A CISO FIELD BRIEF

The Day-Zero Normal

A Practical Reprioritization Guide for CISOs Entering the AI Vulnerability Era

Rob Fuller
VP, Information Security, Fortune 500 Company
Chair, Vulnerability Management Research Group

VERSION 2.3
APRIL 2026

Reviewed by:
- Ariel Litvin, Former CISO @ First Quality Enterprises
- Jacqueline Lebo, Director of Risk Advisory @ SAFE
Head of AI Workgroup @ FAIR Institute

Contents

Why I wrote this

I've spent twenty-five years on the offensive side of this business, and I've read most of the AI-and-security briefs of the last eighteen months. They're mostly right about the threat and useless about what to do on Monday morning. “Improve asset management” and “adopt Zero Trust” insults a community that hasn't been idle since 2015. We know. The question is what moves up, what moves down, what gets killed, and where the money goes.

This document is my answer, written for peers in the CISO seat. It uses NIST CSF 2.0 as the skeleton because that's how we structure programs when we talk to our boards, and because GOVERN is the piece that matters most in an AI-accelerated world. I'll tell you which of the standard Fortune 500 security projects need to change, how, and what to do instead.

Mythos isn't the starting gun. Opus 4.6, XBOW, Raptor, and AIxCC were already doing this work, and the receipts are in the public CVE database: as of mid-April 2026, VulnCheck counted 40 CVEs credited to Anthropic-affiliated researchers using Claude, including 28 Firefox vulnerabilities rated 8.8 or higher in two disclosure drops. Mythos made the trend legible to boards and mainstream press. That's the window we have documented.

(Source: https://www.vulncheck.com/blog/anthropic-glasswing-cves)

The operating reality in one paragraph

For a growing class of high-value vulnerabilities, attackers now generate working exploits in hours to days. Patch development, QA, and deployment still take weeks. Your program lives in that gap. That means segmentation that holds, identity boundaries that enforce, detections that fire **before** EDR, agents that remediate without a human in the loop for the boring 80%, and an incident response muscle that treats a breach as a weather event, not a surprise..

> *Stop trying to outrun the rain with faster patching;* ***build a roof so that when it pours, the business stays dry****.*

What moves up / moves down

The executive summary in table form. If you're reading this on a plane and only have five minutes, this is the page to screenshot.

Legend: ▽- Divest / ▼- ▲- Reprioritize / ● - New

PROGRAM / INVESTMENT	PRE-AI PRIORITY	Δ	2026 PRIORITY
Annual third-party pentest	HIGH (board-visible)	▽	LOW divest, replace with continuous CART (Continuous Automated Red Teaming)
Signature-based AV / legacy IDS	MEDIUM HIGH	▼	LOW EDR + behavioral only
SIEM (rules-based)	HIGH	▼	MEDIUM keep the log lake, kill the rules engine
SAST/DAST alert queues	HIGH	▽	LOW replace with LLM triage in PR flow
Asset inventory / CMDB	MEDIUM (checkbox)	▲	CRITICAL runtime-validated, identity-joined
Phishing-resistant MFA rollout	MEDIUM	▲	CRITICAL 100% for privileged, no exceptions
Network / identity segmentation	MEDIUM	▲	CRITICAL the thing that saves you post-breach
Detection engineering (earlier kill chain)	LOW MEDIUM	▲	CRITICAL move left of EDR
Traditional VM (scan, SLA, patch)	HIGH	▼	MEDIUM restructured around exploitability-in-context, not CVSS
VulnOps / LLM-driven code review	*Did not exist*	●	CRITICAL **new team, new budget line**
Agent identity & governance (NHI)	LOW	▲	CRITICAL agents are now privileged users
TPRM questionnaires	HIGH	▽	LOW replace with continuous posture API
Annual SAT / compliance training	MEDIUM	▽	LOW replace with just-in-time coaching
Tabletop / IR exercises	MEDIUM (annual)	▲	HIGH quarterly, agent-compromise scenarios
Disposable / rebuildable infra	LOW	▲	HIGH recovery speed is the new MTTR

A NOTE ON REGULATED INDUSTRIES AND AUDITORS

Several of the 'divest' recommendations above run headlong into compliance requirements. PCI DSS 11.3 requires an annual penetration test. HIPAA, SOX, NYDFS, and DORA each have control expectations that presume a specific form of an activity the table marks as low-priority. Read this paragraph twice before you decommission anything.

Auditors have more latitude than most programs use. If a mandated control is being replaced by one that achieves the same outcome better, that is a conversation you can have with your auditor. In my experience, most good auditors welcome it. They are tired of watching programs waste money on controls that stopped working in 2018. Walk in with evidence: what the old control caught, what the new control catches, why the new one is a superset, and how you will prove it over time. Most auditors will meet you halfway.

If the regulation is rigid and the auditor is not, there are legal ways to satisfy the letter of the requirement without letting it eat your budget. Turn your continuous automated red teaming to 'high intensity' for one week a year and call that your annual pentest. Bring in a consultancy to 'pilot' your CART platform and use the findings as your pentest report. Run your VulnOps program all year and produce an annual summary that maps onto the pentest deliverable format. These are higher-assurance versions of the control the regulation was originally trying to mandate, and a reasonable auditor will accept them with the right paperwork. 'Compliance requires the old control' is rarely true when you read the text; **it requires an outcome,** and you have more ways to produce the outcome than you did ten years ago.

Walking the NIST CSF 2.0 functions

CSF 2.0 adds GOVERN as a first-class function alongside Identify, Protect, Detect, Respond, and Recover. The CISO's job has shifted from running controls to governing outcomes across systems the security team doesn't own, and AI-driven attack velocity is what forces that shift.

◆ GOVERN: The function most people are underinvesting in

CSF 2.0 put GOVERN on top of the wheel because the controls are fine; the governance is where programs fall apart. In an AI-accelerated threat environment, governance failure is a velocity failure. You cannot move at the speed this moment requires if every autonomous remediation needs a CAB ticket and three sign-offs.

Safe velocity comes down to two words: **scoped** and **reversible**. When I talk to CIOs and CTOs about machine-speed response, the first thing they hear is “outage,” and they are right to hear it. A fully automated containment system with no guardrails turns one compromised host into an afternoon of incident calls and a bad press release. Every recommendation in this section (the authority matrix, the NHI program, the AI usage policy) lets the program move at threat speed inside boundaries that have been negotiated, documented, and signed off on in advance.

