INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-06-30 14:47 UTC

Fake Perplexity Extension Captured Every Address Bar Keystroke Before Users Pressed Enter

SECURITY ANALYSIS | HIGH | CVSS 5.0

SCC Item ID	SCC-STY-2026-0307
Type	Security Analysis
Severity	HIGH
CVSS Base Score	5.0
Affected Products	Google Chrome (extension ecosystem), users of Perplexity AI-branded extension
Published	2026-06-29T14:40:09
Discovery Source	Rss

Executive Summary

A malicious Chrome extension impersonating the Perplexity AI brand captured every keystroke typed in the browser address bar in real time, routing that data to an attacker-controlled server before users ever pressed Enter, according to Microsoft's Defender Security Research team. This technique goes beyond prior AI-branded extension attacks by intercepting input pre-submission, meaning credentials, search terms, and sensitive queries could be harvested before any network request was initiated by the user. The campaign signals a deliberate escalation in browser-based data collection tradecraft, exploiting consumer trust in AI brand names to bypass user scrutiny of extension permissions.

Technical Analysis

Microsoft's Defender Security Research team identified and disclosed the extension in June 2026, with coordinated removal from the Chrome Web Store. The extension impersonated Perplexity AI, a recognized AI search brand, applying a masquerading technique (MITRE T1036.005) to reduce user suspicion at install time. Once installed, the extension registered a listener against the Chrome address bar input field, capturing keystrokes character by character in real time before the user submitted a query (T1056.001, keylogging at the browser layer). Captured data was routed to an attacker-controlled server over standard web protocols (T1071.001, T1041), after which users were transparently redirected to legitimate search results, suppressing any visible anomaly that might prompt uninstallation. Server-side logging code, reviewed by Microsoft, confirmed deliberate data collection rather than incidental telemetry drift. The infrastructure relied on a web service acquired for this purpose (T1583.006). The adversary-in-the-middle positioning (T1557) enabled

interception of input that would never traverse standard network monitoring controls, since the data was exfiltrated before the browser initiated any legitimate outbound request. Microsoft had documented a related pattern in March 2026 (https://www.microsoft.com/en-us/security/blog/2026/03/05/malicious-ai-assistant-extensions-harvest-llm-chat-histories/), when a separate campaign harvested LLM chat histories from AI assistant extensions; the June campaign represents a technical escalation from post-submission to pre-submission interception. The defensive gaps exploited here are structural: Chrome's extension permission model does not require elevated or unusual permissions for address bar input access in all contexts, and enterprise browser management policies frequently lack controls that distinguish legitimate extensions from impersonators. CWE-356 (missing protection against unauthorized address bar access), CWE-441 (unintended proxy behavior routing data through attacker infrastructure), and CWE-923 (improper restriction of communication channel to intended endpoints) all apply. Attribution to a specific threat actor is not confirmed in available source material.

Action Checklist

1. Step 1: Assess exposure, audit all Chrome extensions installed across managed endpoints; identify any extension referencing 'Perplexity', 'AI search', or similar AI-branded naming, and cross-reference against the Chrome Web Store removal notice from Microsoft's June 2026 disclosure
2. Step 2: Review controls, verify that browser management policies (via Google Admin Console or equivalent MDM) enforce an allowlist of approved extensions; confirm NIST AC-3 (Access Enforcement) and AC-4 (Information Flow Enforcement) are implemented at the browser layer, and that CIS 2.3 (Address Unauthorized Software) and CIS 2.1 (Establish and Maintain a Software Inventory) cover browser extensions as software assets
3. Step 3: Update threat model, add AI-brand-impersonation browser extensions as a tracked TTP pattern in your threat register, citing MITRE T1036.005 (Masquerading: Match Legitimate Name or Location), T1056.001 (Keylogging), and T1041 (Exfiltration Over C2 Channel); reference Microsoft's March 2026 and June 2026 disclosures as a documented campaign cluster
4. Step 4: Communicate findings, brief leadership on the credential pre-interception risk: any employee who installed an AI-branded Chrome extension outside an approved software process may have had address bar input, including internal URLs, credentials typed in the address bar, and sensitive search queries, exfiltrated to an unknown third party
5. Step 5: Monitor developments, track Microsoft's Defender Security Research blog and the Chrome Web Store policy feed for follow-up disclosures on this campaign cluster; watch for additional AI-branded extensions applying the same pre-submission keylogging pattern

IR / Forensic Enrichment

Triage Priority	URGENT
Escalation Criteria	Escalate to CISO and legal/privacy counsel immediately if Step 1's audit confirms installation on endpoints with access to regulated data (PII, PHI, PCI-scoped systems, or privileged administrative credentials), as pre-submission keystroke capture of credentials or sensitive queries may constitute a reportable data breach under GDPR Article 33, HIPAA Breach Notification, or applicable state privacy statutes, independent of whether exfiltration to the C2 server can be confirmed.

