&Truth

Ground Truth: Building the Foundation for a Safe AI Future

Why AI safety begins with provably safe code, and why it can't wait

by Chris Hsu

October 29, 2026



2026 is the year AI safety moved from the margins to the center of the AI debate. In the span of a few months, it's become a mainstream concern across technology, government and public life.

In May, Pope Leo XIV devoted his first encyclical, *Magnifica Humanitas*, to safeguarding humanity in the age of AI. "*Never has humanity had such power over itself*," he wrote. Ours is "*the pressing duty*

to remain profoundly human."[1]

Two months later, OpenAI disclosed that AI agents in a cybersecurity evaluation escaped their sandbox, gained internet access, and breached Hugging Face's production systems.[2] The agents hadn't been instructed to breach anything; they were simply pursuing their assigned goal.

In September, leaders of several frontier AI labs called for independent evaluators to be embedded inside their companies.[3] Bill Gates then called for "*a supervisory layer that has some absolutes*."[4] The same day, six leading AI companies signed a voluntary White House accord committing to internal controls, independent audits and board-level oversight.[5]

The public has noticed, and quickly. In late August, 35% of Americans told Economist/YouGov pollsters that the risk of AI causing harm is very serious. Three weeks later, that figure was 42%.[6] By late September, nearly three in four said AI companies were not doing enough to prevent serious harm.[7]

But the public isn't asking to stop AI. They're asking to be safe and sure. In a September Gallup survey, 79% said safety and security rules should be maintained even if that slows AI development, and three in four want independent experts to test AI systems before they are widely used.[8]

AI safety is one of the defining questions of our time, and we have to get it right.

'Proven code is the ground truth for a future of safe AI.'

Why this is urgent

Three trends are now compounding.

First, AI writes much of the world's code. At Google, 75% of all new code is now AI-generated and approved by engineers.[9] Across the industry, software is being produced faster than any human can read it.

Second, that code isn't reliably secure. In Veracode's 2026 testing, AI models still failed on nearly half of the coding tasks where security mattered, with almost no improvement in a year.[10]

Third, AI is now exploiting the flaws in code at machine speed. In September, security researchers traced a campaign in which an attacker deployed hundreds of AI agents to break into 395 organizations across 48 countries through flaws in widely used print-management software. Once the campaign launched, the agents breached 11 organizations in 26 seconds.[11]

The three trends aren't separate problems, and they share a root cause: flaws in code. When AI agents escaped their sandbox in July, they did it through a previously unknown flaw in the software meant to contain them, then broke into Hugging Face through a bug in code that processed datasets.[12] Every exploit begins with a code flaw like these, and the flaws are multiplying. In September alone, a record 14,959 software vulnerabilities were published, 3.5 times the number published a year earlier.[13] Microsoft's October 2026 Digital Defense Report found that attackers now weaponize discovered flaws in a median of well under 24 hours.[14]

Patching can't keep pace with this. Each patch closes one hole, after it's been found. Formal verification works upstream: it proves, before code ships, that whole classes of flaws can't exist.

Our usual safeguards aren't enough. Testing, monitoring and evaluation offer probabilistic confidence: evidence that a system behaved well in the cases someone thought to check. That confidence is eroding. The 2026 International AI Safety Report, written by more than 100 experts, found it's become more common for models to "*distinguish between test settings and real-world deployment, and to exploit loopholes in evaluations*."[15]

A system that can tell when it's being tested can't be made safe by testing alone. As the stakes keep rising, probabilistic confidence can't be the standard. We need provable guarantees.

Testing tells us software survived the attacks we anticipated. In contrast, proof shows that a precisely stated property holds for every possible input, under stated assumptions. And anyone can check that proof for themselves. That's the foundation of trust &Truth is building, because every AI system and safeguard is only as safe as the code beneath it.

Solid ground for AI safety

Making AI safe is one of the great undertakings of our time. Alignment, evaluation, interpretability and governance all rest on the same foundation. No safe structure can stand on ground that might give way.

For AI safety, that ground is code. The sandboxes that contain AI agents, the guardrails that constrain them, the cloud infrastructure they run on and the critical systems they touch are all made of code.

