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

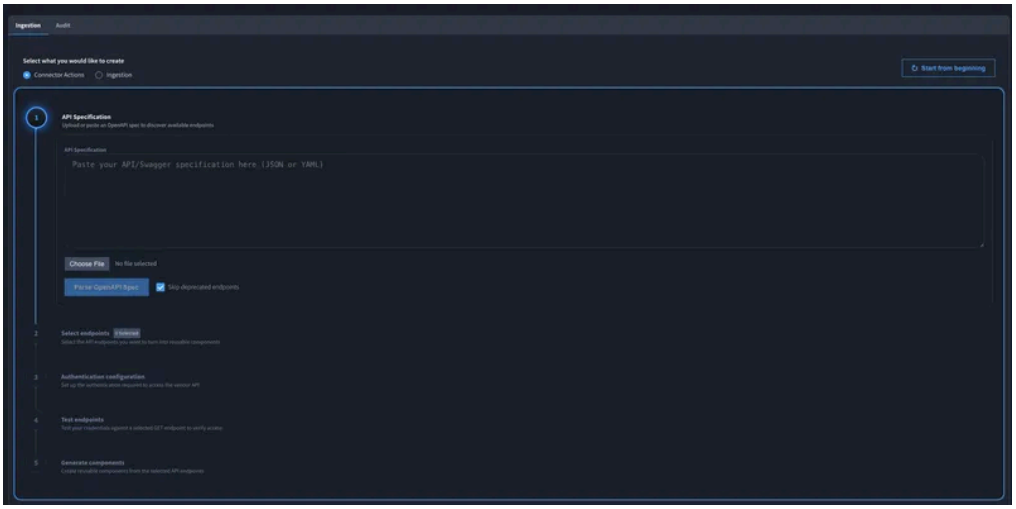
The **AI Ingestion** application helps you build vendor connectors and alert ingestion pipelines quickly — without writing code or learning JSONata. A guided custom widget builds Turbine components from an uploaded OpenAPI specification, one per endpoint. Hero AI then assists in mapping incoming raw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

Use this guide for **day-to-day analyst and operations workflows in the central tenant** after MSSP sync is configured. This page does not cover installing solution layers, configuring webhooks or assets, or onboarding a new client tenant.

Choose Your Path

If You Need To...	Go To
Configure webhooks, assets, or sync for client or central tenants	**Configure AI SOC for MSSP**
Add another customer client tenant	**Onboard a Client Tenant**
Fix missing or stale sync	**Validate and Troubleshoot MSSP Sync**
First MSSP deployment	**Getting Started for MSSP**
MSSP overview	**AI SOC for MSSP**

Daily Workflow

1. Open the **AI SOC MSSP Central** workspace.
2. Review incoming and recently updated records in **Central Case Management**.
3. Filter by **Client_Name** to inspect tenant-specific workloads.
4. Identify records requiring follow-up with client SOC teams.
5. Review TI cache updates for frequently recurring observables.
6. Open **Central SOC Reporting Workspace** for cross-client aggregates.
7. Use **Usage Statistics** to track workload and adoption across clients.

Central Reporting and Dashboards

Use central applications and workspaces to monitor clients. Client tenants also have **AI SOC MSSP Reporting** for local aggregates that sync to central.

Where	What to use
Central tenant	**Central Case Management**, **Threat Intelligence Artifact Cache**, **Central SOC Reporting**, **Usage Statistics**, **AI SOC MSSP Central** workspace, **Central SOC Reporting Workspace**
Each client tenant	**AI SOC MSSP Reporting** / **AI SOC MSSP Reporting Workspace**, and the analyst **AI SOC** workspace from Core

Reporting Goal	Recommended View
Compare workload by client	Central case list filtered by **Client_Name**

1.1.8.1. Playbook Flow Reference

This reference maps AI SOC playbooks and flows to triggers, inputs, outputs, and handoffs. Use it with **Playbook Types and Usage** (when to use each playbook) and **Architecture and Data Flow** (conceptual pipeline diagrams).

> **This is a reference page, not a starting point.** Read **Architecture and Data Flow** first for diagrams and the end-to-end story. Use this page when you need flow titles, triggers, handoffs, or playbook troubleshooting. For analyst procedures, see **Getting Started** and **Operations and Guidance**.

> Playbook bundle names, flow titles, and component counts can differ by installed package version. Treat flow tables marked **Verify in tenant** as outlines—confirm exact flow names and step order in **Orchestration** → **Playbooks** after import.

