Report | **Feb 2026**

Ninja Browser & Lumma Infostealer

Delivered via Weaponized GOOGLE SERVICES

CTM360®

INDUSTRY	COUNTRY	REGION
Global	Global	Global

OVERVIEW :

CTM360 has identified a threat where cybercriminals are exploiting **Google Groups** and **Google Discussions** to distribute malware. In particular, attackers are embedding malicious download links in Google conversations with catchy keywords that is involved with organization name, disguising them as legitimate software updates or tools.

By leveraging the trust within these Google discussion communities, they trick users into downloading **malware**. This trend highlights the increasing sophistication of social engineering tactics within trusted forums, posing significant risks to both individuals and organizations.

CTM360 continues to monitor, uncover emerging patterns and take action to reduce the spread of these malicious attacks.

Key Findings:

- The global malware campaign is actively exploiting Google services – Google Groups and Google file-hosting platforms such as Docs and Drive. CTM360 team has identified a **sample** of over **4,000 malicious Google Groups** and more than **3,500 Google-hosted URLs (Docs and Drives)** being used to distribute malware.

- The attackers are targeting **organizations worldwide** by embedding their names and relevant keywords within the malicious Google content to increase credibility.

- Malicious redirectors are used to detect and reroute the victim's operating system **(Windows or Linux)** and deliver malware:

  Windows Users – Info-stealer with Lumma family will be dropped once the file executed. Info-stealers behave maliciously in the device by delivering victim's credentials to the threat actor.

  Linux Users – Malicious browsers will be downloaded that feature bogus anonymity features, but are actually designed to enable future compromise while appearing harmless.

Malware Lifecycle: Google Groups Exploitation

Distribution

Google Groups Google Drive

Victims

Lumma Infostealer

Ninja Browser

Windows Infection

Lumma Infostealer Execution
- Browser credentials
- Saved passwords
- Session cookies

System Reconnaissance
- OS version & hardware profiling
- Installed software enumeration

Command & Control (C2)
- Encrypted outbound traffic
- Remote task execution

Persistence Mechanisms
- Registry modification
- Scheduled tasks / startup abuse

Linux Infection

Ninja Browser Deployment
- **Fake Browser Execution**
- Masquerades as legitimate Chromium-based browser
- User-driven credential input

Credential & Session Theft
- Login credentials
- Session tokens

Unauthorized System Access
- Local user compromise
- Potential privilege escalation

Stealthy C2 Communication
- Low-noise outbound connections
- Payload updates

Data Theft & Remote Compromise

Credentials Stolen
- Browser credentials
- Saved passwords
- Session tokens

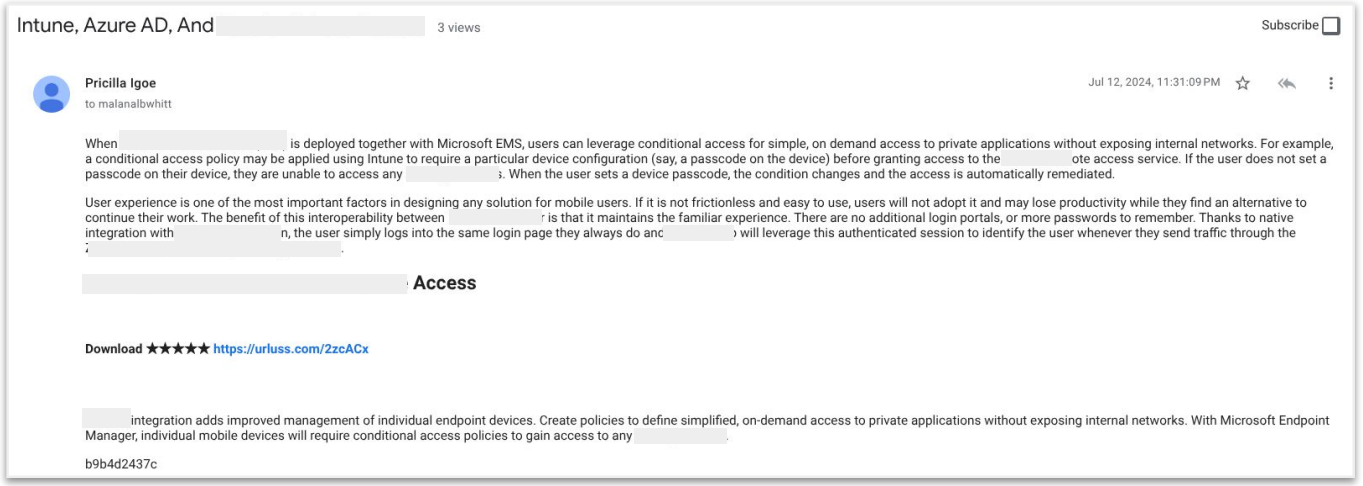
C2 Comnunication
- Data exfiltration
- Encrypted command exchange

Remote Access
- Persistent attacker access
- Follow-on attacks

Observations Summary

Threat actors create **email groups**, posting legitimate-sounding queries to deceive users. The functionality of email groups allow attacker to include **multiple recipient in single group**. Threat actor embeds malicious download links disguised as software updates, often using URL shorteners to evade detection. Clicking these links leads to malware downloads, which install malicious software.

These infections allow attackers to exfiltrate credentials and establish persistence on compromised systems. In enterprise environments, followed up with possible privilege escalation, lateral movement, and attacks on cloud services, leading to data breaches and account takeovers.


Intune, Azure AD, And 3 views

Subscribe

Pricilla Igoe

to malanalbwhitt

Jul 12, 2024, 11:31:09 PM

When is deployed together with Microsoft EMS, users can leverage conditional access for simple, on demand access to private applications without exposing internal networks. For example, a conditional access policy may be applied using Intune to require a particular device configuration (say, a passcode on the device) before granting access to the ote access service. If the user does not set a passcode on their device, they are unable to access any s. When the user sets a device passcode, the condition changes and the access is automatically remediated.

User experience is one of the most important factors in designing any solution for mobile users. If it is not frictionless and easy to use, users will not adopt it and may lose productivity while they find an alternative to continue their work. The benefit of this interoperability between r is that it maintains the familiar experience. There are no additional login portals, or more passwords to remember. Thanks to native integration with n, the user simply logs into the same login page they always do and will leverage this authenticated session to identify the user whenever they send traffic through the

Access

Download ★★★★★ https://urluss.com/2zcACx

integration adds improved management of individual endpoint devices. Create policies to define simplified, on-demand access to private applications without exposing internal networks. With Microsoft Endpoint Manager, individual mobile devices will require conditional access policies to gain access to any

b9b4d2437c

" Google Groups malicious emails with the malware URLs"

Key data:

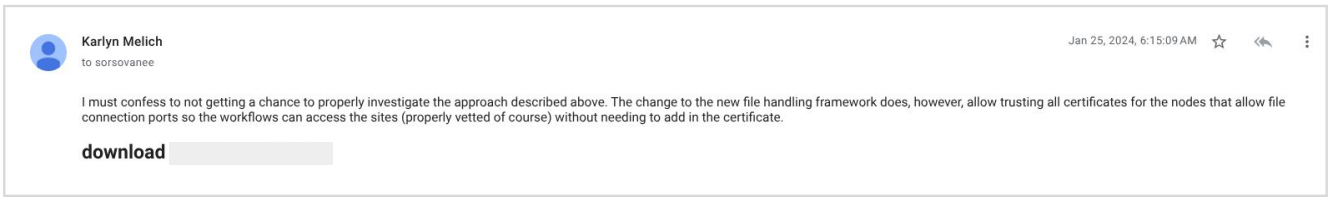
- CTM360's research team, using their tools have detected a **sample** of over **4000+ Google groups targeting global entities.**
- In addition to over **3500+ Google URLs (Docs, Drives)** that distributed the malicious URLs.