Scoped and reversible is the floor. The ceiling, where programs should be heading over the next 24 months, is pre-validation: simulating the blast radius and side effects of a proposed action before executing it. Almost no program can do this today. The ones that get there first will operate at speeds the rest cannot safely match. The Standing Authority Matrix is the on-ramp; the reachability graph in IDENTIFY is what feeds it.

WHAT MOVES UP

- **Autonomous action authority.** Your CIO will push back on this bullet hardest, so let me address it up front. Nothing in a Standing Authority Matrix authorizes a machine-speed action against a production service without a pre-defined scope and a documented rollback. The actions on the matrix are small and reversible by design: block an IP at the edge for 24 hours, isolate a single endpoint, revoke a single OAuth token, fail a single build, roll back a single deploy. Note the word “single” in every one of those. Every entry has a named approver who owns the decision, an audit cadence, and a rollback path reviewed quarterly [see Appendix A for a sample matrix with rollback paths specified for every entry]. Actions that cannot be safely scoped and rolled back do not go on the matrix; they stay in the human-in-the-loop queue where they belong. What this buys you is the ability to contain the 80% of incidents that are boring and obvious, at the speed the threat now moves, without paging a director at 3 a.m. for permission to block a known-bad IP. Get your GC and CIO to sign the matrix, review it quarterly, and treat it as the most important governance artifact you will produce this year.
- **Non-human identity (NHI) governance.** Agents, service accounts, and workload identities outnumber human users by a wide margin in most enterprises. CyberArk's

2024 Identity Security Threat Landscape report puts the ratio at roughly 45-to-1, and GitGuardian's 2024 State of Secrets Sprawl found similar scale in source control alone. The number varies by industry and measurement method, but every serious survey lands in the same order of magnitude: non-human identities are the majority of your privileged population, and most programs govern them like an afterthought. Stand up an NHI program with a named owner, a lifecycle process (provision, rotate, deprovision), and quarterly access reviews that apply to bots the same way they apply to humans. If you own IAM, this is now half of your job, or more.

- **AI usage policy with teeth.** A specific policy that says which models are approved for which data classifications, which agents have production write access, what logging is required, and who is liable when an agent takes a bad action. Generic "don't put PHI in ChatGPT" memos do none of this work. Liability is the question most programs are avoiding, and it's the one that will bite first. A strategic note while you are at it: the AI control frameworks that will be mandatory in three years are being shaped now by what auditors and standards bodies see working at the programs that moved early. Write the policy that defines the standard, or rewrite your program later to fit someone else's. The first is cheaper, and it puts you in the room when the language gets finalized.
  - One operational consequence worth naming: a standing policy with clear data-class-to-model mappings removes the need for an AI governance committee to vote on every new use case. That committee is the velocity bottleneck most programs are about to discover the hard way. A policy with teeth is what replaces it; if you have both, you have neither.

45 : 1

NON-HUMAN IDENTITY RATIO

Non-human identities outnumber human users by roughly 45 to 1 in the typical enterprise (CyberArk, 2024). Most programs still govern them like an afterthought.

WHAT MOVES DOWN

- **Annual risk assessments as the primary governance artifact.** A point-in-time risk register updated once a year is a museum piece. Replace it with a continuously-updated risk view fed from your control telemetry. If your GRC team is spending more than 20% of its cycles on the annual refresh, reallocate those cycles to control validation. (Control validation means continuously testing that the control actually fires on its trigger, breach-and-attack simulation against the segmentation policy, automated reachability tests against IAM, fire drills against the SOC playbook. *Self-attestation in a spreadsheet does not count.*)
- **Committee-driven exception processes.** Every security exception that takes more than 48 hours to approve is a vulnerability you created yourself. Delegate exception authority down and audit the decisions quarterly, rather than gatekeeping them up front.

(Source: https://www.cyberark.com/resources/blog/why-machine-identities-are-essential-strands-in-your-zero-trust-strategy)

IDENTIFY: Rebuilding asset management

Every brief says “improve asset management.” Few say how, because the how is hard and unglamorous. In 2026, given what AI changes, here is the approach.

The traditional approach: CMDB plus quarterly discovery scans plus a spreadsheet of crown jewels.

This fails for three reasons now. First, the refresh rate is wrong by three orders of magnitude: your attackers are enumerating your estate in minutes with LLM-driven recon, and you're refreshing the CMDB weekly at best. Second, it's not joined to identity, so you can't answer “what can this compromised token reach?” in under an hour. Third, it doesn't include the things that matter more every day: agents, MCP servers, model endpoints, and the service accounts that wire them together.

REFRESH RATE	IDENTITY JOIN	AI-LAYER ASSETS
Attackers enumerate in minutes. Weekly CMDB refreshes miss three orders of magnitude of what an adversary already knows.	“What can this token reach?” is the incident-hour question. If assets and IAM live in separate tools, you cannot answer it.	Agents, MCP servers, model endpoints, and the service accounts wiring them together. None of it is in your CMDB.

THE PRACTICAL REBUILD

- **Make the runtime the source of truth.** Your EDR, your cloud APIs (AWS Config, Azure Resource Graph, GCP Asset Inventory), your Kubernetes control plane, and your identity provider already know what exists. Stand up a CAASM (Cyber Asset Attack Surface Management) layer (Axonius, runZero, JupiterOne, or a home-built graph on Neo4j if you have the engineering) that ingests these feeds and treats the CMDB as one input among many.
- **Join assets to identity on day one.** The graph needs to answer “which identities can reach this asset, and which assets can this identity reach.” This is the query that matters during an incident, and it's the query most programs cannot run. If your asset inventory and your IAM data live in different tools with no join key, fix that before you do anything else in the Identify function.
- **Build toward what-if, not just what-is.** The reachability query answers the incident-hour question. The next one, how reachability changes under a proposed deployment, new route, or new grant, before it ships, is what lets the program move at machine speed safely. Almost no one runs it today. The graph is the foundation; put the query on the roadmap.
- **Inventory the AI layer explicitly.** Add asset classes for: approved LLM endpoints, MCP servers, agent identities, coding assistants with repo access, and model artifacts in registries. Most CAASM products don't do this natively yet; you'll need to extend them. Start with a spreadsheet if you have to, but get it into the graph within two quarters.
- **Measure staleness, not coverage.** Track “median age of the freshest record per asset.” Days are fine. Weeks is a detection gap your adversary will exploit.

