SOLUTION BRIEF • COMPETITIVE ANALYSIS

Salt Security vs. Akamai (Noname)

Akamai added API security to a WAF platform via acquisition. Salt built agentic security from the ground up. That architectural difference determines what each platform can and cannot protect.

CATEGORY	PLATFORMS COMPARED	YEAR
Agentic Security	Salt Security & Akamai (Noname)	2026

OVERVIEW

Edge-based API security was not built for the agentic era

Akamai acquired Noname in 2024 to add API discovery capabilities to its CDN and WAF platform. That heritage still defines the product: traffic-centric, perimeter-focused, and architected before AI agents existed as an attack surface. Salt Security was purpose-built for the agentic era — covering the LLM, MCP, and API layers that Akamai's edge-dependent model cannot reach.

For organizations evaluating API and agentic security, the question is not which platform has more features. It is which platform was architected for the threat model you face today, where agents create connections, access data, and execute actions at machine speed across infrastructure that never touches a CDN edge.

8

Years of purpose-built API security research vs. Akamai's 2024 API security acquisition

0

Traffic mirroring required. Salt operates fully out-of-band with no edge dependency.

4

Data sources in Salt's Agentic Security Graph. Akamai: no equivalent cross-fabric correlation.

PLATFORM APPROACHES

Purpose-built vs. bolted-on: the architecture determines the coverage.

SALT SECURITY

Purpose-built agentic security

Salt Security was architected for the full agentic attack surface: LLM connections, MCP servers, and the API fabric where agents execute real-world actions. The Agentic Security Graph maps and correlates every layer out-of-band — no edge dependency, no traffic mirroring required, and no blind spots when agents act inside the perimeter.

Salt Surface discovers external exposure with zero deployment. Salt Collect monitors live traffic across 70+ infrastructure technologies. Salt Code governs API and MCP creation in repositories before deployment. No other platform combines these four coverage dimensions.

AKAMAI (NONAME)

WAF-native API security

Akamai's API security is a 2024 acquisition layered onto a CDN and WAF platform built for perimeter traffic analysis. It discovers APIs through traffic and source code scanning, detects threats through pattern and anomaly analysis at the edge, and can tag APIs that connect to AI services.

The architectural constraint: Akamai's coverage is strongest for traffic that flows through its edge infrastructure. APIs that bypass the edge, east-west internal traffic, and the behavioral attack sequences that span multiple individually valid requests are all areas where the edge-dependent model falls short.

THE COVERAGE GAP

Akamai can tag APIs it observes. Salt correlates the full agentic fabric across LLM connections, MCP servers, APIs, and identities. Akamai has no Agentic Security Graph equivalent, no identity-aware sequence correlation, and no runtime-to-code remediation loop. For organizations deploying AI agents, those are not optional capabilities — they are the difference between knowing an agent exists and knowing what it is doing.

CAPABILITY COMPARISON

Salt Security vs. Akamai (Noname): 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.	**PARTIAL** Akamai discovers APIs via traffic analysis and source code scanning. MCP servers and AI-driven assets outside monitored traffic paths may not be visible.
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 AVAILABLE** Akamai can tag AI-connected APIs but has no equivalent to the Agentic Security Graph — no LLM, MCP, API, identity correlation in a unified model.
Salt Code Governance	**STRONG** Governs API and MCP creation in repositories and developer workflows before risky logic ships to production.	**NOT AVAILABLE** Akamai does not govern API and MCP creation in developer repositories or CI/CD workflows.
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 AVAILABLE** Akamai 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.	**NOT AVAILABLE** Akamai analyzes requests for patterns and anomalies. Identity-aware multi-step sequence correlation across sessions and services is not available.
Behavioral Action-Layer Protection	**STRONG** Detects machine-speed business logic abuse beyond signatures, schemas, or known patterns.	**PARTIAL** Akamai detects known attack patterns and rate anomalies. Business logic abuse composed of individually valid requests is difficult to detect at the edge.
Internal & East-West Coverage	**STRONG** Protects internal APIs and downstream service interactions that edge-only and model-only tools miss entirely.	**PARTIAL** Akamai monitors traffic at the edge. Internal east-west API traffic between microservices that does not traverse the edge is not covered.
Action-Layer Data Security	**STRONG** Maps sensitive data in motion across APIs, MCP servers, and agent actions across the full fabric.	**PARTIAL** Akamai detects data exposure at monitored traffic points. Sensitive data in motion across the full API fabric is not mapped.

