Swimlane Solutions and Applications

CONTENTS

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.
Compliance Audit Readiness (CAR) Solution	Solution for managing compliance controls, evidence, and audit readiness.
Collaboration Extension	Extends Turbine with collaboration capabilities across teams and workflows.
Crafted AI Prompts	Prebuilt AI prompts and configurations for common security and automation tasks.
Detection Engineering	Supports detection rule development, testing, and lifecycle management.

1.1.4. AI SOC Ingestion

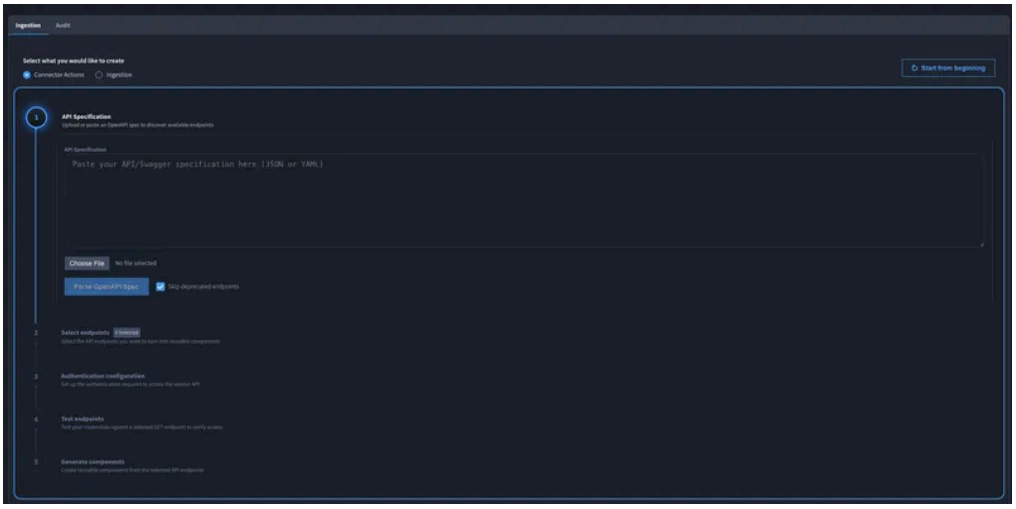
The **AI Ingestion** application helps you build vendor connectors and alert ingestion pipelines quickly — without writing code or learning JSONata. A guided custom widget builds Turbine components from an uploaded OpenAPI specification, one per endpoint. Hero AI then assists in mapping incoming raw 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.



The following section describes the widget modes and how to run each flow.

> **Alert Ingestion only:** AI Ingestion helps quickly build alerts ingestion pipelines and not Email ingestion.

What the AI Ingestion Widget Does

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

- Rule identity: Rule Name, Description, **Rule Order**, rule-uuid.
- Logic: Conditions, Target, **Rule Application**.
- Execution: **Selected Playbook**, **Status**, **Records Matched**.
- **Audit**: Created by, Last updated by, First Created, Last Updated, History.

2a) Tabs

- Rule: Define conditions and rule application, and associate a playbook.
- History: **Review** routing rule changes over time.
- **Support**: **Run** the rule manually against pending signals and view **Records Matched**.

2a) Rule validation

1.1.7.7. Understanding Verdict Generation

The **Generate Verdict** step in the Determination phase uses Hero AI to analyze the signal and produce a verdict.

Verdict Inputs

The verdict generation uses:

- Signal tracking ID
- Signal data (observables, severity, source, etc.)
- Threat intelligence enrichment results
- Knowledge Base Articles linked to the signal
- Similar signals and their verdicts
- Investigation comments and summaries

Collect Outputs Before the Verdict

The verdict must receive the outputs from all completed investigation actions. In AI SOC 26.2.0 playbooks, **AI SOC - Collect Playbook Outputs** runs immediately before the verdict component and collects successful action outputs from the current playbook run.

> **Upgrading a playbook created with AI SOC content earlier than 26.2.0:** Add **AI SOC - Collect Playbook Outputs** immediately before the verdict component and set **Playbook Run Id** to `$run.id`. Without this step, a playbook that calls multiple components inside a loop may pass only the last component output to Hero AI, which can produce an incomplete or incorrect verdict. New playbooks created from AI SOC 26.2.0 content already include this component.

For the full playbook structure, see **Building Routing Rule Playbooks**.

Verdict Outputs

After running the verdict step, you'll see:

Final AI Verdict:

- **Malicious:** Confirmed threat requiring response
- **Suspicious:** Needs further investigation
- **Benign:** False positive, no threat
- **Unknown:** Insufficient data to determine

Verdict Analysis:

- Narrative explanation of the verdict
- Confidence score (0-10 scale)
- Summary of key factors

1.2.1. Installing SOC Solutions Bundle

The SOC Solutions Bundle is a solution bundle that is made of four smaller, interconnected solutions: Phishing Triage, Alert Triage, Threat Intelligence (TI), and Case and Incident Management (CIM). For more information, see the corresponding sections.

