SALT

SOLUTION BRIEF • COMPETITIVE ANALYSIS

Salt Security vs. Traceable (Harness)

Traceable requires instrumentation to see your APIs. AI agents connect to the ones you have not instrumented yet. Salt Security covers all of it from day one — no agents, no tracing projects, no deployment overhead.

CATEGORY	PLATFORMS COMPARED	YEAR
Agentic Security	Salt Security & Traceable (Harness)	2026

OVERVIEW

Instrumentation dependency creates the blind spots attackers exploit

Traceable builds its security picture from OpenTelemetry-based tracing agents deployed into your services. It can only see APIs that have been instrumented. Shadow APIs, legacy systems, external endpoints, and MCP servers that agents connect to without prior security review all exist outside that boundary — and those are precisely the assets attackers target first.

Acquired by Harness in March 2025, Traceable is now bundled inside a broader DevOps platform, complicating standalone procurement. Salt Security requires no instrumentation rollout, no tracing agents, and no deployment project. Coverage starts across your full environment from day one — external exposure, cloud infrastructure, code repositories, and runtime.

0

Tracing agents or sidecars required. Salt covers your full API fabric with no instrumentation project.

100%

API coverage including uninstrumented services, shadow APIs, and legacy endpoints.

8

Years of production API security research. Traceable was acquired and bundled into a DevOps platform in 2025.

PLATFORM APPROACHES

Instrumentation-dependent coverage vs. full-fabric visibility from day one.

SALT SECURITY

Full coverage without instrumentation

Salt Security requires no tracing agents, platform agents, or instrumentation rollouts. The Agentic Security Graph discovers and correlates LLM connections, MCP servers, APIs, identities, and sensitive data across code, cloud, and runtime — covering shadow APIs, uninstrumented services, and every agentic connection from day one.

Salt Surface discovers external exposure with zero deployment. Salt Collect monitors live runtime traffic across 70+ technologies. Salt Code governs pre-production. The result is immediate, comprehensive coverage that Traceable's instrumentation model takes months of rollout projects to approximate — and never fully achieves.

TRACEABLE (HARNESS)

Instrumented API telemetry and security

Traceable deploys OpenTelemetry-based tracing agents into your services to collect telemetry. Its security analysis is built on that instrumented data — behavioral detection, sensitive data tracking, and API discovery are all dependent on having tracing coverage in place.

The constraint: coverage is bounded by what has been instrumented. Shadow APIs, external endpoints, legacy systems, and MCP servers created without security review are invisible. And as a 2025 Harness acquisition, standalone procurement is increasingly complex — requiring engagement with a broader DevOps platform purchase.

THE COVERAGE GAP

Traceable sees what gets traced. Salt sees what exists. The APIs agents connect to without prior registration, the shadow MCP servers developers create outside governance processes, and the legacy endpoints attackers map first are all visible to Salt from day one. Traceable's instrumentation model cannot reach them by design.

CAPABILITY COMPARISON

Salt Security vs. Traceable (Harness): 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** Traceable discovers APIs within instrumented services. Uninstrumented legacy systems, shadow APIs, and external endpoints are outside discovery scope.
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** Traceable has no Agentic Security Graph. Correlated LLM, MCP, API, and identity visibility in a unified model is not available.
Salt Code Governance	**STRONG** Governs API and MCP creation in repositories and developer workflows before risky logic ships to production.	**NOT AVAILABLE** Traceable does not govern API and MCP creation in developer repositories pre-production.
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.	**PARTIAL** Traceable provides alert routing and integrations. Automated runtime-to-code remediation feeding AI coding assistants is not a primary capability.
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** Traceable correlates traces within instrumented services. Identity-aware sequence correlation across uninstrumented services and external MCP connections is not available.
Behavioral Action-Layer Protection	**STRONG** Detects machine-speed business logic abuse beyond signatures, schemas, or known patterns.	**STRONG** Traceable provides strong behavioral anomaly detection within instrumented API traffic using ML-based analysis.
Internal & East-West Coverage	**STRONG** Protects internal APIs and downstream service interactions that edge-only and model-only tools miss entirely.	**PARTIAL** Traceable monitors east-west traffic within instrumented microservices. Services without tracing agents deployed are not covered.
Action-Layer Data Security	**STRONG** Maps sensitive data in motion across APIs, MCP servers, and agent actions across the full fabric.	**STRONG** Traceable provides sensitive data tracking and PII detection within instrumented traffic — a genuine strength within the instrumentation boundary.

