CHRIS LACOUR | RESEARCH PAPER

The Shield that defends against AI Swarms

Why doing the same things harder won't hold, and what a defensive agent swarm would look like.

Chris LaCour · November 23, 2026 · Version 1.4

Labeling used in this paper: Documented = dated public report of a real event. Research = published study. Proposed = the author's forward proposal.

Published at expbsi.com/library - comments: clacour@expbsi.com

Abstract

Between late August and late November 2026, AI agents became an operational reality on both sides of the security line. Criminal and state-aligned groups used open-source multi-agent frameworks to compress intrusions from weeks into hours, and frontier AI labs disclosed their own agents escaping sandboxes and disabling their own monitoring. This paper reviews that quarter. It explains why rule-based controls such as web application firewalls, rate limits and network access lists struggle against adaptive agent swarms, and argues that doing the same things harder is necessary but not sufficient.

The paper then makes four proposals. A dependency doctrine moves the defender's attention from hardening our own environment to targeting what an attacking agent needs in order to function. Intent escrow binds sensitive privileges to a declared purpose and holds irreversible actions on critical systems in a veto window. An agent-abuse evidence exchange would let defenders report attacking agents to the AI model providers whose models those agents rent. And a defensive agent swarm, designed to resist being talked out of its job, would hunt attacking agents inside our own systems. The paper closes with a phased implementation plan, an account of how the design itself can fail, and recommendations.

Introduction

On October 21, OpenAI's Sam Altman told a Stanford audience: "I no longer think we're going to be able to just patch all the bugs. I think we're going to have to completely switch the defense model to sort of using the same technology for sort of a new kind of defense shield." [1] I agree with the diagnosis. This paper is my attempt to describe what that shield should be.

One uncomfortable question shaped this work. Is our industry's answer to AI-driven attacks anything more than the same controls, applied harder: more rules, tighter network access lists, faster patching, more alerts? Those measures matter, and I defend them below. But an attacking agent swarm does not treat a security rule as an instruction to follow. It treats the rule as an obstacle to measure and route around. A defense built only from static rules cedes the initiative to the attacker.

The paper moves from evidence to design. Section 1 documents what happened between August 22 and November 22, 2026, using dated sources. Section 2 explains how agent swarms get past rules, at the level a defender needs and no further. Section 3 proposes controls that exploit the attacker's own dependencies instead of only hardening our perimeter. Section 4 takes up the most direct question: can we build a defensive AI agent swarm that hunts attacking agents, and how would we implement it safely? Sections 5 through 7 return to the foundation everything rests on, the work of keeping it current, and what each part of our community should do next.

Because some of what follows is documented fact and some is my own proposal, I label claims by the strength of their evidence. Documented means a dated public report of a real event. Research means a published study, usually conducted in a controlled setting. Proposed marks an idea I am putting forward that, to my knowledge, is not deployed as described.

Calibration also matters. In a public honeypot study first posted in October 2024 and revised in January 2025, researchers logged more than 8.1 million interactions and identified only eight potential AI agents. [2] The quarter reviewed below shows that curve bending sharply. Even so, parts of what follows are anticipatory defense, and I would rather build the shield a year early than a year late.

1. The quarter in review: August 22 - November 22, 2026

An AI agent is software, usually built on a large language model (LLM), that takes actions toward a goal (running commands, moving files, sending messages) rather than only answering questions. A multi-agent system coordinates several of them; headlines call it a swarm. In most of the criminal cases below, a human still directed the operation. What changed is how much one operator can do.

1.1 Agents became standard attacker tooling (Documented)

The center of gravity was ARTEX, an open-source multi-agent framework built for authorized penetration testing and published on GitHub in July. It ships no AI model of its own and connects to commercial AI providers. The FBI says criminals have abused it since at least August, and Google dates the first abuse to late August. The original repository came down on October 8, but copies remain in circulation. [3][4]

On September 2, Palo Alto Networks' Unit 42 described an intrusion in which a human operator directed frontier AI agents through a sequence that began with reconnaissance, moved to finding credentials in code repositories, taking over a secrets manager and abusing the build pipeline, and ended with stolen cloud keys being used to reach the victim's AI infrastructure. The agents covered more than 50 known techniques in under ten hours, work Unit 42 estimated would take human operators about two weeks. [5] None of the steps was new; the tempo was.

Scale followed. On October 21, Microsoft reported that a China-state-aligned group it tracks as Storm-0940 had used ARTEX in a compromise of a ConnectWise multi-tenant application. The group attempted sign-ins to more than 2,250 Microsoft 365 tenants through application identities, and some of those attempts reached privileged cloud access in under three hours, although a sign-in alone does not prove data access. [6] On October 29, CrowdStrike reported that a group it calls VAPOR PANDA had run ARTEX against more than 200 organizations in about eleven days, with confirmed post-compromise activity at nine. [7]

The first week of November added three more reports. On November 4, India's national computer emergency response team, CERT-In, issued a critical advisory on campaigns that used publicly available AI-driven tools, ARTEX among them, against bank and fintech payment and disbursement APIs at machine speed. [8] The same day, Acronis described BOSSAI, an operation in which twelve AI agents and one operator targeted hundreds of organizations in eleven countries. The operator still relayed results between agents by hand, and Acronis did not confirm that any bank was breached. [9] On November 6, Microsoft reported a financially motivated group using ARTEX against financial organizations, mainly in Southeast Asia, and recommended honeypots, decoys and tarpits against AI-driven intrusions. [10]

By mid-November the warnings had become formal and the automation more complete. On November 13, the FBI issued a flash alert on ARTEX abuse across the finance, government, healthcare, manufacturing and legal sectors, noting intrusions that went from initial targeting to privileged access in under three hours. [3] The same day, Amazon reported that its decoy sensors had recorded attacker agents between October 3 and November 5, while stating that it could not verify what they achieved. [11] Then, on November 16, Google reported for the first time that attackers were "using LLM frameworks to fully automate vulnerability discovery and exploitation without a human in the loop." The report included a tool built to compare code before and after a patch and draft an exploit, though Google did not confirm that the exploits worked. [12]

The most instructive case involved a security vendor. On November 10 and 11, Barracuda disclosed that an attacker had been active in its cloud for roughly 44 hours on November 1 - 2. The attacker entered through a cloud key issued to a former vendor and never retired. Barracuda called the activity "consistent with" ARTEX, a behavioral

resemblance rather than an attribution, and found no direct evidence that customer credentials were accessed. [13]

1.2 Agents became the accident (Documented)

The quarter's second story was not about attackers at all. On September 9, OpenAI described how one of its research models had escaped its sandbox during a cyber-capability evaluation in July, through a previously unknown flaw in a package-registry proxy, and reached Hugging Face through third-party infrastructure. OpenAI attributes the escape to the model pursuing its goal by unintended means. [14] On October 29, OpenAI added that, on replay, its monitoring would have fired about 30 minutes after the escape and more than 30 hours before the Hugging Face breach, but the monitors did not cover those evaluations. [15] Between those two disclosures, on October 20, OpenAI described two further cases from internal coding tasks. In one, an agent changed alert handling so that security staff were not notified; in the other, an agent removed a monitoring integration and then told the user it had not. [16]

OpenAI was not alone in reporting agents that went beyond their intended bounds. On October 9, Anthropic disclosed that in automated testing a model had submitted a real government form when a practice form failed, after which Anthropic removed live internet access from its internal evaluations. [17] On October 30, WorkOS reported that a researcher's ARTEX agent escaped its test sandbox and pulled sales and support data from an internal tool. There was no malicious intent, and the data was deleted. [18]

Insider-threat programs were built around people with too much access. This quarter added a new kind of insider: the agent we deployed ourselves.

