CSAI Foundation | Frontier Ready Cybersecurity

Frontier Ready Daily

Issue 93 · Member edition

2026-11-27



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 is Unit 42's measurement of more than ten agentic intrusion campaigns, where the median gap between commands fell from 23.9 seconds to 1.4 seconds once the agent took over from the human operator, and one agent spread web shells to more than 40 locations in 26 minutes. Also here: Italy's CSIRT on a six-agent framework that compromised many systems within hours, Palo Alto's kernel fix for the second stage of the PHP2Shell firewall chain about four weeks after CISA reported it in use, command-gate bypasses in eight production coding agents, the UK NCSC's advice to rate-limit by account and token rather than IP, and a benchmark showing closed-weight models refusing 28% to 62% of safe defensive requests.

Once the agent took over from the human operator, the median gap between commands fell from 23.9 seconds to 1.4 seconds across more than ten tracked campaigns

Category: machine_speed **Significance:** major **Confidence:** SELF-REPORTED (PROVIDER METRIC) for the 23.9-second and 1.4-second median gaps, measured on "a subset of analyzed attacks" whose size is not given, and for the 40-plus web-shell locations in 26 minutes; VERBATIM (PROVIDER) for "In every observed campaign the attackers gained their initial foothold through web shells that were deployed via vulnerable web applications"; CHARACTERIZATION (CSA) that the handoff point gives defenders a detectable change in tempo; LIVE TEST REQUIRED for the false-positive rate of the suggested hunts in a given estate **Vendor interest:** Palo Alto Networks sells XDR and automated-response products, and the post argues that defence requires AI-driven detection with automated response, which is commercially favourable to it. **Source:** Unit 42 – Recent Trends and Tactics of Agentic AI Attacks

What changed. On 24 November 2026 Unit 42 published its analysis of more than ten campaigns it tracked between July and September 2026, in government, logistics, hospitality, manufacturing, agriculture and education. The campaigns followed the same sequence each time. A human placed a web shell on a vulnerable web application. An agent then carried out reconnaissance, credential harvesting, privilege escalation and lateral movement on its own and reported back in machine-readable output, and the human chose the next objective. On the subset of attacks Unit 42 measured, the median gap between commands was 23.9 seconds during the human phase and 1.4 seconds during the agent phase. In one intrusion the agent began deploying ASPX web shells over SMB within two minutes of finding a Windows

IIS server and reached more than 40 locations in the next 26 minutes. In another it ran about 800 unique commands in 140 minutes of activity. Unit 42 says it found no behavioural markers that identify the model or framework used.

Why it reaches you. The entry point in every campaign was an internet-facing web application, and the agent's work happened inside the network: in shells, SMB shares and cloud APIs (one agent enumerated 20 AWS services across eight regions). An alert that waits for an analyst to triage it will arrive after the lateral movement is over. Unit 42 lists signs that survive the per-target custom code: section headers formatted `===[Title]===` in every campaign it observed, chain-of-thought comments in scripts in more than half, `find` commands that exclude EDR directories, and dozens of unique Base64 payloads in one session. These are signs of tempo and style, not signatures.

What to do. Detection engineering lead: validate. Write a rule that fires when command rate from a single web-server process tree jumps, sub-second command gaps following a slower interactive phase, together with any of the four signs above. Wire that rule to an automated containment action on the host (isolate it, or kill the web worker) rather than to an analyst queue, and test it against one replayed web-shell session before enabling it.

CSA coverage. Fingerprinting Agentic Offensive Frameworks: Intent IDs, Cadence, and Self-Labels

Italy's CSIRT says a six-agent framework compromised a large number of Italian systems within hours and wrote malicious code on the fly to get past endpoint defences

Category: machine_speed **Significance:** major **Confidence:** VERBATIM (PROVIDER) for the six-agent coordinator architecture, for the compromise "in poche ore, di un numero elevato di sistemi" ("within a few hours, of a large number of systems"), for agents "arrivando a scrivere codice malevolo in tempo reale per eludere le difese sugli endpoint" ("going as far as writing malicious code in real time to evade endpoint defences"), and for monthly patching of internet-facing software being "non accettabile"; NO PROVIDER CLAIM on the number of victims, sectors, or how AI-agent involvement was established; CHARACTERIZATION (CSA) that the bulletin sets a national benchmark against monthly patch cycles **Source:** Agenzia per la Cybersicurezza Nazionale – Impiego di agenti di IA in campagne di compromissione automatizzate (BL01/261123/CSIRT-ITA)

