INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-03-29 18:41 UTC

QuickLens Chrome Extension Supply Chain Compromise Delivers ClickFix, Crypto Wallet Theft, and Credential Harvesting to ~7,000 Users

THREAT CAMPAIGN | **HIGH** | CVSS 7.5

SCC Item ID	SCC-CAM-2026-0017
Type	Threat Campaign
Severity	HIGH
CVSS Base Score	7.5
Affected Products	Google Chrome (QuickLens extension, version unspecified); MetaMask, Phantom, Coinbase Wallet, Trust Wallet, Solflare, Backpack, Brave Wallet, Exodus, Binance Chain Wallet, WalletConnect, Argon Wallet (browser-based crypto wallets); Windows; macOS (AMOS stealer involvement unverified, confidence: low)
Published	2026-03-13

Executive Summary

The QuickLens Chrome extension was purchased via a marketplace and weaponized within 17 days, exposing approximately 7,000 users to credential theft, cryptocurrency wallet draining, and session hijacking before Google removed it. At least 11 browser-based crypto wallets were targeted, with Trust Wallet disclosing approximately $7 million in user losses attributable to a related campaign, though the precise causal link to this specific extension requires independent verification. Stolen seed phrases, credentials, and session tokens remain compromised indefinitely post-removal, as cryptographic material does not expire; users must treat all such secrets as fully exposed and rotate them.

Technical Analysis

The QuickLens Chrome extension (version unspecified) was compromised through a supply chain acquisition attack (MITRE T1195.001). After acquisition, the extension was modified to strip browser Content Security Policy headers (CWE-693), neutralizing inline XSS protections and enabling arbitrary JavaScript execution across all visited pages (CWE-79). A polling loop contacted a C2 server at 5-minute intervals (T1071.001) to retrieve modular payloads, enabling staged delivery of three distinct capabilities: (1) ClickFix social engineering lures prompting users to execute attacker-controlled commands (T1204.001, T1204.002, T1566); (2) targeted extraction of seed phrases and private key material from at least 11 named browser-based crypto wallets including MetaMask, Phantom, Coinbase Wallet, Trust Wallet, Solflare, Backpack, Brave Wallet, Exodus, Binance Chain Wallet, WalletConnect, and Argon Wallet (T1555.003, T1552.001); and (3) broad harvesting of

credentials and session tokens from browser storage (T1539, T1056.001, T1056.003). Payloads were delivered in obfuscated form (T1027) and exfiltrated over standard web protocols (T1041). The extension bypassed security controls at acquisition (unsigned or unverified marketplace delivery, CWE-494) and remained unsigned through subsequent payload delivery. Credentials were stored and transmitted without adequate protection (CWE-312). No CVE has been assigned. AMOS macOS stealer involvement has been claimed but is not independently verified in available primary sources, confidence: low. Threat actor attribution links to 'LLC Quick Lens' and the contact address support@doodlebuggle.top; no established threat group has been confirmed. Google has removed the extension from the Chrome Web Store and Chrome's auto-disable mechanism has deactivated installed copies.

Action Checklist

1. Step 1, Immediate: Identify any managed or employee-owned devices with QuickLens installed. Force-remove the extension if auto-disable has not already deactivated it. Verify removal by checking chrome://extensions and enterprise extension management logs.
2. Step 2, Immediate: Treat any user who had QuickLens installed as potentially compromised. Require immediate rotation of all browser-stored credentials, active session tokens, and any API keys accessible via the affected browser profile.
3. Step 3, Immediate (crypto wallets): Any user with a browser-based crypto wallet installed alongside QuickLens should treat their seed phrase and private keys as fully compromised. Wallet recovery requires generating a new wallet with a fresh seed phrase and transferring assets, not simply changing a password.
4. Step 4, Detection: Query endpoint and browser management logs for the extension ID associated with QuickLens. Search proxy and DNS logs for outbound connections to the identified C2 domain (doodlebuggle.top) and any subdomains. Flag 5-minute interval beaconing patterns from browser processes as a behavioral indicator.
5. Step 5, Assessment: Inventory which affected users have browser-based crypto wallets installed. Correlate QuickLens installation timelines against any anomalous account access, credential reuse events, or unauthorized transaction activity in identity and financial systems.
6. Step 6, Communication: Notify affected users directly with specific remediation steps. If your organization handles customer data accessible via affected browser sessions, assess breach notification obligations under applicable regulatory frameworks, verify with legal counsel.
7. Step 7, Long-term: Review browser extension governance policy. Implement allowlisting for approved extensions via enterprise browser management (e.g., Chrome Enterprise, group policy). Establish a review process for extension ownership changes in any approved extension inventory. Consider blocking unapproved extensions on managed endpoints.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE

Escalation Criteria	Escalate to executive leadership and legal counsel immediately if more than 10 users are confirmed to have installed QuickLens AND hold access to customer PII, financial data, or regulated systems (HIPAA, PCI-DSS, SOC 2 scopes); escalate to external incident response firm if any forensic analysis reveals confirmed data exfiltration to attacker infrastructure or unauthorized cryptocurrency transfers totaling >$10,000 per affected user.
Recovery Notes	Post-containment: (1) Retain forensic images of affected endpoints for 90 days pending law enforcement notification (FBI IC3 or local cybercrime task force); provide law enforcement access to blockchain transaction records for civil recovery support. (2) Implement post-incident monitoring: increase DNS/proxy alerting for 90 days on beaconing patterns, require monthly credential rotation audits for affected users, and establish monthly extension inventory reviews until remediation closure. (3) Conduct post-incident review: compare extension governance policy against industry benchmarks (CIS Controls v8, NIST CSF), document root cause (supply chain trust model failure), and update vendor risk assessment procedures to include extension marketplace monitoring or third-party security reviews before approved software additions.
Forensic Artifacts	Windows Event Viewer Security Log (Event ID 4688 for chrome.exe process creation, Event ID 4624 for logon activity, Event ID 4625 for failed logon attempts) | Chrome Extension database (%APPDATA%\Google\Chrome\User Data\Default\Extensions and Sync Data files) | Chrome Network Activity (chrome://net-internals/#events, HAR exports, or packet captures showing DNS/HTTPS to doodlebuggle.top) | Browser Password/Credential Storage (LoginData SQLite file, Local Storage and Session Storage contents) | DNS Client logs (Microsoft-Windows-DNS-Client/Operational Event ID 3008, or proxy/firewall DNS query logs for 90-day historical review) | Blockchain transaction history for affected wallet addresses (immutable public ledger records from chain explorers) | Identity/Financial System Audit Logs (failed password attempts, unusual account access, unauthorized transaction activity)

Per-Action IR Details

Step 1, Immediate: Identify any managed or employee-owned devices with QuickLens installed. Force-remove the extension if auto-disable has not already deactivated it. Verify removal by checking chrome://extensions and enterprise extension management logs.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.2.3 (containment strategies)

Controls: NIST 800-53 SI-2 (software updates and patches), CIS 4.1 (inventory of software assets), CIS 13.2 (removal of unapproved software)

Compensating: Without enterprise browser management: manually verify chrome://extensions on each device by screenshot (timestamp and username visible). Use PowerShell Get-Item 'HKCU:\Software\Google\Chrome\Extensions' to enumerate installed extension IDs across user profiles. Cross-reference against QuickLens ID (if known from vendor advisory). For BYOD devices, provide written removal instructions with screenshot verification requirement.

Evidence: Before removal, capture: (1) chrome://extensions page screenshot showing all extensions, version numbers, and enable/disable state for each user profile; (2) Windows Registry export of HKEY_LOCAL_MACHINE\Software\Policies\Google\Chrome\ExtensionInstallBlacklist and HKEY_CURRENT_USER\Software\Google\Chrome\Extensions; (3) macOS equivalent from ~/Library/Application Support/Google/Chrome/*/Extensions/; (4) browser cache and cookie exports (--user-data-dir flag location) before extension removal, as they may contain injected malicious scripts.

Step 2, Immediate: Treat any user who had QuickLens installed as potentially compromised. Require immediate rotation of all browser-stored credentials, active session tokens, and any API keys accessible via the affected browser profile.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.2.4 (eradication strategies for malware)

Controls: NIST 800-53 IA-4 (credential management), NIST 800-53 AC-7 (unsuccessful login attempts), CIS 5.2 (credential rotation policies)

Compensating: Without centralized credential management: (1) provide users a written checklist of high-value accounts (email, VPN, banking, corporate apps) with instructions to change passwords from a clean device; (2) export Chrome Saved Passwords using --show-password flag and audit for business-critical accounts manually; (3) instruct users to revoke active sessions in account settings (Gmail, Office 365, AWS, GitHub, etc.) by logging out of all sessions; (4) for API keys, query application logs for any API calls originating from the affected browser's IP/user-agent combo within 30 days prior.

Evidence: Before credential rotation, capture: (1) Chrome stored passwords export (requires admin access; use chrome://settings/passwords with export function or extract from LoginData SQLite DB in %APPDATA%\Google\Chrome\User Data\Default\); (2) browser history and download history (chrome://history) with focus on login pages and credential manager pages; (3) active session tokens from browser local/session storage (DevTools > Application > Local Storage and Session Storage); (4) any browser extensions that may have injected additional fields in login forms (content scripts in extension manifests).