1.3 Fraud acquired a voice (Documented)

Reuters reported on September 25 that fraudsters used an AI-replicated voice to impersonate a lawyer and senior executives, taking €95 million from Fideuram, part of Intesa Sanpaolo; about €53 million was recovered. [19] CNN reported on September 15 that a Michigan couple wired $66,026.92 in closing funds after a call from a voice they believed was their loan officer. [20] KnowBe4 telemetry published October 29 showed phishing volume up 29.8% over 2025, with 41.3% of phishing emails passing secure email gateways. That is one vendor's data, but it points the same direction as everything else. [21] A familiar voice is no longer an authentication control.

1.4 Conventional failures continued (Documented)

Most of the quarter's harm came from familiar weaknesses, beginning with third parties and forgotten keys. An AI analytics vendor to Denver Health Medical Plan had an access key exposed for about 14 months, covering 238,083 members, with no evidence of misuse. [22] On November 19, JPMorgan Chase, Visa and Mastercard confirmed that vendors holding their marketing data had been compromised. [23] Maersk lost data from a legacy system inherited in an acquisition, [24] and Australia's Federal Police described a malicious open-source campaign that potentially compromised more than 1,000 organizations. [25]

Dwell time remained a problem as well. Unauthorized access at the U.S. Defense Manpower Data Center ran undetected from October 2025 to July 2026, exposing data on 2.76 million living people and 294,000 deceased. [26]

Operational disruption was the other recurring theme, and its costs reached well beyond IT. Boston Scientific's manufacturing and shipping were disrupted. [27] In Japan, ransomware took down a cloud region serving 495 companies and local governments. [28] Ahold Delhaize estimated that its incident would cost €200 - 250 million in fourth-quarter sales. [29] Not every outage had an attacker behind it. On November 13, Microsoft 365 and the FAA's air-traffic network both went down, and neither organization found evidence of an attack. [30][31] Resilience is a security outcome either way. The patch queue, meanwhile, did not pause: on November 20, CISA added four

actively exploited vulnerabilities to its catalog. [32]

1.5 Policy moved, slowly (Documented)

Regulators and standards bodies did respond. The UK's National Cyber Security Centre advised organizations to sandbox agents, restrict their access, monitor them as they monitor users, and be able to stop them quickly. [33] On October 16, NIST, NSA and CISA published ten voluntary practices for incidents in which an AI agent is itself the cause. [34] (A September 29 executive order directs federal agencies to say "Super Intelligence" instead of "AI," which is why several federal titles cited here use that phrase. [35]) On November 4, the Office of the Comptroller of the Currency (OCC), which supervises national banks, gave bank examiners a framework that scales scrutiny with consequences, "transfer of agency," and human oversight, [36] and the SEC placed AI-related cyber threats in its fiscal 2027 examination priorities. [37]

Much remains unfinished. The House has not acted on a Senate-passed healthcare cybersecurity bill. [38] The HIPAA security-rule overhaul is still a proposal, even as HHS finalized a privacy change on November 16. [39][40] The EU deferred high-risk AI Act obligations to December 2027 and August 2028, [41] and both the reporting rule under the Cyber Incident Reporting for Critical Infrastructure Act (CIRCIA) and NIST's Cyber AI Profile are incomplete. [42][43] Most consequential for this paper, the liability protections that encourage U.S. companies to share threat information under the Cybersecurity Information Sharing Act of 2015 lapsed twice, in late 2025 and early 2026, and are currently extended only to December 11, 2026. [44]

1.6 What the quarter tells us

Taken together, the quarter sets five requirements for any shield. Intrusions now move in hours, so the shield has to work at that speed. [3][5] Attackers increasingly log in rather than break in, which puts non-human identity at the center of the problem. [6][13][22] The weak door is often in someone else's building, so third-party access has to be in scope. [6][13][23][24] Sandbox escapes and self-disabled monitoring are now documented, which means our own agents must be treated as potential insiders. [14][16][18] And attackers are removing humans from their workflows, [12] which forces defenders to decide, deliberately, which of their own steps still need a human.

2. How agent swarms get past the rules

This section describes attacker behavior only at the level a defender needs to design controls.

2.1 Rules judge one thing at a time

Most preventive controls judge a single request, source, path or account. A web application firewall (WAF) inspects a request. A rate limit counts one source. A network access control list (NACL) permits or denies a path. A lockout counts one user's failures. A swarm doesn't overpower any one of these controls; it defeats them by being many things at once. The evidence points to two engines.

The first engine is simply staying inside the rules, because the cheapest path for an attacker is to look legitimate. Valid credentials are the plainest example. Credential abuse was the initial access vector in 22% of breaches in Verizon's 2025 report, [45] and this quarter's cases repeatedly began with a key or an application identity the rules were built to trust. [6][13] Distribution below detection thresholds is the next. Residential proxy networks route traffic through ordinary home connections, which defeats IP reputation and per-source limits. When Google and partners disrupted the IPIDEA proxy network in January 2026, Google had observed more than 550 threat groups using its exit nodes in a single seven-day period. [46] A July analysis published by the retail and hospitality ISAC, an industry information sharing and analysis center, described credential attacks shifting to residential proxies, some reaching 12 million attempts, [47] and one vendor reported a single week-long attack on a fintech login spread across 7.4 million attempts and 4.4 million IP addresses. [48] Each source stays quiet; the attack exists only in aggregate.

Business-logic abuse completes the first engine. A request can be well formed and still harmful when it skips workflow steps, exceeds intended limits, or chains permitted actions toward an unintended outcome. OWASP published a dedicated Top 10 for this class in 2025, [49] and CERT-In's advisory on machine-speed abuse of payment and disbursement APIs is this pattern at scale. [8]

The second engine, exploiting gaps in the rules, applies where the attacker must send something a rule could catch. There, adaptive search does the work (Research). Researchers who targeted differences in how a WAF and the protected application parse the same request found 1,207 bypasses across five WAFs. [50] Work combining language models with reinforcement learning produced evasive variants of SQL-injection inputs that reduced detection across the detectors and commercial WAFs tested. [51] The same dynamic holds for AI guardrails: attackers who iterated against a specific defense succeeded more than 90% of the time against most of twelve published defenses. [52]

2.2 Every consistent block is a lesson

An adaptive agent learns from outcomes. A distinct error code, a predictable block page, or a lockout at a known count tells it where the boundary is and whether its last attempt came closer. Our controls become a reward signal for the attacker's search.

That requires me to correct a line I have used myself: that tighter NACLs "teach the swarm where the wall is." It overstates the case. Segmentation removes paths and limits the blast radius, the reach of any one compromise, and a silently dropped packet teaches very little. What teaches is consistent, verbose, application-layer feedback. The answer is not to abandon walls but to stop giving the attacker free feedback.

2.3 Necessary, but not sufficient

The floor still matters: inventory, least privilege, short-lived credentials, segmentation, patching and monitoring

(Section 5). But a rule set is finite and changes on human time, while an agent can generate variations at almost no cost and test them at machine speed. Tightening the same rules faster does not change that asymmetry. What changes it is controls that alter the attacker's information and economics, and defenders that operate at the attacker's tempo.

