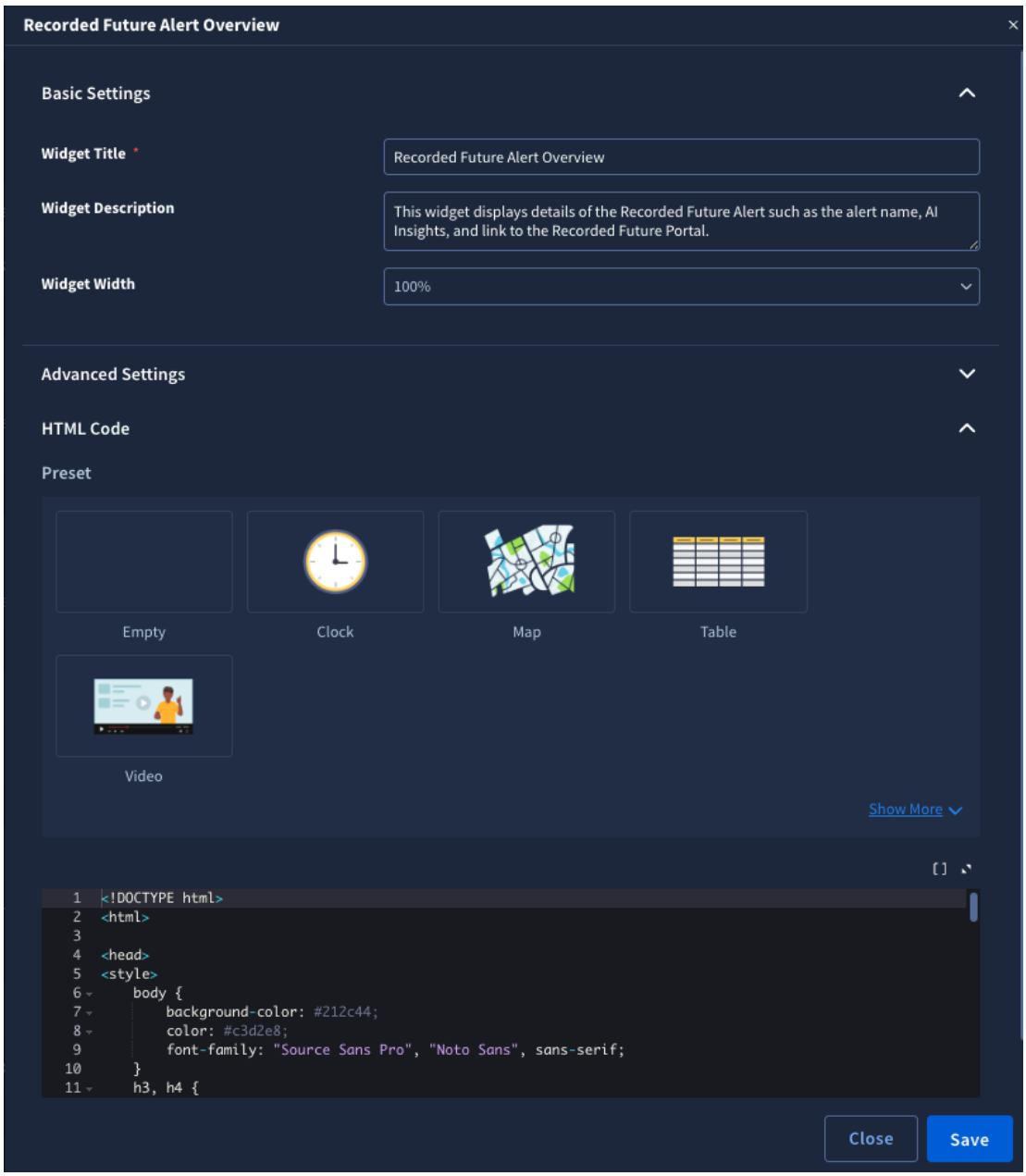

Figure 5. Alert view widget configuration — General tab with title, description, dimensions, and HTML code

2.4.1. Visibility Condition

To avoid cluttering the alert view for non-Recorded Future alerts, add a condition so the widget only renders when the case originates from Recorded Future. Under **Advanced** settings, enable **Conditions** and add the following conditions:

Field	Operator	Value
`[Event.alert_url]`	contains	`recordedfuture.com`
`[Event.Product]`	equals	`Recorded Future Classic Alert`

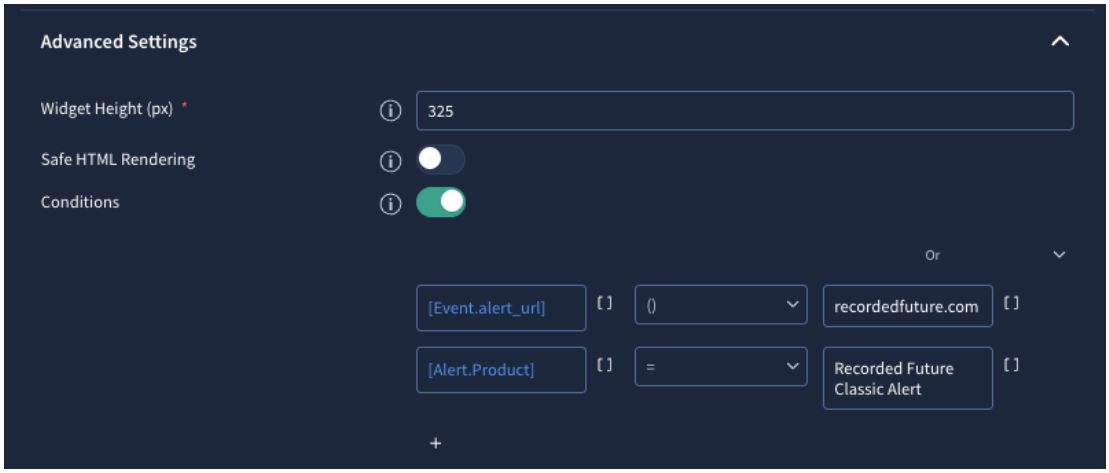

Figure 6. Alert view widget Advanced tab — condition restricting visibility to Recorded Future alerts

2.5. Allowlist and Denylist

The Classic Alerts Connector supports allowlist and denylist filtering based on **Recorded Future rule names**. Rules not matching the allowlist (or matching the denylist) are dropped before SOAR cases are created.

To use the allowlist as a denylist instead, enable **Use whitelist as a blacklist** on the connector configuration.

2.6. Environment Routing

Both alert connectors support environment-based case routing via two parameters:

Parameter	Required	Description
Environment Field Name	No	Name of the field in the alert payload that contains the environment identifier. If not set, all cases go to the default environment.
Environment Regex Pattern	No	Regex applied to the value in **Environment Field Name** to extract or transform the environment string. Default: `.*` (pass-through).

Chapter 3. Actions

This chapter describes each action available in the Recorded Future for Google SecOps SOAR integration. Actions are grouped by functional area.

