Total Visibility and Machine-Speed Governance with Evo AI-SPM

snyk

The governance and posture system of record for every model, dataset, and policy.

THE PROBLEM

The invisible AI attack surface in your code

AI adoption is happening dependency-first. Models, SDKs, skills, orchestration frameworks, and MCP servers are embedded directly in repositories and developer environments long before anything reaches runtime. Yet most security programs still focus on protecting cloud workloads after deployment.

This creates four critical gaps:

1. **Shadow AI everywhere:** AI models, agent toolkits, and MCP servers are introduced into repositories and developer endpoints without oversight.
2. **No system of record for AI risk:** There are no CVEs for poisoned models, unsafe prompt frameworks, or risky agent behaviors embedded in code.
3. **Manual governance that developers bypass:** Static “approved tools” lists and ticket-based approvals slow teams down and get ignored.
4. **Late detection, expensive remediation:** AI risks are often discovered only after deployment, when architectural mistakes are harder and more costly to reverse.

Organizations can’t govern what they can’t see.

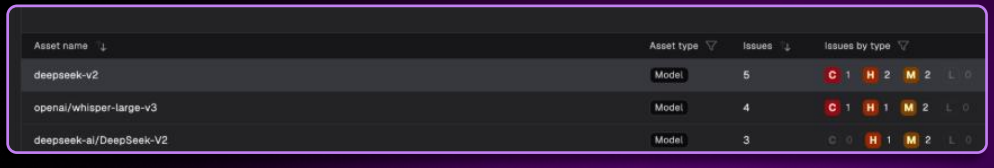
THE SOLUTION

Code-first AI Security Posture Management (AI-SPM)

Evo AI-SPM (AI Security Posture Management) is a core module of Evo by Snyk. It delivers continuous discovery, governance, and contextual risk intelligence for AI systems before they reach production. Evo AI-SPM secures AI as it’s built and evolves.

It provides:

- Continuous AI asset discovery directly in code configurations and developer frameworks, including detection across Java, Go, Python, and TypeScript codebases, configuration files (JSON, YAML, ENV, Dockerfiles, Helm charts), and custom internal AI libraries used to access models or orchestrate agents. Discovery now extends beyond code, and a unified inventory can find AI agents and MCP servers running on developer machines,
- Detect Agent Skills running in repos (in SKILL.md bundles),
- End-to-end contextual AI risk prioritization
- Natural language policy creation with enforceable guardrails
- Developer-native remediation across Git and CI/CD, with the Evo MCP Server bringing AI risk insights into agentic developer tools like Claude Code, Gemini CLI, and Factory AI



Asset name	Asset type	Issues	Issues by type
deepseek-v2	Model	5	C 1 H 2 M 2 L 0
openai/whisper-large-v3	Model	4	C 1 H 1 M 2 L 0
deepseek-ai/DeepSeek-V2	Model	3	C 0 H 1 M 2 L 0

Unlike runtime-only AI security tools, Evo AI-SPM starts at the source, where AI risk enters.

Why start with AI-SPM?

Starting with Evo AI-SPM delivers the highest risk reduction with the lowest friction because most AI risk enters through developer dependencies–models, SDKs, orchestration frameworks, and MCP servers embedded directly in code long before deployment. By applying code-level visibility and governance, AI-SPM protects the largest and most immediate AI attack surface across all use cases, integrates into existing SCA, SAST, SBOM, and CI/CD workflows for faster adoption, and aligns with global standards that sequence AI controls around inventory and supply chain before runtime behavior.

