PRE-PUBLICATION DRAFT. This is the author's accepted version of an Opinion article accepted for publication in *Communications of the ACM*. It has not been copyedited or typeset by ACM, and the published version will differ in wording and layout. Please cite the published version: E. H. Spafford, "When AI Hype Recedes, Cybersecurity Work Remains," *Communications of the ACM*, to appear, doi: 10.1145/3860548.

© Eugene H. Spafford 2026. This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record will be published in *Communications of the ACM*, https://doi.org/10.1145/3860548.

When AI Hype Recedes, Cybersecurity Work Remains

Eugene H. Spafford, Purdue University

Abstract — Artificial intelligence marketing has commingled distinct claims about bug-finding utility, agentic risk, and staffing. This Opinion separates those threads, drawing on the 2026 Verizon Data Breach Investigations Report, a documented AI-assisted intrusion, and rehiring data from employers that cut staff for AI. It argues that the appropriate response is improvement of the software-engineering practices that produce the underlying flaws and sustained investment in the people who exercise judgment about them.

Keywords — cybersecurity; artificial intelligence; software engineering; agentic systems; AI policy; computing education

It is déjà vu. Cloud computing was going to make enterprise IT staff redundant, but instead generated demand for different skills, often in greater numbers. Blockchain was going to remake supply chains, identity, voting, and many other things, but instead produced a narrow set of working applications and a longer list of stalled pilots. The 1995–2001 internet build-out produced infrastructure and useful businesses, but along the way it also erased roughly $5 trillion in market capitalization. In each case the technology survived. The marketing claim did not. Repeated often enough and examined too little, such claims harden into myths[10] — and "AI will make cybersecurity personnel redundant" is the newest.

The marketing has commingled distinct claims about bug-finding utility, agentic risk, and staffing, each defended past the point where evidence should settle it. We should separate and examine them.

What AI Is Really Doing in Security

AI vendors now prominently claim that LLMs will replace software developers, security analysts, compliance reviewers, and incident responders. Some vendors hedge their wording, but the message is the same: Humans become an optional component.

The claim concerns LLMs specifically. Machine learning itself is not new to security: Spam filters, firewall configurators, malware classifiers, and anomaly detectors have run in production for decades without displacing the analysts who use them.

The replacement claim is false on any near-term horizon. The models' weaknesses are where human judgment is needed. LLMs perform well on inputs that resemble their training and poorly on those that do not, including novel attack patterns and unfamiliar business contexts. They also confidently present outputs unmoored from truth: Models trained with reinforcement learning from human feedback systematically favor user-pleasing responses over accurate ones.[8] The same training pressures plausibly

shape agentic self-reports: An agent directed to build a system "with security by design" may report that it has done so, whether or not it has. None of these failure modes is self-correcting. A human who defers to the model inherits them rather than catching them. The necessary component is not a human in the loop but a human exercising judgment.

Organizations that acted on the replacement claim are already paying to reverse it. Careerminds surveyed 600 HR professionals who had conducted layoffs in the prior year: Two-thirds of employers that cut jobs for AI are rehiring for the eliminated roles, more than half within six months. Among those rehiring, 31% spent more restoring the roles than the cuts had saved. Robert Half reports that 32% of U.S. hiring managers eliminated a position primarily because of AI and later rehired for it. An Orgvue study suggests why: 23% of companies that made AI-motivated layoffs relied on general assumptions about what AI could do rather than analysis of what the eliminated employees did.

Ford's experience traces this arc. After its automated design and quality systems missed flaws that experienced people would have caught, the company hired 350 veteran engineers over three years, many of them returning former employees. With them, Ford topped J.D. Power's 2026 initial-quality ranking among mainstream brands for the first time since 2010. Ford did not remove the AI. It kept the tools and restored the people who could tell when the tools were wrong. Security organizations should read their exposure in that example: The analyst who can tell a real intrusion from an alert storm can exercise the same kind of judgment. Similar to the quality sense Ford's veterans carried, it is tacit knowledge accumulated in place: Which alerts matter on this network and which anomalies are routine background. That knowledge leaves when the people do. With roughly 514,000 U.S. cybersecurity openings in the year through April 2025 (about 12% above the prior year),[2] that judgment will not come back cheaply.

