CSAI Foundation | Frontier Ready Cybersecurity

Frontier Ready Daily

Issue 78 · Member edition

2026-11-12



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

Hijacked GitHub accounts pushed a CI secret stealer into 61,764 repositories in about a day, and each payload was reverted with its run logs deleted within at most a minute, faster than any human review of workflow history. Attackers also ran a forked open-source agentic pentest framework against AWS estates, and the other items cover a lab demonstration that turns a stolen Claude API key into a command channel that never leaves the AI provider's own endpoint, an AI-found Apache HTTP/2 flaw that reached a fix in 16 days but not yet most distribution packages, and traffic data showing attackers borrowing AI agents' names to get past bot defences. We examined GitHub's interview on Agentic Autofix, Microsoft's Entra hard-deletion preview and a sponsored opinion piece on agent offboarding, and left all three out because each is a vendor's account of its own product with no independent evidence of practice.

Hijacked GitHub accounts pushed a CI secret stealer into 61,764 repositories in about a day, and each run was reverted and its logs deleted within a minute

Category: machine_speed **Significance:** major **Confidence:** SELF-REPORTED (PROVIDER METRIC) for the 4,837 accounts, 61,764 repositories and the 1,806 to 1,835 repositories that ran the payload (Aikido gives both figures and does not reconcile them); VERBATIM (PROVIDER) for the "at most one minute" revert and run deletion, the AI-tool credentials targeted, and the persistence steps; NO PROVIDER CLAIM from GitHub on takedowns or token revocation; CHARACTERIZATION (CSA) for the reading that log-based CI review cannot see this campaign **Vendor interest:** Aikido sells software supply-chain and CI security tooling, so the scale of the finding is commercially favourable to it. **Source:** Aikido – wfx-loot campaign pushes CI secret stealer to tens of thousands of repositories

What changed. From about 12:45 UTC on 10 November 2026, an actor holding stolen GitHub credentials opened probe pull requests titled "docs: dependency audit notes" to test write access, then pushed a roughly 1,500-line shell script into existing workflows, npm install hooks or new workflows on throwaway branches. Aikido counts 4,837 compromised accounts and 61,764 repositories reached by 11 November 15:00 UTC, with at least 1,806 repositories (1,835 in another passage) running the payload. The actor "reverts within at most one minute and deletes the workflow run, along with its logs". The script takes Actions secrets, cloud and SSH credentials, and tokens for AI providers and coding tools including Anthropic, OpenAI, Gemini, Hugging Face, Cursor and Copilot. The actor then made long-lived AWS keys, set role trust policies to a wildcard principal, added federated credentials named `wfx-ci-…` to Azure applications, and forged passing status checks to get past branch protection.

Why it reaches you. The component is the CI pipeline and the cloud roles its workflows can assume. A one-minute window with the run deleted leaves at most a skipped run number, so any control that depends on someone reading workflow history or Actions logs did not see this. The persistence outlives rotation: a wildcard trust policy or an extra Azure federated credential still admits the attacker after every secret is replaced. AI-provider and coding-agent tokens are now standard loot alongside cloud keys.

What to do. Platform engineering owner, with cloud security: escalate. Search every organisation for a `wfx-docs` branch, the probe PR title, and the commit titles Aikido lists. Where the payload ran, treat every secret and role the workflow could reach as compromised, then audit AWS role trust policies for wildcard principals and Azure applications for unexpected federated credentials, because rotation alone will not remove them. Stream GitHub audit-log events for workflow-run deletion to your SIEM so the cleanup itself becomes the alert. Urgency: escalate.

CSA coverage. The wfx Campaign: Hijacked GitHub Accounts Harvest CI Secrets.

