Identifying PrivateLoader Network Threats

By Sam Lister

Published: 2022-07-26 · Archived: 2026-04-05 20:00:09 UTC

Instead of delivering their malicious payloads themselves, threat actors can pay certain cybercriminals (known as pay-per-install (PPI) providers) to deliver their payloads for them. Since January 2022, Darktrace's SOC has observed several cases of PPI providers delivering their clients' payloads using a modular malware downloader known as 'PrivateLoader'.

This blog will explore how these PPI providers installed PrivateLoader onto systems and outline the steps which the infected PrivateLoader bots took to install further malicious payloads. The details provided here are intended to provide insight into the operations of PrivateLoader and to assist security teams in identifying PrivateLoader bots within their own networks.

Threat Summary

Between January and June 2022, Darktrace identified the following sequence of network behaviours within the environments of several Darktrace clients. Patterns of activity involving these steps are paradigmatic examples of PrivateLoader activity:

1. A victim's device is redirected to a page which instructs them to download a password-protected archive file from a file storage service — typically Discord Content Delivery Network (CDN)

2. The device contacts a file storage service (typically Discord CDN) via SSL connections

3. The device either contacts Pastebin via SSL connections, makes an HTTP GET request with the URI string '/server.txt' or 'server_p.txt' to 45.144.225[.]57, or makes an HTTP GET request with the URI string '/proxies.txt' to 212.193.30[.]45

4. The device makes an HTTP GET request with the URI string '/base/api/statistics.php' to either 212.193.30[.]21, 85.202.169[.]116, 2.56.56[.]126 or 2.56.59[.]42

5. The device contacts a file storage service (typically Discord CDN) via SSL connections

6. The device makes a HTTP POST request with the URI string '/base/api/getData.php' to either 212.193.30[.]21, 85.202.169[.]116, 2.56.56[.]126 or 2.56.59[.]42

7. The device finally downloads malicious payloads from a variety of endpoints

The PPI Business

Before exploring PrivateLoader in more detail, the pay-per-install (PPI) business should be contextualized. This consists of two parties:

1. PPI clients - actors who want their malicious payloads to be installed onto a large number of target systems. PPI clients are typically entry-level threat actors who seek to widely distribute commodity malware [1]

2. PPI providers - actors who PPI clients can pay to install their malicious payloads

As the smugglers of the cybercriminal world, PPI providers typically advertise their malware delivery services on underground web forums. In some cases, PPI services can even be accessed via Clearnet websites such as InstallBest and InstallShop [2] (Figure 1).

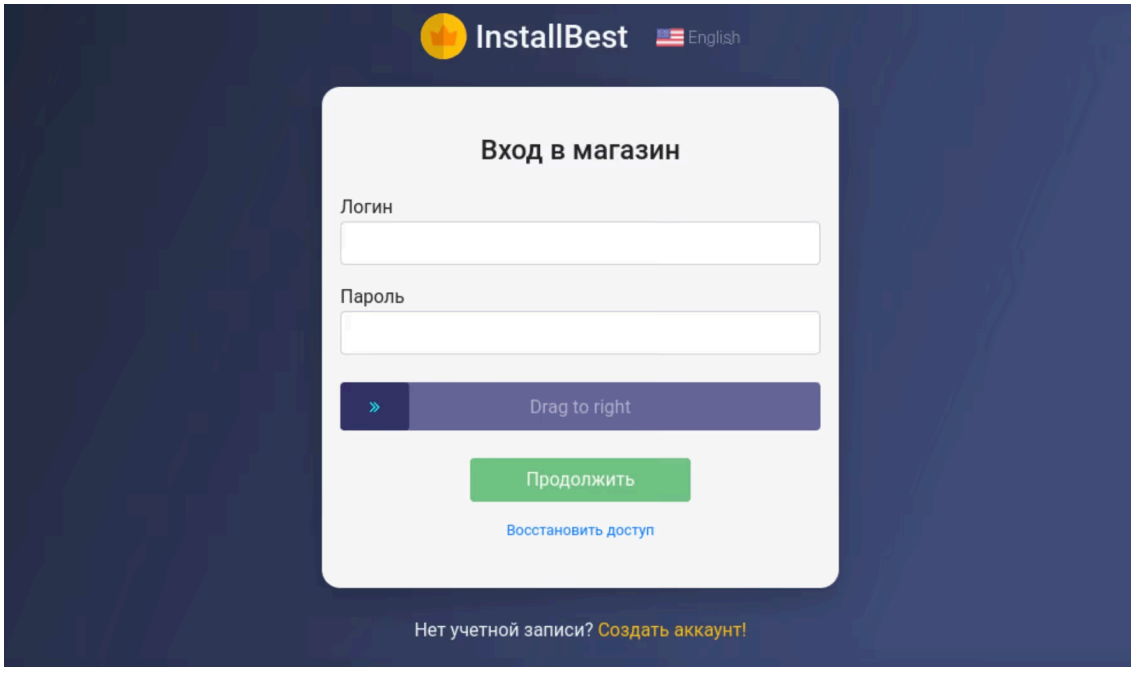


Figure 1: A snapshot of the InstallBest PPI login page [2]

To utilize a PPI provider's service, a PPI client must typically specify:

(A) the URLs of the payloads which they want to be installed

(B) the number of systems onto which they want their payloads to be installed

(C) their geographical targeting preferences.

