CSAI

cloud security alliance®

CSAI Foundation

Cloud Security Alliance AI Safety Initiative

AI Browser Extensions: Shadow AI's Hidden Attack Surface

How AI Plugins Bypass Enterprise Data Controls

Unofficial AI-assisted Research

2026-04-10

Key Takeaways

- In July 2025, Urban VPN Proxy silently introduced code in version 5.5.0 that intercepted AI conversations across eight major platforms — including ChatGPT, Claude, Gemini, and Microsoft Copilot — from more than 8 million browser users on Chrome and Edge, harvesting and selling those conversations to advertisers without meaningful re-consent from existing users [3][4][5].
- In January 2026, OX Security researchers discovered two malicious Chrome extensions impersonating AI assistants with a combined install base exceeding 900,000 users; the extensions used DOM scraping to exfiltrate complete ChatGPT and DeepSeek conversation histories to attacker-controlled domains every 30 minutes, while claiming to collect only "anonymous, non-identifiable analytics data" [1][2][12].
- Cato CTRL disclosed HashJack, a newly disclosed indirect prompt injection technique embedding adversarial instructions within URL hash fragments; when a user visits a crafted URL, an AI browser assistant reads the fragment and may execute instructions to exfiltrate data, conduct phishing, or compromise credentials — without any vulnerability in the extension itself [6].
- Traditional enterprise defenses — data loss prevention (DLP), cloud access security broker (CASB), and endpoint detection and response (EDR) — have limited or no native visibility into DOM-level browser extension behavior: data scraping occurs within the browser runtime and exfiltrates over ordinary outbound HTTPS, generating no anomaly detectable at the network perimeter [7].
- The enterprise governance gap is substantial: 80% of organizations have encountered risky behaviors from AI agents [9], yet only 37% have adjusted their security strategies in response to AI-driven threats [9], and most lack governance policies specifically addressing browser extension procurement, permission review, or data handling requirements.
- Organizations should treat AI-capable browser extensions as a distinct risk tier requiring allowlist-based installation control, per-extension permission auditing, and integration with enterprise browser management platforms; as extensions acquire agentic automation capabilities, the attack surface expands from passive data exfiltration to unauthorized action execution against internal applications.

Background

The browser has become the primary workplace for a substantial portion of the modern enterprise workforce. Email, collaboration tools, internal applications, and cloud services all converge in a single browser session, making it one of the most data-rich environments employees inhabit during working hours. This centrality has not been lost on AI developers: a generation of AI-powered browser extensions now promises to summarize pages, draft emails, automate repetitive workflows, and serve as always-available AI assistants across every website a user visits. These extensions have proliferated rapidly, driven by genuine productivity value and a marketplace model in which extensions can be published with limited security review — the Chrome Web Store's automated scanning has not prevented repeated campaigns of malicious AI-impersonation extensions from accumulating hundreds of thousands of installs before detection.

The security implications of this proliferation are materializing through a series of high-profile incidents in 2025 and 2026 that expose an architectural reality most enterprise security teams have not yet reckoned with. An installed browser extension operates with continuous, silent access to browser state, page content, form inputs, authentication cookies, and user keystrokes — not by exploiting a vulnerability, but by design. The Chrome extension manifest system grants these permissions legally, through install-time consent dialogs that users widely accept without fully reviewing the permissions disclosed. When the extension presents itself as a productivity-enhancing AI assistant, users extend implicit trust that the AI company's stated purpose aligns with the extension's actual behavior. That assumption has proven unreliable.

The threat landscape encompasses three distinct but overlapping attack patterns. The first involves malicious or rogue extensions that impersonate legitimate AI assistants and systematically harvest conversation data for sale or further exploitation. The second involves legitimate extensions that silently expand their data collection scope through automated updates, exploiting users who consented to a narrower use case at install time. The third involves adversarial manipulation of AI extension reasoning through prompt injection, turning the extension's own intelligence against the user's interests. Each pattern exposes different organizational risk surfaces and requires distinct mitigations, but all three share a common enabler: the absence of enterprise-grade controls over which extensions are installed, what permissions they hold, and how they handle the sensitive data flowing through the browser.