Choose Your Path

If You Need To...	Start Here
Understand what each playbook category does	**Playbook Types and Usage**
See how alert and email traffic converges on **Case Management**	**Playbook Handoffs at a Glance** → **Event Channels**
Configure a new SIEM or email source	**Ingestion Template Playbooks** → **Configure Ingestion Playbooks**
Trace what happens after a **CASE-** record is created	**CASE Lifecycle Flows**
Build or debug a routing rule playbook	**Routing Rule Playbooks** → **Building Routing Rule Playbooks**

How to Read This Reference

Each playbook section includes:

Column	Meaning
Flow	Flow title inside the playbook bundle (as shown in **Orchestration**)
Trigger	Sensor, schedule, flow event, or record event that starts the flow
Primary Inputs	Payload, record fields, or event data the flow consumes
Primary Outputs	Records, flow events, or fields the flow produces
Hands Off To	Next flow, event channel, application, or background process

Logical stage rows (for example, Deduplicate, Enrich observables) describe the ingestion pipeline documented in configuration guides when individual flow titles are not named in customer docs.

1.1.9.3. Using Dashboards and Reports Effectively

Morning Routine:

1. Open **Analyst Triage Queue** or **Security Operations Overview** dashboard
2. Review **Signals Requiring Attention** widget
3. Check **Signals : New** report for unclaimed signals
4. Review **Malicious & Critical** widget for active threats
5. Check **Cases Requiring Attention** for urgent cases

Throughout the Day:

- Monitor dashboard widgets for real-time updates
- Use reports to investigate specific patterns or issues
- Check **Signals : Oldest** periodically to prevent backlog

End of Day:

- Review **Signals By Status** to ensure signals are progressing
- Check **Cases By Status** for case workflow health
- Document any trends or issues observed

Weekly Reviews

Performance Analysis:

1. Review **Signals : Verdict & Severity Overall** for trends
2. Analyze **AI Verdicts** vs. **Manual Verdicts** for accuracy
3. Check **Routing Rule Management** for rule effectiveness
4. Review **Signals : Oldest** and **Cases : Oldest** for backlog

Optimization:

- Adjust routing rules based on match patterns
- Update priorities based on verdict and severity trends
- Identify and address workflow bottlenecks

Customizing Reports

Creating Custom Reports:

1. Navigate to the application (**Case Management**, **Threat Intelligence**, and so on.)
2. Go to **Reports** section
3. Click **Add** to create a new report
4. Configure filters, columns, and sorting
5. Save the report for future use

1.2.5. Configure Threat Intelligence Enrichment Integration

Configure Threat Intelligence Enrichment Integration

Threat Intelligence Enrichment gathers reputation information from observables, such as IP addresses, domains, URLs, hashes, email addresses, and so on from one or more enrichment providers using enrichment components. Results are aggregated in Threat Intelligence records, and displayed in Case and Incident Management records as well. Every observable type has a Primary Integlligence Provider (PIP) which is the canonical source of truth for reputation verdict, permalinks, and so on for that observable type.

Prerequisites

Before configuring Threat Intelligence Enrichment:

- Ensure you have access to the SOC Solutions Bundle playbooks
- Have enrichment provider credentials and assets configured
- Understand which observables you want to enrich (IPs, domains, URLs, hashes, etc.)

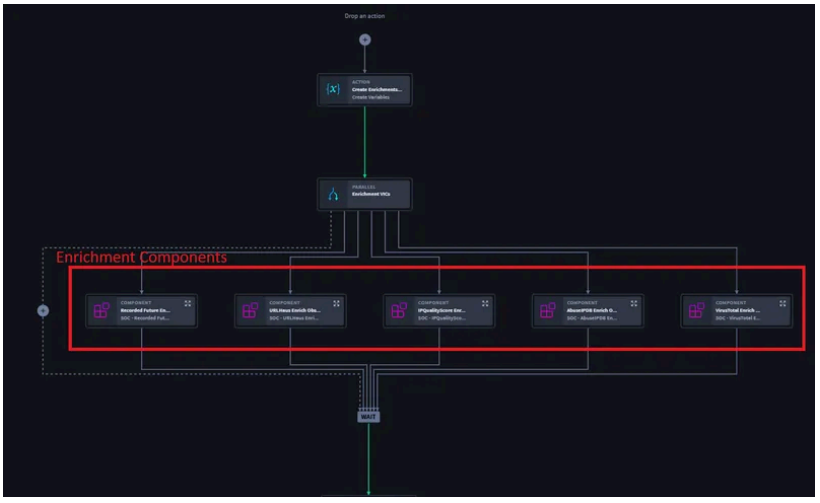
Configuration Steps

Step 1: Navigate to Components

1. Navigate to **Orchestration** in the Swimlane platform.
2. Click on **Components**.
3. Locate and open the **SOC - Enrich Observables** component (or **SOC - Enrich Observable** depending on your version).

