INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-06-24 13:30 UTC

AI Agent Skill Marketplaces Are the New npm: ClawHub Malware Bypasses Automated Scanners With Novel Agentic Attack Techniques

THREAT CAMPAIGN | **CRITICAL** | CVSS 9.5

SCC Item ID	SCC-CAM-2026-0552
Type	Threat Campaign
Severity	CRITICAL
CVSS Base Score	9.5
Affected Products	OpenClaw AI agent platform, ClawHub skill marketplace, macOS systems, TradingView users
Published	2026-06-23T22:00:51+00:00
Discovery Source	Rss:T1 Threatintel

Executive Summary

Attackers published five malicious skills to ClawHub, the official marketplace for the OpenClaw AI agent platform, deploying macOS infostealers that evaded automated security scanners from February through May 2026. The threat is active: the command-and-control server continued receiving new skill deliveries more than three months after public disclosure, and OpenClaw's public deployment base grew significantly during the attack window, dramatically expanding the attack surface. Organizations using OpenClaw to automate business workflows face credential theft, data exfiltration, and supply chain compromise through a trust channel, the official skill marketplace, that most security programs do not yet monitor.

Technical Analysis

Unit 42 researchers identified five malicious skill packages on ClawHub, the OpenClaw AI agent skill marketplace, active between February and May 2026. No CVE is assigned. A CVSS base score of 9.5 was present in source data; this score is medium confidence pending full Unit 42 report verification. Relevant CWEs: CWE-829 (Inclusion of Functionality from Untrusted Control Sphere), CWE-494 (Download of Code Without Integrity Check), CWE-250 (Execution with Unnecessary Privileges), CWE-693 (Protection Mechanism Failure), CWE-77 (Command Injection). Attack vectors include macOS infostealer deployment via malicious skill packages, file-padding techniques to defeat scanner file-size thresholds defeating both VirusTotal and ClawScan screening, and two novel agentic attack techniques specific to AI agent skill marketplaces, details pending full Unit 42 report access. MITRE ATT&CK coverage: T1195.001 (Supply Chain Compromise:

Compromise Software Dependencies), T1059/T1059.004 (Command and Scripting Interpreter: Unix Shell), T1027.001 (Obfuscated Files or Information: Binary Padding), T1539 (Steal Web Session Cookie), T1555 (Credentials from Password Stores), T1566/T1566.002 (Phishing: Spearphishing Link), T1056 (Input Capture), T1041 (Exfiltration Over C2 Channel), T1071.001 (Application Layer Protocol: Web Protocols), T1102 (Web Service). Active C2: 91.92.242[.]30 (confirmed per Unit 42 report). Publisher account 'krajekisbtc' is linked to a Telegram-based crypto key exfiltration cluster. Affiliate injection operator domain: laosji[.]net (medium confidence pending full report access). Affected platforms: OpenClaw AI agent platform, ClawHub skill marketplace, macOS systems, TradingView users. No patch status confirmed; no official remediation advisory verified at time of writing.

