INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-04-11 06:11 UTC

AI-Powered Browser Extensions Identified as Significant Blind Spot for Enterprise Security

SECURITY ANALYSIS | **HIGH** | CVSS 7.5

SCC Item ID	SCC-STY-2026-0054
Type	Security Analysis
Severity	HIGH
CVSS Base Score	7.5
Affected Products	Enterprise environments using Chromium-based browsers with AI-powered extensions (e.g., Chrome, Edge); specific extension versions unspecified
Published	2026-04-10
Discovery Source	Gemini

Executive Summary

AI-powered browser extensions now represent a systemic, undermonitored attack surface in enterprise environments, capable of harvesting credentials, exfiltrating sensitive data, and bypassing perimeter controls, all from within the trusted browser process. A documented Chrome vulnerability, detailed by Palo Alto Networks Unit 42, demonstrates concrete exploitation potential, showing how extensions can hijack active AI sessions such as Gemini Live. This threat signals a structural gap in enterprise security architecture: organizations that have invested heavily in network and endpoint controls may have left the browser, now a primary productivity environment, largely ungoverned.

Technical Analysis

Browser extensions present a deceptively privileged attack surface. Once installed, an extension operates inside the browser process itself, a context that most endpoint detection and response tools and network security appliances treat as trusted. This architectural reality creates four compounding risk vectors that security teams must understand together, not in isolation.

First, data exfiltration. Extensions with the webRequest or declarativeNetRequest permission can inspect, modify, or relay outbound network traffic. API calls to cloud endpoints blend with legitimate HTTPS traffic, and without TLS inspection at the egress layer, content-level inspection is effectively blind. LayerX's analysis of AI extensions specifically flags this pattern, noting that AI extensions routinely transmit DOM content, including form data and page text, to external inference APIs as part of their core function. T1567 (Exfiltration Over Web Service) documents this pattern, where extensions transmit exfiltrated content via web APIs.

Second, credential and session harvesting. Extensions with activeTab or tabs permissions can read the DOM of any page the user visits, including login forms before submission. Session tokens stored in browser storage are accessible via the chrome.storage and chrome.cookies APIs. MITRE ATT&CK documents this under T1539 (Steal Web Session Cookie) and T1555 (Credentials from Password Stores). Palo Alto Networks Unit 42's analysis of the Chrome extension vulnerability affecting Gemini Live provides a concrete technical case study: a Chrome vulnerability allowed a malicious or compromised extension to inject into and intercept an active Gemini Live AI session, capturing conversation content and potentially authentication context. This aligns with T1185 (Man in the Browser).

Third, supply chain compromise. The Chrome Web Store and Edge Add-ons marketplace are not zero-risk distribution channels. Typosquatting, malicious updates to previously legitimate extensions, and extensions that request excessive permissions relative to their stated function are documented patterns. CWE-494 (Download of Code Without Integrity Check) is directly applicable here, as enterprise environments frequently lack controls that verify extension integrity or lock extensions to a specific version hash after vetting.

Fourth, permission creep. Browser security reports flag that enterprise-deployed extensions frequently carry permissions, including identity, storage, tabs, and webRequest, that exceed their functional requirements. CWE-272 (Least Privilege Violation) and CWE-284 (Improper Access Control) both map to this pattern. Without a defined permission baseline and periodic audit process, organizations have no reliable mechanism to detect when an extension's permissions changed after an update.

The defensive gap this exploits is structural. Firewalls and DLP appliances inspect network traffic at the perimeter but cannot see inside HTTPS streams without TLS termination. EDR tools monitor process behavior but typically do not generate telemetry for browser extension activity at the API call level. Group Policy and mobile device management platforms can enforce allowlists, but most enterprises have not implemented extension governance policies, leaving the browser as an effectively unmanaged application layer inside an otherwise-controlled endpoint.

