

REFERENCE HANDBOOK

Product Documentation

Platform / API / Galileo CLI

A handbook for providers and clients



ENGLISH EDITION

Complete public documentation

hashiro.ai

How to use this handbook

A practical guide for Hashiro providers and clients.

01 / Start with your role

The provider and client guides explain each organization's capabilities and responsibilities. Then use the platform chapters for everyday tasks.

02 / Navigate the reference

The table of contents includes links and page numbers. PDF bookmarks group the material by part, chapter, and section. Links within the handbook take you directly to the relevant heading.

03 / Read examples in context

Screenshots contain fictional demonstration data. Available profiles, controls, and features depend on organization settings and permissions.

04 / Use the integration references

Galileo CLI and the API have their own parts, covering installation, configuration, commands, and examples. The public API uses <https://api.hashiro.ai/api/>.

05 / Keep the handbook current

This handbook is a snapshot of the public documentation. The website's Downloads area provides subsequent editions. Internal administration documentation is excluded from this PDF.

Contents

Select a chapter or use its page number.

01 / Getting started	5
01.01 / Introduction	6
01.02 / Quick Start	8
01.03 / Provider guide	11
01.04 / Client guide	13
01.05 / Authentication	15
02 / Platform handbook	18
02.01 / Dashboard	19
02.02 / External Attack Surface Management (EASM)	22
02.03 / Assets	25
02.04 / Services	28
02.05 / Issues	31
02.06 / Scope configuration	34
02.07 / Cloud Security	36
02.08 / Connect a cloud account	39
02.09 / Misconfigurations	42
02.10 / Compliance	46
02.11 / Projects	48
02.12 / Scope Management	51
02.13 / Narrative	53
02.14 / Attack Chains	54
02.15 / Report Generation	56
02.16 / Vulnerability Management	58
02.17 / Issue Lifecycle	61
02.18 / CVSS Scoring	64
02.19 / AI-Assisted Review	66
02.20 / AI Assessments	68
02.21 / AI Runs	70
02.22 / Local Runs	75
02.23 / Purple Team Runs	78
02.24 / Agent Types	81
02.25 / Automatic Assessments	84
02.26 / Run Modes	85
02.27 / Checklists	88

02.28 / AI Schedules	90
02.29 / Connectors	92
02.30 / Knowledge Base	94
02.31 / Credits & Billing	96
02.32 / Inventory	98
02.33 / Explore	101
02.34 / Project Calendar	103
02.35 / Templates	105
02.36 / Vulnerability Templates	106
02.37 / Nuclei Templates	108
02.38 / Report Templates	110
02.39 / Share Links	112

03 / Organization & teams 113

03.01 / Organization Management	114
03.02 / Team & Users	117
03.03 / Roles & Permissions	120
03.04 / Customer Management	122
03.05 / Notifications	125
03.06 / Settings	128
03.07 / Integrations	129
03.08 / Jira Integration	130

04 / Galileo CLI 133

04.01 / Galileo CLI	134
04.02 / Installation	135
04.03 / Configuration	137
04.04 / Command Reference	140
04.05 / Imports	151

05 / API reference 153

05.01 / Hashi API	154
05.02 / Authentication	156
05.03 / Projects and scope	158
05.04 / Assets, URLs, and services	160
05.05 / Findings and review	162
05.06 / Organizations and users	164



PART 01

Getting started

Understand the platform, your role, and the first steps.

CHAPTERS / 05

01.01

Introduction

Hashiro is a **Continuous Threat Exposure Management (CTEM)** platform that gives security teams full visibility into their external attack surface, cloud infrastructure, and vulnerability posture from a single console.

Who is Hashiro for?

- **Security teams and SOCs:** centralize asset discovery, vulnerability tracking, and remediation across your organization.
- **Penetration testers:** manage engagements end-to-end with scoping, asset enumeration, finding documentation, attack chains, and report generation.
- **MSSPs and security providers:** serve multiple clients from one platform with tenant isolation, white-labeled reporting, and aggregated estate dashboards.
- **Enterprises:** maintain continuous visibility into your external exposure and cloud posture without assembling a patchwork of point tools.

Core Capabilities

Attack Surface Management (EASM)

Continuously discover and monitor your external attack surface. Hashiro enumerates subdomains, probes HTTP services, detects technologies, captures screenshots, and flags exposed services on a schedule you define.

Cloud Security

Connect your cloud accounts (AWS, Azure, GCP) to detect misconfigurations, track compliance against CIS benchmarks, and monitor your security posture across providers.

Projects

Manage security assessment engagements: penetration tests, bug bounty programs, and security reviews. Define scope, track assets, document findings, build attack chains, and generate PDF reports.

AI-Powered Assessments

Run autonomous security assessments driven by AI agents. Configurable agent types execute multi-stage pipelines (discovery, vulnerability analysis, exploitation, and reporting) with human-in-the-loop or fully autonomous modes.

Vulnerability Management

Track security findings through their full lifecycle: draft, open, retesting, remediation, and closure. CVSS v3.1 scoring, audit trails, AI-assisted review, and candidate verification workflows keep your team aligned.

Inventory & Explore

Search and filter across all assets, URLs, services, and findings across every project in your organization. Faceted filtering by technology, web server, CDN, and more.

Templates

Reusable vulnerability templates, custom Nuclei scanning templates, and LaTeX-based report templates accelerate consistent, professional output.

Integrations

Bidirectional Jira integration, API tokens, the Galileo CLI, and webhook support connect Hashiro into your existing workflows.

How the Platform is Organized

Section	Purpose
Dashboard	At-a-glance KPIs, trends, and alerts
Attack Surface	EASM asset discovery and monitoring
Cloud	Cloud security posture management
Projects	Security assessment engagements
Inventory	Cross-project asset search
Explore	Full-text and regex search across all data
Calendar	Project timeline planning
Templates	Vulnerability, Nuclei, and report templates
AI Library	Agent types, checklists, schedules, connectors
Organization	Team, roles, integrations, settings

Multi-tenant by design Hashiro supports **provider** organizations (security firms) managing multiple **client** organizations, as well as standalone organizations using the platform directly. Tenant isolation is enforced at every layer.

Next Steps

- [Quick Start](#): set up your account and run your first scan
- [Authentication](#): learn about login, MFA, and API tokens

01.02

Quick Start

Get up and running with Hashiro in minutes.

1. Sign In

You'll receive an email invitation from your organization administrator or provider. Click the invitation link to set your password and activate your account.

Use the [Hashiro console](#) for your invitation and account setup.

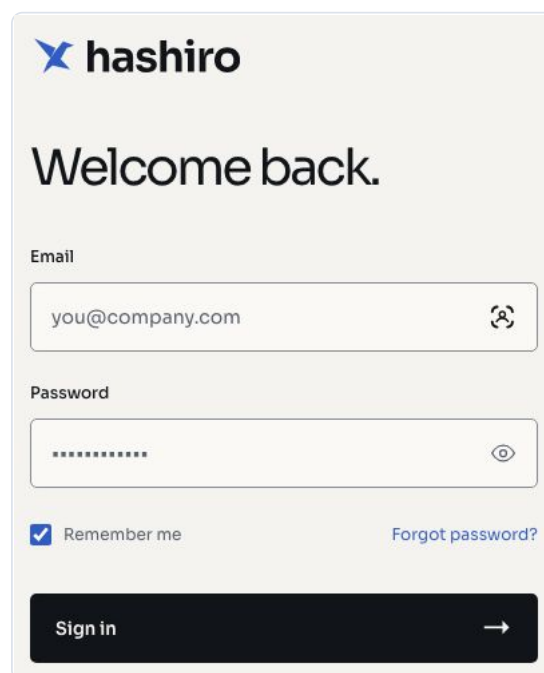
The image shows a Hashiro login page. At the top left is the Hashiro logo. Below it, the text "Welcome back." is displayed. There are two input fields: "Email" with the placeholder "you@company.com" and a toggle icon, and "Password" with a masked password "....." and a toggle icon. Below the password field is a checkbox labeled "Remember me" which is checked, and a link "Forgot password?". At the bottom is a dark "Sign in" button with a right-pointing arrow.

Figure 1 / Hashiro login page

2. Set Up MFA

Hashiro supports multi-factor authentication via TOTP (Time-based One-Time Password). We strongly recommend enabling it immediately.

1. Navigate to **My Account** from the user menu
2. Go to the **Security** section
3. Click **Enable MFA**
4. Scan the QR code with your authenticator app (Google Authenticator, Authy, 1Password, etc.)
5. Enter the verification code to confirm

Some organizations enforce mandatory MFA. If your organization has an OTP policy enabled, you'll be required to set up MFA on first login.

3. Explore the Dashboard

The **Dashboard** is your landing page with an at-a-glance view of your security posture:

- **Open severe risks** and **open vulnerabilities** across all projects
- **Attack Surface** summary: total assets and exposed services
- **Cloud Security** posture score and misconfiguration count
- **AI Agents**: active runs and total assessments
- **Vulnerability trends** over time
- **Needs attention**: recently discovered findings requiring action

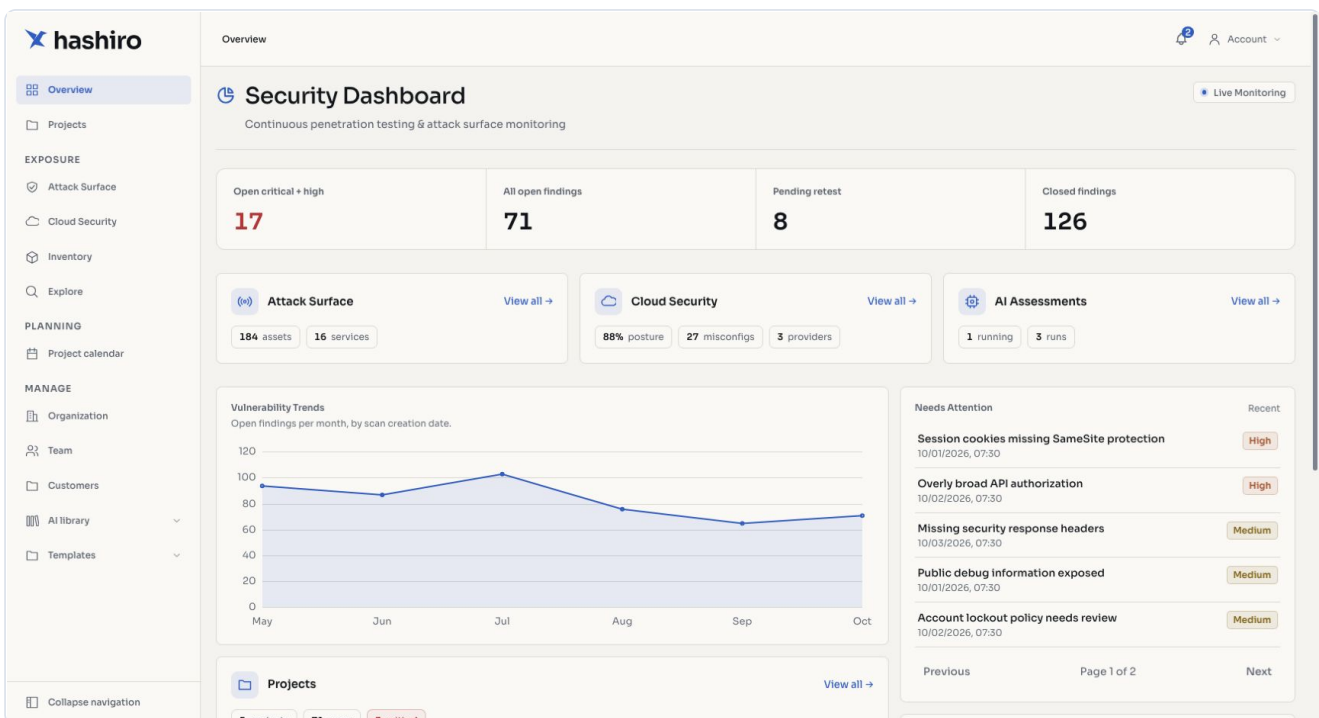


Figure 2 / Dashboard overview

4. Your First Attack Surface Scan

If your organization has EASM enabled:

1. Navigate to **Attack Surface** in the sidebar
2. Go to the **Config** tab
3. Review **in-scope** targets and add exclusions under **Out of scope**. Providers manage included targets by default.
4. Save the scope if your provider account has editing controls; otherwise request the change from your provider.

Without included scope, monitoring does not run. Once configured, Hashiro begins discovering assets, probing services, and identifying issues on your external attack surface automatically.

5. Create Your First Project

For manual security assessments:

1. Navigate to **Projects** in the sidebar
2. Click **New Project**
3. Fill in project details: name, type (black-box, grey-box, or white-box), category, dates
4. Define the **scope**: add in-scope and out-of-scope targets
5. Add team members and assign roles

From here, you can begin adding assets, documenting findings, and building your assessment narrative.

6. Run an AI Assessment

If your organization has AI assessment credits:

1. Open a project and go to the **AI** tab
2. Select an **Agent Type** (this defines the assessment pipeline)
3. Choose a **Run Mode**:
 - **Automatic**: the AI agent works autonomously
 - **Assisted**: you interact with the agent via chat
 - **Checklist**: the agent follows a predefined test list
4. Launch the run and monitor progress in real time via the streaming view

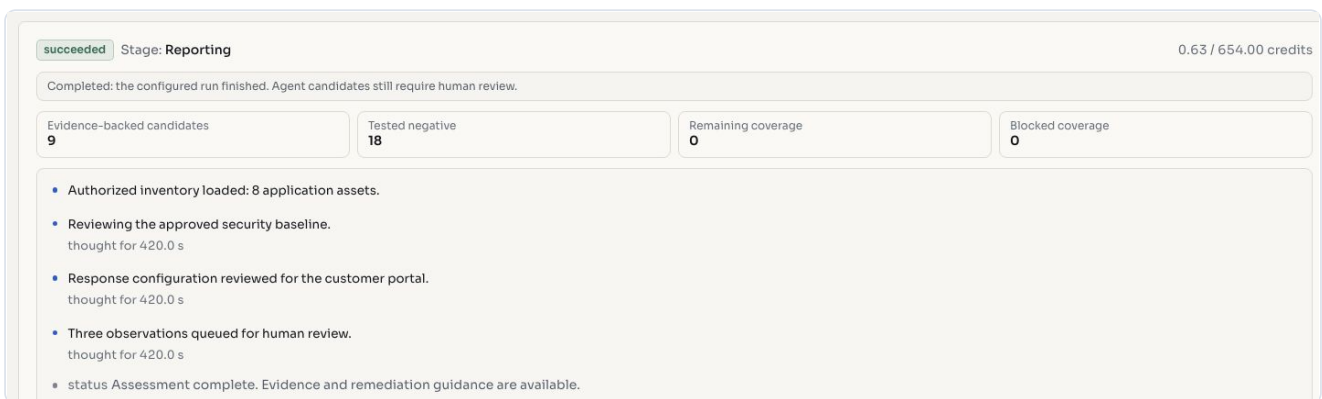


Figure 3 / AI assessment streaming view

What's Next

- [Dashboard](#): understand all dashboard metrics
- [Attack Surface](#): deep dive into EASM features
- [Projects](#): learn about project management
- [AI Assessments](#): configure and run AI-powered assessments
- [Team & Users](#): invite your team members

01.03

Provider guide

Providers deliver security services to linked client organizations. Your role determines the available actions; being a provider does not grant global administration.

Set up your team

Open **Organization** to review your license, name, branding, language, and security policy. Use **Team** to invite managers and pentesters. Managers administer the provider organization; pentesters work within assigned projects.

See [Organization](#) and [Roles](#).

Onboard a client

Use **Customers > New Customer** to provision the client and invite its initial management user. Review the provisional invitation status. Before the client claims the organization, eligible invitation-management controls are available; after claim, its credentials belong to the client.

See [Customer management](#).

Deliver an assessment

Create the client engagement, select the authorized client, define scope and exclusions, assign team members, and document findings. Review AI candidates before publication. Maintain the narrative and generate the report when the work is ready.

Clients can review published results of provider-managed engagements without editing your project configuration.

See [Projects](#), [Findings](#), and [Reports](#).

Surface monitoring

Providers manage in-scope targets by default when the controls are available. Review the configured targets with the client and keep exclusions current. An empty included scope means there are no targets being monitored.

See [Scope configuration](#) and [Surface findings](#).

Cloud security

Guide clients through connecting their own cloud accounts with the required read permissions. Do not assume a provider relationship grants access to every client's cloud connection or credential. Navigate only the account context available to you.

See [Cloud connections](#) and [Cloud findings](#).

API and Galileo

Use the same production API as clients: <https://api.hashiro.ai/api/>. Your token determines provider scope. Configure one Galileo profile and choose the project for each operation.

See [API](#) and [Galileo](#).

01.04

Client guide

Clients work in their own organization. A provider can deliver assessments into this workspace, but your account does not gain access to the provider's other clients.

First access

Accept your invitation, set your password, and configure MFA if required. Open **My Account** for profile, password, token, and notification preferences.

See [Quick start](#) and [Authentication](#).

Team and permissions

Your organization manager handles authorized team operations. Viewers read assigned results, while assessment contributors need the corresponding project permissions.

See [Team](#) and [Roles](#).

Review assessment results

Open **Projects**, choose the engagement, and inspect its narrative, assets, findings, and report. Provider-managed projects keep their configuration under provider control. Client-managed internal projects allow your authorized team to maintain its own assessment data.

Use the findings' evidence, remediation guidance, status, and history to coordinate fixes and retests. Public AI candidates can require review before becoming confirmed findings.

See [Issue lifecycle](#) and [AI review](#).

Attack surface

Open **Attack Surface** to inspect assets, services, and issues. Review the included scope with your provider. The **Out of scope** section remains available for exclusions, with editing determined by permissions. An empty included scope means monitoring is not running against targets.

See [Scope](#) and [Issue status changes](#).

Cloud access and findings

Connect your own AWS, Azure, or GCP account using the documented read-only credentials. Review account connection state separately from assessment results. Open a misconfiguration to inspect the affected resource and remediation, then use the status selector to track Open, Resolved, or Suppressed.

See [Connect a cloud account](#) and [Manage cloud findings](#).

Automation

Your API and Galileo connection uses `https://api.hashiro.ai/api/` with your own token. One profile is enough. Only operations permitted for your organization and project are available.

See [API](#), [Galileo configuration](#), and [Automatic Assessments](#).

01.05

Authentication

Hashiro supports multiple authentication methods: interactive sessions for the web console and API tokens for programmatic access.

Web Console Login

Email and Password

Sign in at the Hashiro console with your email address and password. Your session stays active while you use the console.

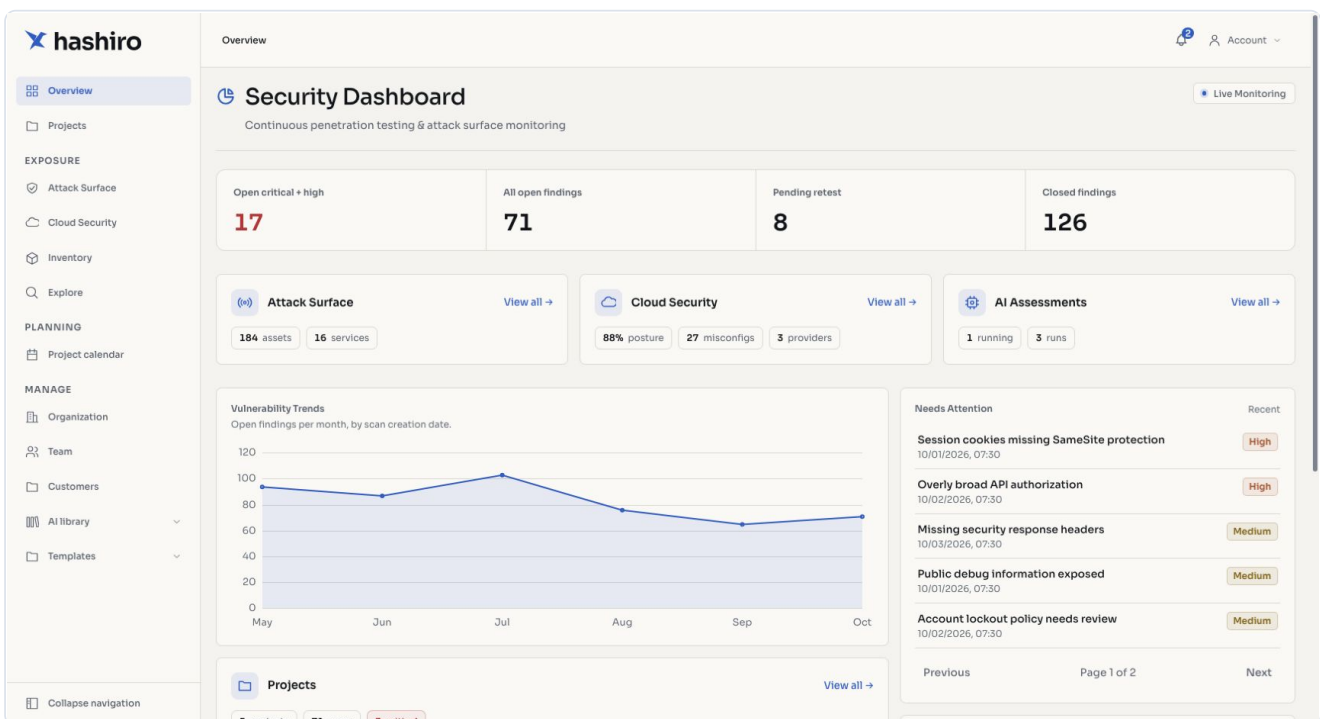


Figure 4 / Dashboard after signing in

Multi-Factor Authentication (MFA)

Hashiro supports **TOTP-based** multi-factor authentication (compatible with Google Authenticator, Authy, 1Password, and similar apps).

Setting up MFA:

1. Go to **My Account** from the user menu
2. Navigate to the security section
3. Click **Enable MFA**
4. Scan the displayed QR code with your authenticator app
5. Enter the 6-digit verification code to confirm setup

Once enabled, you'll be prompted for your TOTP code on every login after entering your password.

Organization-enforced MFA Administrators can enable an OTP policy at the organization level, requiring all members to set up MFA. Users in enforced-MFA organizations will be directed to the MFA setup flow on their next login.

Managing MFA:

- **Disable:** turn off MFA from My Account (if not enforced by organization policy)
- **Reset:** regenerate your MFA secret if you've lost access to your authenticator

Password Management

- **Change password:** update your password from My Account at any time
- **Password reset:** use the "Forgot password?" link on the login page to receive a reset email
- **First-login password change:** users provisioned by an administrator are required to set a new password on their first sign-in

Rate limiting Login attempts, OTP validation, and password reset requests are rate-limited to prevent brute-force attacks. After multiple failed attempts, you may need to wait before retrying.

API Tokens

For programmatic access (scripts, CI/CD pipelines, CLI tools, and integrations), use an API token instead of session cookies.

Generating a Token

1. Go to **My Account**
2. Click **Generate API Key**
3. Copy and securely store the token. It won't be shown again.

Using the Token

Include your API token in the X-HASHIRO-TOKEN header on every request:

```
curl -H "X-HASHIRO-TOKEN: your-token-here" \
https://api.hashiro.ai/api/projects
```

The token inherits the permissions of the user who generated it. All actions performed with the token are attributed to that user.

Galileo CLI Authentication

The [Galileo CLI](#) uses API tokens configured in its TOML profile system. See [CLI Configuration](#) for setup details.

Session Lifecycle

Event	Behavior
Login	Issues access_token (short-lived) and refresh_token (long-lived) as HTTP-only cookies
Token expiry	Automatically refreshed using the refresh token
Logout	Clears both cookies
MFA required	Redirects to OTP verification after password validation
First login	Redirects to password change before granting access

Security Considerations

- All authentication endpoints are served over HTTPS
- Passwords are hashed and never stored in plaintext
- API tokens are hashed at rest. The plaintext is only shown once at generation time
- Rate limiting is applied on all authentication endpoints
- Trusted proxy configuration ensures accurate IP-based rate limiting behind load balancers



PART 02

Platform handbook

Navigate the workspace, review findings, and follow assessments.

CHAPTERS / 39

02.01

Dashboard

The dashboard provides a unified, real-time overview of your organization's security posture. It consolidates metrics from attack surface, cloud security, vulnerability management, and AI assessments into a single view.

The dashboard auto-refreshes every 60 seconds and reloads when you return to the browser tab.

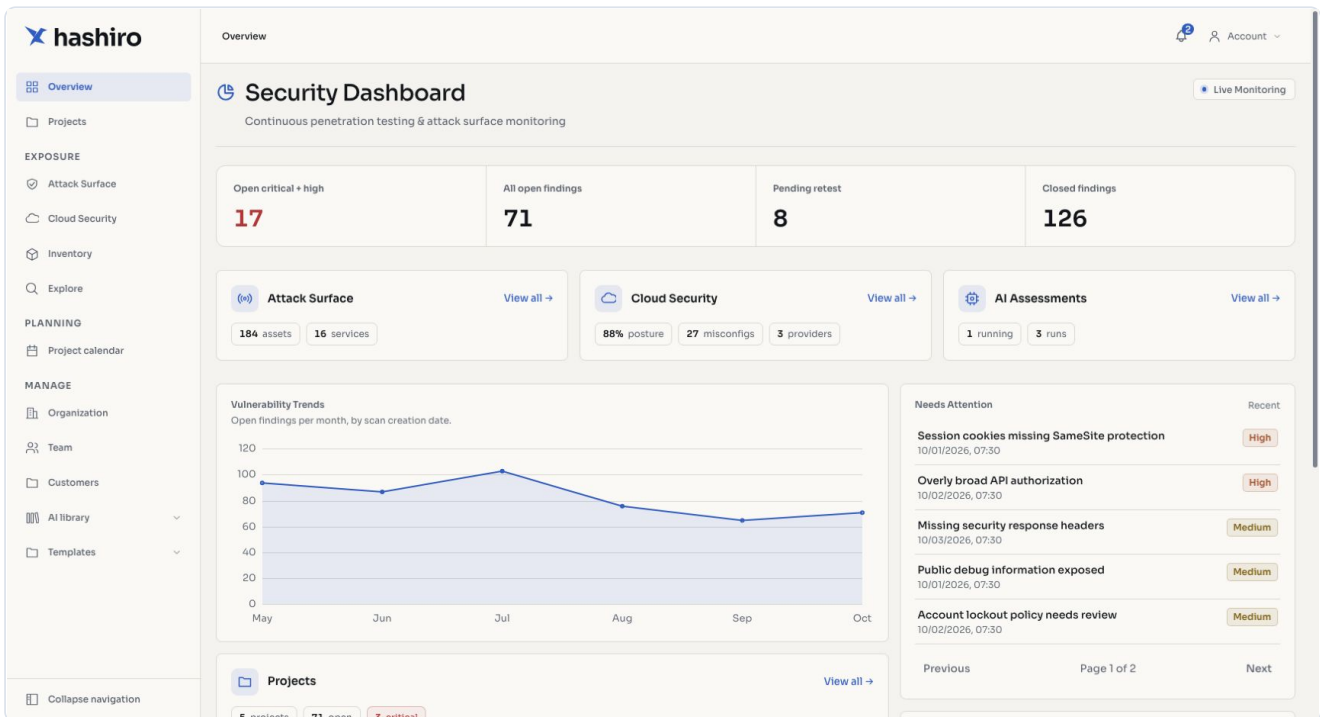


Figure 5 / Dashboard overview

Key Performance Indicators

Four top-level KPIs give you an at-a-glance summary:

KPI	Description
Open Severe Risks	Critical and high-severity vulnerabilities currently open across all projects
Open Vulnerabilities	Total count of all open findings regardless of severity
Pending Retest	Issues awaiting validation after a reported fix
Resolved Vulnerabilities	Findings confirmed as remediated

Environment Cards

Below the KPIs, three cards summarize each major area:

Attack Surface

- **Assets:** total discovered assets under EASM monitoring
- **Exposed Services:** services visible on the internet (open ports)
- **Open Risks:** unresolved issues detected on the attack surface

Cloud Security

- **Posture Score:** aggregate score across all connected cloud accounts
- **Misconfigurations:** total open misconfigurations detected
- **Providers:** number of connected cloud accounts (AWS, Azure, GCP)

AI Agents

- **Running:** assessments currently in progress
- **Total Runs:** cumulative assessment executions

Vulnerability Trends

A line chart showing vulnerability counts by month, broken down by severity level. Use this to track whether your security posture is improving over time.

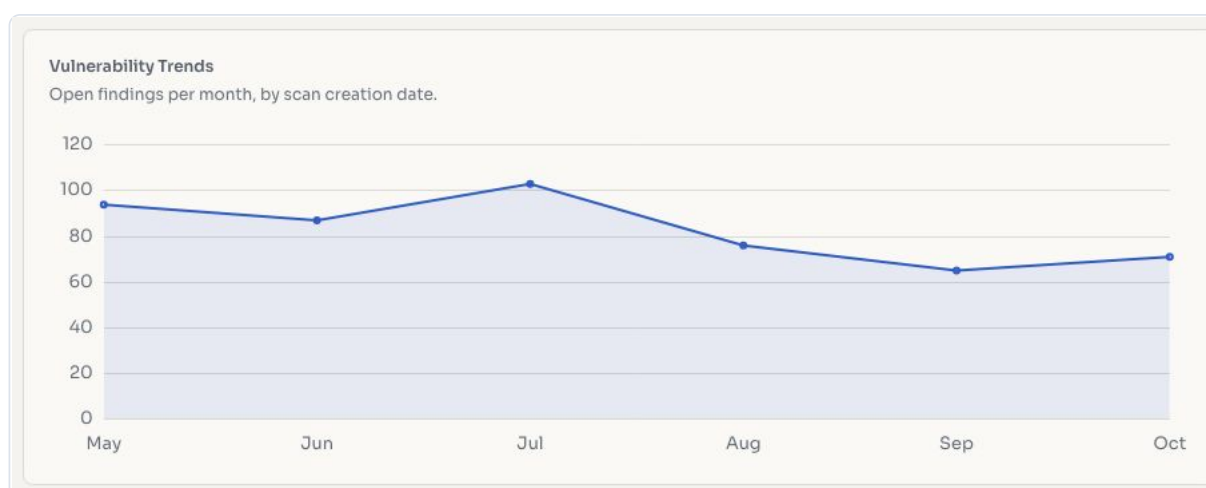


Figure 6 / Vulnerability trends chart

Top Projects

Lists your top projects ranked by vulnerability count with severity breakdown badges, making it easy to identify which engagements need the most attention.