Action Checklist

1. Step 1: Containment. Immediately audit all OpenClaw skill installations across your environment. Block outbound connections to 91.92.242[.]30 (confirmed C2 per Unit 42 report) at perimeter firewall and endpoint controls. Remove or quarantine any skills published by the account 'krajekisbtc'. Treat all skills sourced from ClawHub between February and May 2026 as potentially compromised until individually verified. If OpenClaw is deployed on macOS endpoints, isolate those systems pending investigation. Block DNS resolution and outbound HTTP/S to laosji[.]net (affiliate injection operator, medium confidence pending full report access).
2. Step 2: Detection. Search EDR and endpoint logs on macOS hosts running OpenClaw for: outbound connections to 91.92.242[.]30; DNS queries or HTTP requests to laosji[.]net; process execution chains spawned by OpenClaw skill runtimes invoking shell interpreters (T1059.004); file writes of anomalously large binaries consistent with padding obfuscation (T1027.001); access to macOS Keychain, browser credential stores, or cookie databases (T1555, T1539). Review SIEM for data exfiltration patterns over HTTP/S (T1071.001, T1041) from OpenClaw process contexts. Query AU-2/AU-12 log sources for skill installation events. Cross-reference installed skill package hashes against known-good registry if available from Unit 42 report (URL verification recommended before reliance).
3. Step 3: Eradication. Uninstall all ClawHub-sourced skills installed between February and May 2026 and re-verify necessity before reinstallation. Rotate all credentials, API keys, session tokens, and crypto wallet keys on macOS systems where malicious skills executed (T1555, T1539, T1056 coverage). Remove any persistence mechanisms established by skill runtimes. Apply CIS 2.3 (Address Unauthorized Software) process: treat unverified skills as unauthorized software pending registry validation. Enforce CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory) to include AI agent skill inventories as software assets. Implement CIS 5.4 (Restrict Administrator Privileges to Dedicated Administrator Accounts) on systems running OpenClaw to limit privilege available to compromised skill processes (NIST AC-6, Least Privilege).
4. Step 4: Recovery. Before restoring OpenClaw to production, implement a verified-safe skill allowlist. Confirm no C2 beaconing to 91.92.242[.]30 or laosji[.]net from restored systems over a minimum 72-hour monitoring window. Validate integrity of credentials rotated in Step 3 through account activity review (NIST AU-6, Audit Record Review). Re-enable OpenClaw deployments only after your organization establishes an internal skill vetting process equivalent to software dependency review. Monitor for reinfection indicators per detection guidance below.
5. Step 5: Post-Incident. This incident exposes a control gap: AI agent skill marketplaces are not covered by most existing software supply chain security programs. Extend CIS 7.1 (Establish and Maintain a Vulnerability Management Process) and CIS 2.1 (Establish and Maintain a Software Inventory) to explicitly

include AI agent skills and model plugins as managed software assets. Require code review or sandboxed execution for all third-party skills before production use. Consider NIST CM-5 (Access Restrictions for Change) or vendor-specific sandboxing controls (e.g., OpenClaw's own skill isolation features if available) as compensating measures pending CIS Controls update. Implement D3-UAP (User Account Permissions) to restrict what system resources OpenClaw skill processes can access. Establish threat intelligence subscription to monitor Unit 42 and peer feeds for ClawHub IOC updates. Brief development and AI/automation teams on supply chain risk in agentic AI ecosystems.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE
Escalation Criteria	Escalate to senior IR leadership, legal counsel, and data protection officer immediately if forensic analysis confirms that macOS Keychain data, browser session cookies, or crypto wallet keys were exfiltrated to 91.92.242.30 or laosji.net, triggering potential breach notification obligations under applicable data protection regulations (e.g., GDPR, CCPA, state breach notification laws) or if any affected OpenClaw deployment processed customer PII, financial account credentials, or regulated data; additionally escalate if the C2 server continues receiving beacons post-containment, indicating active attacker persistence beyond the five known malicious skills.
Recovery Notes	Re-enable OpenClaw deployments only after completing a hash-verified skill allowlist, confirming 72 consecutive hours of clean network telemetry with no DNS or HTTP/S activity to laosji.net or 91.92.242.30, and validating that all rotated credentials show no unauthorized access events in downstream service audit logs. Given that OpenClaw's deployment base expanded from 679 to over 31,000 instances in under two weeks, treat any unverified instance that was online between February and May 2026 as potentially compromised and require it to complete the full eradication and monitoring cycle before production restoration. Monitor for reinfection specifically by watching for new ClawHub skill installations, any re-appearance of 'krajekisbtc' publisher entries or lookalike account names, and anomalous OpenClaw process behavior involving shell interpreter spawning, for a minimum of 30 days post-recovery.
Forensic Artifacts	macOS Unified Log archive ('~/Library/Logs/' and system log via 'log collect') covering February–May 2026, specifically subsystem entries for OpenClaw skill runtime execution, Keychain access events (com.apple.securityd), and network connection events linking OpenClaw PIDs to 91.92.242.30 or laosji.net | OpenClaw skill package files at '~/.openclaw/skills/' including package manifests, publisher metadata JSON, and the skill binary payloads themselves — specifically any Mach-O binaries exceeding expected size thresholds consistent with the padding obfuscation technique used by the krajekisbtc-published skills to evade automated scanners | macOS Keychain dump and browser credential/cookie SQLite databases ('~/Library/Application Support/Google/Chrome/Default/Cookies', '~/Library/Application Support/Firefox/Profiles/*/cookies.sqlite') timestamped to identify which secrets were accessed during skill execution windows | LaunchAgent and LaunchDaemon plist files created or modified between February and May 2026 ('~/Library/LaunchAgents/', '/Library/LaunchDaemons/') that reference OpenClaw skill runtime paths, capturing any persistence mechanisms dropped by the infostealer payload post-execution | Full RAM image from affected macOS hosts captured pre-isolation (via osxpmem or equivalent), to recover decrypted C2 communication buffers, in-memory skill bytecode, and any credential material staged in heap memory by the OpenClaw skill runtime prior to exfiltration to laosji.net

