Futurum RESEARCH

Google Cloud



Remediation Is Becoming a Software Delivery Capacity Problem

AUTHORS

Vikram Rathnam
Mitch Ashley
Fernando Montenegro

OCTOBER 2026

KEY TAKEAWAYS

The Constraint Has Moved

Finding flaws is getting cheaper. The queue between a validated finding and an approved fix is now the capacity problem for security and engineering leaders.

The Evidence

A total of 75% of organizations had a production incident in the past year in which AI-generated code, agent actions, or AI tooling contributed, and 39% of those organizations identified vulnerabilities introduced by AI-generated code.[1]

The Operating Model

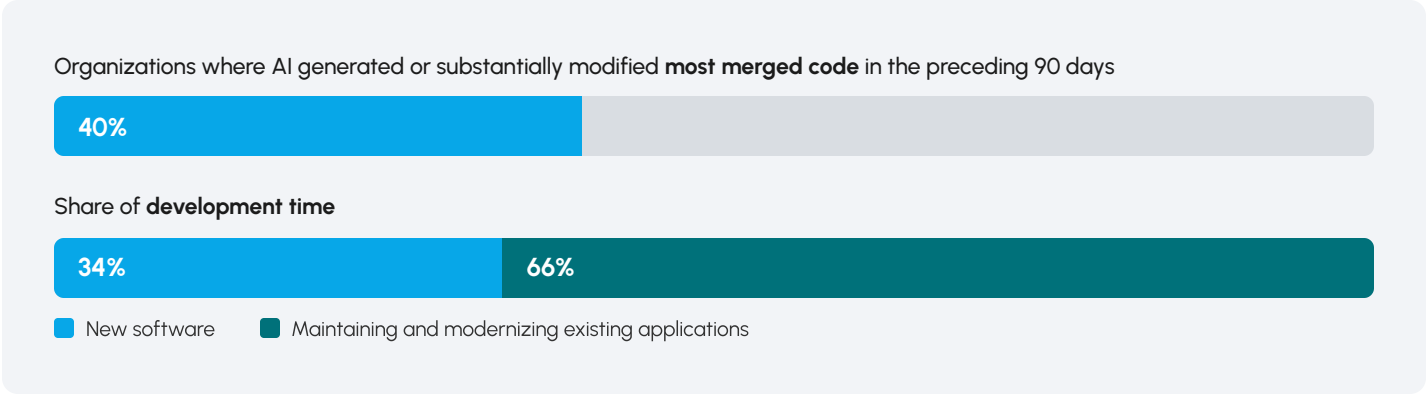
Set approval thresholds by change class, measure time to a validated fix, and require evidence from remediation agents, not finding counts.

The Bottleneck Has Moved From Finding Flaws to Fixing Them

The business problem is the queue between discovery and a safe production change: pull requests waiting on review, findings waiting on an owner, and patches waiting on a test run.

In Futurum's survey, 40% of organizations said AI generated or substantially modified most of the code they merged in the preceding 90 days. Yet only 34% of development time goes to new software; the rest goes to maintaining and modernizing existing applications (see Figure 1).[1] Faster code creation has not given review, testing, and maintenance teams more capacity.

Figure 1. *AI-Generated Code and the Allocation of Development Time*



Source: Data from *2H 2026 Software Lifecycle Engineering Global Enterprise Decision Maker Survey Report*. Futurum Research. *N* = 839 IT decision makers. The maintaining-and-modernizing share is the remainder of development time after new software.

Governments expect AI to help attackers find and exploit flaws at machine speed. The UK National Cyber Security Centre assesses that the time between disclosure and exploitation has already shrunk to days and that AI will almost certainly shorten it further.[2]

1. Futurum Research, *2H 2026 Software Lifecycle Engineering Global Enterprise Decision Maker Survey Report* (N = 839 IT decision makers).
2. National Cyber Security Centre. (2025, May 7). *Impact of AI on cyber threat from now to 2027*. https://www.ncsc.gov.uk/report/impact-ai-cyber-threat-now-2027

Scan-and-Triage Cycles Built for Human-Written Code Cannot Keep Up

Scan-and-triage cycles were built for code written by people at human speed. AI-generated code now arrives faster than those cycles can process it. Findings that arrive without proof that they can be exploited add to that queue, because someone has to sort real risks from false positives. When review and fixing happen outside the developer's workflow, verification debt builds up. Verification debt is the backlog of changes no one has yet confirmed are safe.

Controls have not kept pace. Only 43% of organizations mandate human review of AI-generated code, and 39% require security scanning built for AI-generated code (see Figure 2).[1] Neither number shows whether an organization can fix confirmed risk as fast as it ships software. The measure that does is time to a validated fix: the time from a validated, consequential finding to an approved, regression-tested change.