Needs Attention

A paginated list of recently discovered vulnerabilities that require review. Highlights findings that are new or have been escalated in severity.

Open Issues by Severity

A bar chart breaking down all open issues by severity: Critical, High, Medium, Low, Informative, and Unknown.

Remediation Details

An expandable section with deeper remediation analytics:

- **Effort Matrix:** a priority-versus-complexity matrix that maps open vulnerabilities into quadrants, helping you prioritize by impact and effort.
- **Vulnerability Age Distribution:** shows how long open vulnerabilities have remained unresolved. Aging findings indicate remediation bottlenecks.
- **Attack Surface Stats:** summary of total assets, services, and the ratio of monitored versus unmonitored scope.

License Gating

Dashboard sections are gated by your organization's license tier. Sections for features not included in your plan show an upgrade prompt. Contact your provider or Hashiro sales for plan details.

02.02

External Attack Surface Management (EASM)

EASM gives you continuous visibility into your organization's internet-facing assets. Hashiro automatically discovers subdomains, services, and URLs, then probes and monitors them for changes, vulnerabilities, and misconfigurations.

EASM is a core pillar of Hashiro's **Continuous Threat Exposure Management (CTEM)** approach: rather than point-in-time assessments, your attack surface is monitored on an ongoing basis so new exposures are detected as soon as they appear.

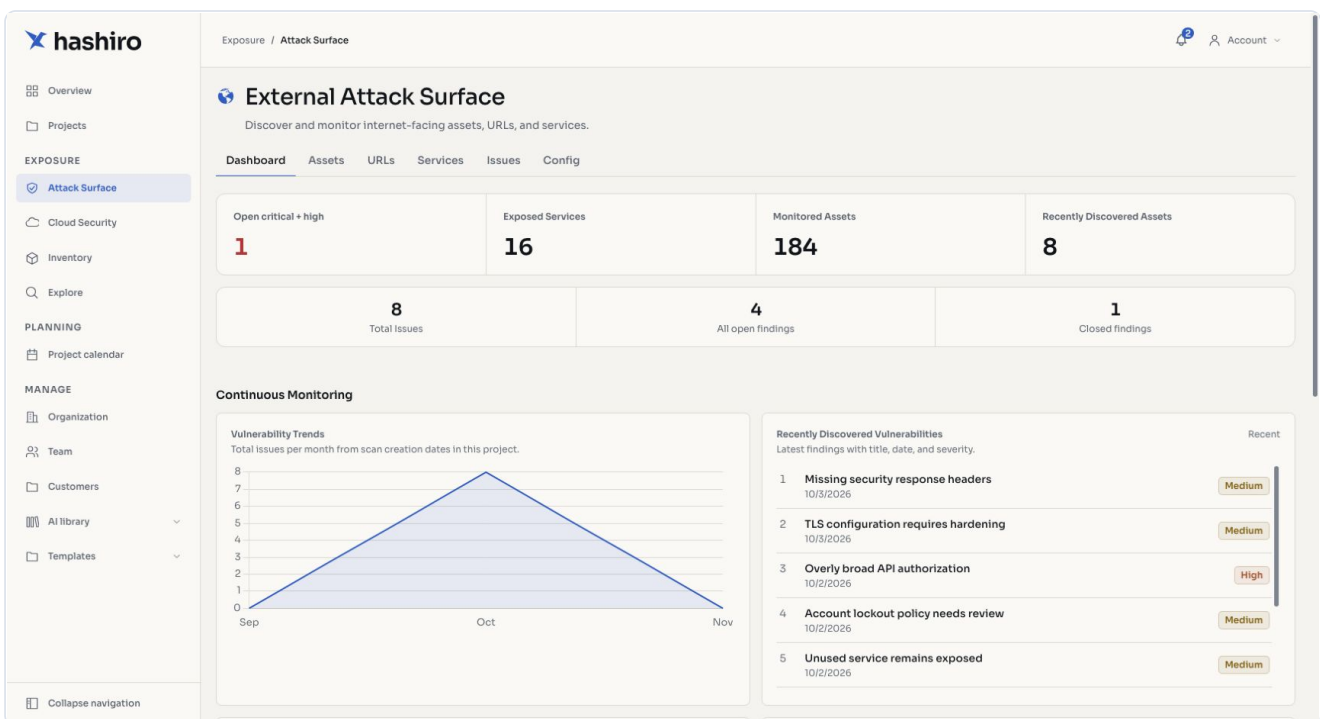


Figure 7 / EASM overview

How It Works

1. **Scope Definition:** you define in-scope and out-of-scope targets (domains, IPs, CIDRs, wildcards).
2. **Automated Discovery:** Hashiro enumerates subdomains, crawls for URLs, and scans for open services.
3. **Probing:** each discovered asset is probed with HTTPX to collect metadata (HTTP status, page title, technologies via Wappalyzer, screenshots, favicon hashes, web server identification, CDN detection, and more).
4. **Vulnerability Scanning:** Nuclei templates run against discovered assets to detect known vulnerabilities, misconfigurations, and exposures.
5. **Continuous Monitoring:** monitoring repeats according to the service configuration. New assets, removed assets, and status changes are tracked.

Module Tabs

The EASM module is organized into six tabs:

Tab	Purpose
Stats	Summary statistics for the entire attack surface
Assets	Paginated list of all discovered assets with probe data, screenshots, and technology details
URLs	Discovered URL paths with probe information
Services	Exposed network services detected via port scanning
Issues	Vulnerabilities and misconfigurations found on the attack surface
Config	Included scope and exclusions

Key Capabilities

- **Asset Discovery:** automated subdomain enumeration and asset identification
- **HTTP Probing:** rich metadata collection for every web-facing asset (title, headers, status codes, technologies, screenshots)
- **Port Scanning:** detection of exposed services across 27 standard ports
- **Vulnerability Detection:** Nuclei-based scanning for known CVEs, misconfigurations, and exposures
- **Screenshot Capture:** visual snapshots of every web asset for quick identification
- **Technology Detection:** Wappalyzer-powered fingerprinting of web technologies, frameworks, and libraries
- **CDN Detection:** identification of CDN providers fronting your assets
- **Change Tracking:** monitor asset availability, technology changes, and new exposures over time
- **Export:** download asset data as PDF or CSV

EASM availability and monitored-asset limits depend on your organization's provisioned service and license.

Related Features

- [Cloud Security](#): monitor cloud infrastructure alongside your external surface
- [Inventory](#): cross-project asset search across all your assets

Navigate the surface view

1. Open **Attack Surface** in the sidebar.

2. Use **Stats** for the overview, **Assets** for hosts, **URLs** for paths, and **Services** for ports.
3. Open a row to inspect its evidence and associated findings.
4. Use **Issues** to filter findings and **Config** to review scope.

The view is limited to the organization you are allowed to inspect. Empty results can mean no in-scope targets or no collected data yet. See [Scope configuration](#).

02.03

Assets

The Assets tab displays all hosts and subdomains discovered on your attack surface. Each asset is enriched with probe data collected automatically during EASM scans.

Asset	Title	Status Code	Content Length	Wappalyzer	Screenshot	Actions
admin.meridian.example.com	Meridian Admin	200	24720	React, Node.js		
api.meridian.example.com	Meridian Api	200	19320	React, Node.js		
auth.meridian.example.com	Meridian Auth	200	20220	Amazon CloudFront, Next.js		
cdn.meridian.example.com	Meridian Cdn	200	22020	Amazon CloudFront, Next.js		
docs.meridian.example.com	Meridian Docs	200	21120	React, Node.js		
partners.meridian.example.com	Meridian Partners	200	23820	Amazon CloudFront, Next.js		
portal.meridian.example.com	Meridian Portal	200	18420	Amazon CloudFront, Next.js		
status.meridian.example.com	Meridian Status	200	22920	React, Node.js		

Figure 8 / EASM assets list

Asset List

Assets are displayed in a paginated table:

Column	Description
Asset	The hostname, IP address, or domain
Type	Domain, IP, wildcard, or CIDR
Title	HTML page title extracted during probing
Status Code	HTTP response status code (200, 301, 403, etc.)
Technologies	Detected technologies via Wappalyzer (frameworks, CMS, libraries, languages)

Column	Description
Web Server	Web server software (nginx, Apache, Cloudflare, etc.)
CDN	CDN provider detected, if any
Availability	Whether the asset is currently reachable

Probe Data

Every web-facing asset is automatically probed using HTTPX. The probe collects:

- **HTTP status code** and response headers
- **Page title** extracted from the HTML
- **Content length** and body hash (for change detection)
- **Favicon hash** for identifying known applications
- **JARM hash** (TLS fingerprint for server identification)
- **Technologies** via Wappalyzer
- **Web server** identification from response headers
- **CDN detection**
- **IP addresses** resolved for the hostname
- **Word and line count** (basic content metrics)
- **Screenshots** via headless browser

Screenshots

Hashihiro captures a screenshot of every web asset during probing. Screenshots appear as thumbnails in the asset list and can be viewed full-size in the asset detail view. They help you quickly identify login pages, API endpoints, default server pages, or applications.

Filtering and Search

Filter assets by:

- **Search:** free-text search across asset names
- **Type:** domain, IP, wildcard, CIDR
- **Availability:** available or unavailable assets
- **Scope:** in-scope or out-of-scope

Actions

- **Copy to clipboard:** copy asset names for use in external tools

- **Delete:** remove assets from the inventory
- **Download:** export the asset list as PDF or CSV
- **Restart probe:** trigger a re-probe of selected assets to refresh metadata

Asset Detail

Click an asset to open its detail view:

- Full screenshot preview
- Complete HTTP response details (headers, status, content)
- Technology stack breakdown
- IP address resolution
- Associated URL paths
- Associated services (ports)
- Related issues found on this asset

The screenshot displays the 'Asset Detail' view for the asset `portal.meridian.example.com`. The interface includes a top navigation bar with 'Back to results', the asset name, and status buttons 'Available' and 'In Scope'. Below this is a tabbed interface with 'Overview', 'Services', and 'URLs'. The 'Overview' tab is active, showing a 'Screenshot' of the website, 'Asset Information', and 'HTTP Response' details. The screenshot shows a 'Welcome back, Alex' message, a 'Recent orders' table, and a 'Your workspace' section. The 'Asset Information' section lists the asset name, IP addresses, and tags. The 'HTTP Response' section shows the status code (200), content length (18.0 KB), page title (Meridian Portal), and web server (CloudFront). Below the screenshot, the 'Technologies Detected' section lists 'Amazon CloudFront' and 'Next.js'. The 'HTTP Headers' section shows 'Content-Type: text/html; charset=utf-8' and 'Strict-Transport-Security: max-age=31536000'.

Back to results

portal.meridian.example.com
meridian-portal

Available In Scope

Overview Services URLs

Screenshot Open observed host in a new tab ↗

meridian

Welcome back, Alex
Here is what is happening in your workspace today.

Active subscriptions
24
All services operational

Orders this month
148
12% above last month

Next billing date
Oct 15
Your payment method is up to date

Recent orders

Order	Created	Amount	Status
Workspace renewal	Oct 5, 2028	\$380.00	Completed
Team seats	Oct 1, 2028	\$100.00	Completed
Storage add-on	Sep 26, 2028	\$40.00	Completed

Your workspace
Meridian Commerce
Business plan
24 active members
Manage workspace
View account settings

Asset Information

ASSET:
portal.meridian.example.com

IP ADDRESSES:
198.51.100.20

TAGS:
environment: production
owner: Platform Engineering

HTTP Response

Status Code 200

Content Length 18.0 KB

Page Title Meridian Portal

Web Server CloudFront

Technologies Detected

Amazon CloudFront Next.js

HTTP Headers 2 of 2 headers

Content-Type text/html; charset=utf-8

Strict-Transport-Security max-age=31536000

Figure 9 / Asset detail view

Assets are automatically re-probed on each scan cycle. You can also manually trigger a re-probe for specific assets to get fresh data immediately.

02.04

Services

The Services tab shows all network services discovered through port scanning on your attack surface. It gives you visibility into which ports are open, what services are running, and where your organization may have unexpected exposure.

Asset ↑	Port	Protocol	Service	State
☐ admin.meridian.example.com	443	HTTPS	HTTPS	open
☐ admin.meridian.example.com	80	HTTP	HTTP	open
☐ api.meridian.example.com	443	HTTPS	HTTPS	open
☐ api.meridian.example.com	80	HTTP	HTTP	open
☐ auth.meridian.example.com	443	HTTPS	HTTPS	open
☐ auth.meridian.example.com	80	HTTP	HTTP	open
☐ cdn.meridian.example.com	443	HTTPS	HTTPS	open
☐ cdn.meridian.example.com	80	HTTP	HTTP	open
☐ docs.meridian.example.com	443	HTTPS	HTTPS	open
☐ docs.meridian.example.com	80	HTTP	HTTP	open

Figure 10 / EASM services list

Service Discovery

Hashiro scans 27 standard ports across your in-scope assets:

Port Range	Common Services
21	FTP
22	SSH
23	Telnet
25	SMTP
53	DNS

Port Range	Common Services
80, 443	HTTP / HTTPS
110, 143, 993, 995	Email (POP3, IMAP)
3306	MySQL
3389	RDP
5432	PostgreSQL
6379	Redis
8080, 8443	Alternative HTTP / HTTPS
27017	MongoDB

Open database ports (MySQL, PostgreSQL, Redis, MongoDB) and remote access services (RDP, Telnet) exposed to the internet are flagged as high-risk. Review these immediately and restrict to internal networks or VPN access.

Service Table

Column	Description
Asset	The hostname or IP where the service was found
Port	The port number
Protocol	TCP or UDP
State	open, filtered, or closed
Service	Identified service name (e.g., http, ssh, mysql)

Port States

- **Open:** the port is accepting connections and a service is running
- **Filtered:** the port is blocked by a firewall; its state cannot be determined
- **Closed:** the port is reachable but no service is listening

Network Exposure Analysis

Use this view to answer critical questions:

- Are any database services exposed to the internet?
- Are administrative interfaces (SSH, RDP) publicly accessible?
- Are there unexpected or legacy services running?
- Are applications running on non-standard ports?

Actions

- **Filter by state:** show only open, filtered, or closed ports
- **Filter by port:** search for specific port numbers
- **Navigate to asset:** click through to the full asset detail view

02.05

Issues

The Issues tab displays all vulnerabilities and misconfigurations detected on your attack surface during EASM scans. These findings are generated automatically through Nuclei template scanning and other detection mechanisms.

External Attack Surface

Discover and monitor internet-facing assets, URLs, and services.

Dashboard

Assets

URLs

Services

Issues

Config

Search...

All severities

☐	Severity ↑	Title	Asset	Status	Actions
☐	High	Session cookies missing SameSite protection	portal.meridian.example.com	Open	
☐	High	Overly broad API authorization	api.meridian.example.com	Draft	
☐	Medium	Missing security response headers	auth.meridian.example.com	Open	
☐	Medium	Public debug information exposed	docs.meridian.example.com	Pending Retest	
☐	Medium	Account lockout policy needs review	cdn.meridian.example.com	Risk Accepted	
☐	Medium	TLS configuration requires hardening	status.meridian.example.com	Closed	
☐	Medium	Sensitive responses cached by browser	partners.meridian.example.com	Open	
☐	Medium	Unused service remains exposed	admin.meridian.example.com	Open	

Showing 1-8

Columns 8 results

Filters

Figure 11 / EASM issues list

Issue List

Issues are presented in a table with severity badges for quick triage:

Column	Description
Title	Name of the vulnerability or misconfiguration
Asset	The affected hostname or IP
Severity	Critical, High, Medium, Low, or Informative
Status	Current issue status (open, closed, etc.)
Detected	When the issue was first discovered

Severity Levels

- **Critical:** exploitable vulnerabilities with severe impact (e.g., RCE, authentication bypass)
- **High:** significant vulnerabilities that could lead to data exposure or system compromise
- **Medium:** moderate-risk findings for normal remediation cycles
- **Low:** minor issues with limited direct impact
- **Informative:** observations that don't represent a direct risk but may be useful for hardening

Automated Detection

EASM issues are discovered through:

- **Nuclei Templates:** community and custom templates that detect known CVEs, misconfigurations, default credentials, exposed panels, and more
- **Configuration Checks:** missing security headers, insecure TLS configurations, open redirects
- **Exposure Detection:** sensitive files, backup files, directory listings, debug endpoints

Bulk Operations

Select multiple issues and perform bulk actions:

- **Change Status:** move selected issues to open, closed, accepted, or not applicable

Issue Detail

Clicking an issue opens its full detail view within the [Vulnerability Management](#) system, where you can:

- Review the full finding description and evidence
- Assign a CVSS score
- Track remediation through the issue lifecycle
- Add retest notes
- View history and audit trail

EASM issues feed into the same vulnerability management workflow as manually reported findings. They share the same lifecycle, statuses, and reporting capabilities. The only difference is they were detected automatically.

Related Features

- [Vulnerability Management](#): full issue lifecycle and remediation tracking
- [Nuclei Templates](#): customize which checks run against your attack surface
- [EASM Configuration](#): review included scope and exclusions

Change issue status

Your organization can track remediation of its surface findings using the status controls available to its account.

1. Open **Attack Surface > Issues**.
2. Filter or search for the findings you want to review.
3. Open a finding to inspect its description, affected asset, evidence, and history.
4. To change several findings, select their rows and choose **Change status**.
5. Select the status and confirm. Check that the table shows the updated state.

Typical choices include **Open**, **Closed**, **Invalid**, **Out of scope**, **Risk accepted**, **Review**, and **Pending retest**. Draft is available to eligible provider accounts. Use Pending retest when a fix needs verification and Closed after correction is confirmed. Risk accepted records a conscious decision to retain a risk.

A status change records the workflow decision; it does not fix the underlying system. If you cannot edit, check your organization and role with its manager. AI candidate decisions can require the separate review workflow.

02.06

Scope configuration

Open **Attack Surface > Config** to inspect the monitored targets and exclusions for your organization.

Scope and context

ScopeOut of Scope

In Scope

Add Line

Asset	Asset Type	Actions
*.meridian.example.com	domain	 
api.meridian.example.com	domain	 
198.51.100.0/24	cdir	 

Copy to clipboard

Figure 12 / Attack surface configuration

In-scope targets

By default, **providers manage in-scope targets**. If your account has the in-scope controls, add, edit, or remove the targets authorized for monitoring. Clients can review the configured scope and request changes from their provider when editing is unavailable.

Type	Example
Domain	example.com
Wildcard	*.example.com
IP address	203.0.113.10
CIDR range	203.0.113.0/24

1. Open the **In scope** section.
2. Add a row and select the target type.
3. Enter the domain, wildcard, IP, or range.

4. Save and review the resulting list.

Only include targets your organization is authorized to monitor. Editing controls depend on your account permissions and how the service is provisioned.

No scope means no monitoring

An empty in-scope list means the surface monitor is **not running against any targets**. An exclusion alone does not start monitoring. If you see no configured scope, ask the provider to define the monitored targets.

Out-of-scope targets

The **Out of scope** section is always available for defining exclusions in the organization's surface view. Write controls still depend on your permissions.

1. Open **Config > Out of scope**.
2. Add the target to exclude and select its type.
3. Save the entry.
4. Confirm it appears in the exclusions list.

Exclusions take precedence over included scope. Use them for third-party systems, sensitive targets, or environments that must not be checked. Edit or remove an exclusion only when the monitoring authorization changes.

Review results

Scope changes do not instantly replace existing results. Use [Assets](#), [Services](#), and [Issues](#) to inspect collected data and manage findings. Historical records can remain visible after an exclusion is added.

02.07

Cloud Security

Hashiro's Cloud Security module provides **Cloud Security Posture Management (CSPM)**: continuous monitoring of your cloud infrastructure for misconfigurations, compliance gaps, and security risks.

Connect your AWS, Azure, or GCP accounts and Hashiro will scan your cloud resources, detect misconfigurations, and track your compliance posture over time.

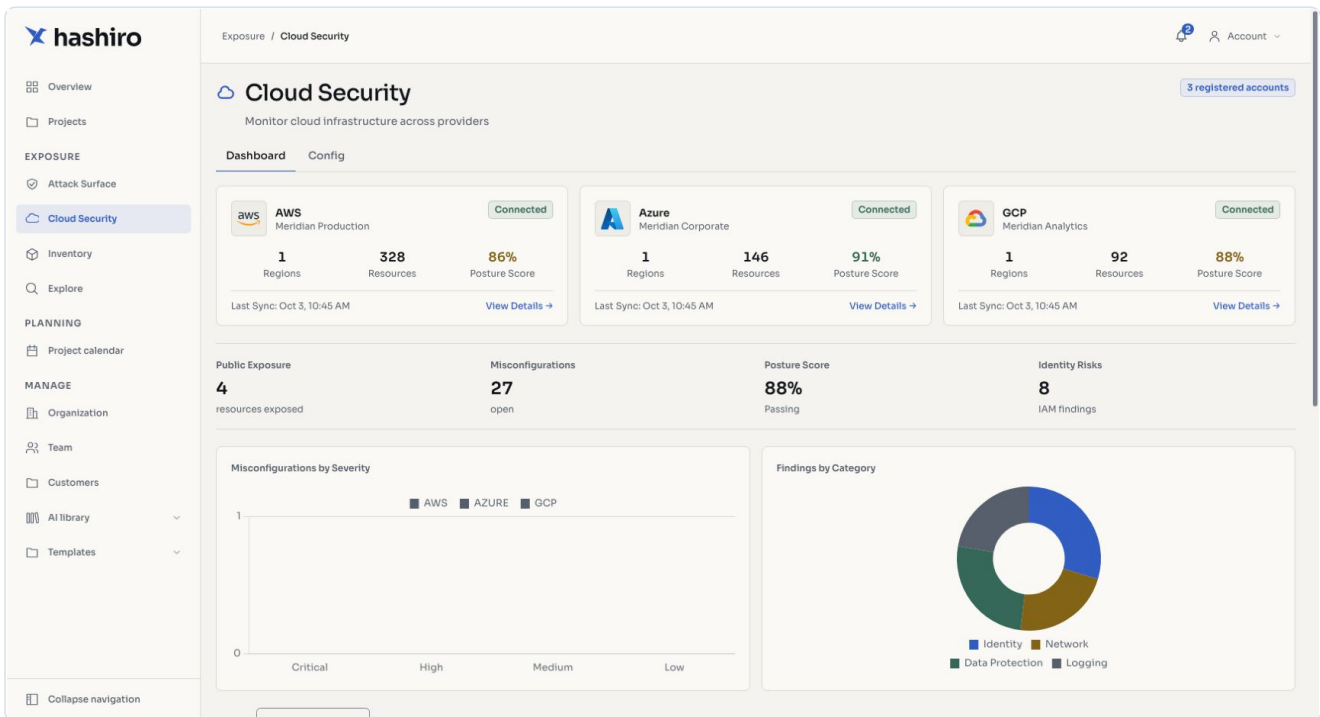


Figure 13 / Cloud security dashboard

Multi-Cloud Support

Provider	Connection Method
AWS	Access key credentials in the current connection form
Azure	Service principal credentials
GCP	Service account key

You can connect multiple accounts across different providers to get a unified view of your entire cloud estate.

Cloud Dashboard

The cloud dashboard provides an at-a-glance view of your posture:

- **Posture Score:** aggregate security score across all connected accounts
- **Resource Counts:** total cloud resources discovered and monitored
- **Misconfigurations:** open misconfigurations broken down by severity
- **Public Exposure:** resources with public-facing access

Charts and Visualizations

- **Misconfigurations by Severity:** bar chart showing the distribution across critical, high, medium, and low
- **Findings by Category:** doughnut chart breaking down findings into:
 - **Identity & Access Management:** IAM policies, role permissions, MFA enforcement
 - **Network:** security groups, NACLs, public endpoints, VPC configuration
 - **Data Protection:** encryption at rest, encryption in transit, bucket policies
 - **Logging & Monitoring:** CloudTrail, flow logs, audit logging, alerting

Key Capabilities

- **Automated Scanning:** periodic scans of all connected cloud accounts
- **Misconfiguration Detection:** identify security gaps against best practices and benchmarks
- **Compliance Tracking:** map findings to CIS benchmarks and track progress
- **Scan Reconciliation:** automatically resolve findings that are no longer detected
- **Multi-Account:** unified view across multiple accounts and providers
- **Credential Security:** all cloud credentials are encrypted at rest

Getting Started

Use [Connect a cloud account](#) to configure AWS, Azure, or GCP access in the current connection form.

Cloud Security is available on business plans and above. The number of supported provider accounts depends on your license tier.

Related Features

- [Attack Surface \(EASM\)](#): monitor external-facing assets alongside your cloud posture
- [Dashboard](#): see cloud security metrics on the main dashboard

Navigate cloud results

1. Open **Cloud Security**.
2. Review the connected-account cards and latest assessment information.
3. Open an account to focus on its resources and findings.

4. Use the misconfiguration section to filter by severity and status.
5. Open a row to inspect the affected resource, evidence, and remediation.

Connected, **Disconnected**, and **Error** describe the account connection. **Never assessed** or no score means assessment results are not yet available; it is not a clean security result. Use [Connect a cloud account](#) for setup and [Misconfigurations](#) for status changes.

02.08

Connect a cloud account

Connect your own AWS, Azure, or GCP account to let Hashiro inspect its configuration. You need permission to manage cloud connections in your organization and permission to create a read-only scanning identity in the cloud account.

Open the connection form

1. Open **Cloud Security** in the sidebar.
2. Use the account connection action to open the form.
3. Select AWS, Azure, or GCP.
4. Fill in the account identifiers and credentials below.
5. Click **Connect** and check the result.

Add Account [X]

Provider

aws AWS Azure GCP

This will replace the existing "Meridian Production" account. Only one account per provider is supported.

Account Name

Meridian Staging

AWS Account ID

123456789012

Access Key ID

AKIAIOSFODNN7EXAMPLE

Secret Access Key

wJaLrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY

Regions

us-east-1, us-west-2

Comma-separated list of AWS regions to scan

Cancel Connect

Figure 14 / Cloud account connection

A successful connection does not mean a security assessment has completed. Check the latest assessment time and results separately.

AWS

The current connection form accepts:

Field	Value
Account ID	The 12-digit AWS account identifier
Access Key ID	Access key for the scanning identity
Secret Access Key	Secret paired with that key
Regions	Comma-separated regions to assess, when specified

Create a dedicated IAM identity for read-only security inspection. AWS provides a [SecurityAudit managed policy](#) for configuration inspection. The exact required read permissions depend on the services assessed; ask your cloud administrator to review the scanner's requirements rather than granting unrestricted write access.

Generate an access key, copy its ID and secret into the form, and keep the secret in your organization's secret-management system. The visible form does not provide a role-ARN setup wizard. Role-based scanning configuration, when used, is handled through the appropriate administrative setup.

Azure

Create an application and service principal in Microsoft Entra ID. Assign the appropriate read access on the subscription, such as the Azure **Reader** role for resource inspection, with additional security read permissions only when needed. See [Azure built-in roles](#).

The form requires:

Field	Value
Tenant ID	Directory identifier
Subscription ID	Subscription being assessed
Client ID (App ID)	Registered application's identifier
Client Secret	Secret value, not the secret's identifier

Record the expiry date and rotate the secret before it expires. Use the subscription where the service principal has been granted access.

GCP

Create a dedicated service account in the project to assess and grant the read permissions required for resource and security configuration inspection. Review [Google Cloud IAM roles](#) for the relevant services.

The form requires **Project ID** and the **service-account JSON key**. Paste the complete JSON object. If your organization prohibits service-account key creation, coordinate an approved setup with your cloud administrator. Follow [Google's key-management guidance](#).

Validate and manage the connection

From the connected-account list, open the account's action menu and choose **Revalidate** after rotating credentials or changing permissions. Check the resulting connection state and validation timestamp. The menu can also remove a connection where your permissions allow it.

