SALT

SOLUTION BRIEF • COMPETITIVE ANALYSIS

Salt Security vs. Noma Security

Two distinct approaches to agentic security. Why model-layer governance is not enough when agents are acting across your API fabric.

CATEGORY	PLATFORMS COMPARED	YEAR
Agentic Security	Salt Security & Noma Security	2026

OVERVIEW

Agentic security requires protecting the action layer

AI agents do not create risk by thinking. They create risk by acting: calling APIs, invoking MCP servers, accessing databases, triggering downstream workflows. Salt Security was built to secure what agents do at runtime, across the full API fabric, from the internet in. Noma Security was built for a different problem: what is the AI system configured to do, and is that configuration safe?

Both platforms provide genuine value. The difference is architectural. Noma works from within agent platforms outward. Salt works from the external attack surface inward, and through live runtime traffic. The coverage gap between those two starting points is where the majority of agentic risk lives.

8

Years of API security heritage vs. Noma's 2 years in market

70+

Infrastructure technologies monitored by Salt Collect at runtime. Noma: zero runtime API monitoring.

4

Data sources in Salt's Agentic Security Graph vs. Noma's single-source Risk Map

PLATFORM APPROACHES

Two starting points. One covers the layer that matters.

SALT SECURITY

Action-layer security

Salt Security was built to secure what AI agents act on: the APIs, MCP servers, and infrastructure agents invoke when they take real-world actions. Eight years of API security expertise, now extended to the agentic layer, means Salt brings both infrastructure depth and runtime visibility to protect the action layer at scale.

Salt Surface discovers external exposure from the internet out with zero deployment. Salt Collect monitors live API and MCP traffic at runtime across 70+ technologies. The Agentic Security Graph correlates all four data sources into a single correlated security model.

NOMA SECURITY

Model-layer security

Noma's platform is built around understanding and governing the AI system itself. It connects to agent platforms, code repositories, and cloud providers to build inventory. Its Agentic Risk Map visualizes agent-to-agent connections based on that configuration-level discovery.

The architectural constraint: Noma's discovery starts from within connected platforms and produces a picture of what agents are configured to do. It has no visibility into what agents are actually doing at runtime across your API fabric.

THE COVERAGE GAP

Noma produces a picture of what your agentic infrastructure is configured to do. Salt produces a picture of what it is actually doing. Runtime API traffic monitoring, external attack surface discovery with zero deployment, and behavioral anomaly detection across 70+ infrastructure technologies are all capabilities Noma does not offer. That is the gap where agentic attacks succeed.

CAPABILITY COMPARISON

Salt Security vs. Noma Security: side by side

CAPABILITY	SALT SECURITY	COMPETITOR
Unified Agentic Discovery	**STRONG** Discovers APIs, MCP servers, and AI-driven assets across external exposure, cloud, code repositories, and runtime as a unified environment.	**NOT IN CURRENT SCOPE** Noma has no runtime API traffic monitoring. Discovery starts from within connected platforms and stops before the API layer.
Agentic Security Graph	**STRONG** Correlates LLMs, MCP servers, APIs, identities, and sensitive data in one action-layer context — the only platform with this cross-fabric view.	**NOT IN CURRENT SCOPE** Noma's Agentic Risk Map visualizes agent connections based on configuration data. Runtime correlation across LLM, MCP, and API layers is not available.
Salt Code Governance	**STRONG** Governs API and MCP creation in repositories and developer workflows before risky logic ships to production.	**PARTIAL** Noma integrates with source code repositories as part of discovery. Pre-production API and MCP governance is not a primary capability.
Runtime-to-Code Remediation	**STRONG** Feeds runtime findings back into DevOps workflows and AI coding assistants to fix root causes — closing the loop between detection and fix.	**NOT IN CURRENT SCOPE** Noma does not feed runtime findings back into developer workflows or coding assistants.
Identity-Aware Sequence Correlation	**STRONG** Tracks unique agentic identities and multi-step intent across sessions, tools, and services to detect campaigns, not just individual events.	**PARTIAL** Noma monitors agent behavior at the model and configuration layer. Cross-service sequence correlation at the API level is outside current scope.
Behavioral Action-Layer Protection	**STRONG** Detects machine-speed business logic abuse beyond signatures, schemas, or known patterns.	**PARTIAL** Noma detects anomalies at the agent and model layer. API-level behavioral baselines derived from runtime traffic are outside current scope.
Internal & East-West Coverage	**STRONG** Protects internal APIs and downstream service interactions that edge-only and model-only tools miss entirely.	**NOT IN CURRENT SCOPE** Noma monitors agent behavior within connected platforms. East-west internal API traffic is not monitored.
Action-Layer Data Security	**STRONG** Maps sensitive data in motion across APIs, MCP servers, and agent actions across the full fabric.	**PARTIAL** Noma tracks data exposure at the model output layer. Sensitive data in motion across the API fabric is not mapped.