Parameter types used in tables below:

- **String** — Free-text input
- **Boolean** — True/false toggle
- **Enum** — Selection from a fixed list of values
- **Password** — Secret string field (masked in the UI)

Actions marked **(Async)** are asynchronous — they submit a request and poll for results. The SOAR platform will continue polling on a schedule until the action completes or times out.

3.1. Connectivity

3.1.1. Ping

Test connectivity to the Recorded Future API. Returns success if the API key is valid and the endpoint is reachable.

Entity types: None (integration-level test)

Script result: `is_successful`

3.2. Enrichment

Enrichment actions query Recorded Future's Intelligence API for threat intelligence about SOAR case entities. Each action returns a risk score, criticality label, evidence details, and (optionally) related entity links. The action marks the SOAR entity as **suspicious** or **malicious** when its risk score meets or exceeds the configured threshold.

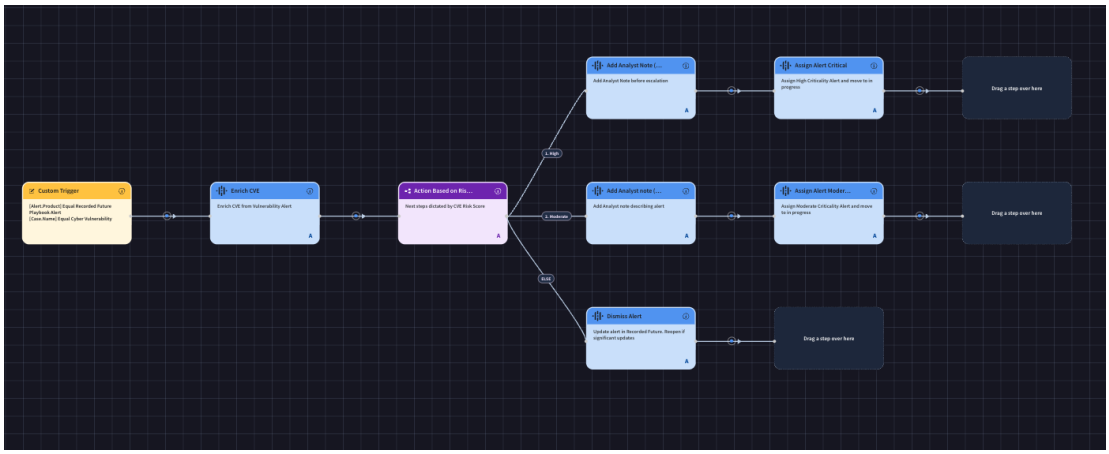

Figure 7. Example enrichment playbook — trigger, enrich entities, mark suspicious, and notify

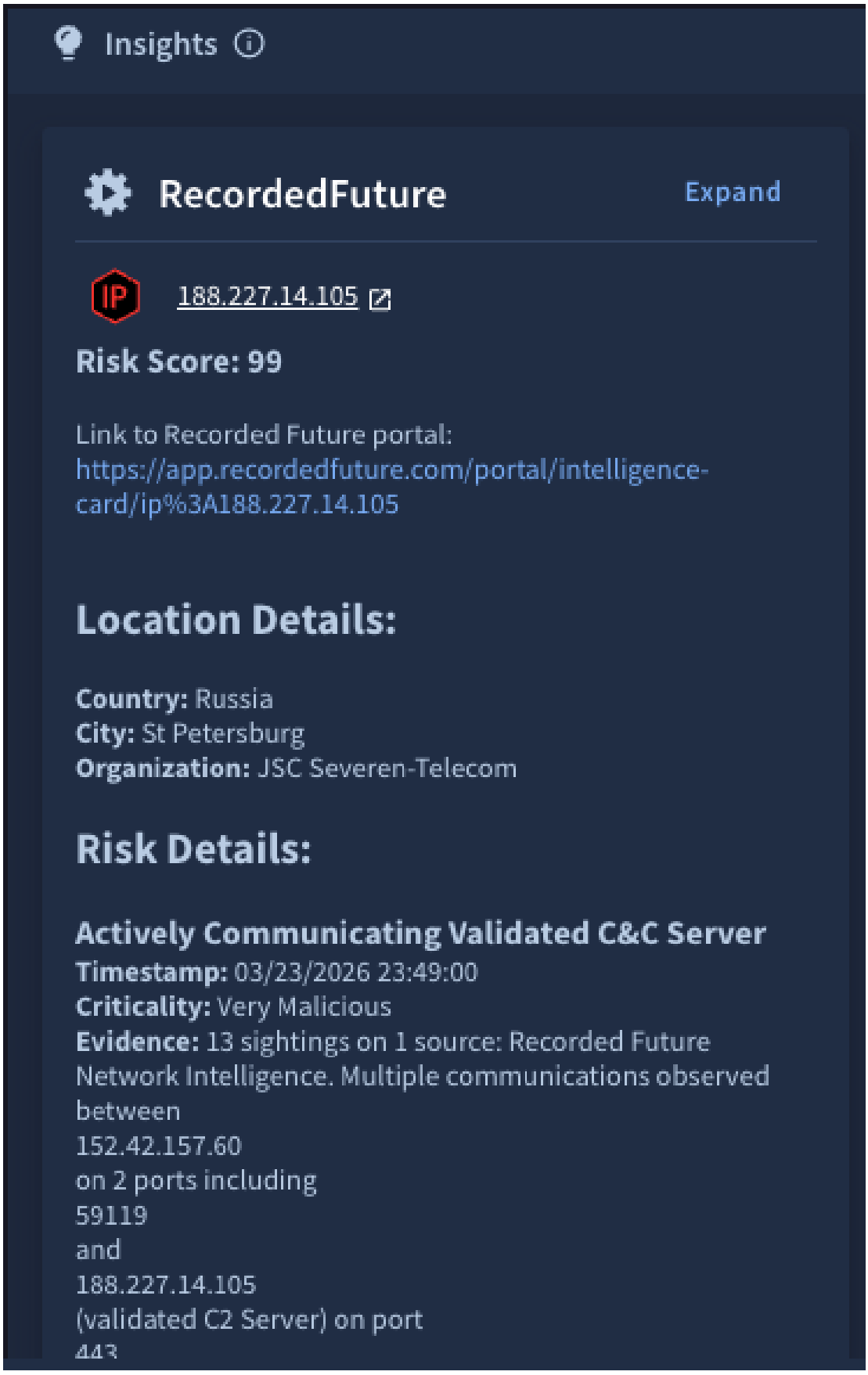

Figure 8. Entity enrichment result — risk score, criticality label, and evidence details shown in the SOAR case insight widget

3.2.1. Common Enrichment Action Parameters

All enrichment actions share the following parameters:

Parameter	Type	Required	Description
Risk Score Threshold	String	Yes	Minimum risk score for an entity to be marked malicious. Range: 0–99. Default: `25`
Include Links	Boolean	No	When enabled, the action retrieves related entity links (associated malware, threat actors, MITRE ATT&CK techniques). Default: `false`
Enable Collective Insights	Boolean	No	When enabled, detected IOCs are contributed back to Recorded Future's Collective Insights program. Default: `true`

Collective Insights contributions require that both the integration-level **Collective Insights** setting AND the action-level **Enable Collective Insights** parameter are enabled. Additionally, the SOAR case source must not originate from Recorded Future.

