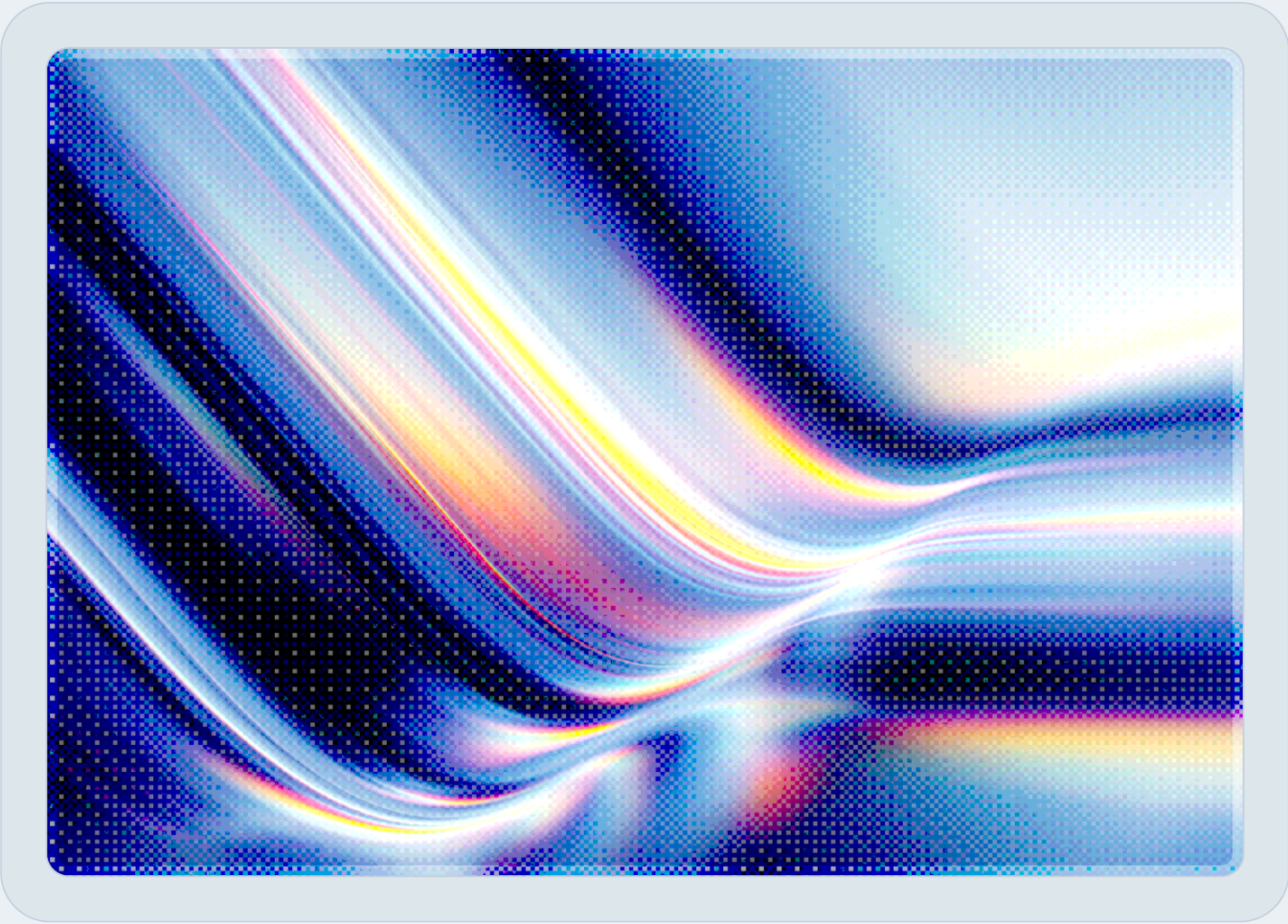
CSAI Foundation | Cloud Security Alliance