Troubleshooting

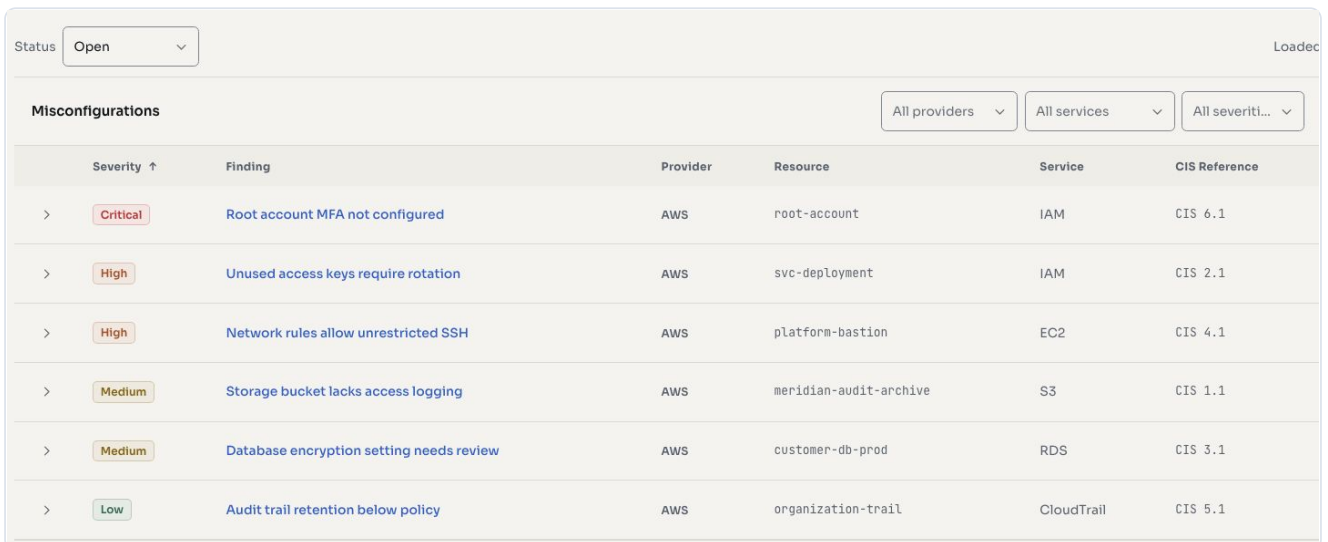
Symptom	What to check
Invalid credentials	Correct account identifiers, key/secret pairing, and expiry
Access denied	Read permissions on the exact account, subscription, or project
No assessment results	A connection can exist before the first completed assessment
Revalidation fails after rotation	Update the stored connection credentials through the available management workflow
Cloud action unavailable	Organization license and cloud-management permissions

Credentials are encrypted at rest and are not returned in ordinary user-facing connection details. Do not place them in issue descriptions, public links, or project reports.

02.09

Misconfigurations

Hashiro detects security misconfigurations across your cloud resources by scanning against established security benchmarks and best practices. Each misconfiguration represents a gap in your cloud security posture that could be exploited or that violates compliance requirements.



Status	Open					Loaded
Misconfigurations			All providers	All services	All severiti...	
Severity ↑	Finding	Provider	Resource	Service	CIS Reference	
> Critical	Root account MFA not configured	AWS	root-account	IAM	CIS 6.1	
> High	Unused access keys require rotation	AWS	svc-deployment	IAM	CIS 2.1	
> High	Network rules allow unrestricted SSH	AWS	platform-bastion	EC2	CIS 4.1	
> Medium	Storage bucket lacks access logging	AWS	meridian-audit-archive	S3	CIS 1.1	
> Medium	Database encryption setting needs review	AWS	customer-db-prod	RDS	CIS 3.1	
> Low	Audit trail retention below policy	AWS	organization-trail	CloudTrail	CIS 5.1	

Figure 15 / Cloud misconfigurations table

Misconfiguration Table

Misconfigurations are displayed in a filterable, sortable table:

Column	Description
Title	Description of the misconfiguration
Provider	AWS, Azure, or GCP
Severity	Critical, High, Medium, or Low
Category	IAM, Network, Data Protection, or Logging
Status	Open, Resolved, or Suppressed
CIS Reference	Associated CIS benchmark control
First Detected	When the misconfiguration was first found

Severity Levels

Severity	Description
Critical	Immediate risk of data exposure or unauthorized access (e.g., public S3 buckets with sensitive data, root account without MFA)
High	Significant security gap to address promptly (e.g., overly permissive security groups, unencrypted databases)
Medium	Moderate risk for regular remediation cycles (e.g., missing logging, unused credentials)
Low	Minor hardening recommendations (e.g., tagging compliance, non-critical best practices)

Categories

Identity & Access Management (IAM)

- Overly permissive IAM policies
- Root account usage without MFA
- Unused or stale access keys
- Missing password policies
- Cross-account access misconfigurations

Network

- Overly permissive security groups (0.0.0.0/0 ingress)
- Public-facing resources that should be private
- Missing network segmentation
- Unencrypted traffic between services
- Missing VPC flow logs

Data Protection

- Unencrypted storage (S3, EBS, RDS)
- Public bucket policies
- Missing encryption in transit
- Insecure key management
- Missing backup configurations

Logging & Monitoring

- Disabled CloudTrail or equivalent audit logging
- Missing flow logs

- No alerting on security events
- Insufficient log retention
- Missing access logging on storage buckets

Status Tracking

- **Open:** the misconfiguration is active and has not been addressed
- **Resolved:** the misconfiguration has been remediated (either manually confirmed or auto-resolved by scan reconciliation)
- **Suppressed:** the misconfiguration has been acknowledged but intentionally not remediated (e.g., accepted risk)

Scan Reconciliation

When Hashiro runs a new scan, it compares current findings against previous results:

- **New findings** are created with `Open` status
- **Findings no longer detected** are automatically moved to `Resolved` (the underlying resource was fixed or removed)
- **Persistent findings** remain in their current status

This reconciliation ensures your misconfiguration list always reflects the current state of your cloud infrastructure.

Misconfiguration Detail

Click a misconfiguration to see:

- **Full description** of the security issue
- **Affected resource** (ARN, resource ID)
- **CIS benchmark reference**
- **Remediation guidance** with step-by-step instructions
- **Risk context:** why this misconfiguration matters and what an attacker could do

Filtering

Filter misconfigurations by:

- **Severity:** Critical, High, Medium, Low
- **Category:** IAM, Network, Data Protection, Logging
- **Provider:** AWS, Azure, GCP
- **Status:** Open, Resolved, Suppressed

Start with Critical IAM and Network findings. These represent the highest risk to your environment.

Change a finding's status

1. Open **Cloud Security** and find the misconfiguration table.
2. Filter by the status or severity you want to review.
3. Open the finding's title to view its detail page.
4. Read the affected resource and remediation guidance.
5. Use the status selector near the top of the page to choose **Open**, **Resolved**, or **Suppressed**.
6. Check the confirmation message and updated status. If saving fails, the selector returns to the previous value and displays an error.

Use Resolved after remediation is confirmed. Use Suppressed for a deliberately accepted or excluded finding. Changing this label does not modify the cloud resource. The next assessment may reconcile the finding again according to the scanner's rules.

Return to the table and adjust its status filter if the finding disappears from the current view after the change. Editing requires the organization's cloud permissions; viewing a finding alone does not grant status-change access.

02.10

Compliance

Hashiro maps cloud misconfigurations to industry-standard compliance benchmarks, giving you visibility into how your cloud infrastructure aligns with established security frameworks.

CIS Benchmarks

The primary compliance framework used by Hashiro is the **CIS (Center for Internet Security) Benchmarks**: widely recognized best-practice guidelines for securing cloud environments.

Each cloud provider has its own CIS benchmark:

Provider	Benchmark
AWS	CIS Amazon Web Services Foundations Benchmark
Azure	CIS Microsoft Azure Foundations Benchmark
GCP	CIS Google Cloud Platform Foundations Benchmark

Compliance Score

Each connected provider account displays a **compliance score**: the percentage of applicable CIS controls that are passing.

$$\text{Compliance Score} = (\text{Passing Controls} / \text{Total Applicable Controls}) \times 100$$

The score updates automatically after each scan as misconfigurations are detected or resolved.

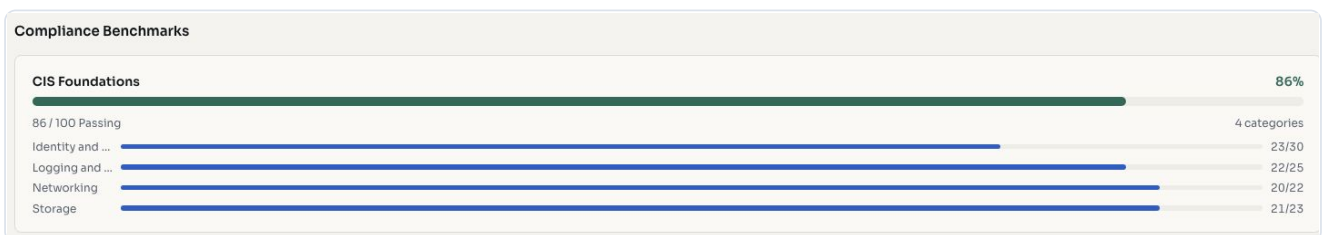


Figure 16 / Compliance score and benchmarks

Benchmark Progress

Compliance benchmarks are displayed as progress bars showing:

- **Passing:** controls where no misconfigurations were found
- **Failing:** controls with one or more active misconfigurations
- **Not Applicable:** controls that don't apply to your environment (e.g., a control about a service you don't use)

Control Mapping

Every misconfiguration in Hashiro is mapped to its corresponding CIS benchmark control. This means you can:

- **Drill down from a benchmark control** to see all related misconfigurations
- **Drill up from a misconfiguration** to see which compliance controls it affects
- **Track remediation progress** at the control level (a control passes only when all related misconfigurations are resolved)

Using Compliance Data

Audit Preparation

Export compliance reports to demonstrate your organization's security posture to auditors. The compliance view shows which controls are met and which have gaps, along with the specific resources involved.

Prioritization

Use compliance alignment to prioritize remediation. Misconfigurations that affect multiple compliance controls or that are required for your regulatory obligations should be addressed first.

Trend Tracking

Monitor your compliance score over time to ensure your cloud security posture is improving and that new deployments don't introduce regressions.

Compliance scores are a useful directional indicator, but they don't replace a thorough security assessment. A high compliance score means you're following established best practices. It doesn't guarantee the absence of all security risks.

02.11

Projects

Projects are the core unit for managing security assessments in Hashiro. Each project represents a testing engagement (penetration test, bug bounty program, or continuous assessment) with its own scope, findings, team, and deliverables.

The screenshot shows the Hashiro Projects portfolio view. The sidebar on the left contains navigation links: Overview, Projects (selected), Exposure (Attack Surface, Cloud Security, Inventory, Explore), Planning (Project calendar), and Manage (Organization, Team, Customers, AI library, Templates). The main content area is titled 'Projects' and includes a search bar, filters for Organization and Status, and a 'Project view' toggle (List/Cards). A table lists 5 projects with their status, organization, and vulnerability counts.

PROJECT	STATUS	ORGANIZATION	ASSETS	URLS	FINDINGS
Cloud Infrastructure	Completed	Atlas Financial	32	218	12
Customer Portal	In Progress	Meridian Commerce	18	142	16
Mobile Backend	Review	Harbor Health	46	294	8
Partner Workspace	Scheduled	Meridian Commerce	39	256	10
Payments API	In Progress	Harbor Health	25	180	14

Figure 17 / Projects portfolio view

Portfolio View

The portfolio gives you a centralized view of all engagements. Switch between **list** and **card** layouts to suit your preference.

- **Search:** filter projects by name
- **Status filter:** narrow by project status (active, completed, archived)
- **Client filter:** providers can filter by client organization
- **Pagination:** navigate large portfolios

Each project card shows vulnerability counts by severity, project dates, assigned members, and current status.

Project Types

Type	Description
Black-box	No prior knowledge of the target. Simulates an external attacker.
Grey-box	Partial knowledge with some credentials or documentation provided.
White-box	Full access to source code, architecture docs, and credentials.

Provider Feature Providers can define **custom project type definitions** beyond the standard categories to match their service offerings.

Project Tabs

Each project is organized into tabs:

Tab	Purpose
Details	Scope definition, credentials, context, and project configuration
Narrative	Markdown documentation of the assessment methodology and progress
Assets	Discovered hosts and subdomains within the project scope
URLs	Enumerated URL paths on discovered assets
Services	Network services detected on assets (ports, protocols)
Issues	Vulnerability findings with severity, CVSS scores, and status tracking
AI	AI-powered automated assessment integration

The screenshot displays the Hashiro Customer Portal interface. The left sidebar contains navigation menus for Overview, Projects, EXPOSURE (Attack Surface, Cloud Security, Inventory, Explore), PLANNING (Project calendar), and MANAGE (Organization, Team, Customers, AI library, Templates). The main content area is titled 'Customer Portal' and includes a 'Download' button. Below this, there are tabs for Details, Narrative, Assets, URLs, Services, Insights, Issues, Chains, AI Pentest, and Stats. The 'Details' tab is active, showing 'Scope and context' with 'Scope' and 'Out of Scope' sub-tabs. The 'In Scope' section contains a table of assets:

Asset	Asset Type	Actions
*.meridian.example.com	domain	[Edit] [Delete]
api.meridian.example.com	domain	[Edit] [Delete]
198.51.100.0/24	cdir	[Edit] [Delete]

Below the table is a 'Copy to clipboard' button and an 'Add Line' button. To the right, the 'Pentest Details' section shows the status as 'In Progress', code as 'meridian-portal', start date as 'Oct 1, 2026', end date as 'Oct 8, 2026', and a 'Schedule elapsed' progress bar at 34%. The 'Parameters' section lists Category as 'Web Application', Type as 'Black Box', Language as 'English', and Report template as '—'. The 'Members' section shows 'Viewers' as 'No viewers' and 'Pentesters' as 'No pentesters'. The 'Tags' section includes 'production' and 'Security Engineering'.

Figure 18 / Project detail tabs

Member Permissions

Projects support per-member permission assignment. Team members can be given specific access levels within a project, independent of their organization-wide role.

Client vs. Provider Management

- **Provider-managed projects:** created and managed by the security provider with full control over scope, findings, and reports. Clients get read access and can download reports.
- **Client-managed projects:** created by client organizations for internal assessments. Providers may be granted access to collaborate.

Project Deletion

Deleting a project removes all associated data: assets, URLs, services, findings, insights, AI assessment runs, connectors, and notifications.

Project deletion is irreversible. All associated data will be permanently removed.

Scope Management

Scope defines the boundaries of a security assessment: what is authorized to test and what is off-limits. A well-defined scope is the foundation for asset discovery, vulnerability scanning, and manual testing.

Scope and context

ScopeOut of Scope

In Scope

Add Line

Asset	Asset Type	Actions
*.meridian.example.com	domain	 
api.meridian.example.com	domain	 
198.51.100.0/24	cdir	 

Copy to clipboard

Figure 19 / Scope configuration panel

Defining Scope

Every project has two scope lists:

- **In-scope:** targets authorized for testing
- **Out-of-scope:** targets that must not be tested, even if discovered during enumeration

Scope entries are validated on creation to ensure they match the expected format for their asset type.

Asset Types

Type	Example	Description
Wildcard	*.example.com	All subdomains under a domain
Domain	app.example.com	A specific hostname
IP	192.168.1.1	A single IP address

Type	Example	Description
CIDR	10.0.0.0/24	An IP address range
iOS App	com.example.app	An iOS application bundle identifier
Android App	com.example.app	An Android application package name
Source Code	github.com/org/repo	A source code repository

Credentials and Context

Beyond target definitions, the scope section supports additional context:

- **Credentials:** test accounts, API keys, and authentication material needed for grey-box and white-box assessments
- **Context:** markdown-formatted notes describing the target environment, architecture, technology stack, and testing guidelines

Adding detailed context helps AI assessments and team members understand the environment without separate onboarding.

Importing Scope

Scope entries can be imported in bulk. This is useful when onboarding a large engagement or syncing scope from a bug bounty platform.

Scope and Asset Discovery

Scope entries serve as seeds for asset discovery:

1. **Wildcard entries** trigger subdomain enumeration to discover hosts
2. **Domain entries** are probed directly for HTTP responses, technologies, and services
3. **IP and CIDR entries** are scanned for open ports and running services

Assets discovered through enumeration are validated against the scope. Only in-scope assets are tracked and tested. Out-of-scope matches are flagged and excluded from automated scanning.

02.13

Narrative

The project narrative is a markdown document that captures the assessment methodology, progress, and overall findings story. It serves as the qualitative counterpart to the findings tracked in the Issues tab.

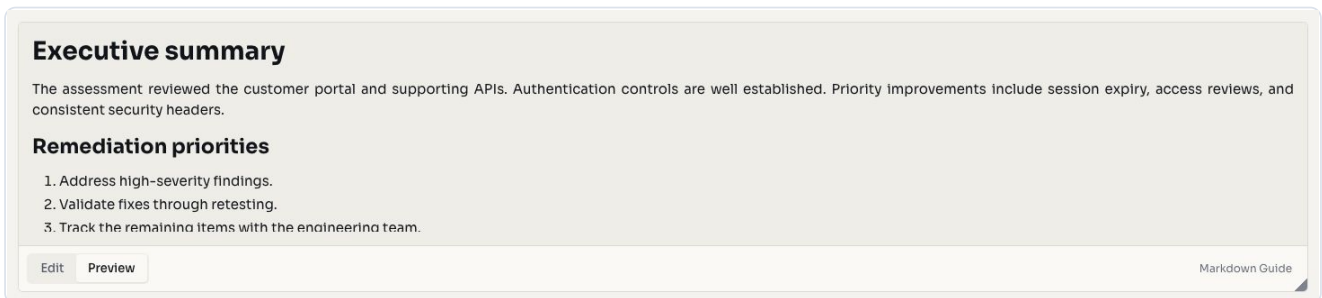


Figure 20 / Narrative editor

Writing the Narrative

The narrative editor supports full markdown: headings, lists, tables, code blocks, inline images, and links.

A typical narrative covers:

- **Executive summary:** high-level overview for non-technical stakeholders
- **Methodology:** approach taken during the engagement (tools, testing phases, coverage)
- **Findings overview:** summary of discoveries organized by impact or category
- **Recommendations:** strategic guidance beyond individual vulnerability fixes

Access Control

Role	Access
Providers / Pentesters	Read and write
Clients	Read-only

Clients can review the assessment documentation without modifying the provider's work.

Narrative in Reports

The narrative content is included in generated PDF reports. When a report is generated, narrative sections are pulled into the executive summary, methodology, and overall assessment description.

Write your narrative with the final report in mind. Content here flows directly into the deliverable, so use professional language and structure it for your audience.

02.14

Attack Chains

Attack chains are visual flow diagrams that show how individual vulnerabilities combine to achieve broader impact. They connect discrete findings into end-to-end exploitation paths that demonstrate real-world risk.

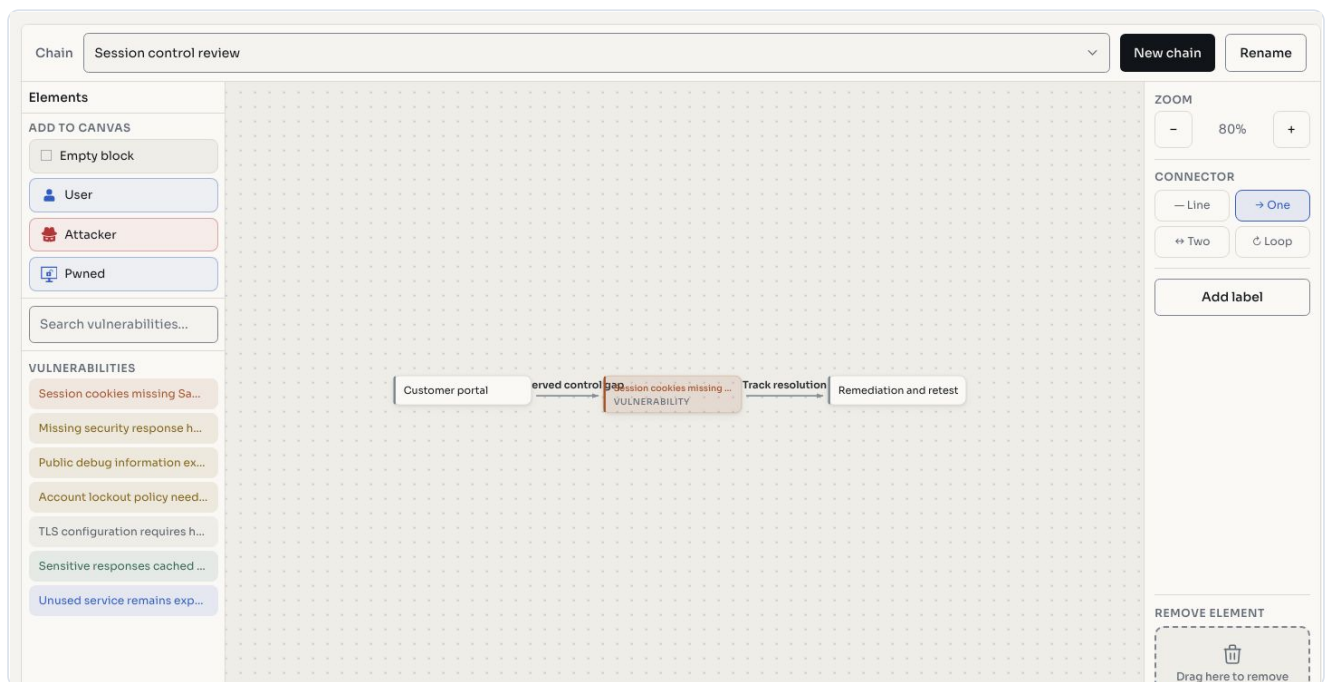


Figure 21 / Attack chain canvas editor

Why Attack Chains

Individual findings often don't convey the full risk picture. A medium-severity information disclosure combined with a low-severity misconfiguration might chain into a critical account takeover. Attack chains make these compound risks visible to stakeholders.

Node Types

Node	Purpose
Vulnerability	A specific finding from the project's issues list
Label	A descriptive text node for annotations or phase names
Block	A system or component (e.g., "Web Server", "Database")
Person	A user role or identity (e.g., "Regular User", "Admin")
Attacker	The threat actor initiating the chain

Node	Purpose
Pwned	The compromised end state (e.g., "Full Account Takeover")

Building Chains

Use the canvas editor to construct attack chains:

1. **Add nodes:** place nodes on the canvas for each step in the attack
2. **Connect edges:** draw directional edges between nodes to show the flow
3. **Link vulnerabilities:** connect vulnerability nodes to actual findings for traceability
4. **Annotate:** use label nodes to describe transitions between steps

You can build multiple chains per project to illustrate different attack scenarios.

Including Chains in Reports

Attack chains are embedded in generated PDF reports alongside the findings they reference. The visual diagram is included as a figure, tying individual vulnerabilities into a coherent risk story.

02.15

Report Generation

Hashiro generates professional PDF penetration testing reports directly from project data, combining narrative, scope, findings, CVSS scores, attack chains, and remediation guidance.

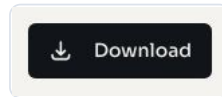


Figure 22 / Report download from project view

Report Contents

A generated report typically includes:

- **Cover page:** project name, client, dates, classification
- **Executive summary:** high-level overview from the narrative
- **Scope:** in-scope and out-of-scope targets
- **Methodology:** testing approach from the narrative
- **Findings:** each vulnerability with severity, CVSS score, description, evidence, and remediation
- **Attack chains:** visual diagrams showing compound exploitation paths
- **Remediation summary:** prioritized action items
- **Appendices:** additional technical details

Generating a Report

1. **Request:** select a report template and initiate generation from the project view
2. **Processing:** the report engine renders the LaTeX template with project data
3. **Download:** once ready, download the PDF from the project

The report status can be polled while generation is in progress.

Templates

Reports are rendered from **LaTeX-based templates** managed by platform administrators. Templates define the visual layout, branding, section structure, and formatting rules.

- Multiple templates can be available per organization
- Templates support **preview** rendering with sample data before use
- Template access can be granted per organization

Contact your platform administrator to customize report templates with your organization's branding, disclaimers, and preferred formatting.

Multi-Language Support

Reports can be generated in English (en) and Portuguese (pt). The project's language setting determines which language is used. Template content and section headers adapt to the selected language.

Report Storage

Generated reports are stored securely and can be re-downloaded at any time. Previous generations remain accessible until the project is deleted.

02.16

Vulnerability Management

Vulnerability management in Hashiro centers around **issues**: individual security findings discovered during assessments. Each issue captures the full lifecycle of a vulnerability from discovery through remediation.

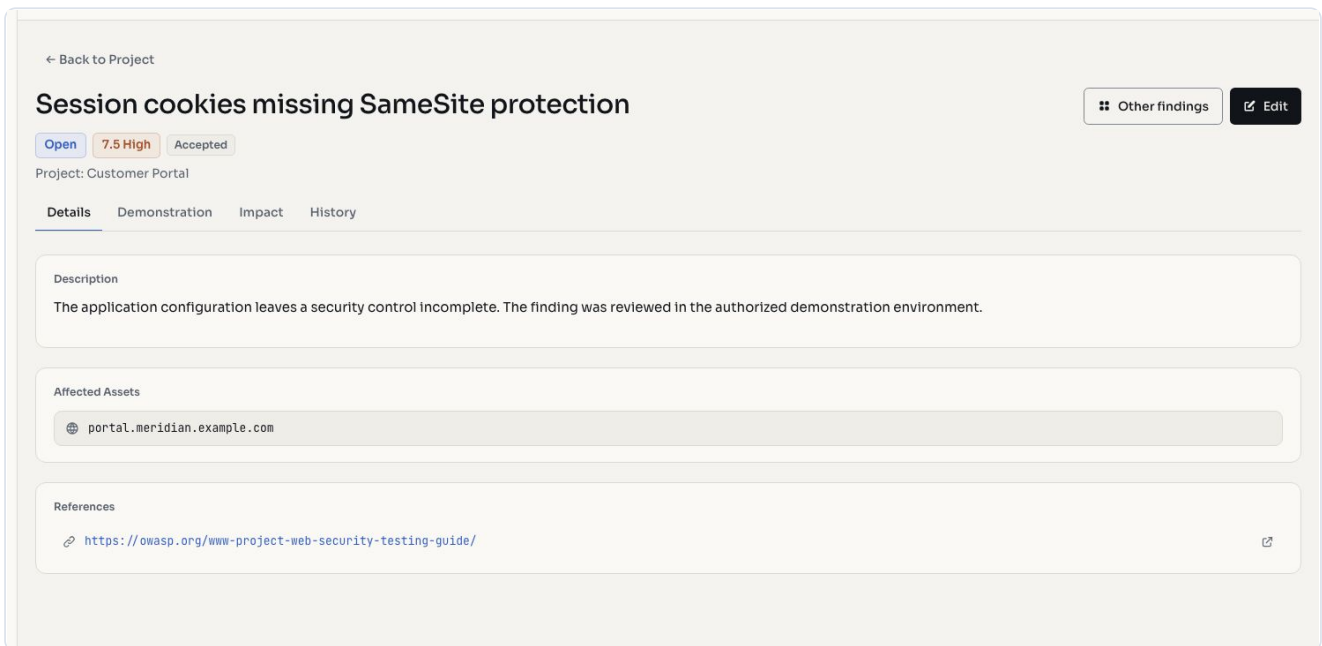


Figure 23 / Issue detail view

Creating Issues

Issues can be created in several ways:

- **Manually:** pentesters create findings using the issue editor
- **From templates:** use a vulnerability template as a starting point
- **Automated scanning:** Nuclei-based scans generate findings
- **AI assessments:** the AI engine discovers and reports findings as candidates

Issue Detail

Field	Description
Title	Clear, descriptive name for the vulnerability
Description	Detailed explanation of the vulnerability and its context
Severity	Critical, High, Medium, Low, or Informative

Field	Description
CVSS v3.1	Standardized scoring with full vector string
Status	Current lifecycle state (see Issue Lifecycle)
Asset	The affected host, URL, or service
Observation	What was observed during testing
Demonstration	Step-by-step proof of exploitation
Remediation	Recommended fix or mitigation
References	External links (CVEs, advisories, documentation)
Evidence	Screenshots, payloads, and response captures

Edit Mode

The issue editor is organized into tabs:

- **Basic Info:** title, description, severity, status, asset, category
- **CVSS / Impact:** CVSS v3.1 calculator and impact assessment
- **Additional Info:** remediation, references, observation, demonstration

The screenshot displays the Hashiro Issue Editor interface. At the top, there is a navigation bar with a 'Back to Project' link. Below this, the issue title 'Session cookies missing SameSite protection' is shown in a text box. To the right of the title are buttons for 'Other findings' and 'Save'. Below the title, there is a dropdown menu set to 'Open', a severity indicator '7.5 High' with a clock icon, and a status indicator 'Accepted'. The project name 'Project: Customer Portal' is displayed below these elements. A tabbed interface is shown with 'Details' as the active tab, followed by 'Demonstration', 'Impact', 'Retest', and 'History'. The 'Description' section is visible, containing the text: 'The application configuration leaves a security control incomplete. The finding was reviewed in the authorized demonstration environment.' At the bottom of the description area are 'Edit' and 'Preview' buttons. A 'Markdown Guide' link is located in the bottom right corner of the editor.

Figure 24 / Issue editor tabs

Attachments

Issues support file uploads for evidence: screenshots, request/response captures, payloads, and supporting documentation.

History Timeline

Every issue maintains an **immutable audit trail** of all changes: status transitions, severity changes, retest results, and who made each change.

Navigation

The **other-findings drawer** lets you quickly switch between issues in the same project without returning to the issues list.

Nuclei Scan Results

Issues from automated Nuclei scanning appear in a dedicated view with paginated results, severity classification, and direct issue creation from scan results.

02.17

Issue Lifecycle

Every vulnerability finding in Hashiro follows a structured lifecycle from discovery through resolution. Understanding these stages helps teams track remediation progress and maintain clear communication.