3.2.2. Enrich IP

Query Recorded Future for intelligence about IP address entities.

Entity types: `Address`

Output fields include: entity, intelCard URL, location (organization, CIDR, country, ASN), risk score, criticality label, evidence details, first/last seen timestamps, and (if enabled) actor/tool/TTP links.

Script result: `is_risky`

3.2.3. Enrich Host

Query Recorded Future for intelligence about hostname/domain entities.

Entity types: `Hostname`

Output fields include: entity, intelCard URL, risk score, criticality label, evidence details, and (if enabled) related MITRE ATT&CK techniques and linked entity references.

Script result: `is_risky`

3.2.4. Enrich Hash

Query Recorded Future for intelligence about file hash entities (MD5, SHA-1, SHA-256).

Entity types: `FileHash`

Output fields include: entity, intelCard URL, risk score, criticality label, evidence details (malware verdicts, sandbox reports, Insikt Group reports), hash algorithm, first/last seen timestamps, and (if enabled) MITRE ATT&CK links.

Script result: `is_risky`

3.2.5. Enrich URL

Query Recorded Future for intelligence about URL entities.

Entity types: `DestinationURL`

Output fields include: entity, intelCard URL, risk score, criticality label, evidence details, and (if enabled) linked entity references.

Script result: `is_risky`

3.2.6. Enrich CVE

Query Recorded Future for intelligence about CVE vulnerability entities.

Entity types: `CVE`

Output fields include: CVE description, CVSS score, NVD severity, evidence details (exploit availability, Insikt references, malware links), and first/last seen timestamps.

Script result: `is_risky`

3.2.7. Enrich IOC

Query Recorded Future for intelligence across multiple entity types in a single action. Supports IPs, hostnames, file hashes, URLs, and CVEs simultaneously.

Entity types: `Hostname`, `CVE`, `FileHash`, `Address`, `DestinationURL`

Script result: `is_success`

3.2.8. Enrich IOCs Bulk

Bulk enrichment action for large numbers of IOCs. Operates similarly to Enrich IOC but is optimized for high-volume use cases.

Entity types: `Hostname`, `CVE`, `FileHash`, `Address`, `DestinationURL`

Enrich IOCs Bulk does not create entity insights on the SOAR case for enriched entities.

Script result: `is_success`

3.3. Analyst Notes

3.3.1. Add Analyst Note

Add a note to Recorded Future entities from within a SOAR playbook. The action adds the note to all in-scope entities that have previously been enriched and have a corresponding Recorded Future entity.

Entity types: `FileHash`, `DestinationURL`, `CVE`, `Hostname`, `Address`

Parameter	Type	Required	Description
Note Title	String	Yes	Title for the analyst note. Default: `Note Title`
Note Text	String	Yes	Body text of the analyst note. Default: `Note Text`
Topic	Enum	No	Categorizes the note by topic. Options: `None`, `Actor Profile`, `Analyst On-Demand Report`, `Cyber Threat Analysis`, `Flash Report`, `Indicator`, `Informational`, `Malware/Tool Profile`, `Source Profile`, `Threat Lead`, `Validated Intelligence Event`, `Weekly Threat Landscape`, `YARA Rule`. Default: `None`

Script result: `is_success`

3.4. Classic Alerts

3.4.1. Get Alert Details

Fetch full details for a Recorded Future Classic Alert by its alert ID. Returns entities, risk scores, references, and review status.

Entity types: None (alert ID is passed as a parameter)

Parameter	Type	Required	Description
Alert ID	String	Yes	Recorded Future alert ID to fetch (e.g. `feRxxx`).

Output fields include: alert title, triggered timestamp, alert rule name and URL, entity list with risk scores and evidence, review status (assignee, note, status), related document count.

Script result: `is_success`

3.4.2. Update Alert

Update the status, assignee, or note on a Recorded Future Classic Alert.

Entity types: None

Parameter	Type	Required	Description
Alert ID	String	Yes	ID of the alert to update.
Assign To	String	No	User to assign the alert to. Accepts user ID, username, user hash, or email address.
Note	String	No	Note text to add to the alert.
Status	Enum	Yes	New status for the alert. Options: `None`, `New`, `Pending`, `Dismissed`, `Resolved`, `Flag for Tuning`. Default: `None`

Script result: `is_success`

3.5. Playbook Alerts

3.5.1. Get Playbook Alert Details

Fetch detailed information for a specific Recorded Future Playbook Alert. Returns status, evidence summary, and additional data depending on the alert category.

Entity types: None

Parameter	Type	Required	Description
Playbook Alert ID	String	Yes	Recorded Future Playbook Alert ID (e.g. `task:ba62d37e-⋯`).

Parameter	Type	Required	Description
Category	String	Yes	Category of the playbook alert. Values: `domain_abuse`, `cyber_vulnerability`, `code_repo_leakage`, `third_party_risk`, `identity_novel_exposures`, `geopolitics_facility`, `malware_report`.

Output fields vary by category but typically include: status (status, priority, created, updated, entity, risk score), evidence summary (exposed secrets, malware family, infrastructure, technologies), DNS records, WHOIS data, and action log.

Script result: `is_success`

3.5.2. Update Playbook Alert

Update the status, priority, assignee, or reopen strategy for a Recorded Future Playbook Alert.

Entity types: None

Parameter	Type	Required	Description
Playbook Alert ID	String	Yes	ID of the playbook alert to update.
Playbook Alert Category	String	Yes	Category of the playbook alert being updated.
Assign To	String	No	User hash of the user to assign the alert to.
Log Entry	String	No	Comment to attach to the update action in the alert log.
Status	Enum	No	New status. Options: `None`, `New`, `In Progress`, `Dismissed`, `Resolved`. Default: `None`
Priority	Enum	No	New priority. Options: `None`, `High`, `Moderate`, `Informational`. Default: `None`
Reopen Strategy	Enum	No	How the alert should behave if it is later reopened. Options: `None`, `Never`, `Significant Updates`. Default: `None`

Script result: `is_success`

3.5.3. Refresh Playbook Alert

Pull the latest data for a Playbook Alert from Recorded Future and update the SOAR case created by the Playbook Alerts Connector.

This action is intended to be run after the SOAR case is initially ingested from the Playbook Alerts Connector, or to pull the latest alert data from Recorded Future and update the case. Use it in playbooks to keep case details current as Recorded Future updates the alert over time.