Figure 2. *The Control Gap: AI Exposure Versus Controls in Place*



Source: Data from *2H 2026 Software Lifecycle Engineering Global Enterprise Decision Maker Survey Report*, Futurum Research. *N* = 839 IT decision makers. AI-introduced vulnerabilities are shown as a share of organizations that reported an AI-involved production incident; all other values are shares of all respondents.

Security and Engineering Leaders Now Share the Fix

Code origination is spreading to everyone building with AI, while accountability for securing and fixing that code lands back on IT and security. Fixes stall when security and engineering work apart: security knows the risk, and engineering knows which behavior cannot break. The shift is already reaching security teams. Among cybersecurity decision makers asked about application security (n=93), 50% ranked moving AppSec staff from manual code review to designing automated AI guardrails among the top three ways AI-assisted development is changing the team.[3]

Security sets the approval threshold: the rule for which classes of change need human sign-off, what exploit evidence is enough, and when exceptions apply. Small fixes can skip review in the developer's local loop, but every change is still approved at the pull request. The pull request is where engineering checks that a change behaves correctly, so security verification and fixes need to happen before code gets there.

3. Futurum Research, *1H 2026 Cybersecurity Global Enterprise Decision Maker Survey Report* (N = 929 cybersecurity decision makers).

What Leaders Should Require of Any AI Code Remediation Agent

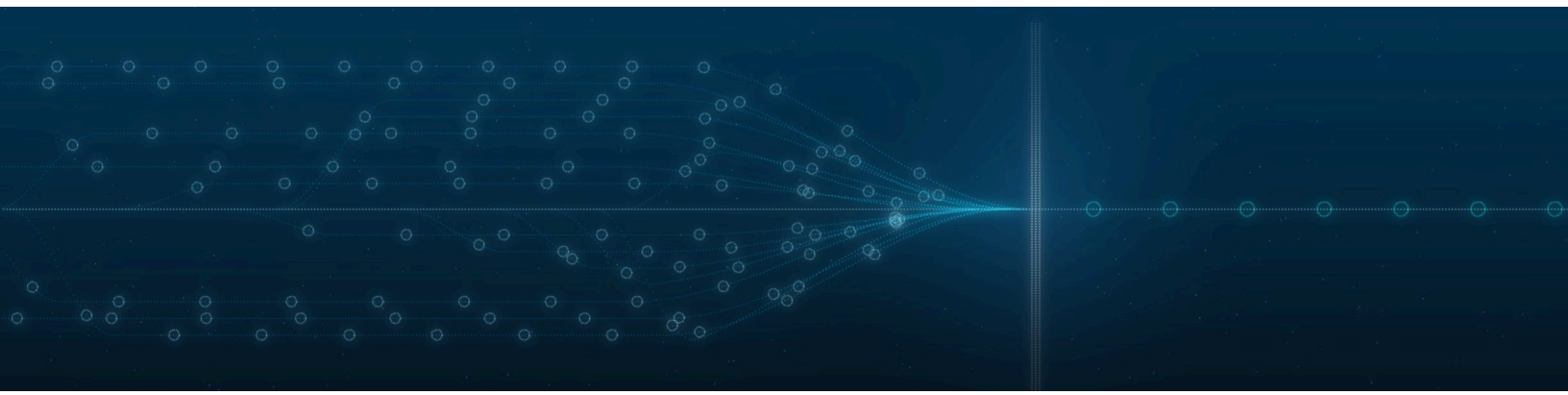
A useful remediation agent produces evidence that security and engineering can both act on. Humans lead the loop: people set the rules and design the system, and agents do more of the fixing. Six capabilities matter more than the number of findings on a dashboard (see Table 1):

Table 1. *Six Requirements for AI Code Remediation Agents*

Requirement	Evidence Produced	Question to Ask a Vendor
Exploit proof	Reproducible, sandboxed demonstration of the flaw	*Can our team rerun the exploit and inspect its assumptions?*
Regression-tested fix	Patch plus results from the team's own tests	*When a patch breaks behavior, does the pipeline block the merge, and who sees why?*
Approval threshold	Role-based approval rules and an audit trail that feeds existing audit and ticketing systems	*Which actions can the agent take before a developer signs off?*
Runtime context	Exposure, reachability, and business-criticality signals	*Can we prioritize what is reachable and exposed in production over everything the scanner flags?*
Least-privilege scope	Defined permissions for the agent and a record of blocked actions	*What can the agent access, and what stops it from exceeding that scope?*
Code handling and retention	Written terms for where code is processed, how long it is kept, and how it is deleted, plus a commitment not to train models on it	*Which parts of our code leave our environment, and can we delete them on demand?*

