Swimlane Solutions and Applications

1. Solutions and Applications

Turbine solutions are end-to-end use cases that contain components, playbooks, and assets that are mostly preconfigured to reduce a practitioner's time and level of effort when creating common security solutions.

Current Turbine Solutions

Solutions are available in Turbine under **Library** → **Swimlane Content**. Current solutions and extensions include:

Solution	Description
AI SOC Solution	AI-powered SOC solution for alert ingestion, signal triage, threat intelligence enrichment, and case management. Includes Hero AI–driven investigation plans, signal routing rules, dashboards, reports, and the AI Ingestion application for building connectors and ingestion pipelines.
SOC Solutions Bundle	Security operations center solution for SOC workflows, triage, and response.
Swimlane AI Agents Case Management Extension	Part of the SOC Solutions Bundle. Provides Hero AI agents and a unified UI to speed up SOC analyst workflows for triage, investigation, and incident response. Integrates with the Case and Incident Management (CIM) application; includes on-demand and automated Hero AI analysis, verdict and threat intelligence analysis, and MITRE ATT&CK & D3FEND mapping. Install from **Library** → **Swimlane Content** after the SOC Solutions Bundle is installed.
Business Continuity Management (BCM) Solution	End-to-end use case for business continuity and resilience planning, testing, and response.
Complian	Solution for managing compliance controls, evidence, and audit readiness

1.1.4. AI SOC Ingestion

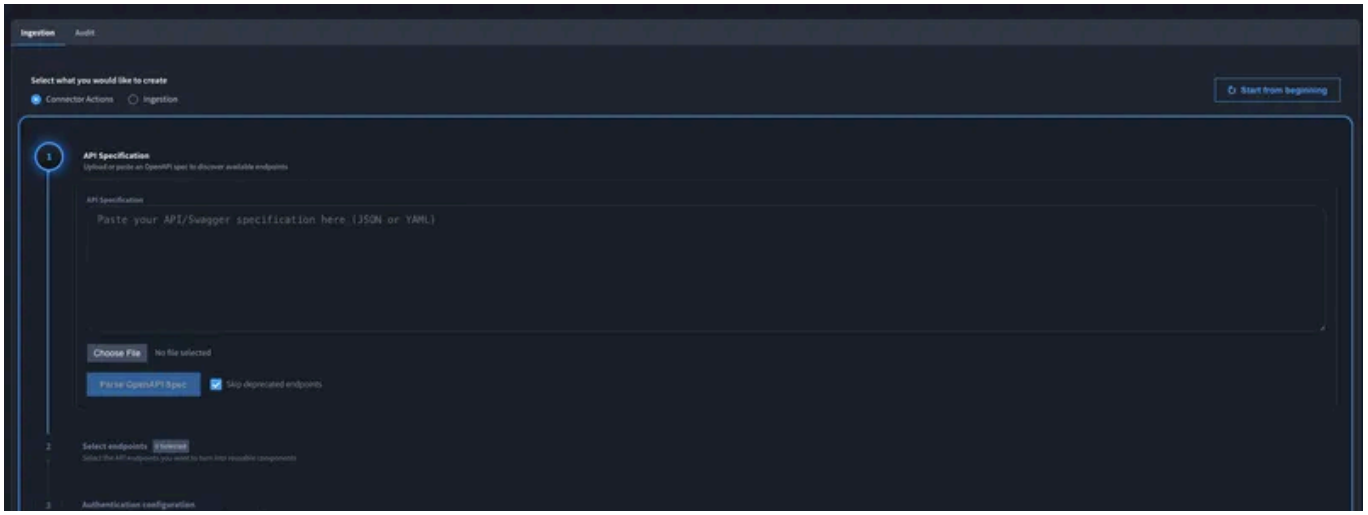
The **AI Ingestion** application helps you build vendor connectors and alert ingestion pipelines quickly — without writing code or learning JSONata. A guided custom widget builds Turbine components from an uploaded OpenAPI specification, one per endpoint. Hero AI then assists in mapping incoming raw alert data — from any source — to a standardized Turbine Schema object, producing an ingestion component ready to use in downstream AI SOC playbooks.

Why Use the AI Ingestion

Use the AI Ingestion when you want to **integrate new alert sources faster**, **prototype ingestion paths** against real vendor APIs, and **hand standardized Turbine Schema output** to your existing AI SOC playbooks without hand-building every connector and mapping from scratch.