> This installation and configuration documentation is for the Canvas version of the SOC Solutions Bundle only.

> The SOC Solutions Bundle is not available in **Swimlane Content** or the Marketplace. To install it, request the SSP from your Swimlane representative, then import the package.

Important Definitions

- Ingestion Component - A component or set of components that:
  - Interact with a vendor endpoint, such as run a SIEM search, look up observable reputation, or initiate a remediation action
  - Normalize inputs and outputs using interfaces to enable swappable component usage
  - Convert various 3rd-party alerts, reputations, emails, and so on to a common schema for use in Turbine solutions such as SOC Solutions
- **Turbine Extendable Data Schema (TEDS)** - Turbine's native common schema for interacting with alerts, emails, reputations, and so on.

Request and Import the Package

To install the SOC Solutions Bundle, request the SSP from your Swimlane representative.

After you have the SSP:

1. Import the package into your tenant. For steps, see **Import Swimlane Solution Packages**.
2. If the import prompts you to overwrite content that already exists in your environment, review the selection. You can deselect items you do not want to overwrite. If you deselect too many items, the solution may not import completely.
3. Enable playbooks and webhooks after import if they remain disabled.
4. Reconfigure credentials, key store values, and secure asset fields as required.

Once you have imported the SOC Solutions Bundle, configure it to ingest data from your tools for your use cases.

1. For SIEM, XDR, or EDR Alert Triage, see **SOC - Alert Ingestion (Cron) - Template Playbook Overview** and **SOC - Alert Ingestion (Webhook) - Template Playbook Overview**.
2. For Phishing Triage, see **SOC - Bulk Ingest Phishing Emails - Template Playbook Overview**.
3. After you configure one or more alert sources, you can add Threat Intelligence to enhance your observables. See **Configure Threat Intelligence Enrichment Integration**.
4. You can then configure CIM for your environment. See **Configure Custom Case and Incident**

1.3.3. Using Business Continuity Management

Workspaces

Utilities – BCM

- Used for importing data records (Resources, BIAs, Solutions, Tasks, Threat Scenarios).

Business Continuity Management

- Daily BCM workspace.
- Applications include: Approvals, BIAs, Resources, Threat Scenarios, Incident Repository, Inventory - Solutions, Solutions, Inventory - Tasks, Tasks, Unavailability Scenarios, and Documents.
- Provides dashboards for both program-level and owner-level views.

Application Records

Application Records are the data structures that power BCM. They establish relationships between business processes, resources, continuity plans, incidents, and approvals.

Resources

Represents assets (applications, buildings, suppliers, equipment, infrastructure, etc.).

Business Impact Assessments (BIAs)

Determines criticality of processes and defines recovery objectives. Includes **Initial Pre-Screening** and the **full BIA workflow** for Critical processes.

Inventory – Solutions

Continuity and contingency plans linked to BIAs and Threat Scenarios.

Inventory – Tasks

Step-level actions tied to continuity plans.

Threat Scenarios

Risks that may disrupt business operations (for example, cyber, human, natural, technical).

Incident Repository

Central record of continuity-related incidents.

Approvals

Workflow records for BIA validation.

Inventory – Documents

Supporting documentation linked to other records.

Unavailability Scenarios

1.4.7. SCF Evidence Application Overview

The solution builds the **SCF Evidence** application during installation. An SCF Evidence record stores information about audit evidence artifacts, collection methods, and review status. The application automatically populates up to 230 pre-defined evidence records from the SCF data based on what compliance frameworks are chosen during installation. The pre-defined evidence records are provided as a courtesy of Swimlane but certainly do not represent the whole body of evidence in the user's compliance program. The pre-defined evidence records will come with the following fields populated as read-only values:

Application Field Name	Desciption
ERL #	Evidence ID is the primary key of an evidence record.
Area of Focus	Single-select: Groupings of security related areas that the evidence relates to.
Documentation Artifact	The name of the evidence artifact.
SCF Control Mappings	Pre-mapped SCF control ID.
Artifact Description	Description of the evidence artifact.

SCF Evidence records also have editable fields to input information about the evidence such as collection type, evidence owner, and implementation notes.

Evidence Collection Types:

- A user can import files as evidence such as pdf, excel, or word documents.
- Users can provide an external URL link as evidence.
- A user can also link Turbine playbooks as evidence. This would be the case if the audit evidence is contained in a Turbine automation use case. This is a URL field where you can link a Turbine playbook URL. If a user does not have the "Orchestrator" role for the linked playbook, the URL will not work.