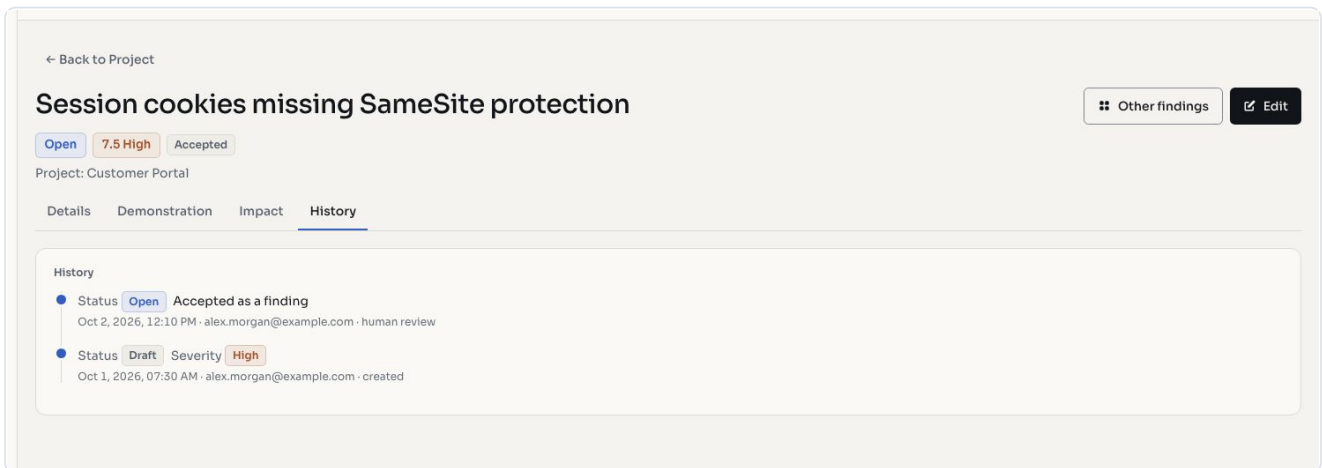


Figure 25 / Issue lifecycle flow

Statuses

Status	Description
Draft	Finding is being documented, not yet visible to clients
Open	Confirmed vulnerability, visible and awaiting remediation
Closed	Successfully remediated and verified
Accepted	Risk accepted by the client, no remediation planned
Pending Retest	Client reports a fix, awaiting verification by the tester
Retesting	Tester is actively verifying the reported fix
Retest Failed	Fix was insufficient, vulnerability still present
Not Applicable	Finding does not apply to the target environment
Duplicate	Finding duplicates another existing issue
Informative	Observation with no direct security impact

Typical Flow

Draft -> Open -> Pending Retest -> Retesting -> Closed
-> Retest Failed -> Open (cycle repeats)

Alternative paths:

Open -> Accepted (risk accepted)
Open -> Not Applicable
Open -> Duplicate
Draft -> Informative

Verification Workflow

For AI-generated findings, an additional verification layer applies:

Verification State	Meaning
Candidate	AI-generated finding awaiting human review
Accepted	Reviewer confirmed the finding is valid
Rejected	Reviewer determined the finding is invalid
Needs Evidence	Additional evidence requested before acceptance
Merged	Finding combined with another existing issue

Candidate findings from AI assessments can be made **publicly visible** to clients before full verification, so clients can see what automated testing discovered while the provider reviews for accuracy.

Retest Tracking

When a client submits a fix for verification:

1. The issue moves to **Pending Retest**
2. The pentester picks it up and moves to **Retesting**
3. After verification, it becomes either **Closed** (fix effective) or **Retest Failed** (vulnerability persists)

Retest comments capture what was tested and the outcome, providing a clear record of remediation attempts.

Bulk Operations

For managing large finding sets efficiently:

- **Bulk status change:** select multiple issues and transition them simultaneously
- **Bulk delete:** remove multiple findings at once

Bulk operations affect all selected findings immediately. Review your selection carefully before applying.

Audit Trail

Every status change, severity modification, and retest event is recorded in the issue's **history timeline**. This immutable log includes what changed, who made the change, and when it occurred.

02.18

CVSS Scoring

Hashiro integrates the **Common Vulnerability Scoring System v3.1 (CVSS)** into every security finding. CVSS provides a standardized method for rating vulnerability severity and communicating risk.

CVSS 3.1 Score Calculator

Attack Vector (AV)

Network (N) Adjacent (A)

Local (L) Physical (P)

Attack Complexity (AC)

Low (L) High (H)

Privileges Required (PR)

None (N) Low (L) High (H)

User Interaction (UI)

None (N) Required (R)

Scope (S)

Unchanged (U) Changed (C)

Confidentiality (C)

None (N) Low (L) High (H)

Integrity (I)

None (N) Low (L) High (H)

Availability (A)

None (N) Low (L) High (H)

Score

7.5 High

CVSS 3.1 Vector String

CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N

Figure 26 / CVSS score calculator

Score Calculator

The built-in calculator lets you set each vector component and automatically computes the overall score.

Exploitability Metrics

Metric	Values	Description
Attack Vector (AV)	Network, Adjacent, Local, Physical	How the vulnerability is exploited
Attack Complexity (AC)	Low, High	Conditions beyond the attacker's control
Privileges Required (PR)	None, Low, High	Access level needed before exploitation
User Interaction (UI)	None, Required	Whether a user must take action

Scope

Metric	Values	Description
Scope (S)	Unchanged, Changed	Whether exploitation impacts resources beyond the vulnerable component

Impact Metrics

Metric	Values	Description
Confidentiality (C)	None, Low, High	Impact on information disclosure
Integrity (I)	None, Low, High	Impact on data modification
Availability (A)	None, Low, High	Impact on service availability

Severity Classification

The CVSS score maps to a severity level:

Score Range	Severity
9.0 – 10.0	Critical
7.0 – 8.9	High
4.0 – 6.9	Medium
0.1 – 3.9	Low
0.0	Informative

Vector String

Each assessment produces a **vector string** encoding all metrics in a compact format:

```
CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
```

This string is stored with the finding and included in reports.

How CVSS Is Used in Hashiro

- **Prioritization:** findings are ranked by score in project views and dashboards
- **Reporting:** scores appear alongside each finding in PDF reports
- **Dashboard metrics:** severity breakdowns derive from CVSS classifications
- **Remediation planning:** the effort matrix combines severity with remediation complexity

02.19

AI-Assisted Review

Hashiro's AI assessment engine generates vulnerability findings as **candidates**, draft findings that require human review before being finalized. The AI review workflow streamlines triaging, validating, and refining automated discoveries.

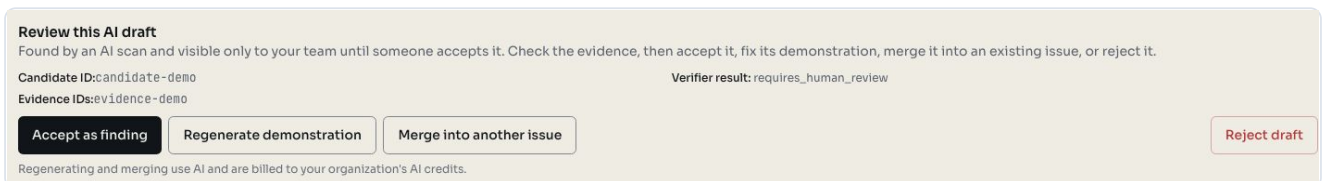


Figure 27 / Candidate review panel

Candidate Review Panel

When AI assessments produce findings, they appear in the **candidate review panel** within the project's Issues tab. Each candidate can be actioned with:

Action	Effect
Accept	Confirms the finding as valid; it becomes a full issue
Reject	Dismisses the finding as a false positive or irrelevant
Needs Evidence	Requests additional proof before making a decision
Merge	Combines with an existing finding covering the same vulnerability

Candidates can be marked as **publicly visible** to clients before full verification. This lets clients see what automated testing discovered while the provider completes the review.

AI Demonstration Regeneration

For accepted or in-progress findings, the AI can **regenerate the exploitation demonstration**:

1. Select a finding and request demonstration regeneration
2. The AI produces an updated step-by-step proof of exploitation
3. Review the generated content in a preview
4. **Apply** the demonstration to update the finding, or discard it

This is useful when the original demonstration needs clearer steps, better formatting, or additional context.

AI Merge Proposals

When the AI detects multiple findings describing the same or overlapping vulnerabilities, it can **propose a merge**:

- The AI analyzes similarity between findings
- A merge proposal shows which findings would be combined
- You can accept or reject the merge
- Accepted merges consolidate findings into a single, comprehensive issue

This prevents duplication and produces cleaner reports.

Workflow Integration

AI-assisted review integrates with the standard [issue lifecycle](#):

```
AI Assessment -> Candidate -> Review -> Accepted (becomes Open issue)
                                         -> Rejected (archived)
                                         -> Merged (combined with existing)
```

All review actions are recorded in the finding's history timeline.

02.20

AI Assessments

An AI assessment organizes security review within a project. An **AI run** is one execution of that assessment: it has a selected profile, scope, operating mode, budget, progress, and results. A project can have several runs so your team can compare observations, review changes, and follow remediation over time.

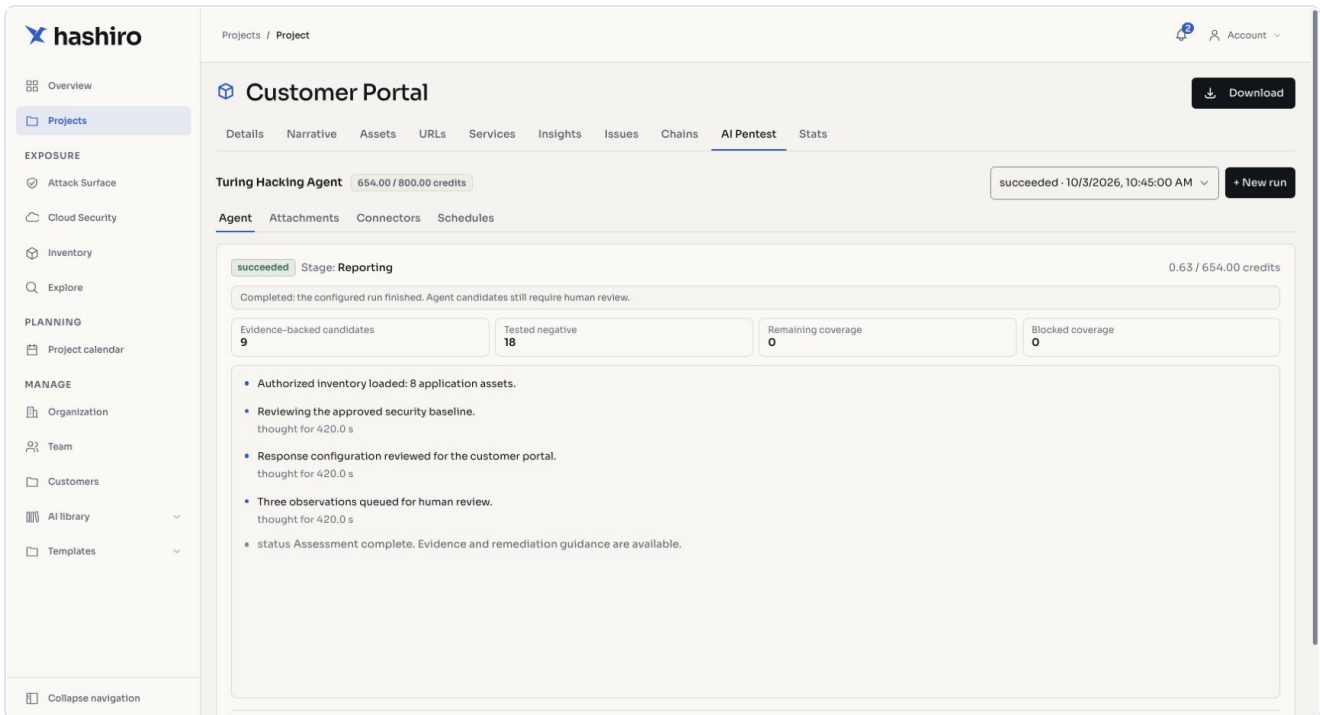


Figure 28 / AI assessment workspace

Understand the available run types

Four choices describe different aspects of a run. Choosing one does not automatically choose the others.

Choice	What it determines	Examples
Interaction mode	How the operator participates	Automatic, Assisted, Checklist
Assessment intent	The breadth and depth of the review	Lightweight, Balanced, Deep, Continuous
Execution location	How the assessment reaches its environment	Platform-managed access, local/private-network access
Assessment purpose	What outcome the team wants to evaluate	Security findings, remediation review, purple team validation through a plugin

[AI Runs](#) explains these choices, prerequisites, budgets, progress, results, and review in detail. [Run Modes](#) shows the controls and how to start an assessment.

Automatic, Assisted, and Checklist

Automatic follows the configured profile without continuous operator chat. **Assisted** lets a provider operator guide the review through chat. **Checklist** organizes the work around a selected list and records item coverage. Availability depends on the organization, project permissions, and configured profiles and checklists.

Clients normally use automatic runs when their provider enables them for the project. Provider controls and client controls can differ. See [Automatic Assessments](#) and [AI Runs](#).

Local runs

A private environment can require local access through a project connector. This describes connectivity to the environment, rather than another interaction mode. A connected relay and an authorized scope are separate requirements. See [Local Runs](#) for readiness, responsibilities, and result interpretation.

Purple team runs through a plugin

When enabled for your organization, a plugin can support coordinated validation of defensive controls. The team reviews whether the intended behavior occurred, whether relevant telemetry was available, and whether detection and response worked as expected. See [Purple Team Runs](#) for the purpose, roles, result categories, and review process.

Review the results

Progress and generated observations do not replace human review. A finished run can contain candidate findings, incomplete coverage, or blocked work. Review the evidence, accept or reject candidates, merge duplicates when appropriate, and track accepted findings through the project's issue workflow.

Related configuration

Topic	Purpose
Agent Types	Profiles available for the project's category
Checklists	Repeatable review objectives and item coverage
Connectors	Connectivity to private environments
Knowledge Base	Reference material and engagement context
Credits & Billing	Available credits and budget management
Schedules	Recurring AI assessments where available

02.21

AI Runs

An **AI run** is a single assessment execution associated with a project. It records the configuration used for that execution, its progress, and its results. The project holds the broader engagement, included scope, exclusions, team access, and issue history.

Use separate runs when the assessment objective changes or when you need a new review after remediation. A previous run remains useful evidence of what was reviewed at that time; it does not establish the current state of an application that has since changed.

Choose the right kind of run

Do not confuse the interaction mode with the assessment intent or execution location. For example, an automatic run can use Balanced intent and a connector for a private environment. A checklist can organize the review of that same environment. Purple team validation through a plugin has a different purpose from discovering application findings.

Your objective	Relevant choice	Review afterward
Run a configured assessment with periodic operator review	Automatic	Evidence, candidates, and incomplete work
Keep an operator involved throughout the review	Assisted	Operator decisions, observations, and resulting findings
Follow a repeatable set of review objectives	Checklist	Each item's outcome and supporting evidence
Review an environment that requires local network access	Local/private-network access	Connectivity, scope, and assessment results
Evaluate detection and response as a coordinated exercise	Purple team through a plugin	Execution evidence, telemetry, detections, and response

Interaction modes

Automatic

The run follows its configured profile without continuous chat input. The operator still defines the engagement context, reviews progress, and evaluates the resulting candidates. Automatic means that the configured review can proceed unattended; it does not mean unlimited coverage, guaranteed findings, or automatic acceptance of every observation.

Use it for a defined baseline review, a repeat assessment, or an assessment that can proceed within an established scope and budget. Check prerequisites and available credits before starting. See [Automatic Assessments](#).

Assisted

A provider operator guides the assessment through chat. This is useful when the review requires interpretation of application behavior, clarification of context, or decisions about which observations need further review.

Keep directions within the project's existing scope and permissions. Chat does not expand project access. An operator can take over an active automatic run when the corresponding control is available. Availability differs between provider and client accounts.

Checklist

The run follows an available checklist instead of relying only on a broad assessment objective. The checklist expresses what should be reviewed and makes coverage easier to compare across runs.

A curated checklist groups specific items into bundles. A coverage template organizes review by categories. Read the outcome of each item together with its evidence: reviewed without a finding, finding identified, not applicable, and blocked work have different meanings. Do not interpret an unreviewed item as a passed control.

The selector appears only when an appropriate checklist is available. A checklist helps structure evidence; it does not by itself establish compliance with an entire standard.

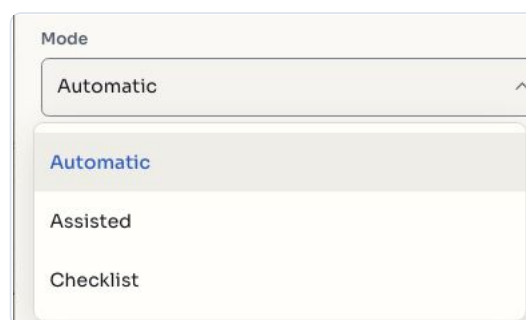


Figure 29 / Available interaction modes

Assessment intent

Intent describes how thorough the configured review should be. It is separate from the choice of Automatic or Checklist. Assisted runs use operator direction and do not present this depth selector.

Intent	Intended emphasis	Planning consideration
Lightweight	A narrower initial review	Useful when time or credits are limited; expect less coverage.
Balanced	A bounded review of the most relevant areas	Review the resulting coverage and any blocked areas before deciding whether more work is needed.
Deep	More detailed review of the configured environment	Allow more time and credits; increased effort still does not guarantee complete coverage.
Continuous	Repeated assessment cycles within configured limits	Set a deliberate budget and monitor changes, coverage, and resource use.

Continuous intent is not a recurring calendar schedule. A schedule creates separate runs at configured times; Continuous applies to the work within one run. Neither removes the need to review results.

The 'New run' configuration form contains the following settings:

- Profile:** Web application assurance
- Model combination:** Balanced · 10000 tokens / credit
- Mode:** Automatic
- Assessment intent:** Lightweight, **Balanced**, Deep, Continuous. Below the buttons, it states: 'Focus on the issues that most likely exist: the probable vulnerabilities across the in-scope surface, time-boxed.'
- Budget (credits):** 50. Below the input field, it states: 'Max credits for this run. Default 50. Set 0 for no limit.'

Figure 30 / Profile, intent, and budget settings

Profiles and model combinations

A profile defines the assessment's configured stages and review objectives. The project category and organization configuration determine which profiles are offered. Choose a profile appropriate for the environment and the purpose of the engagement.

A model combination, when offered, selects the available AI configuration for that run. It can affect credit use and the way the profile processes its work. Do not assume that a more expensive configuration guarantees more valid findings. Compare outcomes using evidence and coverage, rather than cost alone.

Scope, access, and prerequisites

Before creating a run, review included scope, exclusions, project permissions, relevant environment context, and credits. An empty included scope means the assessment cannot run. An available profile does not make every possible target part of the engagement.

For a client, the project must allow client AI runs. For a provider, confirm that the selected project belongs to the intended authorized engagement. If the environment requires private-network connectivity, review [Local Runs](#). If it requires a plugin, check that the organization has the corresponding feature enabled.

Budgets and stopping

The credit budget limits the resources available to a run. A value of **0** means no explicit per-run credit cap in the start form; it does not mean free usage. Available organization credits and other applicable limits still matter.

A run can stop before all planned work is complete because of its budget, an operator action, a missing dependency, or an execution error. Review the final status and the work already recorded. Stopping retains existing observations and findings; it does not certify the remaining scope.

Follow progress

Use the run selector to open the execution you want to review. Check its status, current stage, credit usage, recorded observations, and coverage information. The history of a completed run remains available separately from a newer run.

Distinguish waiting for prerequisites from active work, completion, cancellation, and failure. A connectivity problem or absent prerequisite is a reason to investigate setup, rather than evidence that the environment has no security issues.

For available controls and the start procedure, see [Run Modes](#).

Interpret the outcome

Outcome	What it tells you	What it does not tell you
Candidate finding	An observation needs evidence review	That the issue is already accepted or verified by a person
Accepted finding	A reviewer has accepted the issue into the project workflow	That remediation has been completed
Reviewed without a finding	The recorded review did not establish an issue	That every possible issue was excluded
Blocked or incomplete coverage	Part of the intended review could not be completed	That the affected area passed
Completed run	The execution finished	That all candidates are valid or all coverage is complete

Review candidate evidence, project context, and affected assets before deciding. Accept valid candidates, reject unsupported ones, or merge duplicates when appropriate. Keep the run outcome separate from issue status: an execution can be complete while its accepted findings remain open.

Repeat assessments and compare runs

Record what changed between runs: scope, environment version, profile, checklist, intent, budget, and access conditions. Compare coverage and evidence alongside the number of findings. Fewer findings may reflect remediation, different scope, incomplete work, or different access.

For remediation follow-up, review the affected finding and use the project's retest workflow. A new run can provide additional observations, but its completion alone should not close an issue.

Provider and client responsibilities

Providers make appropriate profiles and project permissions available, coordinate environment access, and review the engagement's objectives. Clients review their authorized project, available credits, results, and remediation work. The buttons shown to each account can differ.

If a profile, checklist, local-access option, or plugin feature is missing, ask the provider or organization administrator to check availability. Do not substitute a different execution method without reviewing the engagement requirements.

02.22

Local Runs

Local access is relevant when an assessment needs to review an environment that is reachable only from an authorized machine or private network. In the documented project workflow, a **connector** provides access to that environment while the assessment remains associated with its Hashiro project.

Local access is an execution-location and connectivity choice. It is not a fourth interaction mode: the run can still use the available Automatic, Assisted, or Checklist mode.

Local access and local execution are different

Concept	Meaning
Private-network access	The assessment reaches an approved private environment through a connected project connector.
Fully local execution	The assessment engine itself runs on the operator's machine or within the local environment. This requires a separately supported execution option.
Plugin execution	A plugin provides a particular assessment capability, such as purple team validation, when enabled for the organization.

Selecting a connector does not establish that the assessment engine runs entirely on your machine. It also does not establish that processing is offline or that results never leave the environment. Check the applicable organization configuration and data-handling requirements with your provider.

When local access is useful

Examples include private applications, an internal API, a staging environment reachable only from an approved network, or an environment with specific access requirements. Reachability is only one prerequisite: the project must also contain the intended included scope and exclusions.

A machine being reachable through a connector does not automatically make every resource reachable from that machine part of the assessment.

Responsibilities before the run

The provider and environment owner should agree on the intended environment, approved access, review objective, time window, and how evidence will be handled. Keep the project context current so the reviewer can interpret the resulting observations.

The operator should confirm that the project offers the required connector, that it belongs to the intended environment, and that its status is connected. Use the organization's provided connector setup instructions and review access with the environment owner. Installation and access configuration depend on the environment and should not be inferred from a generic screenshot.

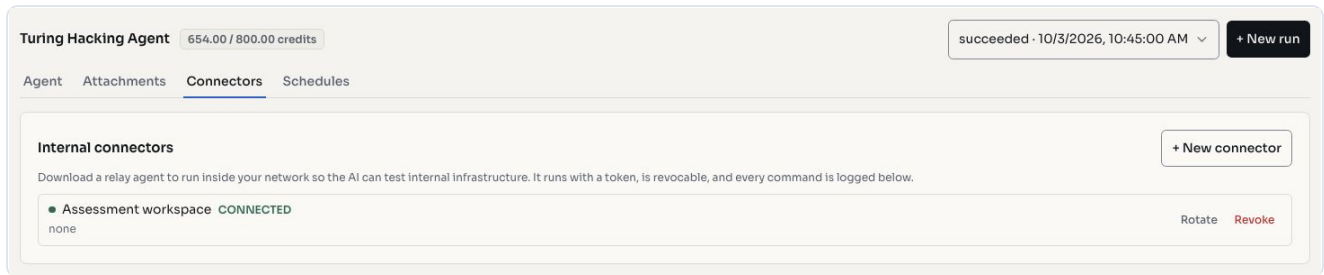


Figure 31 / Project connector status

Readiness in the project

Open the project's **AI Pentest** area and review its connector panel. A connected status indicates that the relay is available; it is not proof that every intended application is reachable or that every assessment prerequisite is satisfied.

Review readiness in three parts:

Part	What to confirm
Project	Correct engagement, included scope, exclusions, and permissions
Environment	Appropriate access to the intended private resources and required context
Run	Appropriate profile, interaction mode, intent, credit budget, and selected connector

If more than one connector is offered, identify the environment each represents before choosing. Do not select one solely because it is currently connected.

Monitor access separately from results

Follow the connector's connection status and last-seen information alongside the run's progress. Available activity records can help the operator distinguish an access problem from an assessment observation.

If connectivity is lost, part of the review may remain blocked or incomplete. Restore the approved access through the environment's normal support process and review the affected coverage before deciding whether a follow-up run is needed. Do not mark unreviewed areas as passing because the run stopped without finding an issue.

A connected connector does not guarantee a successful run. Missing project context, unavailable dependencies, insufficient credits, and execution errors can still affect the outcome.

Review and close out

Review candidates and evidence through the project, using the same issue workflow as other AI runs. Keep the connector's activity history separate from the acceptance of a finding: activity demonstrates that work occurred, while a finding needs evidence of the issue.

At the end of the engagement, have the environment owner review whether local access is still needed. Connector revocation and rotation belong to the authorized access-management process. Preserve the assessment evidence according to the organization's retention requirements.

For operating modes, see [AI Runs](#). For a coordinated defensive exercise using a plugin, see [Purple Team Runs](#).

02.23

Purple Team Runs

A purple team run is a coordinated exercise that evaluates defensive visibility and response against an agreed validation objective. When enabled for the organization, a **plugin** provides the assessment capability. Availability depends on the organization's access and the configured plugin; it is not included automatically with every AI assessment.

The purpose is to understand the relationship between observed activity, telemetry, detections, and response. An application finding and a detection gap are different results and should be reviewed separately.

Compare the assessment purposes

Aspect	AI security assessment	Purple team run through a plugin
Main question	What supported security observations should the team review?	Did the intended defensive controls observe and respond to the agreed validation?
Typical context	Project scope, application context, profile, and coverage objectives	Validation objective, environment, available telemetry, and defensive expectations
Evidence	Observations supporting candidate findings and remediation review	Execution records, telemetry, detections, and response records
Follow-up	Accept or reject candidates and track remediation	Investigate detection or response gaps and verify improvements

A purple team exercise does not replace an application assessment. An application assessment also does not establish that monitoring and response would detect the behavior evaluated in a purple team exercise.

Roles and coordination

The **provider or exercise coordinator** defines the authorized validation objective, confirms feature availability, and coordinates the exercise with the environment owner. The **environment owner** confirms the approved environment, time window, and access requirements. The **defensive team** identifies the relevant telemetry and the expected detection and response outcomes.

The **reviewer** reconciles the records from the run with the defensive team's evidence. These roles can be performed by the same people in a small team, but their responsibilities remain distinct.

Provider and client visibility can differ. A client should see only the validation options and results made available to its organization. Plugin administration and access setup remain with the authorized administrators.

Define the objective before execution

Write down the defensive question you want to answer. Examples include whether a control produces the expected telemetry, whether an alert reaches the responsible team, or whether the response process records the expected follow-up.

Document the environment and agreed success criteria so the reviewer can judge the result. A useful record includes the objective, permitted environment, expected evidence sources, relevant time window, responsible reviewers, and conditions that would make the result inconclusive.

Avoid treating every successful execution as a successful defense validation. The run may complete while the telemetry needed for evaluation is missing.

Run types and depth

Purple team describes the purpose of the exercise. It should not be confused with Automatic, Assisted, or Checklist in the AI run form. A plugin can present different controls and prerequisites from an ordinary AI run.

A single focused validation answers a narrower defensive question. A broader exercise can review multiple objectives, but each still needs its own evidence and outcome. A repeat validation checks an improvement against the earlier baseline. Scheduling, where offered, creates another execution and does not eliminate the need to coordinate review.

Use only the options actually made available by the plugin for the organization. The presence of an AI profile does not imply that the plugin supports it, and a plugin catalog entry does not imply that all execution prerequisites are ready.

Availability and readiness

The organization needs the feature enabled, an available validation option, the necessary permissions, and an environment that meets the plugin's readiness requirements. Where a local component is required, the environment owner should use the supplied setup instructions for that specific environment.

The documented AI connector and a plugin's local component are separate concepts. Do not assume that either can substitute for the other. See [Local Runs](#) for the distinction between private-network access and local execution.

If no compatible option appears, ask the provider to review access and setup. A connection problem, missing prerequisite, or unavailable telemetry should remain visible as an incomplete or inconclusive outcome rather than being recorded as a passed control.

Monitor the run and collect the evidence

Keep the run's execution status separate from the defensive outcome. Review whether the intended validation occurred, whether the environment was ready throughout the relevant period, and whether the required evidence is available.

Where the plugin exposes execution history or artifacts, use them with the defensive team's telemetry and incident records. Correlate the relevant environment and time window. An execution record alone cannot establish whether an alert was generated, delivered, investigated, or resolved.

Interpret outcomes separately

Result category	Question for the reviewer
Execution	Did the agreed validation occur, or was it blocked or incomplete?
Telemetry	Was the expected observation recorded and available for review?
Detection	Did the intended control produce the expected detection?
Response	Was the expected investigation or response carried out and recorded?

These categories can have different outcomes in the same run. Activity can occur without the expected detection. A detection can occur without a completed response. Missing telemetry can make a conclusion impossible even when the execution completed.

Use clear outcomes such as supported, not supported, blocked, and inconclusive in the exercise report. Preserve the evidence and explain why the reviewer reached the conclusion. Do not claim a broader level of protection than the exercise established.

Findings and remediation

A defensive gap should identify the reviewed control, expected outcome, actual observation, evidence, and the team responsible for follow-up. It may call for a configuration change, telemetry improvement, alert review, or process correction.

Track accepted work through the organization's issue process when that workflow is available. Keep issue status separate from run status. A completed purple team run does not mean that its follow-up work is closed.

After improvements, compare a repeat validation with the original objective and evidence. Record changes in the environment or validation conditions so that the comparison remains meaningful.