Per-Action IR Details

Step 1: Containment — Immediately audit all OpenClaw skill installations across your environment. Block outbound connections to 91.92.242[.]30 at perimeter firewall and endpoint controls. Remove or quarantine any skills published by the account 'krajekisbtc'. Treat all skills sourced from ClawHub between February and May 2026 as potentially compromised until individually verified. If OpenClaw is deployed on macOS endpoints, isolate those systems pending investigation. Block DNS resolution and outbound HTTP/S to laosji[.]net.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.3 — Containment Strategy

Controls: NIST AC-4 (Information Flow Enforcement), CIS 4.4 (Implement and Manage a Firewall on Servers), CIS 4.5 (Implement and Manage a Firewall on End-User Devices), CIS 2.3 (Address Unauthorized Software)

Compensating: On macOS endpoints without EDR: run 'sudo lsof -i | grep -E "91\.92\.242\.30|laosji"' to identify active C2 connections before isolating. Use 'networksetup -setairportpower en0 off' or pull the Ethernet cable for immediate isolation on laptops. On the perimeter, if no enterprise firewall is available, push a hosts-file null-route ('0.0.0.0 laosji.net') via script to all macOS endpoints and add a static null route for 91.92.242.30 using 'sudo route add 91.92.242.30 127.0.0.1'. Enumerate installed OpenClaw skills via the OpenClaw CLI ('openclaw skills list --all') or by inspecting '~/.openclaw/skills/' on each macOS host; cross-reference publisher field against 'krajekisbtc'.

Evidence: BEFORE isolating any macOS host, capture: (1) full RAM image using osxpmem or 'sudo osxmemcompressor' to preserve in-memory skill runtime state and any decrypted C2 payloads; (2) active network connections via 'netstat -anp tcp' and 'lsof -i' output to document live C2 sessions to 91.92.242.30:443 or laosji.net; (3) running process tree via 'ps aux --forest' to document OpenClaw skill runtime child processes and any spawned shell interpreters; (4) OpenClaw skill runtime logs at '~/.openclaw/logs/' and '~/Library/Logs/OpenClaw/' before any skill removal; (5) macOS Unified Log stream snapshot via 'log collect --last 7d --output /tmp/unified_log.logarchive' to preserve skill installation and execution telemetry.

Step 2: Detection — Search EDR and endpoint logs on macOS hosts running OpenClaw for: outbound connections to 91.92.242[.]30; DNS queries or HTTP requests to laosji[.]net; process execution chains spawned by OpenClaw skill runtimes invoking shell interpreters (T1059.004); file writes of anomalously large binaries consistent with padding obfuscation (T1027.001); access to macOS Keychain, browser credential stores, or cookie databases (T1555, T1539). Review SIEM for data exfiltration patterns over HTTP/S (T1071.001, T1041) from OpenClaw process contexts. Query AU-2/AU-12 log sources for skill installation events. Cross-reference installed skill package hashes against known-good registry if available from Unit 42 report (URL verification recommended before reliance).

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.2 — Detection and Analysis

Controls: NIST AU-2 (Event Logging), NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST AU-12 (Audit Record Generation), CIS 8.2 (Collect Audit Logs)

Compensating: Without SIEM/EDR, deploy osquery on macOS hosts and run: 'SELECT pid, name, path, cmdline FROM processes WHERE name IN ("bash","zsh","sh","python3") AND parent IN (SELECT pid FROM processes WHERE name = "openclaw");' to detect shell interpreter spawning from OpenClaw skill runtimes. For Keychain access detection, query macOS Unified Log: 'log show --predicate "subsystem == \"com.apple.securityd\" AND category == \"KeychainManager\"" --last 7d'. For large padded binary detection, run 'find ~/.openclaw/skills/ -type f -size +50M -exec file {} \;' to flag anomalously oversized skill packages. Use Wireshark or tcpdump ('sudo tcpdump -w /tmp/openclaw_capture.pcap host 91.92.242.30 or host laosji.net') to capture any residual C2 traffic. Write a Sigma rule matching process parent 'openclaw' with child image endswith any of ['/bin/bash','/bin/zsh','/usr/bin/python3'].

