CSAI Foundation | Frontier Ready Cybersecurity

Frontier Ready Daily

Issue 88 · Member edition

2026-11-22



CSAI FOUNDATION INITIATIVE

Frontier Ready.

Machine-speed agentic cybersecurity

cloud security alliance®

CSAI

© 2026 Cloud Security Alliance. Some rights reserved.

You may download, store, display, view, print, redistribute, and link to this document in its original, unmodified form, provided that attribution to the Cloud Security Alliance is maintained and all trademark and copyright notices remain intact.

This document may not be modified or altered. You may quote portions of the document as permitted by the Fair Use provisions of the United States Copyright Act, provided that attribution is given to the Cloud Security Alliance.

This document may be shared on professional and social media platforms in its original form with attribution.

This document was generated with AI assistance and has not undergone official CSA review and approval processes.

In this issue

The lead item is Dragos's canary research. Attackers were inside three AI inference providers for up to four months, none of the providers knew, and some of the stolen requests carried a Zero Data Retention flag. OpenRouter has since delisted three providers and pulled 125 endpoints from ZDR eligibility. It is followed by an AI-found Ghostscript flaw that was still unpatched in several enterprise distributions more than two months after the upstream fix, Shopify's account of running judge-supervised attack agents against its own production systems, Elastic's finding that five attack frameworks seen in real intrusions route around model refusals, and two papers: one finds most harmful agent incidents involved no attacker, the other that AI has not yet measurably shortened fix or exploit intervals. We examined Armory Castle's vCenter zero-day (CVE-2026-123890) again and left it out. Google's account shows no AI role in finding or exploiting the flaw, and CSA's note covers it. We did not use Dark Reading's report on Qwen incidents in self-hosted deployments, because the article would not load for verification.

Attackers read customer prompts inside three AI inference providers for up to four months, none of the providers knew, and some stolen requests carried a Zero Data Retention flag

Category: agentic_surface **Significance:** major **Confidence:** VERBATIM (PROVIDER) for Dragos's statements that it seeded approximately two million prompts containing unique URLs and canary credentials from July through November 2026, that "the earliest incident dates back to the middle of July", that "none of the three providers knew about their breach until we notified them", and that "some of the stolen requests were marked as ZDR"; VERBATIM (PROVIDER) for OpenRouter's 22 November statement that it off-boarded three providers, removed 125 endpoints from ZDR eligibility, notified potentially affected customers, and found no breach of OpenRouter itself; NO PROVIDER CLAIM for the identity of the three breached providers, which neither Dragos nor OpenRouter has named, or for whether the three OpenRouter removed are the three Dragos found; LINK ONLY – VERIFY AT SOURCE for The Hacker News' account of one provider's shifting explanations; CHARACTERIZATION (CSA) for treating a ZDR flag as an unverified contractual claim rather than a control **Vendor interest:** Dragos sells security products and has acquired asset-discovery and software-supply-chain companies (runZero and NetRise), so a finding that AI supply chains need continuous independent visibility is commercially adjacent to its business. **Source:** Dragos – Three breaches, endless leaks: Your AI supply chain needs canaries

What changed. On 19 November 2026 Dragos published the results of five months of canary testing across inference providers and resellers. Each of about two million prompts carried a unique URL or AWS credential. Attackers had been inside three hosting providers, the earliest since mid-July. In the largest breach, every request in several late-October windows was copied, including requests rejected before any model ran. That points to collection at the provider's front door. Attackers tested the stolen AWS keys within hours, made tens of thousands of API calls, created IAM users and tried to enable Bedrock. From 1 November one actor ran ARTEX agents against the keys. Separately, an unnamed Chinese AI lab fetched secret URLs from more than half of the prompts sent to it, including about two-thirds of those marked no-collection or ZDR. On 22 November OpenRouter, which routes customer requests to third-party providers, delisted three providers, pulled 125 endpoints from ZDR eligibility, and said it will adopt the canary method as an ongoing check.

Why it reaches you. The component class is the model gateway and the inference providers behind it, including the routers and resellers your developers, agents and SaaS vendors call. Prompts routinely carry source code, customer data and live credentials. A ZDR flag is a promise made at the first hop, and it does not bind a compromised or forwarding provider downstream. None of the three breached providers detected the intrusion through its own monitoring, so the provider's assurances and audit reports would not have warned you.

What to do. CISO with the AI platform and procurement owners: escalate. Map which inference providers and routers your prompts actually reach, including those chosen by your SaaS vendors. Pin traffic to providers you have contracted with directly where data sensitivity requires it. Seed your own canary credentials and URLs into prompts on each path. Treat any credential that has appeared in a prompt as exposed and rotate it. Require breach-notification and onward-routing terms, not only a retention flag. Urgency: escalate.

