AI × APPLICATION SECURITY × INSURANCE

AI-written code is outrunning insurance security teams

Regulators and cyber underwriters moved on AI-generated code before most security teams did. This teardown maps the forces now converging on insurance security programs, and what the carriers ahead of the curve are already doing about it.

Four forces converged on insurance security in one planning cycle:

REGULATION	AI CODE	EXPLOIT WINDOW	VOLUME
3 days	**62%**	**<1 day**	**10×**
federal deadline for the highest-risk fixes, cut from 15 (CISA BOD 26-04)	of latest-model code carries an exploitable flaw (Cycode 2026)	disclosure to exploitation, down from 2.3 years in 2019 (CSA)	more security findings in six months, same repos (Apiiro)

About 42% of the code committed inside large enterprises is now written or assisted by AI, and roughly three in four developers reach for an AI tool every day. In independent 2026 testing, most of what the current models produce ships with an exploitable flaw. Insurance is absorbing that code straight into the systems that underwrite policies, pay claims, and hold customer data.

> The code arrives faster, carries more flaws, and leaves less time than ever to fix any of it.

It comes at the worst possible moment. The time between a vulnerability going public and its first exploitation, once measured in months, is now measured in hours. Regulators and cyber underwriters noticed before most security teams did.

This spring the Cloud Security Alliance, with SANS and the OWASP GenAI project, put numbers to it. AI-driven vulnerability discovery now outruns the programs built to contain it. Their prescription is a security program organized around machine-speed remediation and a permanent **VulnOps** function. Here is what that shift means for a carrier's security program.

REGULATION

Regulators put a clock on remediation

The rule that binds insurers most directly is European. Under **DORA**, in force since January 2025, an EU-authorized insurer or reinsurer must scan its critical systems for vulnerabilities **at least weekly**, set deadlines to install fixes, and escalate when it misses them. Penalties reach **2% of global annual turnover**.

New York goes at the person rather than the program. NYDFS requires an annual certification of timely, risk-based remediation, signed by both the most senior executive and the security chief. "We are working on it" no longer clears the bar; someone now puts their name to the claim that it is handled.

The federal benchmark keeps tightening. In June, CISA's **Binding Operational Directive 26-04** cut the clock for the highest-risk vulnerabilities from 15 days to **3 calendar days**, with mandatory forensic triage, and set the next tier at 14. It binds federal agencies rather than carriers, but it is where examiners and underwriters are heading, and CISA cited AI-accelerated exploitation as the reason. Its first enforcement case, an Ivanti gateway flaw, carried a three-day deadline because attackers were backdooring exposed systems within 40 hours of a public exploit.

THE REGULATORY CLOCK · 2023 → 2027

"We are working on it" stopped being an answer.

Dec 2023 PASSED

NAIC Model Bulletin on AI adopted; named senior accountability, now live in about half of US states.

Oct 2024 PASSED

NYDFS AI cyber guidance atop timely, risk-based remediation and dual CEO/CISO certification.

Jan 2025 IN FORCE

EU DORA — weekly scans, patch deadlines, escalation; up to 2% of global turnover.

Dec 2025 PASSED

Colorado life-insurer model testing; second annual attestation deadline in force.

Jun 2026 ISSUED

CISA BOD 26-04 — highest-risk fixes cut to **3 calendar days** (from 15), next tier 14, with forensic triage. Federal, but the benchmark everyone now cites.

End 2027 PROPOSED

EU AI Act high-risk obligations for life/health pricing & risk assessment. Under revision via the Digital Omnibus; not yet final.

The regimes do not overlap cleanly: DORA covers EU-authorized carriers; Colorado and the EU AI Act name life and health; NAIC and NYDFS are broader; CISA is federal. No carrier has been publicly sanctioned under the AI-specific provisions yet. That makes the first enforcement action the reference case for everyone else.

