CODEHUNTER®
ZERO TRUST FOR CODE

Security Brief

Vibe Hacking

May 13, 2026

For the regulated-enterprise CISO

The Claim

AI agents are now capable of dynamically generating custom hacking tools and scripts during an active intrusion, allowing attackers to bypass traditional detection methods that rely on known signatures. Zero Trust for Code addresses this by enforcing what code and actions are permitted to execute, regardless of how they are generated.

The Threat

Threat actors are now using AI agents to generate custom hacking tools dynamically during intrusions, replacing reliance on prebuilt malware or known frameworks. This is known as “vibe hacking”.

These AI-generated scripts support reconnaissance, exploitation, and lateral movement, with each instance differing enough to evade signature-based detection.

This approach speeds up the attack process and enables continuous adaptation, allowing threat actors to iterate tooling and tactics in real time while remaining operationally stealthy.

The Problem

• Detection Collapse: Signature-based tools lose effectiveness when malware and scripts are dynamically generated each time.

• Operational Speed: AI drastically accelerates the time between reconnaissance, exploitation, and persistence making it hard for to defense to keep up.

• Tool Obsolescence: Traditional “known bad tool” detection is bypassed meaning tools no longer need to exist before they are used.

• Lower Barrier to Entry: Attackers can generate sophisticated attacks without deep technical expertise.

• Adaptive Users: AI enables attackers to modify tactics in real time, evading static defenses.

Zero Trust for Code lens: Authentication verifies identity and security tools detect known threats, but neither fully control what dynamically generated code is allowed to execute in real time.

The real breakdown is not that defenses fail, but rather that modern controls assume threats are pre-existing and identifiable. AI-driven attacks invalidate that assumption entirely.

The Impact

• Attack pace compresses to near real-time.

• Signature- and IOC-based detection loses relevance.

• SOCs face noise from highly variable artifacts, reducing confidence.

• Control effectiveness evidence weakens.

• Unknown, one-off tools execute outside policy validation, increasing operational exposure.

What to Watch For

• Rapidly changing scripts or binaries executing in environments.

• Legitimate sessions generating previously unseen commands or behaviors.

• High-frequency experimentation patterns (trial-and-error execution).

• Activity without known tool signatures but with clearly malicious outcomes.

• Indicators of on-the-fly tool or script generation.

A consistent pattern is the disconnect between what is executed and what is recognized. Legitimate sessions now produce behaviors that deviate from historical norms, often driven by dynamically generated scripts with minimal forensic consistency. Detection must evolve to assess not just access, but what that access enables systems to do. Without this visibility, anomalous activity blends with legitimate use, reducing detection efficacy and delaying response.

CODEHUNTER

Zero Trust for Code Value

Zero Trust for Code introduces runtime enforcement over all generated and executed artifacts, regardless of origin. It ensures that only actions within clearly defined behavioral boundaries are allowed to run, while dynamically generated scripts are evaluated before execution.

Any unauthorized or anomalous activity is blocked in real time, rather than just being detected after impact, shifting control to defense the moment it matters most.

This approach directly addresses the core gap exposed by AI-driven attacks: the inability to govern code that did not exist until execution.

By moving from reactive detection to pre-execution decisioning, organizations can regain control over unpredictable AI-generated attack methods, which are becoming more prevalent by the day.

Trust but verify.

CISO Action Brief

- Define behavioral execution policies for critical systems (what actions are allowed, not just who can act).
- Implement pre-execution control points to evaluate scripts and commands before they run.
- Augment detection with behavioral and intent-based analytics, not signatures.
- Prioritize controls that operate at machine speed, matching AI-driven attackers.

CISOs should shift from access control to action control by defining acceptable system behaviors and enforcing pre-execution validation. Detection must evolve toward behavioral analytics, reducing reliance on signatures. At the same time, organizations should invest in high-speed controls to keep pace with AI-driven threats and prevent anomalous activity before execution.

Methodology & Sources

Dark Reading (May 2026) reporting on AI-generated hacking tools, Trend Micro TrendAI research, and CodeHunter Labs analysis of AI-driven attack evolution

CODEHUNTER®