CAPABILITY	SALT SECURITY	COMPETITOR
No Tracing or Platform Agents Required	**STRONG** Salt delivers full agentic security without deploying tracing agents, sidecars, or platform agents — eliminating a complex, multi-team deployment requirement.	**REQUIRES AGENTS** Traceable requires OpenTelemetry-based tracing agents deployed into each service to generate the telemetry its security analysis depends on.
Coverage Beyond Instrumented Traffic	**STRONG** Salt finds shadow, legacy, unauthenticated, and public-facing APIs without relying on tracing coverage — the APIs nobody is watching because they were never instrumented.	**NOT AVAILABLE** Traceable's security analysis is bounded by instrumented coverage. APIs, services, and MCP servers without tracing agents deployed are invisible.
Faster Time to Value	**STRONG** Salt Surface delivers external exposure visibility with zero deployment. Full runtime coverage adds without instrumentation projects or service team coordination.	**SLOW TIME TO VALUE** Meaningful Traceable coverage requires instrumenting services across your environment — a multi-team project that typically takes months to achieve broad coverage.

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 you need coverage from day one

Salt Surface can show you the external exposure of your agentic infrastructure today — before any deployment, agent rollout, or instrumentation project. Ask Traceable what they can show you on day one before any agents are deployed. The comparison will demonstrate the coverage difference immediately.

If uninstrumented services are a concern

Ask your Traceable team to show you the APIs and MCP servers your agents are connecting to that have no tracing agents deployed. Those are your unmonitored attack surface. Salt covers them automatically. Traceable cannot, by architectural design.

If the Harness acquisition affects your procurement

Traceable is now part of the Harness platform. If your organization needs a dedicated, standalone API and agentic security platform — not bundled into a DevOps suite — that procurement consideration is relevant to your evaluation. Salt is a purpose-built, standalone agentic security platform.

If behavioral detection in production traffic is needed

Traceable's behavioral detection within instrumented services is strong. The evaluation question is whether that instrumented boundary covers the APIs and services where your agents are actually operating. Shadow APIs, partner integrations, and MCP servers created by developers are typically outside that boundary — and that is where Salt adds coverage Traceable cannot match.

Key differentiators

SALT SECURITY IS THE STRONGER CHOICE WHEN…

- Full API and agentic coverage from day one — no instrumentation rollout required
- Shadow API, legacy endpoint, and uninstrumented service coverage is needed
- Agentic Security Graph correlation across LLM, MCP, and API layers is required
- Standalone API security procurement outside a DevOps platform bundle is preferred
- East-west coverage without tracing agents in every microservice is needed
- External attack surface discovery with zero deployment is a requirement
- Eight years of dedicated API security production experience matters

TRACEABLE (HARNESS) MAY BE CONSIDERED WHEN…

Traceable may be considered in a very narrow scenario:

- All critical services are already fully instrumented with OTel tracing agents and all relevant APIs are within that instrumentation boundary
- Strong behavioral detection within the instrumented perimeter is the primary requirement and no coverage of uninstrumented services or shadow APIs is needed

For organizations with any uninstrumented services, shadow APIs, or agents creating new connections at runtime, Traceable's boundary will not cover the highest-risk assets.

THE COMBINED PICTURE

Instrumentation-based and instrumentation-free security provide fundamentally different coverage guarantees. Traceable's behavioral detection within instrumented services is strong, but the instrumentation boundary creates structural blind spots that attackers exploit. Salt's coverage starts at the external attack surface and extends through every API regardless of instrumentation state — providing the comprehensive visibility that agentic environments require.

SEE THE ACTION LAYER FOR YOURSELF

Salt Surface can scan your full agentic environment today with zero deployment, showing you APIs and MCP servers your agents are connecting to — including the ones with no Traceable tracing agents deployed. **Request a demo at salt.security**

SALT