THE CODE

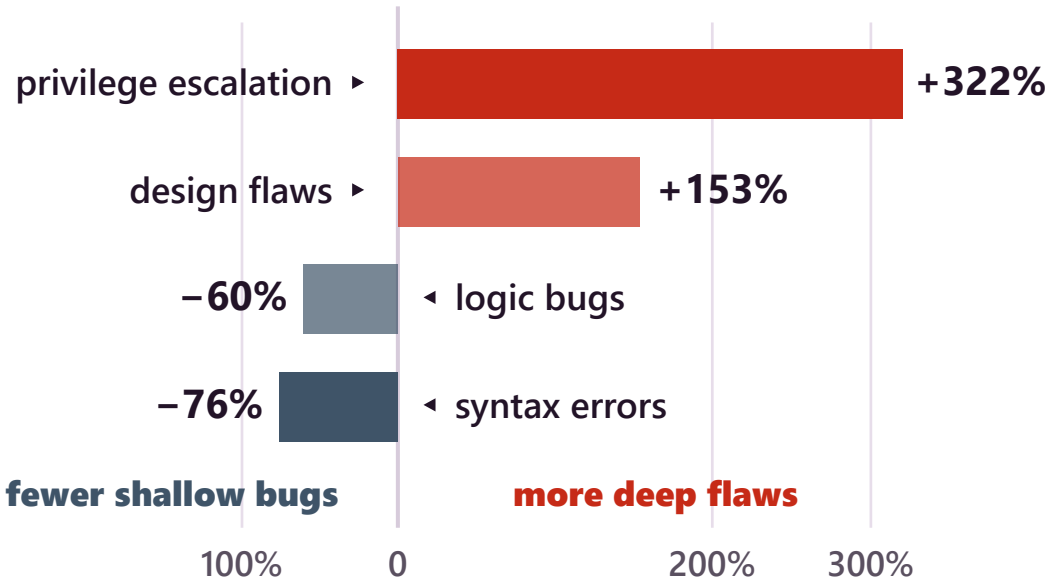
The code changed

For most of twenty years, application security lived with a gap between how fast teams could find vulnerabilities and how fast they could fix them. Finding got cheaper every year; fixing stayed slow and human. AI has pulled both ends of that gap apart at once.

The code entering the pipeline carries more flaws, and it is not improving. Veracode re-ran its generative-code security test in March 2026 and found the pass rate flat at roughly 55%, with about **45% of AI-generated samples still carrying an OWASP Top-10 flaw** even as the models got better at writing code that works. Cycode's number is starker. **62% of code from the latest models contains at least one exploitable vulnerability**, every company it surveyed already has AI-generated code in production, and 81% of security teams cannot see where that code is. When the work moves to autonomous agents it gets worse: a Carnegie Mellon and Endor Labs benchmark found **87% of agent-generated code** contained at least one vulnerability.

APIIRO · PRODUCTION REPOSITORIES · DEC 2024 → JUN 2025

AI didn't just add bugs. It shifted the mix toward the dangerous ones.



Same six months, same repos: **3–4× faster commits, ~10× more security findings.** The shallow classes fell, while the classes that take real architectural work to fix (privilege escalation and design flaws) climbed. All four bars share one scale. Source: Apiiro, Sept 2025, re-confirmed in the CSA "Vibe Coding's Security Debt" note, Apr 2026.

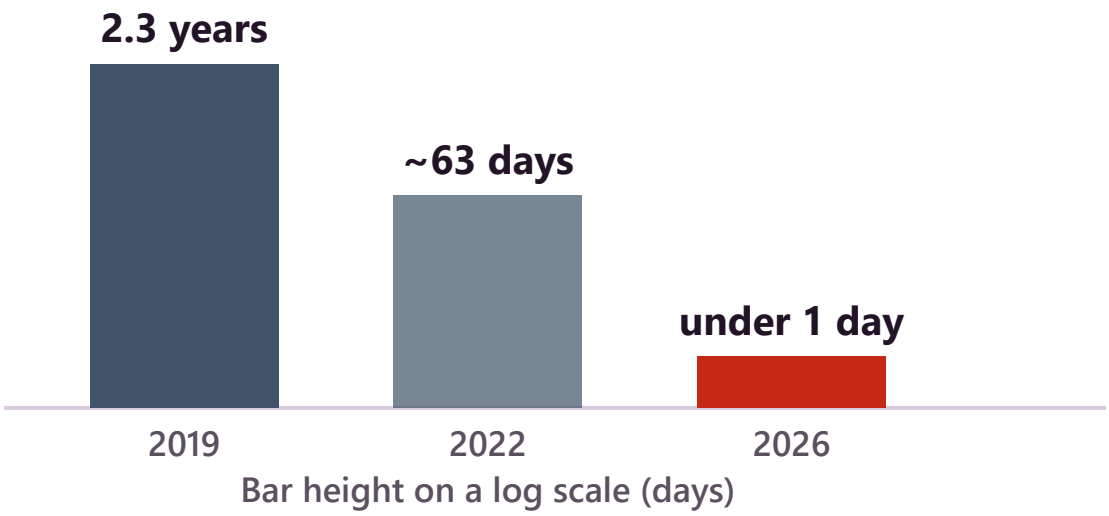
Volume compounds the defect rate. Opsera's 2026 benchmark across a quarter-million developers found AI-generated pull requests waiting more than four times longer for review, with code duplication climbing as more software gets assembled at speed rather than maintained. The obvious rebuttal is that the same AI can fix code as fast as it writes it. In part, true. But generation scales with raw compute, while safe remediation still needs a review loop that has not scaled the same way. The flawed volume lands faster than the verified fixes.

THE WINDOW

From years to hours

ZERO DAY CLOCK · DISCLOSURE → EXPLOITATION

The time to fix a flaw before it's used has collapsed.



