Cybersecurity & Risk

1. Cybersecurity & Risk: Overview

The **Cybersecurity & Risk** module serves as the central hub for all security operations within your environment, providing a unified platform to safeguard system integrity and protect sensitive data. By integrating multiple layers of defense and proactive monitoring tools, Security Core enables your organization to anticipate, detect, and respond to threats in real time.

Security Core includes the following sections:

1. Incident Reports
2. Violation Reports
3. Malware Reports
4. Penetration Reports

Benefits at a Glance

1. **Holistic Visibility:** Correlate data across incidents, vulnerabilities, malware events, and testing results to identify systemic weaknesses and emerging attack vectors.
2. **Faster Response:** Automated alerts, playbooks, and remediation tools reduce mean time to detect (MTTD) and mean time to respond (MTTR).
3. **Regulatory Compliance:** Maintain comprehensive logs, audit trails, and evidence of controls to satisfy GDPR, HIPAA, PCI DSS, and other industry mandates.
4. **Continuous Improvement:** Ongoing vulnerability scans and regular pen tests feed back into your security strategy, driving a cycle of assessment, remediation, and validation.

Use Cases

#1. Security Assurance
Detects and manages vulnerabilities to prevent security breaches, ensuring system integrity.

#2. Operational Continuity
Minimizes disruptions by quickly identifying and resolving issues, maintaining business operations.

#3. Improved Collaboration
Facilitates seamless tracking and assignment of tasks among teams, enhancing efficiency.

#4. Root Cause Analysis
Collects closure details for lessons learned, helping prevent similar incidents in the future.

Security Core ensures robust protection by providing structured workflows for detecting, tracking, and resolving security threats across the ecosystem.

2. Incident Reports

The **Incident Reports** section in Security Core is essential for tracking and managing security-related or general system issues such as bugs, vulnerabilities, and unexpected behavior. It ensures efficient incident resolution, reduces downtime, and strengthens system security by providing structured workflows for issue management.

Use Cases

1. **Create Incident:** A security engineer or developer logs a new issue, ensuring all security concerns are documented.
2. **Assign & Update:** The incident is assigned to the relevant person, severity is set, and progress is tracked for accountability.
3. **Track in Board:** Team members move the incident card across workflow stages (Open → In Progress → Resolved) for clear visibility.
4. **Closure:** Upon closing, the system collects details on the resolution, supporting files, and lessons learned to prevent future occurrences.

Where to Use Incident Reports

1. **Software Development**: To track bugs and unexpected system behavior during development and deployment.
2. **Financial Systems**: For monitoring and resolving vulnerabilities in transaction processes.
3. **Web Applications**: To manage issues affecting user experience, performance, and security.
4. **Enterprise IT**: For handling system outages, security incidents, and compliance issues across departments.

Key Components of Incident Reports

> The Incident Reports Action tracks and manages security-related or general system issues (e.g., bugs, vulnerabilities, unexpected behavior). It provides two main views (Table and Board), plus the ability to filter, change statuses, and add closure details.

1. Table View

Use **Table** view for a detailed, spreadsheet-style list. Columns include:

Column	Description

Severity ⇅	Icon indicating Low (↓), Medium (=), High (↑), or Critical (⇈). Sortable.
Project ⇅	Name of the project this incident belongs to. Sortable.
Name	Clickable incident title; opens the edit drawer.
Assigned to ⇅	Engineer(s) responsible (name + UID). Sortable.
Component	Affected system component or module.
SLA	Target remediation date/time. **If the SLA has passed and the incident is *not* Resolved**, the entire SLA cell is highlighted **red** to draw immediate attention.
Created at ⇅	Timestamp when the incident was first logged. Sortable.
Updated at ⇅	Timestamp of the most recent update. Sortable.
Status ⇅	Current status badge (Open, In Progress, Resolved). Click to change. Sortable.
Actions	✏ Edit opens the side panel; ◫ Delete prompts confirmation.

> **Incident Reports – Clickable Name**
> In the Table view for Incident Reports, the **Name** column entries are clickable: clicking any Name opens the full-width “View Incident Report” drawer, showing all fields, lesson-learned history, attachments, root-cause analysis, and summary.

Status Control: Click the colored status label to choose a new state from a dropdown.

Edit Incident:
Click the ✏ Edit icon in the Actions column to open the side panel.
Update fields like Assigned to, Severity, Status, SLA, or Component.

