ML LABS INTELLIGENCE

Security Hardening for a Production AI Platform

A dual-track security assessment — Extra Security's penetration testing alongside ML LABS' AI-augmented review — surfaced 34 findings on a production AI platform. Every finding was closed with zero downtime.

CASE STUDY

Author Omar Trejo

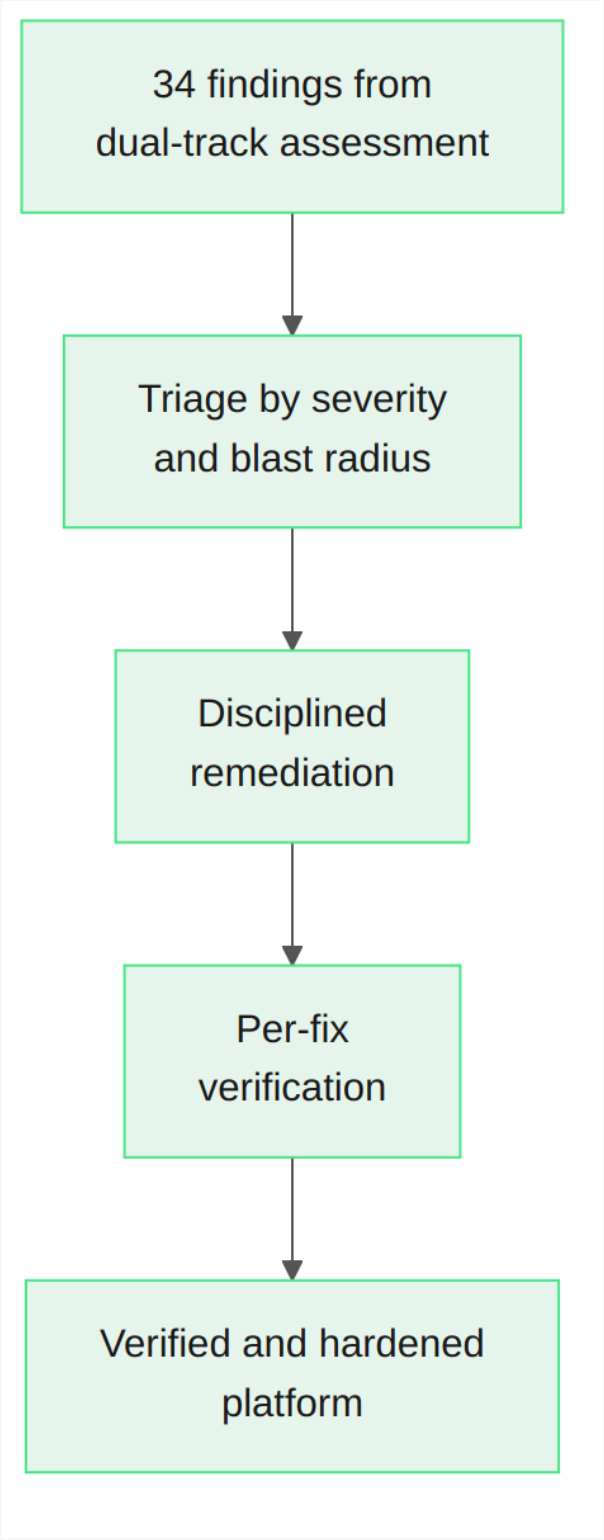
Date 2026-02-04

A regulated SaaS platform processing sensitive data across multiple organizations needed a comprehensive security assessment. The client engaged **Extra Security** for an external penetration test and ML LABS for an independent AI-augmented security review and remediation. The system had real workloads, real users, real data flowing through AI inference pipelines.

The dual-track approach surfaced 34 findings across authentication, input handling, session management, data redaction, network configuration, and infrastructure — 18 from ML LABS' AI-augmented review and 16 from Extra Security's penetration test, with each track catching vulnerabilities the other missed. This engagement is a strong example of how specialized vendors can collaborate effectively for a client's benefit — Extra Security brought deep offensive security expertise, ML LABS brought AI platform knowledge and AI-augmented execution, and the client got broader coverage and faster closure than either approach alone. Every finding was remediated or dispositioned without disrupting production operations.

How the Hardening Sprint Worked

ML LABS triaged every finding from both assessment tracks by blast radius, remediated in priority order, and verified each fix individually. No finding was marked closed until a corresponding test passed — a test designed to fail against the original vulnerability and pass only with the fix applied.