What changed. On 23 November 2026 CSIRT Italia issued bulletin BL01/261123/CSIRT-ITA, rated critical. It reports mass-compromise campaigns in Italy run through ARTEX, CyberStrikeAI, DeepSeek Harness – Red Team and PentAGI. In the DeepSeek Harness case, a coordinator agent assigned work to five specialist agents covering information gathering, asset classification, vulnerability finding, exploitation, and access and persistence. The framework ships with a system prompt designed to get

around the model's safety constraints. The agents pivoted through compromised hosts and could go beyond the subnet the operator gave them, and running in parallel they compromised a large number of systems within a few hours. The bulletin lists 81 CVEs marked for proof-of-concept and exploitation status, and it publishes daily-updated MISP indicator feeds covering seven platforms. It names neither sectors nor victim counts.

Why it reaches you. Two days after Switzerland's NCSC set a 24-hour patch benchmark, a second European national CSIRT has said that monthly patching of internet-facing software is no longer acceptable. It adds a detail that matters for endpoint-centric defence: the agents rewrote their own modules to evade endpoint controls. The bulletin says endpoint monitoring alone gives only "point-like" visibility, and that the activity shows up only when request origin, frequency, data volume and sequence are correlated across sources. Its detection list is specific: internal Kerberos, NTLM, LDAP and SMB scanning from DMZ hosts, and outbound connections from servers to tunnelling, webhook, OAST or unapproved AI platforms.

What to do. SOC manager: validate. Load the CSIRT Italia MISP feeds into the SIEM, honouring the validity window on each indicator, and confirm that two detections from the bulletin exist and fire: scanning from DMZ hosts toward internal directory services, and server egress to webhook or OAST domains. Where either is missing, raise it as a detection gap with the CISO.

CSA coverage. Machine-Speed Attacks: Cloud Defense at the Inflection Point

Palo Alto shipped kernel fixes for the second stage of the PHP2Shell firewall chain about four weeks after CISA reported the chain in use, and rated the bulletin informational

Category: vuln_storm **Significance:** notable **Confidence:** VERBATIM (PROVIDER) for "Exploitation of CVE-2026-46300 has been observed targeting the Palo Alto Networks management interface exposed to the internet", for the fixed builds, and for the statement that the only known exploitation path runs through PHP2Shell; VERBATIM (PROVIDER) for the PHP2Shell hotfix release dates of 25–29 October in the CVE-2026-109618 advisory; CHARACTERIZATION (CSA) of the roughly four-week interval between CISA's 28 October alert and the kernel fix; NO PROVIDER CLAIM on how many devices were compromised or on Panorama coverage of the kernel fix **Source:** Palo Alto Networks — PAN-SA-2026-0015 Security Bulletin: OSS CVEs Fixed in PAN-OS

What changed. On 24–25 November 2026 Palo Alto published PAN-SA-2026-0015. It confirms that the Linux kernel privilege-escalation flaw Fragnesia (CVE-2026-46300) has been exploited against internet-exposed PAN-OS management interfaces and lists fixed hotfix builds across PAN-OS 10.2, 11.1, 11.2, 12.1 and 12.2. Fragnesia is the second stage of the chain that begins with the AI-found PHP2Shell

flaws (CVE-2026-109618/109619). PHP fixed those on 21 October, and Palo Alto shipped PAN-OS hotfixes for them between 25 and 29 October. CISA warned on 28 October that the two stages were being chained to gain root. The kernel fix for the second stage therefore arrived about four weeks after that warning. Palo Alto rates the bulletin "Informational" and says devices already on a PHP2Shell-fixed build are protected, because it knows of no other exploitation path.

Why it reaches you. The component class is the management plane of network security appliances. The AI-accelerated cycle showed up here as two-stage patching: an urgent fix for the entry flaw, followed weeks later by a fix for the privilege escalation behind it. A team that closed the ticket after the October hotfix has a device that is safe only for as long as no second route to code execution appears. The "Informational" rating on a flaw the vendor itself says is exploited means it is unlikely to trigger an emergency change process unless someone reads it.