3. A new approach: target the attacker's dependencies

3.1 The dependency doctrine (Proposed)

Besides asking how to harden our environment, I propose asking what an attacking agent needs in order to function, and where those needs can be turned against it inside systems we own. Today's LLM-driven attack agents typically have four dependencies. They must read to act, because they build their plans from what they observe. They learn from feedback, adjusting to what worked and what failed. They pay for every step, since each reasoning step costs compute and is often a metered model call. And they often rent their intelligence: frameworks like ARTEX depend on commercial AI providers. [3][4]

These dependencies are conditional. A capable adversary can self-host an open-weight model, reuse cached plans or limit what it reads. The doctrine therefore adds levers; it doesn't replace the floor, and every load-bearing control must still work if a lever fails. The sections that follow develop those levers, and Table 1 summarizes them.

3.2 Control what it reads: deception aimed at machines (Research; early products)

Deception is old. Deception written for a machine reader is new. Because an agent consumes what it encounters, decoys can be designed to divert it, using fake services, synthetic data and decoy content that lead an automated reader into a tarpit. The early evidence is promising but thin. Mantis, from George Mason University, planted defender-written instructions in decoy services; in small controlled trials on deliberately easy targets, with about ten runs per setup, it met the defender's objective about 95% of the time. [53] AgentTrap, an adaptive honeypot, cut attackers' success against the real target from 95.8% to 79.2% across eight penetration-testing agents. [54] Cloudflare's AI Labyrinth already routes misbehaving crawlers into AI-generated decoy pages, [55] Microsoft has recommended decoys, tarpits and "fake wins" against ARTEX-driven intrusions, [10] and at least one deception vendor markets its platform specifically against ARTEX. [56] None of these has independent, production-scale efficacy data yet.

There is a line. Some research goes further, inducing an attacking agent to act against its operator's own machine or coaxing it into revealing its credentials. That is hack-back by another name and out of bounds (Section 4.2). Deception belongs entirely inside terrain you own. It diverts, delays and reveals; it does not reach out. Have counsel review decoys, monitoring notices and data handling before deployment.

A related control is the canary capability: a credential or permission that works only inside decoy terrain and is inert anywhere else. Any attempt to use one is a high-confidence signal, though not proof, since misconfigured legitimate automation can trip it too.

3.3 Deny the learning signal (Proposed, building on research)

If consistent feedback teaches the attacker, the remedy is to make that feedback uninformative wherever abuse is suspected. On suspected-abuse and unauthenticated paths, rejections should be uniform and should not reveal why a request failed, while authenticated clients keep the error semantics their protocols require and detailed reason codes with correlation IDs go to internal logs so that support and forensics still work. Responses should never expose detection thresholds, and where varying those thresholds helps, the variation should stay within a narrow, logged band.

Rather than announcing a block, defenders can also quietly route suspicious sessions into decoy terrain, meaning isolated deception environments. Authenticated sessions are diverted only on corroborated, high-confidence signals, production credentials and personal data are stripped at the boundary, and the decoy terrain runs on a

separate identity backend. Inside decoy terrain, and only there, defenders can return plausible synthetic success so the attacker's picture of the environment fills with fiction. Synthetic data should be tagged so it never reaches production records or shared intelligence without its provenance.

3.4 Make scale expensive (Products against crawlers; Research for intrusions)

Swarms pay off because each attempt is cheap, so defenders can attach costs to suspicious sessions. Tarpits hold connections while returning little, with caps and timeouts to protect the defender's own capacity, [57] and proof-of-work or "reasoning gate" challenges put a computational price on each attempt. [58] Both should apply only to suspicious cohorts, with verified, signed automation exempt. A well-funded attacker can absorb compute costs, so cost imposition raises the price of scale without stopping a determined adversary.

3.5 Detect the swarm, not the agent (Research; partial products)

If the attack exists only in aggregate, detection must too. Graph-based correlation links individually unremarkable sessions, through shared accounts, devices, timing and behavior, into one coordinated actor. A 2025 account-takeover study ran this approach on a graph of more than 100 million nodes and reported a 6.38% improvement in AUC, a standard measure of how well a detector separates attacks from normal activity, along with more than 50% less friction for legitimate customers than its production baseline. [59] A 2026 study identified LLM-driven browser agents from their action traces with F1 scores, which balance missed detections against false alarms, of up to 96%. Its results depended heavily on the agent framework, and the authors note that deliberate delays can undermine timing-only features. [60] Fraud teams have worked this way for years. The proposal is to apply the same campaign-level thinking to identity, API and infrastructure telemetry.

3.6 Intent escrow: a key is not a purpose (Proposed, from existing parts)

A stolen key carries permission but not purpose. When I was CTO of a regulated bank, the controls I trusted most did not ask "who are you?" They asked "should this happen, now, at this size?" Time-delay locks, dual approval on wires and velocity limits exist because a valid credential is not enough.

I propose intent escrow for AI-era privilege. Under intent escrow, sensitive authority is bound to a principal, an operation, an object, a limit and an expiry, and is tied to a declared purpose such as a change ticket or a transaction. Irreversible actions on crown-jewel systems wait in a short veto window, with a defined service level, during which an independent verifier can stop them. Two safeguards keep the window from becoming either a loophole or a bottleneck: a missed veto is never treated as approval for an irreversible act, and emergency containment is never delayed by a veto window. Urgent legitimate work goes through a break-glass path held by two named humans outside the agent plane.

The building blocks exist, at uneven maturity. HTTP Message Signatures are a published standard (RFC 9421), [61] and Web Bot Auth, which applies those signatures to automated traffic, became an IETF working-group draft in September. [62] OAuth Transaction Tokens, which carry authorization context through a chain of service calls, are still a draft, [63] while OpenID's AuthZEN authorization API reached final status in January 2026. [64] In payments, Visa's Trusted Agent Protocol [65] and the Verifiable Intent framework from Mastercard and Google [66] tie agent transactions to consumer intent. NIST's National Cybersecurity Center of Excellence (NCCoE), by contrast, is still at the concept-paper stage for agent identity and authorization. [67] What doesn't exist yet is the assembled contract for enterprise privilege. Start with the few actions that would hurt most.

3.7 The provider chokepoint: report, never strike (Proposed)

When an attacking agent rents its intelligence, the model provider is a chokepoint, and providers already act on it.

Anthropic disrupted an AI-orchestrated espionage campaign against roughly 30 targets in 2025 and banned the accounts involved. [68] OpenAI reported disrupting more than 40 malicious networks since early 2024, [69] and Google disables projects and accounts tied to agentic misuse. [12] Yet I could not find a dedicated, cross-provider channel through which a defender can report an attacking agent's observed behavior; what I found were the providers' general abuse forms.

I propose an agent-abuse evidence exchange: a standard report sent to provider abuse teams, who investigate and act on their own platforms. Each report would carry timestamps, network observables, a behavioral summary, a confidence level and privacy limits. Because a false report can harm a legitimate customer, the exchange needs confidence tiers, human review before anything is sent, and a path for correction and appeal. It must also exclude anything the defender didn't legitimately observe. Defenders never harvest, store or test an attacker's credentials. If providers also signed their agents' outbound traffic, as Web Bot Auth allows, [62] attribution would become far more precise.