> **Wider Detail Drawer (Resolved Incidents):** When an incident’s status is Resolved, the side-panel pops out in an **expanded, full-width** layout to neatly display your **Lesson learned** history, attachments, screenshots, or long-form notes without cramped columns or horizontal scrolling.

Post-Resolution Editing

Even after an incident is marked **Resolved**, you can still open the Edit drawer (✏ icon) and change its details:
Editable Status: The **Status** field remains a dropdown—click it to switch from **Resolved** back to **Open** or **In Progress**, or vice versa.
Full Field Access: All other fields (Component, Severity, SLA, Description, Summary, etc.) remain

writable. After making adjustments, click **Save** to update the record.

2. Board View:
Kanban-Style Board: Switch to Board view to see incidents arranged by status column (e.g., Open, In Progress, Resolved).
Drag & Drop: Move incident cards between columns to reflect status changes.
Resolving an Incident: When you move an incident to Resolved, a dedicated form may appear, prompting you to describe how the issue was fixed, attach any proof or files, and add a root cause analysis or lessons learned.

3. Filtering & Searching:
Filter Panel: Click **Filter** to open filters for **Status, Severity**, and **Assigned to**. Select your criteria and click **Save** to narrow the list or board.
Search Bar: In either view, type a full or partial incident name (or keyword) into the **Search** field to locate specific reports instantly.

How to Delete an Incident Report

Deleting an Incident Report removes it permanently from the system and records who performed the deletion and which report was removed. Follow the steps below.

Prerequisites:
You must have **Delete** permissions on the **Incident Reports** module.
Ensure you really intend to remove the record, as deletion cannot be undone.

Deletion Steps

1) Open the Incident Reports List:
In the left-hand navigation, select **Incident Reports**.
Locate the row for the report you wish to delete.
2) Trigger Deletion: In the **Actions** column of that row, click the **Delete** icon.
3) Confirm Deletion:
A confirmation pop-up appears:

> **Are you sure?** [Cancel] [Delete]

Click **Delete** to proceed.
4) Completion: The report is removed from the table.
A success toast or message confirms the deletion.

Audit Logging

Every deletion is recorded in the system audit log to maintain a clear trail of administrative actions. The log entry includes:
Report Name – The title or unique identifier of the deleted incident report.
Deleted By – The username of the person who performed the deletion.
Timestamp – When the deletion occurred.

> **Tip:** Regularly review audit logs to ensure all deletions were intentional and comply with your data-retention policies.

3. Violation Reports

Violation Reports generally refer to compliance or policy violations that an automated scanner identifies. For instance, a daily script might check your codebase or server configurations and log any suspicious results:

1. **NPM**: Could be scanning for vulnerable dependencies in a Node.js project.
2. **SERVER_SCAN**: Might check server configurations, open ports, or outdated libraries.
3. **SYNC**: Another custom tool or integration that reports code or config discrepancies.

> Once a violation is “found,” security engineers review it, assign it a State (e.g., “In progress”), and, after investigation, mark it “Resolved” or “Not processed” if it’s a false positive or low priority.

Use Cases

#1. Updating Vulnerable Dependencies
A daily NPM scan detects outdated packages in a Node.js project. Engineers mark the report as "In progress", update the dependencies, and resolve the issue.

#2. Server Configuration Errors
A SERVER_SCAN identifies open ports. The IT team secures the ports and marks the violation as "Resolved".

#3. Sync Discrepancies
A SYNC scan flags code inconsistencies after deployment. Developers review the logs, sync configurations, and close the report.

#4. False Positives Management
An automated scan reports a minor issue. The security team reviews the report and marks it as "Not processed" if deemed harmless.

Typical Workflow

1. Daily/Periodic Scans
A security scanner (via API integration, not by default) runs on a server or code repository on a set schedule, reporting:
"notFound" – No issues detected.
"found" – Issues identified for review.

2. Report Creation
The system automatically creates a Violation Report entry, or a security engineer manually logs it. Fields include:

Server name: Which server was scanned.
Tool: Name of the scanning tool (e.g., NPM, SERVER_SCAN, SYNC).
Result: Was a violation discovered (found) or not?
State: Whether the issue is “Not processed,” “In progress,” or “Resolved.”
Project: Which project or environment the server is linked to.
Created at/Updated at: Timestamps for when the record was created or last updated.
Description: Any extra details or logs from the scan.

3. Engineer Review
A security engineer checks the new violations.
If the issue needs action, they mark it as “In progress.”