The CSA's Zero Day Clock puts the gap between disclosure and exploitation at about **2.3 years in 2019 and under a single day in 2026**; Mondoo's recent read is 63 days down to 5. CrowdStrike found **42% of exploited vulnerabilities in 2026 were hit before public disclosure**, and Verizon reports a median of zero days for critical edge devices. The Zero Day Clock is a leading indicator, not a measure of realized damage; cited as such.

Researchers have separately shown a frontier model can read a vulnerability's public description and write a working exploit for most of them without human

help. That is a lab capability rather than the dominant real-world pattern today, but it points where the curve is going. So the old gap did not just widen. The volume and severity of what enters the codebase is rising while the time to resolve any of it falls toward zero. For a carrier putting AI into the core of underwriting and claims, that is happening inside the systems that hold its customers.

Who this hits hardest depends on how much software you write yourself. A carrier with a real engineering organization shipping first-party code carries the full exposure. A regional carrier running mostly vendor policy-administration software carries less of it directly, but inherits much of it through the supply chain. That is how the MoveIt breach, a single flaw in one shared file-transfer tool, reached a roll-call of insurers at once.

THE RESPONSE

Another scanner won't help

The reflex, when findings pile up, is to buy better detection. Modern platforms have already done real work here; posture management, reachability analysis, and cross-scanner deduplication all exist to cut the noise before a human sees it, and they work.

But ranking a list is not the same as resolving it. Datadog's 2026 research found only **18% of findings rated "critical" stay critical** once runtime context is applied, and the average organization now absorbs more than 800,000 security alerts a year. JFrog found two-thirds of CVEs have low real-world applicability. Most carriers already surface far more genuine, ranked risk than they can act on, and a smarter scanner bolted onto a program that cannot resolve what it sees only turns the dashboard redder.

Your peer set, from the data: our own analysis of **184 open security roles across 38 insurance carriers** (Sept 2025–Jul 2026). Directional, concentrated in the largest carriers, and the one house-generated data point here; everything else is third-party.

THE COMMON STACK

SonarQube is the most-named scanner, with GitHub Advanced Security, JFrog Xray, and Checkmarx / Snyk / Veracode close behind. **GitHub Copilot** now appears in carriers' own security job postings, including at several of the largest carriers.

THE ROLE MIX

Application and cloud security dominate, alongside SOC and GRC. Openings cluster in a few large carriers; the long tail hires in ones and twos. Team size isn't in postings, so open roles per carrier is the honest proxy.

AI-SECURITY, STILL NASCENT

About 8% of postings mention AI, concentrated at one top-10 auto insurer standing up a dedicated AI-security function. Prompt-injection and AI-governance roles are only now appearing.

AI is already writing code in the same shops that are still building the function meant to secure it. One carrier states the goal outright, to automate its application security policies while preserving developer experience, with an agent in the pipeline that sorts and routes the critical findings to developers. Judging by what they are staffing, the work carriers want to speed up is the deciding and the fixing, not the finding.

WHERE IT GOES

From detection to decision to design

The first move is to treat resolution as its own operations discipline, the permanent VulnOps function the CSA prescribes, a function that runs after detection the way operations runs after development. It decides which findings are actually exploitable, clears the noise automatically, and produces the fix as a reviewed change rather than a ticket, fast enough to matter against a closing window.

How far that reaches today is worth being honest about. Automated remediation is strongest on the shallow, patternable classes that AI is already reducing, and weakest on the privilege-escalation and design flaws it is multiplying, which still need human architectural review. So the second move is earlier. As coding agents write more of the software, the specification becomes the place security is won or lost, because a flaw in a prompt or a design assumption propagates into everything generated from it. For a carrier whose products now include models making decisions about real people under regulatory scrutiny, review at the design stage is one of the few controls that scales with the volume of generated code.

Where a program actually stands

- **Can your remediation data back your next attestation?** Pull the real time-to-remediate distribution for your open KEV-listed criticals and set it beside the sentence a named executive is about to sign. If the data does not support the sentence, one of them has to change first.
- **Did review scale with AI-written code?** Get the share of code merged last quarter that was AI-assisted, then check whether review coverage grew with it, not headcount. If volume grew and coverage did not, that gap is your exposure.
- **What happens to a finding that misses SLA?** A documented exception and risk-acceptance workflow is the artifact auditors ask for before the SLA number itself, and the thing most programs cannot produce on demand.

The window is not reopening

The detection era solved the problem of its time. AI created a different one, and insurance holds the sharpest version of it, on the shortest clock and under the closest supervision. The three questions above are how a program finds out where it actually stands.

What insurance security leaders should be reading

Current sources on the forces above. Most of it is independent, third-party research.

The AI Vulnerability Storm: Building a Mythos-Ready Security Program — CSA / SANS / OWASP GenAI, Apr 2026. The VulnOps case and the Zero Day Clock.