Recovery Notes	After the malicious extension is removed and affected credentials are rotated, verify that Chrome's extension install policies from Step 2 are enforced and confirmed via `chrome://policy` on representative endpoints before returning affected users to normal operation. Monitor Sysmon Event ID 3 network connections from `chrome.exe` on previously affected endpoints for 30 days post-remediation, specifically watching for outbound connections to any domain identified in the Microsoft June 2026 disclosure or to newly registered domains matching AI-brand naming patterns, as campaign actors frequently pivot infrastructure after public disclosure. If credential rotation was required for privileged accounts, enable enhanced audit logging (Windows Security Event ID 4624, 4648) on those accounts for 60 days to detect any use of harvested credentials that may have already been operationalized by the threat actor prior to discovery.
Forensic Artifacts	Chrome extension directory on affected endpoints — `%LOCALAPPDATA%\Google\Chrome\User Data\Default\Extensions\\` (Windows) or `~/Library/Application Support/Google/Chrome/Default/Extensions//` (macOS) — containing `manifest.json` (showing declared API permissions such as `tabs`, `webNavigation`, or `webRequest` that enabled address bar interception) and background JavaScript files containing the keystroke capture and exfiltration logic | Chrome `Preferences` and `Local State` JSON files in the Chrome profile directory, which record the extension install timestamp, install source (Web Store vs. sideloaded), and whether the extension was force-installed or user-installed — critical for bounding the credential exposure window | DNS query logs and proxy/firewall logs for the attacker-controlled C2 domain(s) identified in the Microsoft June 2026 Defender disclosure, queried across the full log retention period from affected endpoint IP addresses to determine whether exfiltration connections were successfully established and to estimate data volume transmitted | Sysmon Event ID 3 (Network Connection Detected) logs on affected Windows endpoints filtering on `chrome.exe` as source process and non-Google destination IPs or domains, timestamped against the extension install period — this is the primary network-layer evidence that the pre-submission keylogging payload was actively exfiltrating data | Browser history and typed URL cache (`%LOCALAPPDATA%\Google\Chrome\User Data\Default\History` SQLite database, table `urls` and `keyword_search_terms`) for the exposure window — while the attacker captured keystrokes before submission, this database confirms what URLs and queries were typed on the affected endpoint and provides a victim-side record of what data categories were at risk

Per-Action IR Details

Step 1: Assess exposure — audit all Chrome extensions installed across managed endpoints; identify any extension referencing 'Perplexity', 'AI search', or similar AI-branded naming, and cross-reference against the Chrome Web Store removal notice from Microsoft's June 2026 disclosure

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.2 — Detection and Analysis: scope the affected population by identifying all endpoints where the malicious Perplexity-branded extension was installed, establishing blast radius before containment decisions are made

Controls: CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory), CIS 2.1 (Establish and Maintain a Software Inventory)

Compensating: On Windows endpoints without MDM, run: `reg query 'HKLM\SOFTWARE\Google\Chrome\Extensions' /s` and `reg query 'HKCU\SOFTWARE\Google\Chrome\Extensions' /s` to enumerate installed extension IDs, then cross-reference against the extension ID published in Microsoft's June 2026 Defender blog disclosure. On macOS, query `~/Library/Application Support/Google/Chrome/Default/Extensions/` via `ls -la`. A two-person team can script this as a PowerShell or bash loop across a host list exported from Active Directory or your asset inventory.

Evidence: Before any remediation action, capture the full Chrome extension manifest and background script from the compromised profile directory: on Windows `%LOCALAPPDATA%\Google\Chrome\User Data\Default\Extensions\\`, on macOS `~/Library/Application Support/Google/Chrome/Default/Extensions//`. Preserve the `manifest.json` (which will show declared `chrome.webRequest` or `chrome.tabs` permissions enabling address bar keystroke capture) and any background JS files containing the keylogging or exfiltration logic. Also capture Chrome's `Local State` and `Preferences` JSON files, which log extension install timestamps and sources — critical for determining initial infection date. Capture these files before the extension is removed.