Evidence: BEFORE concluding analysis and moving to eradication: preserve macOS Unified Log archive ('log collect --last 30d') covering the February–May 2026 window of malicious skill availability; export browser SQLite credential/cookie databases from '~/Library/Application Support/Google/Chrome/Default/Cookies' and equivalent

Firefox/Safari paths to establish what may have been exfiltrated; capture Keychain access audit entries via 'security dump-keychain'; collect OpenClaw skill package files from '~/.openclaw/skills/' including metadata JSON to extract publisher hashes for comparison; snapshot DNS cache via 'sudo killall -INFO mDNSResponder' and review '/var/log/system.log' for laosji.net resolution events.

Step 3: Eradication — Uninstall all ClawHub-sourced skills installed between February and May 2026 and re-verify necessity before reinstallation. Rotate all credentials, API keys, session tokens, and crypto wallet keys on macOS systems where malicious skills executed (T1555, T1539, T1056 coverage). Remove any persistence mechanisms established by skill runtimes. Apply CIS 2.3 (Address Unauthorized Software) process: treat unverified skills as unauthorized software pending registry validation. Enforce CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory) to include AI agent skill inventories as software assets. Implement CIS 5.4 (Restrict Administrator Privileges to Dedicated Administrator Accounts) on systems running OpenClaw to limit privilege available to compromised skill processes (NIST AC-6, Least Privilege).

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication

Controls: NIST AC-6 (Least Privilege), CIS 2.3 (Address Unauthorized Software), CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory), CIS 5.4 (Restrict Administrator Privileges to Dedicated Administrator Accounts)

Compensating: Without enterprise credential rotation tooling: generate a prioritized rotation list by cross-referencing macOS Keychain entries accessed during the February–May 2026 window (from Keychain audit logs preserved in Step 2) against known crypto wallet apps and browser-stored credentials. For persistence removal, audit LaunchAgents and LaunchDaemons: 'ls -la ~/Library/LaunchAgents/ /Library/LaunchAgents/ /Library/LaunchDaemons/' and cross-reference any plist added between February and May 2026 for references to OpenClaw skill paths. Remove OpenClaw skills via 'openclaw skills remove ' for each krajekisbtc-published package and then manually delete residual files under '~/.openclaw/skills/'. Restrict OpenClaw process permissions by wrapping its execution in a dedicated low-privilege macOS user account with no Keychain or Full Disk Access entitlements.

Evidence: BEFORE rotating credentials or removing persistence mechanisms: capture a full listing of LaunchAgent/LaunchDaemon plists with timestamps ('ls -laT ~/Library/LaunchAgents/') to document persistence artifacts; export the complete macOS Keychain in encrypted form ('security export -k ~/Library/Keychains/login.keychain-db -t all -f pkcs12 -o /tmp/keychain_backup.p12') to preserve evidence of what secrets were accessible; record all currently active session tokens and API keys in use by OpenClaw via its configuration files at '~/.openclaw/config.yaml' or equivalent before revoking them; document crypto wallet application data directories (e.g., '~/Library/Application Support/MetaMask/') for signs of unauthorized access or exfiltration prior to key rotation.

Step 4: Recovery — Before restoring OpenClaw to production, implement a verified-safe skill allowlist. Confirm no C2 beaconing to 91.92.242[.]30 or laosji[.]net from restored systems over a minimum 72-hour monitoring window. Validate integrity of credentials rotated in Step 3 through account activity review (NIST AU-6, Audit Record Review). Re-enable OpenClaw deployments only after your organization establishes an internal skill vetting process equivalent to software dependency review. Monitor for reinfection indicators per detection guidance below.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST AU-6 (Audit Record Review, Analysis, and Reporting), CIS 2.1 (Establish and Maintain a Software Inventory), CIS 2.2 (Ensure Authorized Software is Currently Supported), CIS 7.1 (Establish and Maintain a Vulnerability Management Process)

Compensating: Without enterprise allowlisting tooling: create a plaintext SHA-256 hash registry of approved OpenClaw skills ('shasum -a 256 ~/.openclaw/skills/*/*.whl > /etc/openclaw/approved_skills.sha256') and enforce it via a pre-execution shell wrapper that computes and compares hashes before OpenClaw loads any skill. For the 72-hour monitoring window without SIEM, schedule a cron job ('*/15 * * * * /usr/local/bin/check_c2.sh') that runs 'curl --max-time

5 --silent http://91.92.242.30' and 'nslookup laosji.net' and logs results with timestamps to a centralized syslog server. Review rotated account activity manually via cloud provider audit logs (AWS CloudTrail, GCP Audit Logs) or service provider dashboards for unauthorized access attempts using the old credentials.