Step 2: Configure Enrichment Sources

The component uses a Parallel node to run multiple enrichment sources simultaneously. You need to configure which enrichment sources to use:



1.4.1. Installing and Configuring

To install the CAR solution, you need to request the SSP from your Swimlane representative. Import the SSP into the Turbine application.

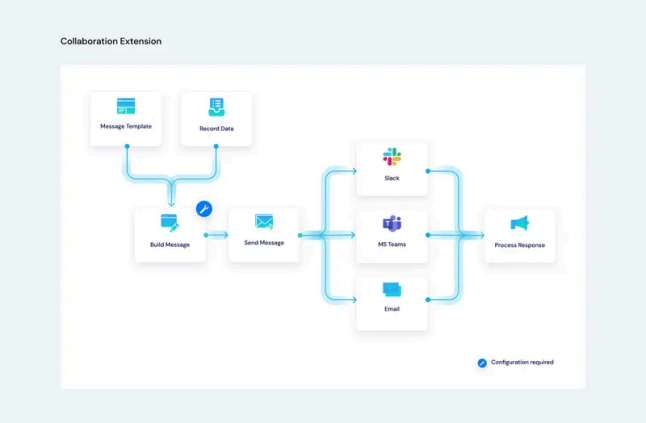
Configuring the solution:

1. Install the **SCF Common Controls and Evidence Management** connector from the **Swimlane Content**.
2. Navigate to **Orchestration** > **Assets**.
3. Click on **SCF Configuration Management**.
4. You can edit the **Title** and **Description** of the asset from the **Asset Settings**.
5. From the **Asset Inputs**:
    1. The **Version** field displays the SCF version.
    2. Select the frameworks by clicking **Add an item** from the **Selected Frameworks**.
    3. You can add the following frameworks using the drop-down:
        - AICPA TSC 2017-2022 SOC 2
        - APAC Australia Essential 8
        - APAC Japan ISMAP
        - CIS CSC v8.1
        - CIS CSC v8.1 IG1
        - CIS CSC v8.1 IG2
        - CIS CSC v8.1 IG3
        - COBIT 2019
        - CSA CCM v4
        - EMEA EU DORA
        - EMEA EU GDPR
        - EMEA EU NIS2
        - IEC 62443-4-2
        - ISO 22301 v2019
        - ISO 27001 v2022
        - ISO 27002 v2022
        - ISO 27701 v2019
        - ISO 42001 v2023
        - MITRE ATT&CK 10
        - NIST 800-53 rev5
        - NIST CSF v2.0
        - OWASP Top 10 v2021
        - PCI DSS v4.0
        - Shared Assessments SIG 2024

1.5. Collaboration Extension

Overview

The Collaboration Extension is a collection of bundled components to create context-aware messages from predefined templates to send to external messaging systems, such as email, Slack, and/or Microsoft Teams for processing by Swimlane, or non-Swimlane, users. These messages can trigger an action based on an arbitrary amount of user-defined choices, such as **Approve**, **Confirm**, **Deny**, or **Request Contact**. In this way, non-Swimlane users can participate in security workflows by responding to Turbine generated messages.



Capabilities

The Collaboration Extension can:

- Craft message templates in the **Collaboration Template Manager** application using mustache syntax.
  Example: {{Field Display Name}}
- Drag and drop the **Collaboration Extension Applet** into any application for your use case.
- Load, modify, and manage the resulting message in the **Collaboration Extension Applet** before sending (or send) the message automatically after loading.
- Use the dedicated and re-usable **Collaboration Message Sender** application designed to send messages over a variety of communication channels (Slack, MS Teams, email) and handle action-button responses from the recipient.
- Send messages to one or multiple recipients across one or multiple communication channels.

Common Use Case Examples

Some common use cases for Collaboration Extension include:

Practitioner Event Notification

Notify a team of practitioners (for example, security analysts, vulnerability analysts) across a desired communication channel (Slack, MS Teams, email) that an activity has occurred. Examples of possible activities are: specific alerts being generated, cases reaching desired checkpoints, or a new vulnerability being detected.