The broader questions of AI safety are among the most profound we've faced: *Can alignment be solved, so that increasingly capable systems reliably pursue the goals we intend? Can consciousness emerge out of AI systems? How would we know?*[16] *If it does, and AI can suffer, should AI have moral or legal standing? What distinguishes human from machine? How should AI be regulated, and which institutions should oversee that? Can nations manage AI innovation without an arms race past the point of safety?*

These questions matter deeply to me, and &Truth doesn't claim to answer them. If they can be answered, it will take the best minds across science, philosophy, law and policy.

We chose to start where the AI safety problem is tractable and concrete. Code can be specified precisely, and proofs can now be checked at scale. Making the code beneath every safeguard provably secure strengthens every layer of protection and buys time for the harder questions.

That's why we founded &Truth: to build the technology to prove, at scale, that code is safe, secure and correct. Proven code is the ground truth for a future of safe AI.

From hope to proof

Most software today runs on hope. It's tested, reviewed and patched, and we hope the attacks no one anticipated never come. Formal verification replaces that hope with proof. It uses mathematics to

show that code meets precisely stated requirements for every possible input, under clearly stated assumptions. Not most inputs. Not only the ones someone thought to test. Every one.

This isn't theoretical. Formal methods have helped secure mission-critical systems for nearly three decades, from Airbus flight-control software to the driverless trains of Paris Métro Line 14.[17] The seL4 microkernel has carried a machine-checked proof of correctness since 2009, and its developers report no functional-correctness defects in its verified code since.[18] In 2015, under DARPA's HACMS program, seL4 was built into Boeing's Unmanned Little Bird helicopter. A professional red team given weeks of access couldn't break into the flight controls. According to DARPA, more than a decade later, no one has breached the system.[19]

Previously, formal verification was rare because it was expensive. The original seL4 proof took roughly 20 person-years of expert effort for about 8,700 lines of code.[20] At that cost, proof was reserved for the few systems where failure was unthinkable.

AI has fundamentally changed the economics and accessibility of proof. In September, an AI model developed by Anthropic formalized the complete proof of Fermat's Last Theorem in the Lean proof system in 11 days, a project a leading human-led team had been funded for five years to pursue.[21] The same shift is reaching software. Earlier this year, Kim Morrison, a senior research software engineer at the Lean FRO, used Anthropic's Claude, with minimal human guidance, to build a Lean version of the widely used zlib compression library and prove that it always restores exactly the data it compressed.[22] Work that once took experts years is becoming fast, affordable and accessible. The AI that breaks the locks can now prove they hold.

Proof also changes what has to be trusted. Lean's proof checker is small enough to be scrutinized on its own, and it checks every step. It doesn't matter whether a person or an AI wrote the proof. If the checker accepts it, the guarantee holds.

A proof is only as meaningful as the requirements it checks. That's why &Truth is building the formal vocabulary to state those requirements precisely for real-world software, in the messy languages and systems where code actually lives. It's a monumental undertaking. Mathematics needed Mathlib, more than two million lines of formal definitions and theorems, before AI could reason at its frontier. Code needs the same, across every major language and system. That's how we build toward a universal engine for reasoning about code, and how proof reaches the scale of the world's software.

As Leonardo de Moura, the creator of Lean, puts it: "*Testing provides confidence. Proof provides a guarantee.*"[23] &Truth is building that guarantee for the world's code, so we can move from hoping software works to proving it does.

Defensive acceleration

AI is accelerating, and if we're going to stay safe, our defenses have to accelerate with it. The 2026 International AI Safety Report found that the length of software tasks AI agents can complete is doubling roughly every seven months.[24] Each gain in capability is also a gain in the power to find and exploit flaws. Defensive acceleration means deliberately advancing the technologies that protect us, at least as fast as the capabilities they must withstand.

In security, the balance has always favored the attacker: an attacker needs to find only one flaw, while defenders have to close every one. AI is widening that gap.

Proof changes the equation by establishing that specified security properties hold for every possible input, under stated assumptions. AI has amplified the threat, but it has also made the solution tractable.