CAPABILITY	SALT SECURITY	COMPETITOR
Edge-Independent Coverage	**STRONG** Salt discovers and protects every API regardless of whether traffic routes through any edge infrastructure or gateway.	**NOT AVAILABLE** Akamai's API security coverage is strongest for traffic that traverses its edge infrastructure. APIs that bypass the edge remain in blind spots.
No Sample-Based Discovery Ceiling	**STRONG** Salt does not depend on sampled traffic or request thresholds to discover APIs. Low-volume and agent-generated connections are visible from day one.	**LIMITATION** Akamai's traffic-based discovery can miss low-volume APIs and endpoints that do not generate sufficient traffic to cross discovery thresholds.
Agentic Attack Sequence Detection	**STRONG** Salt detects multi-step agentic attack campaigns that unfold across dozens of individually valid requests and multiple API endpoints.	**NOT AVAILABLE** Akamai's per-request pattern matching and rate-based detection does not reconstruct multi-step attack sequences across API calls.

Questions to ask in your evaluation

The most productive 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 agents create connections outside your CDN edge

This is the most important question for any Akamai evaluation. Salt Surface can show you the APIs and MCP servers your agents are connecting to that Akamai has never seen. Run that scan before your next Akamai conversation. The exposure Akamai cannot see is the exposure that matters most to an attacker.

If you need behavioral detection beyond pattern matching

Akamai's ML-enhanced detection excels at identifying known attack patterns at the request level. What it cannot do is reconstruct intent across multi-step sequences where each individual request is valid. For AI-driven business logic abuse — the attack category purpose-built to bypass per-request detection — Salt's behavioral correlation is the only architectural answer.

If you have EU AI Act compliance obligations

The EU AI Act's security requirements apply to the action layer: the APIs agents call, the data they access, and behavioral anomalies that constitute detectable incidents. Salt's direct article-by-article compliance mapping covers Arts. 9, 10, 11, 12, 13, 14, 15, 20, 26, 72, and 73 at the infrastructure layer where those obligations are imposed.

If you are evaluating both Akamai and Salt

Ask Akamai to demonstrate coverage for three things: APIs your agents call that never traverse the Akamai edge, east-west internal microservice traffic, and a multi-step business logic attack sequence that uses only valid authenticated requests. These are the three gaps Salt closes. Make them concrete in the evaluation.

SUMMARY

Key differentiators

SALT SECURITY IS THE STRONGER CHOICE WHEN…

- Coverage of APIs that bypass the Akamai edge is required
- Agentic Security Graph correlation across LLM, MCP, and API layers is needed
- Multi-step behavioral attack detection beyond per-request pattern matching is a requirement
- East-west and internal API traffic visibility is needed
- Zero traffic mirroring overhead and no edge dependency is important
- EU AI Act compliance at the action layer is in scope
- Runtime-to-code remediation is a requirement

AKAMAI (NONAME) MAY BE CONSIDERED WHEN…

Akamai's coverage is strongest in one specific scenario:

- Traditional WAF and edge protection is the primary requirement and all critical API traffic routes through Akamai's CDN infrastructure
- API discovery from traffic analysis at the perimeter is sufficient and no east-west, agentic, or off-edge coverage is required

For organizations with AI agents, internal microservices, or APIs that do not traverse the Akamai edge, these conditions will not hold.

THE COMBINED PICTURE

Edge security and agentic security solve different problems. Akamai protects the perimeter with strong WAF and traffic-based API detection capabilities. Salt secures the full agentic fabric — including everything agents do after crossing or bypassing the perimeter. In environments with AI agent deployments, the post-perimeter action layer is where risk is realized and where Akamai's coverage ends.

SEE THE ACTION LAYER FOR YOURSELF

Salt Surface can scan your environment today with zero deployment and show you every API, MCP server, and agent endpoint visible to an attacker — including the ones Akamai has never seen. **Request a demo at salt.security**

SALT