BUDGET

A CAASM deployment at Fortune 500 scale runs **$400K–$1.2M/year** in tooling plus two FTEs; smaller enterprises can land closer to $150K with open-source plus one FTE. That is less than most organizations spend on their annual pentest retainer.

EXTERNAL ATTACK SURFACE MANAGEMENT: YOUR REAL CMDB

Your CMDB knows what procurement bought. Your CAASM knows what your internal tools have seen. Your EASM knows what the internet can see. At mid-size companies the two lists overlap. At Fortune 500 scale they diverge by thirty percent or more: acquired subsidiaries nobody finished integrating, marketing microsites that stand up and tear down faster than asset review cycles, dev environments baked into a vendor's DNS for a demo in 2019. All of it faces the public internet. None of it lives in your CMDB.

This gap is where Mythos-class capability hits hardest. Your internal patch cadence is a policy discussion. Your external patch cadence is a race against a model that enumerates your estate, fingerprints every version banner, and probes for known vulns in the time it takes you to read this paragraph. 'We roll patches on the quarterly cycle for internet-facing systems' is done. If it is public, it is patched now.

One heuristic to hand your VM team: **point of view matters more than CVSS score.** An external scanner flagging a Medium on a public asset is elevated work; an attacker sees the same thing and can reach it from their laptop. A host-based agent flagging the same Medium on a second-order vuln (not exploitable until something else gets a foothold) is defense-in-depth, not a 3 a.m. page. Triage by where the finding came from, then by the score.

Mythos also changes chain analysis. Chris Gates built a decade of Red Team talks on the 'Low2Pwned' thesis: chain five info disclosures, an IDOR, and a permissive CORS policy and you have pre-auth RCE. Red teams did that in weeks. Mythos-class models will do it in minutes, running against your whole attack surface at once. The EASM products that matter in 2027 are the ones that feed findings into a model and rank by chain-exploitability. Hadrian and watchTowr are closest to that model today. Palo Alto Xpanse, CrowdStrike Falcon Surface, Tenable ASM, Randori (IBM), Rapid7 Surface Command, IONIX, and Assetnote (Searchlight Cyber) cover the breadth. Pick the one on a chain-reasoning roadmap. The CVSS-list vendors are selling you last year's category.

■ PROTECT: Where the reprioritization is sharpest

NO MORE LEGACY EXCEPTIONS

Every security program carries a list of systems that have been "out of scope" for a decade. The app the vendor stopped supporting in 2014. The PLC firmware from a company that was acquired, spun off, and dissolved. The ERP module nobody will touch because the one engineer who understood it retired. The medical device running an OS that went end-of-life before the current CISO was hired. The standard answer has been: segment it, monitor it, pray. That answer was always a compromise. It is now a liability.

The reason the excuse held for twenty years is that the economics of fixing it were brutal. Reverse-engineering a binary, reconstructing lost source, porting a protocol stack, rebuilding a driver for a modern kernel, each was a multi-quarter project that no board would fund against a system that "still works." AI changes that math. A model can read the firmware, rebuild a readable specification, generate a patched binary, and produce the test harness that proves it behaves the same on the inputs that matter. What used to be a capital project is now a sprint.

The shift that follows is uncomfortable and correct: **if the vendor cannot or will not fix the system, the operator has to, and now the operator can**. The "vendor went out of business" line stops being a shield. Three practical moves:

- **Treat orphaned systems as AppSec scope, not infrastructure scope**. If the code or firmware is running in production and nobody is patching it, your VulnOps function owns it. That reframe alone changes the budget conversation.
- **Commission an AI-assisted reverse-engineering and hardening pass on the top ten orphaned systems in your estate**. Start with the ones facing the internet or sitting on the IT/OT boundary. The output is a current specification, a threat model, and a prioritized fix list. The cost is one engineer-quarter plus compute; the alternative is waiting for the breach.
- **Stop accepting "the regulator requires us to keep it" as a terminal answer**. Regulators require outcomes, not artifacts. A rebuilt, signed, modern replacement that passes the same validation is better evidence than the original ever was. Walk it in the same way you walk in the pentest-replacement conversation earlier in this brief.

This applies hardest where the industry has accepted stagnation longest: OT and ICS, medical devices, industrial process control, legacy ERP, custom trading systems, airline operations. Those are the exact places where the next wave of AI-discovered vulnerabilities will land. The programs that start rebuilding now will have options. The programs that keep writing "compensating controls" memos will be watching their compensating controls fail in real time.

The era of "it's always been that way" is ending, and it is ending because the thing that made it that way, the cost of fixing it, is collapsing. Defend the program that uses that collapse. Do not defend the tech debt.

PHISHING-RESISTANT MFA: NO LONGER OPTIONAL, NO LONGER PHASED

Most programs have an MFA rollout in some state of partial completion. In 2026, partial is failure. AI-driven AiTM (adversary-in-the-middle) phishing kits and deepfake voice attacks have made SMS and push MFA dangerous. The move is FIDO2/WebAuthn hardware keys or platform passkeys for 100% of privileged human accounts, and workload identity federation for 100% of non-human privileged access. No exceptions for executives or legacy systems. If a legacy system can't support it, move the system behind a jump host that can.

> **BUDGET**
>
> **$50–$150 per privileged user** for hardware keys is a rounding error against the cost of one AiTM incident.

SEGMENTATION: THE CONTROL THAT SURVIVES A ZERO-DAY

The most underreported finding in the Mythos preview is that **Linux kernel defense-in-depth held**. The model found exploitable primitives but couldn't chain them into a working remote exploit because the kernel's internal boundaries got in the way. Segmentation is what converts "exploitable" into "unexploitable in your environment."

- **East-west segmentation in the data center.** If you're still running flat VLANs in your production DC, this is your top capital project for the next 18 months. Illumio, Akamai Guardicore, Cisco Secure Workload, or native cloud primitives: pick one and commit. Base the policy on application identity, **not IP ACLs**. I've seen this go badly when teams cut the corner, blame the tool, and rip it out.
- **Identity segmentation.** Tier-0/Tier-1/Tier-2 admin separation is 2015 guidance that most environments still haven't implemented. Get Privileged Access Workstations for anyone touching domain controllers, cloud root, or the CI/CD control plane, with no email, no browsing, no exceptions.
- **Egress filtering.** Every workload should have a deny-by-default outbound policy. This one control breaks most commodity ransomware and most data exfiltration. It will also break things on day one. Start in one segment, log what breaks, build the allowlist from the logs. The first month is painful and the alternative is funding someone else's ransom payment in eighteen months.

