CRASH OVERRIDE

Engineering Relationship Management for Security

Don’t Chase Shadows, See What’s Real, In Real Time

Crash Override’s Engineering Relationship Management (ERM) platform gives security teams the clarity to identify risk early, act decisively, and reduce waste. It connects your code, tools, people, and environments into a single, trusted source of truth, so you can focus on what matters now, next, or never.

From Blind Spots to Blueprints

Engineering environments are fragmented by default. Data lives in silos, ownership is unclear, and shadow engineering thrives. When a vulnerability hits, security teams scramble, often relying on spreadsheets, outdated inventories, and assumptions about where code lives and who owns it.

According to CrowdStrike’s 2024 State of Application Security, 74 percent of teams still rely on documentation to track applications, and 68 percent use spreadsheets. That manual overhead creates dangerous lag when things go wrong, with 30 percent of teams able to resolve critical incidents in under 12 hours. The rest stay exposed, creating a wide window for exploitation.

ERM changes that. Crash Override captures real-time telemetry during builds, flags unapproved container images and legacy frameworks, and maps every change across environments. It logs code lineage, deployment, and tool usage, so security teams can trace vulnerable services, isolate non-compliant components, and prioritize fixes with confidence.

What ERM Delivers

Crash Override inspects every build and deployment, recording who changed what, when, and how. It surfaces shadow systems, flags unapproved tools, and captures out-of-band activity that puts systems at risk. Whether it’s identifying rogue container registries, tracking down expired web certificates, or surfacing development in banned languages, ERM makes invisible visible.

Code provenance is crystal clear, untracked infrastructure is exposed, tooling is standardized, and audit-ready logs back every decision.

Real Results from Real Teams

Global Auto Manufacturer

With thousands of developers and tens of thousands of repositories, this team had scale, but not visibility. Security couldn’t trace ownership, track changes, or enforce standards. After adopting Crash Override, they restored traceability, aligned teams, and focused response on what mattered most.

What if you could map every deployment to its owner? Detecting drift before it becomes dangerous? Move from reactive to proactive?

Global Cybersecurity Provider

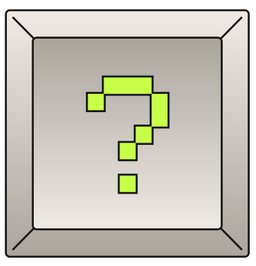
This security vendor was losing time and budget to alert fatigue and manual tracking. Risk was managed in spreadsheets and developers bypassed policies. With Crash Override, they integrated source-level security, eliminated redundant tools, and cut through the noise to focus on real threats.

How much time could you save by filtering out noise automatically? What could you secure with fewer tools and clearer signals?

Speed Is the New Attack Surface

Software moves fast and that means so does risk. AI-generated code, tool sprawl, and siloed processes expand the attack surface faster than most teams can track. Unauthorized services spin up, expired web certs go unnoticed, and container drift erodes consistency between environments.

Crash Override gives security a front-row seat to every change, every build, every deployment. You see what’s live, what’s broken, and what’s next. You cut through noise, streamline compliance, and eliminate waste, all without slowing delivery.



Ready to See It?

We don’t oversell or overpromise, we listen. Then we show you what Crash Override can do to strengthen your security posture, without the overhead.

Book a Demo