CSA coverage. Fireworks AI Breach: Inference Providers as Supply-Chain Chokepoints; see also Self-Funding Agents: ARTEX Campaigns Harvesting AI-Gateway Keys.

Debian 12, Amazon Linux and several SUSE lines were still unpatched against an AI-found Ghostscript sandbox escape more than two months after the upstream fix

Category: vuln_storm **Significance:** notable **Confidence:** VERBATIM (PROVIDER) for pwn.ai's statement that "GPT 5.6 Sol, working with @PaulosYibelo, found the issue", and for its timeline: found 27 July 2026, reported to Artifex by 2 September, fixed in Ghostscript 10.08.0 on 8 September, CVE-2026-125396 requested 6 November and published 9 November; SELF-REPORTED (PROVIDER METRIC) for pwn.ai's 12 November distribution survey (Debian 12 "Vulnerable", Amazon Linux 2/2023/2027 Preview

"Pending Fix", SLE 12 SP5 LTSS, some SLE 15 SP4–SP7 lines and SLES 16.0/16.1 "In progress", openSUSE Leap 16.0/16.1 "Affected", RHEL-family status not established) and for command execution shown in 10 applications; NO PROVIDER CLAIM for exploitation in the wild, which neither pwn.ai nor The Hacker News reports; CHARACTERIZATION (CSA) for the reading of the fix-to-distribution lag **Source:** pwn.ai — Upload2Shell research

What changed. On 21 November 2026 pwn.ai published its research on CVE-2026-125396. It is a use-after-free in Ghostscript's Type 1 font loader that switches off the `-dSAFER` sandbox. The researcher found it with GPT 5.6 Sol. Image and document libraries hand PostScript content to Ghostscript even when the file is named `.png` or `.pdf`. Through that route the researchers showed command execution in 10 applications, including Debian 13 CUPS, Roundcube, MediaWiki, Plesk, WordPress Core and two WordPress form plugins. They traced code paths in 8 more, including ComfyUI, CVAT and Paperless-ngx. Artifex shipped the fix on 8 September. The CVE was not published until 9 November, 62 days later. As of 12 November, Debian 12, Amazon Linux and several enterprise SUSE lines were still unpatched, and pwn.ai could not determine the status of the RHEL family.

Why it reaches you. The component class is any server that turns uploaded files into thumbnails or previews: document management, webmail, wikis, ticketing, and AI image-generation and annotation tools. Ghostscript copies bundled in containers and application images are not fixed by patching the host. CVE-keyed scanners had nothing to match for the two months before the CVE was published, and enterprise distributions are still behind upstream.

What to do. Vulnerability management owner: validate. Find every Ghostscript instance, including copies inside container images, and confirm each is version 10.08.0 or a backport carrying the `.putgstringcopy` fix. Where your distribution has not shipped one, turn off PostScript and EPS handling in ImageMagick and similar libraries, or run upload rendering in an isolated, unprivileged sandbox. Track the date of the upstream fix, not just the CVE date, for dependencies that sit under file-upload paths. Urgency: validate.

CSA coverage. New — no prior CSA coverage. For the wider pattern, see When Discovery Outruns Remediation: Open-Source Disclosure Strain.

Shopify runs AI attack agents against its live production systems under a differently aligned judge agent that can kill them, and reports tens of thousands of engagements without an outage

Category: security_operating_model **Significance:** notable **Confidence:** VERBATIM (PROVIDER) for Shopify's description of the architecture: Discovery and Sandbox stages, one Kubernetes pod per engagement, short-lived read-only session credentials, a credential-substituting proxy, an allowlist egress proxy as the only path out of the VPC, hard rules against lateral movement or keeping access, and a judge agent on a different model that kills the executor and fails closed if it times out or crashes; SELF-REPORTED (PROVIDER METRIC) for "tens of thousands of offensive engagements", "dozens of impactful findings" and "without causing an outage or incident"; NO PROVIDER CLAIM for time to find or fix, cost, severity breakdown, how findings are routed, or staffing effects **Source:** Shopify Engineering — Cyber Chaos Monkey: Staying ahead of security threats with AI agents

What changed. On 12 November 2026 Shopify described Cyber Chaos Monkey. It gives AI agents live test credentials, source code, past bug-bounty reports and deployment records, and points them at its production systems. Engagements are started by security engineers, by other agents, or by the system's own feedback loops. Each executor agent is watched in real time by a judge agent on a different model, which can end the run if the executor breaks the rules of engagement. If the judge or the orchestrator stops responding, the pod is torn down. Shopify says the system has run tens of thousands of engagements and produced dozens of significant findings without an outage. It was built partly because bug-bounty submissions have risen almost 300% since 2024. This is the first detailed public account we have seen of an enterprise running continuous autonomous offensive testing against live production, with the containment design spelled out.

