OpenClaw Security Report: January 2026

January 31, 2026 · by Mark Bartlett

Download this guide as PDF

> **Quick Answer:** January 2026 saw three high-severity CVEs, a supply chain attack that poisoned 12% of ClawHub, and 21,000+ exposed instances — all in five days. A single malicious link could steal your auth token and give an attacker full shell access, even on local-only installations. Update to version 2026.1.29 or later immediately if you haven't already.

Related: OpenClaw Security Guide · OpenClaw February 2026 Security · ClawHub Security Alert · Best OpenClaw Alternatives · OpenClaw Setup Guide

Contents

January 2026 was the month OpenClaw went from a niche AI agent project to the center of a multi-front security crisis. Three high-severity CVEs, a coordinated supply chain attack on ClawHub, and over 21,000 instances exposed to the public internet — all within the last five days of the month.

This report covers every confirmed security event from January 2026 with sources for each claim. For February's fixes, see our February 2026 security report.

Summary table

CVE / Event	Severity	CVSS	Affected Versions	Fix Version	Status	Disclosed
CVE-2026-25253	High	8.8	< 2026.1.29	2026.1.29	Resolved	Feb 1
CVE-2026-24763	High	8.8	< 2026.1.29	2026.1.29	Resolved	Feb 2
CVE-2026-25157	High	7.5	< 2026.1.29	2026.1.29	Resolved	Feb 4
CVE-2026-28458	High	7.5	2026.1.20 – 2026.2.0	2026.2.1	Resolved	Mar 5
ClawHavoc supply chain attack	Critical	N/A	All versions	N/A	Partially resolved	Feb 2
21,639 exposed instances	High	N/A	All versions	N/A	Ongoing	Jan 31

All three January CVEs were patched in the same release: **version 2026.1.29**. If you're running anything older, you're exposed to remote code execution via a single click.

CVE-2026-25253: WebSocket token theft

Severity	High (CVSS 8.8)
CWE	CWE-669: Incorrect Resource Transfer Between Spheres
Affected versions	All versions before 2026.1.29
Fix	Version 2026.1.29
Disclosed	February 1, 2026
Discovered by	DepthFirst researchers
Advisory	GHSA-g8p2-7wf7-98mq

What happened

OpenClaw's Control UI accepted a `gatewayUrl` query parameter from the URL without validation and automatically initiated a WebSocket connection to whatever address was specified. The authentication token was transmitted as part of the handshake.

A malicious webpage could redirect a user to the Control UI with a crafted `gatewayUrl` pointing to an attacker-controlled WebSocket server. The UI would connect automatically and leak the user's auth token. The attacker then used the stolen token to connect to the victim's local OpenClaw gateway, disable safety controls, and execute arbitrary commands.

The entire attack takes seconds. A public proof of concept demonstrates the full chain — one click to RCE.

Impact

This is the worst of the three January CVEs. Even local-only OpenClaw instances (never exposed to the internet) were vulnerable because the attack starts in the user's browser. The victim visits a malicious page, the page extracts the token, the attacker connects to `127.0.0.1:18789` on the victim's machine.

OpenClaw agents have shell access, file system access, and API key access by design. Once an attacker has the auth token, they have full control.

Sources: NVD · Hackers Arise · The Hacker News

CVE-2026-24763: Docker command injection

Severity	High (CVSS 8.8)
CWE	CWE-78: OS Command Injection
Affected versions	All versions before 2026.1.29
Fix	Version 2026.1.29
Disclosed	February 2, 2026
Advisory	GHSA-mc68-q9jw-2h3v
Patch commit	771f23d

What happened

OpenClaw's Docker sandbox execution mechanism did not safely handle the `PATH` environment variable when constructing shell commands. An authenticated user who could control environment variables could influence command execution within the container context.

The practical risk: if an attacker had any foothold — via a malicious ClawHub skill, for example — they could escalate from the sandboxed container to arbitrary command execution.

Impact

The Docker sandbox is OpenClaw's primary isolation mechanism. This vulnerability undermined it. Combined with the ClawHub supply chain attack happening the same week, attackers had a clear path: malicious skill installs into the sandbox, command injection breaks out of it.