Payment of course, is also required. To fulfil their clients' requests, PPI providers typically make use of downloaders - malware which instructs the devices on which it is running to download and execute further payloads. PPI providers seek to install their downloaders onto as many systems as possible. Follow-on payloads are usually determined by system information garnered and relayed back to the PPI providers' command and control (C2) infrastructure. PPI providers may disseminate their downloaders themselves, or they may outsource the dissemination to third parties called 'affiliates' [3].

Back in May 2021, Intel 471 researchers became aware of PPI providers using a novel downloader (dubbed 'PrivateLoader') to conduct their operations. Since Intel 471's public disclosure of the downloader back in Feb 2022 [4], several other threat research teams, such as the Walmart Cyber Intel Team [5], Zscaler ThreatLabz [6], and Trend Micro Research [7] have all provided valuable insights into the downloader's behaviour.

Anatomy of a PrivateLoader Infection

The PrivateLoader downloader, which is written in C++, was originally monolithic (i.e, consisted of only one module). At some point, however, the downloader became modular (i.e, consisting of multiple modules). The modules communicate via HTTP and employ various anti-analysis methods. PrivateLoader currently consists of the following three modules [8]:

- The loader module: Instructs the system on which it is running to retrieve the IP address of the main C2 server and to download and execute the PrivateLoader core module
- The core module: Instructs the system on which it is running to send system information to the main C2 server, to download and execute further malicious payloads, and to relay information regarding installed payloads back to the main C2 server
- The service module: Instructs the system on which it is running to keep the PrivateLoader modules running

Kill Chain Deep-Dive

The chain of activity starts with the user's browser being redirected to a webpage which instructs them to download a password-protected archive file from a file storage service such as Discord CDN. Discord is a popular VoIP and instant messaging service, and Discord CDN is the service's CDN infrastructure. In several cases, the webpages to which users' browsers were redirected were hosted on 'hero-files[.]com' (Figure 2), 'qd-files[.]com', and 'pu-file[.]com' (Figure 3).

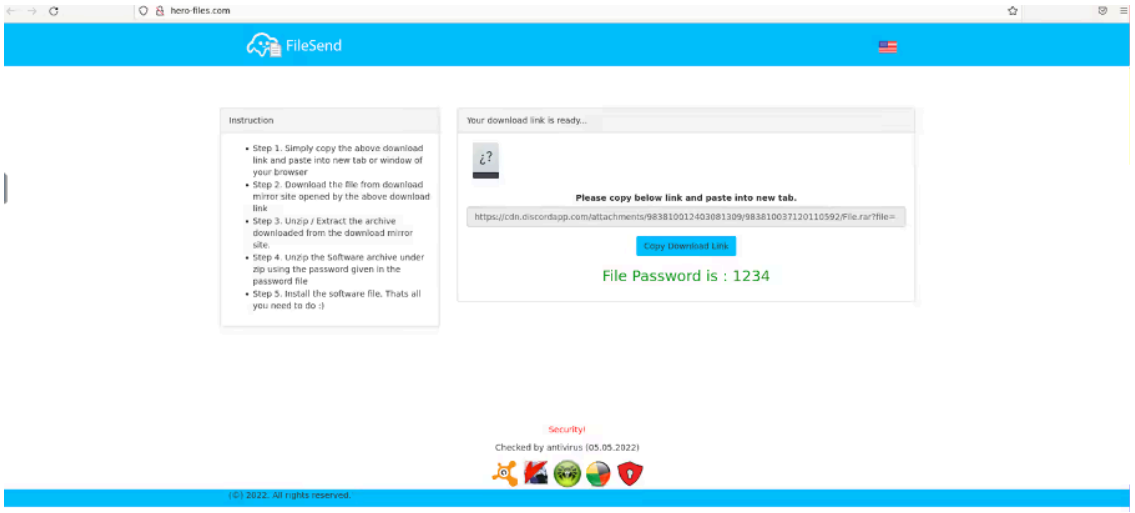


Figure 2: An image of a page hosted on hero-files[.]com - an endpoint which Darktrace observed systems contacting before downloading PrivateLoader from Discord CDN

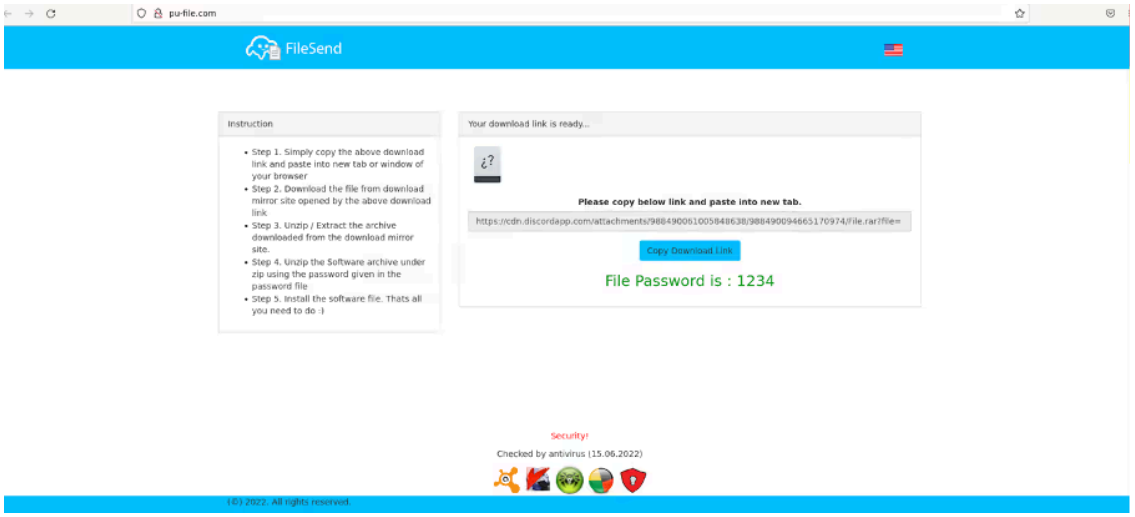


Figure 3: An image of a page hosted on pu-file[.]com- an endpoint which Darktrace observed systems contacting before downloading PrivateLoader from Discord CDN

On attempting to download cracked/pirated software, users' browsers were typically redirected to download instruction pages. In one case however, a user's device showed signs of being infected with the malicious Chrome extension, ChromeBack [9], immediately before it contacted a webpage providing download instructions (Figure 4). This may suggest that cracked software downloads are not the only cause of users' browsers being redirected to these download instruction pages (Figure 5).