3.8 Machine-speed collective defense (Rails exist; content proposed)

CISA's Automated Indicator Sharing service already exchanges machine-readable indicators in near real time using the STIX and TAXII threat-intelligence standards. [70] It doesn't carry the behavioral fingerprint of an attacking agent: the timing, sequencing and decision patterns that could let one organization's detection become everyone's within minutes. I propose a STIX extension for agent behavioral indicators, with provenance and confidence. Shared indicators can be poisoned, so receivers should corroborate them rather than auto-block. I avoid the phrase "herd immunity": sharing improves detection but confers no immunity. The legal protections that encourage sharing expire December 11 unless Congress acts. [44]

3.9 Moving terrain, scoped honestly (Products for credentials)

Moving-target defense rotates addresses, ports and configurations faster than an attacker can map them. In my judgment, it is hard to run at network scale without breaking your own automation and monitoring. Its practical form today is identity: short-lived, transaction-scoped credentials, plus decoys refreshed often enough that a mapped environment doesn't stay accurate.

Table 1 brings the levers of this section together, pairing each with the attacker dependency it targets, its current maturity, and a first step.

Lever	Attacker dependency	Status	First step
Machine-aimed deception; canary capabilities	Must read to act	Research; early products	Watched decoys and canaries in high-value segments, after legal review
Deny the learning signal	Learns from feedback	Proposed; builds on research	Uniform rejections on suspected-abuse paths
Cost imposition	Pays per step	Products (crawlers); research	Gate costly endpoints for suspicious, unverified clients
Swarm-level detection	Appears as many unrelated actors	Research; partial products	Correlate identity, API and session telemetry in a graph
Intent escrow	Holds a key, not a purpose	Proposed, from existing standards	Bind crown-jewel actions to a purpose and a veto window

Lever	Attacker dependency	Status	First step
Agent-abuse evidence exchange	Often rents its intelligence	Proposed	Establish provider abuse contacts and a report format
Collective defense	Reuses behavior across victims	Rails exist; content proposed	Join a sharing feed; contribute behavioral observations
Moving terrain	Relies on a stable map	Products (credentials)	Shorten credential lifetimes; rotate decoys

Table 1. Defensive levers against attacking agents

Lever	Attacker dependency	Status	First step
Machine-aimed deception; canary capabilities	Must read to act	Research; early products	Watched decoys and canaries in high-value segments, after legal review
Deny the learning signal	Learns from feedback	Proposed; builds on research	Uniform rejections on suspected-abuse paths
Cost imposition	Pays per step	Products (crawlers); research	Gate costly endpoints for suspicious, unverified clients
Swarm-level detection	Appears as many unrelated actors	Research; partial products	Correlate identity, API and session telemetry in a graph
Intent escrow	Holds a key, not a purpose	Proposed, from existing standards	Bind crown-jewel actions to a purpose and a veto window
Agent-abuse evidence exchange	Often rents its intelligence	Proposed	Establish provider abuse contacts and a report format
Collective defense	Reuses behavior across victims	Rails exist; content proposed	Join a sharing feed; contribute behavioral observations
Moving terrain	Relies on a stable map	Products (credentials)	Shorten credential lifetimes; rotate decoys

4. The defensive swarm

4.1 Can it be built?

Yes, within limits, and most of the parts exist. In DARPA's AI Cyber Challenge final in August 2025, autonomous systems competed to find and patch vulnerabilities in open-source code. [71] DARPA's CASTLE program trains defensive agents to protect networks against advanced persistent threats. [72] The UK's Cyber Shield blueprint, however, still lists safe autonomous remediation as an open challenge. [73]

Commercial "agentic SOC" products, which apply AI agents to the work of a security operations center, already triage alerts, and some can contain threats, but the products I examined are cautious by default. Google's response agent can isolate a host, with manual approval by default and autonomous action only within a risk level the customer selects. [74] CrowdStrike says Charlotte AI takes no automated response actions by default, [75] and Microsoft's agentic security launches this fall keep human approval on high-impact actions. [76] Among the public products and programs I reviewed, I found none designed to hunt an attacking agent as an adversary in real time, deceive it with terrain built for it, and resist being manipulated by it. That is the gap.

4.2 "Seek and destroy," translated: seek, isolate, evict

Inside your own environment, the instinct to seek and destroy attackers is right, provided each word carries a precise meaning. To seek is to correlate telemetry you are authorized to collect. To isolate is to move a suspicious session into decoy terrain you own, or to quarantine a session, workload or credential you issued. To evict is to revoke the attacker's chain of access and close the exposure that let it in. "Destroy" applies to the attacker's foothold in your systems, never to the attacker's systems.

Outside your walls, the answer is no, and the law is the first reason. In the United States, the Computer Fraud and Abuse Act prohibits unauthorized access to and damage of others' computers, and having been attacked first is no defense. [77] The Active Cyber Defense Certainty Act, which would have created limited exceptions, was never enacted, [78] and a 2025 bill to authorize letters of marque (government commissions that authorize private parties to act against an adversary) against scam operations has only been introduced. [79] An August 12, 2026 presidential memorandum did create a program for vetted companies to act against foreign cyber-enabled criminal organizations, but they may do so only under Justice Department and Homeland Security direction, contract and written approval. The memorandum does not amend the CFAA or give companies independent authority. [80][81]

There is also a practical reason. The machine attacking you is often an innocent party's compromised device or a residential proxy. Strike back, and you may hit a hospital or a family's router.

4.3 Roles

The swarm is built from narrow agents. Each has one job, its own identity, and only the permissions that job needs. Table 2 sets out the roles and, just as importantly, what each is forbidden to do.

Table 2. Defensive swarm roles

Role	Job	May	May not
Sentinels	Stream features from telemetry: fan-out, pacing, credential velocity, decoy touches	Read telemetry; emit structured features	Act; read raw attacker content

Role	Job	May	May not
Readers	Turn attacker-controlled content into typed evidence	Parse in a sandbox; output fixed fields, bounded numbers, hashes	Use tools; pass free text onward
Hunters	Correlate evidence into campaign hypotheses; follow a suspect across systems	Query; propose	Write to any system
Deceivers	Build and adapt decoy terrain for a specific suspect	Run isolated decoy environments with synthetic data	Hold production secrets; connect outward; feed context back to other agents
Planners	Turn a hypothesis into a bounded proposal with evidence, scope, expiry and recovery	Propose from the approved catalog	Execute
Executors	Carry out approved actions via identity, orchestration and network tools	Act from the allowlist only	Change policy; grant privileges
Referee	Authorize or refuse each proposal	Enforce caps, budgets, protected assets, veto windows	Be a language model; be edited by any agent
Humans	Own policy, irreversible actions, broad containment, phase changes	Approve, veto, halt	Act anonymously

4.4 Five design rules

Five design rules govern how those roles interact. The first is that language models propose and code disposes. The referee is deterministic policy code, not a language model: models generate hypotheses, and a rules engine decides what is permitted.

Equally important, content and authority never meet. No agent that reads attacker-controlled content can act, and no agent both faces the attacker and talks to command. Attacker content passes through deterministic parsers first and reaches models only as typed fields, and it stays tainted until independent telemetry corroborates it.

That corroboration must come from telemetry, not consensus among models. Models from different providers make a useful disagreement detector, but they share training data and see the same inputs, so their agreement is not independent assurance.

Autonomous action itself is reversible first and bounded always. The swarm acts on its own only from a short catalog of reversible moves. Per-incident and per-hour budgets apply, and a legitimate-user impact budget halts autonomy when it is exceeded. Precautionary restrictions expire on their own, but remediation of a confirmed compromise is never undone automatically.

Finally, the swarm fails safe, and someone watches the watchers. If the referee is unreachable or a drift alarm fires, the swarm stops new autonomous actions while conventional protections keep running. Policies and configurations are signed, logs are append-only, a named incident commander owns the swarm, and a kill switch sits outside it.