Evidence: Before re-enabling OpenClaw in production: verify that no OpenClaw skill runtime files remain from the February–May 2026 malicious batch by re-running the hash comparison against the approved registry established above; confirm macOS Unified Log shows no laosji.net DNS resolution or connections to 91.92.242.30 for the full 72-hour window by running 'log show --predicate "eventMessage CONTAINS \"laosji\" OR eventMessage CONTAINS \"91.92.242.30\"" --last 3d'; validate that new skill package publisher signatures match OpenClaw's verified developer registry and not the 'krajekisbtc' account or any newly registered lookalike accounts.

Step 5: Post-Incident — This incident exposes a control gap: AI agent skill marketplaces are not covered by most existing software supply chain security programs. Extend CIS 7.1 (Establish and Maintain a Vulnerability Management Process) and CIS 2.1 (Establish and Maintain a Software Inventory) to explicitly include AI agent skills and model plugins as managed software assets. Require code review or sandboxed execution for all third-party skills before production use (NIST CM controls, no specific control ID mapped from knowledge base for skill sandboxing — no mapped control). Implement D3-UAP (User Account Permissions) to restrict what system resources OpenClaw skill processes can access. Establish threat intelligence subscription to monitor Unit 42 and peer feeds for ClawHub IOC updates. Brief development and AI/automation teams on supply chain risk in agentic AI ecosystems.

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity

Controls: CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 7.2 (Establish and Maintain a Remediation Process), CIS 2.1 (Establish and Maintain a Software Inventory), CIS 5.4 (Restrict Administrator Privileges to Dedicated Administrator Accounts), NIST AC-6 (Least Privilege)

Compensating: Without a commercial threat intelligence platform: subscribe to the free Unit 42 threat intelligence RSS feed and configure a daily digest alert filtered on keywords 'ClawHub', 'OpenClaw', 'krajekisbtc', and '91.92.242.30'. Author a YARA rule targeting the padding-obfuscated binary pattern (large null-byte or repeated-byte sections exceeding 40MB within ELF/Mach-O skill packages) and run it as a pre-install hook via a Git pre-commit or CI step in any internal skill packaging pipeline. Document OpenClaw skills formally in the software asset inventory using CIS 1.1 fields (asset name, version, publisher hash, install date, approval status) in a shared spreadsheet with monthly review cadence until a CMDB is available.

Evidence: For the post-incident lessons-learned record, preserve: the complete timeline of skill installation events from '~/.openclaw/logs/' and macOS Unified Log covering February–May 2026 to document the dwell time of malicious skills; a copy of the ClawHub marketplace listing pages for all five malicious skills (screenshotted and archived via Wayback Machine or wget) as evidence of the evasion techniques used against automated scanners; the network capture files from the 72-hour recovery monitoring window showing confirmed absence of C2 beaconing; and a sanitized incident summary (IOCs, TTPs, affected skill names, publisher account) formatted for sharing with OpenClaw's security team and relevant ISACs covering the AI/ML software supply chain sector.

Detection Guidance

Priority detection targets for environments running OpenClaw on macOS. Network: block and alert on all traffic to 91.92.242[.]30 (active C2, confirmed per Unit 42 report) and laosji[.]net (affiliate injection operator, medium confidence pending full report access). Monitor HTTP/S POST requests from OpenClaw process contexts to external IPs not in approved egress allowlist (T1071.001, T1041). Endpoint: on macOS hosts, alert on OpenClaw skill runtime processes spawning shell interpreters (bash, zsh, sh), maps to T1059.004. Alert on OpenClaw-related processes reading from ~/Library/Keychains, browser profile directories, or cookie stores, maps to T1555, T1539. Alert on skill package installs writing files larger than expected baseline (file-padding evasion, T1027.001), establish a size baseline from known-good skill packages. Identity: audit for credential use

anomalies on accounts active on macOS systems running OpenClaw, particularly API keys and crypto wallet keys consistent with krajekisbtc cluster's Telegram-based exfiltration pattern. Marketplace hygiene: cross-reference all installed ClawHub skills against publisher 'krajekisbtc' and any skill published February through May 2026 without independently verified provenance. Note: CVSS 9.5 and all technical specifics are not confirmed from verified source access; treat all IOCs and technical details as medium confidence pending full Unit 42 report verification.