Entity types: None

Parameter	Type	Required	Description
Playbook Alert ID	String	Yes	ID of the playbook alert to refresh.
Category	String	Yes	Category of the playbook alert. Same values as Get Playbook Alert Details.

Script result: `is_success`

3.6. Sandbox

3.6.1. Detonate File *(Async)*

Submit a file to the Recorded Future Sandbox for dynamic behavioral analysis. The action is asynchronous — it submits the file and polls for the completed report.

Entity types: `FileHash` (the hash is used to identify the file; the file itself is retrieved from the specified source)

Parameter	Type	Required	Description
File Path	String	Yes	Path to the file to submit. Interpreted relative to the selected File Source.
File Source	Enum	Yes	Where to retrieve the file from. Options: `GCP Bucket`, `Local File System`. `GCP Bucket` pulls the file from a Google Cloud Storage bucket. `Local File System` reads from the SOAR server's local filesystem.

Parameter	Type	Required	Description
Profile	String	No	Sandbox analysis profile to use (e.g. specific OS environment). Leave blank to use the default profile.
Password	String	No	Password to unlock an archive sample before analysis (e.g. for password-protected zip files).

Output fields include: sample ID and score, task results (behavioral, static, URL scan), behavioral signatures with MITRE ATT&CK TTPs, extracted network IOCs (IPs, domains, URLs), extracted malware configs (C2 servers, family), and PE metadata.

Figure 9. Sandbox detonation result — score, behavioral signatures, and extracted IOCs on the SOAR case

Script result: `is_success`

3.6.2. Detonate URL *(Async)*

Submit a URL to the Recorded Future Sandbox for dynamic analysis. The action is asynchronous.

Entity types: `DestinationURL`

Parameter	Type	Required	Description
Profile	String	No	Sandbox analysis profile to use. Leave blank for the default profile.

Output fields include: sample ID and score, behavioral task results, signatures with TTPs, and extracted network IOCs.

Script result: `is_success`

3.6.3. Search Hash Malware Intelligence

Query the Recorded Future Sandbox for existing analysis reports associated with file hash entities. Unlike Detonate File, this action queries previously submitted samples and does not trigger a new detonation.

Entity types: `FileHash`

Parameter	Type	Required	Description
My Enterprise	Boolean	No	When enabled, restrict results to samples submitted by your organization only. Default: `false`
Start Date	String	No	Earliest submission date to include. Accepts absolute dates (`2026-01-23`) or relative offsets (`-30d`). Default: `-30d`
End Date	String	No	Latest submission date to include. Accepts absolute dates or relative offsets. Leave blank for today.

Output fields include: sample ID, score, tags, static analysis (hashes, PE metadata, signatures), dynamic analysis (network activity, extracted configs, behavioral signatures), and task results per analysis type.

Script result: `is_success`

3.7. Detection Rules

3.7.1. Search Detection Rules

Search Recorded Future for detection rules (YARA, Sigma, and Snort) produced by the Insikt Group. All filter parameters are optional — omitting them returns all accessible rules up to the result limit.

Entity types: None

Parameter	Type	Required	Description
Detection Rule Type	String	No	Comma-separated list of rule types to filter on. Supported values: `yara`, `sigma`, `snort`.
Filter on Target Entities	Boolean	No	When enabled, the action maps SOAR case entities to Recorded Future entity IDs and adds them to the entity filter automatically.
Entity ID	String	No	Comma-separated list of Recorded Future entity IDs. Rules tagged with any of these entities are returned.
Created Before	String	No	Return rules created before this date or relative date (e.g. `2026-01-01` or `-7d`).
Created After	String	No	Return rules created after this date or relative date.
Updated Before	String	No	Return rules last updated before this date or relative date.
Updated After	String	No	Return rules last updated after this date or relative date.
Detection Rule ID	String	No	Filter by a specific Recorded Future document ID (e.g. `doc:aaaaa`).
Detection Rule Title	String	No	Filter by rule title (substring match).
Tagged Entities	Boolean	No	When enabled, only return rules that have at least one tagged entity.
Max Results	String	No	Maximum number of rules to return.

Output fields include: rule ID, type, title, description, created/updated timestamps, and rule content (file name and raw rule text) along with associated tagged entities.