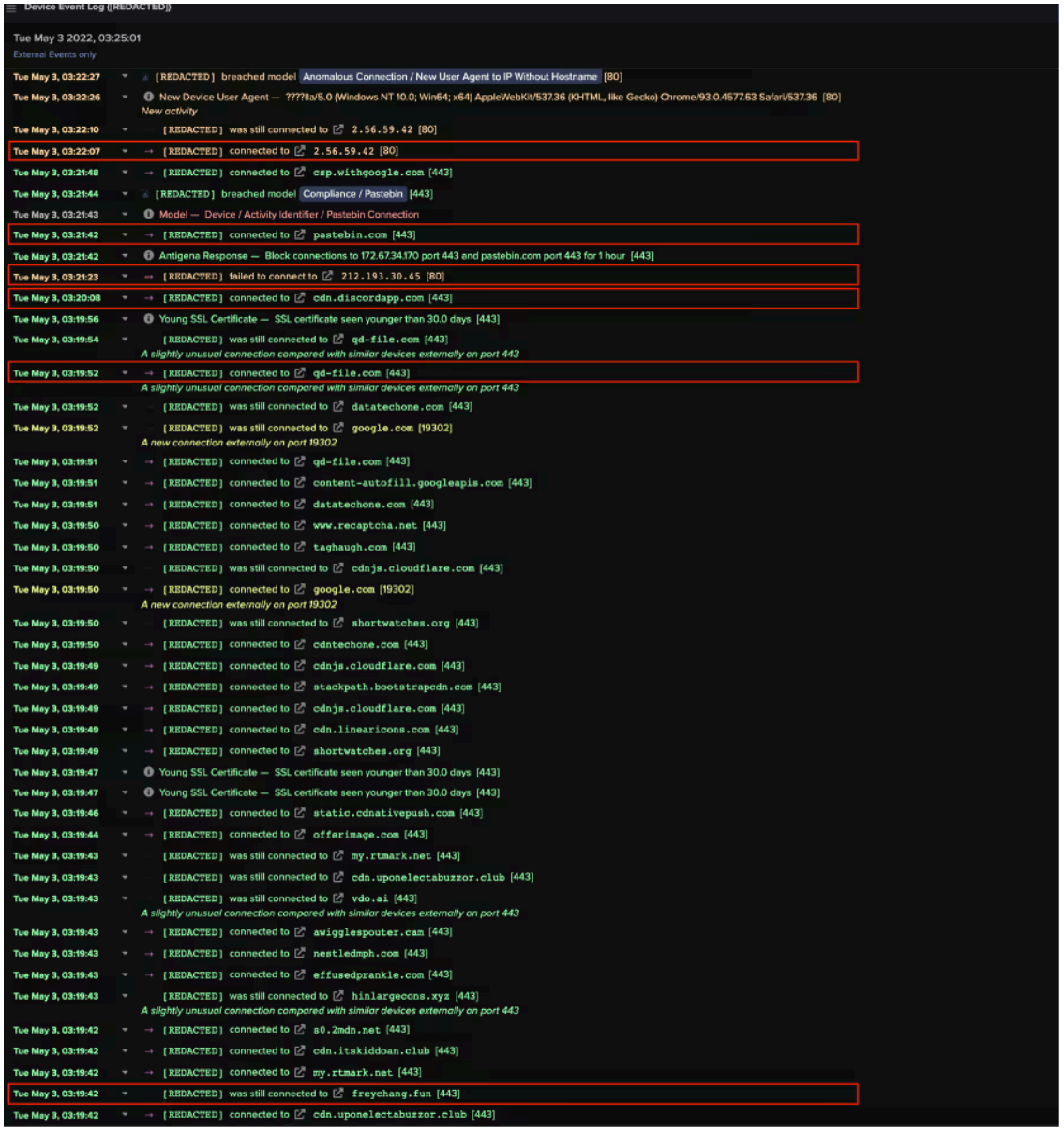


Figure 4: The event log for this device (taken from the Darktrace Threat Visualiser interface) shows that the device contacted endpoints associated with ChromeBack ('freychang[.]fun') prior to visiting a page ('qd-file[.]com') which instructed the device's user to download an archive file from Discord CDN

Figure 5: An image of the website 'crackright[.]com'- a provider of cracked software. Systems which attempted to download software from this website were subsequently led to pages providing instructions to download a password-protected archive from Discord CDN

After users' devices were redirected to pages instructing them to download a password-protected archive, they subsequently contacted cdn.discordapp[.]com over SSL. The archive files which users downloaded over these SSL connections likely contained the PrivateLoader loader module. Immediately after contacting the file storage endpoint, users' devices were observed either contacting Pastebin over SSL, making an HTTP GET request with the URI string '/server.txt' or 'server_p.txt' to 45.144.225[.]57, or making an HTTP GET request with the URI string '/proxies.txt' to 212.193.30[.]45 (Figure 6).

