bugcrowd

CASE STUDY

How CodeAI built a proactive security program with Bugcrowd

Industry:
Education technology (EdTech)/Nonprofit

Founding date:
2013

Number of employees:
100+

Headquarters:
Seattle, Washington

Website:
www.code.org

The situation

CodeAI is a nonprofit dedicated to ensuring that **every K–12 student has the opportunity to understand digital fluency, built on AI science, computer science, and data science.** Tens of millions of students and teachers rely on the platform daily, which means the personal information CodeAI stores, such as names, locations, and account credentials, carries significant weight. Student emails are fully anonymized and hashed, invisible even to CodeAI's own staff. Like other leading education companies, CodeAI maintains compliance with a demanding set of international, federal, and state regulations, including the GDPR, FERPA, COPPA, PPRA, SOPIPA, and HB 5469.

However, CodeAI's security program goes beyond what compliance requires. As an open-source platform serving students, CodeAI understands the risk of a breach: Even basic identity data sold on the dark web can impact a student for years. This becomes even more problematic in the current threat environment, where attackers can use AI to sweep across many targets at scale and uncover exploits to extract sensitive information. This puts CodeAI—a platform whose codebase is publicly available for examination—in an extremely vulnerable position.

From early on, CodeAI embedded security review checkpoints and automated regression testing into the software development life cycle. But point-in-time testing and internal review are designed to catch vulnerabilities in isolated pieces of software, whereas attackers often chain vulnerabilities across systems to exploit them. Therefore, **as CodeAI expanded its feature set and user base, the gap between what structured internal processes could surface and what a motivated external researcher might find grew wider.**

The challenge

> **To address security gaps, CodeAI first invested in penetration testing, which revealed some gaps but, for the most part, didn't uncover anything the team wasn't already aware of.**

What was missing was a deeper dive into the nuances of their specific application—probing how its features interact, how its data flows, and where its particular implementation might behave unexpectedly. These are the corner cases that general testing rarely reveals. Meanwhile, the infrastructure engineering team, already busy triaging incoming issues, lacked the capacity to uncover and investigate these cases for every new piece of software shipped.

Given the high stakes, the team looked to proactive security to strengthen its posture. **"In the age of AI-empowered cyber threats, attacks are easier and everyone is a target.** Students deserve to have their data protected and that's what drives us to continue investing in proactive, continuous security," says Darin Webb, Infrastructure Engineering Manager at CodeAI.

The Bugcrowd solution

In December 2018, CodeAI partnered with Bugcrowd to launch a private Managed Bug Bounty program. The goal was to **go beyond periodic penetration testing and tap into a global community of security researchers** who could go deeper into CodeAI's application and remediate vulnerabilities before attackers could exploit them.

In addition to continuous monitoring, Bugcrowd's managed triage helped CodeAI **identify issues that needed attention without adding extra load to the infrastructure engineering team's plate.** This allowed the team to identify which areas of their attack surface needed extra attention, helping them build a clear roadmap to improve their security posture. This also helped them with scaling their team—they used this roadmap to put together a clear business case to hire a dedicated security engineer who looks across product and infrastructure.

"Leveraging Bugcrowd took triage off our plate and gave us both the visibility into our real-world attack surface and the business case to invest in dedicated security resources," says Webb.

> **"Having that expertise in-house has changed everything. We're not just reactive anymore; we're building a real program."**

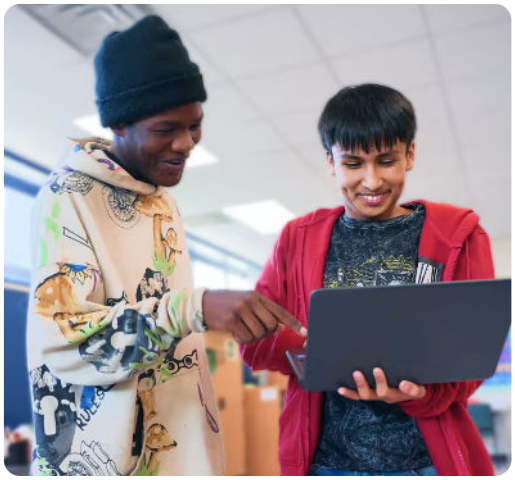
That mindset shift, from reactive to proactive, is built into how the program works. "The findings don't stop, and there are always things to discover and improve," says Webb. **"You're never truly done.** That reality, made visible through a continuous program, demonstrates why this is a critical piece of both proactive and reactive security maintenance, not a line item you can cut."

The outcome

Through their partnership with Bugcrowd, CodeAI has taken the leap into the next phase of their security journey. Instead of treating security as a reactive checklist item, CodeAI treats it as a first-class citizen, with a program that prioritizes preemptive security and has buy-in across the organization, which has real impact on student outcomes.

For example, a researcher recently discovered a request amplification vulnerability that allows an attacker to use a small number of requests to overload servers and cause outages. For a platform used daily by millions of students and teachers, that kind of disruption erodes the trust of the schools and districts that depend on CodeAI and directly interrupts student learning. By catching this issue early, CodeAI ensured platform uptime and interruption-free learning for millions of students.

This shift has also permeated across the organization. "CodeAI's security program is focused on making security a core competency across our entire team—not just within our dedicated security function. We want security embedded in how every engineer thinks and works. The goal is a culture where security isn't someone else's job," says Webb.



> **This sense of shared ownership shows up across how the teams work. When a vulnerability report comes in, CodeAI triages it and brings in the engineering team with the deepest expertise in the relevant area to implement a fix.**

Engineering and product teams now also consult with the security team when building new features, thinking through security considerations from the start rather than treating vulnerabilities as something to address after the fact.

Success snapshot

- ✓ 134 vulnerabilities rewarded through the Bug Bounty program
- ✓ Moved from one or two annual audits to always-on, continuous security testing
- ✓ Surfaced implementation-specific vulnerabilities that internal testing missed
- ✓ Freed the infrastructure engineering team from triage responsibilities, enabling investment in a dedicated security engineer
- ✓ Built a security-aware engineering culture with cross-team collaboration on every report

Products involved

Bug Bounty Programs