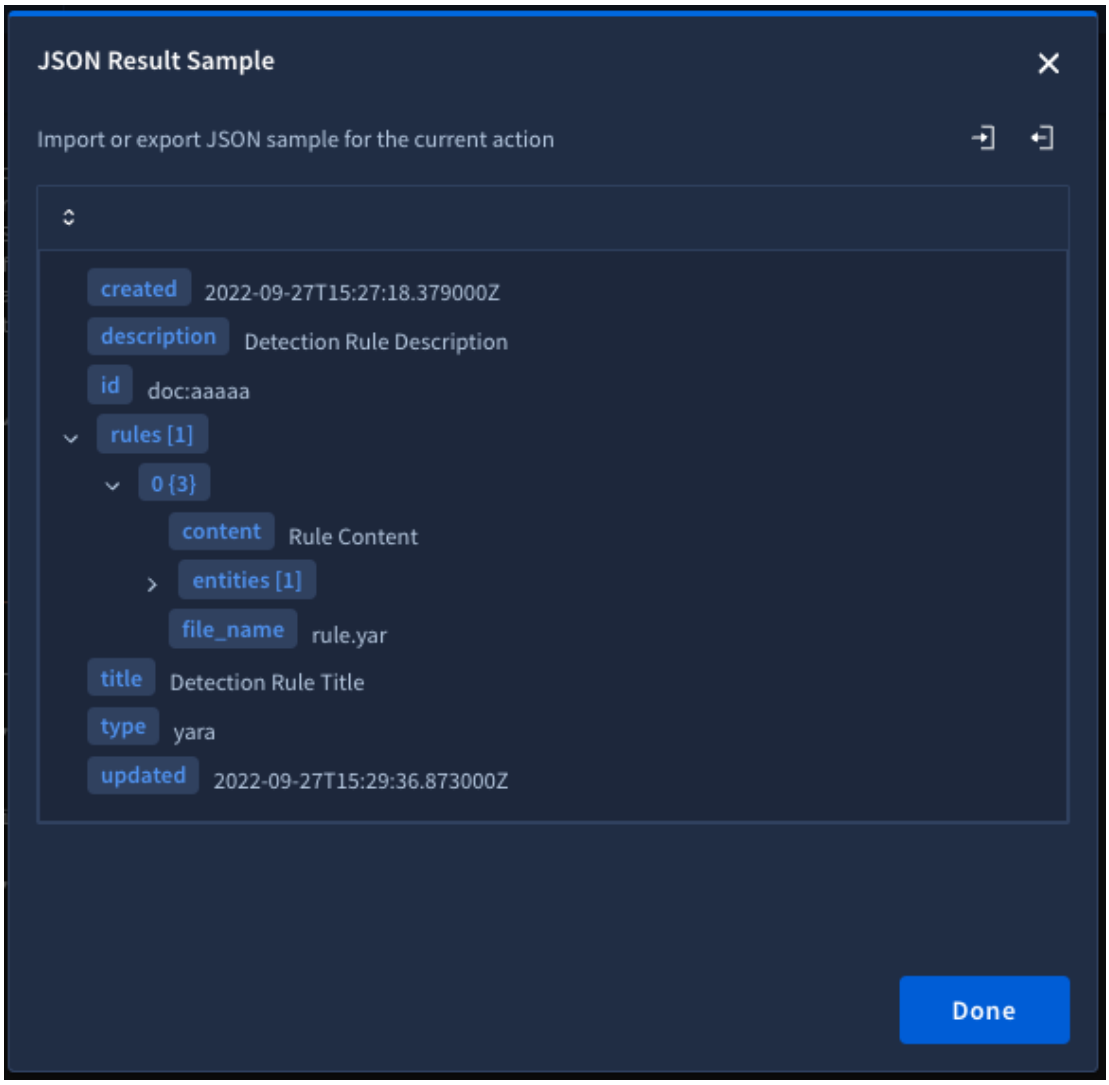

Figure 10. Search Detection Rules result — YARA/Sigma/Snort rules returned from the Recorded Future platform

Script result: `is_success`

3.7.2. Fetch Detection Rule

Retrieve a single Recorded Future detection rule by its document ID.

Entity types: None

Parameter	Type	Required	Description
Rule ID	String	Yes	Recorded Future document ID of the detection rule (e.g. `doc:aaaaa`).

Output fields include: rule ID, type, title, description, created/updated timestamps, rule content (file name and raw text), and tagged entity associations.

Script result: `is_success`

3.8. Entity Utilities

3.8.1. Entity Lookup

Retrieve Recorded Future entity details by Recorded Future entity ID. Useful when a playbook has resolved an entity ID (e.g. from an enrichment result or a detection rule) and needs to fetch its full profile.

Entity types: None

Parameter	Type	Required	Description
Entity ID	String	Yes	Recorded Future entity ID to look up (e.g. `ANtblN`).

Output fields include: ID, type, name, common names, aliases, and threat actor flag.

Script result: `is_success`

3.8.2. Entity Match

Resolve a free-text entity name to one or more Recorded Future entity IDs. Returns matched and unmatched results.

Entity types: None

Parameter	Type	Required	Description
Entity Name	String	Yes	Free-text name to search for (e.g. a malware family name, threat actor, or organization).
Entity Type	String	No	Comma-separated list of Recorded Future entity types to restrict the match to (e.g. `Organization`, `Malware`).
Limit	String	No	Maximum number of results to return. Maximum 100. Default: `10`

Output: Matched results contain the Recorded Future entity ID, name, and type. Unmatched results contain the input name and a not-found message.

Script result: `is_success`

3.9. Lists

Recorded Future Lists allow organizations to maintain curated collections of entities, IPs, domains, hashes, and other items in the Recorded Future platform. The following actions provide full list lifecycle management from within SOAR playbooks.

3.9.1. Create List

Create a new Recorded Future list.

Entity types: None

Parameter	Type	Required	Description
List Name	String	Yes	Display name for the new list.
List Type	Enum	No	Type of list to create. Options: `entity`, `source`, `text`, `custom`, `ip`, `domain`, `tech_stack`, `industry`, `brand`, `partner`, `industry_peer`, `location`, `supplier`, `vulnerability`, `company`, `hash`, `operation`, `attacker`, `target`, `method`, `executive`. Default: `entity`

Output: New list metadata including ID, name, type, created/updated timestamps, and owner details.

Script result: `is_success`

3.9.2. Add List Entities

Add SOAR case entities (or a named entity) to a Recorded Future list.

The action resolves which entity to add using the following priority order: **Entity ID** (if provided) ¬ **Entity Name and Entity Type** (if both provided) ¬ SOAR case target entities. Only one of these paths is used per invocation.

Entity types: All (operates on SOAR case entities by default)

Parameter	Type	Required	Description
List ID	String	Yes	Recorded Future list ID to add entities to (e.g. `report:mfLASl`).
Entity ID	String	No	Recorded Future entity ID to add. If provided, takes priority over Entity Name/Type and SOAR case entities.

Parameter	Type	Required	Description
Entity Name	String	No	Name of a specific entity to add. Must be provided together with Entity Type to override SOAR case entities.
Entity Type	String	No	Type of the explicitly named entity to add. Must be provided together with Entity Name to override SOAR case entities.

Script result: `is_success`

3.9.3. Fetch List

Retrieve list metadata (name, type, owner, timestamps) for a Recorded Future list by its ID.

Entity types: None

Parameter	Type	Required	Description
List ID	String	Yes	Recorded Future list ID to retrieve.

Script result: `is_success`

3.9.4. Get List Entities

Retrieve all entities currently contained in a Recorded Future list.

Entity types: None

Parameter	Type	Required	Description
List ID	String	Yes	Recorded Future list ID.

Output: Array of entity objects, each with entity ID, name, type, annotation, status, and date added.

Script result: `is_success`

3.9.5. Get List Status

Retrieve status information about a Recorded Future list (metadata only — does not return the list's entities).

Entity types: None

Parameter	Type	Required	Description
List ID	String	Yes	Recorded Future list ID.

Script result: `is_success`

3.9.6. Remove List Entities

Remove entities from a Recorded Future list.

The action resolves which entity to remove using the following priority order: **Entity ID** (if provided) ¬ **Entity Name and Entity Type** (if both provided) ¬ SOAR case target entities. Only one of these paths is used per invocation.

Entity types: All (operates on SOAR case entities by default)

Parameter	Type	Required	Description
List ID	String	Yes	Recorded Future list ID to remove entities from.
Entity ID	String	No	Recorded Future entity ID to remove. If provided, takes priority over Entity Name/Type and SOAR case entities.
Entity Name	String	No	Name of a specific entity to remove. Must be provided together with Entity Type to override SOAR case entities.
Entity Type	String	No	Type of the explicitly named entity to remove. Must be provided together with Entity Name to override SOAR case entities.

Script result: `is_success`

Chapter 4. Connectors

Connectors run on a polling schedule within Google SecOps SOAR and ingest Recorded Future alerts as SOAR cases. Each connector maintains its own state (last seen alert timestamps and IDs) to avoid duplicate ingestion across polling cycles.

Three connectors are available:

- **Classic Alerts Connector** — Ingests Recorded Future Classic Alerts
- **Playbook Alerts Connector** — Ingests new Recorded Future Playbook Alerts
- **Playbook Alerts Tracking Connector** — Ingests lifecycle change events for existing Playbook Alerts

All connectors share the common base parameters documented in the Configuration chapter (API URL, API Key, Fetch Max Hours Backwards, Max Alerts To Fetch, Enable Overflow, Verify SSL, and proxy settings).