Security Analysis

The Permission Architecture as an Attack Surface

Browser extension permissions represent a fundamental tension between user utility and data exposure. To function effectively as an AI assistant — summarizing content, drafting responses, navigating pages — an extension requires broad access to browser state. The `tabs` permission allows reading URLs and tab metadata. The `activeTab` permission enables access to the current page's full document object model (DOM). The `scripting` permission allows content scripts to read and modify any element on a page. The `cookies` permission provides access to session tokens that authenticate the user to cloud services. Host permissions specified as `<all_urls>` extend these capabilities to every website a user visits, meaning a single extension can observe every interaction across an entire browsing session without any per-site notification to the user [7].

These permissions are not obtained through exploitation. They are disclosed at install time in a consent dialog that most users treat as a formality. LayerX Security's Enterprise Browser Extension Security Report 2025 found that 53% of enterprise users have extensions carrying high or critical permission scopes capable of accessing sensitive data including cookies, passwords, and full page contents [17]. In an enterprise context where a single browser session holds access tokens for cloud services, SaaS platforms, and internal applications, this permission scope is equivalent to granting a third-party application real-time read access to everything the user can see — with no ongoing audit trail, no scope restriction by data classification, and no mechanism for the enterprise to revoke that access short of uninstalling the extension.

A particular concern is the auto-update mechanism built into both Chrome and Edge extension platforms. Extensions update automatically and silently in the background; the initial consent granted at install time covers all future versions of the extension by default. This design enables the rapid iteration that makes extensions useful, but it also creates a structural channel through which a developer can introduce new data collection behaviors to an established user base without requiring explicit re-authorization. As the Urban VPN incident demonstrates, this is not a theoretical risk.

Data Exfiltration via Malicious and Rogue Extensions

The Urban VPN incident, disclosed in December 2025, established the scale at which browser extensions can execute covert data collection. Urban VPN Proxy had accumulated more than 6 million Chrome users and 1.3 million Edge users, lending it apparent legitimacy through high install counts and user ratings [3][4]. The company's version 5.5.0, released on July 9, 2025 as an automated silent

update, introduced code that intercepted conversations from eight AI platforms — ChatGPT, Claude, Gemini, Microsoft Copilot, Perplexity, DeepSeek, Grok, and Meta AI — and transmitted the full content to Urban VPN's servers for sale to advertisers [3][5]. Because the update was delivered silently to existing users, the 7.3 million people who had installed the extension before July 9 never saw an updated consent prompt explaining the new data collection behavior [5]. The same harvesting code appeared in seven additional extensions from the same publisher, bringing the total affected user base to more than 8 million [4]. The Urban VPN incident reveals that the automated update mechanism — designed to improve user experience — also functions as a covert deployment channel for data harvesting that bypasses the one-time consent event users experience at install.

The campaign discovered by OX Security in January 2026 demonstrates that dedicated adversaries are building parallel infrastructure specifically to exploit the AI extension ecosystem [1]. Two malicious extensions — "Chat GPT for Chrome with GPT-5, Claude Sonnet & DeepSeek AI" with approximately 600,000 users, and "AI Sidebar with Deepseek, ChatGPT, Claude, and more" with approximately 300,000 users — operated a systematic data exfiltration scheme in which DOM scraping extracted complete conversation content from ChatGPT and DeepSeek interfaces, and that content was transmitted to attacker-controlled domains (chatsaigpt.com and deepaichats.com) on a 30-minute interval [1][2][12]. The extensions obscured their behavior through a false consent prompt claiming that collected data was "anonymous, non-identifiable analytics," while the actual payload included full, identifiable conversation histories alongside comprehensive browsing data [1]. Microsoft's Security blog reported that the campaign was active across more than 20,000 enterprise tenants, indicating meaningful organizational penetration rather than purely consumer exposure [11]. The enterprise consequence is concrete: employees who discussed strategic plans, customer accounts, legal positions, or proprietary technical details with AI assistants — over these extensions — had that content harvested and transmitted to adversarial infrastructure.

Indirect Prompt Injection and Agentic Abuse

A second threat vector targets the AI reasoning capability of extensions rather than their data access permissions. Indirect prompt injection describes an attack in which adversarial instructions are embedded in content that an AI system reads as part of normal operation — web pages, emails, documents — causing the AI to interpret those instructions as if they originated from a trusted user [6][8]. In the context of AI browser extensions, every webpage, embedded script, advertisement, and dynamically loaded element that the extension processes represents a potential injection surface. The attack does not require a vulnerability in the extension; it exploits the design principle that the AI agent should be helpful by acting on observed content.