Distinctive user-agent strings such as those containing question marks (e.g. '????ll') and strings referencing outdated Chrome browser versions were consistently seen in these HTTP requests. The following chrome agent was repeatedly observed: 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/74.0.3729.169 Safari/537.36'.

In some cases, devices also displayed signs of infection with other strains of malware such as the RedLine infostealer and the BeamWinHTTP malware downloader. This may suggest that the password-protected archives embedded several payloads.

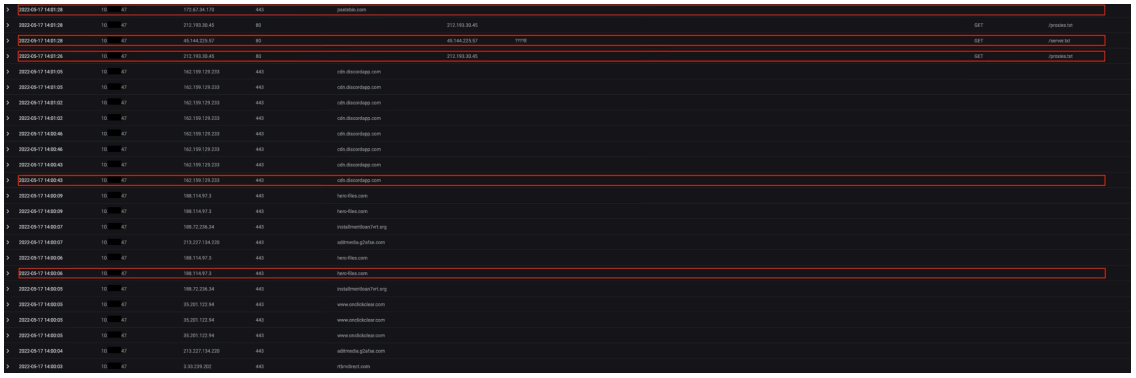


Figure 6: This figure, obtained from Darktrace's Advanced Search interface, represents the post-infection behaviour displayed by a PrivateLoader bot. After visiting hero-files[.]com and downloading the PrivateLoader loader module from Discord CDN, the device can be seen making HTTP GET requests for '/proxies.txt' and '/server.txt' and contacting pastebin[.]com

It seems that PrivateLoader bots contact Pastebin, 45.144.225[.]57, and 212.193.30[.]45 in order to retrieve the IP address of PrivateLoader's main C2 server - the server which provides PrivateLoader bots with payload URLs. This technique used by the operators of PrivateLoader closely mirrors the well-known espionage tactic known as 'dead drop'.

The **dead drop** is a method of espionage tradecraft in which an individual leaves a physical object such as papers, cash, or weapons in an agreed hiding spot so that the intended recipient can retrieve the object later on without having to come in to contact with the source. When threat actors host information about core C2 infrastructure on intermediary endpoints, the hosted information is analogously called a 'Dead Drop Resolver' or 'DDR'. Example URLs of DDRs used by PrivateLoader:

- https://pastebin[.]com/...
- http://212.193.30[.]45/proxies.txt
- http://45.144.225[.]57/server.txt
- http://45.144.255[.]57/server_p.txt

The 'proxies.txt' DDR hosted on 212.193.40[.]45 contains a list of 132 IP address / port pairs. The 119th line of this list includes a scrambled version of the IP address of PrivateLoader's main C2 server (Figures 7 & 8). Prior to June, it seems that the main C2 IP address was '212.193.30[.]21', however, the IP address appears to have recently changed to '85.202.169[.]116'. In a limited set of cases, Darktrace also observed PrivateLoader bots retrieving payload URLs from 2.56.56[.]126 and 2.56.59[.]42 (rather than from 212.193.30[.]21 or 85.202.169[.]116). These IP addresses may be hardcoded secondary C2 address which PrivateLoader bots use in cases where they are unable to retrieve the primary C2 address from Pastebin, 212.193.30[.]45 or 45.144.255[.]57 [10].

