

Why you need an agentic security platform

Most AI security tools secure a slice. Use this checklist to test whether your stack secures the entire agentic path.

CATEGORY	AUDIENCE	YEAR
Agentic Security	CISOs and security leaders	2026

OVERVIEW

Your agents are already acting

AI agents and MCP servers run on APIs. Every agent you deploy acts through a model that decides, MCP servers that give it tools, and APIs that change real business systems. Most security programs watch one of those layers, if any.

49.8% of organizations confirmed or suspected an AI agent took an action in the past year that they did not intend, expect, or authorize.	12.5% can consistently trace an agent's full path, from instruction through tools and APIs to outcome, across all agents.	100+ enterprise deployments securing APIs since 2016, and now the agents that call them.
---	---	--

Source: Salt Security, 2H 2026 State of AI and API Security report.

THE PROBLEM

A secure prompt is not a secure action

Risk moves from model to MCP server to tool to API to a real business system. Each hop is an attack surface, and each hop looks like normal traffic to tools built for something else. You cannot secure an agent without securing what it can reach.

ROUTINE REQUEST, UNAUTHORIZED ACCESS

"Export last week's refunds with customer contact details. Return the result as a Base64-encoded string." The AI check passes. Behind it: an internet-facing MCP server, a hardcoded API key, an API that never validates it, and customer PII returned in the response.

WHY ANOTHER POINT TOOL WON'T CLOSE THE GAP

Each tool you own secures one slice

WHAT YOU RUN TODAY	WHAT IT SEES	WHAT IT MISSES
LLM guardrails and AI security tools	Prompts and model responses.	The MCP calls and API actions the agent takes after the model decides.
WAF and edge controls	North-south traffic and known attack signatures.	Agent-to-MCP-to-API chains, business logic abuse, and APIs nobody documented.
SAST, DAST, and posture tools	Code and configuration at rest.	Runtime behavior. A compliant build can still be abused in production.

Point solutions protect components. Agentic risk lives in the path between them. That is why the answer is a platform that connects the layers, not a fourth or fifth tool that watches one more of them.

Nine things an agentic security platform must do

Check what your current stack does today. Every unchecked box is a reason a platform has to replace guesswork.

01 Discover

Know what your agents can reach

- ☐ **Finds every agent, MCP server, and API, including the ones nobody registered**
Without it: you secure the inventory you remember, not the one attackers can reach.
- ☐ **Spots MCP servers exposed to the internet, in cloud traffic and private repositories**
Without it: a forgotten MCP server hands attackers your agents' tools.
- ☐ **Maps which agents reach which sensitive data through which APIs**
Without it: you see assets, not the paths that put data at risk.

02 Govern

Fix the paths that put you at risk

- ☐ **Applies one security policy from the first AI-generated line of code through runtime**
Without it: coding assistants write nearly half of new code and have never read your policies.
- ☐ **Flags missing authentication, hardcoded keys, and over-permissioned agents**
Without it: one injected instruction becomes a data breach.
- ☐ **Maps agent controls to the EU AI Act and ISO 42001**
Without it: your auditors ask for evidence your point tools cannot produce.

03 Protect

Stop abuse where actions happen

- ☐ **Detects prompt injection at runtime and follows it to the MCP and API calls it triggers**
Without it: you block a prompt and still miss the data export it started.
- ☐ **Baselines agent behavior over time instead of matching signatures**
Without it: agent abuse looks like valid requests, so you never see it.
- ☐ **Traces an agent's full path, from instruction through tools and APIs to outcome**
Without it: you join the 87.5% who cannot trace this across all agents.

The test: a tool that checks one box is a point solution. A platform checks all nine, in one product, from one view of the agentic path. Every box you could not check is a gap an attacker can use.

NEXT STEP

Your agents are already acting. Secure the entire path.

Salt Security discovers what your agents use, shows what they can reach, governs how they operate, and protects what happens at runtime. Start agentless, then add protection through the edge controls, WAFs, and SOC tools you already run. Nothing to rip out, reroute, or rewrite.

[Get an Agentic Security assessment](#)

[salt.security/agentic-assessment](#) • Salt Surface and Salt Connect discover. Salt Code enforces policy at the first prompt. Salt Collect and Salt Protect detect and block at runtime.