OT AND ICS: THE SEGMENT-PLUS GAP

Everything above about segmentation, identity boundaries, and runtime truth applies to your IT estate. For OT and ICS, the playbook is thinner and the stakes are higher.

The short list of what you can do now: deep segmentation with protocol-aware filtering at the IT/OT boundary; continuous monitoring of anything crossing that boundary (Claroty, Dragos, Nozomi: pick one, deploy, tune); jump hosts with phishing-resistant MFA for any engineering access; pressure on your PLC, HMI, and medical device suppliers to ship signed firmware and coordinated disclosure; and an inventory of which systems are built by vendors that still exist, which are built by vendors that don't, and which nobody knows because the plant engineer who installed it retired in 2014. That inventory alone is a six-month project at most large enterprises.

That's Monday morning. The larger problem is this: if the industry does not think differently about OT, this is where the AI vulnerability cataclysm stops being a rhetorical device and becomes real. The story the press has not yet written is the one where Mythos-class capabilities get pointed at firmware from a vendor that went out of business in 2011, running on a PLC that controls something that matters, and the patch does not exist and cannot exist. Segmentation buys time, not forever.

Fortinet is the shape of what is coming next. Two years of pre-auth RCE and auth-bypass in FortiOS fits the signature of AI-assisted firmware analysis at scale: binaries available behind a free support account, steady cadence across releases. Consumer IoT is the next rung, because vendor support pages host that firmware as a public download, and the first Mirai-shaped botnet of AI-discovered bugs is an eighteen-month problem. OT is the rung after that. PLC and HMI firmware stays off public download pages, and that friction has been the protection. One leaked sample from a pentest report, an FCC filing, or a compromised supplier ends the protection for an entire product line. The compensating controls you write for OT are stopgaps with an expiration date. Size them for the leak timeline, not for forever.

Segmentation should not be the only option for OT. Until the market catches up, segmentation plus honest inventory plus vendor pressure is what you have. Do all three, and lean on your peers to push the next layer into existence.

APPSEC AND VULNOPS: THE MERGE

Your AppSec program probably has SAST, DAST, SCA, a bug bounty, and a secure SDLC policy nobody reads. The SAST and DAST queues are where analyst hours go to die: thousands of unverified findings, most of them noise. The rebuild:

- **Put an LLM in the PR path.** Claude Code Security, OpenAI's Codex Security, Semgrep Assistant, or a home-built pipeline using Anthropic's API against your internal coding standards. This is not theoretical at scale: OpenAI reports Codex Security has contributed to over 3,000 critical and high-severity fixes across 1,000+ open-source projects since launch. The agent reviews every PR for security defects before human review, comments inline, and blocks merges on high-confidence findings. Target: 80% reduction in human hours spent on SAST triage within two quarters.
- **Get on the verified-access paperwork now**. OpenAI's Trusted Access for Cyber (expanded April 14, 2026 to thousands of verified defenders and hundreds of enterprise teams) and Anthropic's Glasswing consortium are the routes to models with relaxed refusals on legitimate security work: binary reverse engineering, exploit reasoning, malware analysis. Standard commercial models will refuse half your dual-use queries by policy, and that refusal is why your AppSec team keeps telling you the tools "don't work for security." They work; you just don't have the right tier. Verification takes weeks and involves your GC and procurement teams. Start the paperwork this quarter so the access is in place when you need it, not after.

  (Source: https://openai.com/index/scaling-trusted-access-for-cyber-defense/)
- **Stand up VulnOps as a named team.** One security engineer with LLM tooling experience, one developer with AppSec background, direct reporting line to the CISO (not buried under AppSec), and a mandate to own the discover-prioritize-patch-validate loop end to end. Open-source starting toolkit: Semgrep, CodeQL, Raptor, OpenAnt. This team replaces about 30% of what your current AppSec program does and makes the rest work.

BUDGET

VulnOps first year: **two FTEs plus $100K–$200K** in tooling and compute for a mid-to-large enterprise; smaller orgs can pilot with one FTE and open-source only.

- **Dependency firewall.** Sonatype Nexus Firewall, JFrog Xray, or Socket.dev in blocking mode. No package enters your build environment without provenance validation. This control would have stopped most npm supply chain attacks of the last two years, and it matters more each quarter.
- **Shadow code from citizen developers.** Developers no longer write all the production code in your company. A product manager with a Claude Code license ships a feature in an afternoon. A finance team builds a reconciliation agent that writes to the general ledger. An ops lead stands up a webhook service that never touches the engineering org chart. I know of one engineering manager who rejected a PR from their own team so their vibe-coded feature could ship first, because they did not want to wait for review. Your AppSec program's SDLC gates (PR review, SAST in CI, dependency firewall, human eyes on diff) never touch code that entered the org around the outside. The provenance gap is the real problem. M&A due diligence used to ask “does anyone have the source code for this app.” The 2027 version: “does anyone remember the prompt that generated this feature, or the model version that generated it?” Treat citizen-developer output as a supply-chain category with a lifecycle: who generated it, when, against what requirements, with what model, reviewed by whom, committed where. Your platform team can offer a sanctioned path (approved IDEs, approved models, required checks) that is easier than the shadow path. Do that, or the tech debt compounds faster than any refactor budget.

VDP AND THE BUG BOUNTY ECONOMY

HackerOne paused new vulnerability submissions to its Internet Bug Bounty (IBB) program on March 27, 2026. The IBB is the oldest crowdsourced open-source VRP, launched in 2013, and fundamental to how open-source maintainers receive coordinated disclosure. HackerOne's stated reason for the pause: the gap between AI-assisted discovery and the ability of open-source maintainers to ship remediations had become impossible to bridge. Daniel Stenberg, the curl maintainer, had been describing the same math from the project level for two years: AI-generated "slop" reports drowning triage, a small set of maintainers unable to keep up. When a platform the size of HackerOne suspends the IBB, it is the canary.

The economy that emerges on the other side of Mythos looks different in four ways. Platforms integrate Mythos-class scanners into the program itself and charge you for it at renewal. Payouts move from proof-of-vulnerability to proof-of-exploitation; the bar for "this is a real finding" rises. Researcher value moves up the stack to business logic, authentication flows, multi-step chains. And the low-hanging fruit economy that paid a generation of researchers rent while they learned goes away.