Action Checklist

1. Step 1: Assess exposure, inventory all browser extensions currently installed across enterprise endpoints using Chrome Browser Cloud Management (Google), Microsoft Intune (Microsoft Edge), or equivalent platforms such as Jamf Pro or Kandji; flag any AI-powered extensions that transmit data to external APIs
2. Step 2: Review controls, verify whether TLS/SSL inspection is enabled on egress traffic to detect anomalous API calls from extensions; confirm EDR coverage includes browser process telemetry; check whether extension allowlisting policies are enforced via Group Policy or browser management platforms
3. Step 3: Update threat model, incorporate T1539, T1555, T1185, T1176 (Browser Extensions), and T1567 into your threat register with browser extensions as the access vector; reference Palo Alto Networks Unit 42's analysis of the Gemini Live hijacking vulnerability as a concrete exploitation scenario for threat modeling exercises
4. Step 4: Communicate findings, brief leadership on the specific risk that AI productivity extensions, if unvetted, can exfiltrate data handled in the browser (including credentials, documents, and AI session content) through channels that bypass standard perimeter controls
5. Step 5: Monitor developments, track Palo Alto Networks Unit 42 advisories and Chrome security updates for follow-up disclosures related to extension vulnerabilities; monitor Chrome Web Store and Edge Add-ons for supply chain compromise reports; subscribe to CISA and vendor browser security

advisories for emerging extension threat intelligence

IR / Forensic Enrichment

Triage Priority	URGENT
Escalation Criteria	Escalate immediately to CISO and legal if osquery or EDR telemetry confirms an AI-powered extension with webRequest or cookies permissions has made outbound POST requests to external AI API endpoints containing browser session tokens, OAuth tokens, or document content — particularly if the affected endpoints processed data classified as PII, PHI, or PCI-DSS in-scope data, as this may trigger breach notification obligations under GDPR Article 33, HIPAA §164.410, or applicable state law.
Recovery Notes	Post-containment, force-remove any flagged extensions via Group Policy ExtensionInstallBlocklist or Chrome Browser Cloud Management and invalidate all browser session tokens and OAuth refresh tokens for affected users by forcing re-authentication through your IdP (e.g., Entra ID Conditional Access session revocation or Okta session clear). For users whose Gemini Live, ChatGPT, or equivalent AI sessions were active while a flagged extension was installed, treat those session transcripts as potentially exfiltrated and assess whether they contained credentials, documents, or PII that require breach notification evaluation. Monitor re-enrollment of extensions on remediated endpoints for 30 days using osquery scheduled queries to detect reinstallation attempts, as users may manually re-add removed extensions if allowlist enforcement is not locked at the browser management policy level.
Forensic Artifacts	Chrome extension manifest.json files at 'C:\Users\\AppData\Local\Google\Chrome\User Data\Default\Extensions\\\manifest.json' — the 'permissions' array in each manifest identifies whether the extension declared webRequest, tabs, cookies, identity, or '' access, which are the exact permissions enabling the Gemini Live session hijacking technique documented by Unit 42 | Chrome Network Service log and browser process network connections — on Windows, Sysmon Event ID 3 records filtered on Image='chrome.exe' or Image='msedge.exe' showing outbound TCP connections to AI API endpoints (api.openai.com, generativelanguage.googleapis.com) with POST method, which would indicate extension-initiated data transmission rather than user-browsing traffic | Chrome 'Default/Preferences' JSON file containing the 'extensions.settings' key — this records each installed extension's install time, install source (webstore vs. sideloaded), enabled state, and granted permissions at the time of installation, providing a timeline of when high-risk AI extensions entered the environment | Egress proxy or NGFW logs showing HTTPS CONNECT or TLS SNI records to AI API domains during business hours, correlated against user activity timelines — the specific exfiltration pattern for browser extension data theft is high-frequency small POST requests to API endpoints that do not match the user's browser history of intentional AI tool usage | Windows Registry keys 'HKCU\SOFTWARE\Google\Chrome\PreferenceMACs' and 'HKLM\SOFTWARE\Policies\Google\Chrome\ExtensionInstallAllowlist' — the first detects tampering with Chrome's extension preference integrity checks (a sign of forced extension installation), and the second reveals whether allowlist policies were absent or misconfigured, establishing the pre-incident control gap

Per-Action IR Details

Step 1: Assess exposure — inventory all browser extensions currently installed across enterprise endpoints using Chrome Browser Cloud Management, Microsoft Edge management policies, or an equivalent browser

management platform; flag any AI-powered extensions that transmit data to external APIs