For AI interaction modes and result review, see [AI Runs](#) and [Run Modes](#).

02.24

Agent Types

Agent types are reusable assessment pipeline definitions. Each agent type configures the full behavior of an AI assessment: stages, tools, models, and capabilities.

Edit profile [X]

Basics

Name

Web application assurance

Description

Review application security controls and organize evidence for human review.

Directions

Review the approved inventory and document security control observations for human review.

Always-on brief for this agent: what it's for and how it should operate. Applied to every stage. 90 / 8192

Main AI license **Default model**

Select a license Medium

Orchestrates the run; stages can override it below. Resolved by the run's chosen model combination.

Figure 32 / Agent type builder interface

Creating an Agent Type

The **Agent Type Builder** lets you define every aspect of an assessment pipeline.

Stages

Each agent type consists of one or more sequential stages:

Field	Description
Goal	What the agent should accomplish in this stage
System prompt	Instructions that shape the agent's behavior
Tools	Available tools (exec, http_request, galileo, knowledge_base, etc.)
Operation classes	Categories of allowed operations
Effect limits	Boundaries on destructive or high-impact actions
Token budget	Maximum tokens the agent can consume

Field	Description
Loops	How many execution loops the agent can perform

A typical pipeline defines four stages: Discovery, Vulnerability Analysis, Exploitation, and Reporting.

Bundles

Bundles are **capability packages** that group related tools and permissions. They let you compose agent capabilities without configuring each tool individually.

License Assignment

Each agent type is associated with an AI license that determines which LLM provider and API key to use.

Model Selection

Configure which LLM model the agent uses for:

- **Execution:** the primary model that makes tool calls and processes results
- **Strategist:** an optional reasoning model that advises on next steps

Model combinations allow tier-based selection (lightweight, medium, smarter) so different tasks use appropriately sized models.

Advanced Features

Breadth Sweep

Enable **breadth sweep** to run a background generalist scanning sub-agent alongside the main assessment. This agent performs broad surface-level testing across the entire scope while the primary agent goes deep on specific areas.

Coordination

Enables multi-agent assignment scheduling. Multiple agents are orchestrated to cover different parts of the scope without duplicating work.

Exploitation

Toggle **autonomous exploitation mode** to allow the agent to attempt exploitation without human confirmation.

C2 Integration

For adversary emulation assessments, agent types can include **C2 (command-and-control)** capabilities for red team operations.

Default vs Custom Agent Types

Hashihiro provides **default agent types** available to all organizations, maintained by the platform team. Organizations can also create **custom agent types** tailored to their testing methodologies or compliance requirements.

Providers can use the **Hide default agents** toggle to show only their custom types.

Import & Export

Agent types can be exported as JSON and imported into other organizations, making it easy to share configurations across teams.

02.25

Automatic Assessments

Automatic Assessments are the unattended mode of the project's AI evaluation. Availability depends on your license, credits, project access, and the agent types enabled for your organization.

Before starting

Review the project's included and excluded scope, environment context, authorized test boundaries, and available credits. Confirm that the selected agent type is appropriate for the engagement. Clients use their own authorized project context; providers use the authorized client engagement.

Configure an assessment

In the project's **AI** tab, the start form offers agent type, model combination when available, run mode, depth, credit budget, connector when required, and additional instructions. Choose **Automatic** for an unattended run.

These options shape the evaluation within the project's configured permissions and boundaries. Do not use instructions to expand the authorized scope.

Monitor progress

Use the streaming view to follow stage progress, tool results, recorded observations, artifacts, and candidate findings. A completed run is not a guarantee that every possible vulnerability was tested or that every candidate is valid.

Controls

Automatic mode has stage controls, redo controls, and stop. Chat is available when a run is taken over into the assisted workflow, rather than as an always-on automatic-mode control. Stopping preserves the results already recorded; check the final run status and credit transaction afterward.

Review the outcome

Open the project's findings and review candidates against their evidence. Accept valid findings, reject invalid ones, request additional evidence, or merge duplicates where permitted. Track remediation through the normal issue lifecycle.

See [AI-assisted review](#), [Credits](#), and [Run modes](#).

Recurring assessments

If schedules are available to your organization, use the [AI schedules](#) feature to create recurring evaluations. AI schedules are separate from the platform's internal surface-monitoring schedules.

02.26

Run Modes

Choose a mode when starting an AI assessment from a project's **AI Pentest** tab. Available controls depend on your organization's license, your permissions, the project's client settings, and the assessment profiles and checklists configured for the project.

For a detailed comparison of interaction modes, intent, execution location, and outcomes, see [AI Runs](#). Local access is explained in [Local Runs](#), and plugin-based defensive validation in [Purple Team Runs](#).

Choose a mode

Mode	How you work	Availability
Automatic	The assessment follows the selected profile and reports progress for you to review.	Providers and clients whose project allows client AI runs.
Assisted	A provider operator guides the assessment through chat and reviews its observations.	Providers with the relevant project permissions.
Checklist	The assessment follows a selected checklist and records item coverage.	Shown when an available checklist is configured.

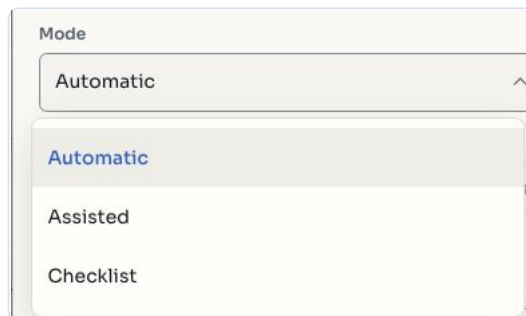


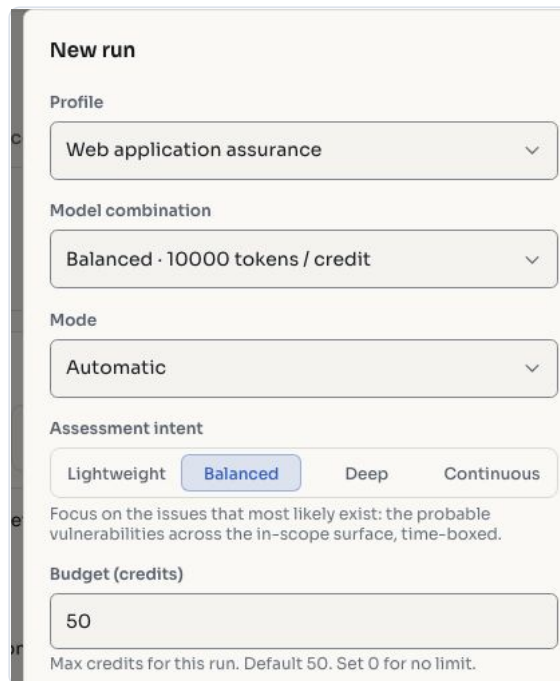
Figure 33 / Automatic, Assisted, and Checklist mode options

Client accounts normally start automatic assessments and may not see a mode picker. If a control is absent, ask your provider to check your project access and available profiles.

Start an assessment

1. Open **Projects** and choose the project you are authorized to assess.
2. Review its included scope and exclusions. Without included scope, the assessment cannot run. Confirm that the project has the required access and an available credit balance.
3. Open **AI Pentest**, then click **New run**.
4. Select a **Profile**. Only profiles available for the project's category are listed.
5. Choose a **Model combination** if that field is shown.

6. Choose **Automatic**, **Assisted**, or **Checklist** when the mode picker is available. For Checklist, also select the checklist.
7. For Automatic and Checklist, choose the coverage depth: **Lightweight**, **Balanced**, **Deep**, or **Continuous**. Continuous repeats the configured workflow until its limits are reached; set an appropriate credit budget. Assisted does not use the depth selector.
8. Set the credit budget, choose a connector if the project requires one, and add instructions describing the permitted scope and review objectives.
9. Review the configuration, then click **Start**. The execution enters the queue and its progress appears in the project.



New run

Profile

Web application assurance

Model combination

Balanced - 10000 tokens / credit

Mode

Automatic

Assessment intent

Lightweight **Balanced** Deep Continuous

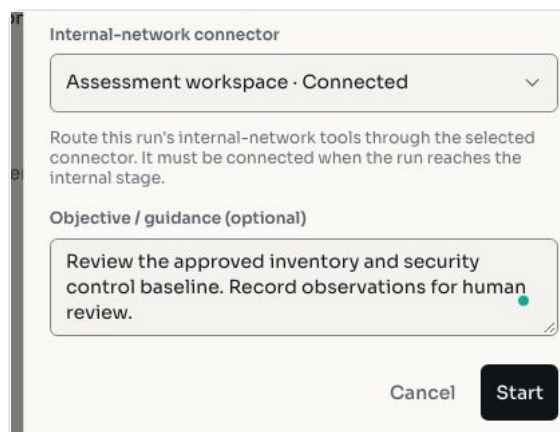
Focus on the issues that most likely exist: the probable vulnerabilities across the in-scope surface, time-boxed.

Budget (credits)

50

Max credits for this run. Default 50. Set 0 for no limit.

Figure 34 / Profile, mode, assessment intent, and budget



Internal-network connector

Assessment workspace - Connected

Route this run's internal-network tools through the selected connector. It must be connected when the run reaches the internal stage.

Objective / guidance (optional)

Review the approved inventory and security control baseline. Record observations for human review.

Cancel Start

Figure 35 / Connector, review objective, and Start button

Monitor and review

Select a run to see its status, current stage, timeline, and recorded observations. Automatic runs follow the profile without continuous chat input. Assisted runs let the operator provide direction through chat. Checklist runs record progress against the selected items.

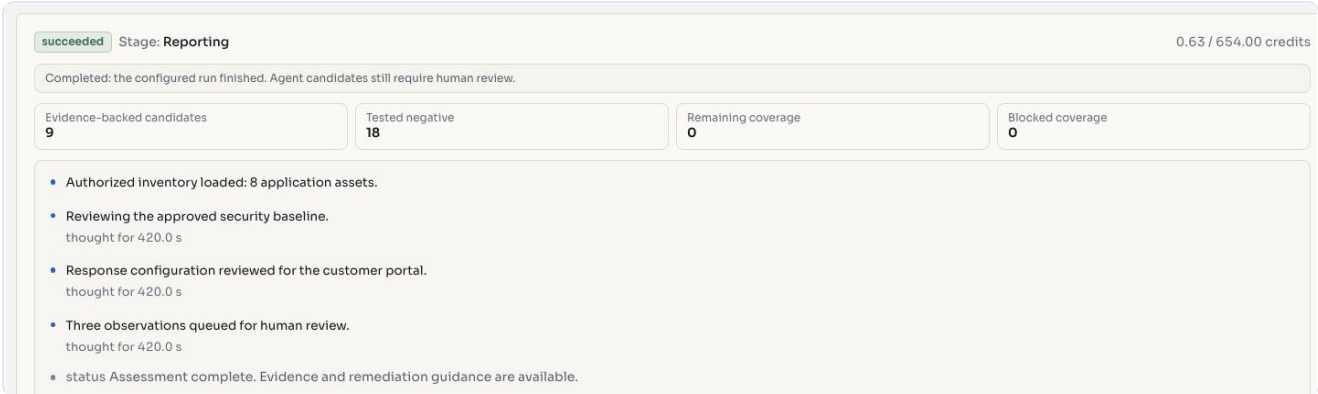


Figure 36 / Assessment progress and timeline

Run controls

The controls shown depend on the run's status, mode, and your permissions. Completed runs retain their history for review.

Control	Purpose
Stop	Stop an active run while retaining findings already recorded.
Skip stage	Request the next configured stage when available.
Take over	Take control of an active automatic run through Assisted mode when permitted.
Redo	Repeat a selected part of the assessment.
Chat	Provide direction in Assisted mode.

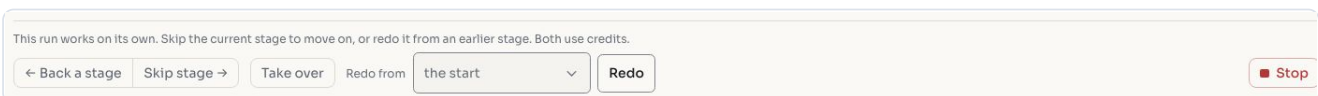


Figure 37 / Active assessment controls

Review candidate findings before accepting them. Link candidates to existing issues or create new records as appropriate, then track remediation through the [issue lifecycle](#). See [automatic assessments](#) for client access and prerequisites.

02.27

Checklists

Checklists provide structured testing guides for AI assessments running in [Checklist mode](#). They define exactly what should be tested to ensure systematic coverage.

Figure 38 / Checklist builder interface

Checklist Builder

Create and manage checklists with:

- **Title and description:** what the checklist covers
- **Test items:** individual checks the agent will perform, each with a description and expected behavior
- **Bundles:** groups of related test items organized by category

Types of Checklists

Regular Checklists

A flat list of test items executed sequentially. Each item is marked as passed, failed, or not applicable as the assessment progresses.

Coverage Checklists

Coverage checklists classify test items by category, enabling **coverage tracking**. The agent must achieve a minimum coverage threshold before transitioning to the reporting stage.

Default vs Custom Checklists

Hashiro provides **default checklists** covering common methodologies (OWASP Top 10, API security, web application testing). Organizations can create **custom checklists** for:

- Internal security standards
- Compliance frameworks (PCI-DSS, SOC 2, HIPAA)
- Application-specific test cases
- Client-specific requirements

Providers can use the **Hide default checklists** toggle to show only their custom checklists.

Using Checklists

1. Select **Checklist** as the run mode when creating a new AI assessment
2. Choose a checklist from the library
3. The agent works through each item, executing tests and recording results

Each item's status provides clear evidence of what was tested and the outcomes, useful for compliance reporting and audit trails.

02.28

AI Schedules

AI schedules run automated assessments on a recurring basis using cron expressions. Set up a schedule once, and Hashiro launches assessments at the defined intervals.

Edit schedule [X]

Profile
Web application assurance [v]

Mode
Automatic [v]

Assessment intent
☐ Lightweight ☒ **Balanced** ☐ Deep ☐ Continuous
 Focus on the issues that most likely exist: the probable vulnerabilities across the in-scope surface, time-boxed.

Model combination
Select... [v]

Frequency
☐ Every day ☒ **Weekly** ☐ Twice a month ☐ Monthly
☐ Custom cron

Day of week
☐ Sun ☒ **Mon** ☐ Tue ☐ Wed ☐ Thu ☐ Fri ☐ Sat

Max credits per run

Override the credit budget for each scheduled run. 0 uses the agent type's default.

Objective / guidance (optional)

[Cancel] [Save]

Figure 39 / Schedule builder with cron expression

Creating a Schedule

The schedule builder lets you configure:

Field	Description
Agent type	Which assessment pipeline to use
Run mode	Automatic, Assisted, or Checklist
Assessment depth	How thorough the assessment should be
Cron expression	When to run (e.g., 0 2 * * 1 for every Monday at 2 AM)
Project	Which project to assess

Managing Schedules

Each schedule can be:

- **Enabled:** actively running on its cron schedule
- **Disabled:** paused, no new runs are created
- **Edited:** update any configuration field
- **Deleted:** permanently removed

Execution History

Every schedule has an **execution history** tab showing:

- Past runs with status (completed, failed, canceled)
- Start and end times
- Number of findings discovered
- Link to the full run details

Use Cases

- **Continuous validation:** schedule weekly or monthly assessments to catch new vulnerabilities from deployments or configuration changes.
- **Regression testing:** periodic re-tests to confirm fixes hold.
- **Compliance:** evidence of regular security testing for frameworks requiring periodic assessments.

Combine schedules with [Checklists](#) for fully automated compliance testing: the assessment runs on schedule, follows a standardized checklist, and produces auditable results.

02.29

Connectors

Connectors are lightweight relay agents that let Hashiro's AI assessments reach targets inside private networks without requiring VPN access or firewall changes.

How Connectors Work

A connector is a small binary (turing-connector) deployed inside the customer's network. It establishes an **outbound** connection to the Hashiro control plane, creating a secure relay for assessment agents to reach internal targets.

```
[AI Agent] -> [Hashiro Control Plane] <- [Connector] -> [Internal Targets]
```

Because the connection is outbound, no inbound firewall rules are needed.

Security Model

Feature	Description
Token authentication	Each connector authenticates with a unique token
TOFU key pinning	Trust-on-first-use server key pinning prevents MITM attacks
Token rotation	Tokens can be rotated without redeploying the connector
Self-destruct	Connectors can be remotely deactivated, terminating the relay
Outbound-only	No inbound ports need to be opened

Capabilities

Once connected, the relay supports:

- **Command execution:** run assessment commands on the internal network
- **HTTP proxy:** route HTTP requests through the connector to reach internal web applications

Deployment

1. Download the turing-connector binary for your platform
2. Configure it with your organization's connector token
3. Run it on a host inside the target network
4. The connector appears in the Hashiro UI under the project's connector panel

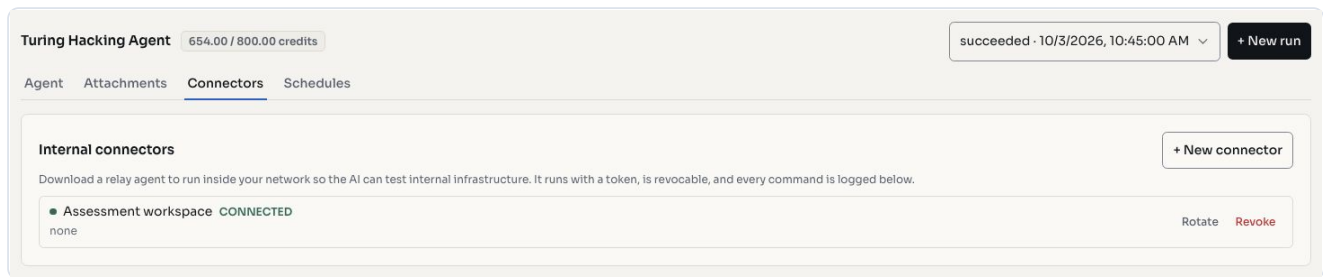


Figure 40 / Connector activity panel showing connection status

Monitoring

The **Connector Activity** panel shows:

- Connection status (online/offline)
- Last seen timestamp
- Activity log of relayed operations

Connectors inherit the network permissions of the host they run on. Deploy on hosts with appropriate access to the assessment targets.

Use Cases

- Testing internal web applications not exposed to the internet
- Assessing staging environments behind corporate firewalls
- Evaluating internal APIs and microservices
- Running authenticated assessments against internal identity providers

02.30

Knowledge Base

The knowledge base provides reference material that AI agents query during assessments. It gives agents domain-specific context that improves testing decisions.

What Goes in the Knowledge Base

Knowledge base entries are **markdown notes** containing information relevant to your assessments:

- Application architecture and technology stack details
- Known authentication flows and session management patterns
- Business logic descriptions and critical workflows
- Previously discovered vulnerability patterns
- Environment-specific configuration details
- Testing methodology notes and custom techniques

Managing Entries

From the Knowledge Base panel:

- **Add** new entries with a title and markdown content
- **Edit** existing entries to keep information current
- **Delete** entries that are no longer relevant

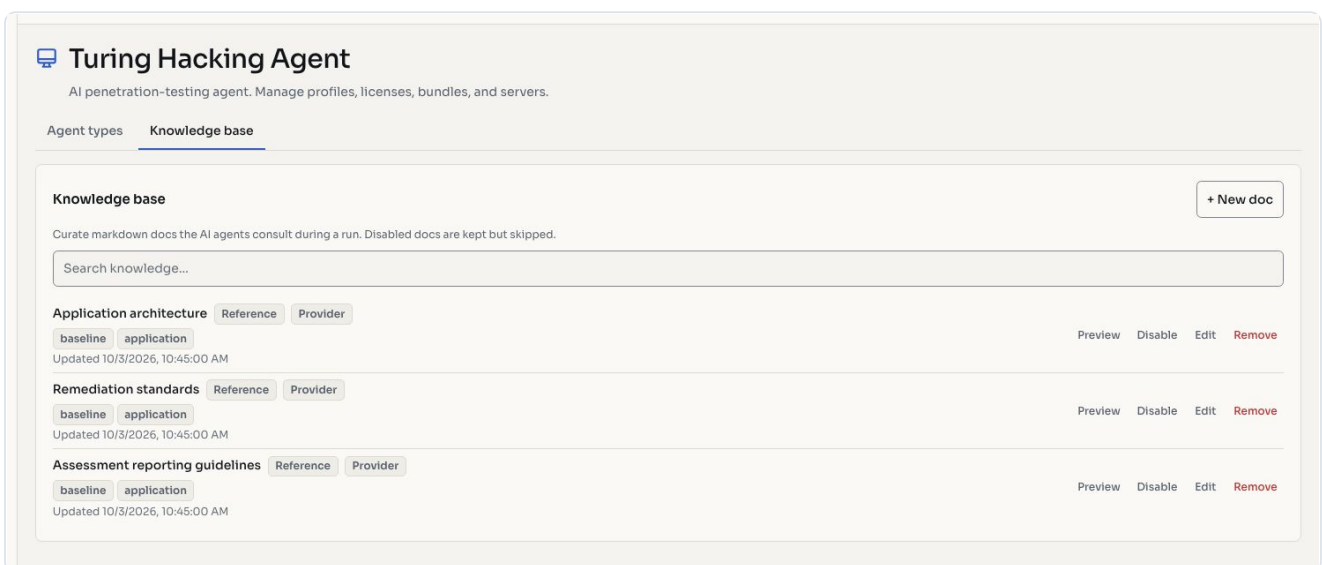


Figure 41 / Knowledge base panel with entries

How Agents Use the Knowledge Base

During an assessment, the agent searches and retrieves relevant entries to:

- Understand the target application's architecture before testing
- Apply domain-specific testing techniques
- Avoid repeating tests documented as dead-ends
- Focus on areas known to be high-risk

The [strategist model](#) also draws from the knowledge base when recommending next steps.

Coverage Templates

Knowledge base documents can be linked to **coverage templates**: keyword-triggered test class assignments. When the agent encounters certain technologies in the target, coverage templates point it to the relevant KB entries and testing procedures.

This creates a feedback loop: as your team documents findings and testing approaches, future AI assessments become more targeted.

Treat the knowledge base as a living document. Update it after each assessment with lessons learned, new attack patterns, and application changes.

02.31

Credits & Billing

AI assessments consume **credits**, a unified currency that abstracts away LLM token costs. Credits fund assessment runs, AI-assisted issue review, and template generation.

Credit Pools

Each organization has two credit pools:

Pool	Description
Monthly allowance	Granted automatically each month based on your license tier. Resets monthly.
Purchased credits	Bought through Stripe. Do not expire.

Credits are consumed **monthly-first**: the monthly allowance is spent before purchased credits, keeping purchased credits as a reserve.

Purchasing Credits

1. Navigate to the AI Assessments section
2. Open the **Credits** panel
3. Click **Purchase Credits**
4. Complete checkout via Stripe

The screenshot shows a 'Credits' panel with a light gray background. At the top left, it says 'AI credits' and 'Available'. On the top right, the total balance is '2160.00'. Below this, there are two boxes: 'Monthly used' showing '0.00 / 1500.00' and 'Purchased' showing '900.00'. A line of text states: 'Monthly credits renew each month and are spent first; purchased credits carry over.' Below this, there is a section 'Credits to buy' with a text input field containing '100' and a black button labeled 'Buy credits'. At the bottom, a note says: 'Purchased in multiples of 10; charged in your org's currency. Bought credits never expire.'

Figure 42 / Credits panel showing balance and purchase option

Subscription Plans

Subscriptions include a monthly credit allowance and are managed through Stripe:

- **Subscribe** to a plan from the billing section

- **Manage** your subscription via the Stripe billing portal
- **View status** of your current subscription (active, past due, canceled)

If a subscription enters **past due** status, there is a 30-day grace period before billing is frozen. Keep your payment method current to avoid interruption.

Credit Escrow

When a run starts, credits are **escrowed** (reserved from your balance) to cover the estimated cost. If the run uses fewer credits, the difference is returned. Unused escrowed credits are released if the run is canceled.

Monitoring Usage

The Credits panel shows:

- Current balance (monthly + purchased)
- Monthly allowance and reset date
- Recent credit transactions
- Per-run credit consumption

02.32

Inventory

The Inventory provides a **cross-project view** of all discovered assets across your organization. Instead of navigating into individual projects, the inventory aggregates everything into a single, filterable interface.

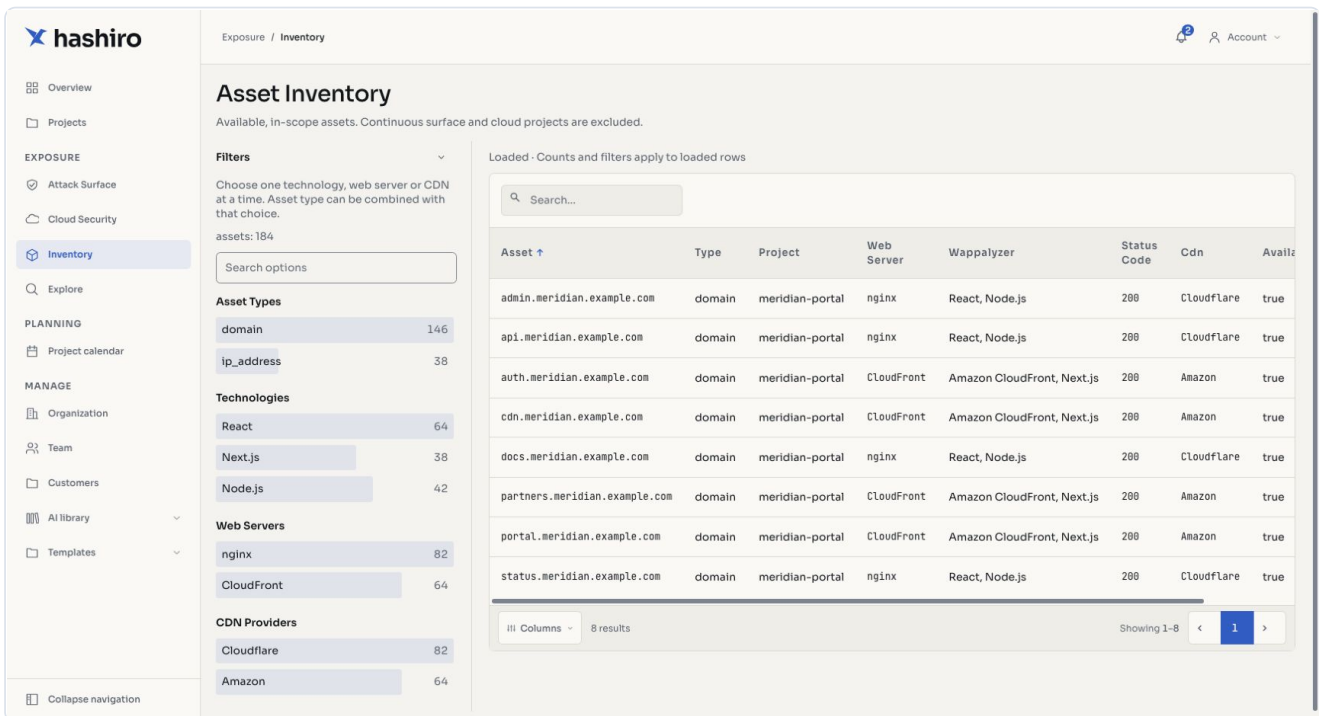


Figure 43 / Inventory with faceted filters

Faceted Filtering

The filter sidebar lets you narrow down assets using multiple dimensions:

Filter	Description
Type	Asset type (domain, IP, wildcard, etc.)
Technology	Detected technologies via Wappalyzer (e.g., React, nginx, WordPress)
Web Server	Underlying web server (Apache, nginx, IIS, etc.)
CDN Provider	Content delivery network in use (Cloudflare, Akamai, etc.)

Each filter option shows a **count bar** indicating how many assets match, helping you understand the distribution of your attack surface at a glance.

Filters support **in-filter search** for quickly finding specific values in long lists.

Asset Table

The main view displays assets with the following columns:

Column	Description
Asset	Hostname, IP, or identifier
Type	Asset classification
Project	Which project the asset belongs to
Web Server	Detected web server software
Technologies	Wappalyzer-detected technology stack
Status Code	HTTP response status from the last probe
CDN	CDN provider, if detected
Availability	Whether the asset is currently reachable

Asset Detail

Clicking an asset opens its **detail view**:

- Screenshot preview from the last HTTP probe
- Full HTTP response details (headers, status, content)
- Technology detection results
- Availability and scope status
- Associated services and URL paths
- External link to the live asset

[Back to results](#)

 **portal.meridian.example.com** [meridian-portal](#)

Available In Scope

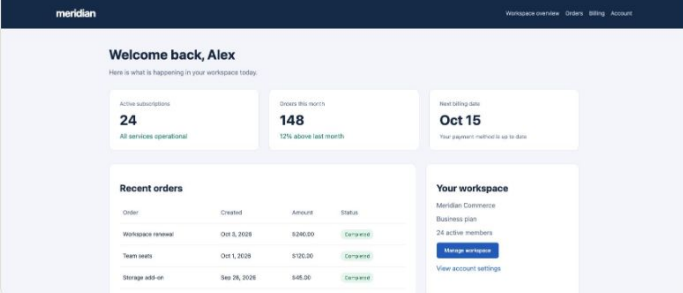
Overview

Services

URLs

 **Screenshot**

[Open observed host in a new tab](#)



 **Asset Information**

ASSET:
portal.meridian.example.com
IP ADDRESSES:
198.51.100.20
TAGS:
environment: production
owner: Platform Engineering

 **HTTP Response**

Status Code200

Content Length18.0 KB

Page TitleMeridian Portal

Web ServerCloudFront

 **Technologies Detected**

Amazon CloudFront Next.js

 **HTTP Headers**

2 of 2 headers

Content-Type
text/html; charset=utf-8

Strict-Transport-Security
max-age=31536000

Figure 44 / Asset detail view

02.33

Explore

Explore provides **cross-project search** across all assets, URLs, and findings in your organization. Quickly locate specific targets, patterns, or vulnerabilities without knowing which project they belong to.