Step 2: Review controls — verify that browser management policies (via Google Admin Console or equivalent MDM) enforce an allowlist of approved extensions; confirm NIST AC-3 (Access Enforcement) and AC-4 (Information Flow Enforcement) are implemented at the browser layer, and that CIS 2.3 (Address Unauthorized Software) and CIS 2.1 (Establish and Maintain a Software Inventory) cover browser extensions as software assets

NIST Phase: Preparation

Reference: NIST 800-61r3 §2 — Preparation: verify that preventive browser management controls exist and are enforced before assessing whether this specific extension could have been blocked at install time, informing both immediate containment and long-term hardening

Controls: NIST AC-3 (Access Enforcement), CIS 2.1 (Establish and Maintain a Software Inventory), CIS 2.3 (Address Unauthorized Software)

Compensating: Without Google Workspace or Intune, enforce Chrome extension allowlisting via Group Policy on Windows: set `HKLM\SOFTWARE\Policies\Google\Chrome\ExtensionInstallAllowlist` with approved extension IDs and set `ExtensionInstallBlocklist` to `*` to block all unlisted extensions. Verify the policy is applied with `gpresult /h gp_report.html`. On Linux/macOS endpoints outside MDM, deploy a managed Chrome preferences JSON at the system level. Document the gap formally if enforcement cannot be confirmed — this control failure is a finding, not just a recommendation.

Evidence: This step does not alter live system state and does not destroy volatile evidence. However, document the current state of browser policy enforcement — export the existing Chrome policy via `chrome://policy` on a representative endpoint and screenshot or export to JSON — before any policy changes are applied. This creates a before/after baseline showing whether the allowlist gap existed at the time of the incident, which is relevant to any breach notification or regulatory assessment.

Step 3: Update threat model — add AI-brand-impersonation browser extensions as a tracked TTP pattern in your threat register, citing MITRE T1036.005 (Masquerading: Match Legitimate Name or Location), T1056.001 (Keylogging), and T1041 (Exfiltration Over C2 Channel); reference Microsoft's March 2026 and June 2026 disclosures as a documented campaign cluster

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity: update organizational threat models and detection logic based on lessons learned from this campaign cluster, per the post-incident lessons-learned requirement to improve detection and prevention capability

Controls: CIS 7.1 (Establish and Maintain a Vulnerability Management Process)

Compensating: Without a commercial threat intelligence platform, maintain a threat register in a shared spreadsheet or wiki documenting: extension name patterns used (AI-branded, Perplexity, ChatGPT, Copilot impersonators), the pre-submission keystroke capture technique (distinct from form-field keylogging because it fires on `chrome.tabs.onUpdated` or `chrome.webNavigation` APIs before any HTTP request), and the C2 exfiltration endpoint domains identified in the Microsoft disclosure. Create a YARA rule targeting the specific JavaScript pattern used for address bar event listeners (e.g., targeting `chrome.tabs.onUpdated` combined with XHR or fetch calls to non-Google domains) for periodic scanning of the Chrome extensions directory on managed endpoints.

Evidence: No volatile evidence capture is required for this post-incident threat modeling step. If the campaign C2 domains from the Microsoft disclosure are known, query historical DNS logs and proxy logs for those domains across the full retention window — this may reveal additional infected endpoints not identified in Step 1's extension audit.

Step 4: Communicate findings — brief leadership on the credential pre-interception risk: any employee who installed an AI-branded Chrome extension outside an approved software process may have had address bar input — including internal URLs, credentials typed in the address bar, and sensitive search queries — exfiltrated to an unknown third party

NIST Phase: Containment

Reference: NIST 800-61r3 §3.3 — Containment: execute stakeholder notification as a required containment activity; leadership must understand that credential pre-interception (before Enter is pressed, before any browser network request is initiated) means standard web proxy and DLP controls would not have captured this data in transit

Controls: NIST AC-2 (Account Management)

Compensating: Without a formal IR communications platform, prepare a one-page briefing document covering: (1) the pre-submission capture mechanism — keystrokes were intercepted by the extension's JavaScript event listener before the browser sent any network request, bypassing proxy inspection entirely; (2) the data types at risk — any string typed in the Chrome address bar, including credentials entered directly, internal application URLs containing session tokens, and sensitive search queries; (3) the affected population from Step 1's audit; (4) the immediate credential rotation requirement for any accounts whose credentials may have been typed in the address bar on an affected endpoint. Distribute via encrypted email to CISO and relevant business unit leads.

Evidence: Before initiating credential rotation (which constitutes a live state change in identity systems), capture: (1) the list of affected user accounts identified in Step 1's extension audit tied to specific endpoints; (2) the extension's active installation period (from the Chrome `Preferences` file `install_time` field) to bound the credential exposure window; (3) any available proxy or DNS logs showing outbound connections from affected endpoints to the attacker-controlled C2 domain identified in the Microsoft disclosure, timestamped to correlate with the installation period. These records establish the scope and timeline required for breach notification assessment.