What it helps you do:

- **Quick integration:** Upload an OpenAPI specification and generate Turbine components per endpoint so you can connect to vendor APIs in a repeatable way.
- **Prototyping and iteration:** Test API calls, inspect sample payloads, and adjust mappings before you commit to production routing and automation.
- **Consistent normalization:** Hero AI suggests how raw vendor fields map to Turbine Schema fields (the Turbine Schema is the mapping target you work against) so alerts land in a predictable shape for downstream Flows.
- **Reuse across playbooks:** Components you generate can be referenced from multiple playbooks and templates, so you invest once and reuse.



1.1.6.4. Signal Routing Rules (RULE)

Signal Routing Rules control how signals are routed to playbooks. Each rule is a **record** (for example, RULE-29) that you can open, edit, and enable.

Routing Rule Management dashboard

The primary place to manage routing rules is the **Routing Rule Management** dashboard. The dashboard shows all rules in a table ordered by **Rule Order** (ascending). Rule order is **unique** and evaluated like firewall rules: the **first matching rule** runs, so order matters.

From the dashboard you can:

- **Reorder rules:** Use the **drag handles** (stack of dots) in the **Order** column to drag and drop rules into the desired order. Reordering changes which rule runs first when multiple rules could match.
- **Open the rule record:** Click the **ID** column icon (open-in-new view) or the **Edit** (pencil) icon to open the rule record and change conditions, playbook, or description.
- **Open the associated playbook:** Click the **Associated Playbook** link to open the playbook that runs when the rule matches.
- **Enable or disable a rule:** Use the **Enabled** toggle for each row. Rules are inactive when disabled; only enabled rules evaluate and trigger playbooks.
- **Edit a rule:** Click the **Edit** (pencil) icon to open the rule record in a slider or view so you can modify conditions, **Rule Application**, **Selected Playbook**, or other fields.
- **Delete a rule:** Use the **Delete** (trash) icon to remove a rule. Use with care; deletion cannot be undone from the UI.

Use **Add New Rule** on the dashboard to create a rule, or **Apply** to save order changes after reordering.

1) Open **Signal Routing Rules**

- Navigate to **Application Records** -> **Signal Routing Rules** or open the **Routing Rule Management** dashboard to view and manage all rules.
- Open an existing rule via the dashboard **Edit** icon or by opening it from the application list.

2) Core fields

1.1.8.1. Dashboards

Dashboards provide real-time visibility into SOC operations through interactive widgets. Use dashboards to monitor current state, identify trends, and quickly access records requiring attention.

The **AI SOC** workspace in current packages (for example, AI SOC beta v1.2.4) includes these dashboards: **Routing Rule Management**, **Analyst Triage Queue**, **Security Operations Overview**, **MITRE ATT&CK Techniques**, and **Threat Intelligence Overview**. If your workspace lists different dashboard titles, open **Dashboards** under the **AI SOC** workspace and use the view that matches your need (queue, overview, MITRE, or threat intelligence).

Analyst Triage Queue

The **Analyst Triage Queue** dashboard supports daily operations on **Case Management** workload: active queue, ownership, blocked work, and priority or severity views.

Location: Navigate to **Dashboards** → **Analyst Triage Queue**

Cards included (package default):

- **Active Triage Queue**
- **Requires Attention by Current Owner**
- **Cases Created Over Time**

1.2.3. SOC - Alert Ingestion (Cron) - Template Playbook Overview

This document provides detailed information about the **SOC - Alert Ingestion (Cron) - Template** playbook, which leverages scheduled tasks to pull alerts from external systems, process them, and handle actions such as enrichment, correlation, and case creation. Users need to configure only the following components:

- **Placeholder - Create TEDS Alert List**
- **Custom Alert Data Extension (Cron)**
- **Correlate (Cron)**

Note: These playbooks can and should be **duplicated** and the **Placeholder - Create TEDS Alert List** component swapped out to match the technology stack in your organization. This ensures the playbook works seamlessly with your chosen vendor's tools.

Overview

- **Objective:** Automate scheduled alert ingestion using connectors, assets, and Turbine logic.
- **Key Workflow Steps:**
    1. Alerts are ingested on a cron schedule, fetching data from external systems.
    2. Fetched alerts are standardized into TEDS objects.
    3. Each alert undergoes deduplication, enrichment, and correlation.
    4. Enriched and correlated alerts are used to create cases for investigation.

