

Stopping AI attacks *before they become API attacks*

Salt AI-DR delivers native runtime protection for LLM interactions, then connects what happens at the model to the MCP servers, tools, and APIs downstream. Built into Salt Agentic Detection and Response.

4

LAYERS CONNECTED IN ONE
ATTACK VIEW: MODEL, MCP
SERVER, TOOL, AND API

1st

PLATFORM TO LINK A PROMPT
INJECTION TO THE API ATTACK
THAT FOLLOWS

0

GUARDRAILS OR GATEWAYS YOU
HAVE TO REPLACE TO ADOPT SALT
AI-DR

THE PROBLEM

An attack on your model becomes an attack on your business systems.

AI agents connect models to tools and business systems. An attack that starts with a malicious prompt can expose those connections and lead to unauthorized access or actions downstream. Direct and indirect prompt injection, jailbreak attempts, and unsafe model behavior are no longer conversational problems. They are entry points into the systems that run the business.

Meanwhile, enterprises run AI guardrails across cloud platforms, endpoint and SASE solutions, AI gateways, and managed AI services. Each control protects the interactions it covers, leaving teams to identify inconsistencies and gaps across environments on their own. Homegrown agents running in Kubernetes often sit outside all of them.

The result is a visibility problem with a business consequence. Security teams can see that a prompt was malicious. They cannot see what it reached.

THE SHIFT

The goal is no longer to block a bad prompt in isolation. It is to see how an attack moves across the model, the MCP server, the tool, and the API behind it, and to understand which business systems are at risk. Salt AI-DR is built to do exactly that.

HOW SALT AI-DR WORKS

Protection across *the full agentic path*

AI-DR is built into Salt Agentic Detection and Response, extending its runtime protection to the LLM layer. The Salt Agentic Security Graph connects agents to their models, MCP servers, tools, and downstream APIs, so an attack against the model and the MCP and API attacks that follow are seen as one event in one platform.

STAGE	OUTCOME	SALT CAPABILITY	HOW IT WORKS	COVERAGE
01 Map	Every agent, model, and MCP server accounted for	Agentic Security Graph	Connects each agent to the models it calls, the MCP servers and tools it uses, and the APIs behind them. This is the map every detection downstream is written against.	NATIVE
02 Assess	Guardrail coverage you can actually see	Guardrail and gateway visibility	Through AG-SPM, Salt gives teams one view of guardrail configurations across cloud platforms, endpoint and SASE solutions, AI gateways, and managed AI services, and the agents each one protects.	UNIFIED
03 Protect	LLM interactions defended at runtime	Native AI-DR	Real-time protection against direct and indirect prompt injection, jailbreak attempts, unsafe model behavior, and other runtime threats to the model layer.	NATIVE
04 Connect	The prompt linked to what it reached	Cross-layer attack correlation	Ties an attack against the model to the subsequent MCP and API attacks in the same platform. Teams see how the attack started, what it targeted, and which business systems are at risk.	NATIVE
05 Follow	Attacks tracked past the agent	MCP and API runtime detection	Detection continues when the attacker abandons the agent and targets the API directly, so the campaign stays visible as one attack rather than two unrelated alerts.	NATIVE
06 Close	Protection where guardrails do not reach	Gap-fill deployment	Where protection is missing, including homegrown agents running in Kubernetes, native AI-DR fills the gap inside AG-DR. Existing gateways and security products stay exactly where they are.	NATIVE