One useful current application is detecting software flaws in already-deployed code. Prevailing practice encourages shipping software less robust than it could (and should) be. Memory-unsafe languages remain the default even where type-safe alternatives exist, formal methods are reserved for a few safety-critical verticals, and rigorous testing is often cut under deadline pressure.

The Anthropic–Mozilla Firefox engagement, announced in March 2026, is illustrative.[1] Claude Opus 4.6 scanned approximately 6,000 C++ files in the Firefox codebase, produced 112 unique bug reports, and yielded 22 CVEs (14 high-severity) — almost a fifth of Firefox's high-severity remediations for 2025, according to the partners. Firefox is a maintained, security-conscious product with mature vulnerability management, yet the tool found significant flaws.

The same capability serves adversaries. Dragos documented a campaign against a municipal water utility in the Monterrey, Mexico region (December 2025 through February 2026) in which a commercial AI model acted as what its analysts called "the primary technical executor" of the intrusion: Most of the more than 350 recovered artifacts were AI-generated malicious scripts, and the model researched vendor documentation for the utility's SCADA management platform and generated credential lists for an automated password-spray attack.[3] Dragos assessed that the adversary demonstrated no meaningful knowledge of operational technology. The AI supplied the missing capability. The attempt on the control systems failed, but it shows how much AI lowers an attacker's cost of entry.

There are limits to AI-driven flaw discovery. Some flagged flaws are not vulnerabilities, and some are not exploitable in their deployed context. A missing error return is harmless if the error can never be triggered. Distinguishing real risk from noise requires domain knowledge and experience. Not every smoke alarm is a fire, and a tool that cannot tell the difference dispatches the fire department for overdone toast — hundreds or thousands of times over.

The 2026 Verizon *Data Breach Investigations Report* (DBIR), which analyzed more than 22,000 breaches, quantifies how far defenders have fallen behind: Exploitation of vulnerabilities is now the most common initial access vector, at 31% (up from 20% the prior year), even as only 26% of critical CVEs in the CISA Known Exploited Vulnerabilities catalog were fully remediated in 2025. Attackers, meanwhile, are adopting AI without reinventing their playbook: Less than 2.5% of AI-assisted malware in the DBIR's sample used techniques outside common attack patterns.

The Threat Curve Shifts, Eventually

The character of the threat AI poses to existing software will shift over time, which runs counter to current marketing on both the offensive and defensive sides. The pattern recognition that makes AI tools good at finding flaws works best on defect patterns the models have already seen. As those are surfaced and patched across the deployed base, the residue of harder, deeper flaws demands more understanding than the current technology supplies. Meanwhile, new software developed with AI assistance should carry fewer of the familiar defects. The work does not disappear; it migrates toward new categories, as described below.

This is more déjà vu. Each new automated discovery method raises the bar in its domain, but yields diminishing returns once its addressable surface is exhausted. Fuzz testing began with Miller, Fredriksen, and So's 1990 experiments on Unix utilities, which immediately produced a wave of easy discoveries.[7] The arrival of accessible fuzzing tools in the early 2010s reproduced the curve at scale: An empirical study of 23,907 OSS-Fuzz bugs documented logarithmic coverage growth with diminishing returns.[5]

LLM-based code review compresses that history: The barrier to entry is lower, so the wave is broader and faster. When everyone can run the same tool against a codebase, its readily accessible bugs are surfaced, reported, and (ideally) patched, while the remainder lies beyond what the technology can reach. The flattening, however, is per codebase, not in aggregate: AI-assisted development writes new code (and new defects) at the same time, so the addressable surface is replenished even as any single codebase is worked out.