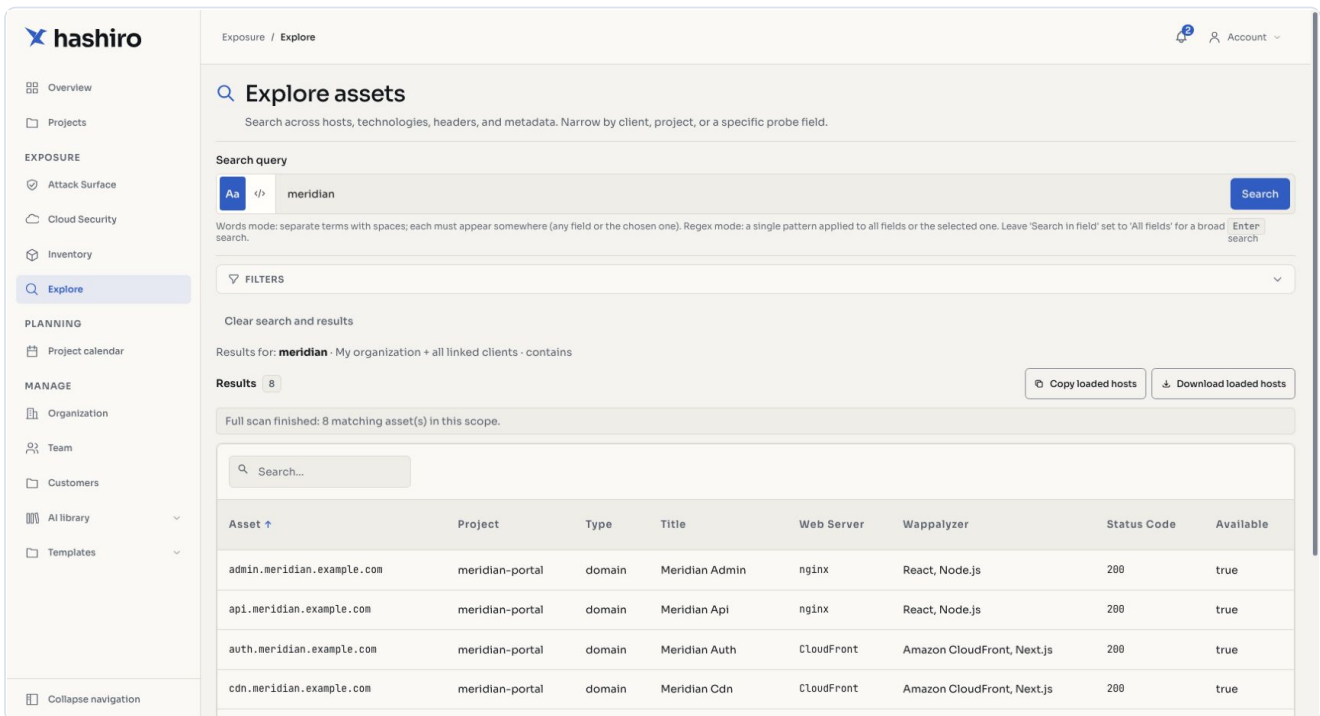


Figure 45 / Explore search interface

Search Modes

Mode	Description
Text Contains	Simple substring matching. Finds assets containing the search term.
Regex	Regular expression pattern matching for advanced queries.

Example Searches

- admin: find all assets with "admin" in the name
- .*\.staging\.example\.com: regex to find all staging subdomains
- wp-login: find WordPress login pages across all projects
- 10\.0\.\.*: regex to find assets in the 10.0.x.x range

Scope-Aware Results

Search results are scoped based on your role:

Role	Search Scope
Provider	All assets across the provider's network (own org + all client orgs)
Client	Assets within the client's own organization only

Results

Search results display matching assets with key metadata. You can see which project each result belongs to, view probe status, and navigate directly to the asset detail.

02.34

Project Calendar

The project calendar provides a visual timeline of security assessment engagements, helping you plan, schedule, and track projects across your organization.

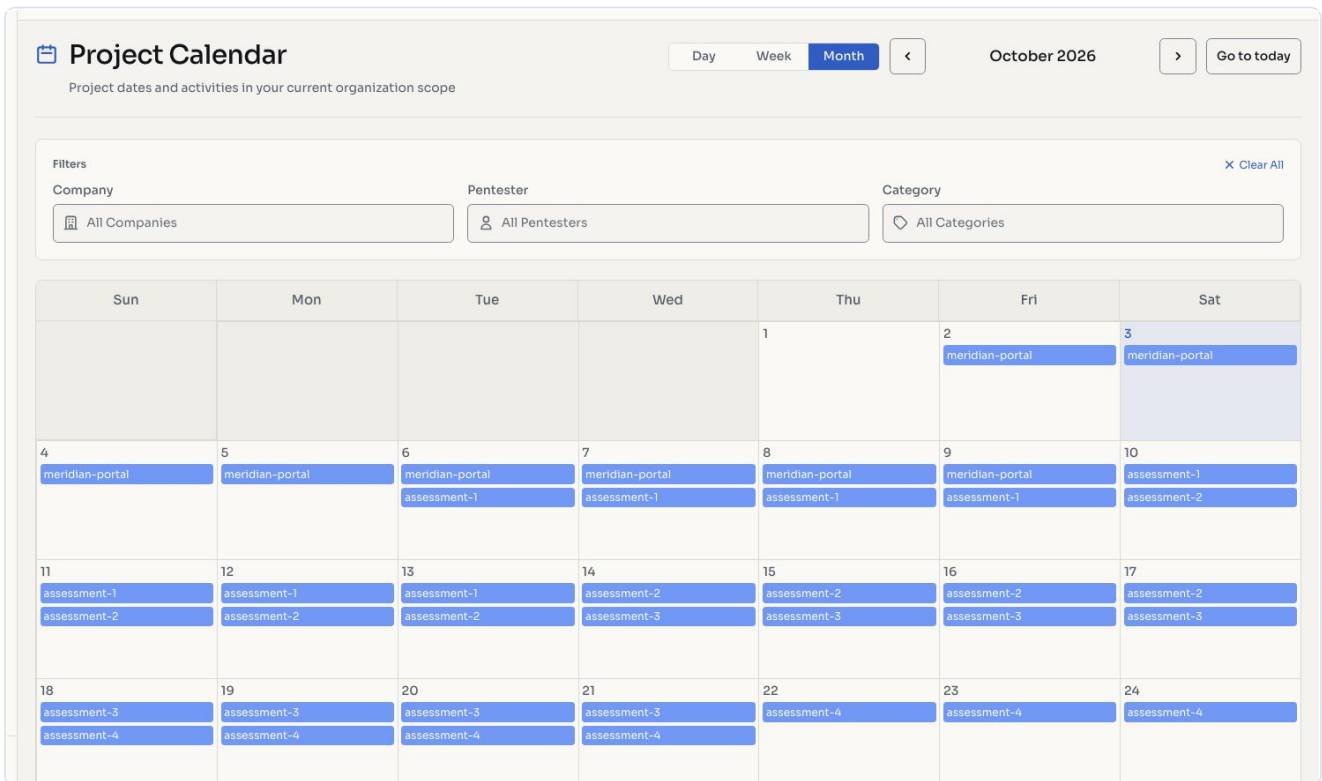


Figure 46 / Project calendar month view

Views

View	Best for
Month	Seeing project distribution and planning ahead
Week	Day-to-day coordination
Day	Checking what's scheduled on a specific date

Navigation

- **Previous / Next:** move forward or backward by one period
- **Today:** jump to the current date

Filters

Providers can filter the calendar to focus on specific subsets:

Filter	Description
Company	Show projects for a specific client
Pentester	Show projects assigned to a specific team member
Category	Filter by project category or type

Project Display

Projects appear as spans covering their start-to-end date range. Each entry shows the project name and gives a visual indicator of duration and overlap with other engagements.

02.35

Templates

Templates in Hashiro provide reusable, standardized content for vulnerability reporting, automated scanning, and PDF report generation. They improve consistency and reduce time spent on repetitive documentation.

Template Types

Vulnerability Templates

Pre-built finding templates with standardized titles, descriptions, CVSS scores, severity ratings, and remediation guidance. Use them when creating new findings for consistent, high-quality write-ups.

Nuclei Templates

YAML-based scanning definitions that power automated vulnerability detection in AI assessments. Define custom checks for organization-specific requirements.

Report Templates

LaTeX-based layouts that control the format and structure of generated PDF pentest reports. Configure headers, footers, branding, sections, and styling to match your organization's deliverables.

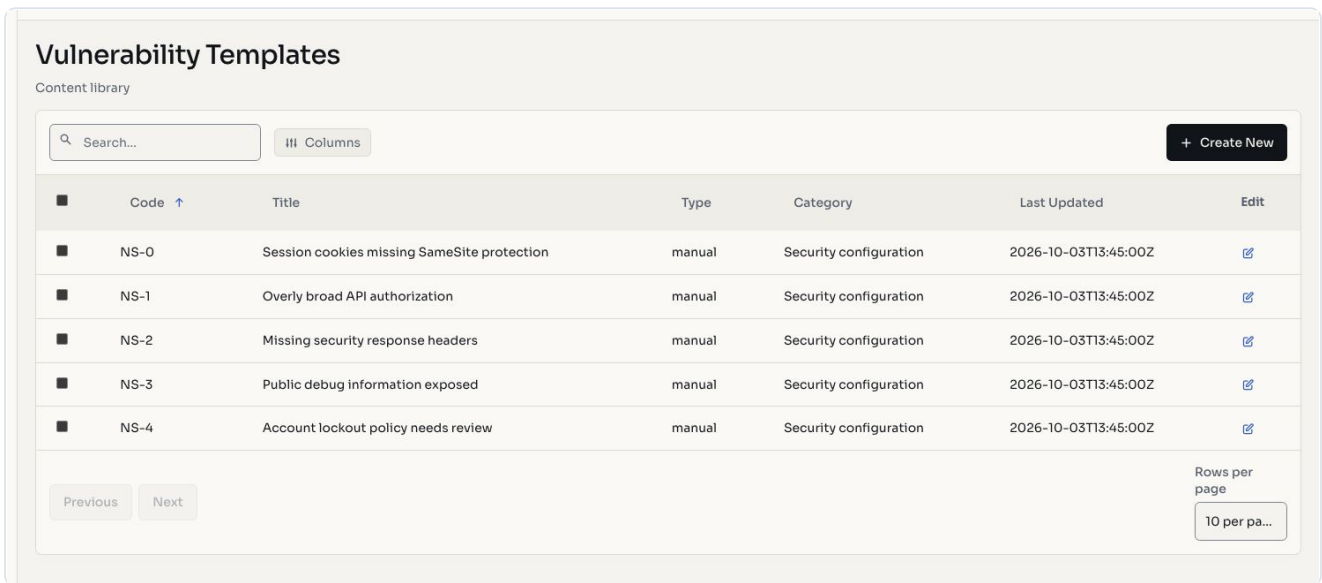
Benefits

- **Consistency:** every finding follows the same structure and quality standard
- **Efficiency:** pre-filled content reduces writing time per finding
- **Standardization:** organizational knowledge is codified and reusable
- **Branding:** reports match your organization's visual identity

02.36

Vulnerability Templates

Vulnerability templates are reusable finding definitions that pre-fill common fields when creating new issues. They codify your organization's knowledge about known vulnerability types into a searchable library.



Code	Title	Type	Category	Last Updated	Edit
NS-0	Session cookies missing SameSite protection	manual	Security configuration	2026-10-03T13:45:00Z	Edit
NS-1	Overly broad API authorization	manual	Security configuration	2026-10-03T13:45:00Z	Edit
NS-2	Missing security response headers	manual	Security configuration	2026-10-03T13:45:00Z	Edit
NS-3	Public debug information exposed	manual	Security configuration	2026-10-03T13:45:00Z	Edit
NS-4	Account lockout policy needs review	manual	Security configuration	2026-10-03T13:45:00Z	Edit

Figure 47 / Vulnerability template library

What's in a Template

Field	Description
Title	Standardized vulnerability name
Description	Detailed explanation of the vulnerability
Severity	Default severity rating
CVSS v3.1 Score	Pre-calculated CVSS vector and score
Remediation	Recommended fix or mitigation steps
Category	Classification (e.g., injection, authentication, configuration)
References	Links to relevant standards, CVEs, or documentation

Template Library

The template library provides:

- **Search:** find templates by keyword or title
- **Filter:** narrow by severity, category, or other fields
- **Column selector:** customize which fields are visible
- **Pagination:** browse large collections efficiently

Creating Templates

Manual Creation

Create templates from scratch by filling in each field. Useful for organization-specific vulnerability types.

AI-Assisted Generation

Describe a vulnerability in plain text, and Hashiro's AI generates a complete template including title, description, CVSS scoring, and remediation guidance.

AI-assisted generation is a fast way to seed your template library. Generate a draft, then refine the details to match your standards.

Using Templates

When creating a new finding in a project:

1. Select a template from the library
2. Fields are pre-filled with the template's content
3. Customize any field for the specific instance (e.g., add target-specific evidence)
4. Save the finding

02.37

Nuclei Templates

Nuclei templates are YAML-based scanning definitions that power automated vulnerability detection across the platform. They define what to check, how to check it, and how to classify the results.

What Are Nuclei Templates

[Nuclei](#) is an open-source vulnerability scanner that uses YAML templates to define detection logic. Hashiro integrates Nuclei for automated scanning in [AI Assessments](#).

Each template defines:

- **Request:** what HTTP request (or other protocol interaction) to send
- **Matchers:** conditions that indicate a vulnerability is present
- **Extractors:** data to pull from responses for evidence
- **Severity:** classification of the finding
- **Metadata:** CVE references, tags, descriptions

Custom Templates

Hashiro includes a catalog of built-in Nuclei templates. You can also create **custom templates** for:

- Proprietary application vulnerabilities
- Internal configuration standards
- Custom compliance checks
- Application-specific business logic tests

YAML Editor

The Hashiro UI includes a YAML editor with syntax highlighting and validation for writing Nuclei templates.

Nuclei Templates

Content library

Show Column Selector

Create New

Search...

English

NAME	SEVERITY	STATUS	LAST UPDATED	EDIT
Missing security response headers	medium	published	2026-10-03T13:45:00Z	✎
Overly broad API authorization	high	published	2026-10-03T13:45:00Z	✎
Session cookies missing SameSite protection	high	published	2026-10-03T13:45:00Z	✎

Previous

Next

10 per page

Figure 48 / Nuclei template library

Importing Templates

Import existing templates from the public Nuclei template repository, your internal collections, or third-party sources.

How Templates Are Used

In AI Assessments

AI agents can trigger Nuclei scans as part of their pipeline. The agent selects relevant templates based on discovered technologies.

Results

Nuclei scan results appear as findings in the project's issues list with severity badges and template metadata attached. They can be reviewed and managed like any other finding.

02.38

Report Templates

Report templates control the format and layout of PDF pentest reports generated from Hashiro projects. They use LaTeX to produce professional, branded deliverables.

How Report Generation Works

When you generate a report from a project, Hashiro:

1. **Collects** all project data: scope, findings, CVSS scores, attack chains, narrative, and remediation guidance
2. **Applies** the selected report template to structure and format the content
3. **Renders** the LaTeX template into a PDF document
4. **Delivers** the finished PDF for download

Template Editor

Report templates are authored in LaTeX using a built-in editor:

- **CodeMirror editor:** syntax-highlighted LaTeX editing with line numbers and code folding
- **Live PDF preview:** see rendered output as you edit, with sample data
- **Template variables:** placeholders replaced with real project data at generation time

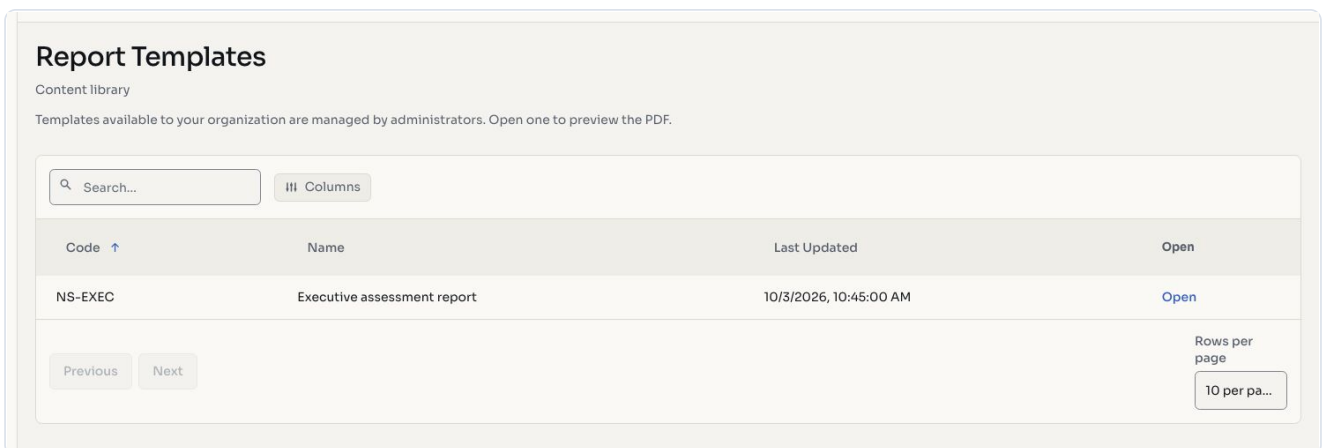


Figure 49 / Report template library

Multi-Language Support

Templates support multiple output languages:

- **English (en)**
- **Portuguese (pt)**

The project's language setting determines which variant is used. Section headers, labels, and boilerplate text adapt to the selected language.

Access Grants

Templates can be assigned to specific organizations, enabling:

- **White-label reports:** different clients receive reports with different branding
- **Compliance-specific formats:** templates tailored to regulatory requirements
- **Tiered service levels:** premium templates for higher-tier clients

Generating Reports

From any project, select a report template and initiate generation. Small projects may generate synchronously for immediate download. Larger projects generate asynchronously: the report is queued, processed by a worker, and made available for download once complete.

You can check the report status and download it once ready. Generated reports are stored securely and can be re-downloaded at any time.

02.39

Share Links

Share links provide **public, unauthenticated access** to project findings or individual issues. Use them to share security assessment results with stakeholders who don't have a Hashiro account.

What Can Be Shared

Type	Content
Project	All findings within a project, including severity, status, and details
Individual Issue	A single vulnerability finding with its full detail

Creating a Share Link

1. Navigate to the project or issue you want to share
2. Create a new share link with:
 - **Label:** a descriptive name (e.g., "Q3 Report for Stakeholders")
 - **Expiry:** when the link should stop working
3. Copy the generated URL

The link can be sent to anyone. No login is required to view the shared content.

Security

- **Token-based:** each link contains a unique, cryptographically generated token
- **Hashed storage:** tokens are stored as hashes in the database, not in plain text
- **Expiry:** links automatically expire after the configured date
- **Revocable:** links can be revoked at any time, immediately disabling access
- **Read-only:** shared content is strictly view-only

Share links provide unauthenticated access to potentially sensitive vulnerability data. Use expiry dates and revoke links when no longer needed.

Managing Share Links

From the management view, you can:

- **List** all active share links for a project or issue
- **Revoke** links that are no longer needed
- **View** labels and expiry dates



PART 03

Organization & teams

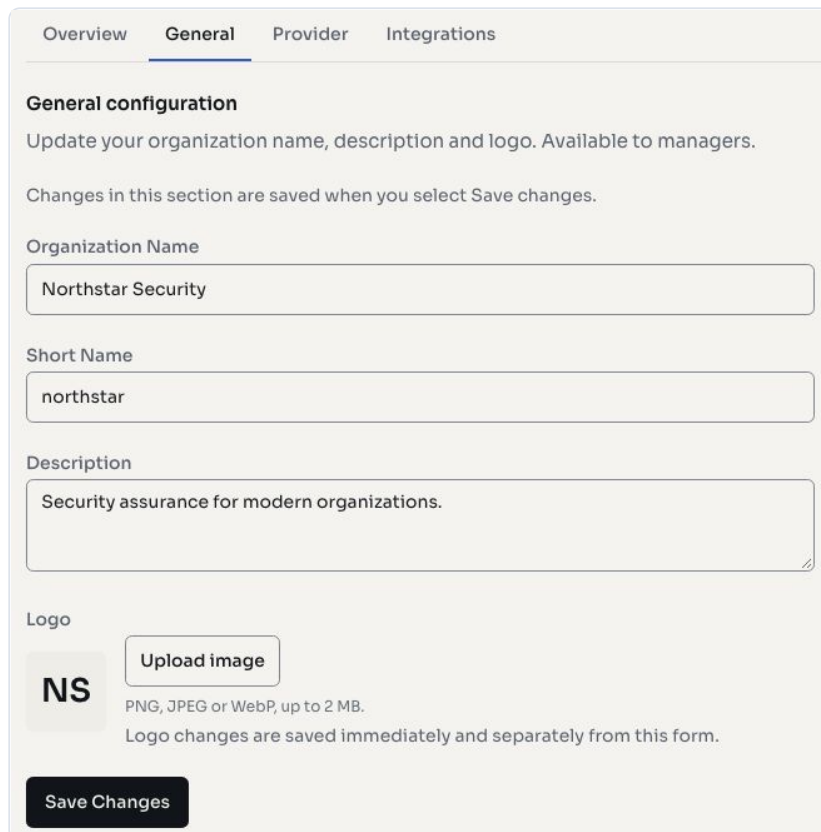
Manage people, access, customer relationships, and integrations.

CHAPTERS / 08

03.01

Organization Management

Every user in Hashiro belongs to an **organization**, the fundamental unit of tenancy, billing, and access control. All data (assets, findings, projects, configurations) is scoped to an organization and isolated from other tenants.



Overview **General** Provider Integrations

General configuration

Update your organization name, description and logo. Available to managers.

Changes in this section are saved when you select Save changes.

Organization Name

Northstar Security

Short Name

northstar

Description

Security assurance for modern organizations.

Logo

NS

Upload image

PNG, JPEG or WebP, up to 2 MB.

Logo changes are saved immediately and separately from this form.

Save Changes

Figure 50 / Organization settings page

Organization Types

Hashiro supports three organizational models:

Provider Organizations

Security firms, MSSPs, and consultancies that deliver services to multiple clients. Providers can:

- Provision and manage **client organizations**
- Access client data through established relationships
- View aggregated metrics across all clients via the **Estate Dashboard**
- Manage shared templates and configurations
- Define custom project types

Client Organizations

Organizations managed by a provider. Clients have their own isolated workspace but are linked to their provider for service delivery. Clients can:

- View and interact with their own projects, assets, and findings
- Manage their own team members (within provider-set limits)
- Request account deletion through a formal process

Direct Organizations

Standalone organizations using Hashiro independently, without a provider relationship. They have full access to all platform features within their license tier.

Organization Settings

Access organization settings from the sidebar under **Organization**.

General

- **Name:** your organization's display name
- **Short name:** abbreviated identifier used in URLs and references
- **Description:** optional organization description
- **Logo:** upload a custom logo (displayed in the sidebar and reports)
- **Default language:** set the default language for the organization (English or Portuguese)

Security

- **OTP policy:** when enabled, all organization members are required to configure multi-factor authentication
- **Email domain restriction:** limit user invitations to specific email domains

License Tiers

Your organization's license determines which features are available and at what scale.

Tier	Description
Free Business	Basic features with limited asset counts
Small Business	Project management for smaller teams
Medium Business	Increased capacity and contracted features
Custom Business	Enterprise features with custom limits
Provider	Multi-client management with estate dashboards

Tier	Description
Hunter	Bug bounty and independent researcher features

License limits include:

- Maximum number of users
- Maximum EASM assets
- Maximum internal projects
- Access to specific modules (EASM, Cloud, AI Assessments)

Contact your administrator or Hashiro sales to upgrade your license tier or adjust limits.

Creating an Organization

- **Providers** can create client organizations from the **Customers** page
- **Staff** can create any organization type from the admin panel
- Account activation follows an invitation or the platform's provisioning process

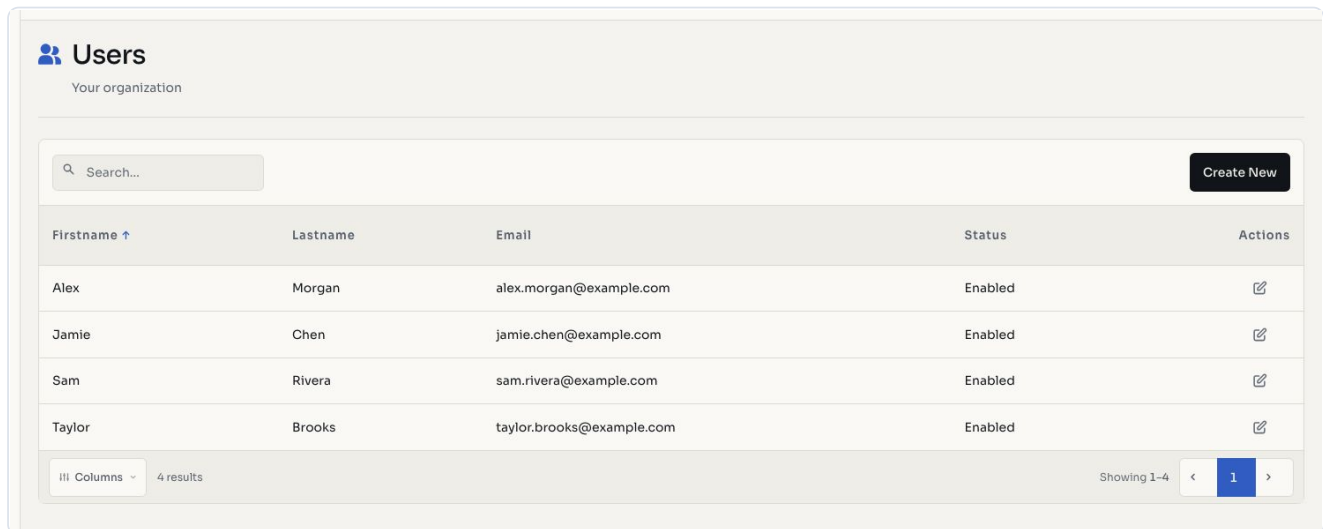
Related

- [Team & Users](#): manage members
- [Roles & Permissions](#): configure access control
- [Customers](#): provider client management
- [Settings](#): organization configuration

03.02

Team & Users

Manage the people who have access to your Hashiro organization. Invite team members, assign roles, and configure individual account settings.



Firstname ↑	Lastname	Email	Status	Actions
Alex	Morgan	alex.morgan@example.com	Enabled	
Jamie	Chen	jamie.chen@example.com	Enabled	
Sam	Rivera	sam.rivera@example.com	Enabled	
Taylor	Brooks	taylor.brooks@example.com	Enabled	

Figure 51 / Team members list

Viewing Your Team

Navigate to **Team** in the sidebar to see all members of your organization. The team list shows:

- **Name** and **email** for each member
- Current **role**
- Account **status** (active, invited, provisional)

Inviting Users

Administrators and managers can invite new users to the organization.

1. Go to **Team**
2. Click **Add User**
3. Enter the user's email address and full name
4. Select a **role**
5. Send the invitation

The invited user receives an email with a link to set their password and activate their account.

Email domain restrictions If your organization has email domain restrictions configured, invitations can only be sent to email addresses matching the allowed domains.

Invitation Management

- **Resend invite:** re-send the invitation email if the user hasn't received or has lost the original
- **Revoke invite:** cancel a pending invitation before the user has accepted it

User Statuses

Status	Description
Active	User has accepted their invitation and can sign in
Invited	Invitation sent, not yet accepted
Provisional	Account created by an administrator, awaiting first login

Managing Users

Edit User

Update a user's name, email, or role. Role changes take effect immediately on the user's next API call.

Reset Password

Administrators can trigger a password reset for any user in the organization. The user will be required to set a new password on their next login.

Remove User

Remove a user from the organization. This revokes their access immediately. The user's historical actions (finding edits, comments, audit trail entries) are preserved.

My Account

Every user can manage their own account from the **My Account** page, accessible from the user menu in the top-right corner.