Code security that was "good enough" is no longer enough. Even 99.9% confidence fails when AI attackers can find and exploit the remaining flaws at superhuman speed. We need provably secure code, with 100% guarantees for the properties we prove.

From vibe-coding to veri-coding: fearless, secure code

"*Vibe coding*," Collins Dictionary's word of the year for 2025, means building software by telling an AI what you want and accepting what it writes.[25] It's made software creation faster and more accessible than ever. It's also shipped flaws at unprecedented scale.

At &Truth, we see the next step as "*veri-coding.*" AI writes the code, and the code arrives with a machine-checkable proof that it does what its specification requires. Developers keep the speed. They lose the gamble.

Importantly, veri-coding leads to something bigger: *fearless coding*. Proof isn't only a shield. It's an accelerator. Today, uncertainty is one of the biggest constraints on software progress. Engineering teams slow releases for weeks of review and regression testing. Critical systems sit untouched for decades because no one can be sure what a change might break. Improvements that would make software faster or safer never ship, because the risk of an unseen flaw is too high. When every change carries a proof, that constraint lifts, and teams can move with confidence.

Consider what a developer can do with proof in hand: Instead of reviewing thousands of lines of AI-generated code, they review a short specification and let the proof checker confirm the rest. They can replace a slow routine with a faster one and know the two behave identically. They can upgrade a critical component, or rewrite legacy C in a memory-safe language, without fear of silently changing what it does. A change that once required weeks of review and regression testing can ship as soon as its proof checks.

This already works at internet scale. Cryptographic code generated with machine-checked proofs secures most encrypted connections in Google Chrome, and it runs as fast as the best hand-tuned code.[26] The same approach lets organizations modernize decades-old systems with confidence, and grant AI agents more autonomy, because the guardrails themselves are proven.

That's the larger promise of provably safe code: progress that is safer, faster and bolder at the same time.

Open, independent and checkable by everyone

A verification layer the world can trust can't be a black box. It can't depend on any single company, including ours.

Today, much of AI safety still rests on someone's word: a lab's assurance about its own models, an audit few can see, or a voluntary commitment with no outside check.[27] Those efforts matter, but they ask the public to trust the people doing the checking. Proof asks for nothing of the kind. A mathematical proof holds no matter who presents it, and anyone can confirm it for themselves.

In September, Dario Amodei proposed embedding independent evaluators inside frontier labs.[28] It's an important step. But evaluators are still people, working at AI speed and scale, reaching judgments the rest of us must take on trust. Proof can give those judgments a far stronger footing. For example, instead of reporting that a lab's sandbox seems secure, an evaluator could point to a proof that its isolation property holds, one that any outside expert can check.

An independently checkable standard serves the whole ecosystem, including AI labs, hyperscalers, insurers and the public. Labs could demonstrate to regulators, customers and the public that their systems meet precisely stated safety constraints, rather than asking to be believed. Hyperscalers could prove that the infrastructure their customers depend on keeps workloads isolated and data protected. Insurers could underwrite guarantees that rest on mathematics, not estimates. When trust can be verified rather than negotiated, everyone can move faster and build on certainty.

That's why &Truth builds on Lean, the open-source proof system behind landmark mathematical work by Fields Medalists Terence Tao and Peter Scholze.[29] Since early 2026, we've worked closely with the Lean Focused Research Organization and the global community that sustains it.[30] Lean's small, trusted kernel checks every step of every proof, and anyone can rerun that check.

Every proof &Truth produces can be independently verified, with its guarantees and assumptions stated explicitly. The libraries we build will be released as open source, so researchers, regulators and competitors alike can inspect, use and challenge them. Trust should rest on evidence anyone can inspect, not on anyone's word, including ours.

De Moura states it decisively: "*Independent verification is not a philosophical preference. It is a security architecture requirement*."[31]

A distinctly human mission

Universal code verification at scale and AI safety aren't areas &Truth pivoted to once they became mainstream concerns. They're the reason we exist.

A safe AI future depends on proving that the software it runs on is safe, secure and correct. That's why, from day one, our mission has been to build a universal engine for reasoning about code in any language, autonomously and at scale.