Sources: NVD · SentinelOne · DailyCVE

CVE-2026-25157: SSH command injection

Severity	High (CVSS 7.5 NIST / 7.7 GitHub CNA)
CWE	CWE-78: OS Command Injection
Affected versions	All versions before 2026.1.29
Fix	Version 2026.1.29
Disclosed	February 4, 2026
Advisory	GHSA-q284-4pvr-m585

What happened

Two separate injection points in OpenClaw's SSH handling:

1. **`sshNodeCommand`**: Constructed a shell script without escaping the user-supplied project path in an error message. When the `cd` command failed, the unescaped path was interpolated into an `echo` statement, allowing arbitrary command execution on the remote SSH host.

2. **`parseSSHTarget`**: Did not validate that SSH target strings could not begin with a dash. An attacker-supplied target like `-oProxyCommand=...` would be interpreted as an SSH configuration flag rather than a hostname, allowing arbitrary command execution on the local machine.

Impact

Both vectors allow remote code execution. The `sshNodeCommand` vector executes on the remote host. The `parseSSHTarget` vector executes locally. Users who connected OpenClaw to remote machines via SSH were particularly exposed.

Sources: NVD · CVEDetails · GitHub Advisory

CVE-2026-28458: Browser Relay auth bypass

Severity	High (CVSS 7.5)
CWE	Missing Authentication
Affected versions	2026.1.20 through 2026.2.0
Fix	Version 2026.2.1
Disclosed	March 5, 2026

What happened

The Browser Relay extension's `/cdp` WebSocket endpoint at `ws://127.0.0.1:18792/cdp` did not require authentication tokens. Malicious websites could connect via loopback and access the Chrome DevTools Protocol, enabling them to steal session cookies and execute JavaScript in other browser tabs.

Impact

This is a different class of vulnerability from the other three — it requires the Browser Relay extension to be active but does not require any user interaction beyond visiting a malicious page. The attacker gets access to browser session data, not the OpenClaw agent itself, but that often includes credentials, tokens, and active sessions.

This CVE was introduced in version 2026.1.20 and was not fixed until 2026.2.1, meaning it was present throughout the entire January crisis period.

Source: RedPacket Security

ClawHavoc supply chain attack

Severity	Critical (no CVE — supply chain, not a code vulnerability)
First malicious skill	January 27, 2026
Disclosed	February 2, 2026 (Koi Security report)
Skills affected	341 confirmed malicious (of 2,857 total in registry at time of audit)
Status	Partially resolved — malicious skills removed, but vetting process remains weak

What happened

Starting January 27, 2026, a coordinated campaign — later tracked as ClawHavoc — began uploading malicious skills to ClawHub, OpenClaw's official skill marketplace. Koi Security audited all 2,857 skills in the registry and identified 341 malicious entries. Of those, 335 were traced to a single operation.

The malicious skills disguised themselves as popular tools (Kubernetes management, web search, project management) and used "ClickFix" social engineering: burying malicious instructions within lengthy documentation to trick users into running commands.

The primary payload on macOS was the Atomic macOS Stealer (AMOS), which harvested browser credentials, keychains, Telegram data, SSH keys, and cryptocurrency wallets. The campaign specifically targeted the kind of user who runs OpenClaw: technically proficient, often on always-on machines like Mac minis.

At the time, ClawHub's only requirement to publish a skill was a GitHub account at least one week old. No automated static analysis, no code review, no signing requirement.

Why it matters for January

The first malicious skills were uploaded on January 27 — the same week OpenClaw was rebranding and going viral. By January 31, when the CVE disclosures began, the malicious skills were already in circulation. Users who installed skills during this window were potentially compromised before anyone knew there was a problem.

Subsequent scans in February found the problem had grown to over 800 malicious skills across an expanded registry. See our ClawHub security alert for the full breakdown.

Sources: The Hacker News · eSecurity Planet · CyberPress

Exposed instances

Severity	High
Instances exposed	21,639 (Censys, Jan 31)
Status	Ongoing

What happened

Censys published their scan results on January 31, 2026, identifying 21,639 publicly exposed OpenClaw instances. The platform had grown from roughly 1,000 instances to over 21,000 in under a week as adoption exploded.

Despite OpenClaw's documentation recommending SSH tunnels or local-only access on TCP/18789, thousands of instances were deployed directly to the public internet. The highest concentrations were in the United States, China, and Singapore, with over 30% running on Alibaba Cloud.