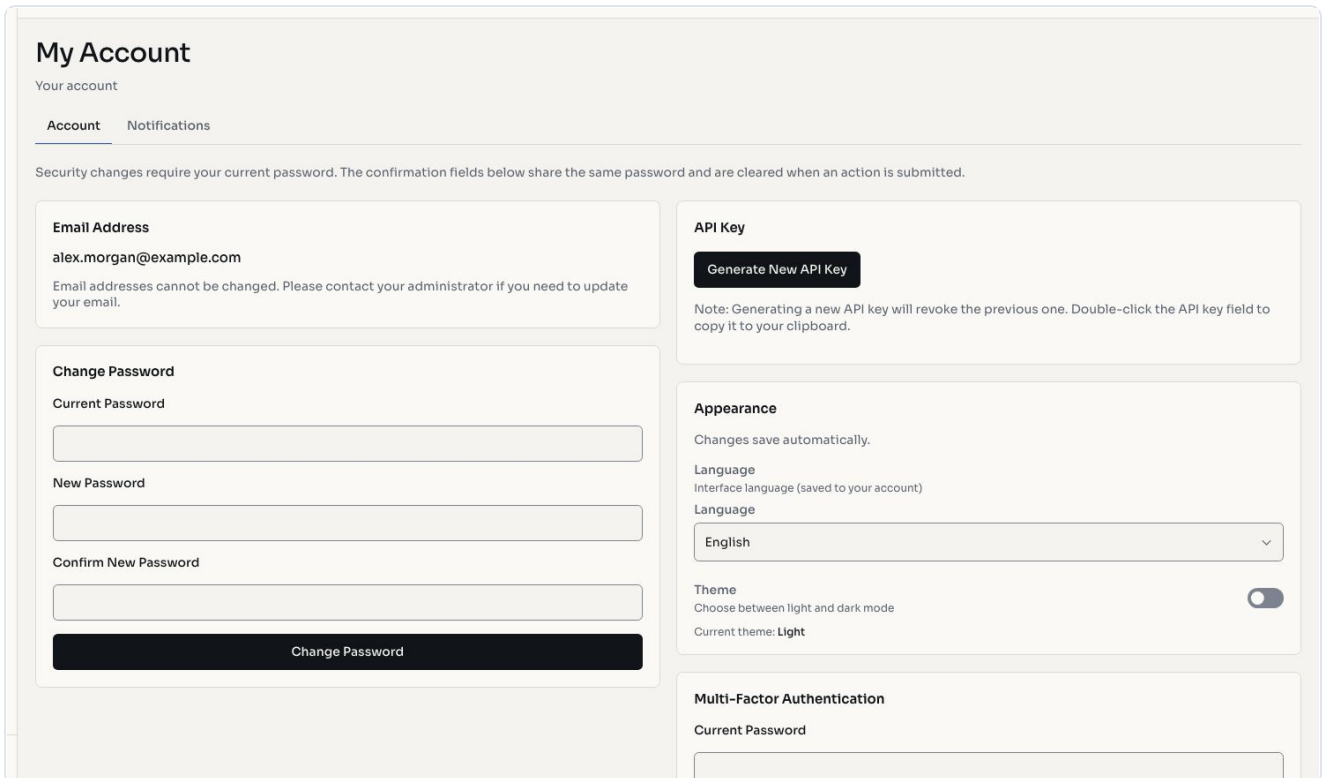


Figure 52 / My Account page

Profile

- View your email address and organization
- Update your display name

Security

- **Change password:** update your password at any time
- **MFA:** enable, disable, or reset your multi-factor authentication setup

API Key

Generate an API token for programmatic access. See [Authentication](#) for usage details.

API keys are shown only once at generation time. Store them securely. Generating a new key invalidates the previous one.

Appearance

- **Theme:** toggle between dark and light mode
- **Language:** switch between English and Portuguese

Notifications

Configure your notification preferences from the dedicated tab. See [Notifications](#) for details.

03.03

Roles & Permissions

Hashiro checks role permissions and organization scope independently. Every user has one organization role; project membership adds the corresponding project access.

Roles

Platform administrator

administrator is reserved for Hashiro platform staff. Global operations require both the reserved staff license and the relevant Administration permission.

Manager

manager administers the provider or client's own organization, team, templates, and authorized projects. It does not grant global administration.

Pentester

pentester works within explicitly assigned projects. Read and write capabilities depend on the permissions assigned inside the project.

Hunter

hunter supports independent research and bug bounty workflows within its license and permitted project and template access.

Viewer

viewer reads explicitly assigned projects without management permissions.

Permission model

Permissions use domain-specific Create, List, Update, and Delete capabilities. Projects also use AllProjects and project membership.

Domain	Examples
Projects	Scope, members, and details
Assets	Inventory, URLs, and services
Findings	Documentation, severity, and status
Templates	Finding and report models
Users	Invitations, roles, and removal
Organization	Settings, integrations, and logo

Domain	Examples
AI Assessments	Authorized runs and configuration

All projects

AllProjects allows access to authorized projects without individual membership. Managers have this capability; pentesters and viewers require explicit membership.

Providers and clients

The organization's license determines data scope. Providers can access their own organization and linked clients where the operation permits it. Clients are confined to their own organization. The same role does not imply the same data scope.

Provider-managed projects keep configuration writes under provider control. Client-managed projects allow the client's own authorized team to manage the assessment.

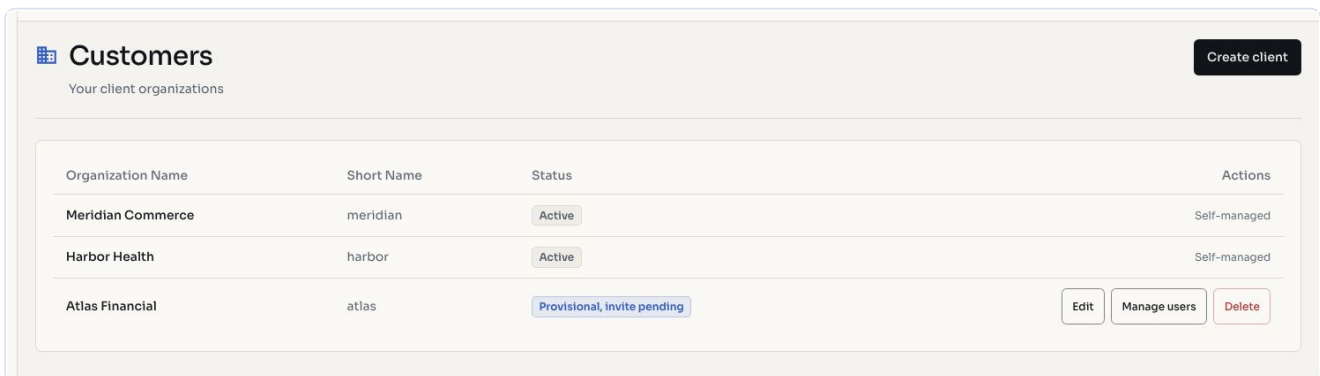
Role selection Use Manager for organization administration, Pentester for assessment work in assigned projects, and Viewer for stakeholders who need to read results. Check project permissions as well as the organization role.

03.04

Customer Management

Provider-only feature Customer management is available to **provider** organizations (security firms and MSSPs that manage security services for multiple client organizations).

Providers use the **Customers** page to provision, manage, and monitor their client organizations within Hashiro.



The screenshot shows the 'Customers' page in Hashiro. At the top left is the 'Customers' header with a grid icon and the subtitle 'Your client organizations'. At the top right is a 'Create client' button. Below this is a table with columns: Organization Name, Short Name, Status, and Actions. The table lists three organizations: Meridian Commerce (Active, Self-managed), Harbor Health (Active, Self-managed), and Atlas Financial (Provisional, invite pending, with Edit, Manage users, and Delete buttons).

Organization Name	Short Name	Status	Actions
Meridian Commerce	meridian	Active	Self-managed
Harbor Health	harbor	Active	Self-managed
Atlas Financial	atlas	Provisional, invite pending	Edit Manage users Delete

Figure 53 / Customer list

Viewing Customers

Navigate to **Customers** in the sidebar to see all your client organizations. Each customer entry shows:

- Organization name and logo
- Account status (active, provisional, deletion-requested)
- Number of users
- Creation date

Provisioning a New Customer

1. Go to **Customers** and click **New Customer**
2. Enter the client organization's details:
 - **Organization name** and **short name**
 - **Contact email** for the initial administrator
 - **Contact name**
3. Submit the form

This creates the client organization and sends an invitation email to the contact. The initial user receives the client organization's management role.

Customer Lifecycle



Status	Description
Provisioned	Organization created, invitation pending
Invitation Sent	Email sent to the initial contact
Claimed	Contact has accepted the invitation and set their password
Active	Organization is fully operational
Deletion Requested	Client has requested account deletion (awaiting processing)

Before the Invitation is Claimed

While a client organization is still provisional (unclaimed), providers can:

- Resend the invitation email
- Delete the provisional organization entirely

Once the invitation is claimed, the organization becomes active and can only be removed through the deletion request process.

Managing Customer Organizations

Users

Provider access to client-user management depends on the client's provisional state and the caller's permissions. Invitation and provisional-password controls apply before the organization is claimed. A claimed client manages its own credentials.

Settings

Review authorized client details and contracted limits. Editing depends on the operation, permissions, and organization ownership.

Access

Inspect projects, assets, and findings for explicitly linked clients. A provider relationship does not grant unrestricted access to every per-organization cloud connection or configuration.

Client data isolation is maintained at the platform level. Providers can only access client organizations they have an explicit relationship with. There is no cross-tenant data access between unrelated organizations.

Deletion Requests

Clients can request the deletion of their own organization from the organization settings page. When a deletion request is submitted:

1. The request enters a **review queue**
2. Platform staff reviews and approves or rejects the request
3. Approved deletions enter a **retention window** before permanent removal
4. During retention, the organization can be restored if needed

This process ensures that no organization is accidentally or maliciously deleted without proper review.

03.05

Notifications

Hashiro keeps you informed about important events through **in-app notifications** and **email alerts**, with granular control over what you receive and how.

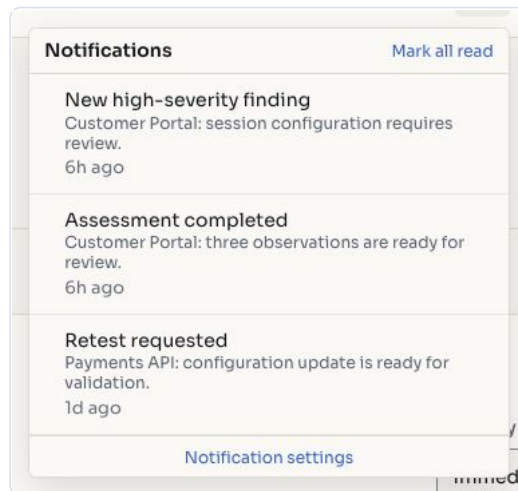


Figure 54 / Notification bell and panel

In-App Notifications

The notification bell icon in the top navigation bar shows your unread notification count. Click it to open the notification panel.

From the panel you can:

- View your recent notifications
- Click a notification to navigate to the related resource
- **Mark all as read** to clear the unread count
- Mark individual notifications as read
- Delete notifications you no longer need

Notification Categories

Notifications are organized into three categories:

Security

Events related to your security operations:

- Scan completed or failed
- New finding created
- Severity escalated on an existing finding
- Candidate findings requiring review

Account

Events related to team and organization management:

- Team member accepted an invitation
- New member added to the organization
- User role changed
- AI credits granted
- Payment failed

System

Platform-wide announcements and operational notices:

- Platform announcements
- Scheduled maintenance windows
- Digest summaries

Email Notification Preferences

Configure how and when you receive email notifications from the **Notification Settings** tab in your account.

In-app notifications are always on. Choose how you'd like to receive email for each category.

Email delivery
In-app notifications are always on and can't be turned off.

Security
Scans and findings.

Account & organization
Invites, membership, roles, and credits.

System & summaries
Announcements, maintenance, and digests.

Security
Immediate ▾

Account & organization
Immediate ▾

System & summaries
Digest ▾

Digest

Frequency: Daily ▾

Send at: 09:00 ▾

Timezone: America/Sao_Paulo

Digest hour uses this timezone: America/Sao_Paulo.

Changes saved.

Figure 55 / Notification settings

Per-Category Settings

For each notification category, choose your preferred delivery mode:

Mode	Behavior
Immediate	Receive an email as soon as the event occurs
Digest	Batch notifications and receive a summary email at your preferred schedule
Off	No email notifications for this category (in-app only)

Digest Configuration

When using digest mode:

- **Frequency:** choose daily or weekly digest delivery
- **Delivery hour:** select what time of day you'd like to receive your digest
- **Timezone:** set your timezone to ensure accurate delivery scheduling

Digest mode is useful for reducing email noise while still staying informed. Critical security events can remain on "Immediate" while account and system notifications are batched.

One-Click Unsubscribe

Every notification email includes an unsubscribe link at the bottom. Clicking it will disable email notifications for that category without requiring you to sign in.

Recommendations

Scenario	Security	Account	System
Active pentester	Immediate	Digest	Off
Team manager	Immediate	Immediate	Digest
Stakeholder / viewer	Digest	Off	Off
Automation account	Off	Off	Off

03.06

Settings

The Settings page serves as a central hub for configuring your Hashiro experience, with quick links to all configuration areas.

Appearance

Theme

Toggle between **dark mode** and **light mode**. Your preference is saved per-device and applies immediately across the entire console.

Language

Switch the interface language between **English** and **Portuguese**. This setting affects all UI labels, messages, and system-generated content.

Account Settings

Quick links to manage your personal account:

- [My Account](#): profile, password, API key, MFA
- [Notification Settings](#): email delivery preferences per category

Organization Settings

Quick links to manage your organization (administrator access required):

- [Organization](#): name, logo, description, language, security policies
- [Team & Users](#): manage members and roles
- [Integrations](#): configure external service connections
- [License](#): view your current license tier and limits

03.07

Integrations

Hashiro connects with external tools and services to fit into your existing security workflows. Integrations are managed at the organization level by administrators and managers.

Available Integrations

Jira

Bidirectional synchronization between Hashiro findings and Jira issues. Push vulnerability findings to your Jira projects, receive status updates via webhooks, and keep both systems in sync.

[Learn more about the Jira integration ->](#)

Managing Integrations

Adding an Integration

1. Navigate to **Organization -> Integrations**
2. Click **Add Integration**
3. Select the integration type
4. Enter the required credentials and configuration
5. Click **Test Connection** to verify connectivity
6. Save the integration

Testing Connections

After creating an integration, you can test the connection at any time to verify that credentials are still valid and the external service is reachable. This is useful after credential rotations or service changes.

Editing an Integration

Update credentials, URLs, or configuration options for an existing integration. Changes take effect immediately.

Removing an Integration

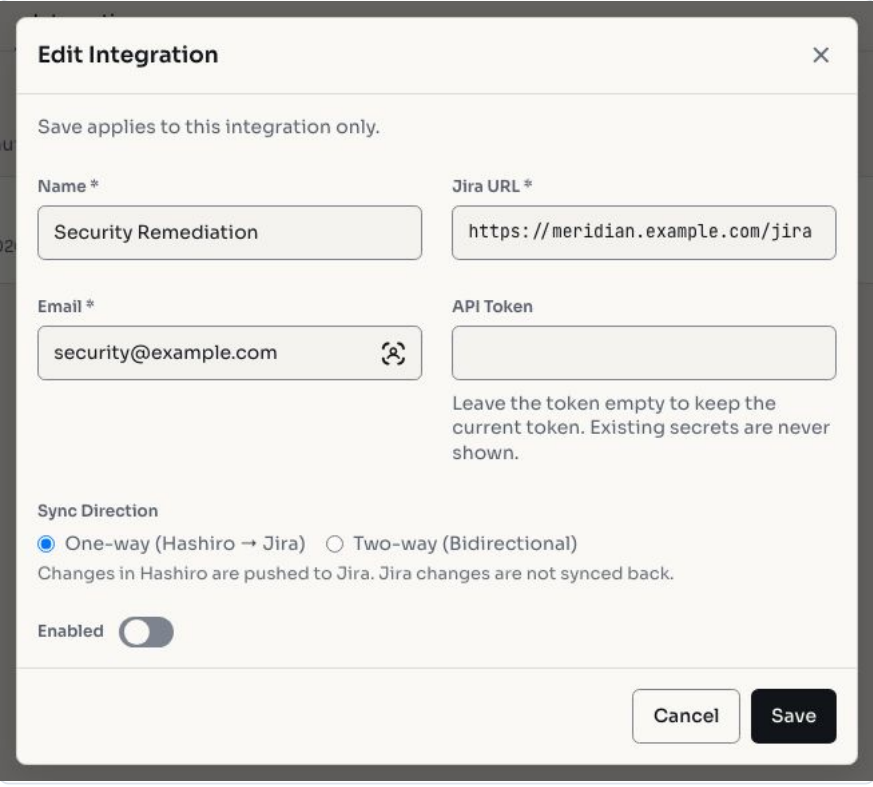
Delete an integration to remove the connection entirely. Existing sync links between Hashiro findings and external issues will be preserved for historical reference but will no longer be actively synchronized.

Removing an integration does not delete any data in the external service. It only disconnects the synchronization.

03.08

Jira Integration

The Jira integration enables **bidirectional synchronization** between Hashiro vulnerability findings and Jira issues. Security teams can push findings to development teams' Jira projects and receive status updates without leaving either tool.



The screenshot shows the 'Edit Integration' dialog box for Jira. It has a title bar with a close button (X). Below the title bar, it says 'Save applies to this integration only.' The form contains several fields: 'Name *' with the value 'Security Remediation', 'Jira URL *' with the value 'https://meridian.example.com/jira', 'Email *' with the value 'security@example.com' and a copy icon, and 'API Token' which is empty. Below these fields is a note: 'Leave the token empty to keep the current token. Existing secrets are never shown.' There is a 'Sync Direction' section with two radio buttons: 'One-way (Hashiro → Jira)' (selected) and 'Two-way (Bidirectional)'. Below this is a note: 'Changes in Hashiro are pushed to Jira. Jira changes are not synced back.' At the bottom left is an 'Enabled' toggle switch, which is currently turned off. At the bottom right are 'Cancel' and 'Save' buttons.

Figure 56 / Jira integration setup

Setup

Prerequisites

- A Jira Cloud or Jira Server instance
- An API token (Jira Cloud) or credentials with project access
- Administrator or manager role in Hashiro

Creating the Connection

1. Navigate to **Organization > Integrations**
2. Click **Add Integration** and select **Jira**
3. Enter your connection details:
 - **Jira URL:** your Jira instance URL (e.g., `https://yourteam.atlassian.net`)
 - **Email:** the email associated with your Jira account

- **API Token:** a Jira API token generated from your Atlassian account settings
4. Click **Test Connection** to verify
 5. Save the integration

Generating a Jira API token In Jira Cloud, go to **Account Settings > Security > Create and manage API tokens**. Create a new token and copy it into Hashiro.

Project Discovery

Once connected, Hashiro can discover your Jira projects and their metadata:

- **Projects:** all Jira projects accessible with the configured credentials
- **Issue types:** available issue types per project (Bug, Task, Story, etc.)
- **Priorities:** priority levels defined in your Jira instance
- **Custom fields:** any custom fields configured in your Jira projects

Field Mapping

Configure how Hashiro fields map to Jira fields for each project.

Creating a Mapping

1. Go to the integration detail page
2. Click **Add Mapping**
3. Select the **Jira project** and **issue type**
4. Map Hashiro fields to Jira fields:
 - Finding title > Jira summary
 - Finding description > Jira description
 - Severity > Jira priority
 - Custom field mappings as needed
5. Save the mapping

You can create multiple mappings for different Jira projects or issue types.

Editing and Removing Mappings

Update field mappings at any time to reflect changes in your Jira project configuration. Deleting a mapping stops future syncs but preserves existing links.

Synchronization

Pushing Findings to Jira

Findings can be synced to Jira in two ways:

- **Full sync:** synchronize all mapped findings at once

- **Single-entity sync:** push an individual finding to Jira on demand

When a finding is synced, Hashiro creates a Jira issue using the configured field mapping and stores a **sync link** between the two records.

Retest Comments

When a finding enters the retesting phase in Hashiro, you can push a **retest comment** to the linked Jira issue, notifying the development team of the retest status.

Webhook Sync (Jira to Hashiro)

Set up webhooks to receive real-time updates from Jira:

1. Go to the integration's **Webhooks** tab
2. Click **Create Webhook**
3. Hashiro generates a unique webhook URL
4. Configure this URL in your Jira project's webhook settings

When Jira issues are updated (status changes, comments, etc.), the webhook notifies Hashiro, and the corresponding finding is updated automatically.

Webhook security Each webhook URL contains a unique identifier. Do not share webhook URLs publicly. You can regenerate or delete webhooks at any time.

Monitoring

Sync Status

View the current synchronization state from the integration dashboard:

- Total synced entities
- Last sync time
- Any sync errors

Sync History

Review a log of all synchronization events: pushes, pulls, webhook receipts, and errors. Useful for troubleshooting sync issues.

Sync Links

View all active links between Hashiro findings and Jira issues. You can remove individual links to disconnect specific findings from their Jira counterparts.

Per-Project Configuration

In addition to organization-level Jira configuration, individual Hashiro projects can have their own Jira settings, specifying which Jira project and mapping to use when syncing findings from that particular engagement.



PART 04

Galileo CLI

Install, configure, and use the command-line reference.

CHAPTERS / 05

04.01

Galileo CLI

Galileo is Hashiro's command-line interface for projects, scope, assets, URLs, services, findings, and templates.

Both providers and clients connect to <https://api.hashiro.ai/api/> using one configured profile and their own API token. Commands obey the same organization and project permissions as the console.

First steps

1. [Install Galileo](#) for macOS, Linux, or Windows.
2. [Configure your connection](#).
3. List the projects you can access:

```
galileo projects list
galileo projects default my-project
galileo assets list
galileo scan list --json
```

Provider and client workflows

Providers use Galileo to maintain assessment data for authorized projects and linked clients. Clients use it to inspect their own data and manage their client-owned projects where permitted. A CLI command does not bypass project ownership or access restrictions.

Capabilities

- Project creation, default selection, scope, narrative, credentials, and context.
- Asset, URL path, and service records.
- Finding documentation and structured JSON output.
- Reusable vulnerability templates.
- HackerOne and Intigriti imports.
- Optional proxy and automatic binary updates.

See the [command reference](#) for syntax, flags, inputs, outputs, and limitations, and [imports](#) for platform-specific credentials.

04.02

Installation

Galileo is available for macOS, Linux, and Windows. Everyone downloads it from <https://api.hashiro.ai/api/>.

Check the release

```
curl -fSL https://api.hashiro.ai/api/galileo/latest
```

The JSON response contains version and a downloads map. Architecture keys use underscores; the download is a ZIP archive, not a raw executable. Use the architectures published by the current release.

System	Download key	Executable inside ZIP
macOS Apple Silicon	darwin_arm64	galileo-darwin-arm64
macOS Intel	darwin_amd64	galileo-darwin-amd64
Linux x86-64	linux_amd64	galileo-linux-amd64
Linux ARM64	linux_arm64	galileo-linux-arm64
Windows x86-64	windows_amd64	galileo-windows-amd64.exe

macOS

Run `uname -m`: `arm64` means Apple Silicon, while `x86_64` means Intel.

For Apple Silicon:

```
curl -fSL https://api.hashiro.ai/api/galileo/download/darwin_arm64 -o galileo.zip
unzip -o galileo.zip -d galileo-download
mkdir -p "$HOME/.local/bin"
install -m 755 galileo-download/galileo-darwin-arm64 "$HOME/.local/bin/galileo"
export PATH="$HOME/.local/bin:$PATH"
```

For Intel, use `darwin_amd64` in the URL and install `galileo-darwin-amd64` from the extracted archive. Confirm the archive contents with `unzip -l galileo.zip` if a release uses a different filename.

Add `export PATH="$HOME/.local/bin:$PATH"` to `~/.zshrc` to keep the `PATH` setting in new terminals.

Linux

You need `curl` and `unzip`. Run `uname -m`: use `linux_amd64` for `x86_64` or `linux_arm64` for `aarch64` / `arm64`.

For `x86-64`:

```
curl -fSL https://api.hashiro.ai/api/galileo/download/linux_amd64 -o galileo.zip
unzip -o galileo.zip -d galileo-download
mkdir -p "$HOME/.local/bin"
install -m 755 galileo-download/galileo-linux-amd64 "$HOME/.local/bin/galileo"
export PATH="$HOME/.local/bin:$PATH"
```

For ARM64, replace the download key with `linux_arm64` and the executable name with `galileo-linux-arm64`. Add the PATH export to your shell configuration, such as `~/.bashrc` or `~/.zshrc`.

Windows

Open PowerShell. The published Windows build is x86-64. Use the release metadata to confirm available architectures.

```
Invoke-RestMethod https://api.hashiro.ai/api/galileo/latest
Invoke-WebRequest https://api.hashiro.ai/api/galileo/download/windows_amd64 -OutFile galileo.zip
Expand-Archive -Path galileo.zip -DestinationPath galileo-download -Force
$galileoBin = Join-Path $env:LOCALAPPDATA "Hashiro\bin"
New-Item -ItemType Directory -Force $galileoBin
Copy-Item .\galileo-download\galileo-windows-amd64.exe (Join-Path $galileoBin "galileo.exe")
$env:Path = "$galileoBin;$env:Path"
```

To keep the PATH setting, open **Edit environment variables for your account**, edit the user **Path**, and add the Hashiro\bin directory created under LocalAppData. Open a new terminal after saving.

Install script for macOS and Linux

You can also download and run the production install script:

```
curl -fSL https://api.hashiro.ai/api/install -o install-galileo.sh
bash install-galileo.sh
```

The script detects the platform, downloads and extracts the ZIP, and installs the executable. Windows uses the PowerShell instructions above.

Configure and verify

Create [the configuration file](#) before running Galileo. Then:

```
galileo version
galileo projects list
```

Use `galileo version`, not `galileo --version`. If the command is not found, check PATH. If permission is denied on macOS or Linux, check executable permissions. A 404 download means the architecture is not available in the current release; consult the metadata.

Updating

```
galileo update
```

Automatic updates are enabled by default. Disable them with `no_update = true` or `--no-update`; explicit update still works. The install script can also install the latest release again.

04.03

Configuration

Providers and clients use the same production API, <https://api.hashiro.ai/api/>. Configure one profile for your organization. Your token determines your identity, organization, license, and permissions; a profile name does not grant provider access.

Create the configuration file

Generate an API token in **My Account** in the [Hashiro console](#). Create the following file:

System	File
macOS / Linux	~/.config/galileo/config.toml
Windows	%USERPROFILE%\config\galileo\config.toml

On macOS or Linux:

```
mkdir -p ~/.config/galileo
$EDITOR ~/.config/galileo/config.toml
chmod 600 ~/.config/galileo/config.toml
```

On Windows PowerShell:

```
New-Item -ItemType Directory -Force "$env:USERPROFILE\config\galileo"
notepad "$env:USERPROFILE\config\galileo\config.toml"
```

Paste this configuration and replace the token placeholder:

```
current_project = ""
no_update = false
no_skills_update = true

[[profiles]]
name = "hashiro"
default = true
api_address = "https://api.hashiro.ai/api/"
api_key = "REPLACE_WITH_YOUR_API_TOKEN"
```

Field reference

Field	Meaning
current_project	Default project identifier. Empty until you select a project.
no_update	Set to true to disable automatic binary updates. Default is false.
no_skills_update	Set to true to disable automatic refresh of installed Claude Code command files.

Field	Meaning
[[profiles]]	A TOML array entry. Use this syntax even with one profile.
name	Local label for the connection.
default	true selects this profile by default.
api_address	Always use https://api.hashiro.ai/api/. Galileo normalizes the API suffix.
api_key	API token generated by your user.
proxy	Optional HTTP proxy URL. Omit when not needed.
default_project	Optional project identifier inside a profile; it seeds the current project when the profile loads.

Do not use [profiles.default], url, token, or current_program; those are not the current configuration format.

Provider and client access

A provider token can reach the provider's own data and linked clients where the endpoint permits it. A client token is confined to its own organization. Use the same API address for both. Project membership and project ownership still determine which operations are allowed.

Verify the connection

```
galileo config profile list
galileo config profile default
galileo projects list
```

The profile list marks the default with *. Project listing is a read operation and shows only authorized projects.

Select a default project

```
galileo projects default my-project
galileo projects default
galileo assets list
galileo assets list --project another-project
galileo projects default --unset
```

--project selects a project for a single command. Keep current_project at the top level if you want the default-project command to control it; a profile's default_project can seed it again when configuration loads.

Proxy

```
galileo config proxy set http://127.0.0.1:8080
galileo config proxy unset
```

These commands modify the default profile. The proxy is optional and does not change the API instance or your access rights.

Configuration management

`galileo config set ./config.toml` copies an existing TOML file to the standard path. It is not a key/value setter. Create the standard file before the first invocation because Galileo loads configuration before executing commands.

With one profile, no profile switching is needed. If you later add another identity on the same instance, `galileo config profile default NAME` sets the default and `--profile NAME` overrides it for one invocation.

Updates and troubleshooting

Use `galileo --no-update projects list` to skip the automatic binary update for one invocation. `galileo update` explicitly checks for updates even when automatic updates are disabled.

Problem	Check
No config file found	Create the file at the system-specific path above.
TOML parsing error	Use <code>[[profiles]]</code> , quote string values, and keep global settings before the profile.
Authentication failure	Replace the placeholder with a valid token; a newly generated token invalidates the previous one.
Permission denied	Check your role, license, project membership, and organization relationship.
Wrong project	Pass <code>--project</code> explicitly or review the saved default.

The config file contains a credential. Keep it private and outside version control.

04.04

Command Reference

This reference covers the project API commands available to providers and clients. Your token and project permissions govern each operation. Start with [configuration](#).

Help and global flags

```
galileo --help
galileo assets service --help
galileo --no-update --no-skills-update projects list
```

Flag	Meaning
--help, -h	Show command help.
--profile NAME	Use a named profile for this invocation. One profile is enough for the standard setup.
--no-update	Skip the automatic binary update.
--no-skills-update	Skip refresh of installed Claude Code files.

Use the project identifier from `projects list`. Project-scoped commands accept `--project / -p` or the saved default. **Exception:** `projects create -p` selects the platform, not a project.

Version and update

```
galileo version
galileo update
```

`version` prints the installed version; it may report a newer available release. `update` updates only the binary and ignores automatic-update opt-outs.