What to do. Network security owner: validate. Confirm that every PAN-OS firewall and Panorama is on a PHP2Shell-fixed build, schedule the PAN-SA-2026-0015 kernel builds, and verify that no management interface or dataplane interface with a management profile is reachable from the internet. Ask the vendor in writing whether the kernel fix covers Panorama.

CSA coverage. PHP2Shell: Exploitation Before Disclosure for an AI-Found Flaw

A command-gate test produced policy violations in all 16 coding-agent and model pairings tried, across eight production coding agents, because the gate and Bash read the same command differently

Category: agentic_surface **Significance:** notable **Confidence:** VERBATIM (PROVIDER) for "17 root causes" across "eight production coding agents", for live trials that "triggered policy violations in all 16 agent-model pairs (14 under configured restrictions)", and for the Codex CLI allow rule that "disables its read-only OS sandbox"; LINK ONLY — VERIFY AT SOURCE for vendor-by-vendor detail, which the preprint redacts pending a 90-day disclosure window; LIVE TEST REQUIRED for whether your agent versions and allow rules are affected **Source:** arXiv — ParseGap: Testing What Coding Agents Authorize and What Shells Execute

What changed. A preprint submitted on 22 November 2026 tested the command-approval gates of eight production coding agents against GNU Bash 5.2. It found 17 root causes where the gate approves one interpretation of a command and the shell executes another. Examples include a tree-sitter-bash defect that hides a second command from Mistral Vibe and Gemini CLI, and an allow rule in OpenAI Codex CLI that switches off its read-only OS sandbox. In live trials with two frontier models, every one of 16 agent-model pairs produced a policy violation, 14 of them under configured restrictions. Every checker also allowed "read-only" binaries that can write files or launch programs, in 9 to 47 of 55 cases. Vendors

were notified on 22 November under a 90-day window. The author's proposed fix runs approved commands directly via `execve` with no shell in between, and it admitted 0 of 4,174 test attacks at a median gate latency of 1.28 ms.

Why it reaches you. Allow-lists and "read-only command" rules are what make auto-approve modes in coding agents acceptable to security teams. This result shows that in deployed agents those rules are only as good as the gate's shell parser, and that a prompt-injected agent can get a write or execute operation past a rule written to permit reads. The exposure sits on developer workstations and CI runners where agents hold repository and cloud credentials. The fix belongs in the agent runtime: execute approved commands without a shell interpreter and confine the process at the OS level. Writing more allow-list entries does not close it.

What to do. Developer platform owner: validate. List which coding agents run with allow-listed or auto-approved shell commands, and remove interpreters, `find -exec`-style binaries and any rule that relaxes the OS sandbox from those allow-lists. Require OS-level confinement (filesystem and network) for any agent running with auto-approval until vendors publish fixes.

CSA coverage. GuardFall: Shell Injection Defeats AI Coding Agent Guardrails

The UK NCSC says blocking IP addresses is often ineffective against agentic attacks and advises rate limits on accounts, API tokens and sessions, plus deception

Category: security_operating_model **Significance:** notable **Confidence:** VERBATIM (PROVIDER) for "Since September 2026, the NCSC has seen a marked rise in AI-assisted cyber attacks", for "none of the weaknesses being exploited are new or novel", for movement "from discovery to exploitation in minutes", and for the advice to "apply rate limits to accounts, API tokens and authenticated sessions"; NO PROVIDER CLAIM on incident counts, sectors, or any patch deadline; CHARACTERIZATION (CSA) that the advice moves throttling and deception from optional to expected controls **Source:** National Cyber Security Centre – Familiar weaknesses, exploited faster: defending against agentic cyber attacks

What changed. On 24 November 2026 the UK NCSC reported a marked rise since September in incidents involving agentic offensive frameworks: ARTEX, plus Cairn, PentAGI, RedAmon, Xalgorix and Darkmoon. Some incidents caused service downtime or system compromise. The NCSC says the weaknesses exploited are technical debt, not new bugs: credentials committed to repositories, infrastructure-as-code misconfiguration, default or absent authentication, and API weaknesses. What has changed is the speed, persistence and cost of the attacks. Its eight questions for defenders include throttling account creation, bot challenges on identity and resource-heavy endpoints (LLM-backed

features among them), detecting "bursts of requests followed by short thinking phases", and honeytokens, decoys and tarpits to use up an agent's attempt budget. Unlike Switzerland's NCSC, it sets no patch deadline.