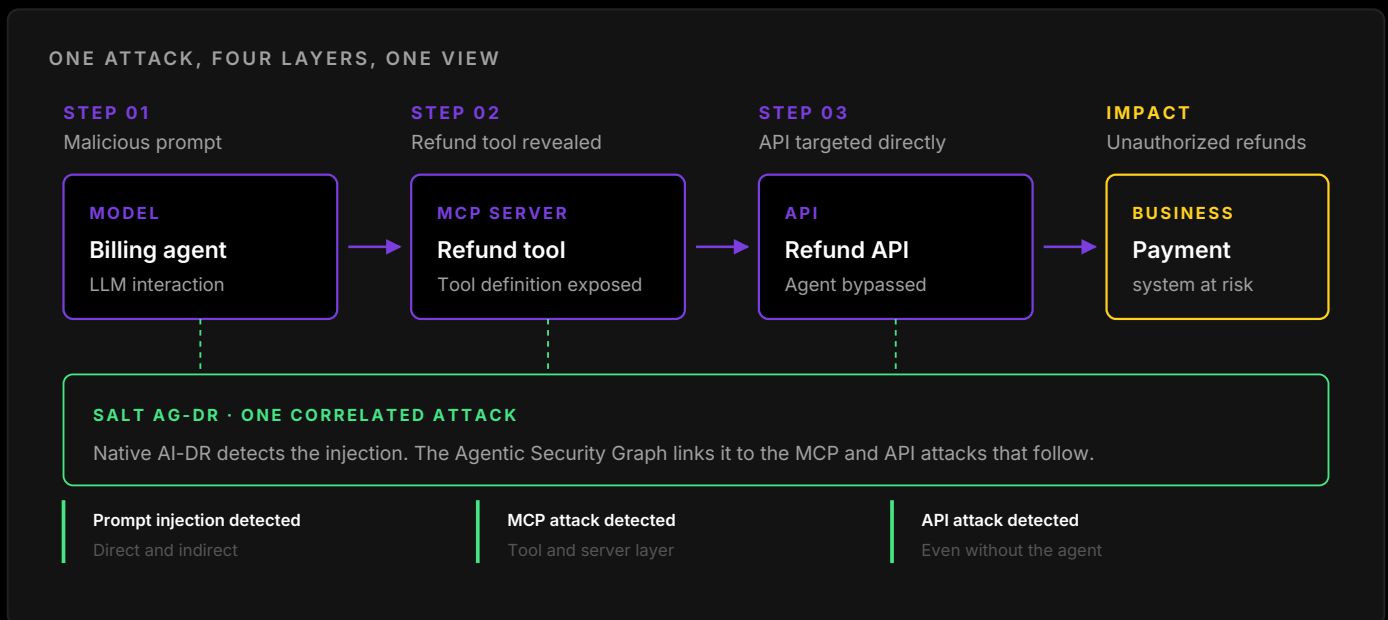
PLATFORM CONTEXT

AG-DR and Agentic Security Posture Management (AG-SPM) form the Salt Agentic Security Platform. AG-SPM helps teams understand their agents and security gaps. AG-DR detects and connects attacks across those agents and the systems they use.

ANATOMY OF AN ATTACK

How a prompt becomes *a payment problem*

An attacker tricks a billing agent into revealing information about a refund tool on an MCP server and the API behind it. The attacker then targets that API directly to attempt unauthorized refunds. What began as a malicious prompt has become an attack on a payment system. Salt shows these steps together.



WHERE ENTERPRISES RUN AI GUARDRAILS TODAY

- Cloud platforms
- Endpoint and SASE
- AI gateways
- Managed AI services
- Homegrown agents in Kubernetes

Salt brings these controls into one security view through AG-SPM and the Agentic Security Graph, then fills the gaps with native AI-DR. Customers establish more consistent protection across models, agents, MCP servers, tools, and downstream APIs while keeping the guardrails they already use.

"An attack on an AI model can become an attack on the systems that run the business. Our native AI-DR protects LLM interactions and connects what happens at the model to the tools and APIs downstream. Teams can see the full attack, understand what is at risk, and fill protection gaps while keeping the guardrails they already use."

ROEY ELIYAHU · CO-FOUNDER AND CEO, SALT SECURITY

WHY IT MATTERS

What changes when AI-DR is *in your stack*

Native AI-DR does not add another isolated guardrail to the pile. It changes what a security team can see when an AI attack happens, and how quickly they can say what is at risk.

/ 01

The full attack, in one view

From the prompt injection through the MCP server to the API attack. Teams see how the attack started, what it targeted, and which business systems are exposed, instead of stitching together alerts from four tools after the fact.

/ 02

Protection where your guardrails end

Coverage gaps get filled natively, including homegrown agents running in Kubernetes that sit outside every gateway and managed service you already pay for.

/ 03

Your existing controls stay in place

Salt unifies visibility across the guardrails and gateways you run today rather than replacing them. Consistency improves without a rip-and-replace project or a new set of enforcement points to argue about.

/ 04

Model attacks measured in business risk

A jailbreak attempt stops being a model-quality metric and becomes what it actually is: an attempted path to a payment system, a customer record, or a production API.

AG-SPM

Agentic Security Posture Management. Helps teams understand their agents, their models, their MCP servers, and the security gaps between them.

AG-DR

Agentic Detection and Response. Detects and connects attacks across those agents and the systems they use, at runtime.

AI-DR

Native LLM runtime protection, built into AG-DR. Extends detection and response to the model layer itself.

A prompt injection you cannot trace is an API breach you cannot explain.

Native AI-DR capabilities are available within Salt AG-DR as part of the Salt Agentic Security Platform.

[Request a Demo →](#)