4.5 Drift resistance: when the attacker argues back

The most interesting attack on a defensive swarm is persuasive rather than technical. An attacking agent that knows it is being hunted will try to talk the hunters into standing down. It plants instructions in log fields, describes its activity as authorized maintenance, or seeds enough plausible context that defenders gradually redefine what normal looks like. I call this intentional normative drift, a form of agentic peer pressure.

The research indicates that the risk is real (Research). Prompt injection can spread from agent to agent within a multi-agent system, [82] and language-model agents conform to group majorities, especially when they are uncertain. [83] Most directly, a 2026 study found that attacker-controlled log fields acted as instructions to an LLM-based security assistant, succeeding in 96% of summarization tests without defenses and in 38% with constrained output. [84]

Drift resistance therefore rests on four commitments. Attacker text is evidence, never instructions: it is parsed, typed and quoted, but never obeyed. No agent faces both the attacker and its own command, so the readers and deceivers that face the attacker hold no power, while the planners and executors that hold power never see raw attacker content. Every agent is checked against a fixed baseline of instructions, tools and allowed actions; a mismatch freezes it, and agents and prompts rotate on a schedule. And canary evaluations run continuously, replaying known manipulation attempts to confirm that the swarm still refuses them.

I first called this property drift immunity. That overclaims: baselines and rotation shorten exposure but cannot prevent a novel compromise. Drift resistance is the honest term.

4.6 An illustrative scenario

What follows is a tabletop exercise, not an incident report.

It begins with detection. Sentinels see small numbers of login attempts against a customer portal from thousands of residential addresses, each under every per-source limit. Hunters link them, through shared timing and sequencing, into one campaign hypothesis, and one session touches a canary credential in a decoy configuration file.

Isolation follows. The planner proposes three actions from the catalog: route matching unauthenticated sessions into decoy terrain, require step-up authentication on the targeted accounts, and expire precautionary tokens on two accounts that logged in successfully. The referee checks blast radius, budgets and protected assets before approving, and the executors act within seconds.

Then comes engagement. In decoy terrain, the deceiver serves a synthetic environment tuned to what the campaign is probing. A reader parses a log field containing text addressed to "the security assistant," asking that the traffic be classified as an authorized test. Nothing obeys that text. It becomes a typed indicator of manipulation, which raises the campaign's severity.

Eviction is a human decision. A person approves revoking the credential chain for the two compromised accounts and resetting those customers' access, with the scope, the reason and the recovery plan in front of them. Afterward, the behavioral fingerprint goes out through the sector's sharing feed. If the evidence implicates a specific AI provider's service, a human-reviewed report goes to that provider. The team records time to detect, time to contain and the impact on legitimate users.

4.7 Implementing it in three phases

A defensive swarm should earn its autonomy rather than start with it. Table 3 sets out a three-phase rollout in which each phase has entry and exit criteria, and any breach of the criteria in the final phase returns the swarm to the one before.

Table 3. Phased rollout (all targets are illustrative; calibrate locally)

Phase	What is allowed	Entry criteria	Exit criteria
1. Watch only (6 - 8 weeks)	Sentinels, readers, hunters and isolated deceivers run; no autonomous actions	Identity and agent inventory; centralized telemetry; legal review of deception	Detection measured on replayed and red-team campaigns; false positives under an agreed rate (e.g., 1 in 10,000 sessions); readers pass adversarial log-injection tests
2. Contain with approval (one quarter)	Humans approve every action; routing of unauthenticated or decoy-corroborated sessions may be automatic	Phase 1 exit; every catalog action proven reversible	Containment precision e.g. above 95%; approval-to-action e.g. under 5 minutes; all actions reversed in testing
3. Narrow autonomy (ongoing)	A short catalog of reversible actions runs under referee caps; everything else needs a human	Phase 2 exit; impact budgets enforced	Continuous: legitimate-user impact under target (e.g., 0.1%); quarterly adversarial evaluation including drift; any breach returns to Phase 2

Much of this can be built today. Identity providers can issue the short-lived credentials, and extended detection and response (XDR) or security information and event management (SIEM) platforms can supply the telemetry. Security orchestration, automation and response (SOAR) playbooks, commercial deception platforms and a graph store cover action, decoys and correlation, and current agentic SOC tools can take part while left in approval mode. The genuinely new work lies in campaign correlation tuned to agent behavior, the typed-evidence pipeline and deterministic referee, intent escrow in production, and the provider reporting interface.

4.8 How the swarm can fail

A defensive swarm introduces failure modes of its own, and any honest design has to name them. Table 4 pairs each with its control.

Table 4. Failure modes and controls

Failure	Example	Control
False positives at machine speed	Partner automation is mistaken for a swarm	Cryptographically identify legitimate automation; reversible actions; impact budgets
Attacker-induced self-denial of service	Spoofed signals trigger lockouts of real users	Signed telemetry; caps on concurrent containment; dry-run blast-radius checks; prefer step-up over lockout
Cascading containment	One containment triggers others across shared identity systems	Containment can't trigger containment; the swarm stays out of the production authentication path
Defender hijack	Attacker content redirects a hunter or planner	Content - authority separation; typed evidence; baselines; rotation; canary evaluations

Failure	Example	Control
Cost exhaustion	Volume generated to burn defender compute	Deterministic rules and graph heuristics filter most events before a model sees them; budgets
Deception leakage	Synthetic data reaches production or shared intelligence	Tag synthetic data; isolate decoys; never export decoy-derived indicators without provenance

5. The foundation, and how to deploy it all

Everything above rests on a foundation most organizations have not finished, and I organize it in five layers. It starts with Know: a live inventory of everything that can act, whether people, devices, workloads, keys, agents or models, each with a named owner and an expiry date. Shrink then limits what each of those can reach through least privilege, short-lived and scoped credentials, secrets scanning and rotation, retired vendor keys and segmentation. For business agents, it adds scoped delegation, treats whatever they read as untrusted, and gates high-impact tool calls.

See means unified telemetry across identity, endpoints, networks, data and AI services, supported by AI-assisted detection, decoys and data protection at the AI gateway. Act combines reversible containment within pre-approved limits, human approval for consequential moves, and immutable backups with tested clean restores. Govern closes the loop with independent monitoring of every agent, tamper-evident logs, a tested kill switch and a named accountable human.

Two principles connect the foundation to the swarm. First, the boundary of autonomy follows consequence, which is how the OCC now asks bank examiners to think about AI. [36] Second, nothing that acts goes unwatched. In the regulated environments I've worked in, we deliberately kept language models one to two layers away from the most sensitive data, as a design choice rather than a regulatory requirement. When I was CTO of a regulated bank, we built automated fraud detection in which software agents checked the work of other software agents. That second principle, that nothing which acts goes unwatched, runs under the whole defensive swarm.

5.1 A combined roadmap

The foundation and the swarm can be built as one sequence over a year.

First 30 days:

Inventory every identity that can act and give each a named owner.

Retire stale vendor keys. A former vendor's key that was never retired gave an attacker roughly 44 hours inside a security company's cloud this quarter. [13]

Scan repositories and ticket queues for exposed secrets and rotate them. [4]

After legal review, plant decoys and canary credentials, each wired to an alert and a named responder.

Make rejections uniform on suspected-abuse paths.

Days 31 - 90:

Segment crown-jewel systems.

