Scribe

Scribe Use Case:

How Scribe Security Enables Organizations to Speed up and Innovate, Harnessing AI-Driven Development, Without Compromising Security

INVINCIBLE PRODUCT SECURITY

Additional information is available at https://scribesecurity.com/

Scribe

How Scribe Security Enables Organizations to Securely Harness AI-Driven Development Without Compromising Speed or Innovation

Introduction: The Problem of AI-Generated Code in Modern Software Development

The rapid advancement of artificial intelligence (AI)—particularly generative AI—has reshaped how organizations develop software. AI-powered coding tools like GitHub Copilot, The Cursor editor, OpenAI's GPT-based assistants, and numerous other automated code-generation tools are now commonplace, significantly boosting development speed and productivity. A recent GitHub survey revealed that more than **90% of developers** reported using AI coding assistants regularly (AKA "Vibe coding"), highlighting how pervasive AI-generated code has become.

Yet, this remarkable acceleration in coding velocity comes with critical drawbacks. The ease and speed of AI-generated code can unintentionally lead to the widespread introduction of vulnerabilities, misconfigurations, insecure patterns, and poor-quality code into software systems. While AI assistants can produce highly effective snippets, they frequently lack context about the organization's security standards, architectural guidelines, regulatory frameworks, or best practices for secure coding. Consequently, "bad" or risky code is entering software repositories at unprecedented volumes.

One consequence of AI-driven code assistants is that software organizations face bottlenecks in their code review cycles. Security and DevOps teams are overwhelmed by the sheer quantity of AI-generated code, leading to a troubling trade-off: either compromise on review quality or significantly slow down development to manage the growing backlog. This tension between innovation velocity and secure software delivery leaves organizations vulnerable. As developers and AppSec engineers are pressured by deadlines and the amount

of work, they might circumvent guidelines, allowing for lower-quality code to enter the main repositories, leading to the introduction of vulnerabilities and long-term quality degradation.

To safely leverage the tremendous innovation potential of AI-generated development and yet, ensure that while vide-coding becomes the norm, the quality of the produced code does not suffer, a new approach to software supply chain security and automated risk management is urgently needed.

Risks Introduced with Agentic Coders / Vibe Coding

Following is a partial list of governance and operational risks associated with Ai-generated code:

- **Responsibility Diffusion:** Unclear accountability between developers, agent creators, and platforms
- **Security Knowledge Decay**: Declining human security expertise as tasks shift to agents
- **Over-reliance Risk**: Critical security decisions delegated to agents without adequate oversight
- **Compliance Documentation Gaps**: Difficulty proving security controls when code generation is opaque
- **Licensing Violations**: Unintentional incorporation of incompatibly licensed code fragments
- **Limited Explainability**: Difficulty understanding security implications of complex generated code
- **Homogenized Vulnerabilities**: Identical weaknesses replicated across many codebases

---

Securely Harnessing AI-Driven Development with Scribe Security

Scribe Security is uniquely positioned to solve the pressing challenges introduced by AI-generated code. Unlike traditional application security testing (AST) tools—which rely on manual or periodic assessments—Scribe integrates seamlessly into development workflows,

providing continuous, automated security assurance tailored specifically for modern DevOps and AI-driven environments.

Automated Attestations Across Every Stage of the SDLC

Scribe's platform automatically generates machine-readable, cryptographically signed attestations for every stage of the Software Development Lifecycle (SDLC). When developers—whether experienced engineers or "citizen developers" using low-code platforms—introduce AI-generated code into repositories, Scribe captures essential security evidence:

- **Trusted Auditable Trail**:
  Scribe continuous signing and attestation technology provides an unequivocal audit trail of all coding and SDLC activities and serves accountability - there is an evidence trail to the code introduced, so the developer should better act responsibly.

- **Build & Release Verification**:
  Each build artifact and software release includes detailed, cryptographically signed evidence confirming compliance with predefined security policies and standards.

These automated attestations enable software organizations to confidently prove what each piece of AI-generated code came from and that it aligns with internal security standards and external regulatory frameworks, all without manual overhead or delays.

Enhanced Visibility & Trust in AI-Generated Code

Scribe Security offers comprehensive visibility into the software supply chain through detailed Software Bills of Materials (SBOMs) and in-depth provenance tracking. This transparency allows organizations to quickly trace, verify, and manage the security posture of every AI-generated component introduced into their products:

- **Comprehensive SBOM Generation**:
  Scribe automatically creates detailed SBOMs for every software artifact and every new software build, clearly indicating the origin of AI-generated code, dependencies involved, and potential vulnerabilities within these components.

- **Frictionless Visibility Expedites Delivery**:
  When AI generates a whole module, you can at least get a high-level view of what's inside it, even if you do not review the code itself.

By providing unprecedented transparency, Scribe mitigates the “black box” risk associated with AI-generated code, allowing security teams to identify, assess, and manage software risks proactively.

Zero Trust & Continuous Assurance Embedded in DevOps Pipelines

Scribe Security implements zero-trust security controls directly within DevOps workflows, ensuring that AI-generated code meets strict security criteria before reaching production environments:

- **Guardrails-as-Code**:
  Scribe automates security policy enforcement through guardrails. Any AI-generated code snippet containing critical vulnerabilities, or failing to meet any mandatory secure SDLC best practice, can provide immediate feedback to the developers and drive them to review again what the code they committed..

- **One Pane of Glass to Understand Risk**:
  Scribe integrates with the org's existing code scanning tools (SAST, DAST, etc.) to ensure the rapid detection and mitigation of vulnerabilities by all its AppSec tools as AI-generated code moves through the development lifecycle.

This continuous assurance ensures that AI-driven development remains fast-paced and innovative without sacrificing security. Developers gain the freedom to experiment and rapidly iterate using AI-generated code, confident that security guardrails automatically mitigate risk in real-time.

Automated Compliance at Scale

Finally, Scribe Security enables software producers to scale compliance by automatically validating the adherence of AI-generated code against cybersecurity frameworks, SSC regulatory mandates, and industry SDLC standards:

- **Regulatory Framework Compliance**
  Automated attestations and continuous policy enforcement enable compliance with cybersecurity regulations such as the EU Cyber Resilience Act (CRA), Executive Order 14028, 14144, FedRAMP, NIST SSDF, and many others.

- **Evidence-Based Auditability**
  Comprehensive, machine-readable reports and cryptographically signed attestations provide auditors and regulators with verifiable proof of compliance, removing friction from the audit process.

Scribe ensures that the rapid introduction of AI-generated code into software products does not come at the expense of regulatory compliance or security due diligence.

Conclusion: The Scribe Security Advantage for AI-Driven Development

As organizations increasingly turn to AI-driven code generation to accelerate innovation, they inevitably encounter challenges related to code quality, security risk, and compliance management. Traditional manual security reviews and periodic testing simply cannot keep pace with AI-accelerated development.

Scribe Security transforms this challenge into a strategic advantage. By embedding automated attestations, comprehensive visibility, zero-trust guardrails, and scalable compliance directly into development pipelines, Scribe empowers organizations to securely harness AI-generated code—unlocking unprecedented speed and innovation without compromising security.

In doing so, Scribe Security enables the full promise of AI-driven software development: rapid innovation paired seamlessly with continuous, automated cybersecurity assurance.

To learn more about how Scribe Security can empower your organization to safely accelerate AI-driven software development, please visit scribesecurity.com.