Binding Operational Directive 26-04 — CISA. The primary text of the 3-day clock.

Spring 2026 GenAI Code Security Update — Veracode. AI code is not getting safer as models improve.

Product Security in the AI Era 2026 — Cycode. 62% of latest-model code carries an exploitable flaw.

Agent Security League 2026 — Endor Labs / Carnegie Mellon. Agentic code is worse, not better.

4× Velocity, 10× Vulnerabilities — Apiiro. Production telemetry on the vulnerability-class shift.

State of Vulnerabilities 2026 — Mondoo. Time-to-exploit from 63 days to 5.

LLM Agents Can Autonomously Exploit One-Day Vulnerabilities — Fang et al., UIUC. The leading indicator.

DORA RTS on ICT risk management (2024/1774), Art. 10 — the weekly-scan and patch-deadline text.

tl;dr sec — Clint Gibler's weekly on the AI-AppSec frontier.

SOURCES & METHOD

- AI-code share ~42%, ~72% of developers use AI daily: **SonarSource State of Code 2026** (Jan 2026).
- ~45% of AI-generated samples fail an OWASP Top-10 check, flat as models improve: **Veracode Spring 2026 GenAI Code Security Update** (Mar 24 2026). 62% of latest-model code has ≥1 exploitable vulnerability; 100% of surveyed companies have AI-generated code; 81% lack visibility: **Cycode, Product Security in the AI Era 2026**. 87% of AI-agent code contains ≥1 vulnerability: **Endor Labs / Carnegie Mellon, Agent Security League 2026**.
- 3–4× commit velocity → ~10× monthly findings (Dec 2024→Jun 2025); privilege-escalation +322%, design +153%; syntax −76%, logic −60%: **Apiiro** (Sept 2025), re-confirmed in the CSA note "Vibe Coding's Security Debt" (Apr 2026). AI PRs wait 4.6× longer for review: **Opsera AI Coding Impact 2026** (Mar 2026).
- Zero Day Clock 2.3 years (2019) → <1 day (2026): **CSA Mythos whitepaper** (leading indicator, cited as such). Time-to-exploit 63 → 5 days: **Mondoo 2026**. 42% of exploited vulns hit before public disclosure: **CrowdStrike Global Threat Report 2026**. Median 0 days disclosure-to-mass-exploitation for critical edge devices: **Verizon DBIR 2026**. Frontier model auto-writes exploits from CVE text (lab capability): **UIUC, Fang et al.** (2024, arXiv:2404.08144).
- Only 18% of "critical" findings stay critical after runtime context; ~865k average alerts per org: **Datadog State of DevSecOps 2026**. ~66% of CVEs have low real-world applicability: **JFrog 2026**.
- DORA weekly-scan + patch-deadline + 2%-turnover: **EU DORA RTS 2024/1774, Art. 10**, in force Jan 2025. NYDFS 23 NYCRR 500 §500.5/§500.17 + Oct 2024 AI guidance. NAIC Model Bulletin Dec 2023 (~half of US states). Colorado SB21-169 (life). EU AI Act Annex III (life/health), heaviest obligations proposed end-2027 — provisional, Digital Omnibus, not yet adopted. CISA BOD 26-04 issued Jun 10 2026 (federal); first enforcement Ivanti Sentry KEV.
- Insurance-specific: 37% of insurance-sector orgs report three breaches in 12 months, highest of any vertical: **Checkmarx, The Future of AppSec (2026 outlook)**. MoveIt = SQL-injection flaw in a shared third-party file-transfer tool.
- Hiring panel: **Pixee's own analysis** of 184 security postings across 38 insurance carriers (Sept 2025–Jul 2026); directional, concentrated — the one house-generated data point.

Closing this gap is operational work. Pixee publishes a free, ungated framework on it: Security Backlog Burndown — a CISO playbook.

Pixee publishes on how enterprises resolve vulnerabilities at machine speed. This teardown is industry analysis; the third-party data belongs to the sources above.

Pixee

ABOUT PIXEE

Pixee closes what your scanners leave open.

Triage & Fix runs across the 12+ scanners you already have. It decides which findings are actually exploitable, cutting false positives by up to **95%**, then ships each fix as a reviewed pull request your developers merge at a **76% merge rate**. A human approves every change.

Foresight works upstream of all of it: it reviews the design before an agent writes the code, then verifies the shipped code kept the plan. Both run on one context graph, so every fix sharpens the next design review and the backlog shrinks from both ends. All of it is GA today, on the stack you already own.

Teams at MoneyGram, Oracle, Standard Chartered, and Olympus run the Pixee platform.

See it run on your own repositories.

Book a working demo. No rip-and-replace — Pixee runs on the scanners you have.

pixee.ai/get-a-demo