NIST Phase: Preparation

Reference: NIST 800-61r3 §2 — Preparation: Establish IR capability through asset visibility and attack surface enumeration before exploitation occurs

Controls: NIST SI-5 (Security Alerts, Advisories, and Directives) — consume Unit 42 Gemini Live hijacking research as an external advisory driving this inventory action, NIST CM-8 (System Component Inventory) — browser extensions are system components and must be inventoried to known-good baselines, CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory) — extensions installed on enterprise endpoints are assets with network transmission capability and must appear in the inventory, CIS 2.1 (Establish and Maintain a Software Inventory) — AI-powered browser extensions (e.g., Gemini, Copilot, ChatGPT extensions) are software that must be enumerated and assessed for authorization status, CIS 2.2 (Ensure Authorized Software is Currently Supported) — flag extensions that lack a verifiable vendor support lifecycle or are not pinned to a known-safe version

Compensating: For teams without Chrome Browser Cloud Management or Intune: run 'Get-ItemProperty HKLM:\SOFTWARE\Google\Chrome\Extensions' and 'HKCU:\SOFTWARE\Google\Chrome\Extensions' via PowerShell on sampled endpoints to enumerate installed extensions by CRX ID; cross-reference each ID against the Chrome Web Store API endpoint (https://chrome.google.com/webstore/detail/{extension_id}) to resolve name and publisher. On Linux/macOS endpoints, enumerate ~/.config/google-chrome/Default/Extensions/ directories. Use osquery with the query 'SELECT name, identifier, version, permissions FROM chrome_extensions;' across the fleet for structured output without an enterprise console.

Evidence: Before inventorying, snapshot the current state to establish a forensic baseline: export Chrome extension state from each endpoint at 'C:\Users\\AppData\Local\Google\Chrome\User Data\Default\Extensions\' (Windows) or '~/Library/Application Support/Google/Chrome/Default/Extensions/' (macOS); capture the 'manifest.json' for each installed extension, paying particular attention to 'permissions' arrays listing 'tabs', 'webRequest', 'cookies', 'storage', 'identity', and '' — these are the permissions that enable session hijacking and credential harvesting as documented in the Unit 42 Gemini Live research. Also preserve the Chrome Preferences file at 'Default/Preferences' which logs extension install timestamps and sources.

Step 2: Review controls — verify whether TLS/SSL inspection is enabled on egress traffic to detect anomalous API calls from extensions; confirm EDR coverage includes browser process telemetry; check whether extension allowlisting policies are enforced via Group Policy or browser management platforms

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.2 — Detection and Analysis: Validate monitoring coverage gaps before an active incident confirms the blind spot exploited by extension-based data exfiltration

Controls: NIST SI-4 (System Monitoring) — verify that browser process telemetry from chrome.exe and msedge.exe is captured by EDR, specifically child process creation, network connections initiated by the browser, and extension worker activity, NIST AU-2 (Event Logging) — confirm that egress proxy or NGFW logs capture HTTPS POST requests from browser extension processes to external AI API endpoints (e.g., api.openai.com, generativelanguage.googleapis.com, api.anthropic.com), NIST AU-12 (Audit Record Generation) — validate that audit records include the originating process name for outbound connections so extension-initiated calls can be distinguished from user-initiated browsing, CIS 4.4 (Implement and Manage a Firewall on Servers) — verify egress filtering rules would surface unexpected API destinations contacted by AI extensions, CIS 8.2 (Collect Audit Logs) — confirm browser process network telemetry is flowing to a centralized log store with sufficient retention to support retrospective analysis

Compensating: For teams without TLS inspection or enterprise EDR: deploy Sysmon with a configuration that enables Event ID 3 (Network Connection) filtered on Image containing 'chrome.exe' or 'msedge.exe' to capture all outbound connections initiated by the browser process — use the SwiftOnSecurity Sysmon config as a baseline. Run Wireshark with a capture filter of 'tcp.port == 443 and ip.src == ' on a representative sample of endpoints during business hours to identify TLS handshakes to unexpected AI API hostnames (SNI field visible pre-encryption). For Group Policy verification, run 'gpresult /h gpo_report.html' and search for ExtensionInstallAllowlist and ExtensionInstallBlocklist keys under 'Computer Configuration\Administrative Templates\Google\Google Chrome\Extensions'.