Cato CTRL researchers disclosed HashJack as a concrete instantiation of this threat, demonstrating that adversarial instructions can be embedded within URL hash fragments — the portion of a URL following the `#` character, which browsers treat as client-side navigation metadata and which web applications commonly use for state management [6]. When a user navigates to a URL containing embedded HashJack instructions, the AI browser assistant processes the fragment content as part of page analysis and may execute instructions to exfiltrate sensitive data, redirect authentication flows, conduct social engineering against the user, or initiate account compromise sequences [6]. Cato CTRL identified six primary risk categories arising from this technique: callback phishing, credential theft, misinformation, malware guidance, medical harm, and data exfiltration in agentic contexts [6].

The Wiz 2025 Year-End Review on agentic browser security documented additional attack variants discovered by independent researchers, including the "Gemini Trifecta" (background API calls that leak sensitive data out of the browser session), "Tainted Memories" (cross-site request forgery enabling persistent malicious instruction embedding), and a one-click hijack of the Perplexity Comet browser using a crafted URL [13]. The breadth and pace of disclosed variants indicates that indirect prompt injection against AI browser agents is an active and rapidly developing research area. OpenAI has acknowledged publicly that this problem class may not admit a complete solution: the company's chief information security officer has described prompt injection as a frontier security challenge, noting that as long as AI agents operate against untrusted web content, adversarial instructions in that content can influence agent behavior [8]. Anthropic's published research on prompt injection defenses in browser-use contexts frames the challenge similarly — as an ongoing mitigation problem rather than a solvable boundary condition [14].

The risk profile escalates as browser extensions acquire agentic capabilities. While the distinction is not absolute — a passive extension that captures authentication tokens can enable downstream account compromise — extensions that can observe page content present primarily a data exfiltration risk. Extensions that can click, fill forms, submit data, and navigate between pages — the agentic browser assistants now entering enterprise markets — present a qualitatively more direct threat: a prompt-injected agentic extension can take unauthorized actions within internal applications, SaaS platforms, and cloud consoles at the speed of automation, without leaving the observable fingerprint of a separate attacker process on the endpoint.

The Enterprise Visibility and Governance Gap

The threat posed by AI browser extensions is compounded by a governance architecture that was not designed for this risk category. Traditional enterprise security stacks defend the network perimeter, the endpoint file system, cloud service API access, and email — but the browser runtime sits architecturally outside the observation point of each of these tools. CASB platforms monitor API-level traffic to

sanctioned cloud services and identify access by sanctioned identities. DLP solutions inspect file transfers, email attachments, and structured data flows. EDR platforms monitor process behavior, file system activity, and network connections from the endpoint's perspective. These tools typically lack native visibility into DOM-level operations that occur within a browser session: a content script that reads form inputs and uploads them as an HTTPS POST to a third-party domain generates traffic indistinguishable at the perimeter from any other browsing activity, originating from the user's own machine under the user's own credentials [7].

This architectural blind spot is precisely what makes browser extensions an effective attack vector for adversaries who understand enterprise security tooling. The extension operates inside the trusted user session, with the trusted user's access, over the trusted user's network path. There is no lateral movement, no privilege escalation, no anomalous process creation — only the quiet movement of sensitive data through an HTTPS connection that appears, from every instrumented vantage point, to be normal user traffic. Microsoft recognized this structural limitation in its RSAC 2026 announcements, introducing inline DLP enforcement through Microsoft Purview that operates at the browser layer within Edge for Business rather than at the network perimeter [10]. This design moves the observation point into the runtime where the data collection actually occurs, which is the architecturally correct response — but it requires committed enterprise deployment of a managed browser and is specific to a single vendor's ecosystem.

The governance gap compounds the technical visibility problem. Surveys consistently find that unsanctioned AI tool use is near-universal across enterprise workforces — 80% of organizations have encountered risky behaviors from AI agents [9] — yet formal governance policies specifically addressing browser-based AI tools remain the exception rather than the rule: only 37% of organizations have adjusted their security strategies in response to AI-driven threats at all [9]. In the absence of an explicit organizational stance on which AI browser extensions are approved for use, employees make their own procurement decisions based on store ratings, peer recommendations, and marketing claims — the same trust signals that allowed the January 2026 impersonation campaign to accumulate 900,000 installs before detection. Unlike SaaS applications that traverse procurement review and vendor due diligence, browser extensions are installed with a single click and begin accessing sensitive data immediately.