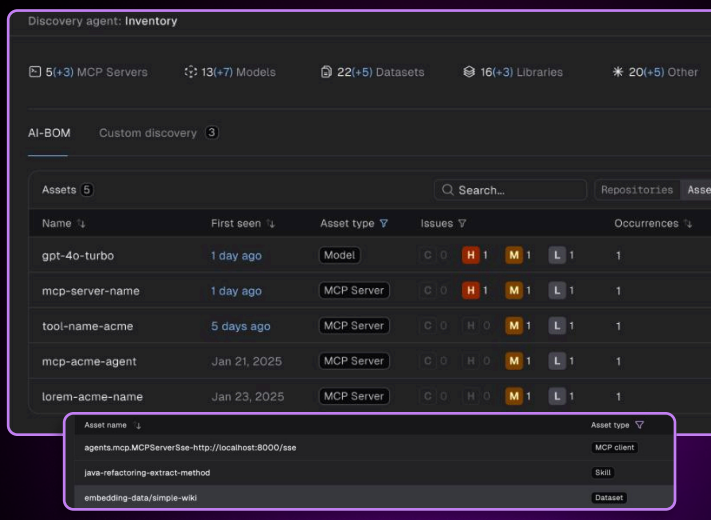
EVO AI-SPM ARCHITECTURE

Evo AI-SPM is built on Evo's shared platform capabilities: Discovery, Risk Intelligence, and Policy to deliver continuous AI posture management. In addition, Evo offers a natural-language interface for exploring AI risk, querying assets and policies, investigating relationships, and navigating directly to relevant AI assets.

Discovery

Discover Shadow AI

Evo analyzes source code, configuration, and infrastructure definitions to detect how models, frameworks, and agent tooling are implemented, capturing AI usage patterns that traditional dependency scanning misses.

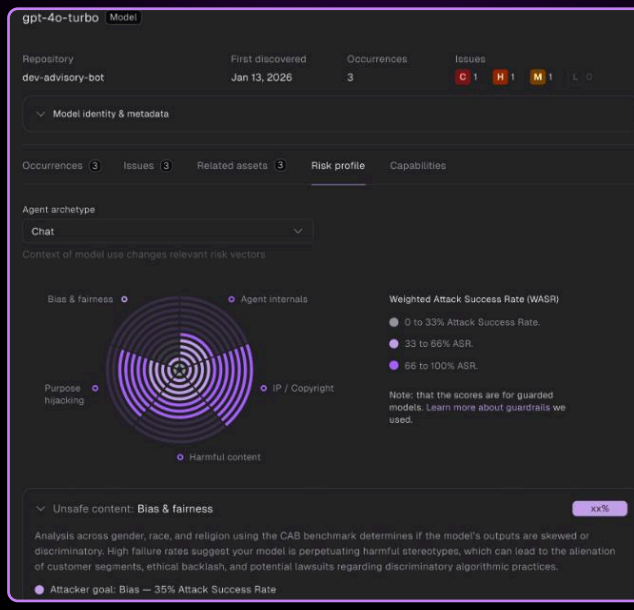


Discovers:

- Models and LLMs
- Frameworks and libraries (e.g., LangChain, agent toolkit s)
- Datasets and data sources
- Models, MCP servers, skills, and agent infrastructure in codebases, and agents and MCP servers on developer machines
- Custom AI libraries: Identifies AI implemented through internal developer libraries and platform SDKs, uncovering AI systems even when model access is abstracted behind proprietary tooling.

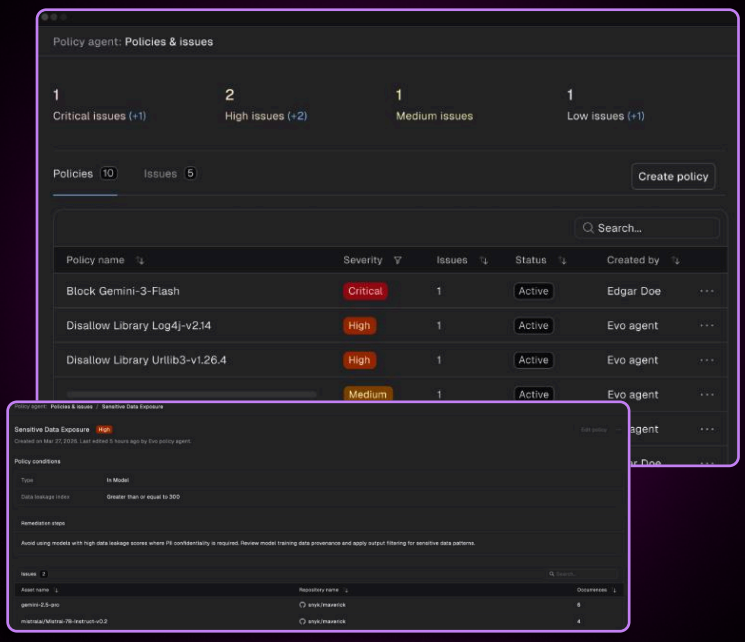
Risk Intelligence

Evo enriches discovered AI assets with adversarial risk intelligence built from Evo's own tests — not static model descriptions. For each model, Evo runs hundreds of real adversarial attacks and publishes Attack Success Rate (ASR): the percentage that succeed, organized by what actually goes wrong. Scores roll up across a three-level taxonomy — from specific attacker goals to sub-categories to nine impact categories — so teams can zoom out to a headline score or drill in to see exactly which attack is succeeding and where:



- **Unsafe content**
- **Information disclosure**
- **Code security**
- **Excessive agency**
- **Data exfiltration**
- **Unauthorized execution**
- **Security degradation**
- **Output manipulation**

Because the same model carries different risk depending on how it's deployed, Evo scores each model inside realistic agent deployment contexts (i.e a coding agent faces backdoors and supply chain attacks; a customer chatbot faces extraction and jailbreaks) All attack vectors are mapped to OWASP guidelines and. scores surface on the asset details page overall and broken down by deployment role so teams have the evidence they need before anything is actioned.



Policy

Translates plain-English intent into enforceable, audit-ready security guardrails.

Enforce policy guardrails

Policies are evaluated instantly against discovered AI assets, enabling rapid governance.

Policies incorporate AI risk intelligence attributes, such as hallucination, bias, adversarial robustness, and licensing, enabling organizations to enforce context-aware governance controls rather than static allowlists. Every Evo tenant also ships with pre-built system policies, active on day one, with minimal setup and the ability to customize. Built on hundreds of real test suites, they evaluate every model in the inventory across risk categories and can automatically raise a high-severity issue when a model's Risk Index crosses a determined threshold.

KEY USE CASES

1. Shadow AI cleanup

Automatically map the codebase to surface hidden AI assets and unauthorized libraries in minutes. AI-BOM now scans every repository by default, scaling coverage to millions of repositories.

2. Agentic governance

Identify untrusted inputs, overly permissive data access, and unsafe model configurations before agents go live.

3. Real-time compliance

Replace months of manual audit prep with defensible, policy-driven AI governance. Manage policies and track editor attribution in the Evo UI, and export findings to CSV for auditors.

4. Secure AI sprints

Bring AI risk discovery and policy enforcement into the sprint, so teams fix issues before merge, not after an incident.

5. Provenance & relationship mapping

Map the relationships across your discovered AI assets to see which applications, models, and MCP servers are attached to your data and how they connect, so teams understand how AI components relate, not just that they exist.

WHY EVO AI-SPM?

Code-first visibility vs. runtime-only security

Cloud security tools see AI after it reaches production. Evo discovers risk wherever AI is built, including local models and MCP tools running in developer environments

System-level AI intelligence

Evo correlates code composition, policy intent, agent behavior, and model switching to understand what an AI system can actually do, not just what it calls.

Built on Snyk's application security platform

Evo extends Snyk's deep code, dependency, and infrastructure intelligence to AI-native systems, leveraging existing integrations across repos, IDEs, and CI/CD

Developer-first governance

Findings surface directly in developer workflows, enabling secure-by-design AI adoption without slowing innovation.

- **CI/CD pipeline enforcement:** Enforce AI governance in build pipelines using the Snyk CLI to evaluate AI assets against policies and warn or block builds that introduce non-compliant AI components.
- **Policy-driven remediation guidance:** Attach custom remediation instructions to policies so developers receive contextual guidance for fixing violations directly in the Evo UI, CLI output, and reports.

BENEFITS BY ROLE

Role	Primary value
CISO and security leadership	• Establish a system of record for AI risk • Translate compliance requirements into enforceable policy • Reduce behavioral and governance liabilities • Prove AI initiatives are governed and defensible
AppSec and security engineers	• Eliminate Shadow AI blind spots • Replace spreadsheets and manual reviews with automation • Govern models, datasets, and MCP servers at scale
Development and platform leaders	• Embed security into AI development workflows • Prevent architectural debt from unsafe models and frameworks • Maintain AI velocity without introducing unmanaged risk

Get started with Evo AI-SPM

AI risk enters at build time—not just at runtime. Evo AI-SPM delivers continuous AI discovery, governance, and contextual risk intelligence directly in code and developer environments, providing real-time visibility, policy enforcement, and scalable AI security across the SDLC

See Evo AI-SPM in action today and uncover every AI component running in your codebase.

snyk