Evidence: Capture current proxy/firewall egress logs showing outbound HTTPS connections from browser processes to AI API domains (api.openai.com, generativelanguage.googleapis.com, bard.google.com, api.anthropic.com, copilot.microsoft.com) — these are the exfiltration channels AI extensions use to transmit harvested session data and credentials. Preserve EDR telemetry showing chrome.exe or msedge.exe network connection events for the prior 30 days before any policy changes are made. Export current Group Policy RSoP (Resultant Set of Policy) to document the pre-remediation allowlisting posture.

Step 3: Update threat model — incorporate T1539, T1555, T1185, and T1176 (Browser Extensions) into your threat register with browser extensions as the access vector; reference the Unit 42 Gemini Live hijacking research as a concrete exploitation scenario for threat modeling exercises

NIST Phase: Preparation

Reference: NIST 800-61r3 §2 — Preparation: Threat modeling against documented adversary techniques ensures detection and response capabilities are aligned to the actual attack surface before exploitation

Controls: NIST RA-3 (Risk Assessment) — incorporate the Unit 42 Gemini Live session hijacking scenario as a documented risk scenario: a malicious or compromised AI extension with 'tabs' and 'webRequest' permissions can intercept in-browser AI sessions without triggering perimeter controls, NIST IR-4 (Incident Handling) — update the incident handling capability to include browser extension compromise as a named incident category with defined detection signatures for T1539 (Steal Web Session Cookie), T1555 (Credentials from Password Stores), T1185 (Browser Session Hijacking), and T1176 (Browser Extensions), NIST SI-5 (Security Alerts, Advisories, and Directives) — formally ingest the Unit 42 Gemini Live hijacking advisory as a threat intelligence input to the risk register, CIS 7.1 (Establish and Maintain a Vulnerability Management Process) — add browser extension vetting as a recurring process within vulnerability management, triggered by new AI extension adoption requests

Compensating: For teams without a formal threat modeling tool: create a one-page threat register entry in a spreadsheet documenting: ATT&CK technique IDs (T1539, T1555, T1185, T1176), the specific mechanism (AI extension with webRequest/tabs permissions intercepting Gemini Live or equivalent sessions), current detection coverage (yes/no per technique), and the Unit 42 advisory URL as the source. Use MITRE ATT&CK Navigator (free, browser-based at attack.mitre.org/resources/attack-navigator/) to create a layer file highlighting these four techniques for inclusion in tabletop exercise materials.

Evidence: Document current detection coverage gaps as evidence of pre-existing risk posture: export ATT&CK Navigator coverage maps showing which of T1539, T1555, T1185, T1176 have active detection rules versus none. Preserve the current extension allowlist policy state (or absence thereof) as a baseline artifact. Capture the Unit 42 Gemini Live advisory publication date and internal receipt date to establish when the organization became aware of the specific exploitation scenario — this timeline is material for post-incident regulatory analysis.

Step 4: Communicate findings — brief leadership on the specific risk that AI productivity extensions, if unvetted, can exfiltrate data handled in the browser (including credentials, documents, and AI session content) through channels that bypass standard perimeter controls

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.2 — Detection and Analysis: Communicating scope and impact estimates to leadership is required to obtain authority for containment decisions and resource allocation before a confirmed incident forces reactive action

Controls: NIST IR-6 (Incident Reporting) — brief leadership on the identified exposure as a precursor incident condition requiring documented acknowledgment and a go/no-go decision on containment actions, NIST IR-8 (Incident Response Plan) — validate that the IR plan's leadership notification thresholds and communication templates cover browser-originated data exfiltration as a named scenario, NIST AU-6 (Audit Record Review, Analysis, and Reporting) — findings from the extension inventory and egress log review (Steps 1-2) must be formally reported with evidence, not summarized verbally without documentation, CIS 7.2 (Establish and Maintain a Remediation Process) — leadership briefing must result in a documented risk acceptance or remediation decision with an assigned owner and deadline

Compensating: For teams without a formal reporting template: produce a one-page brief structured as: (1) Threat — AI extensions with webRequest/tabs/cookies permissions can exfiltrate browser session tokens, credentials, and AI session transcripts to external APIs without triggering DLP or perimeter controls; (2) Exposure — X extensions flagged

in Step 1 inventory with external API transmission; (3) Evidence — Unit 42 Gemini Live hijacking research (cite the advisory); (4) Decision required — approve extension allowlisting enforcement by [date] or accept documented risk. Distribute via encrypted email and retain the response as a documented risk decision.