Accessing the Playbook

To access the playbook:

1. Navigate to **Orchestration** in the Swimlane platform.
2. Click on **Playbooks**.
3. Select **SOC - Alert Ingestion (Cron) - Template**.

Loop-Based Alert Processing in the Playbook

1.3.5. References

Application Record Schemas

Resources

Field	Type	Required	Description
Resource Name	String	Yes	Name of the resource.
Resource Type	Enum	Yes	Application, Building, Supplier, etc.
Resource Criticality	Enum	No	Low, Medium, High.
Resource Status	Enum	No	Active, Inactive, Planned, Deprecated.
Resource Owner	User/Group	No	Assigned owner.
Resource ID	String	No	Unique identifier.
Resource Location	String	No	Physical or logical location.
RPO Required?	Enum	No	Yes/No.

Business Impact Assessments (BIAs)

Field	Type	Required	Description
Name	String	Yes	Name of business entity.
Type	Enum	Yes	Business Process, IT System, Supplier, etc.
Owners	Multi User/Group	Yes	Assigned owners.
BCM Manager	User/Group	No	Continuity manager.
Resources	Relation	No	Linked resources.
Pre-Screening Impact	Enum	No	Initial scoring across impact categories.

1.4.8. SCF Remediation Plans Application Overview

The **SCF Remediation Plans application** documents and tracks corrective actions for audit findings. Each remediation plan is linked to a finding and contains details about ownership, deadlines, status, and supporting evidence.

Plans can be ingested in bulk using the SCF Data Import application or created manually within the application.

SCF Remediation Plan Record Fields

- **Remediation Plan Description** – Overview of the corrective action.
- **Finding Reference** – Linked SCF Finding record.
- **Owner** – Assigned individual responsible for execution.
- **Deadline & Closing Date** – Planned completion and actual closure dates.
- **Status** – Not Started, In Progress, Completed.
- **Finding ID / Remediation Plan ID** – Identifiers for tracking.
- **Risk Level** – Indicates severity and impact.
- **Notes** – Additional context.
- **Evidence Explanation** – Rich text field to capture details of the remediation strategy.

Evidence Files – File upload area for supporting documents.

1.4.9. SCF Reporting Application Overview

The **SCF Reporting** application is the third core component of the SCF Common Controls and Evidence Management solution. It enables users to generate, manage, and track compliance reporting across frameworks in both **ad-hoc** and **scheduled** formats. Reports can be used to view compliance posture, export evidence packages, and monitor control and evidence readiness across multiple compliance frameworks.

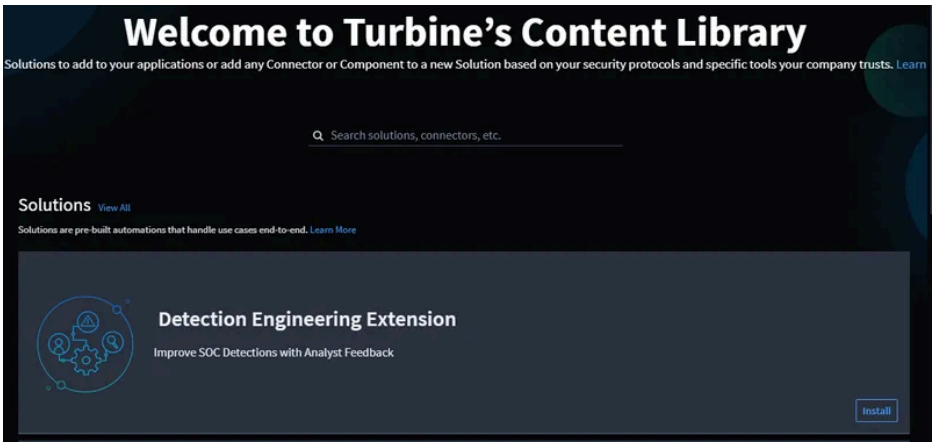
Once the SCF Library and SCF Evidence applications are configured and populated, the SCF Reporting application can be used to create real-time or periodic snapshots of compliance

1.7.1. Installing Detection Engineering Extension

Installing from Library

To install the Detection Engineering Extension from the Library:

1. Log in to Turbine.
2. From your desired tenant, click **Library**.
3. Click **Swimlane Content**.
4. Select **Detection Engineering Extension** from the list of Solutions.