Attackers ran AntiFox, a forked open-source agentic pentest framework, against IIS servers and AWS accounts, making hundreds of authenticated API calls in about 30 minutes and reaching production secrets

Category: machine_speed **Significance:** major **Confidence:** LINK ONLY — VERIFY AT SOURCE for Arctic Wolf's findings as reported by BleepingComputer, including the 30-minute burst and the three intrusion cases; VERBATIM (PROVIDER) for Amazon's statement that "AWS services were not compromised and operated as designed"; NO PROVIDER CLAIM on the number of victim organisations or the time from initial access to secret theft; CHARACTERIZATION (CSA) for the forkability reading **Vendor interest:** Arctic Wolf sells managed detection and response, so findings that favour correlated endpoint-and-cloud monitoring are commercially favourable to it. **Source:** BleepingComputer — Hackers use AntiFox AI agents in AWS credential-theft attacks

What changed. On 11 November 2026 BleepingComputer reported Arctic Wolf research showing attackers hosting self-managed instances of AntiFox, an open-source agentic penetration-testing platform forked from ARTEX 0.1.1, on Alibaba Cloud and using them in live intrusions. In one case the agents planted an ASPX webshell on an internet-facing IIS server and pulled application settings and database connection strings. In another, using valid IAM users and assumed EC2 roles, they made "hundreds of API requests in about 30 minutes". Most were denied, but some succeeded and returned production secrets. A third case read, copied and deleted S3 objects. AntiFox's publishers forbid use against live systems but describe no technical control that enforces it.

Why it reaches you. The exposure path is the credential stored in application configuration on an internet-facing server, which becomes an AWS identity the agent can drive. Closing ARTEX's source did not take its capability out of circulation, because a pre-closure fork now ships openly with a newer release. The signal that mattered was a few successful calls inside a burst of denials. A detector tuned to alert on denials, or to wait for a human to correlate the webshell with the cloud activity, is working on a slower clock than the attack.

What to do. Cloud security owner: validate. Write a detection for any IAM principal that produces a burst of access-denied errors followed by successful secret or S3 reads within the same 30-minute window, and link it to webshell and new-file alerts on the hosts that hold that principal's keys. Remove long-term access keys from application configuration files and require IMDSv2 on EC2. Urgency: validate.

CSA coverage. AntiFox in the Wild: Forkable Agentic Frameworks Reach AWS Secrets; see also Trojanized Offensive AI Tooling: The ARTEX Docker Backdoor.

Darktrace turned a stolen Claude API key into a command channel by passing jobs through Claude memory stores, and exfiltrated 1 GB through the Files API without leaving the provider's endpoint

Category: agentic_surface **Significance:** notable **Confidence:** VERBATIM (PROVIDER) for the memory-store mechanism, the nine jobs, the 1 GB exfiltration, and Darktrace's statements that the technique "does not rely on a vulnerability in the Claude platform" and "has not been observed by Darktrace in the wild"; LINK ONLY — VERIFY AT SOURCE for the use of `--dangerously-skip-permissions` and the statement that OAuth credentials cannot reach memory stores, which appear in Help Net Security's interview but not in Darktrace's text; NO PROVIDER CLAIM from Anthropic; LIVE TEST REQUIRED for whether your own egress and endpoint controls would distinguish this traffic **Vendor interest:** Darktrace sells behavioural network and endpoint detection, so a finding that only behavioural correlation catches the technique is commercially favourable to it. **Source:** Darktrace — Living off the AI: Shared Memories as a Covert C2 Channel

What changed. On 12 November 2026 Darktrace published a lab demonstration. After compromising a developer's Linux machine and taking the Claude Console credentials stored on it, the operator set up a Claude memory store with three text files: a heartbeat, an input file of jobs written as plain-English objectives, and an output file for results. The local Claude Code agent read its jobs from the store and wrote results back. Across nine jobs against a simulated Active Directory network it mapped hosts, pulled a password from an anonymous FTP share, sprayed it at the domain controller, listed SMB shares, and uploaded a 1 GB file in three parts to the Files API. Every connection went to the endpoint the agent already uses for inference.