The last point is the one nobody in the industry is talking about. Low-severity bugs at $50 to $500 each were how the next generation of researchers learned the craft. When a model eats that class of bug, the training ground goes with it. The industry either funds a replacement (apprenticeship tracks, live-fire training programs, dedicated beginner-track bounties) or runs out of senior researchers in a decade. Push your platform on this at your next renewal. BugCrowd and HackerOne will not volunteer it.

For your own program, two practical moves. **One:** assume AI-assisted submissions are the majority and require proof-of-exploitation artifacts (working PoC, payload, session capture) before you pay. **Two:** renegotiate your platform contract to include an AI-augmented continuous scan of your program scope as a standard line item. The platforms will price it as a premium. It should be the baseline.

SUPPLY CHAIN, FOURTH PARTY, AND VENDOR AI USAGE

The fourth-party problem is about to get worse, and dependency firewalls alone do not cover it.

- **SBOMs move from paperwork to an operational artifact.** Most SBOM programs today are a compliance deliverable that sits in a SharePoint folder. Your SBOM needs to be queryable, current, and wired into your asset graph so that when a Glasswing-style advisory drops against a shared library, you can answer 'where am I exposed' in minutes rather than days. Anchore, Chainguard, and Snyk are playing in this space; pick one that integrates with your build pipeline rather than one that generates PDFs.
- **Fourth-party risk is where the next breach lives.** Your tier-1 vendors have their own vendors, and those sub-vendors are often AI-native companies running agents against data you ultimately own. Your TPRM program probably does not touch fourth parties at all. Require tier-1 vendors to disclose their critical sub-vendors and AI usage in specific data-handling contexts, and build an inventory. You cannot govern what you cannot see.
- **Vendor AI usage as a new risk category.** When your SaaS vendor adds an AI feature, a new set of questions opens: what model, hosted where, training on your data yes or no, logging retained for how long, what happens to your data if the feature is deprecated. Most TPRM questionnaires do not ask any of these. Update yours and re-send but think in if/then. If the vendor handles your data or runs revenue-critical infrastructure, send. If they produce marketing videos, skip. If “is our data in your training set?” comes back yes, what is your next move? Decide before you send. Asking the question without a planned response creates discoverable risk you didn't have before.
- **Agentic supply chain: MCP servers, plugins, skills.** Browser extensions are the obvious precedent. A marketplace opened, a decade of wild west followed (extensions reading every page you visited, exfiltrating credentials, injecting ads into your banking site), and enterprise-grade controls like Chrome for Enterprise and managed extension policies landed around 2020. Your MCP/plugin/skill ecosystem is in the marketplace phase right now, on a compressed timeline. Build the hygiene that your package supply chain took twenty years to produce: a registry of approved MCP servers, provenance and signing, scoped permissions enforced at the runtime, an allow-list at the boundary. Any MCP server with shell access or credential access is tier-0 supply chain. Starting points exist today (Anthropic's MCP Inspector, community catalogs); mature commercial options are weeks away.

ENDPOINT REDUCTION: THE CHROMEBOOK LESSON

K-12 schools run some of the most adversarial endpoint environments on the planet. Kids with infinite time and motivation to defeat every control you put in front of them, and yet the K-12 sector has had zero reported successful ransomware attacks on ChromeOS devices. Schools get breached constantly, but the breaches come through email compromise, phishing of staff, the student information system, or a third-party vendor. They almost never come through the endpoint as the entry point. The reason is structural: a Chromebook cannot install third-party software, cannot easily run malware, and resets to a known-good state in minutes. The local attack surface is a small fraction of what a Windows or macOS workstation presents.

Enterprise IT has spent two decades layering EDR, EPP, application control, and host-based firewalls onto endpoints to claw back some of the security posture that the endpoint

architecture itself surrendered. There is another path: surrender less in the first place. Most knowledge workers in most enterprises do not need a fully featured workstation. They need a browser, a productivity suite, a video conferencing client, and access to a handful of SaaS applications. A Chromebook, a locked-down ChromeOS Flex install, or an equivalent thin-client architecture meets that need at a fraction of the endpoint refresh cost, with simpler support.

THE HONEST ACCOUNTING

GIVE UP	GAIN	REPLACES EDR
EDR as you currently use it. Some power-user workflows. Some legacy thick-client applications that should have been retired anyway.	Local attack surface drops by 90%+. Refresh costs drop. Battery life improves. Tier-1 support tickets drop. The endpoint becomes a commodity.	MDM at the device layer, identity and conditional access at the IdP, and network-layer visibility (NAV, egress logging, DNS monitoring). These are your “left of EDR” pieces.

The right model for most enterprises is tiered. Knowledge workers, executives who only do email and meetings, contractors, kiosk users, and most of the long tail of office workers get a Chromebook or equivalent. Engineers, designers, finance power users, and the need-a-workstation population get a workstation, and the trade is transparent: a workstation comes with more capability and more controls, tighter conditional access, more identity friction, more host monitoring, narrower egress policy. Users self-select toward the lighter option more often than you would expect, once they understand what they are signing up for. Put that in internal communications when you launch the program, not on the recruiting page. A two-tier endpoint strategy is honest, defensible, and cheaper to defend than the universal-workstation model most enterprises still default to.

If you take this seriously, the immediate actions are:

1. Inventory your current endpoint population by application requirements, not by job title.
2. Identify the population that needs a workstation and the population that does not.
3. Pilot a Chromebook tier with one business unit.
4. Reallocate the EDR licenses you free up toward identity threat detection and network visibility, where the marginal dollar now buys more security than it does at the endpoint.

This is one of the highest-leverage moves in this document, and most CISOs will not make it, because it requires telling the business that some people are getting a different laptop. That is the pattern, and it is the point.

The Chromebook tier stacks four wins at once: local attack surface collapses, endpoint refresh cost drops, support load drops, and the target population sees little difference day to day. Every enterprise security program has other moves that stack wins the same way. Finding them is the job.

