CASE STUDY

CS-2026-12

ioSENTRIX

Your Trusted Partner in Cybersecurity



AI-ASSISTED DEVELOPMENT SECURITY

How AI-assisted coding changed security risk

An enterprise software company measured the effect across three development environments.

CLIENT
Enterprise software company

SERVICE
Application security assessment

SCOPE
Three development environments

REMEDIATION
DevSecOps roadmap delivered

AT A GLANCE

CLIENT
Enterprise software company

ENVIRONMENTS
A mature platform, a moderately mature system and a fully AI-native application

SERVICE
Application security

SCOPE
Pull requests, static analysis, release-stage scanning and CI/CD pipelines

ALIGNMENT
ISO/IEC 25010, NIST SSDF and OWASP SAMM

REMEDIATION
DevSecOps roadmap delivered

DELIVERED BY
ioSENTRIX application security team

EXECUTIVE SUMMARY

AI sped up development, and in the weakest pipelines it sped up vulnerabilities too

An enterprise software company had adopted AI-assisted development across several teams, and leadership could not say whether it was making the code less secure. It asked ioSENTRIX to measure the effect in three environments: a mature platform, a moderately mature system and an application built entirely with AI tools.

We found that mature teams could adopt AI safely behind strong pull-request gates, that uneven pipelines let AI expose hidden weaknesses, and that in the AI-native application, pull-request vulnerabilities doubled.

3	2x	0
environments compared, from mature to AI-native	pull-request vulnerabilities in the AI-native app	security tools that covered every environment on their own

AI did not create the risk. It removed the time teams used to catch it.

Details in this case study are anonymised and generalised. Methods that could harm other products are withheld.

© 2026 ioSENTRIX LLC

01 What we found

THEME · AI-NATIVE FINDING 1

The AI-native app doubled its vulnerabilities

In the fully AI-native application, complexity and oversized functions surged, duplication rose and maintainability fell. Pull-request vulnerabilities doubled, with high-severity issues rising sharply.

Business impact: without governance, AI-native development was building technical debt and security exposure at the speed of the AI.

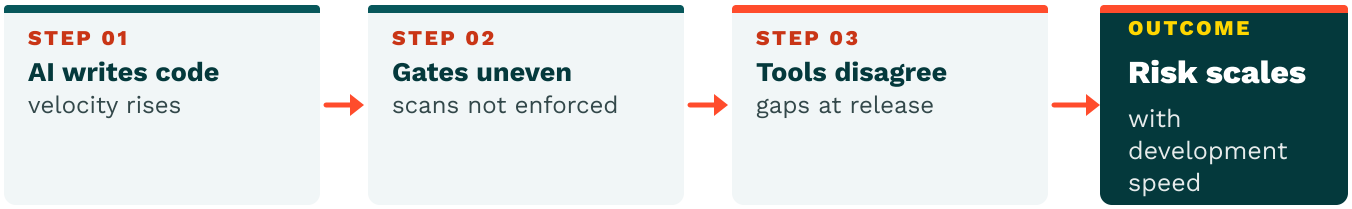
THEME · TOOLING FINDING 2

No single tool saw everything

Pipelines differed in how they enforced scans, with missing dependency (SCA) coverage and limited container controls. Across the security tools, coverage and severity ratings diverged.

Business impact: vulnerabilities could pass through one stage unseen, and leadership had no single view of the real risk.

Figure 1. Where AI-assisted development let risk through



HOW WE FOUND IT

The question was not whether one scan finds a bug, but how risk changed across environments over time. Answering it took analysis at the pull request, at release and in the pipelines, and comparing several tools against each other.

© 2026 ioSENTRIX LLC

02 Results

Figure 2. What we found in each environment, and what we recommended

Roadmap delivered; adoption owned by the client

ENVIRONMENT	WHAT WE MEASURED	WHAT WE RECOMMENDED
Mature platform	Quality stable; security findings up modestly, mainly in authentication and authorisation logic	• Adopt AI behind strong pull-request security gates
Moderately mature	Security findings up significantly in service and orchestration layers; tools disagreed	• Pipeline fixes and a map of high-risk modules
AI-native app	Quality fell; pull-request vulnerabilities doubled; high-severity issues rose sharply	• A governance roadmap with layered controls
CI/CD pipelines	Uneven scan enforcement, missing SCA coverage, limited container controls	• Fixes mapped to pipeline stages, repository types and vulnerability classes

03 Lessons for engineering leaders

AI speeds up risk too

In the AI-native app, vulnerabilities doubled at the pull request. Mature the SDLC as fast as you adopt AI tools.

Finding 1 · NIST SP 800-218, Secure Software Development Framework (SSDF) Version 1.1

Gate at the pull request

The mature team adopted AI safely because its pre-merge gates held. It is the cheapest place to catch AI-written flaws.

Approach

Use more than one tool

No single scanner covered every environment, and their severity ratings diverged. Compare tools before trusting one.

Finding 2

ioSENTRIX

Adopting AI coding tools? See a sample assessment or book a strategy session on measuring their effect on your code.

iosentrix.com/contact-us

contact@iosentrix.com

+1 (888) 958-0554

13800 Coppermine Rd, Suite 190
Herndon, VA 20171

CREST-accredited · ISO/IEC 27001 certified · SOC 2 Type 2 attested

© 2026 ioSENTRIX LLC. General information only. Client details anonymised.