CAPABILITY	SALT SECURITY	COMPETITOR
External Attack Surface Discovery	**STRONG** Salt Surface scans from the internet out with zero deployment, finding exposed APIs, MCP servers, and agent endpoints not registered in any internal system.	**NOT AVAILABLE** Noma's discovery begins from within connected platforms. External internet-facing exposure is outside current product scope.
LLM and Model-Layer Visibility	**PRESENT** Salt provides model identity, version, configuration, and prompt/completion pattern visibility within the Agentic Security Graph.	**STRONG** Noma's primary focus. Deep LLM monitoring, automated red teaming, and model governance are core capabilities.
EU AI Act Compliance Support	**STRONG** Direct article-by-article mapping across Arts. 9, 10, 11, 12, 13, 14, 15, 20, 26, 72, and 73 at the action layer where obligations apply.	**PARTIAL** Noma provides compliance reporting at the model governance layer. Action-layer obligations under Art. 15 are not fully addressed.

Questions to ask in your evaluation

The most productive evaluation question is not which platform has more features, but which platform covers the layer where your risk is highest. Use these scenarios to drive the conversation.

If your concern is what agents are doing at runtime

This is Salt's ground. Runtime API traffic monitoring, behavioral baselines, east-west traffic across MCP servers and internal services, and the correlation of live behavior with posture and code-level findings are all capabilities Noma does not offer. Ask your Noma team to show you runtime API traffic for the APIs your agents are calling. The absence of that data is the answer.

If your agents are deployed in SaaS platforms

Noma's 80+ SaaS platform integrations provide coverage of agents deployed through Copilot Studio, AgentForce, and similar platforms. The important question is what downstream APIs and infrastructure those agents call, and whether you have runtime visibility into that action layer. SaaS agents call APIs that connect to infrastructure Noma cannot see.

If you have EU AI Act compliance obligations

The EU AI Act's most demanding security obligations, under Articles 12, 14, 15, 20, 26, 72, and 73, apply to the action layer: the APIs agents call, the data they access, and behavioral anomalies that constitute a detectable incident. Salt's direct compliance mapping at the layer where those obligations are imposed is materially stronger than Noma's model-governance coverage.

If your primary concern is model governance

Noma's model-layer depth is genuine. For organizations primarily concerned with LLM robustness, prompt safety, and AI governance frameworks, Noma was purpose-built for those problems. Salt provides model layer visibility through the Agentic Security Graph, but organizations typically find that action-layer security is the more urgent gap.

SUMMARY

Key differentiators

SALT SECURITY IS THE STRONGER CHOICE WHEN…

- Runtime visibility into API and MCP traffic is a requirement
- External attack surface discovery with zero deployment is needed
- EU AI Act compliance at the action layer is in scope
- Behavioral anomaly detection across legacy and modern infrastructure is required
- East-west and internal API traffic monitoring is needed
- Your agent footprint spans custom frameworks or environments beyond SaaS
- Eight years of enterprise production credibility is a selection criterion

NOMA SECURITY MAY BE CONSIDERED WHEN…

Noma's scope is narrow relative to the full agentic attack surface. It may be appropriate when:

- LLM governance and model-layer security is the exclusive priority with no runtime infrastructure requirements
- All agents are deployed exclusively through Noma's supported SaaS platforms with no custom API integrations

For any organization with infrastructure footprint beyond SaaS-hosted agents, Noma's coverage ends before the risk begins.

THE COMBINED PICTURE

Model-layer security and action-layer security address different risk surfaces. What organizations consistently find in evaluation is that action-layer visibility — which Salt provides and Noma does not — is the gap with the most immediate risk exposure. An agent that is perfectly governed at the model layer can still exfiltrate data, escalate privileges, or execute unauthorized workflows through the APIs it calls. Salt is the security platform for that layer. No amount of model governance funding closes an eight-year gap in runtime API security expertise.

SEE THE ACTION LAYER FOR YOURSELF

Salt Surface can scan your environment today with zero deployment required and show you the external exposure of your agentic infrastructure. Most organizations find APIs and MCP servers they did not know existed. Ask your Noma team to show you the same view.
Request a demo at salt.security

SALT