4.1. Recorded Future - Classic Alerts Connector

Polls the Recorded Future API for Classic Alerts and creates SOAR cases. Classic Alerts are triggered by Recorded Future alerting rules and cover a broad range of threat scenarios (infrastructure threats, brand risk, vulnerability exploits, etc.).

Allowlists and denylists are matched against **Recorded Future rule names**. Alerts whose rule names are not in the allowlist (or are in the denylist when using the blacklist mode) are dropped before case creation.

4.1.1. Parameters

Parameter	Type	Required	Description
DeviceProductField	String	Yes	Source field used to populate the SOAR alert's Device Product field. Default: `device_product`
EventClassId	String	Yes	Source field used to populate the SOAR alert's Event Class ID. Default: `rule_name`
Environment Field Name	String	No	Field in the alert payload containing the environment name for case routing. If not found, cases go to the default environment.

Parameter	Type	Required	Description
Environment Regex Pattern	String	No	Regex applied to the Environment Field Name value to extract or transform the environment string. Default: `.*`
PythonProcessTimeout	Integer	Yes	Timeout (in seconds) for the connector's Python process. Default: `180`
API URL	String	Yes	Recorded Future API root URL. Default: `https://api.recordedfuture.com`
API Key	Password	Yes	Recorded Future API key.
Fetch Max Hours Backwards	Integer	No	How many hours back to search for alerts on the initial run or after a gap. Default: `1`
Alert Statuses	String	No	Comma-separated list of alert statuses to ingest. Options: `New`, `Pending`, `Resolved`, `Dismissed`. Default: `New`
Max Alerts To Fetch	Integer	No	Maximum alerts to process per polling cycle. Default: `100`
Severity	String	Yes	Severity assigned to SOAR cases created by this connector. Values: `Low`, `Medium`, `High`, `Critical`. Default: `Medium`
Use whitelist as a blacklist	Boolean	No	When enabled, the connector's allowlist is used as a denylist instead. Default: `false`
Enable Overflow	Boolean	No	Use Google SecOps overflow to deduplicate similar alerts arriving in rapid succession. Default: `false`
Extract all Entities	Boolean	No	When enabled, extract all entities from each alert event. When disabled, only the primary entity is extracted. Default: `false`
Verify SSL	Boolean	No	Verify the SSL certificate when connecting to the Recorded Future API. Default: `false`
Proxy Server Address	String	No	Outbound proxy address (e.g. `http://proxy.example.com:8080`).
Proxy Username	String	No	Proxy authentication username.

Parameter	Type	Required	Description
Proxy Password	Password	No	Proxy authentication password.

4.1.2. Use Case

Use the Classic Alerts Connector to automatically open SOAR cases when Recorded Future alerting rules fire. Pair it with playbooks that call `Get Alert Details` to enrich cases and `Update Alert` to close them in Recorded Future once resolved in SOAR.

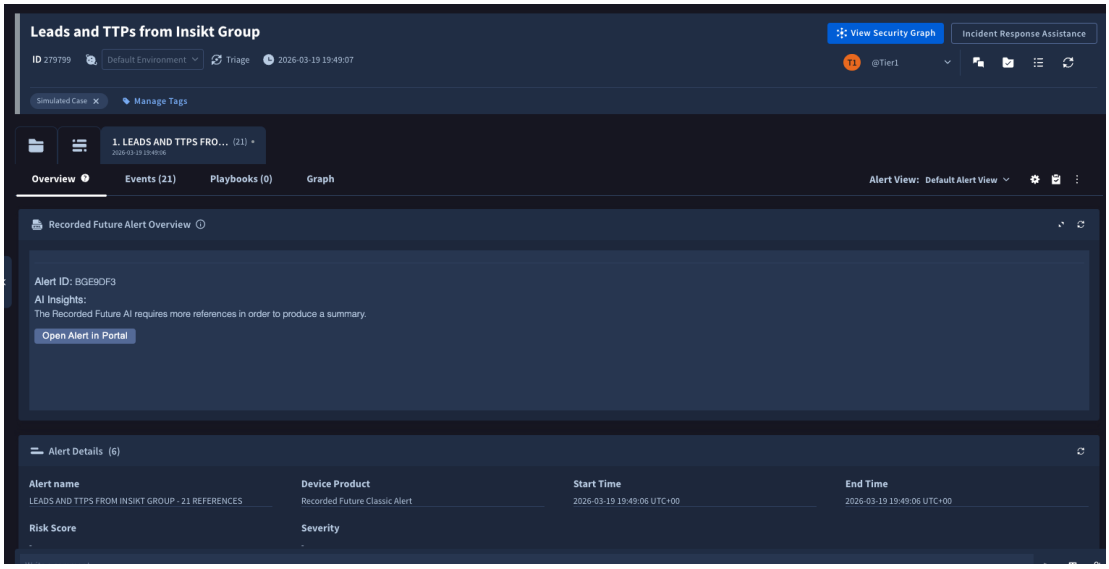

Figure 11. SOAR case created by the Classic Alerts Connector — case title, entities, and alert details

---

4.2. Recorded Future - Playbook Alerts Connector

Polls the Recorded Future API for new Playbook Alerts and creates SOAR cases. Each alert includes structured evidence panels (DNS records, WHOIS data, identity exposures, malware context, etc.) that are surfaced on the SOAR case.

The connector supports filtering by category, status, and priority so teams can focus on the alert types most relevant to their workflows.

4.2.1. Supported Categories

- `domain_abuse`
- `cyber_vulnerability`
- `code_repo_leakage`
- `third_party_risk`

- `identity_novel_exposures`
- `geopolitics_facility`
- `malware_report`

4.2.2. Parameters

Parameter	Type	Required	Description
DeviceProductField	String	Yes	Source field for the SOAR alert's Device Product. Default: `device_product`
EventClassId	String	Yes	Source field for the SOAR alert's Event Class ID. Default: `category`
Environment Field Name	String	No	Field for environment-based case routing.
Environment Regex Pattern	String	No	Regex for environment string extraction. Default: `.*`
PythonProcessTimeout	Integer	Yes	Connector process timeout in seconds. Default: `180`
API URL	String	Yes	Recorded Future API root URL. Default: `https://api.recordedfuture.com`
API Key	Password	Yes	Recorded Future API key.
Fetch Max Hours Backwards	Integer	No	Hours to look back for alerts on the initial run. Default: `1`
Playbook Alert Categories	String	No	Comma-separated list of categories to ingest. Default: all seven categories (`domain_abuse`, `cyber_vulnerability`, `code_repo_leakage`, `third_party_risk`, `identity_novel_exposures`, `geopolitics_facility`, `malware_report`)
Playbook Alert Statuses	String	No	Comma-separated list of statuses to filter on. Options: `New`, `InProgress`, `Resolved`, `Dismissed`. Default: all statuses.
Playbook Alert Priorities	String	No	Comma-separated list of priorities to filter on. Options: `Informational`, `Moderate`, `High`. Default: all priorities.

