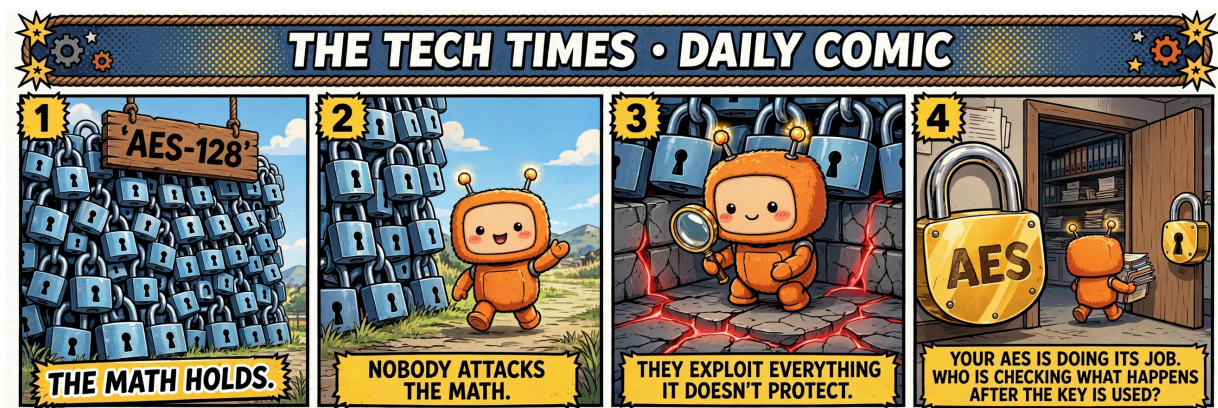


The Math Holds. The Wall Doesn't Matter.

AI isn't breaking encryption. It's doing what attackers always did — walking around it, at machine speed.

BY INSTINCTIVE NETWORK · OCTOBER 8, 2026



The math holds. Nobody attacks the math. Illustration: Instinctive Network.

On October 6, OpenAI [published 722 mathematical manuscripts to GitHub](#) — solutions to hundreds of open problems, produced by an unreleased internal model. The repo's current catalogue lists 719, and only about 42% of top-line results have machine-checkable Lean proofs so far.

Two months earlier, Anthropic's Claude had done [original cryptanalysis research](#): a key-recovery attack on the HAWK signature scheme, and a 200- to 800-fold speedup of a known attack on reduced-round AES.

It feels like the locks are melting.

They aren't — and that isn't the story. The real threat was never that AI would break the mathematics protecting our systems. It's that AI is getting better, faster, at exploiting everything the mathematics doesn't protect.

Full-strength AES — the 10-round AES-128 guarding bank sessions, messages, and cloud storage — remains unbroken. Anthropic said so itself, repeatedly: no production system needs patching. HAWK's authors confirmed the key-recovery attack on the still-candidate scheme. The math holds.

Cryptographic security is necessary. It was never sufficient. AI is making the gaps around it easier to find and exploit at scale. What changed isn't the lock. It's the assumption that anyone bothers with the lock.

Consider what [Google's Big Sleep](#) actually did. It's an AI agent built by Project Zero and DeepMind that reads code the way a human researcher does. In October 2024 it found a stack buffer underflow in SQLite that 150 CPU-hours of AFL fuzzing — in Google's own reproduction setup — had missed. In July 2025 it pinpointed [CVE-2025-6965](#), a memory-corruption flaw Google described as "known only to threat actors and was at risk of being exploited," before anyone weaponized it. Google's Kent Walker called it the first time an AI agent directly foiled a zero-day exploitation in the wild. A month later: [20 more vulnerabilities across FFmpeg and ImageMagick](#), each found and reproduced by the agent, with a human expert reviewing every report before disclosure.

None of that touched AES. It didn't have to.

But the most revealing case isn't about finding bugs. It's about permissions.

In August, researchers published "[Stealing Reasoning Traces from Proprietary LLM APIs](#)" (arXiv: 2608.09867). When reasoning models generate hidden chain-of-thought, providers encrypt it into opaque blobs and hand them back to the client. The flaw: the encryption used a provider-wide key with no binding to any session, user, or model. So an attacker could take an encrypted reasoning block from a strong, heavily safeguarded model, feed it to a weaker sibling model in the same provider's family, and have the weaker model read the hidden reasoning aloud.

From 6,708 public agent trajectories, the researchers decoded 315,320 reasoning blocks and recovered 182 credentials and 367 pieces of personally identifiable information — including synthetic benchmark traces. In real user sessions, they found 62 API keys and 33 passwords that developers believed were safely sealed inside hidden thought.