Source: Futurum Research, October 2026

Most organizations lack the controls these requirements assume. In Futurum's survey, 32% of organizations gate irreversible agent actions with human approval, and only 12% evaluate agent output before promotion (see Figure 2).[1]



How CodeMender Maps to These Requirements

CodeMender is an AI code remediation agent available on Google Cloud's Gemini platform.[4] It runs through a command-line client in local developer environments, works with tools such as VS Code and Antigravity, and can integrate with existing continuous integration and continuous delivery (CI/CD) workflows.[5] It supports generally available Gemini models, including Gemini 4 Argon, released on October 13, 2026. Through Google's Fairwind Program, vetted governments, critical infrastructure providers, and cybersecurity partners can also use Gemini 3.8 Flash Cyber and a version of Argon without cyber safeguards.[4,6]

> "CodeMender brings AI into a critical part of the security lifecycle by accelerating the path from validated vulnerability to tested fix."
>
> **Iain Mulholland**, CISO, Salesforce

AGAINST THE SIX REQUIREMENTS, AS OF OCTOBER 2026

Exploit Proof — MEETS IT

CodeMender builds and runs proof-of-concept exploits in a customer-managed sandbox.[4]

Regression-Tested Fix — THROUGH PIPELINE INTEGRATION

CodeMender runs its own sandbox tests to check that a patch preserves existing behavior. To run a customer's existing test suite, teams build CodeMender's command-line client into their CI/CD pipeline, where it tests patches before developers approve them.[5]

Approval Threshold — PARTIALLY

Developers approve every patch before commit.[4] Google's public materials do not describe exporting approval records to existing audit or ticketing systems.

Runtime Context — THROUGH WIZ

It comes from Wiz in Google's AI Threat Defense, not from CodeMender alone, and Google lists Wiz-triggered CodeMender scanning as coming soon.[4]

Least-Privilege Scope — MEETS IT

By default, a developer must approve every read and write action the agent takes. In pipelines, CodeMender runs in isolated containers and uses Workload Identity Federation to get short-lived, scoped service account permissions.[5]

Code Handling and Retention — LARGELY MEETS IT

CodeMender does not upload full source repositories to Google's servers and does not use customer code to train models. It retains session data, including code snippets, for up to seven days, and customers can trigger an immediate cleanup. Data is isolated by organization, and code can be compiled and tested in a local sandbox.[4]

4. Google Cloud. (2026). CodeMender overview. Gemini Enterprise Agent Platform documentation. Retrieved October 13, 2026.

5. Gerstenhaber, M., & Viernickel, C. (2026, July 21). *Now in preview: Find and fix software vulnerabilities with CodeMender.* Google Cloud Blog. https://cloud.google.com/blog/products/identity-security/find-and-fix-software-vulnerabilities-with-codemender

6. Gerstenhaber, M. (2026, October 13). Gemini 4 Argon available on Google Cloud. Google Cloud Blog. https://cloud.google.com/blog/products/ai-machine-learning/gemini-4-argon-available-on-google-cloud