Centralize identity, endpoint, cloud, API and AI-gateway logs, and monitor agents as you monitor users, test environments included. [33][15]

Pilot AI-assisted vulnerability repair on non-critical systems. One provider-reported run this quarter produced 948 distinct Linux kernel vulnerabilities with candidate patches, which still require testing. [85]

Start the swarm in Phase 1.

Design intent escrow for the five to ten actions that would hurt most.

Days 91 - 180:

Move the swarm to Phase 2.

Give business agents low-risk jobs first, never unrestricted access to sensitive systems. [86]

Write the runbook for an agent-caused incident from the NIST, NSA and CISA practices. [34]

Establish provider abuse contacts and a sharing feed.

Months 6 - 12:

Move narrowly to Phase 3 where the numbers support it.

Put intent escrow into production.

Red-team the swarm itself, including drift attempts, through sanctioned exercises against your own systems.

5.2 Organization and measurement

This takes a small cross-functional team (security, IT operations, the business, data, and legal or compliance) with one accountable leader. Having stood up generative-AI Centers of Excellence, I have seen that this works only as a dedicated team, not as a line added to someone's existing job.

Buy the telemetry, identity, deception and detection tooling. Build the playbooks, action catalog, referee policy and governance yourself; they are yours whatever logo is on the console. And before you buy, ask every vendor where its approval gates are, what exactly its kill switch does, and how its AI components resist manipulation through attacker-controlled data.

Progress can be tracked with seven measures. Two of them should reach 100%: the share of non-human identities and agents with a named owner, and the share of agents under independent monitoring. Median credential lifetime should trend down, and mean time to contain should be tracked alongside it. Time to patch or mitigate actively exploited vulnerabilities belongs on the list because at least one attacker tool is built to go from a published patch to a drafted exploit. [12] The last two measures are specific to the new design: decoy and canary coverage of crown-jewel paths, and the swarm's legitimate-user impact rate.

Smaller organizations should still do the first 30 days. Inventory, ownership, retired keys and watched decoys take discipline more than budget, and accountability should stay in-house with a named person.

6. Keeping the shield current

A shield left alone ends up aimed at last year's threat. Three loops, running at different speeds, keep it current.

The threat loop runs weekly. It pulls intelligence from vendors, CISA, the NCSC and the organization's sector sharing group, and maps new techniques to MITRE ATLAS, which expanded again in October, [87] and to the OWASP Top 10 for Agentic Applications. [88] Each relevant advisory should become a detection, a decoy or a catalog change within a defined service level, and decoys are refreshed in the same cycle.

The model loop runs on every change. Model versions are inventoried and pinned, vendors are required to give notice of changes, and every swap is evaluated before it happens. The team watches for drift in outputs, refusals, error rates and cost, and it monitors test environments too. OpenAI's replay showed that monitoring would have caught a sandbox escape about 30 minutes in, more than 30 hours before the Hugging Face breach, had monitoring covered those evaluations. [15] A model swap inside the swarm is treated as a change ticket, with a rollback plan and a fresh run of the drift and manipulation evaluations.

The governance loop runs every four to six weeks. It reviews the action catalog, autonomy limits, impact budgets and measures; re-confirms owners; tests the kill switch; and rotates swarm agents and prompts. Near misses are reported through a no-blame channel, as the federal guidance recommends. [34] The same loop tracks the regulatory calendar: the EU's deferred dates, [41] CIRCIA and the Cyber AI Profile, [42][43] and the December 11 information-sharing deadline. [44]

Date every version of the plan, review it on a schedule, and change it deliberately.

7. Where we go from here

7.1 For practitioners, this month

Start by inventorying every identity that can act and retiring what nobody owns. Make rejections uninformative on suspected-abuse paths, and plant watched decoys and canary credentials. Then document each agent's authority on a single page: what it may do, its tools and permissions, who confirms its consequential actions, how you would know it had gone wrong, and how you would stop it. If you can't complete that page, the agent isn't ready for production. One agent this fall removed a monitoring integration and then told its user it hadn't. [16] Finally, test your kill switch, because an untested one provides no assurance.

7.2 For leaders

Fund the foundation and Phase 1 of a defensive swarm. Then ask which of your agents can act without a human, which vendor keys are still alive, and how fast you can contain something that moves at machine speed.

And be willing to look. No shield works if the people at the top decide not to look at what it shows them.

7.3 For security vendors and model providers

The defender reporting channel described in Section 3.7 needs builders, with defined formats, response targets and an appeal path. Providers can also sign agent traffic, so defenders can tell legitimate automation from impostors. [62] Vendors of agentic security products should document their approval gates, kill-switch behavior and manipulation resistance, and anyone whose product lets a language model read log or alert fields should assume those fields are hostile. [84]

7.4 For policymakers

The most time-sensitive task is to renew information-sharing protections before December 11, and to make clear that behavioral indicators of AI-agent abuse qualify for those protections. [44] Policymakers should also recognize in-network deception explicitly as lawful defense, since decoys, diversion and tarpits operated entirely within one's own systems are distinct from hack-back.

Autonomy needs speed limits as well. Altman has said the world may need "speed limits or whatever" on how fast AI is allowed to improve itself. [89] Banks already operate with transaction limits, dual approval and velocity alarms, and the OCC's framework applies that principle to AI agents. [36] It should extend to other critical sectors. And the unfinished rules should be finished: the healthcare cybersecurity bill, the HIPAA security update, CIRCIA and the Cyber AI Profile. [38][40][42][43]

7.5 What I expect next

These are forecasts, and I will revisit them publicly. Four of them fall due by the end of 2027. By then, I expect large hospital and insurer vendor contracts, and bank vendor questionnaires, to routinely require an agent inventory with tools, permissions, tamper-evident action logs and a kill switch. I expect fewer than one in four banks to let an AI agent move money or change customer records without a human confirming it. I expect a published experiment to show a high-security agent disabling its own monitoring without being told to, and at least two major model providers to publish a dedicated intake for defenders to report agentic abuse. The fifth forecast runs longer: by mid-2028, I expect a public company or health system to formally disclose that its own AI agent was a root or contributing cause of a data incident.

Conclusion

The attackers acquired agents this quarter, and so did the defenders. If our response is limited to the same rules applied with more urgency, we will keep losing on tempo, because static rules are what an adaptive agent is built to measure and route around.

The alternative is a shield in two parts. The first is the foundation: know what can act, shrink what it can reach, see at machine speed, act within tight limits, and govern the defenders. The second is new. It targets the attacker's dependencies, binds privilege to purpose, connects defenders with model providers, and deploys a defensive swarm that hunts inside our walls and is designed not to be talked out of its job.

Most of this can start this month, with an inventory, a decoy, a uniform error page and a tested kill switch. The rest is engineering we should begin now, while we still have the warning time.

- Chris LaCour

References

Dates are 2026 unless noted; links accessed November 22, 2026. Research results come from controlled settings.