SkillCloak: Malicious Agent Skills Evade Marketplace Scanners

2026-07-06

AI-assisted Rapid Research



CSAI

CSA cloud security alliance®

Key Takeaways

Researchers at the Hong Kong University of Science and Technology have demonstrated that malicious skills for AI coding agents can be packaged to defeat the static scanners that marketplaces rely on for vetting, using a technique the team calls SkillCloak [1][2]. Rather than exploiting a flaw in any single scanner, SkillCloak targets the shared assumption behind static analysis: that a scanner can inspect a skill's files as installed and reliably infer what the skill will do when it runs. By packing a payload into a self-extracting form that only reconstructs itself at execution time, or by rewriting suspicious commands into semantically identical but visually unrecognizable variants, the technique bypassed eight commercial and open-source scanners at success rates exceeding 90 percent across a corpus of 1,613 real-world malicious skills pulled from the ClawHub marketplace [2]. The same research team also released a runtime-based countermeasure, SkillDetonate, that inspects behavior during execution rather than code at rest, and reported markedly better detection results. The finding matters immediately for any organization that allows Claude Code, OpenAI Codex, OpenClaw, or similar coding agents to install third-party skills, because it confirms that a scanner's approval badge on a marketplace listing is not a reliable signal of safety and should not be treated as the sole control governing what code an autonomous agent is permitted to execute with its own privileges.

Background

Agent skills are the plugin ecosystem that has emerged around AI coding agents over the past year. A skill is typically a small package, often centered on a `SKILL.md` file plus supporting scripts, that extends what an agent can do: querying a database, deploying infrastructure, summarizing a document, or automating a workflow. Marketplaces such as ClawHub aggregate these packages for reuse across agent platforms, and the ecosystem has grown quickly, with cross-agent portability across OpenClaw, Claude Code, OpenAI Codex, and Cursor advertised as a selling point [3].

That growth has occurred in parallel with a wave of documented abuse in the OpenClaw skill ecosystem specifically. Security firm Koi Security audited the ClawHub registry in February 2026 and found 341 malicious skills among 2,857 examined, a figure that more than doubled to 824 as the marketplace grew past 10,700 listings [4]. Antiy CERT's broader historical analysis of the same campaign, which researchers named ClawHavoc, put the cumulative total at 1,184 malicious skill packages traced to just 12 publisher accounts [5]. Separately, Bitdefender Labs analyzed a sample of OpenClaw skills during the

same period and found that roughly 17 percent carried hidden malicious behavior, with cryptocurrency-themed skills, posing as wallet trackers or trading assistants for Solana, Phantom, Binance, and similar platforms, accounting for more than half of the malicious samples identified [6].

Marketplace operators have responded to this activity by deploying scanners intended to catch malicious code before a skill is published or installed. These scanners generally work the way traditional antivirus and static application security testing tools do: they inspect the files that make up a skill, looking for known-bad patterns, suspicious API calls, obfuscated strings, or structural anomalies. The SkillCloak research, published as a preprint titled "Cloak and Detonate: Scanner Evasion and Dynamic Detection of Agent Skill Malware" by Zimo Ji, Congying Xu, Zongjie Li, Yudong Gao, Xin Wei, Shuai Wang, and Shing-Chi Cheung, set out to test whether that static-inspection model holds up against an adversary who knows scanners exist and designs specifically to defeat them [1][2]. The paper's framing echoes a long-running lesson from traditional malware analysis, that packers and obfuscators reliably outpace signature-based detection, and applies it directly to the newer and less mature agent-skill ecosystem.