Recommendations

Immediate Actions

Security and IT teams should conduct an immediate inventory of browser extensions deployed across managed endpoints, with particular focus on extensions holding broad host permissions (`<all_urls>`), access to cookies, or clipboard read capabilities. Enterprise endpoint management platforms — including Microsoft Intune, Google Workspace Admin Console, and Jamf — provide mechanisms to enumerate installed extensions; organizations that lack this visibility should treat its deployment as an urgent gap closure. Any extensions impersonating AI assistants with names that reference ChatGPT, Claude, Gemini, DeepSeek, or similar AI platforms but originating from publishers without an established security posture should be blocklisted and removed from managed endpoints, cross-referenced against Microsoft's March 2026 advisory on malicious AI assistant extension campaigns [11].

Organizations should also audit extension version histories for any extensions in current deployment that received silent updates beginning in July 2025 through the present — the period spanning both the Urban VPN incident and the broader campaigns disclosed by Microsoft. Extensions from publishers that cannot provide a clear data handling policy and verifiable terms governing AI conversation content should be treated as unvetted until reviewed.

Short-Term Mitigations

A foundational structural control against shadow AI browser extensions is allowlist-based extension management enforced through enterprise browser policy. Both Chrome Enterprise and Microsoft Edge for Business support group policy configurations that restrict extension installation to an administrator-approved list, preventing users from installing extensions outside the vetted set regardless of how they appear in the extension marketplace. This control eliminates the primary attack pathway — user-installed unapproved extensions — and ensures that the only AI browser tools operating in the environment have been evaluated against the organization's data handling and security standards. The allowlist should be governed by a defined review cadence aligned to the organization's change management processes, with an explicit approval workflow for new AI extension requests.

For extensions that are evaluated for enterprise approval, procurement criteria should require explicit answers to data handling questions: Where are conversation contents processed and stored? How long is conversation data retained? Is conversation content used for model training, and if so, under what opt-out mechanism? Are data processing agreements available? These questions mirror the SaaS vendor

due diligence that mature procurement processes apply to cloud applications, and they are equally applicable to browser extensions that access the same categories of sensitive information through a different delivery mechanism.

Organizations evaluating enterprise AI browser tools should also assess the agentic capability level of candidate extensions — distinguishing passive AI (read-only observation, summarization) from active AI (form filling, click automation, workflow submission) — and apply proportionate controls. Agentic extensions should be subject to scope-limited deployment (specific approved use cases and applications), with monitoring for unexpected action patterns that could indicate prompt injection exploitation.

Strategic Considerations

The emergence of AI browser assistants with genuine agentic capabilities — tools that can navigate, click, fill, and submit across enterprise applications — warrants a formal threat modeling exercise for organizations operating in regulated industries or handling sensitive intellectual property. MAESTRO, the CSA's agentic AI threat modeling framework, provides the layer-by-layer analytical structure to evaluate the risk profile of a given AI browser extension deployment, mapping its capability level to the trust boundaries it crosses and the control gaps those crossings create [16]. Organizations should apply MAESTRO analysis before approving any agentic browser extension for enterprise use, treating each capability expansion (from read-only to read-summarize to read-act) as a distinct risk tier requiring proportionate governance.

At the platform level, enterprise browser management through vendors offering managed extension environments — such as Edge for Business with Purview integration, or dedicated enterprise browser platforms — provides a path toward native visibility into extension behavior that traditional security tooling cannot achieve. Where enterprise browser deployment is feasible, it should be treated as a Zero Trust control applied at the browser layer, consistent with the principle that no tool or agent should be implicitly trusted based solely on user identity and network location.

The security community should also engage browser platform vendors on the auto-update consent model. The current architecture — in which initial install consent covers all future extension updates — is the structural enabler of the Urban VPN category of incident, where new data collection behaviors are deployed to millions of users without any new consent event. Requiring re-consent for extensions that introduce new permissions or expand data collection scope in updates would impose a meaningful accountability requirement on extension publishers without disrupting routine maintenance updates — though precise definition of what constitutes a data collection expansion would require platform-level specification to avoid unnecessary friction for legitimate maintenance releases.