Three that buy the same multi-layer return:

KILL THE CORPORATE VPN	EPHEMERAL DEV ENVIRONMENTS	JIT PRIVILEGED ACCESS
Replace with identity-gated access (Tailscale, Cloudflare Access, Zscaler Private Access, Twingate). Attack surface drops because there is no flat internal network to pivot on. Hardware and license cost drops with the concentrators. Helpdesk load drops because the product connects without a Connect button. The security team loses the lateral-movement highway that underwrites half of the ransomware playbook.	Move engineers onto GitHub Codespaces, Gitpod, Coder, or an internal Kubernetes equivalent. Source code, build secrets, and cloud credentials stop living on laptops. New-hire onboarding goes from a week to half an hour. A lost laptop stops being a supply-chain incident. Engineers gain a consistent environment that eliminates the “works on my machine” class of bugs.	Replace standing admin rights with just-in-time access (Teleport, StrongDM, Okta ASA, native cloud). The privilege a ransomware operator wants does not exist for most of every day. Cloud and license cost drops with the zombie admin accounts. Audit evidence writes itself. On-call engineers get a ten-second approval flow in place of a permission that used to live in their session for months.

Find the ones in your program. Challenge them on the same grounds. The velocity of this threat will not slow down to fit your program's assumptions; speed up the program by breaking them.

▲ DETECT: Move left of EDR

Most detection programs are EDR-centric, which means they fire at exploitation or post-exploitation. When exploitation happens in seconds, that's too late. The reprioritization:

- **Identity-layer detection is now tier one.** Impossible travel, anomalous OAuth grants, new federation trusts, service principal credential additions, dormant account reactivation. These signals fire earlier in the kill chain than anything your EDR will see. If you're not running a dedicated ITDR (Identity Threat Detection and Response) capability like Microsoft Defender for Identity, Falcon Identity Protection, Semperis, or equivalent, that's your next purchase.

> **NOTE ON EDR**
>
> EDR investment stays flat, not down. What changes is what you use it for. Pre-AI, EDR was your primary exploitation and post-exploitation detection layer, and many programs leaned on it to catch things identity and network telemetry should have caught earlier. That “leaning” is what moves. EDR remains critical for behavioral detection on endpoints, agent activity baselining, the crown-jewel servers where you need maximum fidelity, and response actions. It is not where you detect initial access anymore; that job moves left to identity and network. Keep the license count, refocus the use case, and don’t let a vendor talk you into doubling the spend because “AI attacks are harder.” The answer is better telemetry upstream.

- **Canary everything.** Thinkst Canary tokens in source repos, SharePoint, password vaults, S3 buckets, and as fake service accounts. They cost almost nothing and they fire on the exact behaviors AI-driven recon performs. I run them, and I've watched them catch things EDR missed by 40 minutes.
- **Agent behavioral baselines.** For every autonomous agent with production access, establish a behavioral baseline: what repos it touches, what endpoints it calls, what volume of actions per hour is normal. Alert on deviation. The Mythos system card documented a model scrubbing git history to hide its actions. Treat that as a real adversary technique, not a research curiosity.
- **Kill the rules engine, keep the log lake.** Your SIEM's value is the data it holds, not the correlation rules written in 2016. Route the data to a cheaper platform (Snowflake, Databricks, Cribl tiering) and run LLM-assisted investigation on top. Most organizations can cut SIEM spend 40% this way and improve detection at the same time.

► RESPOND: The muscle that needs the most reps

The quality of your incident response in 2026 will be determined by decisions you make and rehearse before the incident. The velocity of AI-assisted attacks does not leave room for real-time deliberation.

- **Pre-staged containment playbooks with pre-approved authority.** For the top 20 incident types, write the containment action, get it pre-approved by GC and the business owner, and wire it into SOAR (Security Orchestration, Automation, and Response). When the alert fires, the action happens. The human is notified, not consulted. Pre-approval is today's bar. Pre-validation, the system simulating blast radius before it executes, is the next one, and the pre-approved authority matrix is how you earn the right to get there.
- **Quarterly tabletops with agent-compromise scenarios.** Add scenarios where the coding agent is the attacker, a compromised MCP server is the pivot, or the IR team is using AI and the adversary is poisoning its context. These scenarios expose the gaps in your current plan.
- **Retain an IR firm with AI-era experience.** Your existing retainer (Mandiant, Unit 42, CrowdStrike, Kroll) is probably fine. Confirm in writing that they have handled incidents involving compromised AI agents, prompt injection at scale, or model supply chain attacks. If they haven't, find a second retainer that has.
- **Know your place in the vendor support queue before you need it.** When the CrowdStrike Falcon update took out 8.5 million Windows machines in July 2024, the companies that got help within the hour were the ones with named technical account managers, premium support contracts, and a pre-existing relationship. Everybody else waited. In a breach, the same logic applies to every critical vendor in your stack: EDR provider, IdP, cloud, backup vendor, the SaaS platforms your business runs on. Ask each of them, in writing, what your response-time SLA is during a declared incident, who your escalation contact is, and where a hundred simultaneous customer incidents would rank you. Most CISOs have never asked. If the answer is “we don’t offer that,” price out the premium tier, or negotiate it into the renewal. This is controllable.

✱ RECOVER: Disposable infrastructure as a security control

Recovery has historically been a BCP/DR function that security teams consulted on. That needs to change. In an environment where you will be breached more often and faster, the speed at which you can rebuild is a security control.

- **Measure and report time-to-rebuild.** For your top 10 critical services, how long does it take to rebuild from source into a clean environment? If the answer is "we've never tested it," that is your answer. Don't guess. Ask your teams. You might be surprised by the answer. Target: under 4 hours for tier-1 services, rehearsed quarterly.
- **Immutable infrastructure where possible.** Services that rebuild from pipeline recover from compromise in minutes. This is a platform engineering investment, but security should fund half of it because security captures most of the value.
- **Backup integrity validation, not just backup existence.** Every backup program has backups. Fewer have tested restores. Almost none have tested restores with integrity validation against a known-good manifest. That is the one that matters.

> *Cyber resilience is the art of making an attacker's biggest win feel like **our team's smallest problem**.*

People: The investment nobody wants to talk about

Your team is watching this happen and wondering if they are about to be automated out of a job. Some will leave if you don't address it. This is what I tell my team:

The work is changing. Tier-1 SOC triage is moving to a model; tier-1 SOC analysts are becoming agent supervisors, detection engineers, and threat hunters. SAST triage is moving to a model; AppSec engineers are becoming the people who configure, tune, and govern those models. GRC analysts who learn to query their control telemetry with an LLM do in a week what used to take a quarter.