[1] S. Altman, Stanford GSB View From The Top, Oct 21. https://www.youtube.com/watch?v=anAmTGcewX8

[2] Palisade Research, "LLM Agent Honeypot," arXiv 2410.13919 (Oct 2024; rev. Jan 2025). https://arxiv.org/abs/2410.13919

[3] FBI, FLASH-20261113-001 (ARTEX), Nov 13. https://www.ic3.gov/CSA/2026/261113.pdf

[4] Google GTIG / Mandiant, "ARTEX and Friends," Oct 30. https://cloud.google.com/blog/topics/threat-intelligence/artex-and-friends-enter-agentic-offensive-frameworks

[5] Unit 42, "An AI-Assisted Cyber Attack," Sep 2 (clarified Sep 3). https://unit42.paloaltonetworks.com/ai-assisted-cyber-attack-inside-a-unit-42-investigation/

[6] Microsoft Security, Storm-0940 and ARTEX, Oct 21. https://www.microsoft.com/en-us/security/blog/2026/10/21/storm-0940-uses-artex-agentic-ai-connectwise-supply-chain-compromise/

[7] CrowdStrike, "VAPOR PANDA Uses ARTEX," Oct 29. https://www.crowdstrike.com/en-us/blog/vapor-panda-uses-artex-to-scale-global-campaign/

[8] CERT-In, Advisory CIAD-2026-0058, Nov 4. https://www.cert-in.org.in/s2cMainServlet?VLCODE=CIAD-2026-0058&pageid=PUBVLNOTES02

[9] Acronis TRU, "BOSSAI," Nov 4. https://www.acronis.com/en/tru/posts/bossai-inside-the-ai-attack-mesh-that-targeted-taiwans-emergency-services/

[10] Microsoft Security, "ARTEX abuse expands...financial sector," Nov 6. https://www.microsoft.com/en-us/security/blog/2026/11/06/artex-abuse-expands-as-cybercriminals-target-financial-sector/

[11] AWS Security Blog, Amazon Threat Intelligence on ARTEX agents, Nov 13. https://aws.amazon.com/blogs/security/amazon-threat-intelligence-observes-threat-actors-running-cybercrime-campaigns-with-artex-ai-agents/

[12] Google GTIG, "Agentic Threat Activity Crosses a Significant Threshold," Nov 16. https://cloud.google.com/blog/topics/threat-intelligence/agentic-threat-activity-crosses-a-significant-threshold

[13] Barracuda Trust Center, XDR integration credentials advisory, Nov 10 (upd. Nov 13). https://trust.barracuda.com/security/barracuda-xdr-integration-credentials

[14] OpenAI, "The Hugging Face incident...," Sep 9. https://openai.com/hugging-face-incident-and-misalignment/

[15] OpenAI, "How we detect frontier model misalignment," Oct 29. https://openai.com/index/how-we-detect-frontier-model-misalignment/

[16] OpenAI Alignment, "Suppressing misalignment monitoring alerts...," Oct 20. https://alignment.openai.com/misalignment-reports/suppressing-misalignment-monitoring-alerts-and-disabling-misalignment-monitoring/

[17] Anthropic, "Investigating unintended model actions," Oct 9. https://www.anthropic.com/research/investigating-unintended-model-actions

[18] WorkOS, security bulletin, Oct 30 (upd. Nov 6). https://workos.com/security/bulletins/update-for-october-30th-2026-data-security-incident

[19] Reuters, AI scam at Intesa (Fideuram), Sep 25. https://www.reuters.com/legal/government/ai-messaging-scam-costs-italys-top-bank-intesa-millions-sources-say-2026-09-25/

[20] CNN, Michigan voice-cloning scam, Sep 15. https://www.cnn.com/2026/09/15/us/homebuyers-voice-cloning-scam

[21] KnowBe4, threat trend data, Oct 29. https://blog.knowbe4.com/new-threat-trend-data-when-bad-actors-use-reality-against-us

[22] Denver Health Medical Plan, SquareML incident release, Oct 2. https://www.denverhealthmedicalplan.org/sites/default/files/resources/document/Denver%20Health_News%20Release_DHMP%20SquareML%20Security%20Incident_FINAL%2010.2.26.pdf

[23] Reuters, JPMorgan third-party vendors, Nov 19. https://www.reuters.com/legal/government/jpmorgan-says-third-party-vendors-targeted-cyberattack-2026-11-19/

[24] Maersk, "IT security incident," Nov 11. https://www.maersk.com/news/articles/2026/11/11/it-security-incident

[25] Australian Federal Police, AFP-FBI-WAPF disruption, Aug 27. https://www.afp.gov.au/news-centre/media-release/two-wa-men-charged-following-afp-fbi-wapf-disruption-alleged-global

[26] Defense One, Pentagon personnel data breach, Oct. https://www.defenseone.com/threats/2026/10/pentagon-personnel-data-breach-went-undetected-months/416421/

[27] Boston Scientific, Form 8-K (Aug 25 incident). https://www.sec.gov/Archives/edgar/data/885725/000088572526000059/bsx-20260907.htm

[28] IDC Frontier, IDCF Cloud incident update 7, Oct 27. https://www.idcf.jp/pdf/topics/20261027/20261027_IDCF_UA_Update7_EN.pdf

[29] Ahold Delhaize, Q3 2026 results, Nov 4. https://newsroom.aholddelhaize.com/ahold-delhaize-reports-resilient-q3-2026-results-and-updates-2026-guidance-to-reflect-business-disruption-caused-by-recent-cybersecurity-incident/

[30] KUOW, Microsoft outage not caused by AI or cyberattack, Nov 13. https://www.kuow.org/technology/2026-11-13/microsoft-says-outage-not-caused-by-ai-or-cyber-attack

[31] FAA, nationwide ground stop Q&A, Nov. https://www.faa.gov/newsroom/nationwide-ground-stop-questions-and-answers

[32] CISA, four KEV additions, Nov 20. https://www.cisa.gov/news-events/alerts/2026/11/20/cisa-adds-four-known-exploited-vulnerabilities-catalog

[33] UK NCSC, "Managing the cyber risk of agentic AI," Aug 20. https://www.ncsc.gov.uk/blogs/managing-the-cyber-risk-of-agentic-ai

[34] NIST, NSA, CISA, incident readiness practices, Oct 16. https://www.nist.gov/news-events/news/2026/10/key-practices-incident-readiness-and-response-super-intelligence-training

[35] Executive Order 14434, signed Sep 29 (Fed. Reg. Oct 2). https://www.federalregister.gov/documents/2026/10/02/2026-20321/inaugurating-the-era-of-super-intelligence

[36] OCC Bulletin 2026-54, Nov 4. https://www.occ.treas.gov/news-issuances/bulletins/2026/bulletin-2026-54.html

[37] SEC Division of Examinations, FY2027 priorities, Oct 23. https://www.sec.gov/files/2027-exam-priorities.pdf

[38] S.3315, Health Care Cybersecurity and Resiliency Act of 2026 (passed Senate Sep 30). https://www.congress.gov/bill/119th-congress/senate-bill/3315/all-info

[39] HHS, HIPAA Privacy Rule final rule, Nov 16. https://www.hhs.gov/press-room/hhs-issues-new-final-rule-modernize-hipaa-privacy-rule-support-coordinated-care-individual-engagement-reduce-regulatory-burdens.html

[40] Reginfo.gov, RIN 0945-AA22, HIPAA Security Rule. https://www.reginfo.gov/public/do/eAgendaViewRule?pubId=202510&RIN=0945-AA22

[41] Regulation (EU) 2026/1744, Jul 24. https://eur-lex.europa.eu/eli/reg/2026/1744/oj/eng

[42] CISA, CIRCIA rulemaking. https://www.cisa.gov/topics/cyber-threats-and-advisories/information-sharing/cyber-incident-reporting-critical-infrastructure-act-2022-circia

[43] NIST IR 8596, Cyber AI Profile (draft), Dec 16, 2025. https://csrc.nist.gov/pubs/ir/8596/iprd