Observations Summary

Step-by-Step Breakdown:

Initial Hook – Social Engineering via Google Groups:

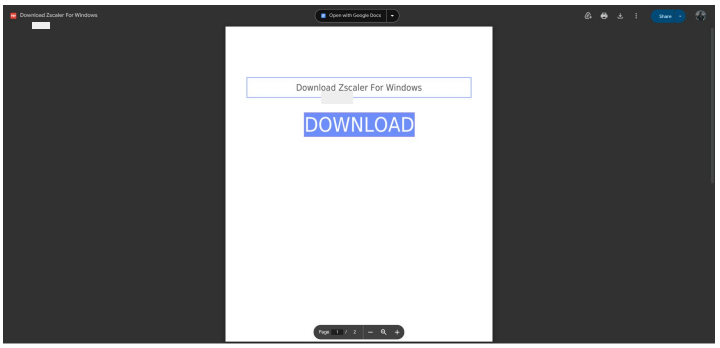
- Threat actors infiltrate Google Groups, forums with relevant keywords related to the targeted industry.
- They post legitimate-sounding technical queries and responses, creating an illusion of authenticity.
- Example discussions include networking issues, migrations, authentication errors, configurations, etc..

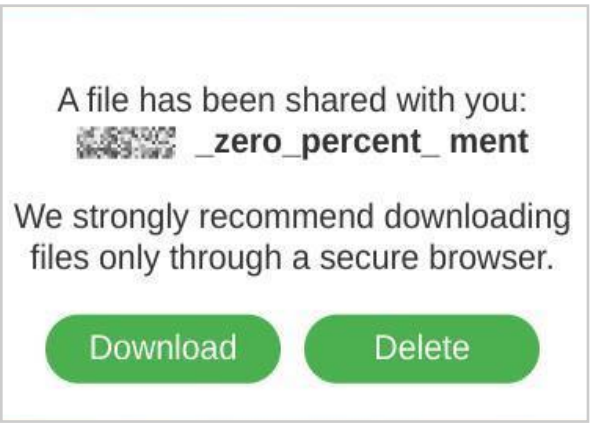


"Fake Emails in the Discussion group resembling legitimate organizations"

Bait – Malicious Download Links:

- Attackers embed fake download links with catchy keywords such as:
- "Download {Organization_name} for Windows 10"
- The embedded URLs presents as two malicious forms.
  - To evade detection, attackers use URL shorteners (e.g., TinyURL, Bitly) and sometimes compromise legitimate websites to host the malicious file.
  - To exploit curiosity, attackers inject malicious redirectors through Google services (Docs, Drives) and share them among group users.





"Malicious Files hosted on Google Drive and mimic file-sharing functionality"

Observations Summary

Post-Interaction – Dual Agent Dropper:

The embedded redirector is gated with two flows that depend on client's user-agent (Windows or Linux).

- **Windows Users (Lumma Dropper) –** will be directed to website that adopt file-sharing function that give the ability to download compressed file with a password protection.
  - hxxps[://]media[.]modecoin[.]xyz/2Million%20Dollars%20Fraud%20At%20Zenith%20Bank,%20Ajose%20Adeogun,%20Victoria%20Island%20-%20Famzn%20News[.]zip

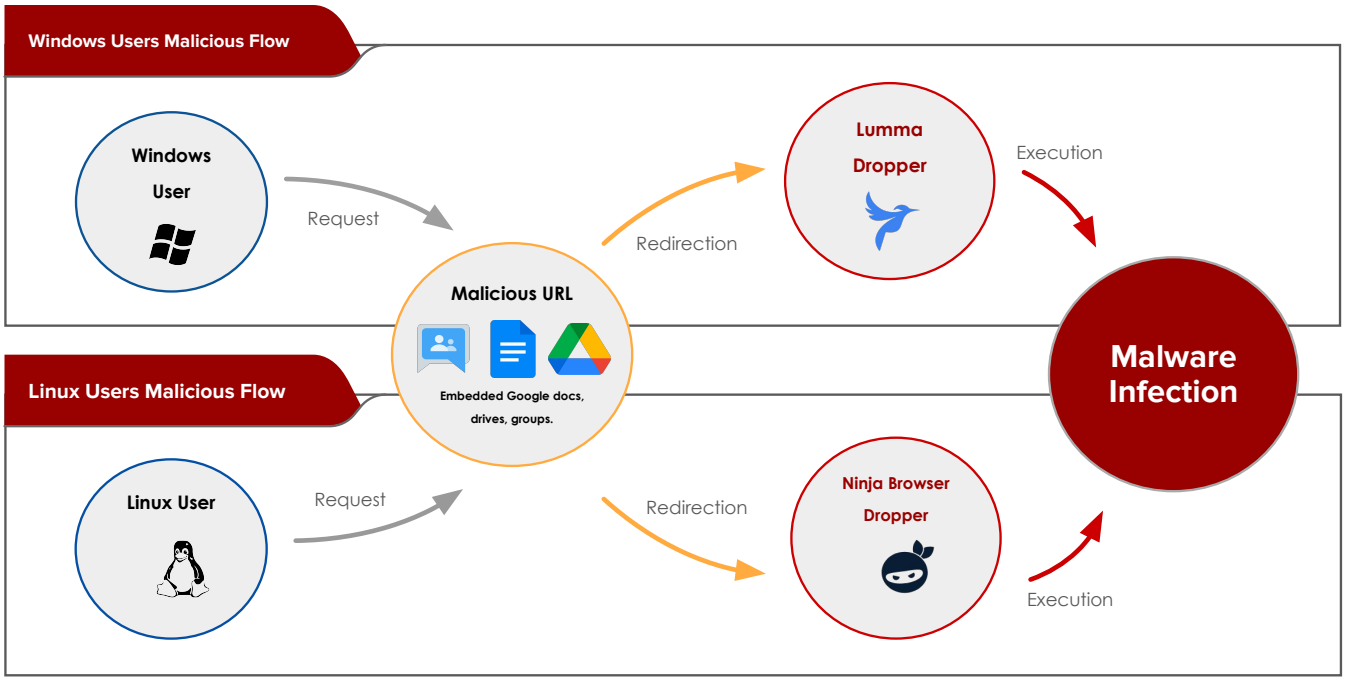
- **Linux Users (Ninja Browser Dropper)** – will be directed to a malicious software that operate as a trojan that mimic browser appearance with an anonymous functionality (built-in VPN extensions, ad-blockers extensions, etc.).
  - hxxps://nbdownload[.]space/setup[.]exe (inactive)
  - hxxps://nb-download[.]com/installer/setup[.]exe

Ninja Browser Functionality

- Bogus anonymity features that is promoted to **deceive users**.
- Install malicious extensions without user's permissions.
  - **Credential stealers** were observed associated with the extensions that aim to exfiltrate victim's credentials to hardened threat actor's C2 servers.
  - **Trojan** malware files where they mimic genuine file appearance although threat actor have remote control over victim's device and exfiltrate victim credentials to C2 server.
- **High persistent** activities on the victim's devices by configuring scheduler that run every 24 hours to update Ninja Browser, and auto-install on new users.
- **Benign activity** for victim's trust resulting a **future compromise** without victim's consist.