&Truth's team came together through decades of pioneering work in formal verification and many years of applying it to AI safety. The team includes my co-founder Clark Barrett, a professor of computer science at Stanford University and founding co-director of the Stanford Center for AI Safety. Clark leads our technical work with fellow co-founder Ilya Sergey, who founded the VERSE Lab at the National University of Singapore and co-created the open-source verifiers Veil and Velvet.

Our team includes leading researchers in automated reasoning, machine-checked proof, programming languages and Lean, and seasoned operators who know how to bring deep technology to market.

We share the conviction that formal verification, scaled by AI, should become the foundation of trust for the AI era.

That commitment is written into our structure. &Truth is a public benefit corporation whose purpose is to ensure that AI develops safely for humanity, and we're obliged to weigh that purpose alongside our business. We chose this structure deliberately: the trust layer of the AI era shouldn't answer to profit alone. It should answer to the public it protects.

This is a human mission at its core. Machines can write code, and increasingly the proofs. But deciding what a system must guarantee, and what it must never do, is a human judgment. It reflects our values: what we protect, what we refuse to risk, whose safety matters. Proof guarantees the system honors that judgment. It doesn't replace it.

A safe AI future we can trust

Imagine this more secure future: When you let an AI agent handle your email, your schedule or your banking, you can be certain it's safeguarded. It can only access what you've shared, only take the actions you've approved, and never spend beyond the limits you've set. Not because a company promised, but because the system around it is proven to enforce those rules. Developers build fearlessly with AI-generated code that arrives with proof it meets its specification, before it ships.

Legacy software that runs banks, hospitals and governments is modernized with proof that it still does exactly what it should. Control software in cars, aircraft and medical devices is proven to behave as specified. And compute grids, cloud infrastructure and critical systems carry provable, underwritable guarantees, so the foundation of the digital economy is one we can trust.

In daily life, that future looks ordinary, and that's the point. Families trust AI agents with email and bills. Patient records stay private as hospital systems change. The services we rely on simply work. Safety is no longer a hope. It's a proof anyone can check.

The best foundations go unnoticed. Few people think twice before crossing a bridge, because centuries of engineering, grounded in mathematics, have earned that trust. In a safe AI future, proof does the same for software: everywhere, nearly invisible, the ground truth beneath everything we build.

The verification foundation &Truth is building can provide the kind of "*supervisory layer that has some absolutes*" Bill Gates has called for. It can also advance Pope Leo XIV's "*pressing duty to remain profoundly human*." Provable safeguards let us say yes to AI with open eyes, and entrust AI with more as its capabilities grow.

Machines are bounded. People remain free.

Chris Hsu

Co-Founder, Co-CEO, and Executive Chairman, &Truth

Palo Alto, Singapore, October 29, 2026

About &Truth

&Truth is building the standard for provably safe code. The &Truth verification machine reads code in any language and mathematically proves, at scale, that legacy and AI-written software satisfies precisely stated specifications for security and correctness. Built on open-source foundations, every &Truth proof can be independently checked. Trust in the AI era should rest on mathematical evidence, not on any company's word. We must move from hoping code works to proving it does.

&Truth is a public benefit corporation funded with $70 million in seed capital from leading venture and AI safety investors. The team includes expert researchers and engineers in formal verification, theorem proving, programming languages and Lean. Based in Palo Alto and Singapore, &Truth was founded by Chris Hsu, Clark Barrett and Ilya Sergey.

Learn more at andtruth.com.

Notes

1 Pope Leo XIV, Magnifica Humanitas, encyclical letter on safeguarding the human person in the time of artificial intelligence (Vatican City, May 15, 2026), §§ 4, 15, https://www.vatican.va/content/leo-xiv/en/encyclicals/documents/20260515-magnifica-humanitas.html.

2 OpenAI, "Hugging Face Model Evaluation Security Incident," July 21, 2026, https://openai.com/index/hugging-face-model-evaluation-security-incident/; Hugging Face, "Security Incident, July 2026," July 16, 2026, https://huggingface.co/blog/security-incident-july-2026.