Step 3, Immediate (crypto wallets): Any user with a browser-based crypto wallet installed alongside QuickLens should treat their seed phrase and private keys as fully compromised. Wallet recovery requires generating a new wallet with a fresh seed phrase and transferring assets, not simply changing a password.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.2.5 (recovery procedures for sensitive systems)

Controls: NIST 800-53 SC-28 (protection of information at rest), CIS 3.13 (secure key management)

Compensating: For non-technical users without formal recovery procedures: (1) provide step-by-step written instructions to export wallet transaction history from the compromised wallet BEFORE transferring assets (record all transaction IDs and timestamps for forensic analysis); (2) instruct users to create new wallet on a separate clean device/browser profile, generate new seed phrase (screenshot and securely store offline); (3) transfer all remaining assets from old wallet to new wallet via blockchain transaction; (4) document old wallet addresses and all transaction activity for potential law enforcement reporting and civil recovery.

Evidence: Before any wallet action, capture: (1) wallet browser extension storage (Chrome Web Storage SQLite from Level DB in %APPDATA%\Google\Chrome\User Data\Default\Local Storage\); (2) blockchain transaction history for all affected wallet addresses (query blockchain explorer with user consent); (3) browser DevTools console logs and Network tab traffic showing any API calls to attacker C2 during QuickLens active period; (4) wallet backup files or seed phrase storage locations (encrypted or plaintext) if accessible on endpoint.

Step 4, Detection: Query endpoint and browser management logs for the extension ID associated with QuickLens. Search proxy and DNS logs for outbound connections to the identified C2 domain (doodlebuggle.top) and any subdomains. Flag 5-minute interval beaconing patterns from browser processes as a behavioral indicator.

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.1.2 (analysis of indicators of compromise)

Controls: NIST 800-53 SI-4 (information system monitoring), NIST 800-53 CA-7 (continuous monitoring), CIS 13.1 (centralized endpoint log collection)

Compensating: Without centralized logging: (1) manually query Windows Event Viewer on each endpoint for DNS client events (Event ID 3008 in Microsoft-Windows-DNS-Client/Operational, filter for doodlebuggle.top and subdomains); (2) use netsh winsock show catalog and netsh trace start scenario=InternetClient for packet capture of live DNS/HTTP traffic; (3) audit browser proxy settings via Group Policy reports or manual registry inspection (HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Internet Settings\ProxyServer); (4) parse browser cache and history using sqlite3 CLI on Chrome databases (History, Cache, Cookies files); (5) use free tool such as Wireshark to capture live network traffic from affected systems and filter for doodlebuggle.top beacons.

Evidence: Capture: (1) Windows Event Viewer DNS logs (Microsoft-Windows-DNS-Client/Operational) for 60 days prior, filtered for doodlebuggle.top and wildcard subdomains; (2) proxy logs if available (firewall, web gateway, or local

system firewall logs showing outbound HTTPS/HTTP to doodlebuggle.top); (3) Chrome network activity dumps from DevTools (save as HAR file) for any user session during active QuickLens period; (4) system firewall logs (Windows Defender Firewall logs in Event Viewer, Event ID 5157 for blocked outbound); (5) browser cache files (indexed.db, Cache files) from %APPDATA%\Google\Chrome\User Data\Default\Cache.

Step 5, Assessment: Inventory which affected users have browser-based crypto wallets installed. Correlate QuickLens installation timelines against any anomalous account access, credential reuse events, or unauthorized transaction activity in identity and financial systems.

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.1.3 (correlation and impact analysis)

Controls: NIST 800-53 IR-4 (incident investigation), NIST 800-53 AU-12 (audit log generation), CIS 8.2 (privileged account monitoring)

Compensating: Without advanced analytics tools: (1) manually inventory installed extensions from Step 1 data; cross-reference against known wallet extension IDs (MetaMask ID: nkbihfbeogaeaoehlefnkodbefgpgknn, etc.); (2) correlate QuickLens install date (from registry or extension management logs) against user login logs (Windows Event ID 4624, 4648 for logon, network logon); flag logins from new IPs or unusual hours as anomalous; (3) query identity system (Active Directory) for failed password attempts (Event ID 4625) or credential changes (Event ID 4724) within 48 hours post-install; (4) for financial systems, manually review transaction logs for transfers to unknown addresses or unusual transaction amounts/frequencies; use blockchain explorers (Etherscan, etc.) to query wallet address activity and flag transfers during active compromise window.