CSA Resource Alignment

This research note engages directly with several active areas of the CSA AI Safety Initiative.

MAESTRO (Agentic AI Threat Modeling Framework) provides the primary analytical framework for evaluating AI browser extension risk. MAESTRO's layer-by-layer decomposition of agentic AI risk — spanning foundation models, agent frameworks, data operations, and deployment infrastructure — maps directly to the trust boundaries crossed by AI browser extensions: the extension mediates between the user's browser session (which contains authenticated access to enterprise systems) and an external AI service (which processes and may retain conversation content) [16]. The prompt injection attacks described in this note correspond to MAESTRO's threat categories around trust boundary violations and unauthorized agent action, and MAESTRO's structured threat assessment methodology should inform enterprise evaluation of agentic extension deployments. The framework's governing principle — that no agent should be implicitly trusted based on identity alone, and that all agent actions should require explicit scope authorization — translates directly into the requirement for scope-limited extension deployment with action monitoring. An agentic browser extension with blanket permission to act on behalf of the user across all applications violates Zero Trust principles in the same way that a service account with administrative access to all systems does.

AI Controls Matrix (AICM) v1.0 addresses supply chain security, data security, and AI customer governance — all directly applicable to the browser extension risk surface. The AICM's AI Customer implementation guidelines specify control requirements for organizations procuring third-party AI tools, including vendor due diligence, data handling verification, access scoping, and logging [15]. An enterprise extension allowlist process operationalizes AICM controls: each approved extension represents a vendor relationship that should meet the AICM's baseline requirements for data processing transparency and security posture.

STAR (Security Trust Assurance and Risk) provides the assurance mechanism for vendor evaluation in extension procurement. AI browser tool vendors that are STAR-registered or STAR-audited offer a verifiable, publicly accessible security posture assessment — a material advantage over the unvetted extension marketplace as a source of AI tools for enterprise deployment.

References

[1] OX Security. "900K Users Compromised: Chrome Extensions Steal ChatGPT and DeepSeek Conversations." OX Security Blog, January 2026.

[2] The Hacker News. "Two Chrome Extensions Caught Stealing ChatGPT and DeepSeek Chats from 900,000 Users." The Hacker News, January 2026.

[3] Infosecurity Magazine. "Urban VPN Proxy Accused of Harvesting AI Chat Conversations." Infosecurity Magazine, December 2025.

[4] Dark Reading. "Browser Extension Harvests 8M Users' AI Chatbot Data." Dark Reading, December 2025.

[5] Malwarebytes. "Chrome Extension Slurps Up AI Chats After Users Installed It for Privacy." Malwarebytes Blog, December 2025.

[6] Cato Networks. "Cato CTRL™ Threat Research: HashJack – Novel Indirect Prompt Injection Against AI Browser Assistants." Cato Networks Blog, 2025.

[7] The Hacker News. "Shadow AI in the Browser: The Next Enterprise Blind Spot." The Hacker News Expert Insights, December 2025.

[8] TechCrunch. "OpenAI Says AI Browsers May Always Be Vulnerable to Prompt Injection Attacks." TechCrunch, December 2025.

[9] Netwrix. "12 Critical Shadow AI Security Risks Your Organization Needs to Monitor in 2026." Netwrix Blog, 2026.

[10] Microsoft Edge Blog. "Protect Your Enterprise from Shadow AI and More: Announcements at RSAC 2026." Microsoft, March 2026.

[11] Microsoft Security Blog. "Malicious AI Assistant Extensions Harvest LLM Chat Histories." Microsoft Security Blog, March 2026.

[12] SecurityWeek. "Chrome Extensions with 900,000 Downloads Caught Stealing AI Chats." SecurityWeek, January 2026.

[13] Wiz. "Agentic Browser Security: 2025 Year-End Review." Wiz Blog, December 2025.

[14] Anthropic. "Mitigating the risk of prompt injections in browser use." Anthropic Research, November 2025.

[15] Cloud Security Alliance. "AI Controls Matrix (AICM) v1.0." CSA, 2024.

[16] Cloud Security Alliance. "Agentic AI Threat Modeling Framework: MAESTRO." CSA Blog, February 2025.

[17] LayerX Security. "Enterprise Browser Extension Security Report 2025." GlobeNewswire, April 15, 2025.