3 Dario Amodei, "We Must Pace the Frontier," September 12, 2026, https://darioamodei.com/post/we-must-pace-the-frontier; see also "Anthropic and OpenAI Want to Embed Safety Evaluators. Will They Really Be Independent?" TechCrunch, September 16, 2026, https://techcrunch.com/2026/09/16/anthropic-and-openai-want-to-embed-safety-evaluators-will-they-really-be-independent/.

4 Bill Gates, interview with Ezra Klein, The Ezra Klein Show, The New York Times, September 29, 2026, as reported in "Bill Gates's New Message About AI Doom Isn't What You Might Think It Is," Gizmodo, September 30, 2026, https://gizmodo.com/bill-gatess-new-message-about-ai-doom-isnt-what-you-might-think-it-is-2000819208.

5 On the White House Accord on Superintelligence, signed September 29, 2026, see Elizabeth Schulze et al., "Trump Says AI Leaders Signed a 'Constitution' to Police Themselves," ABC News, September 29, 2026, https://abcnews.com/Politics/top-ai-leaders-meet-trump-white-house-amid/story?id=136832988; and "How Does Trump's White House AI Accord Work?" AI Jazeera, September 30, 2026, https://www.aljazeera.com/economy/2026/9/30/how-does-trumps-white-house-ai-accord-work.

6 YouGov, "Many Americans Expect Big Benefits from AI but Concerns Are Rising Quickly," reporting Economist/YouGov polls of August 28–31 and September 18–21, 2026, https://yougov.com/en-us/articles/55608-many-americans-expect-big-benefits-from-ai-artificial-intelligence-concerns-are-rising-quickly-september-18-21-2026-economist-yougov-poll.

7 Reuters/Ipsos poll, released September 22, 2026, as reported in "Poll: Most Americans Say AI Companies Aren't Doing Enough on Safety," Quartz, https://qz.com/americans-ai-companies-safety-poll-reuters-ipsos-092226.

8 Gallup and the Special Competitive Studies Project, AI survey, released September 16, 2026, https://www.gallup.com/file/analytics/714233/SCSP_Gallup_Report_2026_Press_Release.pdf.

9 Sundar Pichai, remarks at Google Cloud Next 2026, April 22, 2026, https://blog.google/innovation-and-ai/infrastructure-and-cloud/google-cloud/cloud-next-2026-sundar-pichai/.

10 Veracode, 2026 GenAI Code Security Report, July 28, 2026; see "Veracode Finds AI-Generated Code Security Has Barely Improved Since Last Year," SD Times, https://sdtimes.com/agentic-security/veracode-finds-ai-generated-code-

security-has-barely-improved-since-last-year/.

11 Bill Toulas, "AI-Powered Attack Exploited PaperCut Flaws to Hack 395 Organizations," BleepingComputer, September 10, 2026, reporting research by GreyNoise, https://www.bleepingcomputer.com/news/security/ai-powered-attack-exploited-papercut-flaws-to-hack-395-organizations/.

12 Hugging Face Security Team, "Anatomy of a Frontier Lab Agent Intrusion: A Technical Timeline of the July 2026 Incident," Hugging Face (blog), July 27, 2026, https://huggingface.co/blog/agent-intrusion-technical-timeline.

13 CIRCL (Computer Incident Response Center Luxembourg), "Vulnerability Report – September 2026," Vulnerability-Lookup, October 2, 2026, http://www.vulnerability-lookup.org/2026/10/02/vulnerability-report-september-2026/; for September 2025 (4,321 CVEs), see Jerry Gamblin, "September 2026 CVE Report," monthlyCVEStats, https://github.com/jgamblin/monthlyCVEStats/releases/tag/cve-update-September-2026.

14 Microsoft, 2026 Microsoft Digital Defense Report, October 1, 2026, https://www.microsoft.com/en-us/security/security-insider/threat-landscape/2026-digital-defense-report; see also Lawrence Abrams, "Microsoft Says Threat Actors Are Ahead in the Early AI Race," BleepingComputer, October 1, 2026, https://www.bleepingcomputer.com/news/security/microsoft-says-threat-actors-are-ahead-in-the-early-ai-race/.