Parameter	Type	Required	Description
Max Alerts To Fetch	Integer	No	Maximum alerts to process per polling cycle. Default: `100`
Severity	String	No	*(Advanced)* Override severity for SOAR cases. When set, overrides the severity derived from the Playbook Alert priority. Values: `Low`, `Medium`, `High`, `Critical`.
Enable Overflow	Boolean	No	Deduplicate similar alerts via Google SecOps overflow. Default: `false`
Verify SSL	Boolean	No	Verify the Recorded Future API SSL certificate. Default: `false`
Proxy Server Address	String	No	Outbound proxy address.
Proxy Username	String	No	Proxy authentication username.
Proxy Password	Password	No	Proxy authentication password.

4.2.3. Use Case

Use this connector with playbooks that call `Refresh Playbook Alert` to surface and render the alert context on the SOAR case, `Update Playbook Alert` to set status and assignee, and `Get Playbook Alert Details` to pull the latest alert data from Recorded Future as it is updated.

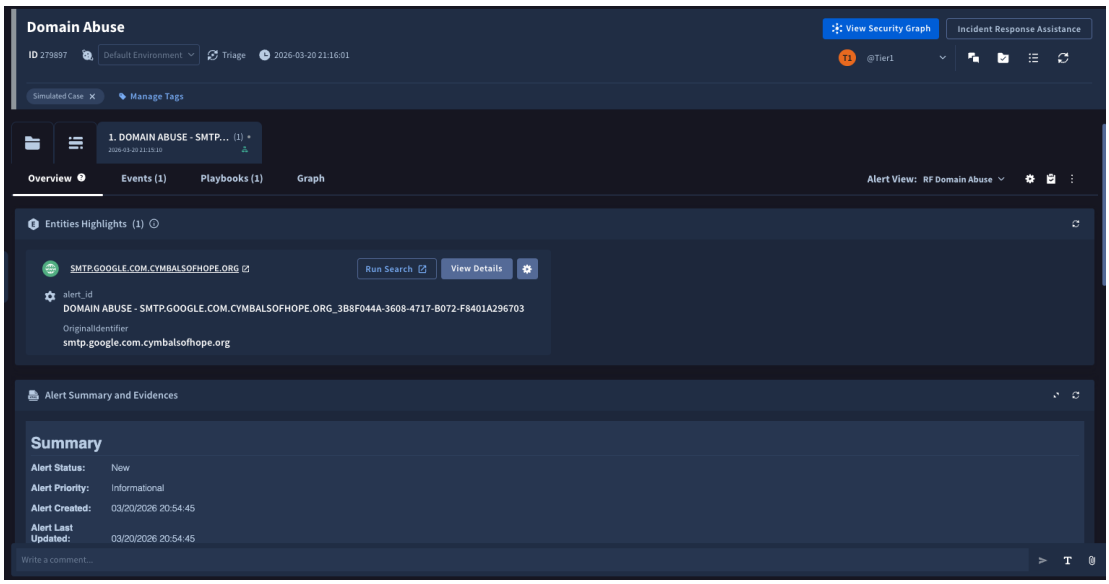

Figure 12. SOAR case created by the Playbook Alerts Connector — structured evidence panels including status, DNS, WHOIS, and entity data

4.3. Recorded Future - Playbook Alerts Tracking Connector

Polls the Recorded Future API for **lifecycle changes** to existing Playbook Alerts and creates new SOAR cases when specific events occur. This connector complements the Playbook Alerts Connector by surfacing significant updates — priority escalations, new assessments, newly added entities, and reopened alerts.

The Playbook Alerts Tracking Connector creates new SOAR cases each time a tracked event is detected. It does not update existing cases.

At least one of the four trigger flags (**Playbook Alert Reopened**, **Priority Increased**, **New Assessment Added**, **Entity Added**) must be enabled or no cases will be created.

4.3.1. Parameters

Parameter	Type	Required	Description
DeviceProductField	String	Yes	Source field for the SOAR alert's Device Product. Default: `device_product`
EventClassId	String	Yes	Source field for the SOAR alert's Event Class ID. Default: `category`
Environment Field Name	String	No	Field for environment-based case routing.
Environment Regex Pattern	String	No	Regex for environment string extraction. Default: `.*`
PythonProcessTimeout	Integer	Yes	Connector process timeout in seconds. Default: `180`
API URL	String	Yes	Recorded Future API root URL. Default: `https://api.recordedfuture.com`
API Key	Password	Yes	Recorded Future API key.
Search Max Hours Backwards	Integer	No	How many hours back to search for updated Playbook Alerts. Default: `1`

Parameter	Type	Required	Description
Playbook Alert Categories	String	No	Comma-separated list of categories to monitor. Options: `domain_abuse`, `cyber_vulnerability`, `code_repo_leakage`, `third_party_risk`, `identity_novel_exposures`, `geopolitics_facility`. Default: all six listed categories.
Playbook Alert Statuses	String	No	Filter by alert status. Options: `New`, `InProgress`, `Resolved`, `Dismissed`.
Playbook Alert Priorities	String	No	Filter by priority. Options: `Informational`, `Moderate`, `High`.
Playbook Alert Reopened	Boolean	No	Create a new SOAR case when a Playbook Alert is reopened. Default: `false`
Priority Increased	Boolean	No	Create a new SOAR case when a Playbook Alert's priority increases. Default: `false`
New Assessment Added	Boolean	No	Create a new SOAR case when a new assessment is added to a Playbook Alert. Default: `false`
Entity Added	Boolean	No	Create a new SOAR case when new entities are added to a Playbook Alert. Default: `false`
Max Alerts To Fetch	Integer	No	Maximum updated alerts to process per polling cycle. Default: `100`
Severity	String	Yes	Severity for SOAR cases created by this connector. Values: `Low`, `Medium`, `High`, `Critical`. Default: `Medium`
Enable Overflow	Boolean	No	Deduplicate similar alerts via Google SecOps overflow. Default: `false`
Verify SSL	Boolean	No	Verify the Recorded Future API SSL certificate. Default: `false`
Proxy Server Address	String	No	Outbound proxy address.
Proxy Username	String	No	Proxy authentication username.

Parameter	Type	Required	Description
Proxy Password	Password	No	Proxy authentication password.

4.3.2. Use Case

Deploy this connector alongside the Playbook Alerts Connector to maintain ongoing visibility into how Recorded Future is evolving its Playbook Alerts. For example, enable **Priority Increased** and **New Assessment Added** to automatically trigger playbook re-investigation when a previously low-priority domain abuse alert escalates to High priority or new evidence is added.

Chapter 5. Changelog

All notable changes to the Recorded Future for Google SecOps SOAR integration are documented in this chapter.