Evidence: The leadership briefing package itself becomes a forensic artifact: preserve the findings report, distribution list, timestamps, and any written responses or approvals. If a breach investigation later examines organizational awareness, this documentation establishes when leadership was formally notified of the browser extension risk — relevant for regulatory inquiries under GDPR, HIPAA, or state breach notification laws if AI session content included PII or PHI.

Step 5: Monitor developments — track the Unit 42 Gemini Live hijacking advisory for follow-up disclosures; monitor Chrome Web Store and Edge Add-ons for supply chain compromise reports; subscribe to LayerX and CISA browser security advisories for emerging extension threat intelligence

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity: Continuous threat intelligence intake on an identified attack surface prevents recurrence and improves detection before the next exploitation wave

Controls: NIST SI-5 (Security Alerts, Advisories, and Directives) — establish a formal process to receive and act on CISA advisories and Unit 42 threat research related to browser extension threats; assign an owner responsible for reviewing new disclosures within 48 hours, NIST IR-5 (Incident Monitoring) — add browser extension supply chain compromise (e.g., a previously approved extension pushing a malicious update) as a monitored incident category with defined alerting criteria, NIST AU-13 (Monitoring for Information Disclosure) — monitor public sources (Chrome Web Store reviews, security researcher disclosures, CISA KEV) for reports of AI extensions known to be installed in your environment being flagged as malicious, CIS 7.1 (Establish and Maintain a Vulnerability Management Process) — integrate browser extension threat intelligence feeds into the vulnerability management cadence so newly reported malicious extensions trigger immediate inventory cross-reference, CIS 2.3 (Address Unauthorized Software) — any extension flagged in an external advisory that matches an installed extension in the enterprise inventory must be treated as unauthorized and trigger the removal process

Compensating: For teams without a commercial TI feed: create a free RSS/Atom feed aggregator (e.g., Feedly free tier) subscribed to: CISA Alerts (cisa.gov/uscert/ncas/alerts), Unit 42 Threat Research (unit42.paloaltonetworks.com), and Chrome Releases blog (googlechromereleases.blogspot.com). Set a weekly calendar task to query osquery across sampled endpoints using 'SELECT name, identifier, version FROM chrome_extensions WHERE identifier IN ();' to check whether newly reported malicious extensions are present in the environment. Subscribe to the Chrome Web Store RSS feed for specific extension IDs of high-risk AI extensions installed in your environment using a free service such as Visualping to alert on update or removal events.

Evidence: Maintain a running threat intelligence log with date-stamped entries for each new advisory ingested, the extension IDs or names referenced, whether those extensions are present in your inventory (from Step 1), and the action taken. This log serves as evidence of due diligence in monitoring and is the foundation for future tabletop exercises and IR plan updates. If a supply chain compromise of a previously approved extension occurs, this log establishes the window between public disclosure and your detection — a key metric for NIST 800-61r3 §4 lessons-learned review.

Detection Guidance

Detection for extension-based threats requires visibility at the browser layer, which most organizations currently lack. Start with what is available.

Proxy and egress log analysis: Query for outbound HTTPS POST requests from browser processes to AI API endpoints (e.g., *.googleapis.com, *.openai.com, *.anthropic.com, and similar) that occur outside expected application contexts. Unusual volume, frequency, or timing, especially during off-hours or from endpoints without licensed AI tools, warrants investigation. Without TLS inspection, you can at minimum detect destination and volume anomalies.

Chrome Browser Cloud Management or Edge management telemetry: If deployed, these platforms provide extension installation events, permission requests, and version change logs. Alert on: (1) new extension installations on managed endpoints not on the approved allowlist, (2) permission changes in existing extensions following an update, and (3) extensions with webRequest, identity, tabs, and storage permissions in combination.