WHAT THE WORK WAS		WHAT IT BECOMES
Tier-1 SOC triage	→	Agent supervision and detection engineering
SAST/DAST queue review	→	Model configuration, tuning, and governance
Annual GRC refresh	→	Continuous control telemetry with LLM-assisted query
Tier-1-to-tier-3 ramp	→	Junior program built around agent supervision and red-team drills

CONCRETE INVESTMENTS

- **Every person on the security team gets a coding agent license and an expectation to use it.** Claude Code, Cursor, Codex: pick one, pay for it, measure adoption. Nobody on my team says “I don’t code” anymore. You don’t need to code; you need to describe what you want in English and review what the agent produces.
- **Retool the junior ramp.** The traditional tier-1-to-tier-3 SOC progression built intuition through repetition. When repetition goes to agents, the intuition has to come from somewhere else. Structure your junior program around agent supervision, prompt injection testing, and red team exercises by day 90. The SANS 2026 Workforce Survey showed 27% of organizations experienced real breaches from skills gaps; this is where you prevent that.
- **Pay for SANS SEC595, SEC545, or equivalent AI-security training.** It is the cheapest thing on this list and the one with the highest morale return.

> **BUDGET**
>
> **$7K–$12K per head** for AI-security training. Cheapest line on the list, highest morale return. Most regulatory regimes also require annual security-team training, so half the cost can be charged to the compliance budget.

- **Name an AI Security Lead.** A principal engineer or director, not a new C-level role, with a mandate to own the AI threat model, the agent governance framework, and the internal tooling strategy. This person is your future deputy CISO.

Cyber insurance: the lever most CISOs are underusing

If you are having trouble getting board approval on any of the recommendations in this document, your cyber insurance renewal is your best unused argument. Underwriters are rewriting questionnaires around AI exposure for the 2026 cycle.

> **COVERAGE CONDITIONS, 2026**
>
> Phishing-resistant MFA for all privileged accounts. Evidence of segmentation beyond flat VLANs. Documented non-human identity governance. Incident response plans that address AI agent compromise. These are moving from 'nice to have' to **'we will not bind or we will raise your premium.'**

- **The practical move:** Get your broker to pull the 2026 questionnaire from your top three markets and walk it line by line against your program. Every item where the answer is 'no' or 'partial' is either a control gap you should fix or a premium increase you should budget for. Bring that walk-through to your board. 'Our insurer is requiring this' ends a conversation that 'the CISO thinks we should do this' does not. I have gotten more budget out of a 20-minute meeting with a broker than out of six months of internal advocacy, and I am not proud of it, but that is how these decisions get made.
- **One warning:** The insurance industry is using AI questionnaires as a pricing lever independent of risk reduction. Some of what they ask for is security theater dressed up as underwriting. Push back on items that duplicate controls you already have under a different name, and document the pushback. Your broker should be advocating for you here; if they are not, that is a broker problem.

There is a longer-game reason to engage with your insurer on AI controls now. Insurance questionnaires are not written in a vacuum any more than audit frameworks are.

Underwriters watch their best-performing accounts and codify what those accounts do into next year's questionnaire, which becomes the year-after's coverage condition, which becomes the industry baseline within three renewal cycles. Programs that engage now help define what reasonable AI controls look like in the underwriting language; programs that wait will be answering questions someone else's CISO already shaped. Your broker can tell you which carriers are soliciting input from mature accounts on the 2027 questionnaire. Get on that list. The conversation is cheap, the influence compounds, and the alternative is finding out three years from now that the standard control expectation for AI agent governance was written by your competitor's security team.

A 90-day action plan

If you're looking for something concrete to take into next Monday's staff meeting, this is it. Twelve weeks, three phases.

PHASE · WEEKS 1–4 **01**	**Inventory the truth** ▸ Pull runtime data from EDR, cloud APIs, IdP, and K8s into a single graph. Do not wait for a CAASM purchase, start with a Python script and Neo4j if you have to. ▸ Enumerate every agent and service principal with production write access. This is your new crown-jewel list. ▸ Draft the Standing Authority Matrix. Get it in front of GC within 30 days. ▸ Measure time-to-patch for your top 20 internet-facing assets. Publish the number internally.
PHASE · WEEKS 5–8 **02**	**Close the obvious gaps** ▸ 100% phishing-resistant MFA for tier-0 admins. No exceptions. Order hardware keys week 1. ▸ Deploy canary tokens in your top 5 target systems (source repos, vault, SharePoint, crown-jewel databases, CI/CD secrets). Thinkst, 30 minutes, done. ▸ Put an LLM reviewer in the PR path for one pilot codebase. Measure false positive rate and developer satisfaction. ▸ Turn on egress filtering in one production segment. Measure what breaks. Fix it.
PHASE · WEEKS 9–12 **03**	**Stand up the new functions** ▸ Post the VulnOps team, one security engineer, one developer, reporting line to CISO. Internal transfers count. ▸ Run a tabletop with an agent-compromise scenario. Invite GC and the CIO. ▸ Decommission one legacy tool from the divest list in the reprioritization table. Use the savings to fund a line item above. ▸ Begin a SIEM data-tiering evaluation: get quotes from Cribl, Snowflake, or Databricks for routing your log lake off the legacy rules engine. Present the reprioritization to the board. Use the Zero Day Clock as your one slide.

Every phase above maps to metrics in Appendix B. Start reporting them from week one. Trajectory is how you prove the program is working, your asset graph staleness going from 72 hours to 19 hours over a quarter is the story, not the 19.

What you should be telling your board

Four sentences, because that's all they'll remember:

01

The time between a bug being discovered and an exploit existing has collapsed from months to hours, and that trend is not reversing.

02

We cannot patch our way out of this, so we are investing in segmentation, identity boundaries, automated response, and internal AI-driven code review that contain breaches when they happen or prevent them before they start.

03

We are standing up an internal capability called VulnOps that uses the same AI tools the attackers use, pointed at our own code, and we are retiring the legacy controls that no longer earn their keep.

04

This is a net increase in cyber spend, and I am not going to pretend otherwise. Every function in this company is pouring money into AI right now, and cyber is no different, because the threat has moved faster than our current program was designed for.

"AI is a line item across the whole business and security needs its share" is an argument every board member is already hearing from the CIO, the CFO, and the head of product. You are asking for parity with a conversation that is already happening.