Indicators of Compromise

Type	Value	Context	Confidence
IP	`91.92.242[.]30`	Active C2 server receiving new skill deliveries; confirmed active more than three months post-disclosure	HIGH
DOMAIN	`laosji[.]net`	Affiliate injection operator domain associated with ClawHavoc campaign	HIGH
URL	`krajekisbtc (publisher account identifier, not a URL)`	ClawHub marketplace publisher account linked to Telegram-based crypto key exfiltration cluster; treat all skills from this publisher as malicious	HIGH

Framework Mappings

MITRE-ATTACK

- **T1059.004** — Unix Shell
- **T1539** — Steal Web Session Cookie
- **T1027.001** — Binary Padding
- **T1555** — Credentials from Password Stores
- **T1566** — Phishing
- **T1059** — Command and Scripting Interpreter
- **T1195.001** — Compromise Software Dependencies and Development Tools
- **T1102** — Web Service
- **T1566.002** — Spearphishing Link
- **T1056** — Input Capture
- **T1041** — Exfiltration Over C2 Channel
- **T1071.001** — Web Protocols

NIST-800-53R5

- **CM-7** — Least Functionality
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **AT-2** — Literacy Training and Awareness

- **CA-7** — Continuous Monitoring
- **SC-7** — Boundary Protection
- **SI-8** — Spam Protection
- **SI-7** — Software, Firmware, and Information Integrity
- **CM-3** — Configuration Change Control
- **AC-6** — Least Privilege
- **SI-10** — Information Input Validation
- **SR-2** — Supply Chain Risk Management Plan

OWASP-TOP10-2021

- **A08:2021** — Software and Data Integrity Failures
- **A01:2021** — Broken Access Control
- **A03:2021** — Injection

CIS-V8

- **2.5** — Allowlist Authorized Software
- **2.6** — Allowlist Authorized Libraries
- **5.4** — Restrict Administrator Privileges to Dedicated Administrator Accounts
- **6.8** — Define and Maintain Role-Based Access Control
- **16.10** — Apply Secure Design Principles in Application Architectures
- **15.1** — Establish and Maintain an Inventory of Service Providers

NIST-CSF-2

- **GV.SC-01** — Cybersecurity supply chain risk management program

ISO-27001-2022

- **A.5.21** — Managing information security in the ICT supply chain
- **A.5.23** — Information security for use of cloud services

SOC2-TSC

- **CC9.2** — Manages risks associated with vendors and business partners

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1059.004	Unix Shell	Execution
T1539	Steal Web Session Cookie	Credential-Access
T1027.001	Binary Padding	Defense-Evasion
T1555	Credentials from Password Stores	Credential-Access
T1566	Phishing	Initial-Access

Technique ID	Technique Name	Tactic
T1059	Command and Scripting Interpreter	Execution
T1195.001	Compromise Software Dependencies and Development Tools	Initial-Access
T1102	Web Service	Command-And-Control
T1566.002	Spearphishing Link	Initial-Access
T1056	Input Capture	Collection
T1041	Exfiltration Over C2 Channel	Exfiltration
T1071.001	Web Protocols	Command-And-Control

Sources

Source	URL	Tier
Unit 42	https://unit42.paloaltonetworks.com/openclaw-ai-supply-chain-risk/	T3
	https://unit42.paloaltonetworks.com/openclaw-ai-supply-chain-risk/	T3
	https://www.techtarget.com/searchsecurity/tip/The-OpenClaw-security...	T3
	https://federalnewsnetwork.com/commentary/2026/05/mitigating-risk-f...	T3
[PDF] OpenClaw's Skill Marketplace and the Emerging AI Supply Chain ...	https://unit42.paloaltonetworks.com/openclaw-ai-supply-chain-risk/?...	T3

Generated 2026-06-24 13:30 UTC by TJS Security Command Center