Lumma Stealer Functionality

- Bogus file names that **impersonate brand name** and catchy topics.
- **Overwhelm** threat detection solutions with a huge file size (~950MB) to **bypass static anti-virus scans**.
- Execute shell-based commands that **exfiltrate victim's credentials** to threat actor's C2 server.
- Exfiltrated data requests are masked with **encryption layers** to evade detection.



Technical Observations

Linux users – Ninja Browser Interaction

Summary – Malware Deployment:

- Users who click the link download a suspicious executable that appears to be the intended software.
- The software have **trajon** functionality that mimics appearance of browser that offer **anonymity features** with the name “Ninja Browser”.
- The browser has built-in extensions that **act maliciously** in the users devices and has a potential to steals users credentials, and remotely control devices.
- Upon execution, the file trigger suspicious activities, allowing attackers to:
  - **Exfiltrate stored credentials** from browsers.
  - Maintain **long-term access** by deploying long-sleep processes and scheduled tasks. These schedulers are configured to **periodically poll the threat actor-controlled website for updates**, which are automatically installed upon update availability —without the user’s knowledge or consent.
  - **Modify system configurations**, allowing further exploitation, & lateral movement.
- These activities allow threat actor potentially to compromise victims later in an **unexpected malicious update** that is scheduled pointing to the threat actor’s server.

Findings & Malicious File Analysis:

Upon decompressing the file, it contains a Portable Executable file called "Ninja Browser." The term "Ninja Browser" is usually used when the browser does not allow user data to be tracked. After investigating the browser, we observed suspicious behavior associated with it. These unusual activities matched malicious signatures with high entropy that behave via outbound requests.

The red flag highlighted the **unusual permissions** through built-in extensions that have been observed with high obfuscation. Through the backend, it appears that the browser functions on a Chromium browser, expanding the potential impact on the victim's device to exposure of credentials stored on the default browser.

The methodology utilized in this software relies on building trust in the software whereas it contains embedded code within a built-in extension with the name “NinjaBrowserMonetisation”.

```
▼<app appid="kaajboeeieaofkhnmabfbfdhpmnmadba">
   <updatecheck codebase="https://d.ninja-browser.com/extensions/NinjaBrowserMonetisation.crx" version="1.0.0.2"/>
  </app>
▼<app appid="nkeopbillbcaikmmbegdblclaninjplo">
   <updatecheck codebase="https://d.ninja-browser.com/extensions/TorrentSearch.crx" version="2.0.0.9883"/>
  </app>
▼<app appid="lkfkklbadhfhpgbpnpiekemiepdmojmd">
   <updatecheck codebase="https://d.ninja-browser.com/extensions/uBlock-Origin.crx" version="1.59.2"/>
  </app>
▼<app appid="eimcpgmlcjealoeaidbmjnmpmmhaegkj">
   <updatecheck codebase="https://d.ninja-browser.com/extensions/XFinderSearch.crx" version="2.0.0.5"/>
  </app>
▼<app appid="doekbkecialdopibjcepmifjfdangjff">
   <updatecheck codebase="https://d.ninja-browser.com/extensions/YouTubeDownloader.crx" version="1.6.34"/>
  </app>
</gupdate>
```

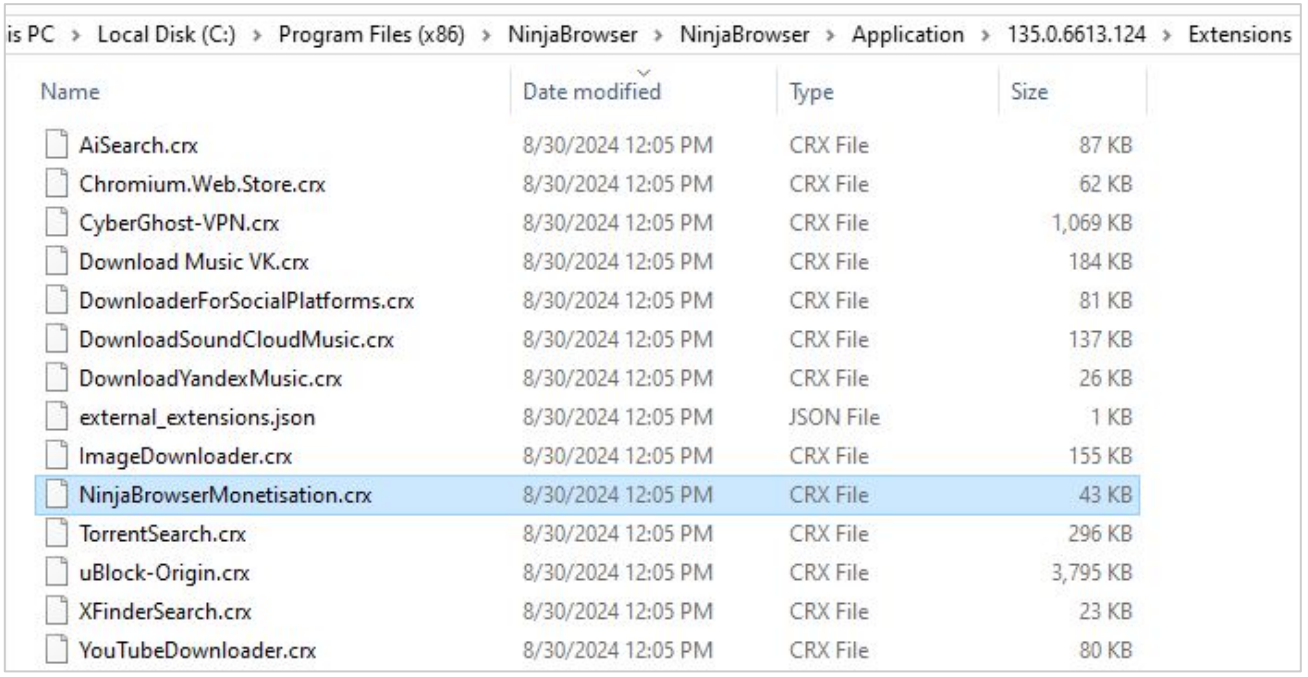
“The installed extensions by the threat actor to the browser from server-side view”

Technical Observations

Linux users – Ninja Browser Interaction

Subfiles Inspection – Built-in Extensions:

Despite the anonymity features claimed on the software license, the "Ninja Browser" functionality proved the opposite by **auto-installing extensions** on the "Ninja Browser" that function on ungoogled-chromium engine.



is PC › Local Disk (C:) › Program Files (x86) › NinjaBrowser › NinjaBrowser › Application › 135.0.6613.124 › Extensions