> Users can select multiple collection types and store as much evidence as they want per one evidence record. There are no hard and fast rules about what can be stored in an evidence record, it is up to the user and their organization.

For example: User uploads five information security policies as PDF's and also includes an external URL to a Confluence knowledgebase article within one evidence record.

Evidence Tracking Section

SCF Evidence records contain an "Evidence Tracking" section which is used to add notes about the last review of the evidence, who reviewed it, and when the evidence expiration date is. Finally,

1.6.1. Installing and Configuring Crafted AI Prompts

The Crafted AI Prompts extension integrates and works well with the Turbine SOC Solutions Bundle and the Case and Incident Management (CIM) application. This user guide uses the SOC Solutions Bundle and CIM application in examples to instruct how to utilize the Crafted AI Prompts extension.

> To best understand the documentation and examples, you can install and configure the SOC Solutions Bundle and the CIM application. For assistance in SOC Solutions Bundle Installation and Setup, contact your Swimlane professional services point of contact.

The Crafted AI Prompts extension is a collection of bundled components to create custom prompts to be sent to large language models such as **OpenAI** or **Claude** on **Amazon Bedrock**. For additional information, navigate to **Crafted AI Prompts**.

Start by installing the extension

Go to Marketplace

Navigate to the Turbine Marketplace. Follow these steps:

1. Log in to Turbine.
2. In the navigation pane, click **TENANTS**.
3. Select the desired tenant.
4. In the navigation pane, click **LIBRARY**.
5. In the LIBRARY navigation pane, click **Swimlane Content**.

Install

Now, you can install the Crafted AI Prompts extension:

1. From the Swimlane Content Library, under **Solutions**, click **Crafted AI Prompts Extension**.
2. Click **Install** on the right-bottom of the solution.
   The Crafted AL Prompts Extension details open.
   - The **Overview** tab shows each individual solution that composes the Crafted AI Prompts Extension and their capabilities.
   - The **Content** tab shows how many and which workspaces, applications, dashboards, and/or reports are included in the Crafted AI Prompts Extension.
   - The **Documentation** tab has a link to the Crafted AI Prompts Extension topic in this user guide.
3. On the right-hand side of the window, click **+Install** again.
4. From the Install Crafted AI Prompts Extension window, choose which content you'd like to install.

> You may see the following warning, if content already exists in your environment

1.8.2.3. Creating Ingestion Pages

When vulnerability data is imported into VRM—either via CSV or through a connector—the system automatically splits the data into multiple ingestion pages. These pages are designed to manage processing in batches, ensuring scalability and reliability even for large datasets.

Each ingestion page corresponds to a portion of the original dataset and is stored as a record in **VRM - Ingestion Page**.

Page Record Contents

Each ingestion page includes:

- **Status:** Tracks progress (e.g., Pending, Submitted, Completed)
- **Page Type:** Source of data (CSV or API)
- **Source:** Logical source name, used in mapping logic
- **Timestamp:** When the page was created/updated
- **JSON File:** Encoded subset of findings
- **Finding Count:** Number of findings in this batch
- **Sample Data:** Preview of the findings payload

Common Statuses:

- Pending Submission to Enrichment: Page is queued but not yet picked up
- Submitting to Enrichment: Page is actively being processed
- Successfully Submitted: All findings in the page have been sent for enrichment

Notes:

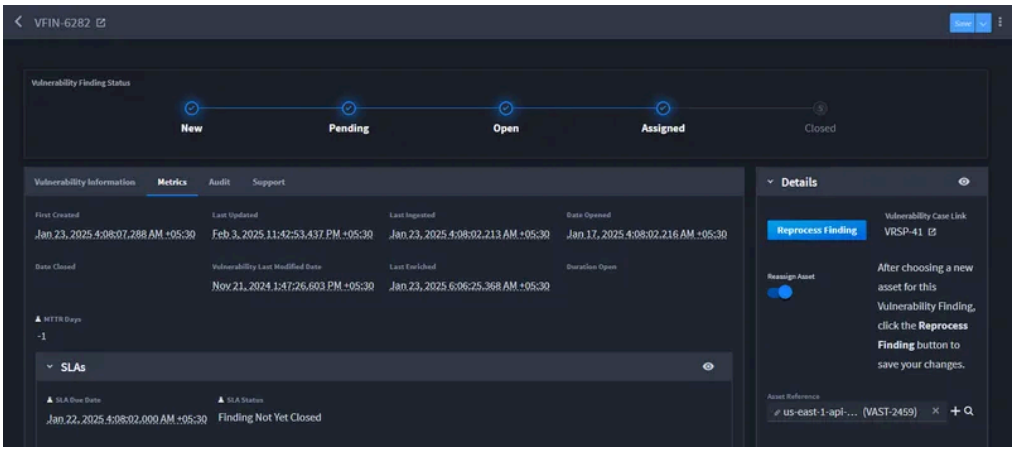
- **Page size** is defined in the import step (default: 1000). This controls how many findings go into each page.
- All ingestion pages are viewable in the **VRM - Ingestion Page** application.
- Errors or issues encountered during enrichment are reflected in page-level status and logs.