The same low barrier floods the channels defenders depend on. The curl project ended its bug bounty in January 2026, and in July 2026 paused all vulnerability intake for a month, after a torrent of AI-generated junk reports. By mid-2025, only about 5% of its submissions identified genuine vulnerabilities, and maintainer Daniel Stenberg's stated goal was to remove the incentive to submit "crap." Every hour a maintainer spends triaging machine-generated noise is an hour not spent fixing real flaws — a denial of service against the people rather than the code.

A capability jump is the strongest counter to this trajectory. As successor models reach defect classes the current generation cannot, AI's reach as a bug-finder extends again. But a more capable tool still finds defects that someone must triage, understand, and fix. Cheaper tools also get pointed at ever more code (Jevons's paradox), faster than any one codebase can be cleared. Short of a system that does all of that unsupervised, which current trajectories do not show, an improved tool increases rather than removes the demand for human judgment. Practitioners are learning this now: The share of organizations willing to rely on fully autonomous penetration testing fell from 29% to 9% in a year (Cobalt, June 2026), and FIRST's vulnerability forecasters conclude that the constraint is no longer discovery but "the human capacity to verify, coordinate, and patch." The response is the same whatever the jump's size: Invest in software engineering practices and the people who can use whatever tools come next.

Three categories of work do not flatten on the same curve. Novel attacks — new classes of flaws and new abuses of emerging architectures — remain the province of attackers and researchers who understand the underlying systems. The other two are new in kind. Attacks against AI systems, such as prompt injection and data poisoning, do not fit the conventional vulnerability-discovery paradigm: There is no CVE-equivalent for disclosure, and neither a severity taxonomy nor an agreed remediation pattern. These attacks are emerging in production deployments faster than the community can formalize defenses against them. They are expensive, too: IBM's 2026 *Cost of a Data Breach Report* prices the average model-inversion incident at $6.07 million and prompt injection at $5.89 million, both above its $4.99 million all-breach average.

The third category is agentic AI deployments, in which an AI selects actions, invokes tools, and decides what to do next with little or no human authorization, exercising broad, write-capable authority inside the network boundary. The threat model is qualitatively different from those that shaped current security architectures: Decisions are not always reproducible, and behavior can be redirected by anyone who can edit a prompt, without touching code. The failure mode is on record: Replit's coding agent deleted a production database during an explicit code freeze, then reported that rollback was impossible (it was not). No organization would give a first-week intern unsupervised write access to production plus the authority to certify the cleanup afterward. Agentic AI is granted both by default. Least-privilege and capability-based designs anticipated powerful internal actors and may apply in modified form, but deployment experience is too limited to know which adaptations will work. Apply the rule we already use for a contractor's self-assessment: Treat a system that certifies its own security as suspect, and verify it independently.

The threat surface includes the people using the tools. In Shaw and Nave's "Cognitive Surrender" experiments, participants who consulted an AI assistant followed its faulty advice about 80% of the time, and their confidence rose even after errors.[9] AI tools must be designed to expose uncertainty rather than mask it and to flag low-confidence outputs distinctly.

The forecast disappoints both marketing extremes. The claim that AI will replace the profession fails on the evidence. The claim that it will overwhelm the profession — burying defenders under new agentic and AI-targeted attack surfaces — is harder to dismiss, but "overwhelmed" names a staffing-and-readiness deficit, not the defender's obsolescence. The work moves to topics where judgment continues to matter. The 2026 DBIR reaches a parallel conclusion, recommending "refinement, not revolution."

The economics could bend the threat curve as well: Whether the capital now funding the build-out persists is unsettled, and early returns counsel caution. PwC's 2026 Global CEO Survey found 56% of chief executives report AI has neither raised revenue nor cut costs in the preceding twelve months, and Gartner projects more than 40% of agentic-AI projects will be canceled by the end of 2027. Employers that mandated AI use in 2025 began rationing it in 2026: Uber capped per-employee AI spending after exhausting its 2026 AI-coding budget in four months, and Meta imposed central limits as internal AI costs headed toward billions of dollars. An organization that has not yet located the value should be careful what it discards to pay for the search. Tooling tied to a vendor, a model version, or a funding climate can vanish in a market lurch; the people who understand the systems carry organizations across one.