> Once it’s handled or deemed harmless, they set State to “Processed” (or a similar status).

Table View

Use **Table** view for a spreadsheet-style overview, sortable and filterable by any column. By default, you’ll see:

Column	Description
Severity ⇅	Visual severity icon (— for Medium, ↓ for Low, ↑ for High/Critical). Click to sort by severity level.
Created at ⇅	Timestamp when the report was first logged.
Title	Clickable report name; opens the Edit panel.
CVSS v3 Score	The numeric CVSS score (e.g. 7.5).
Assigned to	One or more engineer names/UIDs.
Tool ⇅	Scanning tool (e.g. NPM, SYNC, SERVER_SCAN). Click to sort.
Scan type	Code base or Server scan.
Component	If Code base → module or repo path.
Server name	If Server scan → hostname or IP.
Project	Linked project name.
SLA	Target remediation date / time.
Updated at	Timestamp of the most recent update to the report (status change, reassignment, or edit).
Status	Current workflow state of the report (Open, In Progress, Resolved), editable directly from the table via dropdown.

> **Overdue alert:** If the SLA has expired and the report is not closed, the SLA cell is shaded **red** to draw immediate attention.

> **Violation Reports – Clickable Title**
> In the Table view for Violation Reports, the **Title** column entries are clickable: clicking any Title opens the full-width “View Violation Report” drawer, displaying all of that report’s fields, history, attachments, resolution summary, and close details.

Sorting & Total: Sort reports by any column. The **Total** count shows how many entries match your current view.

Board (Kanban) View

> **Board** view—a Kanban-style layout groups reports into columns by **Status**. Drag & drop cards between **Open**, **In Progress**, **Resolved** to update their status in real time.

Use the **Board** view for a high-level, drag-and-drop workflow:

Columns: One column per status—

1. **Open**
2. **In Progress**
3. **Resolved**

Cards: Each report card shows:

1. **Title**
2. **Created at** (with calendar icon)
3. **Snippet of Description**
4. **CVSS score** badge in the top-right

Adding a Violation Report

To log a new compliance or policy violation:

1. Open the Add Form
Click the **+ Add** button in the top-right corner of the **Violation reports** table.

2. Fill in the Report Details
In the “Add violation report” side panel, complete the following fields:

1. **Title:** A short, descriptive name for the issue (e.g. “SQL Injection in Login”).

2. **CVSS v3:** Enter the numeric vulnerability rating (e.g. 7.5) based on the Common Vulnerability Scoring System.
3. **Severity:** Manual classification of the issue (Low, Medium, High, Critical) used for visual prioritization.
4. **Tool:** Select which scanner or pen-test tool generated this report.
5. **Scan Type**
   ▸ **Codebase** → reveals an extra **Component** text field (e.g. the repo path or module name).
   ▸ **Server Scan** → reveals **Server IP** and **Server Hostname** fields.
6. **Component** *(only if Codebase):* Free-text name of the sub-system or code module affected.
7. **Server IP** & **Server Hostname** *(only if Server Scan):* Identify the scanned host (e.g. 192.0.2.15 / api-prod-01.example.com).
8. **Assigned to:** Pick one or more engineers responsible for triage.
9. **Project:** Link this report to the appropriate project or environment.
10. **SLA** *(optional):* Set a target remediation date/time.
11. **Penetration report** *(optional):* Link to a related pen-test entry if available.
12. **Description:** Use the rich-text editor to paste or type detailed logs, error messages, or remediation notes.

3. Save the Report
When all mandatory fields are populated, click **Save** to create the new Violation Report.
The report will now appear in your table (and board) views, ready for review and triage.

Editing a Violation Report

1. Locate the record
In **Table** view, scroll or search to find the row for the violation you want to update.
In **Board** view, find the card in its status column.

2. Open the edit form:
Table: Click the **Edit** (✏) icon in the **Actions** column.
Board: Hover over the card and click the pencil icon or the "⋯" menu, then choose **Edit**.