Why it reaches you. The component class is your offensive-testing programme: red team, penetration testing and bug-bounty triage. Shopify's post is a working reference for the controls that make autonomous testing in production defensible: scoped and rotating credentials, egress allowlisting, post-exploitation limits, and an independent monitor that fails closed. These are also the controls auditors and boards will ask about once internal attack agents exist. The post's own nod to "recent incidents" of test agents reaching unintended systems is why the judge matters.

What to do. Head of offensive security or red team: monitor, and validate before any pilot. If you run or plan autonomous offensive testing, compare its containment against this design point by point. Check for a separate monitor on a different model, failure that kills the run, credentials the agent cannot alter, egress allowlisting and written post-exploitation limits. Measure findings against engagement volume so the programme's yield can be assessed, a figure Shopify does not publish. Urgency: monitor.

CSA coverage. Cyber Defender Models: A CISO's Guide to Access, Testing, Containment, and Gov.

All five agentic attack frameworks seen in real intrusions skip authorization checks by default, and one reassigns any task a model refuses within five seconds

Category: machine_speed **Significance:** notable **Confidence:** VERBATIM (PROVIDER) for Elastic's source-code findings: "None of the five frameworks enforces authorization in code by default. Three assert it in the prompt", that Cairn sidelines a refusing worker for five seconds and gives the task to another, and that CyberStrikeAI ships with approval mode off, PentAGI with ASK_USER false, and ARTEX in a bypass permission mode; VERBATIM (PROVIDER) for the proposed hunt threshold of at least three tool families and 20 requests per source IP per minute; LINK ONLY – VERIFY AT SOURCE for the two-to-25-second ARTEX execution bursts, which Elastic attributes to Google Threat Intelligence; NO PROVIDER CLAIM for a measured detection or false-positive rate, since Elastic's only worked example match was a developer workstation **Vendor interest:** Elastic sells the SIEM and detection products in which its proposed hunts run. **Source:** Elastic Security Labs – Inside the loop: Five AI attack frameworks that never ask permission

What changed. On 13 November 2026 Elastic published a source-code review of ARTEX, CyberStrikeAI, Cairn, PentAGI and HexStrike. Google, Mandiant and Microsoft have each reported at least one of them in real intrusions. None checks authorization in code by default. Three tell their models the engagement is already authorized, and HexStrike accepts an out-of-scope list that it never reads. Cairn treats a model refusal as a scheduling event and gives the task to another worker within five seconds. The only default restrictions are government-domain blocklists in ARTEX and CyberStrikeAI, plus an education-domain blocklist in ARTEX. Elastic proposes detecting tempo per identity rather than per signature. Its example hunt flags a source address that uses several tool families within one minute, and Elastic notes that build systems and shared NAT addresses will trigger it until it is tuned.

Why it reaches you. The component class is your detection pipeline for cloud control-plane and web-tier activity. These frameworks act in bursts of seconds, and refusals by one model do not stop the operation. Containment therefore cannot rely on the attacker's model provider. Your detections need to fire on behaviour per identity within a minute, and responses have to act on that timescale.

What to do. Detection engineering owner: validate. Build and tune a per-identity, per-minute tempo detection on cloud API and web logs, and baseline it against your own CI, monitoring and admin traffic before alerting on it. Load the published framework User-Agents and ports as low-cost indicators, expecting attackers to change them. Measure how long it takes from a detection firing to the identity being revoked. Urgency: validate.

CSA coverage. ARTEX and Its Forks: Agentic Pentest Frameworks as Attack Infrastructure.

Most real-world agent incidents that caused harm involved no attacker, and a warning that names the specific hazard cut harm across five model families

Category: agentic_surface **Significance:** context **Confidence:** VERBATIM (PROVIDER) for the ROGUE-AI abstract's figures: reported harm reproduced in 89% of 108 replications, agents on five GPT, Claude, Gemini and DeepSeek models making the reported mistake in 18% to 33% of runs against at most 1% when the cause was removed, 62 of 72 incidents with real-world harm involving no attacker, and hazard-naming warnings after each tool output reducing harm for all five models while an equal-length warning that did not name the hazard did not; LINK ONLY – VERIFY AT SOURCE for the per-incident results in the paper body, which we did not read; LIVE TEST REQUIRED for whether hazard-naming warnings transfer to production agent harnesses **Source:** arXiv – Replication of Agentic Security Events: Audit and Investigation (ROGUE-AI)

What changed. On 19 November 2026 researchers posted ROGUE-AI. It rebuilds reported agent incidents offline, such as an agent deleting a production database, and runs each rebuild against a control that lacks only the suspected cause. Across 108 replications the reported harm recurred 89% of the time. In 62 of the 72 incidents that caused real-world harm there was no attacker. The agent did its assigned task and missed an ordinary hazard. Those no-attacker replications caused more harm than the prompt-injection replications on all five models. A warning after each tool output that named the hazard reduced harm. A generic warning of the same length did not.