1.7.3. Detection Engineering Extension – How it Works

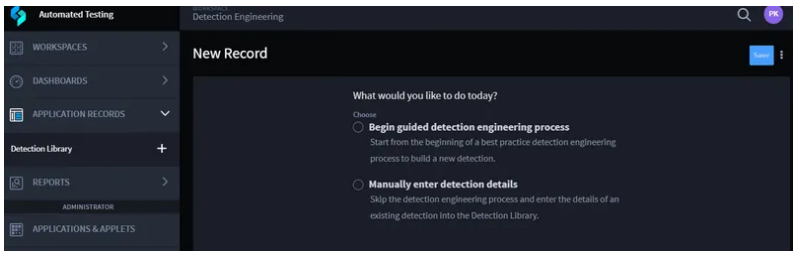
The extension equips Detection Engineers with essential tools to effectively identify and refine detections, ensuring continuous improvement and optimal performance of a SOC's detection capabilities. It also serves as a centralized system of record, maintaining an up-to-date overview of the organization's detection posture. Additionally, the solution enhances the feedback loop between analysts and Detection Engineering by streamlining the collection of detailed feedback directly into the Detection Library application.

Detection Engineering Guided Process Workflow

After you install the Detection Engineering extension:

1. From the **APPLICATION RECORDS**, click next to the **Detection Library**.

2. A workflow to create a **New Record** is displayed.
3. From the **New Record** window, select the desired flow for the detection library. The guided workflow is the recommended option.



4. The guided process widget provides a step by step process guidance on the recommended steps a Detection Engineer should begin to think about the entire process from:
    - Determining a Threat Model
    - Define Scope
    - Determine Required Log Sources
    - Design Detection
    - Test Detection Logic
    - Optimize Detection Logic
    - Stage Detection
    - Craft Detection Runbook
    - Deploy to Production
5. Fill all the fields and check the **Mark as complete** checkbox to save the record.

1.8.2.7. Monitoring Ingestion and Enrichment

The **VRM Utilities Dashboard** and **VRM – Ingestion Page** application provide real-time visibility into the ingestion and enrichment status of vulnerability findings. These tools help users track progress, verify successful processing, and troubleshoot any issues with imported scan data.

Dashboard Features

- **Ingestion and Enrichment Queue:** Displays the total number of vulnerability findings currently submitted for enrichment.
- **Enrichments Over Time:** Shows a time-based chart of findings enriched, enabling users to monitor trends and processing activity over selected periods.
- **Vulnerability Findings Ingested & Submitted to Enrichment:** Indicates the count of findings that have been successfully imported and forwarded for enrichment.
- **Vulnerability Findings Ingested from CSV:** Shows the number of findings imported via CSV file uploads.
- **Submitted Vulnerability Findings by Type:** Breaks down findings by import source or type, such as CSV or API, providing insight into where vulnerability data is coming from.

Using the Dashboard

- Track Processing Progress: Monitor the flow of findings from initial ingestion through enrichment using live counters and charts.
- Verify Completion and Status: Use batch and finding statuses to confirm that all expected data has been ingested and processed.
- Troubleshoot Issues: Identify and investigate any discrepancies between findings ingested and findings enriched. Use error statuses or logs in the VRM - Ingestion Page application to review and address failed or incomplete batches.
- Optimize Operations: Leverage trend data from the Enrichments Over Time chart to refine scan schedules or ingestion timing for optimal system performance.

1.8.2.7.1. Accessing Detailed Batch Information

The **VRM – Ingestion Page** application provides a detailed view of each ingestion batch processed by the system. This view allows users to track the progress, verify the outcome, and troubleshoot any issues related to vulnerability scan imports.

How to Access Batch Details

1. Navigate to the **VRM – Ingestion Page**.

1.8.5.3.1. 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.

1.8.5.3.2. 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**
  - Allows users to manually update the risk score based on new assessments or findings.
  - Useful when the severity of a vulnerability changes over time.
- **First Created**
  - Displays the exact timestamp when the case was initially logged.
  - Helps in tracking the age of the case.
- **Last Updated**
  - Shows the most recent timestamp when the case was modified.
  - Useful for monitoring case activity and ensuring timely updates.
- **Case Owner**

1.8.7.1. 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 Reference	finding.vulnerability-finding-asset-reference	array
Vulnerability Finding Asset Zone	finding.vulnerability-finding-asset-zone	string
Vulnerability Finding Exception Reason	finding.vulnerability-finding-exception-reason	string
Vulnerability Finding Exception Reference	finding.vulnerability-finding-exception-reference	array
Vulnerability Finding Grouping ID	finding.vulnerability-finding-grouping-id	string
Vulnerability Finding Hostnames	finding.vulnerability-finding-hostnames	array
Vulnerability Finding IP Addresses	finding.vulnerability-finding-ip-addresses	array
Vulnerability Finding Last Enriched	finding.vulnerability-finding-last-enriched	string
Vulnerability Finding Last Ingested	finding.vulnerability-finding-last-ingested	string