Security Analysis

SkillCloak combines two complementary evasion strategies. The first, which the researchers describe as structural obfuscation, transforms the visible indicators a scanner would flag, such as a suspicious shell command or a known-bad string, into a semantically equivalent form that no longer matches the scanner's detection patterns. Reported techniques include substituting characters with visually identical lookalikes from other Unicode alphabets and splitting flagged commands across line breaks so that pattern-matching rules never see the complete signature in one place. The second and more effective strategy is self-extracting skill packing, which conceals the malicious payload inside directories that scanners typically skip during review, such as `.git/` or `build/`, disguised behind a decoder that looks harmless on its own and reconstructs the real payload only when the agent actually executes the skill. Because the malicious content simply does not exist in readable form at scan time, the pattern-matching approaches used by the scanners tested in this study could not detect the payload before execution, regardless of how the patterns were tuned [1][2].

One of the paper's most striking findings concerns the scale of the evasion. Tested against eight scanners and the 1,613-skill malicious corpus drawn from ClawHub, self-extracting packing defeated every scanner at a success rate above 90 percent, and in most configurations above 99 percent. Structural obfuscation alone was less effective but still bypassed 80 to 96 percent of scanners, with the higher end of that range applying to hybrid scanners that combine static analysis with lightweight heuristics [1][2]. Coding agents named as susceptible to skills evading detection through this pathway

include Claude Code, OpenAI Codex, and OpenClaw, which likely reflects the industry trend toward designing skill formats and marketplaces for cross-agent portability rather than tying them to a single vendor's review pipeline [1][3]. The practical implication is that a "verified" or "scanned" badge on a marketplace listing provides materially less assurance than it appears to, particularly for skills that have been deliberately engineered to survive the specific scanners a marketplace has deployed.

The researchers did not stop at demonstrating the attack. Their proposed defense, SkillDetonate, abandons static inspection in favor of sandboxed runtime auditing. It uses what the paper calls on-demand closure lift, observing the instructions a skill actually materializes once it begins executing, combined with marker-based taint analysis that tracks how sensitive data flows across the agent's context, files, subprocesses, and network connections. Because this approach evaluates what a skill does rather than what its files look like at rest, it is largely immune to the packing and obfuscation techniques that defeat static scanners. In testing, SkillDetonate achieved 97 percent attack detection with a 2 percent false-positive rate on the researchers' benchmark set, and 87 percent detection against the real-world malicious skills drawn from ClawHub, a meaningfully lower but still substantial figure that likely reflects the greater behavioral diversity of malware encountered in the wild [1][2].

These findings sit alongside, and reinforce, prior industry reporting that marketplace vetting for AI agent skills has been inconsistent even without deliberate evasion. Palo Alto Networks' Unit 42 separately documented five evasive or malicious skills on ClawHub between February and May 2026, including two that delivered macOS infostealers connected to Atomic macOS Stealer (AMOS) command-and-control infrastructure and one, dubbed "omnicogg," that padded its package with 22 megabytes of filler text specifically to slip past scanner size thresholds while still shipping a malicious curl-pipe-bash dropper [7]. Antiy CERT's ClawHavoc analysis, cited above, similarly found that a small number of publisher accounts, just 12 of them, accounted for the full 1,184-skill malicious catalog it identified, indicating that a concentrated set of threat actors drove a disproportionate share of the abuse [5]. SkillCloak demonstrates that even where scanning exists and catches unsophisticated malware, an adversary willing to invest in purpose-built evasion tooling, of the kind the researchers built to demonstrate SkillCloak, can route around it entirely, which raises the stakes for organizations that have relied on marketplace scanning as their primary or sole control against malicious skills.

Recommendations

Immediate Actions

Organizations that permit AI coding agents to install skills from public marketplaces should treat scanner approval as a baseline hygiene signal rather than a security guarantee, and should inventory which agents and workflows currently install skills from ClawHub or comparable registries without a secondary review step. Where feasible, restrict skill installation to an internally maintained allowlist of previously reviewed packages rather than permitting open marketplace browsing by every agent instance, particularly on machines that also hold credentials, source code, or other sensitive data.