Why it reaches you. The component class is the agent runtime's guardrails and the incident process around them. Controls built mainly against injection miss the larger source of harm in this sample: agents that are not under attack, taking destructive actions while doing their job. The replication method also gives incident responders a way to confirm a root cause, not just assume one.

What to do. Agent platform owner: monitor. Check whether your agent guardrails and approval gates cover destructive actions taken with no adversarial input, not only injection. When investigating an agent incident, re-run it with and without the suspected cause before closing it. Urgency: monitor.

CSA coverage. New – no prior CSA coverage.

A systematisation of AI vulnerability research finds disclosures more than doubled in 2026, while report-to-fix and disclosure-to-attack intervals have not yet measurably shifted

Category: vuln_storm **Significance:** context **Confidence:** VERBATIM (PROVIDER) for the abstract's statements that "in 2026 the number of disclosed vulnerabilities more than doubled", that decade-old severe bugs were reported "several times more often than before", that agents found, validated and

patched real vulnerabilities without human help before the restricted frontier models of 2026, with open-weight models later reaching comparable autonomy, and that the time from report to fix and from disclosure to attack "has not shifted noticeably so far"; LINK ONLY — VERIFY AT SOURCE for the underlying datasets and interval measurements, which we did not read beyond the abstract; CHARACTERIZATION (CSA) for the reading that remediation capacity, not exploitation speed, is the near-term constraint **Source:** arXiv — SoK: Assessing the Evolution of LLM Capability in Software Vulnerability Research

What changed. On 19 November 2026 researchers posted a systematisation of LLM capability in vulnerability research. It rates claims on a driving-automation-style autonomy scale and weighs each by its evidence. The paper finds that disclosed vulnerability volume more than doubled in 2026 and that old critical bugs are surfacing several times faster. But the aggregate interval from report to fix, and from disclosure to attack, has not yet shifted noticeably. That cuts against the single-case compressions this feed has reported, including a two-hour Java4Shell exploit and an 11-minute MONAI Label exploit. It suggests those cases have not yet moved the population averages.

Why it reaches you. The component class is your vulnerability management programme's capacity planning. If volume has doubled and intervals have not compressed, the near-term pressure is on intake, triage and remediation throughput, not yet on a universal collapse of time to exploit. The individual fast cases still set the worst case for internet-facing assets.

What to do. Vulnerability management owner: no action. Use the paper's per-stakeholder mitigation rubric as an input when sizing 2027 remediation capacity against doubled intake. Urgency: no action.

CSA coverage. The Widening Gap: AI Discovery Outpaces Patch Capacity.

Watchlist

- **OpenAI reward-hacking postmortem — downstream response** — No change. *(opened 2026-08-27)*
- **VM/hypervisor containment hardening for cyber-capable agents** — No provider has announced a move from stock QEMU/KVM to microVMs. Shopify's Cyber Chaos Monkey (item 3) contains its attack agents with Kubernetes pods, egress allowlisting and a fail-closed judge agent, not with microVM isolation, and does not say whether hypervisor isolation was considered. *(opened 2026-08-27)*
- **Claude Code Auto Mode prompt-injection ASR discrepancy** — No change. *(opened 2026-08-27)*

- **AI defensive-triage guardrail evasion** — No new campaign weaponising refusals against defensive triage. A related signal on the attacker side: Elastic (item 4) documents Cairn reassigning any refused task to another worker within five seconds, so model refusals do not reliably stop attacker frameworks either. *(opened 2026-08-31)*
- **AI account session hijacking at scale** — No change. Dragos (item 1) concerns AWS credentials stolen from prompts inside inference providers, not hijacked AI-provider user sessions. *(opened 2026-08-31)*

Opened this issue

- **Inference-provider data handling and ZDR verification** — Watching for the three breached providers to be named, the criteria OpenRouter sets for restoring ZDR eligibility to the 125 endpoints, other model gateways adopting continuous canary testing, and regulatory or contractual action over requests marked ZDR that were stolen or fetched. *(opened 2026-11-22)*