Later scans by SecurityScorecard and independent researchers found higher numbers — 40,214 and 42,665 respectively — though these were published in February and used different detection methods.

The exposure problem compounded every CVE listed above. CVE-2026-25253 could attack local instances through the browser, but exposed instances didn't even need that step — they were directly reachable.

Sources: Censys · Infosecurity Magazine

Timeline

Date	Event
Jan 27	OpenClaw rebrands from Clawdbot/Moltbot. First ClawHavoc malicious skills uploaded to ClawHub.
Jan 27-31	OpenClaw adoption surges from ~1,000 to 21,000+ deployed instances.
Jan 29	OpenClaw releases version 2026.1.29, patching CVE-2026-25253, CVE-2026-24763, and CVE-2026-25157.

Date	Event
Jan 31	Censys publishes scan results: 21,639 exposed instances. ClawHavoc malicious skill uploads surge.
Feb 1	CVE-2026-25253 (WebSocket token theft) formally disclosed via NVD.
Feb 2	CVE-2026-24763 (Docker command injection) disclosed. Koi Security publishes ClawHub audit: 341 malicious skills.
Feb 4	CVE-2026-25157 (SSH command injection) disclosed.

The patch (2026.1.29) shipped on January 29, two to six days before the CVEs were formally published. This is responsible disclosure working correctly — the fixes were available before the exploits were public. The problem: most of those 21,000+ exposed instances were unlikely to have updated in the two-day window between the patch and the first disclosure.

What to do right now

If you're running OpenClaw

1. **Check your version.** Run `openclaw --version`. If it's below 2026.1.29, update immediately.

```
npm update -g openclaw
# or
brew upgrade openclaw-cli
```

1. **Audit installed skills.** List everything installed and check each against the ClawHub security advisories. Remove any skill you didn't explicitly choose and verify.

2. **Check your exposure.** Your gateway should not be accessible from the internet. Verify: `curl http://YOUR_PUBLIC_IP:18789` should timeout or refuse. If it responds, you're exposed.

3. **Rotate credentials.** If you were running a version older than 2026.1.29 at any point in January: rotate your API keys, browser sessions, and any credentials the agent had access to. CVE-2026-25253 could have stolen your auth token without leaving obvious traces.

4. **Enable authentication.** If you haven't already, set a gateway authentication token in your config. The exposed instance problem is worse for instances running without auth.

If you’re evaluating OpenClaw

Read our OpenClaw security guide before deploying. Consider lighter alternatives if you don’t need OpenClaw’s full feature set — several options avoid these architectural risks entirely.

The bigger picture

January 2026 exposed a fundamental tension in OpenClaw’s design: the agent needs broad system access to be useful, but that access makes every vulnerability a potential full compromise. A WebSocket CORS issue that would be medium-severity in a typical web app becomes high-severity when the endpoint controls an agent with shell access.

The ClawHub supply chain attack made it worse. Users were installing unvetted skills on an agent that could execute arbitrary commands. The marketplace had no meaningful vetting process. The combination of CVE-2026-25253 (steal the auth token), CVE-2026-24763 (escape the sandbox), and a malicious skill (initial payload) created a complete attack chain.

The OpenClaw team patched the core CVEs quickly — 2026.1.29 shipped before the vulnerabilities were public. But the architectural problems — plaintext API key storage, overly broad agent permissions, weak skill vetting — are structural. They weren’t fixed by a point release and many of them are still open as of March 2026.

For the ongoing story, see our February 2026 security report.

Related guides

- OpenClaw Security Guide — hardening your deployment
- OpenClaw Setup Guide — safe installation from scratch
- ClawHub Security Alert — deep dive on malicious skills
- Best OpenClaw Alternatives — lighter tools without these risks
- OpenClaw February 2026 Security — the next month’s fixes
- OpenClaw Token Optimization — cost control alongside security

Sources: NVD CVE-2026-25253, NVD CVE-2026-24763, NVD CVE-2026-25157, Censys, The Hacker News, Koi Security via The Hacker News, Hackers Arise, RedPacket Security, eSecurity Planet, CyberPress, Infosecurity Magazine

Source: https://insiderllm.com/guides/openclaw-security-report-january-2026/