Why it reaches you. The component is the developer endpoint running a coding agent and the AI-provider API key on it. Allow-listing the AI provider's domain, which most organisations have done to let coding agents work, also lets this command channel and the exfiltration through. No vulnerability is involved, so there is nothing to patch, and the attacker needs no infrastructure of its own. Darktrace has not seen it used in the wild, but every prerequisite matches activity we have already reported, from infostealers taking AI-tool tokens to the wfx campaign above.

What to do. Endpoint security owner, with the AI platform owner: validate. Move developer access to AI providers from long-lived API keys to OAuth or federated identity where the provider supports it. Where it does not, issue short-lived, workspace-scoped keys kept out of dotfiles and shell history. Then test whether your endpoint telemetry can tie an agent process's command-line arguments and child processes to its outbound data volume, and alert on upload ratios far above normal inference traffic. Urgency: validate.

CSA coverage. New — no prior CSA coverage of this technique; see also Approved Once, Leaked Forever: Secrets in Claude Code Settings Files.

An AI scanner reported a critical Apache HTTP/2 flaw eight days before Google's researchers found it too, and a day after the fix most distribution packages were still unpatched

Category: vuln_storm **Significance:** notable **Confidence:** VERBATIM (PROVIDER) for Apache's credits to "Anthropic's OSS Scanner" and Google Cloud Vulnerability Research, the 26 October and 3 November report dates, and the 11 November release of 2.4.71; VERBATIM (PROVIDER) for Red Hat's 9.8 score, its Important rating for RHEL 9 and 10, and its statement that public research "has demonstrated a reliable exploit against a default-configured server"; LINK ONLY — VERIFY AT SOURCE for per-release status, which the Debian tracker shows inconsistently for bookworm and which we could not confirm for RHEL errata or Ubuntu; CHARACTERIZATION (CSA) for the distribution-gap reading **Source:** Apache HTTP Server Project — Apache HTTP Server 2.4 vulnerabilities

What changed. CVE-2026-119618 is a race condition in `assign_headers()` in mod_http2 that can corrupt heap memory and lead to unauthenticated remote code execution on httpd 2.4.55 through 2.4.70 with HTTP/2 enabled. Apache rates it critical. Anthropic's OSS Scanner reported it on 26 October 2026, and Google's researchers reported the same flaw independently on 3 November. Apache shipped the fix in 2.4.71 on 11 November, 16 days after the first report. By 12 November, Debian had fixed trixie through its security archive and sid, but its tracker still listed bookworm and forky as vulnerable. Red Hat lists RHEL 9 and 10 as affected, and we could not confirm a released errata or an Ubuntu notice. No exploitation in the wild has been reported.

Why it reaches you. The component is the web tier and reverse proxies built on distribution httpd packages, including container base images that inherit them. Upstream turned the report around in about two weeks, but most enterprises consume httpd from a distribution, and that second leg is where the clock is now running. Two unrelated teams found the same bug eight days apart, which suggests the attacker-side discovery window for flaws of this kind is short. Red Hat says a reliable public exploit exists.

What to do. Vulnerability management owner: validate. Inventory httpd and standalone mod_h2 builds with HTTP/2 enabled, including in container images. Where your distribution has not shipped 2.4.71, disable HTTP/2 (`Protocols http/1.1`) on internet-facing servers until it does, and record the days between the upstream fix and your deployed package as a measured interval. Urgency: validate.

CSA coverage. CVE-2026-119618: AI-Discovered Apache HTTP/2 RCE and the Distribution Patch Gap; see also The Disclosure Bottleneck: When AI Discovery Outpaces Patch Consumption.