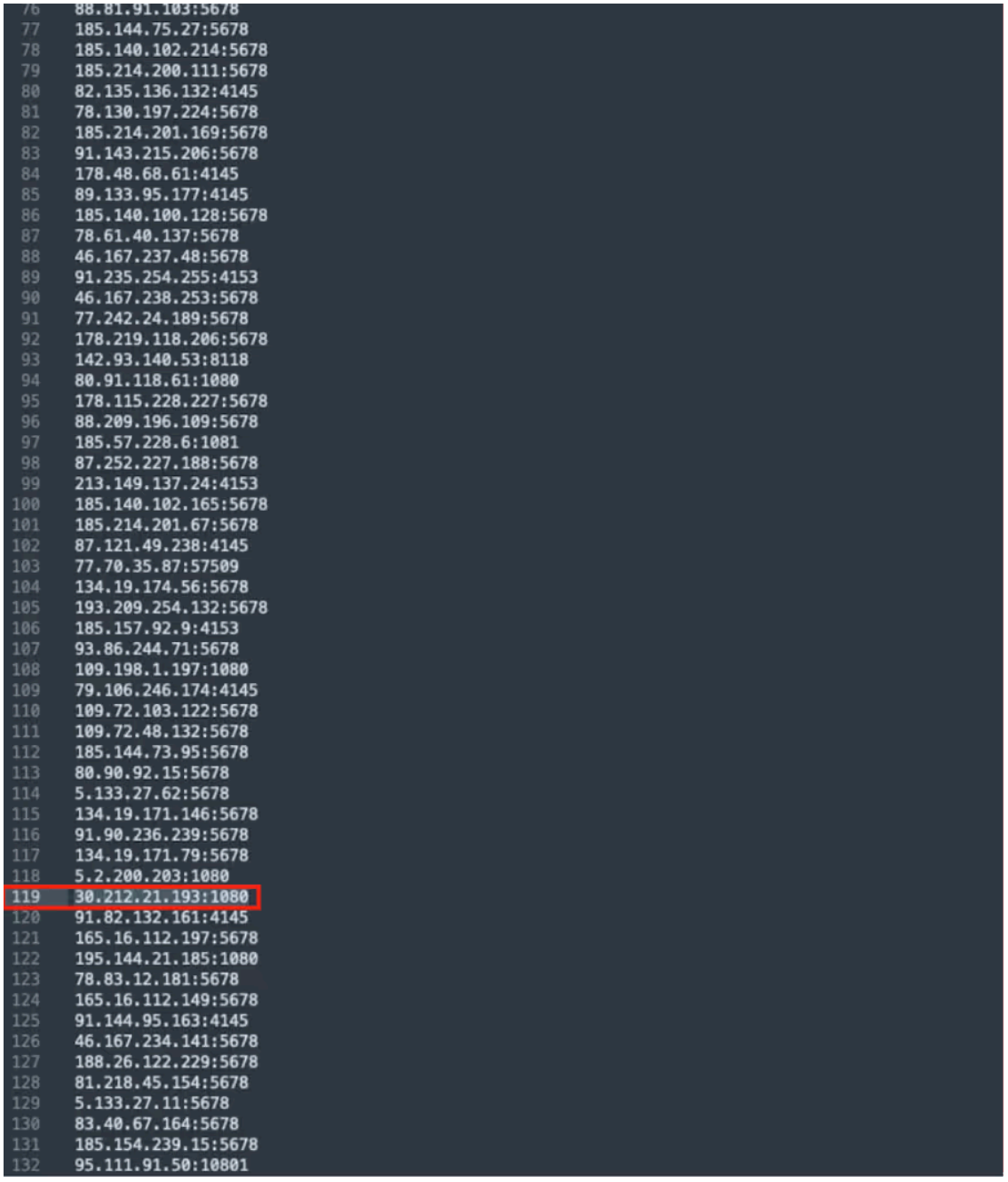


Figure 7: Before June, the 119th entry of the ‘proxies.txt’ file lists '30.212.21.193' - a scrambling of the ‘212.193.30[.]21’ main C2 IP address