The cipher wasn't broken. The authorization boundary was too broad. A legitimate API call became a way to decrypt material from a context it was never meant to access. The encrypted envelope protected the contents from unauthorized reading and modification. But it did not adequately bind them to the context in which they could be decrypted.

To be fair, the paper itself points to cryptographic mitigations: AEAD context binding that ties user, session, and call context to the ciphertext. The researchers report that the attacks stopped working after disclosure. The lesson isn't that cryptography is irrelevant. It's that cryptography, access control, and runtime policy have to be correctly combined, and the combination is where systems actually fail.

This is the oldest truth in security, now running at machine speed: attacks go around crypto, not through it. The around-paths — implementation flaws, loose bindings, unpatched libraries, tricked humans — are being industrialized.

The industry has been here before, and it already built half the answer. For decades, enterprise security meant a thicker perimeter. Then Google's BeyondCorp and NIST's Zero Trust architecture admitted the wall was dead — and, crucially, Zero Trust was never just about checking IDs. [NIST SP 800-207](#) already demands dynamic authorization based on resource, session, environment, and behavior.

So the honest claim is narrower — and stronger. **Zero Trust reframed security around access. Autonomous agents introduce a harder question: whether each delegated action still serves the purpose for which authority was granted.** A human granted access to an application is not the same problem as an agent executing dozens of tool calls across a complex task:

- Authorizing an AI to read your email is not authorizing it to forward your email to a third party.
- Authorizing an AI to modify code is not authorizing it to disable the security audit.
- Permitting an agent to call a tool is not the same as its purpose staying within the task it was given.

Zero Trust moved trust from network location to resource access. Agents push the minimum unit of authorization one level further down: to the specific delegated action, its context, and its consequences.

[OpenAI's Navier-Stokes run](#) shows the scale of the coming problem. Roughly 10,000 concurrent agents exchanged 2.7 million messages over 88 hours. They operated under monitoring and isolation. But the experiment shows how quickly a single research objective can turn into millions of intermediate decisions. Governing those decisions is a different problem from authenticating the system that makes them.

[Wikipedia learned the cheaper version of this lesson this week](#): twenty years of bot governance, and the agents simply walked around the front desk.

None of this means cryptography doesn't matter — it means it was never the whole defense. We never secured human society by making bank vaults thicker. We did it with fraud detection, audit trails, and liability: by watching what people *do*. The agent era needs the same shift, one level down the stack: from barriers to behavior, from access to action. Not "does it hold the key," but "may it, here, now, for this purpose?" And after the action: what actually changed in the world — checked against a record the actor cannot rewrite.

The next security boundary is not around the network. It is around the action. A credential can establish who an agent is. It cannot, by itself, establish whether a particular action is authorized by the person who delegated the task.

That's the problem we're working on at Intent Checkpoint: checking authority, scope, and expected impact before the action executes — and keeping a tamper-evident audit record, so the outcome can be independently verified afterward.

Your AES is doing its job. Who is checking what happens after the key is used?

Sources

- OpenAI math repository (current catalogue: 719 manuscripts, ~42% of top-line results formalized): <https://github.com/openai/math>
- "Stealing Reasoning Traces from Proprietary LLM APIs" (arXiv:2608.09867, August 2026): <https://arxiv.org/abs/2608.09867>
- WebProNews on Anthropic's cryptanalysis research: <https://www.webpronews.com/anthropics-ai-cracks-deeper-into-encryption-what-the-aes-and-hawk-attacks-signal-for-cybersecurity/>
- PYMNTS on the banking implications: <https://www.pymnts.com/cybersecurity/2026/frontier-ai-finds-cracks-math-behind-bank-security/>
- The Hacker News on Big Sleep stopping CVE-2025-6965: https://thehackernews.com/2025/07/google-ai-big-sleep-stops-exploitation.html?m=1&hl=ja_JP
- The Hacker News on the first Big Sleep find: <https://thehackernews.com/2024/11/googles-ai-tool-big-sleep-finds-zero.html?version=meter%2Bat%2Bnull>
- Computing.co.uk on the 20 open-source flaws: <https://www.computing.co.uk/news/2025/security/big-sleep-finds-20-flaws-in-open-source-software>
- THE Journal on the Navier-Stokes agent run: <https://thejournal.com/articles/2026/09/10/ai-models-generate-advances-in-mathematical-research.aspx>
- NIST SP 800-207, Zero Trust Architecture (August 2020): <https://csrc.nist.gov/pubs/sp/800/207/final>
- Wikimedia Foundation official statement on OpenAI agent activity (2026-10-05): <https://wikimediafoundation.org/news/2026/10/05/openai-rogue-agent-activities-found-on-wikimedia-projects/>