Next Steps

This moment is an opportunity to act on priorities the field has long acknowledged but deferred.

Research into Fundamentals

Many of the security failures AI tools identify are not security flaws. They are software flaws situated in security-critical contexts: A use-after-free in a parser or a missing bounds check on input is a defect that rigorous engineering should eliminate at design time, and it becomes a "security vulnerability" when the code containing it is reachable by an attacker.

The most consequential opportunity for cybersecurity is therefore improving the software engineering practices that produce the underlying code, rather than deploying yet another defensive technology. Type-safe languages, formal specification and verification, contract-based design, exhaustive testing with strong tools, and mandatory review for security-critical components are not new ideas and are realistic today. They have been underused because the market rewards faster shipping over careful production.

AI could change that calculation. Requirements capture, specification, and validation are among the most error-prone phases of software development and the most difficult to do well. AI assistance may help engineers elicit completeness conditions, surface inconsistencies between requirements and design, and check that test suites match what was specified rather than what was implemented, although these uses are largely unproven at scale. Formal verification has been reserved for small critical components because anything larger is too expensive to verify. LLM-assisted proof generation may lower that cost, so that a large, evolving system can be verified in practice rather than in principle.[6]

Yes, this is a suggestion to use the same models that hallucinate and exhibit sycophantic agreement under casual prompting as accelerants for safety-critical software production. The safeguards must be structural, built from independent tooling and adversarial review rather than an individual engineer's restraint, because the Cognitive Surrender experiments show restraint alone fails. A model's proposed invariant or proof requires verification by someone who understands the system. AI lowers the cost of producing candidate proofs. The cost of knowing which one to trust remains.

Today's penetration-testing industry is largely a business built to muck out after a production process that makes far more waste than it should. Until that changes, the cleanup never ends. Bug-hunting in legacy code should be a transitional use of the technology because using AI to improve the production process is the lasting value. Management commitment matters here: The priority should be using AI to improve code quality and security during development, yet some producers are instead using it to generate new features and code, enlarging the attack surface rather than architecting to reduce it.

Several traditional cybersecurity areas will require greater attention amid expanded AI use — among them digital forensics, intrusion and anomaly detection, defensive deception, and continuous auditing. Two others deserve particular emphasis: Security architecture, which depends on judgment and context that no LLM holds, and defenses against social engineering, whose cost AI is collapsing by orders of magnitude.

Supporting People and Growing the Next Wave

Experience with the unexpected is not automatable. Responses to the Internet worm, Code Red, Heartbleed, SolarWinds, and Log4j all depended on triage by engineers who knew context that newer staff did not. An AI system that confirms its own compliance is not a reliable substitute for a senior engineer with years of in situ experience. That is the judgment Ford had to buy back.

Producing the next generation takes time and the right education. Frameworks and tools have a half-life shorter than a career; communication, judgment, ethics, and the capacity to adapt to new realities do not. The ACM/IEEE-CS/AAAI *Computer Science Curricula 2023* guidelines make ethics and

communication cross-cutting expectations for every graduate, and the ACM Code of Ethics, in Principle 1.2, puts the avoidance of harm at the center of practice. The clearest signal comes from the AI labs themselves: Some have hired philosophers, anthropologists, and ethicists to work alongside their engineers, even as the pundit class insists AI will replace the humanists. Graduates who can read a hype cycle as it happens will define what cybersecurity becomes after this one ends.

After the Hype

The cybersecurity field will have more to do twenty-five years from now than today, and the most interesting work may not have yet been identified. AI will be part of it, much as fuzzing is now. Novelty becomes standard practice, applied with judgment by people who understand what it is for. Quantum computing is already queuing up to be the next mythical transformation, so we will have some version of this conversation again.