Name	Date modified	Type	Size
AiSearch.crx	8/30/2024 12:05 PM	CRX File	87 KB
Chromium.Web.Store.crx	8/30/2024 12:05 PM	CRX File	62 KB
CyberGhost-VPN.crx	8/30/2024 12:05 PM	CRX File	1,069 KB
Download Music VK.crx	8/30/2024 12:05 PM	CRX File	184 KB
DownloaderForSocialPlatforms.crx	8/30/2024 12:05 PM	CRX File	81 KB
DownloadSoundCloudMusic.crx	8/30/2024 12:05 PM	CRX File	137 KB
DownloadYandexMusic.crx	8/30/2024 12:05 PM	CRX File	26 KB
external_extensions.json	8/30/2024 12:05 PM	JSON File	1 KB
ImageDownloader.crx	8/30/2024 12:05 PM	CRX File	155 KB
NinjaBrowserMonetisation.crx	8/30/2024 12:05 PM	CRX File	43 KB
TorrentSearch.crx	8/30/2024 12:05 PM	CRX File	296 KB
uBlock-Origin.crx	8/30/2024 12:05 PM	CRX File	3,795 KB
XFinderSearch.crx	8/30/2024 12:05 PM	CRX File	23 KB
YouTubeDownloader.crx	8/30/2024 12:05 PM	CRX File	80 KB

"The installed extensions once the Ninja Browser executed"

The extension **"NinjaBrowserMonetisation"** contain a **JavaScript file with a malveristing** functions that have the following capabilities:

- Track users with a unique IDs.
- Inject scripts.
- Load remote content.
- Manipulate browser tabs and cookies.
- Store data in an external storage file.

Foremost, these data were intended to be delivered to an external hosts that are embedded on the extension with a high obfuscation.

n Files (x86)\NinjaBrowser\NinjaBrowser\Applicati

View

al Disk (C:) › Program Files (x86) › NinjaBrows

Name

_locales
icons
main.js
manifest.json

```
ackgroundMode'](),_0x308b03=_0x529956['extensionMode'](),_0x4f182c=(_0x572636['.']={})['a']=function(
return _0x2e37c8&&_0x2e37c8['length']?_0x490a72['parseFloat'](_0x2e37c8[-0x1fa8+0x956+-0x5*-0x477]||'
']](_0xf0589e);try{for(var _0xd769c2=0x36+-0x1fda+0x1e*0x10e,_0x47e911=_0x568369['length'];_0xd769c2<_
le'][trace']();break;case-0x1212+0x8f7+0x91c:_0x193631['warn'](_0x5037df);break;case-0x84a+-0x51a*0x
bfb){_0x529956['ChromeRefHijack']=!(-0x1*-0x23df+-0x318+-0x20c7);var _0x3c0dfd=_0x529956['JSON']=_0xe
b(),_0x529956['Sync']=_0x747144(),_0x529956['History']=_0x5a1d03(),_0x529956['Modules']=_0x1311e0(),_
{var _0x3eba86=document,_0x71aadb='('+function(){Object['defineProperty'](document,'referrer',{'enume
,!(0x597+0x136f+-0x1906),_0x17114d),_0x1481fc['dispatchEvent'](_0x4f1176));}var _0x562b67=function(){
7+-0xba2+-0xe05):setTimeout(_0x5f161c,0x13f5*-0x1+-0x1*-0x11b5+0x24a);}));};_0x5f161c();}else exchgId
));

if (chrome && chrome.hasOwnProperty('runtime') && chrome.runtime.hasOwnProperty('onInstalled')) {
        chrome.runtime.onInstalled.addListener(function(details) {
            if (details.reason == "install")
            {
                fetch('https://lbpx.pathway-link.com/pixel/i?product=ninja_browser').catch(e => {});
            }
        });
}
```

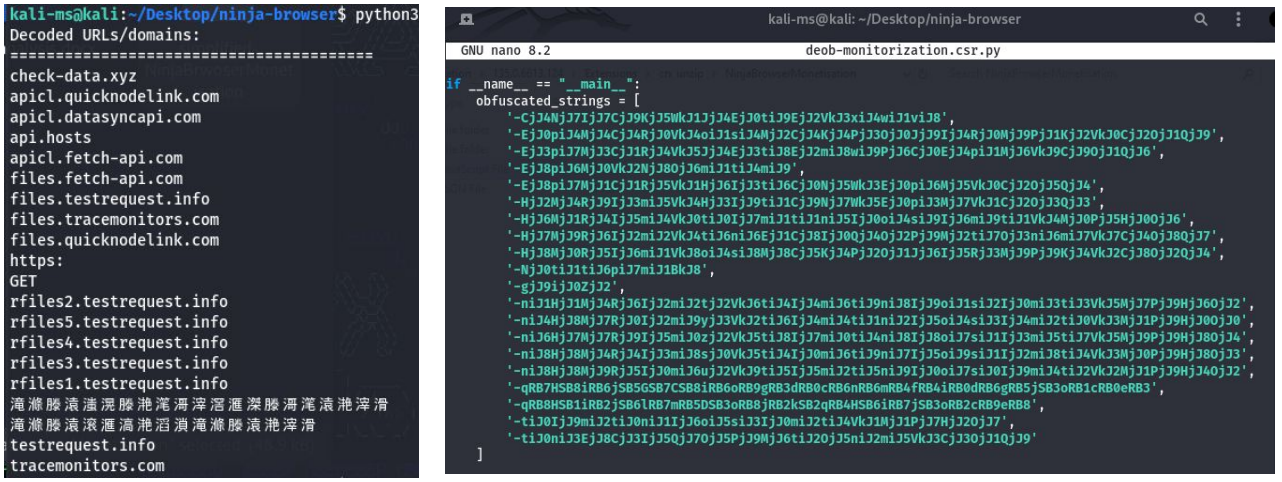
"NinjaBrowserMonetisation extension's content with malicious JavaScript code"

Technical Observations

Linux users – Ninja Browser Interaction

Deobfuscation – Embedded URLs:

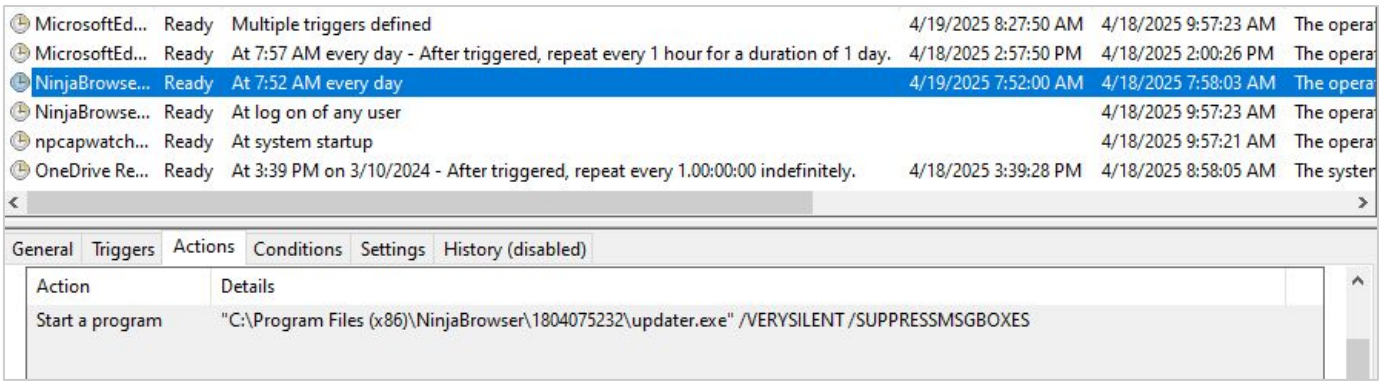
To reveal the embedded URLs on Ninja Browser Monetisation extension. The URLs were highly obfuscated with **XOR and Base56-like encryption** that complicate the detection process on threat detection solutions.