Endpoint process telemetry: Hunt for browser child processes or browser helper objects making network connections to domains not associated with the user's normal browsing baseline. Anomalous DNS resolutions or TLS handshakes to unfamiliar API infrastructure from browser processes are a viable hunting signal.

Credential and session token exposure: Review authentication logs for session tokens appearing from unexpected source IPs or geographic locations. This can indicate harvested session cookies being replayed from attacker-controlled infrastructure (T1539).

Policy gap audit: Run a permission audit of all installed extensions against a defined baseline. Extensions carrying the identity or webRequest permission without documented business justification should be quarantined pending review. Pay particular attention to AI writing assistants, grammar tools, and productivity extensions, as these categories have the broadest functional need for DOM and clipboard access.

Indicators of Compromise

Type	Value	Context	Confidence
URL	`Pending — refer to Palo Alto Unit 42 (https://unit42.paloaltonetworks.com/gemini-live-in-chrome-hijacking/) for published technical indicators`	Unit 42's Gemini Live hijacking research documents a specific Chrome extension exploitation technique; concrete IOCs (extension identifiers, C2 infrastructure, payload hashes) should be retrieved directly from the Unit 42 advisory	LOW

Framework Mappings

MITRE-ATTACK

- **T1539** — Steal Web Session Cookie
- **T1555** — Credentials from Password Stores
- **T1185** — Browser Session Hijacking
- **T1176** — Software Extensions
- **T1567** — Exfiltration Over Web Service

OWASP-TOP10-2021

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

NIST-800-53R5

- **SI-7** — Software, Firmware, and Information Integrity
- **CM-3** — Configuration Change Control

- **AC-3** — Access Enforcement
- **SC-28** — Protection of Information at Rest
- **AT-2** — Literacy Training and Awareness
- **SR-2** — Supply Chain Risk Management Plan
- **SC-13** — Cryptographic Protection
- **SI-4** — System Monitoring

CIS-V8

- **2.5** — Allowlist Authorized Software
- **2.6** — Allowlist Authorized Libraries
- **6.1** — Establish an Access Granting Process
- **6.2** — Establish an Access Revoking Process
- **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
- **8.2** — Collect Audit Logs

HIPAA-SECURITY

- **164.312(a)(1)** — Access Control
- **164.312(d)** — Person or Entity Authentication
- **164.308(a)(6)(ii)** — Response and Reporting
- **164.312(e)(1)** — Transmission Security

SOC2-TSC

- **CC6.1** — The entity implements logical access security software, infrastructure, and architectures over protected information assets
- **CC7.4** — Responds to identified security incidents
- **CC9.2** — Manages risks associated with vendors and business partners

ISO-27001-2022

- **A.8.8** — Management of technical vulnerabilities
- **A.5.34** — Privacy and protection of personal information
- **A.5.21** — Managing information security in the ICT supply chain
- **A.5.23** — Information security for use of cloud services

NIST-CSF-2

- **GV.SC-01** — Cybersecurity supply chain risk management program
- **DE.CM-01** — Networks and network services are monitored

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1539	Steal Web Session Cookie	Credential-Access
T1555	Credentials from Password Stores	Credential-Access
T1185	Browser Session Hijacking	Collection
T1176	Software Extensions	Persistence
T1567	Exfiltration Over Web Service	Exfiltration

Sources

Source	URL	Tier
Vulnerability in Chrome Allowed Extensions to Hijack New Gemini ...	https://unit42.paloaltonetworks.com/gemini-live-in-chrome-hijacking/	T3
The hidden browser threat most enterprise security teams still can't see	https://www.okoone.com/spark/industry-insights/the-hidden-browser-t...	T3
AI Browser Extensions Security Risks - LayerX	https://layerxsecurity.com/learn/browser-extension/ai-powered-brows...	T3
Enterprise browser deployment vs security extensions...what really ...	https://www.reddit.com/r/ITCareerQuestions/comments/1qkl4mb/enterpr...	T3
New Browser Security Report Reveals Emerging Threats for ...	https://thehackernews.com/2025/11/new-browser-security-report-revea...	T3

Generated 2026-04-11 06:11 UTC by TJS Security Command Center