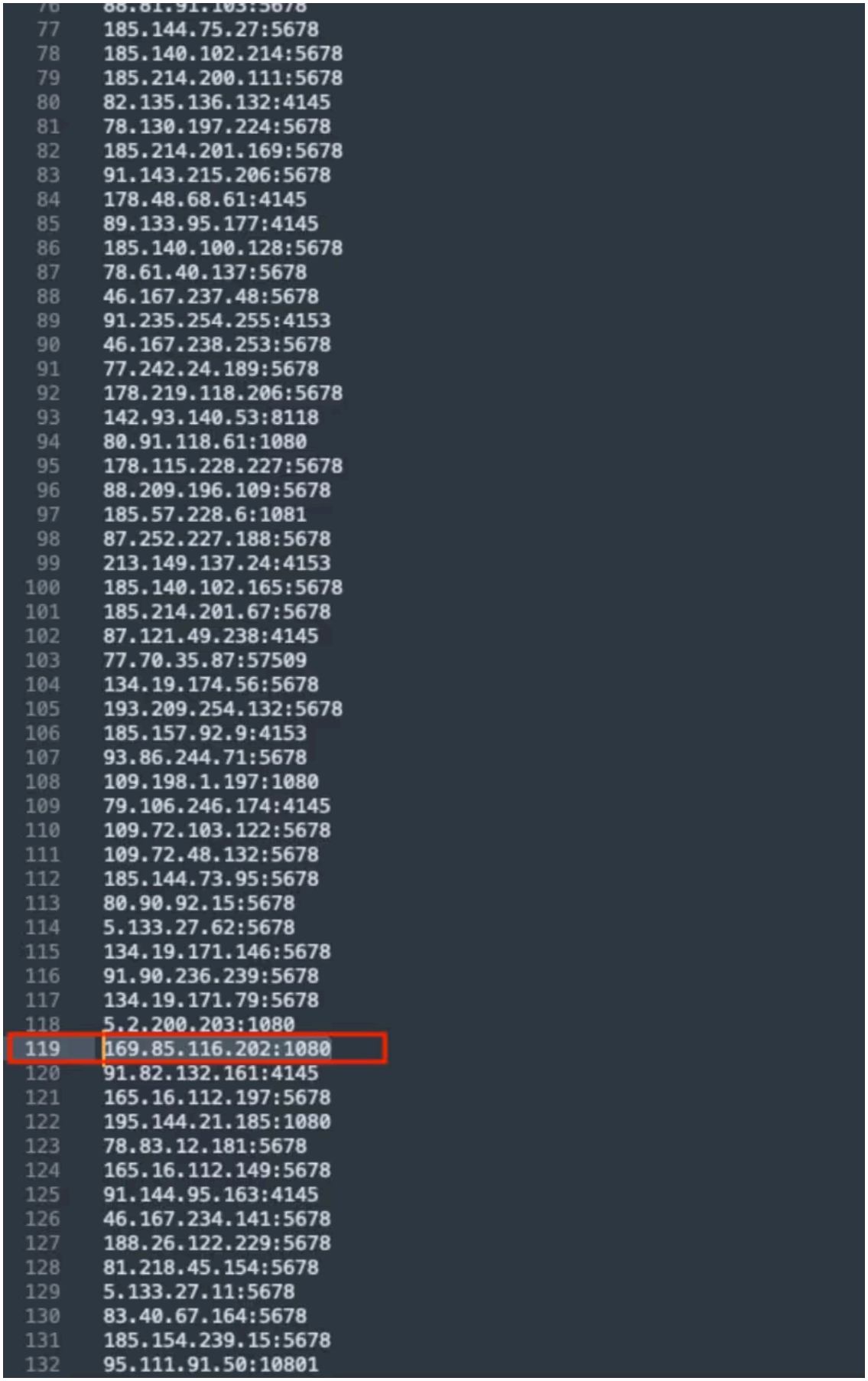

77 185.144.75.27:5678
78 185.140.102.214:5678
79 185.214.200.111:5678
80 82.135.136.132:4145
81 78.130.197.224:5678
82 185.214.201.169:5678
83 91.143.215.206:5678
84 178.48.68.61:4145
85 89.133.95.177:4145
86 185.140.100.128:5678
87 78.61.40.137:5678
88 46.167.237.48:5678
89 91.235.254.255:4153
90 46.167.238.253:5678
91 77.242.24.189:5678
92 178.219.118.206:5678
93 142.93.140.53:8118
94 80.91.118.61:1080
95 178.115.228.227:5678
96 88.209.196.109:5678
97 185.57.228.6:1081
98 87.252.227.188:5678
99 213.149.137.24:4153
100 185.140.102.165:5678
101 185.214.201.67:5678
102 87.121.49.238:4145
103 77.70.35.87:57509
104 134.19.174.56:5678
105 193.209.254.132:5678
106 185.157.92.9:4153
107 93.86.244.71:5678
108 109.198.1.197:1080
109 79.106.246.174:4145
110 109.72.103.122:5678
111 109.72.48.132:5678
112 185.144.73.95:5678
113 80.90.92.15:5678
114 5.133.27.62:5678
115 134.19.171.146:5678
116 91.90.236.239:5678
117 134.19.171.79:5678
118 5.2.200.203:1080
119 169.85.116.202:1080
120 91.82.132.161:4145
121 165.16.112.197:5678
122 195.144.21.185:1080
123 78.83.12.181:5678
124 165.16.112.149:5678
125 91.144.95.163:4145
126 46.167.234.141:5678
127 188.26.122.229:5678
128 81.218.45.154:5678
129 5.133.27.11:5678
130 83.40.67.164:5678
131 185.154.239.15:5678
132 95.111.91.50:10801

Figure 8: Since June, the 119th entry of the ‘proxies.txt’ file lists '169.85.116.202' - a scrambling of the '85.202.169[.]116' main C2 IP address

Once PrivateLoader bots had retrieved C2 information from either Pastebin, 45.144.225[.]57, or 212.193.30[.]45, they went on to make HTTP GET requests for ‘/base/api/statistics.php’ to either 212.193.30[.]21, 85.202.169[.]116, 2.56.56[.]126, or 2.56.59[.]42 (Figure 9). The server responded to these requests with an XOR encrypted string. The strings were encrypted using a 1-byte key [11], such as 0001101 (Figure 10). Decrypting the string revealed a URL for a BMP file hosted on Discord CDN, such as ‘hxxps://cdn.discordapp[.]com/attachments/978284851323088960/986671030670078012/PL_Client.bmp’. These encrypted URLs appear to be file download paths for the PrivateLoader core module.

Figure 9: HTTP response from server to an HTTP GET request for '/base/api/statistics.php'