Closing

I have seen a few “everything changes” moments in this industry. Most of them didn’t. This one is different in one way: the velocity of the attacker’s loop has decoupled from the velocity of the defender’s loop. Closing that gap is a program-design problem, which is what a CISO does. Every recommendation in this document is buildable with technology that exists today and a budget you can find by retiring things you already know don’t work.

Start with the Standing Authority Matrix and the runtime-truth asset graph. Those two artifacts are the foundation everything else sits on.

If this was useful, tell me what I got wrong. I’ve been wrong before and I’ll be wrong again, and the only way any of us get this right is by comparing notes in the open.

Rob Fuller
rob@init6.com · https://robfuller.net

ABOUT THE AUTHOR

Rob Fuller (“mubix”) is VP of Information Security at a Fortune 500 and Chair of the Vulnerability Management Research Group. He has spent twenty-five years on the offensive side of security, is a U.S. Marine Corps veteran, and publishes research at malicious.link.

This document is circulated for peer CISO review. Feedback welcome.

Appendix A: Standing Authority Matrix (sample)

A starter template with seven entries across the action types most programs need. Adapt to your environment, get your General Counsel and CIO signatures, review quarterly, and treat it as a living document.

ACTION	TRIGGER	SCOPE	APPROVER	AUDIT	ROLLBACK
Block IP at edge	IDS high-confidence C2 match	Perimeter FW, all egress	SOC Tier 2 (pre-approved)	Weekly review	Auto-expire 24h
Isolate endpoint	EDR ransomware behavior	Single host	SOC Tier 1	Daily review	Manual release
Revoke OAuth token	Impossible travel + new grant	Single identity	SOC Tier 2	Weekly review	User self-service
Fail CI/CD build	LLM reviewer high-sev finding	Single PR	VulnOps agent	Monthly sample	Dev override w/ log
Rollback deploy	Runtime anomaly > 3σ	Single service	Platform on-call	Incident review	N/A (forward fix)
Disable service account	Dormant NHI reactivated	Single NHI	IAM on-call	Quarterly NHI review	Ticket to reinstate
Quarantine agent	Behavioral baseline deviation	Single agent	AI Security Lead	Monthly review	Human-in-loop restart

COLUMNS EXPLAINED

ACTION — The machine-speed action the SOC or VulnOps function is authorized to take.

TRIGGER — The condition that must be met before the action fires. Be specific; ambiguity here is where automation gets blamed for outages.

SCOPE — The blast radius limit, the guardrail that prevents one action from cascading.

APPROVER	Who owns the standing authority, and by extension who is accountable when it goes wrong.
AUDIT	How often the decisions this authority produced are reviewed.
ROLLBACK	How the action is reversed. Every action in this matrix must have a documented reversal path. If an action cannot be rolled back, it belongs in the human-in-the-loop queue, not the matrix.

Appendix B: KPI scorecard for the new program

I told you to kill patch SLA as a board metric without proposing replacements. Fair criticism. Ten metrics mapped old-to-new, with source system and target. Not all will apply to every program; pick six to eight for your board deck.

OLD KPI		NEW KPI	SOURCE	TARGET
30/60/90 patch SLA %	→	**Exploitability-weighted backlog**	Asset graph + exploit intel	<50 reachable criticals
# vulns open	→	**Bugs fixed before merge**	VulnOps PR telemetry	>80% in 2 quarters
MFA coverage %	→	**Tier-0 on FIDO2 %**	IdP	100%, no exceptions
Mean time to detect	→	**Identity MTTD**	ITDR	<15 min
Mean time to respond	→	**Time-to-rebuild tier-1**	Platform eng	<4 hours, tested qtr
Phishing click rate	→	**Canary token fire rate**	Thinkst / internal	Any fire = tabletop
Training completion %	→	**Agent-compromise tabletop cadence**	IR program	Quarterly
Pentest findings closed	→	**CART coverage of MITRE techniques**	CART platform	>75% T1-T3
Asset inventory %	→	**Asset graph median staleness**	CAASM	<24 hours
Vendor questionnaires sent	→	**Continuous vendor posture monitored**	TPRM API	>90% of tier-1

NOTES

- **First:** Every one of these is measurable with tooling you already have or are recommended to acquire in the main body of this document. None require a net-new purchase.
- **Second:** Report trends. The board does not need to know that your asset graph staleness is 19 hours; they need to know that it was 72 hours last quarter and 19 hours this quarter. Trajectory is how you show a program working.

Appendix C: How to get OT up to speed

OT does not have the market maturity that IT does. The device vendors are smaller, the refresh cycles are measured in decades, the protocols predate the threat model, and the buyers, utilities, hospitals, plants, municipalities, are individually too resource-constrained to demand what IT buyers now take for granted. The result is a segment where the defender's toolkit has barely moved in fifteen years while the attacker's toolkit has compounded.

Segmentation and monitoring are the floor. The ceiling requires markets that do not yet exist, and the CISO community is one of the few groups with the collective leverage to create them. The three gaps below are where that leverage should be applied first. If you run a venture arm, sit on an advisory board, chair an ISAC working group, or write checks to industry consortia, these are the bets to push:

- **AI-assisted firmware analysis and patching for orphaned OT.** Mythos and models like it can pull apart firmware binaries. The next step is companies that take an unsupported firmware image, identify exploitable primitives, and ship a validated binary patch: a legal, supported version of what the community has done informally for years. If you run a venture arm or sit on an advisory board, this is the bet to make.
- **In-line OT protection at the protocol layer.** Think WAF-for-Modbus, WAF-for-DNP3, WAF-for-BACnet. Deep protocol inspection with the ability to block anomalous writes in-line, not just alert on them. Some of this exists in pieces (Claroty xDome, Dragos, TXOne) but the in-line enforcement story is thinner than the detection story. Push your vendors on enforcement.
- **Shared defense funding for small OT vendors.** Many device vendors in this space are 20-person companies that cannot afford an AppSec program. The CISO community should be talking about whether large operators (utilities, hospital systems, manufacturers) collectively fund AI-driven code review and firmware hardening for their shared suppliers. Anthropic's Glasswing model, applied to the long tail of OT vendors, is the shape of the answer. One caveat worth naming: as of April 2026, exactly one CVE (CVE-2026-4747, a FreeBSD NFS RCE) is publicly attributed to Glasswing specifically. The full public accounting is scheduled for July 2026, and that report will tell us whether the shape scales. Watch it.