5. Click **Install** on the extension.
6. The Detection Engineer details window is displayed:
    - The **Overview** tab shows each individual extension that composes the Detection Engineering and their capabilities.
    - The **Content** tab shows how many and which workspaces, applications, dashboards, and/or reports are included in the Detection Engineering.
    - The **Documentation** tab has a link to the Detection Engineering topic in the user guide.
7. On the right-hand side of the window, click **+Install** again.
8. From the **Install Detection Engineering** window, select the content you would like to install.

1.8.2.5. Filtering and Deduplication

Metrics and Logic

During ingestion, VRM compares each batch of findings against the most recently processed export to detect and exclude previously seen data. This ensures only new or modified findings are enriched.

Metrics tracked include:

- **number_unseen_findings:** Number of findings not present in the previous export
- **deduplication_count:** Number of duplicates excluded based on export comparison
- **excluded_findings_count:** Entries excluded through custom filtering logic
- **message:** Description of export status (for example, *New Export* or *Existing Export used*)

VRM - Filtering Activity Record

The **VRM - Filtering Activity** record logs ingestion filtering outcomes for each batch. It provides detailed insights into deduplication and exclusion logic and supports tuning of ingestion performance.
key fields include:

- **Script_Internal_Runtime**: Time taken by the Python component to compare findings
- **Memory_Used_Bytes**: Memory consumed during the filtering process
- **Unique_Excluded_Values_Count**: Number of unique values that triggered exclusion
- **Excluded_Findings_Count**: Total findings skipped due to custom exclusion rules
- **number_unseen_findings**, **deduplication_count**, **message**

This record is created automatically during each ingestion batch and is accessible via the **Filtering Activity** app in Turbine. Reviewing this data helps teams fine-tune ingestion batch sizes and exclusion rules.

Tuning Recommendations

- Begin with a page size of 1000 for typical finding volumes.
- Use **number_unseen_findings** and **deduplication_count** trends to optimize the page

1.8.3.3.1. Vulnerability Case Management - Metrics Tab

The Metrics Tab in the Case Management is used to track key timestamps and durations related to the vulnerability case. This data helps security teams assess how long a vulnerability remains open and how efficiently it is being handled.

- **Time Discovered**
  - Represents the exact timestamp when the vulnerability was first detected.
- **Time Opened**
  - The timestamp when the case was officially opened.
  - Example: Jan 27, 2025, 5:30:00 AM +05:30
  - Helps determine when the remediation process started.
- **Time Closed:** The timestamp when the case was resolved and closed.
- **Discovered Duration**
  - The time elapsed between Time Discovered and Time Opened.
  - Useful for measuring detection efficiency.
  - Since the Time Discovered field is blank, this duration is not calculated.
- **Open Duration**
  - Tracks how long the case has remained open.
  - If Time Closed is not recorded, the case is still active.

1.8.3.3.2. Vulnerability Case Management - Support Tab

The Support Tab in the case management provides essential metadata and administrative options for managing the case. It helps security teams track when the case was created, last updated, ownership, and automation settings.

- **Update Risk Score Button**

1.8.5.1. Vulnerability Finding Data Model

Title	Key	Type
Vulnerability Finding	finding	object
Merged Risk Scores	finding.merged-risk-scores	string
Vulnerability Commercial Exploit Found	finding.vulnerability-commercial-exploit-found	string
Vulnerability CVSS Base Score	finding.vulnerability-cvss-base-score	integer
Vulnerability CVSS Temporal/Threat Score	finding.vulnerability-cvss-temporal-threat-score	integer
Vulnerability CVSS Vector String	finding.vulnerability-cvss-vector-string	string
Vulnerability CVSS Version	finding.vulnerability-cvss-version	string
Vulnerability Description	finding.vulnerability-description	string
Vulnerability EPSS Percentile	finding.vulnerability-epss-percentile	integer
Vulnerability EPSS Score	finding.vulnerability-epss-score	integer
Vulnerability Exploits Trending on Github	finding.vulnerability-exploits-trending-on-github	string
Vulnerability Exploits	finding.vulnerability-exploits	string
Vulnerability Finding Asset Criticality	finding.vulnerability-finding-asset-criticality	integer
Vulnerability Finding Asset Zone Criticality	finding.vulnerability-finding-asset-zone-criticality	integer
Vulnerability Finding Asset Remedation Channel	finding.vulnerability-finding-asset-remedation-channel	string
Vulnerability Finding Asset Remedation Owner	finding.vulnerability-finding-asset-remedation-owner	string
Vulnerability Finding Asset	finding.vulnerability-finding-asset-	array