Evidence: Capture: (1) Windows Security Event Log (System and Security channels) covering 90 days: Event ID 4688 (process creation, chrome.exe and extensions processes), Event ID 4689 (process termination), Event ID 4776 (credential validation); (2) browser extension installation timestamps (from registry or Chrome sync logs if available); (3) identity system audit logs (AD logon, password change, group membership changes); (4) application-layer logs from any financial or identity systems showing account access, API calls, or transaction initiation; (5) blockchain transaction records for affected wallet addresses (immutable, preserved on-chain).

Step 6, Communication: Notify affected users directly with specific remediation steps. If your organization handles customer data accessible via affected browser sessions, assess breach notification obligations under applicable regulatory frameworks, verify with legal counsel.

NIST Phase: Post Incident

Reference: NIST 800-61r3 §3.3 (post-incident activities and lessons learned)

Controls: NIST 800-53 IR-6 (incident reporting), CIS 17.1 (breach notification procedures)

Compensating: Without legal department resources: (1) consult published state/federal breach notification laws (NIST SP 800-188 provides guidance; www.ncsl.org maintains current state breach notification statutes); (2) determine scope: if customer PII was accessible (e.g., email logins, saved passwords containing corporate accounts), prepare breach notification within 30-60 days; (3) document internal assessment showing: date of discovery, affected user count, data types exposed, remediation steps taken, no evidence of actual exfiltration (if applicable); (4) notify affected parties via direct communication (email or registered mail) with clear remediation steps, option to enroll in credit monitoring if SSN or financial data at risk.

Evidence: Preserve: (1) timeline document showing QuickLens install date, detection date, removal date, notification date (supports demonstration of reasonable diligence); (2) user notification log with timestamps and delivery confirmation; (3) assessment documentation showing what data types were theoretically accessible (not required to have been stolen); (4) any communication from affected users reporting unauthorized activity (evidence of actual harm vs. theoretical risk).

Step 7, Long-term: Review browser extension governance policy. Implement allowlisting for approved extensions via enterprise browser management (e.g., Chrome Enterprise, group policy). Establish a review process for extension ownership changes in any approved extension inventory. Consider blocking unapproved extensions on managed endpoints.

NIST Phase: Preparation

Reference: NIST 800-61r3 §3.1 (preparation phase: processes and tools)

Controls: NIST 800-53 CM-7 (least functionality / software whitelisting), NIST 800-53 SI-7 (software integrity monitoring), CIS 4.1 (authorized software inventory), CIS 13.2 (removal of unapproved software)

Compensating: Without enterprise browser management tools: (1) publish approved extension list (document IDs, publishers, business justification); mandate manual review before installation; (2) use Windows Group Policy to set Chrome extension blocklist via registry (gpedit.msc > Computer Configuration > Administrative Templates > Google > Google Chrome > Extensions; block all except whitelist); (3) for macOS, use MDM profile or LaunchAgent to enforce extension policy (~/Library/LaunchAgents/com.apple.Chrome.json); (4) establish quarterly extension audit: query all installed extensions, compare against approved list, require business owner approval for any new extensions; (5) track extension publisher ownership changes using Chrome Web Store API (requires monitoring or manual periodic review) and flag ownership transfers as high-risk (common supply chain attack vector).

Evidence: Document: (1) approved extension inventory with extension IDs, publisher names (capture from Web Store), install date, business justification, and last reviewed date; (2) policy baseline (screenshot of GPO settings or MDM profile); (3) audit logs showing enforcement (Group Policy event logs, MDM compliance reports, or manual audit results); (4) any blocked extension attempts (Windows Event ID 4648 if captured) or user reports of blocked installations.

Detection Guidance

Primary indicators: (1) Extension presence, search endpoint management and Chrome enterprise reports for the QuickLens extension ID. Chrome extension IDs are stable; identify the specific ID from vendor disclosures or Chrome Web Store removal notices and query against your fleet. (2) C2 beaconing, search proxy, firewall, and DNS logs for connections to doodlebuggle.top and any subdomains. A 5-minute polling interval is a behavioral signature; look for regular outbound HTTP/HTTPS requests from browser processes at consistent intervals. (3) CSP header stripping, if you have web application firewall or proxy logs that record response headers, look for sessions where CSP headers are absent on sites that normally send them, this may indicate the extension was active. (4) ClickFix execution artifacts, on Windows endpoints, review PowerShell execution logs (Event ID 4104) and clipboard manager history for attacker-formatted command strings. ClickFix lures typically instruct users to paste commands into Run dialogs or terminals. (5) Credential and session anomalies, review identity provider logs for logins from unusual geolocations or devices within the window the extension was installed. Flag session token reuse from new user agents or IPs. Note: source quality for this campaign is assessed at 0.48, primary technical indicators should be validated against vendor disclosures as they become available. IOC confidence is variable; treat low-confidence indicators as hypotheses requiring corroboration.