3. Make your changes
In the side-panel form you can update any field:

1. **Status** (Open, In Progress, Resolved, etc.)
2. **Severity**
3. **Assigned to**
4. **Scan type**, **Tool**, **Component**, **Server name**
5. **SLA**, **Penetration report**
6. **Description** (detailed notes or logs)

4. Save: Click **Save** at the bottom of the panel to apply your edits.

Closure Workflow

When you mark a Violation Report “Processed,” it now—rather than simply updating the status—opens a mandatory “Close Report” dialog so you capture a concise **Resolution Summary**. This guarantees every closed finding has:
Complete Context: How it was fixed or verified
Accountability: Who closed it and when
Audit Trail: Full details bundled into one log entry

What’s New

1. **Close Dialog Auto-Opens:** As soon as you set a report’s status to **Processed**, the **Close Report** modal pops up—pre-filled with all original fields and forcing you to enter a **Resolution Summary** before the change can be saved.
2. **Mandatory Summary:** You cannot finish without entering a brief resolution note.
3. **Data Snapshot:** Read-only view of all original fields (Project, Title, CVSS, Severity, Tool, Scan Type + Component/Server, Description).
4. **Atomic Audit Log:** The system records the summary, closer’s username, and timestamp together.

How It Works

1. **Locate & Process:** In Table or Board view, set State → Processed.
2. **Review Snapshot:** Confirm Project, Title, CVSS v3, Severity, Tool, Scan Type details, and Description.
3. **Add Summary:** Enter your remediation steps, verification, and notes.
4. **Save to Close:** Click **Save**; the summary appears in details and audit logs.

> **Benefit:** Every closed report is now a self-contained record of what was found, who fixed it, how, and when—making compliance and troubleshooting faster and more reliable.

Deleting a Violation Report

1. **Find the violation:** In **Table** view, locate the row you wish to delete.
2. **Click the trash icon:** Click the **Delete** (□) icon in the **Actions** column for that row.
3. **Confirm deletion:** In the confirmation dialog, click **Delete** again to permanently remove the report

> **Warning:** Deleted violation reports cannot be restored. Be sure you no longer need the record before confirming deletion**.**

Filtering & Searching

1. **Filter Panel:** Click **Filter** to narrow by **Status** or **Tool**.
2. **Search Bar:** Type a partial or full server name in the **Search** field to find specific reports instantly.

4. Malware Reports

Malware Reports track the output of antivirus or anti-malware scans on servers. Common tools include:

- **ClamAV** (open-source antivirus)
- **Rootkit** detection scripts

Use Cases

1. **Detecting Server Malware:** A CLAMAV scan detects malware in email attachments. Security isolates the files and marks the report as "In progress" for further analysis.
2. **Rootkit Detection:** A ROOTKIT scan finds hidden malicious processes. Engineers remove the infected files and mark the report as "Resolved".
3. **Scheduled Security Checks:** Weekly malware scans report no issues. Security logs the "Found = false" status and archives the report.
4. **Emergency Malware Response:** Malware is detected during a live incident. The security team performs an immediate investigation, quarantines infected files, and completes a system clean-up.

Table View

Total: (top-left) shows how many reports are in your system.

Filter launches a sidebar to narrow your list by:
Scan type (e.g. CLAMAV, ROOTKIT)
State (Not processed • In progress • Resolved)
Project
Search finds any term in server names or descriptions.
+ Add (top-right) opens the “Add malware report” form.

Columns

	Column Name ⇅	What It Shows
☑	(checkbox)	Select individual rows for bulk actions.
1	**Server name**	Hostname or IP address scanned.
2	**Project**	Link to the project/environment.
3	**Scan type ⇅**	Which tool ran (CLAMAV, ROOTKIT, etc.).
4	**Vulnerabilities ⇅**	“Detected” or “Not found” based on scan.
5	**Created at ⇅**	When the report was first logged.

	Column Name ⇅	What It Shows
6	**Updated at ⇅**	When any field was last changed.
7	**State ⇅**	Processing status (Not processed, etc.).
8	**Actions**	• ✏ Edit • ◫ Delete

> Security engineers then mark the report as “In progress” to investigate or “Resolved” if no further action is needed.

Adding a Malware Report

1. Click **+ Add**.

2. In the “Add malware report” form:
Server name: Enter the machine’s name or IP.
Scan type: Choose from your configured tools (ROOTKIT, CLAMAV, etc.).
Project: Link it to the correct project.
State: Select “Not processed,” “In progress,” or “Resolved.”
Malware found: Check this box if the scan flagged any threats (it’ll show “Detected” under Vulnerabilities).
Description: Summarize any details or remediation steps.

3. Click **Save**. The new row appears in the table.

Editing Reports

Edit: Click the ✏ icon in the Actions column to open the side-panel. You can change **Server name**, **Scan type**, **State**, **Malware found**, or update the **Description**. Then hit **Save**.