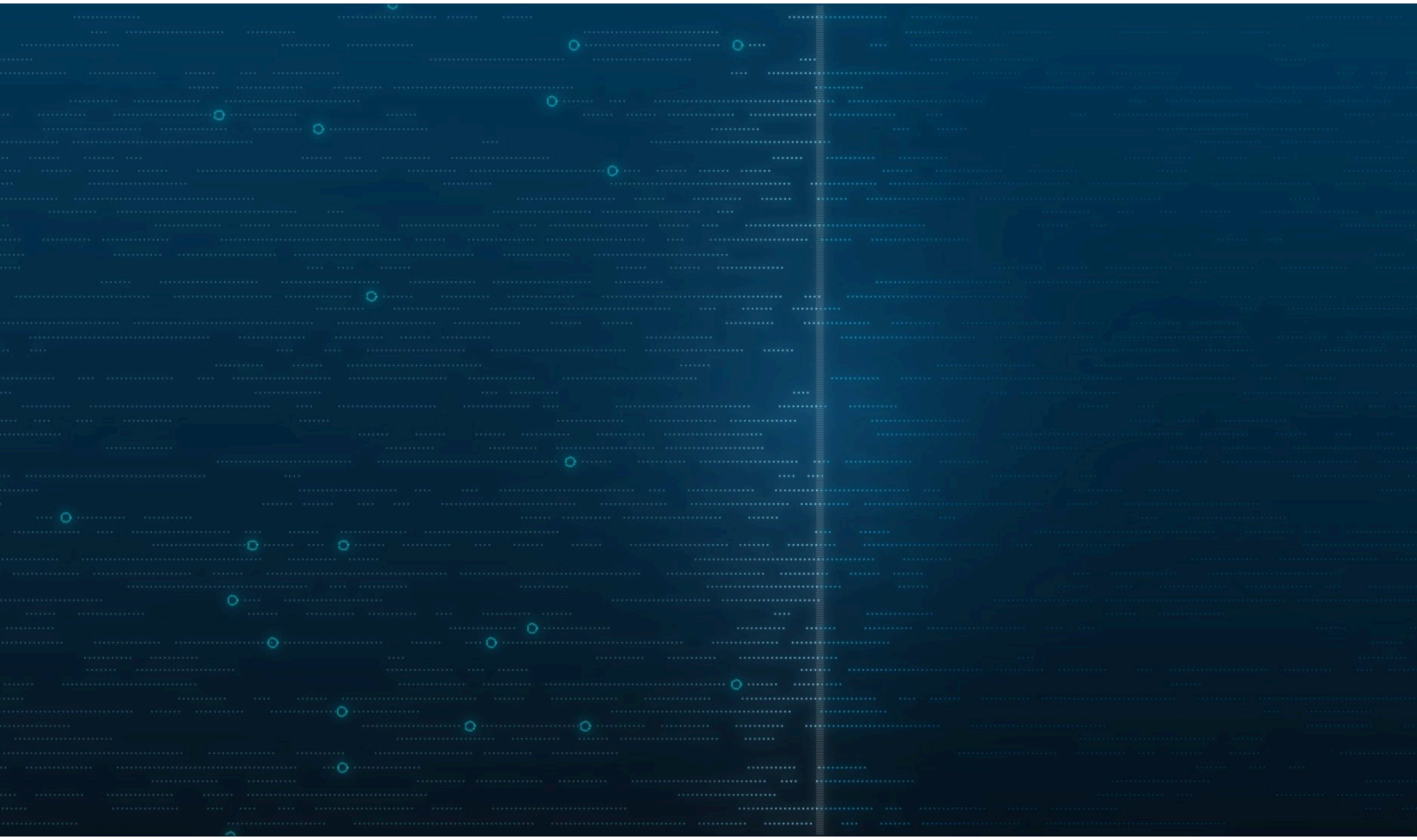
Recommendations

1. **Measure Time to a Validated Fix**
   Track it for every consequential finding, and report it alongside finding counts.

2. **Set Approval Thresholds by Change Class**
   Security defines which changes need human sign-off and what evidence is enough. Engineering applies those rules in the pipeline.

3. **Pilot a Remediation Agent on a Bounded Repository and Test It Under Failure**
   Security engineering and the platform team should test misleading exploit results, patches that alter application behavior, and attempts to exceed the agent's permitted scope.

4. **Require Evidence Artifacts, Not Finding Counts**
   Judge any remediation agent by the exploits, test results, and approval records it produces in your own repositories.



Important Information About This Report

AUTHORS

Vikram Rathnam
Research Director, Software Lifecycle Engineering | The Futurum Group

Mitch Ashley
VP and Practice Lead, CIO and Technology Buyers | The Futurum Group

Fernando Montenegro
VP and Practice Lead, Cybersecurity and Resilience | The Futurum Group

PUBLISHER

Futurum Research

INQUIRIES

Contact us if you would like to discuss this report and The Futurum Group will respond promptly.

CITATIONS

This paper can be cited by accredited press and analysts, but must be cited in context, displaying author's name, author's title, and "The Futurum Group." Non-press and non-analysts must receive prior written permission by The Futurum Group for any citations.

LICENSING

This document, including any supporting materials, is owned by The Futurum Group. This publication may not be reproduced, distributed, or shared in any form without the prior written permission of The Futurum Group.

DISCLOSURES

The Futurum Group provides research, analysis, advising, and consulting to many high-tech companies, including those mentioned in this paper. No employees at the firm hold any equity positions with any companies cited in this document.

Google Cloud

ABOUT GOOGLE CLOUD

Google Cloud offers a powerful, optimized AI stack—including AI infrastructure, leading models like Gemini, data management capabilities, multicloud security solutions, developer tools and platform, as well as agents and applications—that enables organizations to transform their business for the Agentic Era. Customers in more than 200 countries and territories turn to Google Cloud as their trusted technology partner.

Futurum RESEARCH

ABOUT THE FUTURUM GROUP

The Futurum Group is an independent research, analysis, and advisory firm, focused on digital innovation and market-disrupting technologies and trends. Every day our analysts, researchers, and advisors help business leaders from around the world anticipate tectonic shifts in their industries and leverage disruptive innovation to either gain or maintain a competitive advantage in their markets.

© 2026 The Futurum Group. All rights reserved.