1.8.2.4. Processing and Enrichment

1. Scheduled Flow Picks Up Pages
    - A system flow runs every 10 minutes.
    - For each page:
        - JSON content is read
        - Findings are mapped via TEDS and sent to enrichment individually
        - Errors or duplicates are logged
2. Enrichment Pipeline Handles Each Finding
    - Enrichment applies logic like asset association, severity evaluation, and deduplication.
    - Deduplication uses **VRM - Export Results** to skip known findings

1.8.5.1.1. Vulnerability Findings – Metrics Tab

The Metrics tab provides detailed metadata about each vulnerability finding, including timestamps, SLA information, and duration metrics.



Key Components of the Metrics Tab:

- **Timestamps:**
  - **First Created:** Indicates when the finding record was first created.
  - **Last Updated:** Shows the most recent update to the finding.
  - **Date Opened:** Marks when the finding completed enrichment and a VFIN record was first created.
  - **Date Closed:** Displays when the finding was closed (if applicable).
  - **Last Ingested:** Timestamp of the latest ingestion of finding data.
  - **Last Enriched:** Indicates the most recent enrichment of the finding's metadata.
- **Duration Metrics:**
  - **MTTR (Mean Time to Resolution) Days:** Tracks the average time taken to resolve the finding.
- **SLA (Service Level Agreement):**
  - **SLA Due Date:** Displays the deadline for addressing the finding based on organizational SLAs.
  - **SLA Status:** Indicates whether the finding meets the SLA (for example,, "Finding Not Yet Closed", "No SLA Established", "SLA Met", or "SLA Exceeded").

Benefits

The Metrics tab ensures transparency and accountability by:

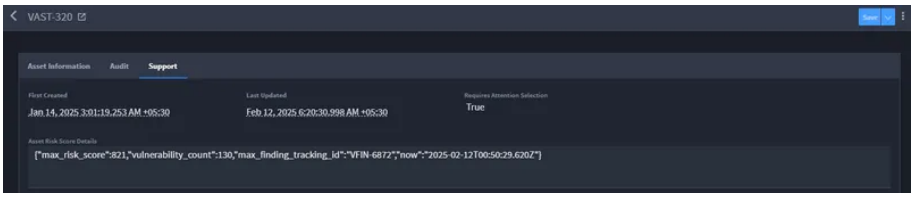
- Tracking the lifecycle of findings with precise timestamps.
- Highlighting SLA adherence to prioritize timely resolution.
- Providing key performance metrics for reporting and analysis.

1.8.5.6.2. Vulnerability Asset – Support Tab

The **Support Tab** in asset management provides critical metadata and risk details about an asset. This information helps security teams track, assess, and prioritize remediation efforts effectively.

Key Fields in the Support Tab:

- **First Created:** Displays the initial timestamp when the asset record was created.
- **Last Updated:** Shows the most recent update timestamp for the asset.
- **Requires Attention Selection:**
  - Indicates whether the asset requires immediate security attention.
  - True means the asset has a high-priority security concern.
- **Asset Risk Score Details:**
  - JSON-formatted data providing insights into the asset's security status, including:
  - Max Risk Score: The highest risk score recorded (821).
  - Vulnerability Count: Total number of detected vulnerabilities (130).
  - Max Finding Tracking ID: The most severe vulnerability associated with the asset (VFIN-6872).
  - Timestamp: The last recorded update for risk evaluation (2025-02-12T00:50:29.620Z).



1.8.5.7. Exception Management

To define a systematic approach for managing and documenting exceptions to identified vulnerabilities that deviate from established remediation protocols.

Key Elements of a Vulnerability Exception:

- **Exception Name:** A descriptive title summarizing the exception.
- **Description:** A detailed explanation of the exception, including its scope and impact.
- **Identifiers**
  - **Asset IDs:** Identifies the resources impacted by this exception. Use * for wildcard.
  - **Vulnerability IDs:** Specific vulnerabilities covered by this exception. Use * for wildcard.
  - **Zone IDs:** Resource zones impacted. Use * for wildcard.
- **Risk Scoring**
  - **Turbine Risk Score Range:** Defines the range of scores for the vulnerabilities affected. Use -1 for wildcards.