Credential-stuffing traffic used a consumer AI agent's name for 414,210 sign-in attempts in one week, about 1.5 times the genuine agent traffic Cequence counted across five industries

Category: agentic_surface **Significance:** notable **Confidence:** SELF-REPORTED (PROVIDER METRIC) for the 414,210 visits, the 281,627-visit agent sample, the telecom traffic rise from about 100 to 19,000–32,000 requests a day, and the 3.51 million blocked logins of which about 82% scored too low for conventional bot detection; NO PROVIDER CLAIM on how genuine and spoofed agent traffic were classified; CHARACTERIZATION (CSA) for the identity-control reading **Vendor interest:** Cequence sells bot management and API protection, so a finding that name-based bot controls fail is commercially favourable to it. **Source:** Help Net Security – Bots pose as AI agents, AI agents pose as humans, and defenses believe both

What changed. On 12 November 2026 Help Net Security reported traffic data from Cequence. Over one week in October, an online marketplace logged 414,210 visits carrying the name of a widely used consumer AI agent, all aimed at the sign-in page and all failing. They came from a credential-stuffing operation, and the volume was about 1.5 times all genuine agent visits Cequence counted across five industries that week. In the opposite case, a real personal agent launched in September presented itself as an ordinary desktop browser and lifted automated traffic on a Tier 1 carrier's sites from about 100 to 19,000–32,000 requests a day. Of 281,627 agent visits sampled in mid-October, about one in seven went beyond reading into booking, buying, signing in or managing accounts. The report follows Cequence's late-October finding of one scanner rotating through about 32 AI-bot identities to hunt for credential files.

Why it reaches you. The component is the customer-facing sign-in and transaction path, and the bot policy in front of it. Allow rules written for named AI agents so that customers' agents can transact now admit attackers who use those names, while real agents that do not declare themselves get no agent-specific controls at all. Independent confirmation would need comparable figures from a second traffic operator, or the agent providers' own published verification data.

What to do. Application security owner, with the fraud team: validate. Remove any allow rule that trusts a self-declared agent user-agent. Admit named agents only through provider-published address ranges, signed requests or a direct integration, and apply step-up verification to payments, payee changes and account-control changes whatever the client claims to be. Urgency: validate.

CSA coverage. New — no prior CSA coverage of agent-name spoofing against customer-facing services; see also NIST's Agent Identity Tenets: A Proposed Federal Reference Point.

Watchlist

- **OpenAI reward-hacking postmortem — downstream response** — No change. No frontier lab disclosed an eval-to-production escape, and there was no new regulatory or contractual development. *(opened 2026-08-27)*
- **VM/hypervisor containment hardening for cyber-capable agents** — No change. No provider or enterprise moved agent sandboxes to hardened microVMs, and there is no new QEMU/KVM/libslirp patch news. *(opened 2026-08-27)*
- **Claude Code Auto Mode prompt-injection ASR discrepancy** — No change. Darktrace's memory-store channel (item 3) runs Claude Code with tool-use confirmations disabled on a compromised host, so it does not bear on Auto Mode's injection resistance. There is still no Anthropic patch or public response. *(opened 2026-08-27)*
- **AI defensive-triage guardrail evasion** — No change. No new campaign triggering refusals to blind triage tooling was reported. *(opened 2026-08-31)*
- **AI account session hijacking at scale** — Adjacent signal only. The wfx campaign (item 1) harvested Anthropic, OpenAI, Gemini, Cursor and Copilot credentials from CI runners, and Darktrace showed what a stolen Claude API key enables on a developer host (item 3). Both concern API keys and tool tokens, not AI-provider user sessions. No provider disclosed a session-hijacking campaign or shipped device-bound session tokens. *(opened 2026-08-31)*

Opened this issue

- **AI-provider APIs as command-and-control channels** (agentic_surface) – Watching for the first in-the-wild use of AI-provider storage features (memory stores, file APIs) as C2 or exfiltration channels after Darktrace's lab demonstration, for reproduction on a second provider, and for providers to add per-key restrictions on storage endpoints or tenant-visible logging of them. *(opened 2026-11-12)*