Configuration commands

```
galileo config set ./config.toml
galileo config profile list
galileo config profile default
galileo config profile default hashiro
galileo config profile default --unset
galileo config proxy set http://127.0.0.1:8080
galileo config proxy unset
```

`config set` copies a TOML file; it does not accept key/value pairs. Create the standard config first. `profile default` shows the current profile with no argument and changes it when given a name. `--unset` clears its default marker; the first profile is then used.

Project examples

```
galileo projects create my-project --platform pentest --tag owner=security
galileo projects list --all
galileo projects default my-project
galileo projects inscope --project my-project --type domain example.com
galileo projects outscope --project my-project --type domain excluded.example.com
galileo projects inscope --project my-project
galileo projects inscope --project my-project --rm example.com
```

Project aliases: project, program, programs, p. list also accepts ls / l; create accepts c / add; remove accepts rm / r. Quote wildcard scope values to prevent shell expansion.

Narrative, credentials, context, and chains

```
galileo projects narrative --project my-project
galileo projects narrative --project my-project < narrative.md
galileo projects context --project my-project < context.md
galileo projects credentials --project my-project < credentials.md
galileo projects chains --project my-project
```

No content reads the current Markdown field; arguments or stdin replace it. chains is read-only. These operations do not grant a client edit access to a provider-managed project.

Asset, URL, and service examples

```
galileo assets insert --project my-project app.example.com
galileo assets list --project my-project --unavailable --show-status-code
galileo assets view --project my-project app.example.com
galileo assets path --project my-project --add https://app.example.com/health
galileo assets path --project my-project https://app.example.com
galileo assets service --project my-project --asset app.example.com
galileo assets service --project my-project --json
galileo assets service --project my-project --add --asset app.example.com --port 443 --protocol
tcp --state open
```

Asset aliases: asset, domain, domains. path also accepts paths / dir; service accepts services. Asset names are positional for insert/view/remove. There is no assets insert --asset flag or assets path --path flag.

Finding examples

```
galileo scan add --project my-project --asset app.example.com --title "Missing security header"
--severity low --description "Observed during review" --remediation "Configure the required
header"
galileo scan list --project my-project --info
galileo scan list --project my-project --id FINDING_ID
galileo scan list --project my-project --json
```

Finding aliases: scans, issue, issues. Use scan list --id to view a finding; there is no scan view command. scan add --raw accepts one JSON finding object through stdin. --requests-file accepts a JSON array of request/response pairs, such as [{"request text", "response text"}].

Templates and imports

```
galileo templates list --verbose
galileo templates view TEMPLATE_CODE_OR_ID
galileo import hackerone
galileo import intigriti
```

Template aliases: template, t. view accepts show. Imports use separate platform credentials described in [Imports](#).

Notifications

```
galileo notify FINDING_ID --discord
```

Configure [discord] with channel and token. Pass finding IDs as arguments or lines on stdin. The implementation uses those IDs; --id is not consumed. Slack sending is not implemented despite its visible flag.

Optional Claude Code files

```
galileo gen-skills ./claude-config
```

Writes a marked section in CLAUDE.md and slash-command files in commands/. The default directory is ~/.claude. This changes local files; --no-skills-update prevents automatic refresh on later commands.

Command and flag tables

Flags below are grouped by command. Repeatable string-array flags accept separate occurrences, such as --tag owner=security --tag env=test. Integer flags accept numbers; boolean flags can be explicitly set to false.

galileo projects create NAME

Create a project. NAME is positional; -p means platform here.

Flag	Description
--platform VALUE / -p	Specify the project platform (intigriti, bugcrowd, hackerone, pentest...)
--inactive / -i	Specify ito create project as inactive
--type VALUE	Specify the project type, like private or public (specific to bug bounty)
--tag VALUE / -t	Add tags to the project (multiple key=value pairs can be provided)
--raw	Read project as raw JSON from stdin

galileo projects list

List authorized projects.

Flag	Description
--all / -a	Show all projects, including inactive ones
--inactive / -i	Only show inactive projects
--tag VALUE / -t	Show only projects that contain certain tag
--value VALUE / -v	Show only projects that contain a specific tag and value (must be used with -t)

galileo projects remove NAME

Delete a project after confirmation.

Flag	Description
--yes / -y	Skip confirmation prompt

galileo projects default [NAME]

Read or set the saved default project.

Flag	Description
--unset	Unset the default project

galileo projects inscope [TARGET...]

Read, add, or remove in-scope entries.

Flag	Description
--project VALUE / -p	Specify project to add asset to scope
--rm / -r	Remove asset from inscope
--bounty VALUE / -b	Specify if asset is eligible to bounty
--type VALUE	Specify asset type (domain / subdomain / android / ios / source / other)
--wildcard / -w	Show only wildcard domains

Flag	Description
--domains / -d	Show only wildcard domains
--tier VALUE	Specify asset tier (numeric value)
--issues VALUE / -i	Specify asset number of known issues
--tag VALUE / -t	Add tags to the asset (multiple key=value pairs can be provided)

galileo projects outscope [TARGET...]

Read, add, or remove exclusions.

Flag	Description
--project VALUE / -p	Specify project to add asset to out of scope
--bounty VALUE / -b	Specify if asset is eligible to bounty
--rm / -r	Remove asset from out of scope
--type VALUE	Specify asset type (domain / subdomain / android / ios / source / other)
--wildcard / -w	Show only wildcard domains
--domains / -d	Show only wildcard domains
--tier VALUE	Specify asset tier (numeric value)
--issues VALUE / -i	Specify asset number of known issues
--tag VALUE / -t	Add tags to the asset (multiple key=value pairs can be provided)

galileo assets insert HOST...

Insert hostnames from arguments or stdin.

Flag	Description
--project VALUE / -p	Specify project to add asset
--httpx	Automatically perform httpx
--screenshot / -s	Take screenshot with HTTPX
--tag VALUE / -t	Add tags to the asset (multiple key=value pairs can be provided)

Flag	Description
--ipaddress VALUE / -i	IP Addresses available to that asset
--available / -a	If the asset is available (false by default!)
--scheme VALUE	Specify asset protocol
--port VALUE	Port of the available address
--title VALUE	Specify page title probe
--status-code VALUE	Specify status code probe
--content-length VALUE	Specify status code probe
--favicon VALUE	Specify favicon hash probe
--lines VALUE	Specify lines count probe
--words VALUE	Specify words count probe
--location VALUE	Specify location header probe
--webserver VALUE	Specify server header probe
--body VALUE	Specify body hash probe
--jarm VALUE	Specify JARM hash probe
--b64screenshot VALUE	Specify base64 screenshot probe
--cdn VALUE	Specify CDN probe
--header VALUE	Add response headers (multiple key=value pairs can be provided)
--wappalyzer VALUE	Specify wappalyzer probe

galileo assets list

List available assets; include unavailable assets with --unavailable.

Flag	Description
--project VALUE / -p	Specify project to list assets
--domains / -d	Specify project to list only assets subdomains
--unavailable / -u	Get unavailable projects as well

Flag	Description
<code>--show-wappalyzer</code>	Show wappalyzer technologies beside each asset
<code>--show-status-code</code>	Show HTTP status code beside each asset
<code>--show-title</code>	Show page title beside each asset

galileo assets view HOST

Show an asset and its available screenshot.

Flag	Description
<code>--project VALUE / -p</code>	Specify project to list assets

galileo assets remove HOST...

Remove selected assets; `--all` removes every asset in the selected project.

Flag	Description
<code>--project VALUE / -p</code>	Specify project to remove asset
<code>--all</code>	Remove all assets
<code>--yes / -y</code>	Skip confirmation prompt

galileo assets path [URL...]

List paths by default; use `--add`, `--rm`, or `--rm-all` to change them. Pass full URLs.

Flag	Description
<code>--project VALUE / -p</code>	Specify project to list assets
<code>--add</code>	Add path's to the asset
<code>--rm</code>	Remove path's to the asset
<code>--rm-all</code>	Remove all asset paths
<code>--httpx</code>	Automatically perform httpx
<code>--source VALUE / -s</code>	Specify source from this path. (e.g.: ffuf)

Flag	Description
--tag VALUE / -t	Add tags to the asset (multiple key=value pairs can be provided)
--title VALUE	Specify page title probe
--status-code VALUE	Specify status code probe
--content-length VALUE	Specify status code probe
--favicon VALUE	Specify favicon hash probe
--lines VALUE	Specify lines count probe
--words VALUE	Specify words count probe
--location VALUE	Specify location header probe
--webserver VALUE	Specify server header probe
--body VALUE	Specify body hash probe
--jarm VALUE	Specify JARM hash probe
--b64screenshot VALUE	Specify base64 screenshot probe
--cdn VALUE	Specify CDN probe
--header VALUE	Add response headers (multiple key=value pairs can be provided)
--wappalyzer VALUE	Specify wappalyzer probe

galileo assets service [HOST]

List recorded services; --add writes a service record. This command does not perform a port scan.

Flag	Description
--project VALUE / -p	Specify project to list assets
--asset VALUE / -a	Specify asset to list services
--add	Add service to the asset
--list / -l	List services (default when no other action specified)
--port VALUE	Port of the available address
--protocol VALUE	Specify protocol of the available address

Flag	Description
--banner VALUE	Specify service banner/description
--state VALUE	Specify port state: open, closed, filtered
--script VALUE / -s	Add scripts to the asset
--tag VALUE / -t	Add tags to the asset (multiple key=value pairs can be provided)
--hosts	Show all alive hosts
--services	Show services in hostname:port service format
--service VALUE	Filter by service name
--ports	Show consolidated list of open ports
--port-filter VALUE	Filter by port(s) (comma-separated)
--json	Output in JSON format
--csv	Output in CSV format

galileo scan add

Create a finding using flags or one JSON object on stdin with --raw.

Flag	Description
--project VALUE / -p	Specify project to add scan
--notify	Auto notify vulnerabilities when adding them
--severity VALUE	Specify scan severity (default unknown)
--asset VALUE	Affected asset in the issue
--title VALUE	Scan result title
--description VALUE	Scan result description
--notified	Mark if it was already notified (default false)
--fp	Mark if it was a false positive (default false)
--source VALUE	Scan result source
--cve VALUE	Inform issue CVE

Flag	Description
--reference VALUE	Add references to the scan
--tag VALUE	Add tags to the scan (key=value pairs)
--template VALUE	Scan template identifier
--demonstration VALUE	Steps to demonstrate the finding
--observation VALUE	Observation / context
--type VALUE	Scan type (e.g. vulnerability type)
--category VALUE	Scan category
--remediation VALUE	Remediation guidance
--retest-demonstration VALUE	Steps for retest demonstration
--requests-file VALUE	Path to JSON file with request/response pairs: [{"req","resp"},...]
--raw	Read scan as raw JSON from stdin and send to server (project via -p)

galileo scan list

List findings; --id selects one finding and --json produces structured output.

Flag	Description
--project VALUE / -p	Specify project to list scans
--title VALUE	Show scans by title
--id VALUE	Show scan result from specific id
--asset VALUE	Filter by asset
--severity VALUE	Specify severity to view scans
--info	Include informative severity in list
--json	Output findings as a JSON array (id/title/asset/severity/scope/status/source)

galileo scan remove ID...

Delete finding IDs passed as arguments or stdin.

Flag	Description
--project VALUE / -p	Specify project
--yes / -y	Skip confirmation prompt

galileo templates list

List vulnerability templates available to your account.

Flag	Description
--verbose / -v	Show template details (type, category, CVSS)

galileo templates view CODE_OR_ID

View a template by code or ID.

galileo notify ID...

Notify findings by ID through Discord. Slack is not implemented.

Flag	Description
--slack	Notify scan on slack
--discord	Notify scan on discord
--id VALUE	Scan ID to notify

Inputs, outputs, and errors

Text commands accept newline-delimited stdin where described. --raw expects one JSON object, not an array. Use `scan list --json` and `assets service --json` for structured output. Do not assume all output is JSON or every validation error yields a nonzero exit code; inspect error messages. Deletions prompt for confirmation unless --yes is used. Check the project and IDs before bulk actions.

04.05

Imports

Galileo imports authorized bug bounty programs into Hashiro projects and maps their included and excluded targets into scope. It uses your normal Hashiro profile plus separate source-platform credentials.

Prerequisites

Configure [the Hashiro connection](#) and use an account allowed to create projects. Add the relevant platform section to `config.toml`:

```
[hackerone]
username = "YOUR_HACKERONE_USERNAME"
api_key = "YOUR_HACKERONE_API_TOKEN"

[intigrity]
api_key = "YOUR_INTIGRITI_API_TOKEN"
```

These platform credentials are separate from the Hashiro `api_key`. Add only the sections you use.

HackerOne

```
galileo import hackerone
```

Alias: `h1`. Galileo uses your HackerOne username and API token, fetches available programs, creates projects, and imports included and excluded targets. Review the output for per-program errors.

Intigrity

```
galileo import intigrity
```

Alias: `int`. Galileo uses the Intigrity bearer API token, retrieves accessible programs, and imports project information and scope. Programs without usable scope or a program ID may be skipped.

Verify the result

```
galileo projects list
galileo projects inscope --project PROGRAM_NAME
galileo projects outscope --project PROGRAM_NAME
```

Review the imported entries against the current program rules. Imported scope does not replace authorization or the source platform's restrictions. Imports may partially complete; review errors before retrying.

Scope mapping

Domains and wildcard targets become corresponding scope entries. Mobile application and source-code entries retain their platform-specific classification where supported. Review unusual target types after import instead of assuming every external type maps identically.



PART 05

API reference

Connect applications using the public API baseline.

CHAPTERS / 06

05.01

Hashiro API

The baseline API serves **providers and clients** on the same production instance:

```
https://api.hashiro.ai/api/
```

All paths in this reference are relative to that base. For example, `/users/me` means `https://api.hashiro.ai/api/users/me`.

Authenticate

Use your user API token in the `X-HASHIRO-TOKEN` header:

```
curl --fail-with-body \
-H "X-HASHIRO-TOKEN: $HASHIRO_TOKEN" \
https://api.hashiro.ai/api/users/me
```

Set `HASHIRO_TOKEN` in your environment to a token generated in **My Account** in the [console](#). See [Authentication](#).

Provider and client scope

Caller	Authorized data
Provider	Own organization and explicitly linked clients, where the route allows provider access
Client	Own organization, including published results from provider-managed engagements

Role, license, project membership, and project ownership are checked independently. A linked client identifier does not grant unrestricted access. Clients cannot edit provider-owned project configuration through baseline project-write routes. Client-managed projects allow their authorized owners to manage their own assessment data.

Reference

- [Projects and scope](#).
- [Assets, URLs, and services](#).
- [Findings and review](#).
- [Organizations and users](#).

Requests and responses

Send Content-Type: application/json for JSON requests. File endpoints use their documented format, which can be multipart or JSON image data.

Response shapes vary by endpoint. A list can be a JSON array, a detail response an object, and a write response a status message or identifier. There is no universal data wrapper.

Typical errors contain error or message. Always check the HTTP status before parsing success data.

Status	Meaning
400	Invalid parameters or body
401	Authentication required or invalid
403	Insufficient permission, license, or organization access
404	Resource not found or not visible to this caller
409	Conflict, including an outdated candidate review version
429	Rate limit reached
5xx	Server or upstream failure

Pagination

Core list endpoints use page and pageSize, not limit. Pages start at 1. Defaults differ: project metrics use 10 items, while assets and findings use 100. Server limits may cap larger requests. Do not assume every list includes a total count or shared metadata shape.

```
curl --fail-with-body -H "X-HASHIRO-TOKEN: $HASHIRO_TOKEN" \
  'https://api.hashiro.ai/api/scans/?project=my-project&page=1&pageSize=100'
```

Fetch subsequent pages until the endpoint returns no more records. Project-scoped paths use the project's identifier, not its display name.

Rate limits and retries

Sign-in, OTP, password recovery, and AI-assisted actions have specific rate limits. On 429, respect a retry header when present or back off before retrying. Avoid blindly retrying writes because they can create duplicates or repeat paid AI actions.

05.02

Authentication

Use **API tokens** for integrations and Galileo. The web console uses HTTP-only session cookies.

API token

1. Sign in at [Hashiro](#).
2. Open **My Account**.
3. Generate an API key and copy it immediately.
4. Store it as a private environment variable or secret.

The token is shown only once. Generating a new token invalidates the old one. Tokens inherit the user's permissions and organization scope.

```
curl --fail-with-body \  
  -H "X-HASHIRO-TOKEN: $HASHIRO_TOKEN" \  
  https://api.hashiro.ai/api/organizations/me
```

Both providers and clients use this header and the same production base URL. Galileo reads the token from `api_key` in [its configuration](#).

POST `/users/generateToken` generates a replacement token for the authenticated user. This rotates a credential, so use the UI for initial setup and avoid calling it as a connectivity test.

Browser sessions

POST `/auth/signin` accepts:

```
{  
  "email": "user@example.com",  
  "password": "your-password"  
}
```

The response can require MFA or a first-login password change before full access. Session cookies are `access_token` and `refresh_token`. The refresh cookie renews an expired access token. GET `/auth/logout` clears the session.

MFA

Method	Path	Purpose
GET	<code>/auth/otp/create</code>	Start TOTP setup in an authenticated setup session
POST	<code>/auth/otp/create</code>	Verify setup and enable MFA
POST	<code>/auth/otp/validate</code>	Complete MFA during login

Method	Path	Purpose
POST	/users/otp/enable	Start authenticated MFA setup
POST	/users/otp/reset	Reset own MFA where allowed
POST	/users/otp/disable	Disable own MFA where policy allows

Use the console for the full setup flow. OTP verification is rate-limited, and organization policy can require MFA.

Password recovery and change

Method	Path	Body
POST	/auth/password-reset/request	<code>{"email": "user@example.com"}</code>
POST	/auth/password-reset/confirm	<code>{"token": "RESET_TOKEN", "password": "NEW_PASSWORD"}</code>
POST	/users/me/change-password	<code>{"currentPassword": "CURRENT_PASSWORD", "newPassword": "NEW_PASSWORD"}</code>

Recovery uses the emailed token. Provisioned users may need to change their password before accessing the console.

Invitation acceptance

POST /auth/register accepts an invitation token and the required registration fields. This activates an invited account; it is not a general-purpose unauthenticated organization-creation endpoint.

Common failures

An invalid token returns an authentication failure. A valid token can still receive 403 for an unauthorized organization or operation. Regenerate a leaked token and update integrations that used it.

05.03

Projects and scope

All paths are relative to `https://api.hashiro.ai/api/` and require `X-HASHIRO-TOKEN`.

Access

Reads require organization-wide project access or project membership. Creation requires organization-wide project permission. Writes require the corresponding permissions and either provider ownership or a client-managed project belonging to the caller. Provider-managed projects are read-only to their client through configuration-write routes.

Routes

Method	Path	Operation
GET	<code>/projects/metrics</code>	List authorized projects with metrics
GET	<code>/projects/:projectName</code>	Read a project
PUT	<code>/projects/:projectName</code>	Create a project
PATCH	<code>/projects/:projectName</code>	Update project details
DELETE	<code>/projects/:projectName</code>	Request deletion of the project and its dependent data
GET	<code>/projects/:projectName/:scope</code>	Read inscope or outscope
POST	<code>/projects/:projectName/:scope</code>	Add scope entries
PATCH	<code>/projects/:projectName/:scope/asset/:assetId</code>	Update a scope entry
DELETE	<code>/projects/:projectName/:scope/asset/:assetId</code>	Delete a scope entry
PATCH	<code>/projects/:projectName/members</code>	Update membership
PATCH	<code>/projects/:projectName/displayName</code>	Update display name
POST	<code>/projects/:projectName/report</code>	Request PDF generation
GET	<code>/projects/:projectName/report/status</code>	Check generation status
GET	<code>/projects/:projectName/report/download</code>	Download the ready PDF
GET	<code>/projects/:projectName/download/:fileFormat</code>	Download a supported report format

List projects

Use GET `/projects/metrics?page=1&pageSize=10`. This is the project-list route used by Galileo. GET `/projects/` is not the documented list endpoint.

Create

Use a lowercase project identifier containing letters, numbers, and hyphens, with at most 50 characters. Keep the identifier separate from its display name. Example body for PUT `/projects/my-project`:

```
{
  "platform": "pentest",
  "type": "private",
  "status": "active",
  "definitions": {"category": "internal"}
}
```

Providers can associate a project with an authorized linked client using `customer_id`. A client cannot select another organization. Available categories and project limits depend on organization settings and license.

Scope

`:scope` is `inscope` or `outscope`. Example body for POST `/projects/my-project/inscope`:

```
{
  "assets": [
    {"asset": "app.example.com", "asset_type": "domain"}
  ]
}
```

Scope type values include `wildcard`, `domain`, `ip_address`, `cdir`, `ios`, `android`, `source`, and `other`. `cdir` is the existing API spelling for CIDR. Use the ID returned for an entry when updating or deleting it.

Reports

Report routes apply to supported provider-managed assessment projects. A client-managed project does not automatically have provider-report access. Request generation, check status, and download only after the PDF is ready. Download responses are files rather than JSON. Template availability is organization-specific.

Deletion

Deleting a project starts removal of its dependent data. Treat it as irreversible and do not reuse the identifier until deletion completes.

05.04

Assets, URLs, and services

Use `https://api.hashiro.ai/api/` with `X-HASHIRO-TOKEN`. IDs and project identifiers must belong to your authorized scope.

Routes

Method	Path	Operation
GET	<code>/assets/</code>	List assets using project, asset, filter, page, pageSize
GET	<code>/assets/:assetId</code>	Read asset by ID
POST	<code>/assets/:projectName</code>	Create assets from a JSON array
PUT	<code>/assets/:projectName</code>	Alternative create route
POST	<code>/assets/search</code>	Search the authorized inventory
GET	<code>/assets/inventory/stats</code>	Read inventory filter statistics
GET	<code>/assets/paths</code>	List paths with project and optional asset filters
GET	<code>/assets/:assetId/paths</code>	List paths for an asset ID
GET	<code>/assets/paths/:pathId</code>	Read a URL path
POST	<code>/assets/:projectName/paths</code>	Add paths, with asset as a query parameter
POST	<code>/assets/:projectName/services</code>	Add service records, with asset as a query parameter
POST	<code>/assets/:projectName/restart-probe</code>	Request refreshed probe data
DELETE	<code>/assets/</code>	Delete selected assets using the required filters
DELETE	<code>/assets/project/:projectName</code>	Delete all project assets
DELETE	<code>/assets/paths</code>	Delete selected paths
DELETE	<code>/assets/path/:pathId</code>	Delete a path by ID

Read assets

```
curl --fail-with-body -H "X-HASHIRO-TOKEN: $HASHIRO_TOKEN" \  
'https://api.hashiro.ai/api/assets/?project=my-project&filter=all&page=1&pageSize=100'
```

`filter` selects availability; use `all` when you need unavailable assets as well. Detail routes use the asset ID, while the asset query filter uses its name. An asset record includes fields such as `id`, `asset`, `project`, `type`, `available`, `scope`, `probe`, `ip_address`, and `services`.

Create assets

Send an array to `POST /assets/my-project`:

```
[
  {"asset": "app.example.com", "type": "domain", "available": true}
]
```

Creation accepts authorized project members where the license and project ownership allow it. Deletion and `restart-probe` routes have stricter provider/project permissions. Do not assume every read permission grants write access.

URL paths

`POST /assets/my-project/paths?asset=app.example.com` accepts an array of path records. Each record can include `path`, `url`, `source`, `probe`, and `tags`. Use full URLs in the Galileo CLI; the API stores the path and URL separately.

Services

`POST /assets/my-project/services?asset=app.example.com` accepts an array of service records:

```
[
  {"port": 443, "protocol": "tcp", "state": "open", "service": "https"}
]
```

This records known observations; it does not itself launch a port scan. Reads expose recorded services through the asset details. States include `open`, `filtered`, `closed`, and `unknown`.

Inventory search

`POST /assets/search` searches the caller's authorized inventory. Provider network access is limited to linked clients; clients search their own organization only. The query is not a mechanism for selecting an unrelated tenant.

05.05

Findings and review

Finding endpoints use `/scans`, the existing API resource name. Use `X-HASHIRO-TOKEN` with `https://api.hashiro.ai/api/`.

Access

Read access follows project membership or organization-wide project access. Creation, update, and deletion require `pentester/project-management` permission and a permitted provider or client-owned project. Clients can review eligible public AI candidates, but cannot access private provider drafts or unrelated findings.

Routes

Method	Path	Operation
GET	<code>/scans/</code>	List findings with project, page, pageSize
GET	<code>/scans/:scanId</code>	Read a finding
PUT	<code>/scans/:projectName</code>	Create a finding
POST	<code>/scans/:projectName</code>	Update a finding using its ID in the body
DELETE	<code>/scans/:projectName/:scanId</code>	Delete a finding
POST	<code>/scans/:projectName/bulk-status</code>	Change selected findings status
POST	<code>/scans/:projectName/bulk-delete</code>	Delete selected findings
POST	<code>/scans/:projectName/:scanId/review</code>	Review a candidate with its expected version
POST	<code>/scans/:projectName/:scanId/ai/demonstration</code>	Generate a draft demonstration for review
POST	<code>/scans/:projectName/:scanId/ai/demonstration/apply</code>	Apply a reviewed draft
POST	<code>/scans/:projectName/:scanId/ai/merge</code>	Propose a merge for review

Create a finding

Send this body to `PUT /scans/my-project`:

```
{
  "title": "Missing security header",
  "asset": ["app.example.com"],
  "severity": "low",
```

```
"description":"Observed during the assessment",
"remediation":"Configure the required header"
}
```

asset is an array, even for one host. Optional fields include observation, demonstration, remediation, references, requests, CVSS, and tags. References must be HTTP or HTTPS URLs. cwe and owasp, when supplied, are also URLs.

Creation returns a result containing success and Ids; preserve those identifiers. Clients' new findings are opened according to the server's policy; do not assume a requested draft status is accepted for every caller.

Update

POST /scans/my-project requires id, title, and a nonempty asset array. Read the current finding first, preserve these required fields, and submit the permitted changes. Pending AI candidates must use the candidate-review route for status decisions.

Bulk status

```
{
  "ids":["FINDING_ID"],
  "status":"pending_retest"
}
```

Send to POST /scans/my-project/bulk-status. Bulk requests are limited to 500 IDs. bulk-delete accepts the ids array without status and permanently removes authorized findings.

Candidate review

Use the review version returned by the candidate and submit the decision with expected_review_version. A conflict requires reading the latest record before deciding again. Review and merge options are constrained by the caller's role, project, and candidate visibility. AI draft generation can consume credits; review the draft before applying it.

See the [issue lifecycle](#) and [AI review guide](#) for the interface workflow.

05.06

Organizations and users

Providers and clients use `https://api.hashiro.ai/api/` with `X-HASHIRO-TOKEN`. Organization permissions do not grant access to arbitrary organizations.

Routes

Method	Path	Operation
GET	<code>/users/me</code>	Read own user
PATCH	<code>/users/me</code>	Update own profile
PATCH	<code>/users/me/appearance</code>	Update theme and language
GET	<code>/organizations/me</code>	Read own organization
PATCH	<code>/organizations/me</code>	Update own organization
GET	<code>/organizations/customers</code>	List authorized organization relationships
GET	<code>/organizations/:organizationId</code>	Read an authorized organization
POST	<code>/organizations/clients</code>	Provider: provision a client and initial user
DELETE	<code>/organizations/:organizationId</code>	Provider: delete an unclaimed client it provisioned
POST	<code>/organizations/me/deletion-request</code>	Request own organization deletion
GET	<code>/users/</code>	List own organization users
GET	<code>/users/:organizationId</code>	List permitted related organization users
PUT	<code>/users/</code>	Create an authorized user
POST	<code>/users/:organizationId/:userId</code>	Update an authorized user
POST	<code>/users/:organizationId/:userId/delete</code>	Remove an authorized user
POST	<code>/users/:organizationId/:userId/resend-invite</code>	Resend a provisional client invitation
POST	<code>/users/:organizationId/:userId/revoke-invite</code>	Revoke a provisional invitation
POST	<code>/users/:organizationId/:userId/reset-password</code>	Reset a permitted provisional password
POST	<code>/organizations/me/logo</code>	Upload own logo

Method	Path	Operation
DELETE	/organizations/me/logo	Remove own logo
GET	/organizations/me/integrations/	List configured integrations
POST	/organizations/me/integrations/	Create an integration
GET	/organizations/me/integrations/:integrationId	Read an integration
PATCH	/organizations/me/integrations/:integrationId	Update an integration
DELETE	/organizations/me/integrations/:integrationId	Remove an integration
POST	/organizations/me/integrations/:integrationId/test	Test an integration

Start with your identity

Read /users/me and /organizations/me to identify your user, organization, and available permissions. A client's management role applies to its own organization. A provider relationship permits only the operations specifically allowed for linked clients.

Provision a client

A provider with organization-create permission can send this body to POST /organizations/clients:

```
{
  "name": "Example Client",
  "shortname": "example-client",
  "initial_user": {
    "email": "manager@example.com",
    "firstname": "Example",
    "lastname": "Manager",
    "mode": "invite"
  }
}
```

Invite mode emails the initial user a password-setup link. The organization remains provisional until claimed. Providers can manage its invitation and delete it only while the provisional-state rules allow it. After claim, organization deletion uses the client's own deletion request and platform review.

Users

User creation and management require the relevant User permission and tenant access. Provider reset/revoke/resend operations are for eligible provisional client accounts; they do not grant ongoing control over a claimed client's credentials.

The regular organization management role is manager. Global administrator access is reserved for HashiCorp platform staff and is not a client or provider role.

Integrations

Integration endpoints are own-organization operations and require the applicable management permissions and license. Credentials are write inputs; do not expect secret values in subsequent responses. Use the [Jira guide](#) for mapping and synchronization steps.