34 findings from dual-track assessment
Triage by severity and blast radius
Disciplined remediation
Per-fix verification
Verified and hardened platform

The ordering mattered. Authentication gates everything else — a system with perfect input validation is still compromised if an attacker can authenticate as a legitimate user. Data handling determines what leaks if other defenses fail. Network configuration locks the perimeter after the interior is sound.

Authentication and Session Hardening

The highest blast-radius findings involved credential storage, session management, and authentication failure handling. Credential storage was upgraded to industry-standard adaptive methods, session exposure was reduced to the minimum necessary window, and rate limiting was enforced across all authentication endpoints. Per-organization authentication isolation ensured that automated credentials could not be reused across tenants, and cross-tenant access paths were eliminated.

The most critical fix addressed silent authentication failures — validation code that could fail without raising an error, allowing downstream processing to continue in an unauthenticated context. Every validation path was rewritten to fail explicitly. Ambiguity in authentication state was eliminated entirely.

> The AI agents caught critical vulnerabilities the human testers missed. The human testers caught critical vulnerabilities the AI agents missed. Neither alone would have been enough.

Input Handling and Data Redaction

All content rendering surfaces were hardened against injection attacks, and XML parsing endpoints received entity expansion protection. Authorization enforcement was tightened so that every endpoint validates not just identity but permission level and resource ownership.

The data redaction overhaul was the most consequential change. The existing approach failed silently every time the schema expanded: new fields containing sensitive data would pass through unfiltered. ML LABS inverted the entire redaction model so that nothing passes through unless explicitly marked safe. New fields are blocked by default until reviewed. The failure mode shifted from silent data leakage to overly aggressive filtering — a problem that surfaces immediately rather than compounding invisibly.

Network and Perimeter Controls

Security headers were added across all API responses. CORS policies were tightened to allow only specific production origins. Open redirect vulnerabilities were closed. Infrastructure configuration was consolidated into a single source of truth with automated drift detection, preventing manual changes from weakening the perimeter after hardening. A web application firewall provided the outer perimeter layer.

AI-Augmented Security

ML LABS ran an independent AI-augmented security assessment in parallel with Extra Security's penetration test. The AI agents systematically scanned every file for dangerous patterns, mapped every authentication surface, traced attack chains, checked dependency vulnerabilities, and flagged compliance gaps against ISO-27001 and SOC-2. The two tracks produced meaningfully different finding sets.

The AI agents caught vulnerabilities that the human pen tester did not — including a critical code injection path, authentication functions that returned errors as values instead of raising exceptions, and exception handling that silently swallowed security-relevant failures. These are the kinds of issues that systematic, exhaustive code scanning excels at: patterns that are easy to miss when a human is focused on attack chains and business logic exploitation. Conversely, Extra Security's penetration testing surfaced critical findings that the AI agents missed, including complex authorization bypass chains and protocol-level session forgery attacks that required deep domain expertise and creative adversarial thinking to discover.

The combination — running both in parallel and merging the findings — compressed what would traditionally be months of security work into days, while achieving broader coverage than either approach alone.

Results

- Inverted redaction eliminated silent data leakage
- 34 findings remediated and verified — zero downtime
- Per-organization authentication isolation across all tenants
- Preliminary ISO-27001 and SOC-2 compliance verified

First Steps

1. **Run parallel assessment tracks.** Combine AI-augmented code scanning with human penetration testing. Each catches what the other misses — neither alone provides full coverage.
2. **Fix authentication first.** Auth weaknesses amplify every other vulnerability class. Harden credential storage, enforce rate limiting, reduce session exposure, and ensure every validation path fails explicitly rather than silently.
3. **Invert your data redaction model.** Define the set of fields explicitly approved for each output context and block everything else by default. This single change eliminates the most dangerous category of silent failures.

Practical Solution Pattern

Most production AI security failures are known vulnerability classes with known fixes — the gap is execution discipline, not knowledge. Execute hardening as a structured sprint with concentrated ownership: one operator who holds the full security picture, not findings distributed across a feature team where no one sees the whole. Every fix ships with a test that would have caught the original vulnerability.

The optimal structure pairs a specialized auditor like **Extra Security** for the offensive assessment with an AI-augmented operator for rapid remediation. If your platform handles sensitive data and has not had a structured security pass, an **AI Technical Assessment** can surface the findings before an external audit does.