Step 5: Monitor developments — track Microsoft's Defender Security Research blog and the Chrome Web Store policy feed for follow-up disclosures on this campaign cluster; watch for additional AI-branded extensions applying the same pre-submission keylogging pattern

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity: establish ongoing monitoring for campaign evolution and new indicators, and feed findings back into detection engineering and the threat model updated in Step 3

Controls: CIS 7.2 (Establish and Maintain a Remediation Process)

Compensating: Without a commercial threat intel feed, configure free RSS monitoring on the Microsoft Security Research blog and Chrome Web Store policy changelog using a self-hosted RSS aggregator (e.g., FreshRSS) or a no-cost service (Feedly free tier). Write a Sigma rule targeting the behavioral pattern — specifically, Chrome child processes or extension background workers initiating outbound HTTP POST requests to non-Google domains within seconds of a `chrome.tabs.onUpdated` event — and deploy it against Windows Event Logs enriched with Sysmon Event ID 3 (Network Connection) filtered on `chrome.exe` as parent process. Review weekly for new AI-branded extension names against the approved allowlist from Step 2.

Evidence: No volatile evidence capture is required for this ongoing monitoring step. However, retain all forensic artifacts collected in Steps 1 and 4 — extension manifests, JS source, Chrome Preferences files, C2 connection logs — for a minimum of 90 days or your organization's incident retention policy, whichever is longer, to support any regulatory inquiry or follow-on investigation if additional victims or campaign infrastructure are identified in subsequent Microsoft disclosures.

Detection Guidance

Hunt for Chrome extensions installed outside your approved inventory by auditing extension IDs across managed endpoints via Google Admin Console or endpoint management tooling (aligns with CIS 2.1 and CIS 2.3). In endpoint logs, look for browser processes initiating outbound HTTPS connections to domains that do not correspond to any known SaaS or business application, particularly short-lived or newly registered domains

consistent with T1583.006 (Web Services acquisition for attacker infrastructure). DNS query logs are a useful secondary signal: extensions exfiltrating pre-submission keystrokes will generate DNS lookups and small outbound POST or GET requests at high frequency tied to user typing cadence rather than page loads. Behavioral anomaly detection should flag browser processes making many low-volume outbound connections in rapid succession without corresponding user-initiated navigation events. For SIEM rules, correlate Chrome extension install events (Windows registry or macOS plist changes) with subsequent new external connection destinations from the browser process. Review NIST AU-2 (Event Logging) and AU-6 (Audit Record Review) configurations to confirm browser-layer events are in scope. NIST SI-4 (System Monitoring) should extend to browser process network behavior on managed endpoints. D3-LAM (Local Account Monitoring) and D3-SFA (System File Analysis) apply for detecting unauthorized extension installations modifying browser configuration files. The cited Microsoft source (https://www.microsoft.com/en-us/security/blog/2026/06/29/chromium-extension-uses-airelated-branding-redirect-browser-search/) may contain specific C2 domains or payload identifiers, consult that disclosure directly for indicator values.

Framework Mappings

MITRE-ATTACK

- **T1176** — Software Extensions
- **T1036.005** — Match Legitimate Resource Name or Location
- **T1041** — Exfiltration Over C2 Channel
- **T1583.006** — Web Services
- **T1071.001** — Web Protocols
- **T1557** — Adversary-in-the-Middle
- **T1056.001** — Keylogging

NIST-800-53R5

- **CA-7** — Continuous Monitoring
- **SC-7** — Boundary Protection
- **SI-4** — System Monitoring

CIS-V8

- **6.3** — Require MFA for Externally-Exposed Applications

HIPAA-SECURITY

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

SOC2-TSC

- **CC6.1** — Logical access security software, infrastructure, and architectures

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1176	Software Extensions	Persistence

Technique ID	Technique Name	Tactic
T1036.005	Match Legitimate Resource Name or Location	Defense-Evasion
T1041	Exfiltration Over C2 Channel	Exfiltration
T1583.006	Web Services	Resource-Development
T1071.001	Web Protocols	Command-And-Control
T1557	Adversary-in-the-Middle	Credential-Access
T1056.001	Keylogging	Collection

Sources

Source	URL	Tier
Security News	https://thehackernews.com/2026/06/malicious-perplexity-chrome-exten...	T2
Chromium extension uses AI■related branding to redirect browser ...	https://www.microsoft.com/en-us/security/blog/2026/06/29/chromium-e...	T1
Malicious Perplexity-Themed Chrome Extension Hijacked Search ...	https://www.mallory.ai/stories/019f1492-7fbc-7549-bab3-01a445177f23	T3
Malicious AI Assistant Extensions Harvest LLM Chat Histories	https://www.microsoft.com/en-us/security/blog/2026/03/05/malicious-...	T1

Generated 2026-06-30 14:47 UTC by TJS Security Command Center