"Deobfuscation process of the embedded URLs on NinjaBrowserMonetisation extension"

The majority of these **Domains** were observed dedicatedly utilized on malicious campaigns that are relevant to **info-stealer activities (e.g. RedLine)** reveal the potential for malicious trigger upcoming to all devices with Ninja Browser installed. The fact of all these indicators existence without functionally using the extension currently on the browser raise the suspect of a handy exfiltration tool that threat actor might utilize later on a update.

Considering the previous activity of configuring **schedulers that run on a daily-basis**, these schedulers are pointing to browser update PE file that install silently from ninja-browser website without installation window appearance. This assure the persistence within the software and awaiting for a payload update to compromise all devices.



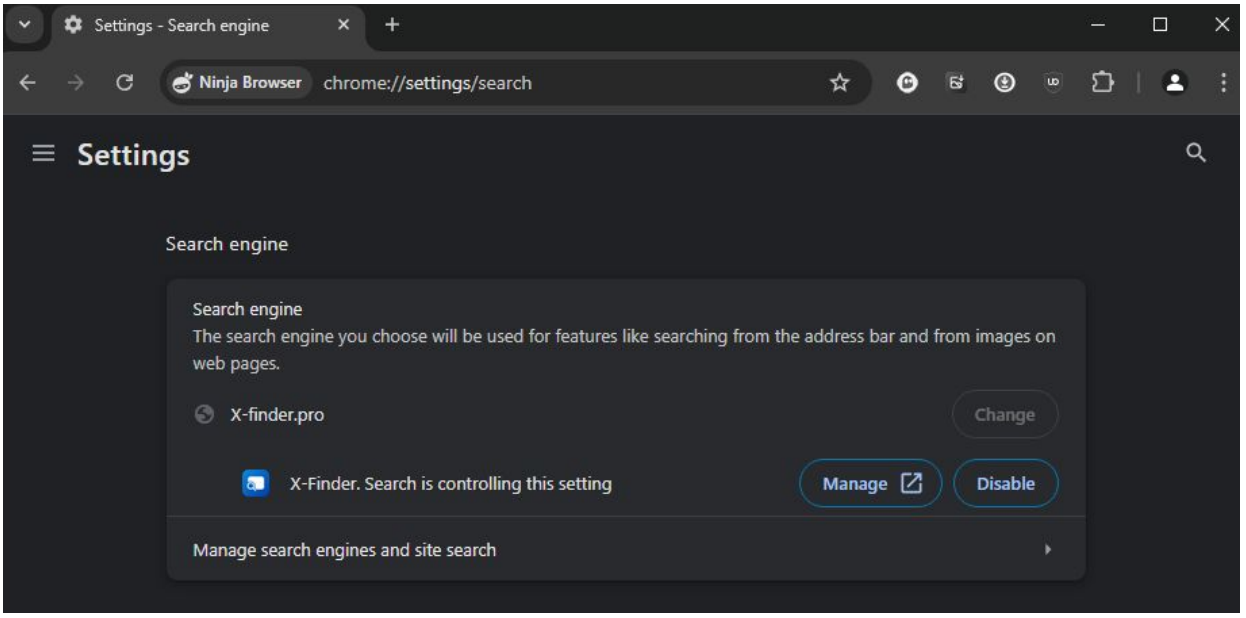
"Update schedulers configured once the Ninja Browser Installed"

Technical Observations

Linux users – Ninja Browser Interaction

Suspicious – Uncommon Indicators:

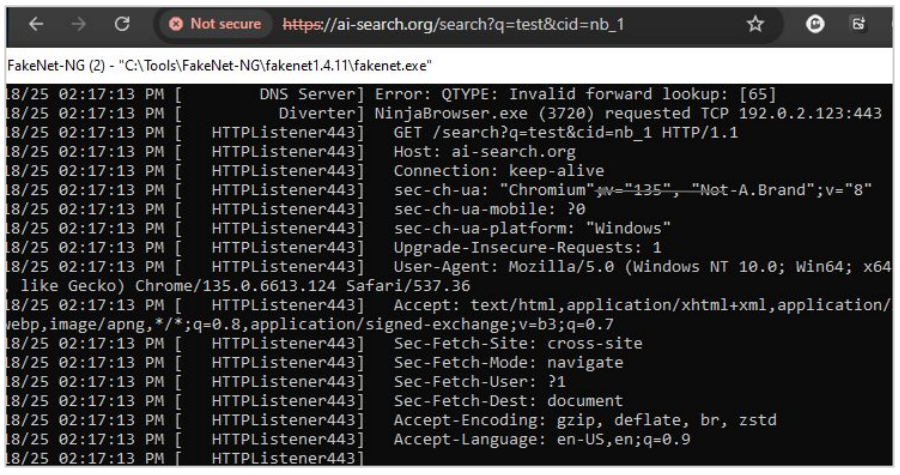
Through simulating user behavior to detect suspicious flags, it has been observed that the default search engine is set to a **Russian-based search engine** with the name of "X-Finder" that is unknown on the industry with a simple layout that contain only a search bar and render Google results once search compiled.

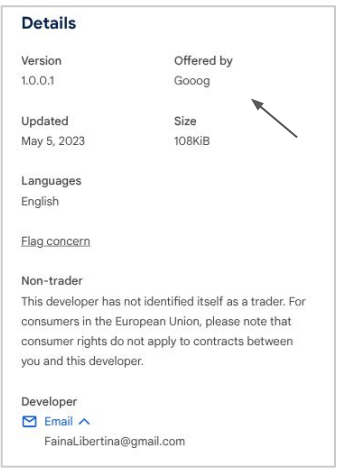


"X-Finder is utilized as a default search engine on Ninja Browser"

In addition to X-finder, another search engine that observed popping up once accessing the browser with a fake AI features. The search engine goes by **"ai-search"** is also offering an extension that is published on Chrome Web Store with a **deceptive details** falsely altering Google name on creator details.

- https://chromewebstore.google.com/detail/chatgpt-in-new-tab/dcdioeaeejohhgbpobifnhbdjnmgeejl





"Ai-search is used on the landing page of Ninja Browser with deceptive creator details"