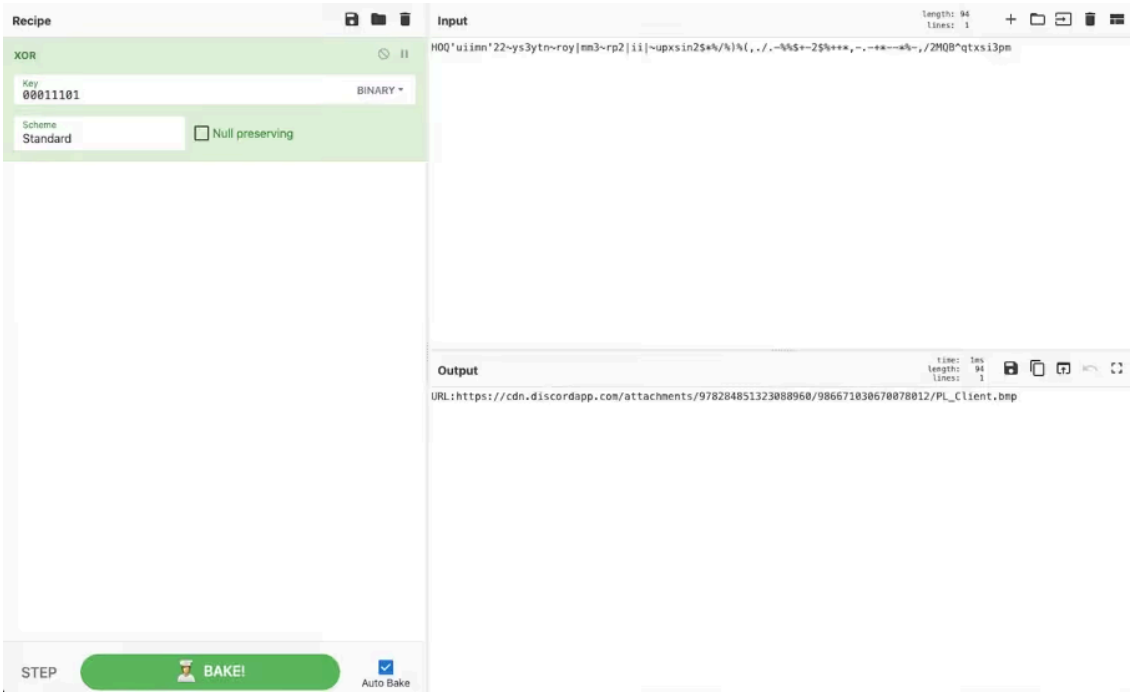


Figure 10: XOR decrypting the string with the one-byte key, 00011101, outputs a URL in CyberChef

After PrivateLoader bots retrieved the 'cdn.discordapp[.]com’ URL from 212.193.30[.]21, 85.202.169[.]116, 2.56.56[.]126, or 2.56.59[.]42, they immediately contacted Discord CDN via SSL connections in order to obtain the PrivateLoader core module. Execution of this module resulted in the bots making HTTP POST requests (with the URI string ‘/base/api/getData.php’) to the main C2 address (Figures 11 & 12). Both the data which the PrivateLoader bots sent over these HTTP POST requests and the data returned via the C2 server’s HTTP responses were heavily encrypted using a combination of password-based key derivation, base64 encoding, AES encryption, and HMAC validation [12].

Figure 11: The above image, taken from Darktrace's Advanced Search interface, shows a PrivateLoader bot carrying out the following steps: contact ‘hero-files[.]com’ --> contact ‘cdn.discordapp[.]com’ --> retrieve ‘/proxies.txt’ from 212.193.30[.]45 --> retrieve ‘/base/api/statistics.php’ from 212.193.30[.]21 --> contact ‘cdn.discordapp[.]com --> make HTTP POST request with the URI ‘base/api/getData.php’ to 212.193.30[.]21

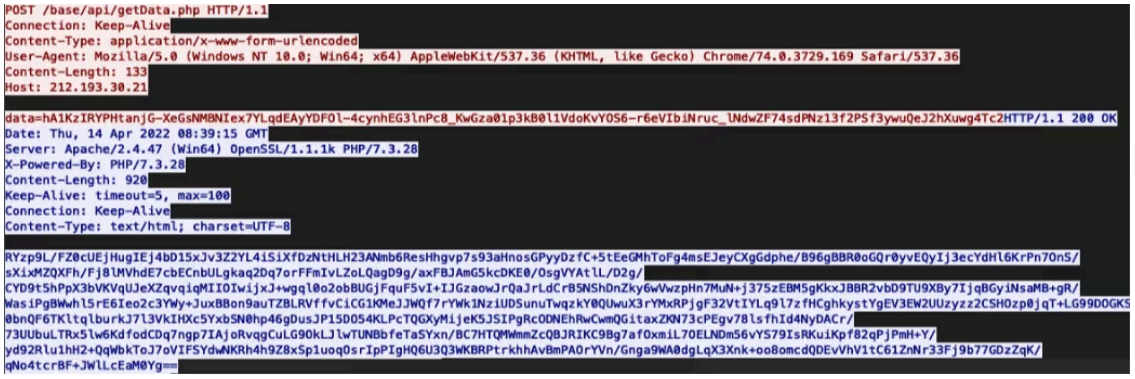
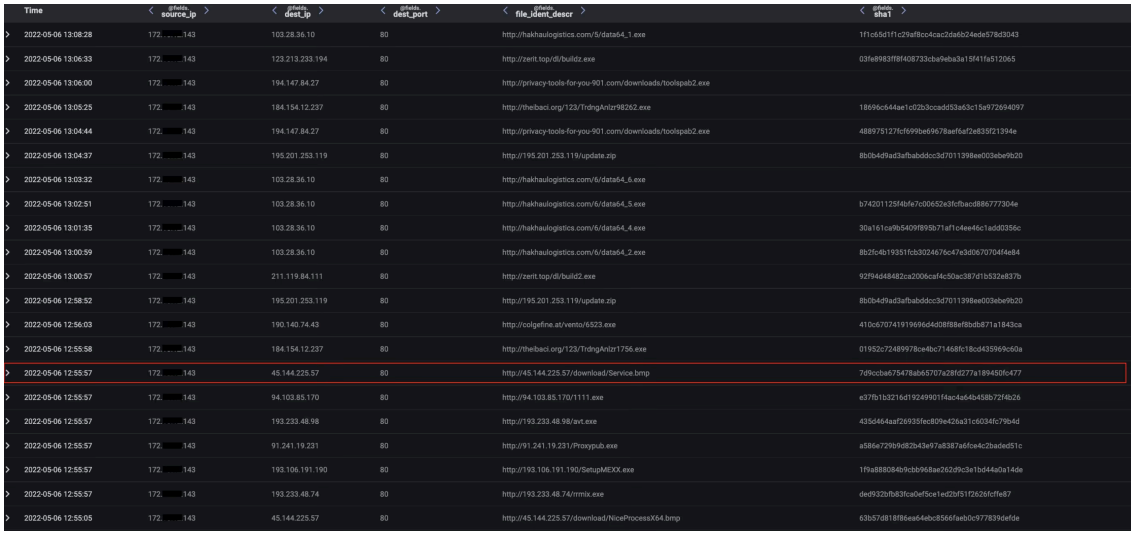