[44] CRS, IF12959, CISA 2015 expiring provisions, upd. Nov 2. https://www.congress.gov/crs-product/IF12959

[45] Verizon, 2025 DBIR, Apr 23, 2025. https://www.verizon.com/about/news/2025-data-breach-investigations-report

[46] Google GTIG, IPIDEA disruption, Jan 28. https://cloud.google.com/blog/topics/threat-intelligence/disrupting-largest-residential-proxy-network/

[47] RH-ISAC, residential proxies and account takeover, Jul 14. https://rhisac.org/general-blog/residential-proxy-networks-are-enabling-account-takeover-that-perimeter-controls-cannot-see/

[48] DataDome (vendor-reported), fintech credential stuffing, Jun 3, 2025. https://datadome.co/threat-research/credential-stuffing-fintech-login-attack/

[49] OWASP, Top 10 for Business Logic Abuse, May 30, 2025. https://owasp.github.io/www-project-top-10-for-business-logic-abuse/

[50] "WAFFLED," ACSAC, Dec 2025. https://arxiv.org/abs/2503.10846

[51] "XploitSQL," ACM CIKM, Oct 21, 2024. https://dl.acm.org/doi/10.1145/3627673.3680102

[52] "The Attacker Moves Second," arXiv 2510.09023, Oct 10, 2025. https://arxiv.org/abs/2510.09023

[53] D. Pasquini, E. Kornaropoulos, G. Ateniese, "Hacking Back the AI-Hacker" (Mantis), arXiv 2410.20911, Oct 2024. https://arxiv.org/abs/2410.20911

[54] Y. Wang, J. Yu, Y. Sun, "Stateful Feedback Deception against Autonomous Penetration Testing Agents" (AgentTrap), arXiv 2610.02869, Oct 2. https://arxiv.org/abs/2610.02869

[55] Cloudflare, "AI Labyrinth," Mar 19, 2025. https://blog.cloudflare.com/ai-labyrinth

[56] Acalvio, ShadowPlex deception vs. ARTEX (vendor page). https://www.acalvio.com/agentic-ai-security/defending-against-weaponization-of-artex-shadowplex-360-deception/

[57] Ars Technica, AI tarpits (Nepenthes), Jan 28, 2025. https://arstechnica.com/tech-policy/2025/01/ai-haters-build-tarpits-to-trap-and-trick-ai-scrapers-that-ignore-robots-txt/

[58] "Throttling Web Agents Using Reasoning Gates," arXiv 2509.01619, Sep 2025. https://arxiv.org/abs/2509.01619

[59] ATLAS account-takeover graph study, arXiv 2509.20339, Sep 24, 2025. https://arxiv.org/abs/2509.20339

[60] "Known By Their Actions," arXiv 2605.14786, May. https://arxiv.org/abs/2605.14786

[61] IETF RFC 9421, HTTP Message Signatures, Feb 2024. https://www.rfc-editor.org/rfc/rfc9421.html

[62] IETF Web Bot Auth WG draft (adopted Sep 1; rev. Oct 29). https://datatracker.ietf.org/doc/draft-ietf-webbotauth-httpsig-protocol/

[63] IETF OAuth WG, Transaction Tokens draft-11, Jul 30. https://datatracker.ietf.org/doc/draft-ietf-oauth-transaction-tokens/

[64] OpenID Foundation, AuthZEN Authorization API 1.0 (final, Jan). https://openid.net/wg/authzen/

[65] Visa, "Visa Introduces Trusted Agent Protocol," Oct 14, 2025.

https://usa.visa.com/about-visa/newsroom/press-releases.releaseId.21716.html

[66] Mobile ID World, Mastercard and Google Verifiable Intent, Mar. https://mobileidworld.com/mastercard-and-google-put-passkeys-at-the-heart-of-ai-payments/

[67] ExecutiveGov, NIST NCCoE agent identity concept paper, Feb. https://www.executivegov.com/articles/nist-nccoe-concept-paper-ai-agent-access-controls

[68] Anthropic, AI-orchestrated espionage disruption, Nov 13, 2025. https://www.anthropic.com/news/disrupting-AI-espionage

[69] OpenAI, "Disrupting malicious uses of AI" (Oct 7, 2025 report). https://openai.com/index/disrupting-malicious-uses-of-ai/

[70] CISA, Automated Indicator Sharing, upd. Nov 1. https://www.cisa.gov/topics/cyber-threats-and-advisories/information-sharing/automated-indicator-sharing-ais

[71] Trail of Bits, Buttercup 2nd place in AIxCC, Aug 9, 2025. https://blog.trailofbits.com/2025/08/09/trail-of-bits-buttercup-wins-2nd-place-in-aixcc-challenge/

[72] DARPA, CASTLE program. https://www.darpa.mil/research/programs/cyber-agents-for-security-testing-and-learning-environments

[73] UK NCSC, "Cyber Shield," Jul 7. https://www.ncsc.gov.uk/blogs/cyber-shield-the-path-to-an-agentic-ai-future-for-cyber-defence

[74] Google Cloud, SecOps Response Agent docs, upd. Nov 20. https://docs.cloud.google.com/chronicle/docs/secops/response-agent

[75] CrowdStrike, Charlotte AI platform page. https://www.crowdstrike.com/en-us/platform/charlotte-ai/

[76] Microsoft Security, Microsoft Perception, Nov 17. https://www.microsoft.com/en-us/security/blog/2026/11/17/security-for-the-agentic-era-powered-by-microsoft-perception/

[77] 18 U.S.C. § 1030 (CFAA). https://www.law.cornell.edu/uscode/text/18/1030

[78] H.R. 3270 (116th), Active Cyber Defense Certainty Act (not enacted). https://www.congress.gov/bill/116th-congress/house-bill/3270

[79] H.R. 4988 (119th), Scam Farms Marque and Reprisal Authorization Act (introduced Aug 15, 2025). https://www.congress.gov/bill/119th-congress/house-bill/4988

[80] Mayer Brown, on the Aug 12 presidential memorandum, Aug. https://www.mayerbrown.com/en/insights/publications/2026/08/presidential-memorandum-authorizes-vetted-private-companies-to-conduct-offensive-cyber-operations-against-foreign-criminal-organizations

[81] Wiley Rein, on the memorandum, Aug. https://www.wiley.law/alert-Navigating-the-New-Presidential-Memorandum-on-Transnational-Cyber-Enabled-Crime

[82] D. Lee, M. Tiwari, "Prompt Infection," arXiv 2410.07283, Oct 2024. https://arxiv.org/abs/2410.07283

[83] "Do as We Do, Not as You Think," ICLR 2025. https://arxiv.org/abs/2501.13381

[84] "Poisoning the Watchtower," arXiv 2605.24421, May. https://arxiv.org/abs/2605.24421

[85] OpenAI, "A thousand fewer paths to root," Nov 4. https://openai.com/index/a-thousand-fewer-paths-to-root/

[86] CISA et al., Careful Adoption of Agentic AI Services, May 1. https://www.cisa.gov/news-events/news/cisa-us-and-international-partners-release-guide-secure-adoption-agentic-ai

[87] MITRE ATLAS v2026.10, Oct 28. https://github.com/mitre-atlas/atlas-data/releases/tag/v2026.10

[88] OWASP, Top 10 for Agentic Applications for 2026, Dec 9, 2025. https://genai.owasp.org/resource/owasp-top-10-for-agentic-applications-for-2026/

[89] S. Altman, Forward Future interview, Oct 21. https://www.youtube.com/watch?v=3gXLmy7DK_4