> If action is required, they set the State to “Processed” or “Not Processed.”

Filtering Malware Reports

To narrow down the list of malware reports, use the **Filter** panel available at the top of the Malware Reports table.
To open filters, click **Filter** in the upper-left corner of the table. A sidebar will appear with the following options:

State
Filter reports by their processing status:

1. Not processed
2. In progress
3. Resolved

This helps track which reports still require investigation versus those already handled.

Scan type
Limit results to reports generated by a specific malware detection tool, such as:

1. CLAMAV
2. ROOTKIT

Vulnerabilities
Use these checkboxes to control whether reports with or without detected threats are shown:

1. Show issues with detected vulnerabilities
2. Show issues without detected vulnerabilities

This is useful for quickly isolating confirmed incidents or reviewing clean scan results.

After selecting the required parameters, click **Save** to apply the filters.
To change the filter set, reopen the panel and adjust the selected values.

5. Security Assessments

Security Assessments or Penetration testing is the practice of simulating attacks on a system or application to uncover security weaknesses:

1. **Black Box**: The tester has no prior knowledge of the system.
2. **White Box**: The tester has detailed knowledge of the system.
3. **Gray Box**: Some knowledge is provided, but not full.

Use Cases

1. **Black Box Testing:** External testers find a login vulnerability. The team patches the issue and retests for confirmation.
2. **White Box Testing:** Full system knowledge reveals code injection risks. Developers implement code fixes and resolve the report.
3. **Gray Box Testing:** Limited access tests expose endpoint vulnerabilities. Engineers secure the endpoints and log retesting results.
4. **Retesting After Fixes:** Vulnerabilities are fixed post-penetration test. Follow-up tests are conducted to ensure no further risks remain.

> Pen testers document discovered vulnerabilities and exploitation paths. In the system, you'd log each test (or each portion of a test) as a Penetration Report, noting the Name and any steps or results in the Description. Security teams typically use it to confirm that known vulnerabilities are patched and no new ones have appeared.

Table View

1. **Total:** (top-left) shows how many penetration reports exist.
2. **Search…** quickly filters by any term in the **Name** or **Description**.
3. **+ Add** (top-right) opens the "Add penetration report" form.

Column	Details
Name ⇅	Title of the test (e.g. "Denial of Service," "Open Redirect"). Clicking the link opens full details.
Description	One-line summary of what was tested or discovered.
File	Uploaded assessment file (e.g. penetration test report or supporting document), downloadable directly from the table.
Project	Link to the related project or environment.

Column	Details
Created at ⇅	Date and time when the report was logged.
Actions	✏ Edit

There’s no built-in delete option for penetration reports—entries are archived by editing or by policy.

> **Penetration Reports – Clickable Name**
> In the Table view for Penetration Reports, the **Name** column entries are clickable. Clicking any Name opens the full-width **View Penetration Report** drawer, showing that report’s Name, Description, Created at timestamp.

Viewing Linked Violation Reports

You can now see which Violation Reports were raised as a result of each penetration test—right in the Penetration Reports table.

Expand the row: In the leftmost column of any report row, click the ▼ arrow.

Review associated violations: A sub-row appears listing each Violation Report linked to that Pen test (with Title, Status, and Date).
Click a Violation Report title to open its detail panel.

No linked violations? You’ll see “No data to display” if no Violation Reports are attached yet.

Adding a Penetration Report

1. Click **+ Add**.
2. In **Add penetration report**:
-Name: Enter a clear title for the engagement.
-Description: Summarize the scope and key findings.
-Project: Select the associated project.
-Attached files: Drag an image or browse to upload one or more PDF documents (e.g. your full pen-test report).
3. Click **Save**. Your new report appears in the table.

Editing a Penetration Report

Click the ✏ icon under **Actions**.
In the **Edit penetration report** panel, update the **Name**, **Description**, or **Project**.
Attached files: Drag an image or browse to upload additional PDFs or replace existing attachments.

Click **Save** to apply changes.

Typical Workflow

Pen Test Execution: Security team or external vendor runs tests (e.g., vulnerability scans, manual exploitation, stress tests).
Report Logging: Each test campaign is logged with a **Name** and **Description** of findings (e.g., “SQL injection found in search endpoint”).
Review & Action: Security engineers review findings, tag them to development/ops teams, and track fixes.

Once remediated, tests may be rerun and the report updated to reflect the final status.