5.1. [12.0] (2026-04-14)

5.1.1. New Features

- **Detection Rule Actions** — Added `Search Detection Rules` and `Fetch Detection Rule` actions to retrieve Recorded Future Insikt Group detection rules (YARA, Sigma, and Snort) directly from SOAR playbooks.
- **Malware Report Playbook Alert Support** — Added `malware_report` as a supported category in the Playbook Alerts Connector.
- **List Management Actions** — Added full list lifecycle support: `Create List`, `Add List Entities`, `Fetch List`, `Get List Entities`, `Get List Status`, and `Remove List Entities`.
- **Entity Utility Actions** — Added `Entity Lookup` to retrieve Recorded Future entity details by Recorded Future ID, and `Entity Match` to resolve free-text names to Recorded Future entity IDs.

5.1.2. Improvements

- General stability improvements and error handling improvements.

5.1.3. Bug Fixes

- Fixed issue creating Classic Alerts with no references.

5.2. [11.0] (2026-03-18)

5.2.1. General

- Updated integration metadata.

5.3. [10.0] (2026-02-05)

5.3.1. Improvements

- Added `Search Hash Malware Intelligence` action to fetch Sandbox analysis reports by SHA-256 hash.
- Added `Enrich IOCs Bulk` action for high-volume entity enrichment directly from SOAR

cases.

5.3.2. Bug Fixes

- Fixed issue rendering enriched entity links data.
- Fixed Tracking Connector parameter name.
- Fixed Tracking Connector Add Entity logic issue.
- Fixed invalid default value for Update Alert actions.
- Fixed issue with Refresh Playbook Alert action for certain Domain Abuse alerts.

5.3.3. General

- Refactored enrichment implementation to reduce API calls.
- Updated JSON action result examples for playbook previews.

5.4. [9.0] (2025-11-04)

5.4.1. Bug Fixes

- Fixed issue adding IP entity to Identity Playbook Alert.

5.4.2. General

- Updated module documentation.

5.5. [8.0] (2025-10-30)

5.5.1. Bug Fixes

- Fixed issue submitting Analyst Note with no topic.
- Fixed issue submitting Collective Insights.

5.5.2. General

- Updated module documentation.

5.6. [7.0] (2025-07-31)

5.6.1. Bug Fixes

- Fixed legacy Classic Alert status parameter in connector and actions.
- Fixed missing Playbook Alert category parameter for Update Playbook Alert action.

- Fixed Recorded Future API SDK logging issue preventing actions from exiting successfully.
- Fixed Classic Alert event fields preventing HTML widget from rendering.

5.7. [6.0] (2025-04-19)

5.7.1. New Features

- **Sandbox Support** — Added `Detonate File` and `Detonate URL` actions for behavioral analysis via Recorded Future Sandbox.

5.7.2. General

- Upgraded Recorded Future API SDK to v2.

5.8. [5.0] (2025-03-25)

5.8.1. General

- Google SecOps Content Hub integration dependency maintenance.

5.9. [4.0] (2025-03-19)

5.9.1. Bug Fixes

- Fixed invalid parameter in Update Playbook Alert action.
- Fixed incorrect alert timestamp key in Playbook Alert Connectors.

5.10. [3.0] (2024-12-26)

5.10.1. New Features

- **Playbook Alerts Support** — Added `Get Playbook Alert Details`, `Refresh Playbook Alert`, and `Update Playbook Alert` actions.
- **New Connectors** — Added Playbook Alerts Connector and Playbook Alerts Tracking Connector.
- Added use case views for Playbook Alert workflows.

5.11. [2.0] (2024-12-09)

5.11.1. General

- Upgraded integration Python version.

5.12. [1.0] (2024-09-30)

5.12.1. New Features

- **Indicator Enrichment** — Comprehensive enrichment support for IP, host, hash, URL, and CVE entity types.
  - Removed separate related entity commands; consolidated into unified enrichment actions.
  - Replaced Related Entities with inline linked entity references.
  - Improved display for entity insights and risk scoring.
  - Improved JSON and CSV reports attached to enriched entities in playbooks.
  - Added new enrichment output fields.
  - Added option to send enriched IOCs to Collective Insights.
- **Classic Alerts** — Initial Classic Alerts connector and action support.
  - Split alert references into multiple events — one event per alert reference.
  - Added support for the *Why The Alert* feature in both the Classic Alerts Connector and `Get Alert Details` action.
  - Added option to enable or disable alert overflow in Google SecOps SOAR.
  - Added AI Insights support in alerts.
  - Automatically extract indicators from alerts and attach them to the SOAR case.
  - Option to extract only the primary entity or all entities per alert.
  - Renamed connector from *Recorded Future Security Alerts* to *Recorded Future Classic Alerts*.
  - Changed alert Device Product from `Recorded Future` to `Recorded Future Classic Alerts`.
  - Removed the *Get Alert Details* option from the connector — all ingested alerts now fetch full details automatically.

5.12.2. Known Limitations

Sandbox

The entity insight widgets for the `Detonate URL` and `Detonate File` actions strip query strings from URLs for improved readability. Full URLs including query strings remain available in the full Recorded Future Sandbox report attached to the case.

Appendix A: Appendix

A.1. recorded_future_classic_alert.html

HTML source for the Classic Alert View widget. Copy the entire contents of this block and paste it into the **HTML Code** field of the HTML widget as described in the Alert View Widget section.

```
<!DOCTYPE html>
<html>

<head>
    <style>
        body {
            background-color: #212c44;
            color: #c3d2e8;
            font-family: "Source Sans Pro", "Noto Sans", sans-serif;
        }

        h3,
        h4 {
            font-weight: 500 !important;
        }

        h3 {
            border-bottom-width: 2px;
            border-bottom-style: solid;
            border-bottom-color: #3a4a6c;
        }

        h4 {
            margin-top: 10px;
            margin-bottom: 5px;
        }

        p {
            font-size: 14px;
            font-weight: 100;
            margin-top: 0;
        }

        button {
            background-color: #6275a3;
            color: #fff;
            border-radius: 4px;
            font-weight: 400;
            font-size: 14px;
```

```
            padding: 0 12px;
            line-height: 24px;
            letter-spacing: 0.5px;
            border: none;
            text-align: center;
            user-select: none;
            cursor: pointer;
        }

        a {
            width: 160px;
            display: flex;
            gap: 2px;
            text-decoration: none;
        }
    </style>
</head>

<body>
    <h3>[Event.alert_title]</h3>
    <div>
        <h4 style="display: inline;">Alert ID: </h4>
        <p style="display: inline;">[Event.alert_id]</p>
    </div>
    <div>
        <h4>AI Insights:</h4>
        <p>[Event.ai_insights_text]</p>
    </div>
    <a href=[Event.alert_url] target="_blank">
        <button>Open Alert in Portal</button>
    </a>
</body>
<script>
    //This script enables the widget theme to reflect the user's choice in the
platform.
    //Removing this script will result in your HTML widget permanently
displayed in the light theme.
    onmessage = evt => {
        for (const [key, value] of Object.entries(evt.data)) {
            document.body.style[key] = value;
        }
    }
</script>

</html>
```