Indicators of Compromise

Type	Value	Context	Confidence
DOMAIN	`doodlebuggle.top`	C2 domain associated with threat actor contact address (support@doodlebuggle.top) and likely C2 infrastructure for the compromised extension's polling loop. Treat connections to this domain and all subdomains as malicious.	MEDIUM

Type	Value	Context	Confidence
URL	`https://chrome.google.com/webstore, QuickLens extension listing (removed)`	Extension has been removed from the Chrome Web Store. The extension ID should be used to query endpoint management systems for installed instances. Specific extension ID not confirmed in available sources, obtain from vendor disclosures before querying.	LOW

Framework Mappings

MITRE-ATTACK

- **T1071.001** — Web Protocols
- **T1204.001** — Malicious Link
- **T1176** — Software Extensions
- **T1059.007** — JavaScript
- **T1555.003** — Credentials from Web Browsers
- **T1059.001** — PowerShell
- **T1041** — Exfiltration Over C2 Channel
- **T1027** — Obfuscated Files or Information
- **T1539** — Steal Web Session Cookie
- **T1056.001** — Keylogging
- **T1195.001** — Compromise Software Dependencies and Development Tools
- **T1204.002** — Malicious File
- **T1566** — Phishing
- **T1056.003** — Web Portal Capture
- **T1552.001** — Credentials In Files
- **T1557** — Adversary-in-the-Middle

NIST-800-53R5

- **CM-7** — Least Functionality
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **SI-7** — Software, Firmware, and Information Integrity
- **CA-7** — Continuous Monitoring
- **SC-7** — Boundary Protection
- **AT-2** — Literacy Training and Awareness
- **SI-8** — Spam Protection
- **CM-3** — Configuration Change Control
- **SI-10** — Information Input Validation

- **SR-2** — Supply Chain Risk Management Plan

OWASP-TOP10-2021

- **A08:2021** — Software and Data Integrity Failures
- **A03:2021** — Injection

CIS-V8

- **2.5**
- **2.6**
- **16.10**
- **6.3** — Require MFA for Externally-Exposed Applications
- **14.2** — Train Workforce Members to Recognize Social Engineering Attacks
- **15.1** — Establish and Maintain an Inventory of Service Providers

ISO-27001-2022

- **A.8.28** — Secure coding
- **A.5.21** — Managing information security in the ICT supply chain

HIPAA-SECURITY

- **164.312(d)** — Person or Entity Authentication

SOC2-TSC

- **CC6.1** — Logical access security software, infrastructure, and architectures
- **CC9.2** — Manages risks associated with vendors and business partners

NIST-CSF-2

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

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1071.001	Web Protocols	Command-And-Control
T1204.001	Malicious Link	Execution
T1176	Software Extensions	Persistence
T1059.007	JavaScript	Execution
T1555.003	Credentials from Web Browsers	Credential-Access
T1059.001	PowerShell	Execution
T1041	Exfiltration Over C2 Channel	Exfiltration
T1027	Obfuscated Files or Information	Defense-Evasion

Technique ID	Technique Name	Tactic
T1539	Steal Web Session Cookie	Credential-Access
T1056.001	Keylogging	Collection
T1195.001	Compromise Software Dependencies and Development Tools	Initial-Access
T1204.002	Malicious File	Execution
T1566	Phishing	Initial-Access
T1056.003	Web Portal Capture	Collection
T1552.001	Credentials In Files	Credential-Access
T1557	Adversary-in-the-Middle	Credential-Access

Sources

Source	URL	Tier
Security News	https://www.bleepingcomputer.com/news/security/quicklens-chrome-ext...	T3
Crypto Security Warning: Trust Wallet Confirms $7 Million Chrome ...	https://www.forbes.com/sites/daveywinder/2025/12/28/crypto-security...	T3
Trust Wallet Chrome Extension Breach Caused $7 Million Crypto ...	https://thehackernews.com/2025/12/trust-wallet-chrome-extension-bug...	T3
TRUST WALLET HACK: $7M STOLEN: HERE'S WHAT YOU NEED ...	https://www.binance.com/en/square/post/34240018166442	T3
Trust Wallet Browser Extension v2.68 Incident: An Update to Our ...	https://trustwallet.com/blog/announcements/trust-wallet-browser-ext...	T3

Generated 2026-03-29 18:41 UTC by TJS Security Command Center