Technical Observations

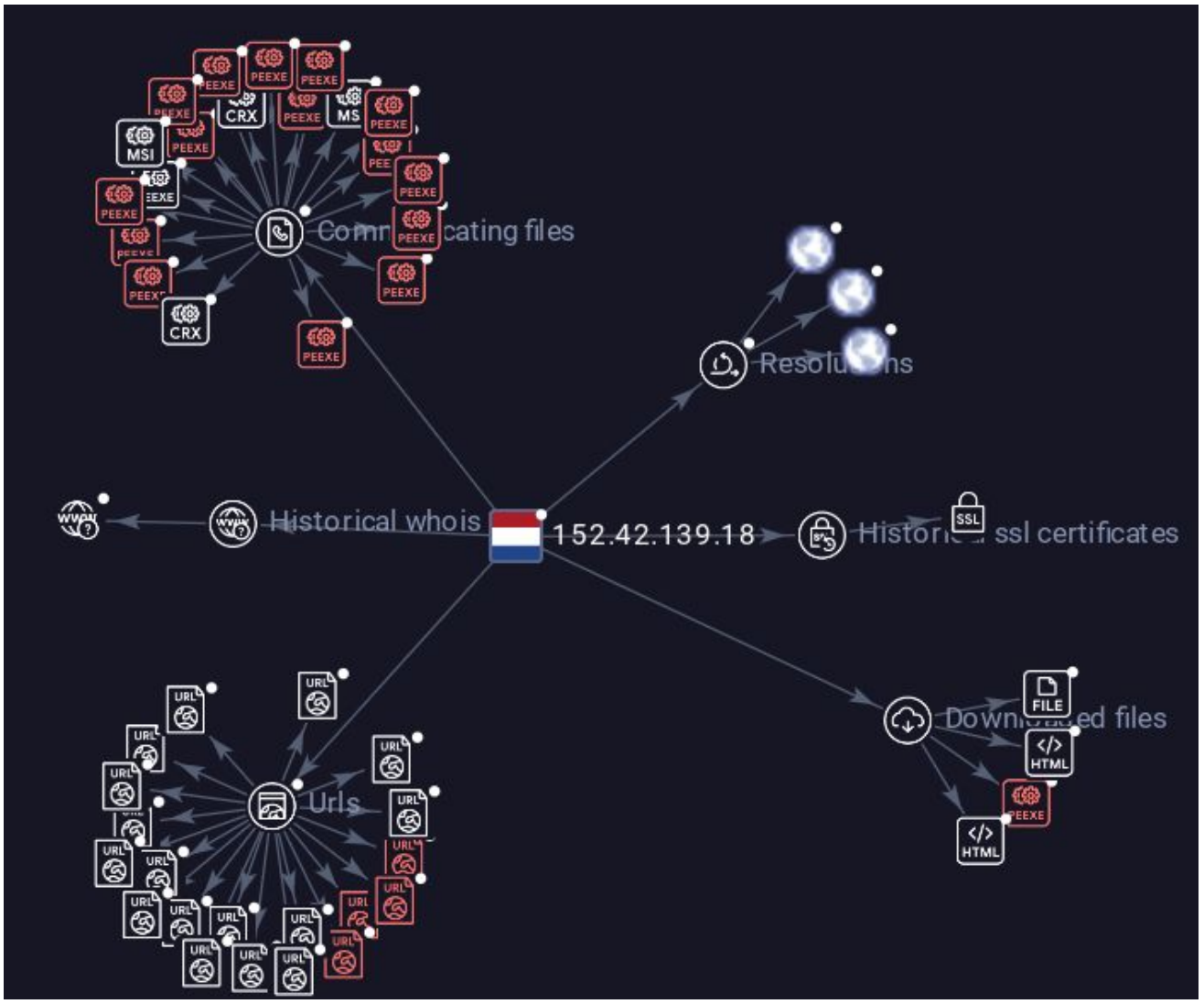
Linux users – Ninja Browser Interaction

Intelligence – Threat Actor's Identifiers:

Additionally, we have observed that the threat actor has been utilizing a mainly a dedicated IP "152.42.139[.]18" to conduct this malicious activity. Through this IP connected with the malicious files, websites, and SSL were transmitting data back and forth to the server used by the threat actor, illustrating the intrusive of the campaign by the same threat actor.

Furthermore, additional URLs were observed utilized to distribute Ninja Browser and all primarily call back to a single website with the name "ninja-browser[.]com"

- nb-download[.]com ➔ "152.42.139[.]18"
- nbdownload[.]space ➔ "89.111.170[.]100"



"Screenshot from VirusTotal highlighting the malicious files connected to the IP"

Technical Observations – Ninja Browser IoCs

Linux users – Ninja Browser Interaction

Indicator of Compromise (IoCs)

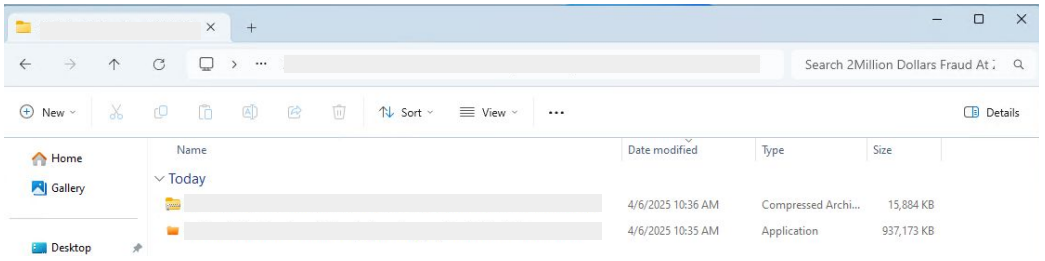
- 89.111.170[.]100
- 152.42.139[.]18
- nbdownload[.]space
- nb-download[.]com
- ninja-browser[.]com
- check-data[.]xyz
- apicl[.]datasyncapi[.]com
- datasyncapi[.]com
- apicl[.]fetch-api[.]com
- files[.]fetch-api[.]com
- fetch-api[.]com
- files[.]tracemonitors[.]com
- tracemonitors[.]com
- files[.]quicknodelink[.]com
- apicl[.]quicknodelink[.]com
- quicknodelink[.]com
- rfiles1[.]testrequest[.]info
- rfiles2[.]testrequest[.]info
- rfiles3[.]testrequest[.]info
- rfiles4[.]testrequest[.]info
- rfiles5[.]testrequest[.]info
- files[.]testrequest[.]info
- testrequest[.]info
- 77d44715bd0ead14867d5c46dba2e6e43b8bc303e88da5309ab60f9828d1c876
- 50be18615cb70a8edfd12e005a025fa94a927807412f48d4e2c1d25b97ed0961
- 3c32716a705717c723e969fd8cc095012f17834cb67d813569f2b1841139d289
- 0f7e470bc92df3320db7aa731a39a73b7a1a394f816564783611861e2cbd5286
- 0b78456a6c1fb8aa77bb2d475ab3815a04a86333111e54bf2a9f69968bbc73c8
- 6e6a323bdfddb54407a075c887a6adef9f3c5fa1e5e81d51dc49ab61846637c6
- 738aa2649686dd0c024bd3ae79cb9e346985859d141ae6615ddd26e3e168b4d8
- 7636a06e18f08b002367e059481eb5585b0aa728c56f01ad8f0ddf489de8dbc6
- 77d44715bd0ead14867d5c46dba2e6e43b8bc303e88da5309ab60f9828d1c876
- 7ab1402b55498ca9e67477958d6b8794f00f2d814f6af37803329f673616bfe2
- c58481335431bcd69398a7e1328f209c34652355af4df62cffbe2da92fe6a416
- ce7d527e68218e33db10361443f54a37f7a4825c3fb89682d5c859940d6d6f91
- f42c249be9b3e68e76ac660b8a189663e30ccd311a763c9078f2f8682b33f05c
- fa9ed218a8056b6142de273cb36bfd840e7d894c90cec8981603b233bb877d36