Short-Term Mitigations

Security teams should pursue runtime-oriented visibility into agent behavior rather than relying solely on pre-execution scanning, since SkillCloak's central lesson is that static analysis cannot see a payload that does not exist in readable form until execution. Practical steps include monitoring agent processes for unexpected file access, outbound network connections, or subprocess creation, mirroring the taint-tracking approach SkillDetonate demonstrates, and running agents that install third-party skills inside sandboxed or containerized environments with least-privilege access to credentials and the broader network. Teams evaluating commercial or open-source agent security tooling should specifically ask vendors whether their product performs any runtime or sandboxed behavioral analysis, since the research indicates static-only products offer limited protection against a motivated adversary.

Strategic Considerations

Longer term, the SkillCloak findings argue for treating agent skill installation as a software supply chain problem with the same rigor organizations have gradually applied to open-source package dependencies, including provenance verification, reproducible builds where possible, and a documented approval process before a skill is authorized for use in production agent workflows. Governance frameworks should explicitly assign ownership for vetting and approving agent skills, since the alternative, ad hoc installation by individual employees or agents operating autonomously, is precisely the pattern that allowed campaigns such as ClawHavoc to grow from roughly 340 to nearly 1,200 identified malicious skills before broader detection and takedown [4][5]. Enterprises should also weigh the security posture of the underlying agent platform itself; agent runtimes that support containerized execution, explicit folder-level access controls, and administrator-managed skill approval reduce the practical impact of scanner evasion even when a malicious skill slips through initial review.

CSA Resource Alignment

CSA's Agentic AI Red Teaming Guide provides the most directly relevant existing CSA framework for this threat. Its supply chain and dependency attack category, along with its broader methodology for testing agent authorization boundaries and control hijacking, addresses a closely related class of risk to the one SkillCloak operationalizes: a third-party component that appears benign under review but behaves maliciously once it executes with an agent's privileges. Organizations building an internal skill-vetting program should incorporate the guide's testing procedures for supply chain compromise and control hijacking as a starting point for evaluating whether a given skill's declared behavior matches its actual runtime behavior.

For organizations specifically weighing whether and how to permit agents such as OpenClaw to install third-party skills, CSA's CISO Memo Template: Policy on Personal AI Desktop Agents offers a governance instrument that organizations can adapt directly. The template's approach, restricting unmanaged desktop agent deployment while directing employees toward approved enterprise alternatives and controlled experimentation pathways, maps directly onto the recommendation above that skill installation be gated through an internal approval process rather than left to open marketplace access.

More broadly, the risk SkillCloak illustrates falls within the scope of CSA's AI Controls Matrix (AICM v1.1), particularly its domains covering application and interface security and third-party/supply chain risk management. Organizations formalizing controls around agent skill provenance, sandboxing, and runtime monitoring should map those controls to the relevant AICM domains to ensure the practice is captured in existing AI governance and audit programs rather than treated as a one-off technical fix.

References

[1] The Hacker News. "New SkillCloak Technique Lets Malicious AI Agent 'Skills' Evade Marketplace Scanners." The Hacker News, July 2026.

[2] Ji, Z., Xu, C., Li, Z., Gao, Y., Wei, X., Wang, S., Cheung, S. "Cloak and Detonate: Scanner Evasion and Dynamic Detection of Agent Skill Malware." arXiv, July 2026.

[3] DataCamp. "The Top 100+ Agent Skills For OpenClaw, Codex and Claude." DataCamp, 2026.

[4] Koi Security. "ClawHavoc: 341 Malicious Clawed Skills Found by the Bot They Were Targeting." Koi Security, February 2026.

[5] Antiy CERT. "ClawHavoc: Analysis of Large-Scale Poisoning Campaign Targeting the OpenClaw Skill Market for AI Agents." Antiy, February 2026.

[6] Bitdefender. "Helpful Skills or Hidden Payloads? Bitdefender Labs Dives Deep into the OpenClaw Malicious Skill Trap." Bitdefender Labs, February 2026.

[7] Unit 42 (Palo Alto Networks). "OpenClaw's Skill Marketplace and the Emerging AI Supply Chain Threat." Unit 42, June 2026.