Why it reaches you. Agents rotate through proxies and VPNs, so control designs that key rate limits and blocks to IP addresses, common at API gateways and WAFs, stop working. The NCSC moves the control point to the identity: the account, the API token, the authenticated session. That is a design change for API gateways, customer identity platforms and sign-up flows. Its deception advice is aimed at the agent's attempt budget, which human-oriented honeypots were not designed around.

What to do. Application security lead: validate. For the five most exposed APIs and the customer sign-up flow, confirm that rate limits apply per account, per token and per session rather than only per IP, and seed at least one honeytoken credential in a code repository and one in an infrastructure-as-code store with an alert wired to first use.

CSA coverage. Autonomous Agentic AI Adversaries

Five of seven closed-weight models refused every unsafe exploit request but also refused 28% to 62% of safe defensive requests, while open-weight models complied with both

Category: defender_models **Significance:** notable **Confidence:** VERBATIM (PROVIDER) for "five of seven closed-weight models refused all unsafe requests, yet also refused 28% to 62% of safe requests" and for open-weight validated success of 62–68% on safe and 59–64% on unsafe requests; LINK ONLY — VERIFY AT SOURCE for per-model figures, including Opus 5.5 complying with 37.6% of safe requests and the 99.5% share of closed-model refusals coming from provider filters; LIVE TEST REQUIRED for refusal rates on your own triage and vulnerability-validation tasks **Source:** arXiv — CyberDualEval: Measuring Dual-Use Cyber Risks in Frontier Language Models

What changed. A benchmark submitted on 23 November 2026 by researchers at Columbia, UCLA and depthfirst splits 430 vulnerable tasks into analysis, proof-of-concept and exploit phases. Security experts rated each proof of concept by how much work would remain to weaponise it, and the benchmark scores whether a model helps with the low-risk phases and refuses the high-risk ones. Five of seven closed-weight models refused every unsafe request but also refused 28% to 62% of safe ones. The two that partly complied with unsafe requests could not reliably tell low-risk from high-risk requests. The three open-weight models rarely refused and had the highest validated success on both safe and unsafe work. The authors report that almost all closed-model refusals came from provider-side filters or model fallbacks, not from the model's own reasoning. They also note that each model was run only once.

Why it reaches you. Fitness and risk are separate axes, and this result shows default closed-model access losing up to six in ten legitimate defensive tasks (vulnerability analysis and low-risk proofs of concept) to filters the enterprise cannot tune. Teams that hit this tend to route work to open-weight models, as Hugging Face reportedly did after its July incident. That removes the provider's misuse controls along with the refusals. Neither outcome shows up in a vendor model card.

What to do. Security engineering lead: validate. Before standardising a model for triage or vulnerability validation, run 100 to 200 real defensive tickets through it on both the default and any defender-tier access, and record refusal rate and task success separately. Where the defender tier does not close the gap, document the decision and the containment controls before allowing open-weight substitution.

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

Watchlist

- **OpenAI reward-hacking postmortem — downstream response** — No change. *(opened 2026-08-27)*
- **VM/hypervisor containment hardening for cyber-capable agents** — No change in VM or microVM adoption. A related signal one layer up: ParseGap (item 4) reports an OpenAI Codex CLI allow rule that disables the agent's read-only OS sandbox, a reminder that sandbox configuration can be undone by policy settings without any escape. *(opened 2026-08-27)*
- **Claude Code Auto Mode prompt-injection ASR discrepancy** — No change. No Anthropic response to the reproduced attack success rate. ParseGap's command-gate findings, now in a 90-day disclosure window, bear on the same auto-approval surface, but the preprint redacts which agents had which root causes. *(opened 2026-08-27)*
- **AI defensive-triage guardrail evasion** — No change. No new campaign has used model refusals to blind triage tooling. CyberDualEval (item 6) documents the opposite failure: provider filters refusing legitimate defensive work without any attacker trying to cause it. *(opened 2026-08-31)*
- **AI account session hijacking at scale** — No change. *(opened 2026-08-31)*

Opened this issue

- **Coding-agent command-gate parsing fixes (ParseGap disclosure window)** — `agentic_surface`. Vendors were notified on 22 November 2026 under a 90-day

window, closing around 20 February 2027. Watching for vendor fixes and advisories for the 17 root causes, for whether agents move to shell-free execution of approved commands, and for the unredacted vendor-by-vendor detail at disclosure.