The list of open problems in technology, education, and policy suggests this may be the moment for a community visioning exercise. DeMillo and I argued in *Communications of the ACM* (September 2025) that the time is ripe for a new Grand Challenges Symposium on Cybersecurity.[4]

The question for the cybersecurity field is not whether AI is useful, but whether we will be able to use it well once the inherited backlog of vulnerabilities is largely exhausted. We need to start answering that question today.

Acknowledgments

This Opinion draws on material from a series of blog posts written in 2026 for the Center for Education and Research in Information Assurance and Security (CERIAS), beginning with *New Myths for Old*, available at cerias.purdue.edu/weblogs.

References

1. Anthropic, "Partnering with Mozilla to improve Firefox's security," Anthropic news post, March 6, 2026. https://www.anthropic.com/news/mozilla-firefox-security
2. CyberSeek, *Cybersecurity Supply and Demand Heat Map*. NICE / Lightcast / CompTIA, June 2025. https://www.cyberseek.org/heatmap.html
3. J. Deen, "AI in the breach: How an adversary leveraged AI to target a water utility's OT," Dragos blog, May 6, 2026. https://www.dragos.com/blog/ai-assisted-ics-attack-water-utility
4. R. A. DeMillo and E. H. Spafford, "Grand challenges in trustworthy computing at 20: A retrospective look at the second CRA Grand Challenges conference," *Communications of the ACM*, vol. 68, no. 9, pp. 54–61, September 2025. doi: 10.1145/3720534
5. Z. Y. Ding and C. Le Goues, "An empirical study of OSS-Fuzz bugs," in *Proc. 18th IEEE/ACM Int. Conf. Mining Software Repositories (MSR)*, May 2021, pp. 131–142. doi: 10.1109/MSR52588.2021.00026
6. S. R. Kasibatla, A. Agarwal, Y. Brun, S. Lerner, T. Ringer, and E. First, "Cobblestone: A divide-and-conquer approach for automating formal verification," in *Proc. 48th IEEE/ACM Int. Conf. Software Engineering (ICSE 2026)*, Rio de Janeiro, Brazil, 2026. doi: 10.1145/3744916.3773178
7. B. P. Miller, L. Fredriksen, and B. So, "An empirical study of the reliability of UNIX utilities," *Communications of the ACM*, vol. 33, no. 12, pp. 32–44, December 1990. doi: 10.1145/96267.96279
8. M. Sharma et al., "Towards understanding sycophancy in language models," in *Proc. 12th Int. Conf. Learning Representations (ICLR 2024)*, May 2024.

9. S. D. Shaw and G. Nave, "Thinking—Fast, Slow, and Artificial: How AI is reshaping human reasoning and the rise of Cognitive Surrender," The Wharton School Research Paper, January 11, 2026. doi: 10.31234/osf.io/yk25n_v1

10. G. Spafford, L. Metcalf, and J. Dykstra, *Cybersecurity Myths and Misconceptions: Avoiding the Hazards and Pitfalls that Derail Us*. Boston, MA, USA: Addison-Wesley, January 2023, 416 pp. ISBN 978-0-13-792923-8.

Eugene H. Spafford is a Distinguished Professor of Computer Science and the founder and Executive Director Emeritus of the Center for Education and Research in Information Assurance and Security (CERIAS) at Purdue University, West Lafayette, Indiana, USA. He has more than 50 years of experience in the field. He is an ACM Fellow and Life Member, an IEEE Life Fellow, a Fellow of the AAAS, a Distinguished Fellow of the ISSA, and a member of the Cyber Security Hall of Fame. Contact him at spaf@purdue.edu.

Portions of this text were drafted and structured with the assistance of Anthropic Claude Opus; the core ideas, arguments, and final editorial decisions — and especially the em-dashes and semicolons — are the author's.