Technical Observations

Windows users – Lumma Interaction

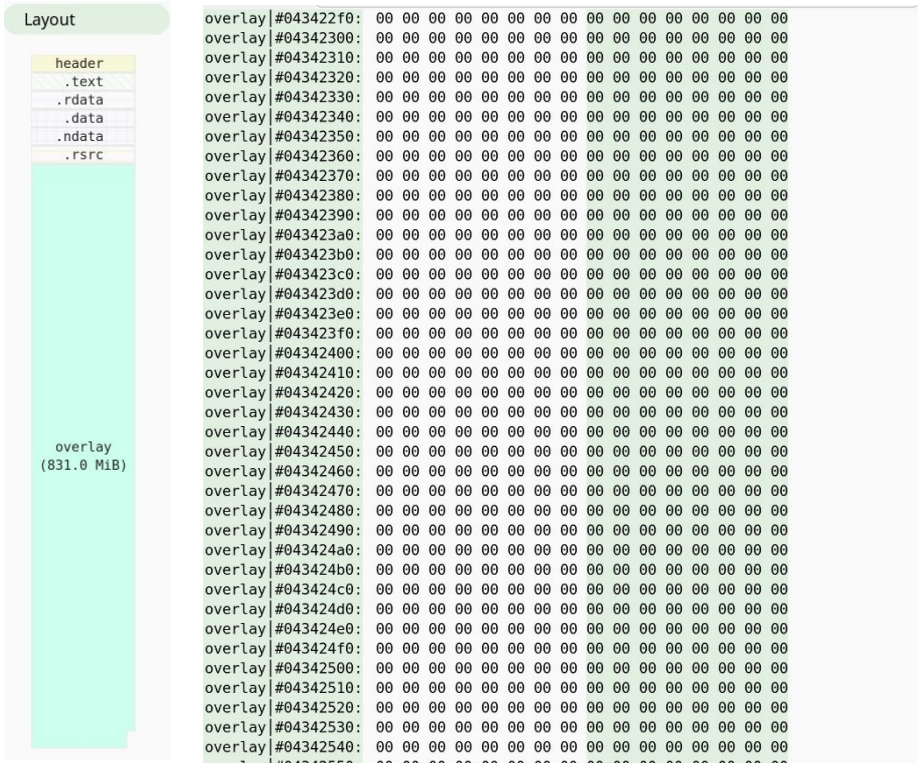
Initial Access Vector & Compressed File Analysis:

- The malicious file will be drop once the user access the bogus file-sharing sites “media[.]modecoin[.]xyz”.
- The malicious file is an archive that is compressed with a password.
  - Decompressed Size: ~950MB
  - Actual size after removing padding: ~33MB



The unusually large decompressed size of approximately **950MB raised a red flag**. Upon analyzing the Portable Executable (PE) content, we found that the file was heavily padded with **null bytes** a common technique used to evade static detection by crashing virus detection tools due to surpass the scannable file size limit.

Anti-virus solutions tend to set a size limit for the analyzable files where threat actor exploited the limitation by expanding the malicious file size.



“The observed null byte padding on the malicious file”

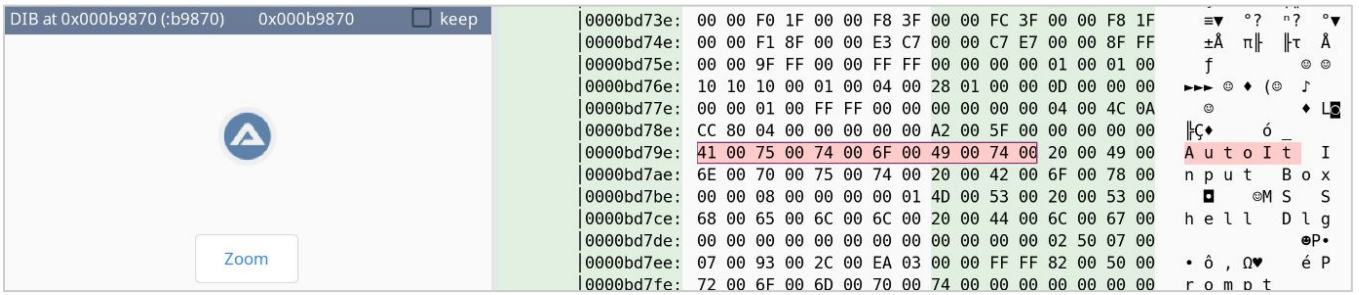
Technical Observations

Windows users – Lumma Interaction

Static Analysis of Reconstructed Files:

Since the executable file is built and compressed with “Autoit” and compiled with “NSIS”. it contained a segmented binary structure of:

- file0001.bin to file0014.bin
- script.bin

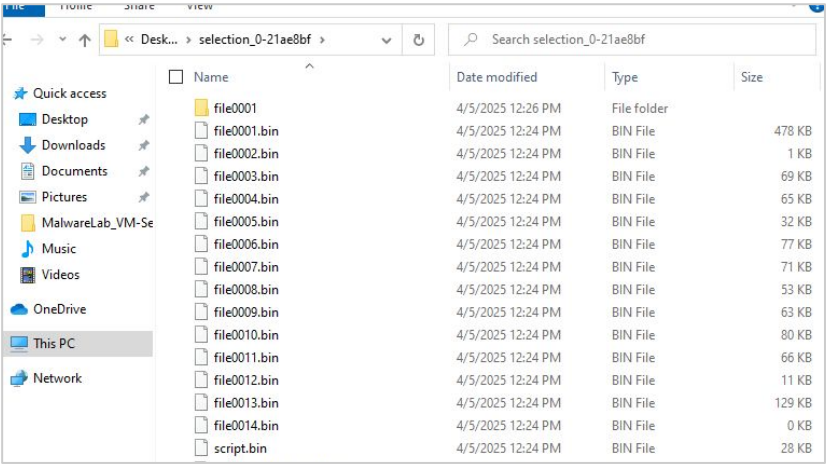


“Auto It metadata existence on the malware file”

The file script.bin include batch login to reassemble the executable through chained copy commands:

- cmd /c copy /b ..\Playlist.csv + ..\Trans.csv + ..\Mentioned.csv + ..\Flexible.csv R

These files are eventually concatenated into a single executable file to launch the malicious script.

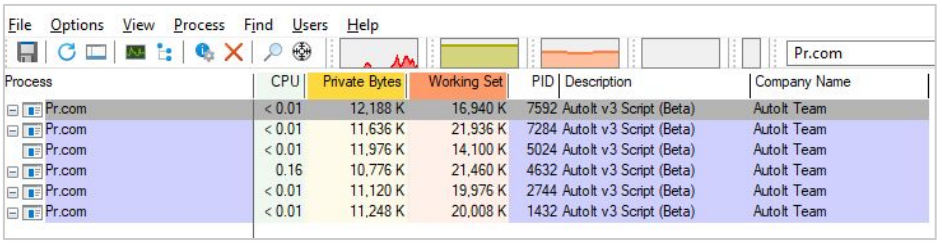


“Files content structures of the malware file”

Execution and AutoIt Behavior:

Once reassembled, the binary launches an AutoIt-compiled executable, loading a .a3x script. This script decrypts and executes a memory-resident payload. The process named **“Pr . com”** is observed to trigger as AutoIt v3 script stubs.

AutoIt is a scripting language designed for automating the Windows GUI and general scripting.