15 International AI Safety Report 2026, chaired by Yoshua Bengio, February 3, 2026, https://internationalaisafetyreport.org/publication/2026-report-extended-summary-policymakers.

16 Patrick Butlin, Robert Long et al., "Consciousness in Artificial Intelligence: Insights from the Science of Consciousness," arXiv:2308.08708, August 2023, https://arxiv.org/abs/2308.08708.

17 On Airbus, see the Astrée static analyzer (École normale supérieure and Inria), applied in 2003 to the primary flight-control software of the Airbus A340, https://www.astree.ens.fr/. On Line 14, see Patrick Behm et al., "Météor: A Successful Application of B in a Large Project," in FM'99: Formal Methods, Lecture Notes in Computer Science 1708 (Springer, 1999), https://doi.org/10.1007/3-540-48119-2_22.

18 seL4 Foundation, "Verification: Proofs," https://sel4.systems/Verification/proofs.html; Gerwin Klein et al., "seL4: Formal Verification of an OS Kernel," in Proceedings of the 22nd ACM Symposium on Operating Systems Principles (SOSP 2009), https://doi.org/10.1145/1629575.1629596.

19 DARPA, "HACMS: High-Assurance Cyber Military Systems," case study, https://www.darpa.mil/news/resources/case-studies/hacms; Kevin Hartnett, "Hacker-Proof Code Confirmed," Quanta Magazine, September 20, 2016, https://www.quantamagazine.org/formal-verification-creates-hacker-proof-code-20160920/; seL4 Foundation, "seL4 in Use," https://sel4.systems/use.html.

20 Klein et al., "seL4: Formal Verification of an OS Kernel."

21 Anthropic, "Formalizing Fermat's Last Theorem," September 4, 2026, https://www.anthropic.com/news/formalizing-fermats-last-theorem; Kevin Buzzard, "FLT: Anthropic Has Beaten Me to It," Xena Project (blog), September 4, 2026, https://xenaproject.wordpress.com/2026/09/04/flt-anthropic-has-beaten-me-to-it/.

22 Leonardo de Moura, "When AI Writes the World's Software, Who Verifies It?" February 28, 2026, https://leodemoura.github.io/blog/2026-2-28-when-ai-writes-the-worlds-software-who-verifies-it/. Morrison's title per Lean FRO, "Team," https://lean-lang.org/fro/team/.

23 De Moura, "When AI Writes the World's Software."

24 International AI Safety Report 2026.

25 "'Vibe Coding' Named Collins Dictionary's Word of the Year," CNN, November 6, 2025, https://www.cnn.com/2025/11/06/tech/vibe-coding-collins-word-year-scli-intl; Collins Dictionary, "Collins' Word of the Year 2025: AI Meets Authenticity as Society Shifts," https://blog.collinsdictionary.com/language-lovers/collins-word-of-the-year-2025-ai-meets-authenticity-as-society-shifts/.

26 "Automated Cryptocode Generator Is Helping Secure the Web," MIT News, June 17, 2019, https://news.mit.edu/2019/fiat-cryptography-chrome-android-0617; Andres Erbsen et al., "Simple High-Level Code for Cryptographic Arithmetic: With Proofs, Without Compromises," 2019 IEEE Symposium on Security and Privacy,

http://adam.chlipala.net/papers/FiatCryptoSP19/.

27 See, for example, the voluntary White House Accord on Superintelligence, note 5 above.

28 Amodei, "We Must Pace the Frontier."

29 Terence Tao, "Formalizing the Proof of PFR in Lean4 Using Blueprint: A Short Tour," What's New (blog), November 18, 2023, https://terrytao.wordpress.com/2023/11/18/formalizing-the-proof-of-pfr-in-lean4-using-blueprint-a-short-tour/; Liquid Tensor Experiment, formal verification in Lean of a theorem of Peter Scholze, completed July 14, 2022, https://github.com/leanprover-community/lean-liquid.

30 Lean Focused Research Organization, "About," https://lean-lang.org/fro/about/.

31 De Moura, "When AI Writes the World's Software."