“AutoIt meta-data assembled on the malware processor”

Technical Observations

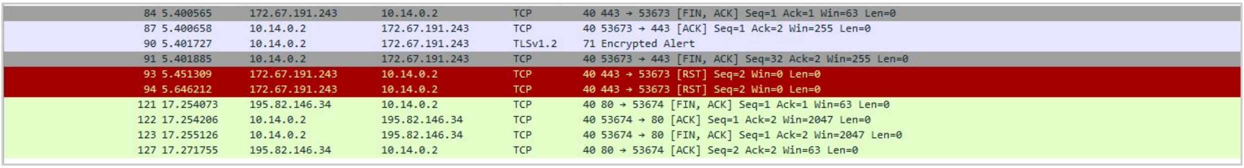
Windows users – Lumma Interaction

Dynamic Behavior and Network Analysis

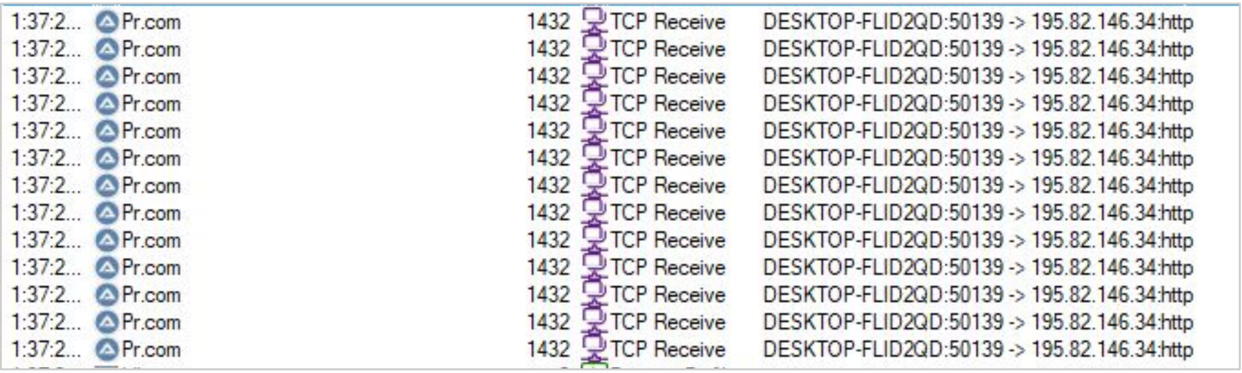
Process creation and behavioral monitoring reveal the following:

- Shell-based command chaining
- Execution of multiple AutoIt processes
- HTTP POST requests to:
    - healgeni[.]live

Furthermore, HTTP traffic were captured to include encoded identifiers and payload data. The use of multipart/form-data POST requests is utilized to mask the exfiltrated content.



84	5.400565	172.67.191.243	10.14.0.2	TCP	40 443 → 53673 [FIN, ACK] Seq=1 Ack=1 Win=63 Len=0
87	5.400658	10.14.0.2	172.67.191.243	TCP	40 53673 → 443 [ACK] Seq=1 Ack=2 Win=255 Len=0
90	5.401727	10.14.0.2	172.67.191.243	TLSv1.2	71 Encrypted Alert
91	5.401885	10.14.0.2	172.67.191.243	TCP	40 53673 → 443 [FIN, ACK] Seq=32 Ack=2 Win=255 Len=0
93	5.451309	172.67.191.243	10.14.0.2	TCP	40 443 → 53673 [RST] Seq=2 Win=0 Len=0
94	5.646212	172.67.191.243	10.14.0.2	TCP	40 443 → 53673 [RST] Seq=2 Win=0 Len=0
121	17.254073	195.82.146.34	10.14.0.2	TCP	40 80 → 53674 [FIN, ACK] Seq=1 Ack=1 Win=63 Len=0
122	17.254206	10.14.0.2	195.82.146.34	TCP	40 53674 → 80 [ACK] Seq=1 Ack=2 Win=2047 Len=0
123	17.255126	10.14.0.2	195.82.146.34	TCP	40 53674 → 80 [FIN, ACK] Seq=1 Ack=2 Win=2047 Len=0
127	17.271755	195.82.146.34	10.14.0.2	TCP	40 80 → 53674 [ACK] Seq=2 Ack=2 Win=63 Len=0



1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http
1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http
1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http
1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http
1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http
1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http
1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http
1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http
1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http
1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http
1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http
1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http
1:37:2...	Pr.com	1432	TCP Receive	DESKTOP-FLID2QD:50139 -> 195.82.146.34:http

"The malicious processor's requests to C2 server"

Behavioral Conclusion and IoCs

The file behavior has been found to match Lumma info-stealer malware that allows the threat actor to gain access to the victim's device and direct the victim's browser credentials to the threat actor's C2 server. Several obfuscation and anti-analysis layers are implemented to evade detection of the executed malware.

C2 Domain

- healgeni[.]live

Observed IPs

- 195.82.146[.]34 (Vault Dweller)
- 104.21.68[.]78 (Cloudflare)
- 172.67.191[.]243 (Cloudflare)

Sample Hash (SHA-256):

- adbcad85bbb372a77da00e528bf564b1d82ad61e2956d902354197d40b11fd7f

Sample of the Targeted Brands

Conclusion

Threat actor continuously look after a method to deceive users into installing harmful software to carry their malicious activities. Despite Google service reputation, unsuspected individual might found themselves a victim of a malicious attack that target their sensitive data, credentials, or to launch another attacks with the victim identity. Therefore, users should practice caution and stay vigilant by avoiding unsolicited downloads, verifying sender information and adopting security solutions.

Risk Involved

Lumma Info-Stealer Risks:

Lumma Stealer is a commercially developed infostealer sold on underground forums and used by a wide range of threat actors. It's designed to quietly harvest sensitive artifacts from compromised endpoints and then exfiltrate those "logs" to remote command-and-control infrastructure for immediate misuse or resale. The associated risks of Lumma Stealer are:

- ***Credential & Session Theft:*** *Harvested passwords, cookies and tokens enable account takeover without needing the account owner's password.*
- ***Financial Fraud:*** *Access to banking, payment, and e-commerce accounts leads to unauthorized transactions and chargebacks.*

Ninja Browser Risks:

Ninja Browser presents itself as a legitimate, privacy-focused browser and performs **benign activity** to build the **victim's trust** - displaying a working UI and normal browsing features.

Ninja Browser masks hidden malicious modules and built-in extensions that have the ability to quietly harvest credentials, install persistent long-sleep schedulers, and periodically fetch updates from attacker servers - enabling a **future compromise without the victim's knowledge or consent**. The associated potential risks of Ninja Browser are:

- ***Credential & session theft:*** *Built-in malicious extensions and injected pages can harvest saved passwords, cookies and session tokens from real browsers, enabling account takeover.*
- ***Remote control / full endpoint compromise:*** *The software can install backdoor-like components that allow remote execution, arbitrary commands, or remote desktop control.*
- ***Silent persistence & automatic malicious updates:*** *Long-sleep processes and scheduled tasks ensure the malware survives deletion and can automatically fetch and install new payloads from attacker servers.*

ABOUT US

CTM360 offers a consolidated security platform on a subscription basis. Through a single environment, CTM360 provides External Attack Surface Management, External Exposure Management, Third-Party Risk Management, Digital Risk Protection, Cyber Threat Intelligence, DMARC, and Takedowns. End-users may subscribe to one or multiple modules; there are no installation or configuration requirements on the end-user side, and CTM360 pre-populates the full platform with publicly available data.

CONTACT US:

+973 77 360 360

info@ctm360.com

www.ctm360.com

21st Floor, East Tower Bahrain Financial Harbour, Kingdom of Bahrain

CTM360®