Figure 12: A PCAP of the data sent via the HTTP POST (in red), and the data returned by the C2 endpoint (in blue)

These ‘/base/api/getData.php’ POST requests contain a command, a campaign name and a JSON object. The response may either contain a simple status message (such as “success”) or a JSON object containing URLs of payloads. After making these HTTP connections, PrivateLoader bots were observed downloading and executing large volumes of payloads (Figure 13), ranging from crypto-miners to infostealers (such as Mars stealer), and even to other malware downloaders (such as SmokeLoader). In some cases, bots were also seen downloading files with ‘.bmp’ extensions, such as ‘Service.bmp’, ‘Cube_WW14.bmp’, and ‘NiceProcessX64.bmp’, from 45.144.225[.]57 - the same DDR endpoint from which PrivateLoader bots retrieved main C2 information. These ‘.bmp’ payloads are likely related to the PrivateLoader service module [13]. Certain bots made follow-up HTTP POST requests (with the URI string ‘/service/communication.php’) to either 212.193.30[.]21 or 85.202.169[.]116, indicating the presence of the PrivateLoader service module, which has the purpose of establishing persistence on the device (Figure 14).



Time	source_ip	dest_ip	dest_port	file_ident_descr	sha1
2022-05-06 13:08:28	172.[redacted].143	103.28.36.10	80	http://hakhaulogistics.com/5/data64_1.exe	1f1c65d1f1c29af8cc4cac2da6b24ede578d3043
2022-05-06 13:06:33	172.[redacted].143	123.213.233.194	80	http://zerit.top/dl/buildz.exe	03fe8983ff8f408733cba9eba3a15f41fa512065
2022-05-06 13:06:00	172.[redacted].143	194.147.84.27	80	http://privacy-tools-for-you-901.com/downloads/toolspab2.exe	
2022-05-06 13:05:25	172.[redacted].143	184.154.12.237	80	http://theibaci.org/123/TrdngAnlzr98262.exe	18696c644ae1c02b3ccadd53a63c15a972694097
2022-05-06 13:04:44	172.[redacted].143	194.147.84.27	80	http://privacy-tools-for-you-901.com/downloads/toolspab2.exe	488975127fcf699be69678aef6af2e835f21394e
2022-05-06 13:04:37	172.[redacted].143	195.201.253.119	80	http://195.201.253.119/update.zip	8b0b4d9ad3afbabddcc3d7011398ee003ebe9b20
2022-05-06 13:03:32	172.[redacted].143	103.28.36.10	80	http://hakhaulogistics.com/6/data64_6.exe	
2022-05-06 13:02:51	172.[redacted].143	103.28.36.10	80	http://hakhaulogistics.com/6/data64_5.exe	b7420112f5f4bfe7c00652e3fcfbacd886777304e
2022-05-06 13:01:35	172.[redacted].143	103.28.36.10	80	http://hakhaulogistics.com/6/data64_4.exe	30a161ca9b5409f895b71af1c4ee46c1add0356c
2022-05-06 13:00:59	172.[redacted].143	103.28.36.10	80	http://hakhaulogistics.com/6/data64_2.exe	8b2fc4b19351fcb3024676c47e3d0670704f4e84
2022-05-06 13:00:57	172.[redacted].143	211.119.84.111	80	http://zerit.top/dl/build2.exe	92f94d48482ca2006caf4c50ac387d1b532e837b
2022-05-06 12:58:52	172.[redacted].143	195.201.253.119	80	http://195.201.253.119/update.zip	8b0b4d9ad3afbabddcc3d7011398ee003ebe9b20
2022-05-06 12:56:03	172.[redacted].143	190.140.74.43	80	http://colgefine.at/vento/6523.exe	410c670741919696d4d08f88ef8bd8871a1843ca
2022-05-06 12:55:58	172.[redacted].143	184.154.12.237	80	http://theibaci.org/123/TrdngAnlzr1756.exe	01952c72489978ce4bc71468fc18cd435969c60a
2022-05-06 12:55:57	172.[redacted].143	45.144.225.57	80	http://45.144.225.57/download/Service.bmp	7d9ccba675478ab65707a28fd2774189450fc477
2022-05-06 12:55:57	172.[redacted].143	94.103.85.170	80	http://94.103.85.170/1111.exe	e37fb1b3216d19249901f4ac4a64b458b72f4b26
2022-05-06 12:55:57	172.[redacted].143	193.233.48.98	80	http://193.233.48.98/avt.exe	435d464aaf26935fec809e426a31c6034fc79b4d
2022-05-06 12:55:57	172.[redacted].143	91.241.19.231	80	http://91.241.19.231/Proxypub.exe	a586e729b9d82b43e97a8387a6fce4c2ba6ed51c
2022-05-06 12:55:57	172.[redacted].143	193.106.191.190	80	http://193.106.191.190/SetupMEXX.exe	1f9a888084b9cbb968ae262d9c3e1bd44a0a14de
2022-05-06 12:55:57	172.[redacted].143	193.233.48.74	80	http://193.233.48.74/rrmix.exe	ded932bfb83fca0ef5ce1ed2bf51f2626fcffe87
2022-05-06 12:55:05	172.[redacted].143	45.144.225.57	80	http://45.144.225.57/download/NiceProcessX64.bmp	63b57d818f86ea64ebc8566faeb0c977839defde

Figure 13: The above image, taken from Darktrace's Advanced Search interface, outlines the plethora of malware payloads downloaded by a PrivateLoader bot after it made an HTTP POST request to the '/base/api/getData.php' endpoint. The PrivateLoader service module is highlighted in red

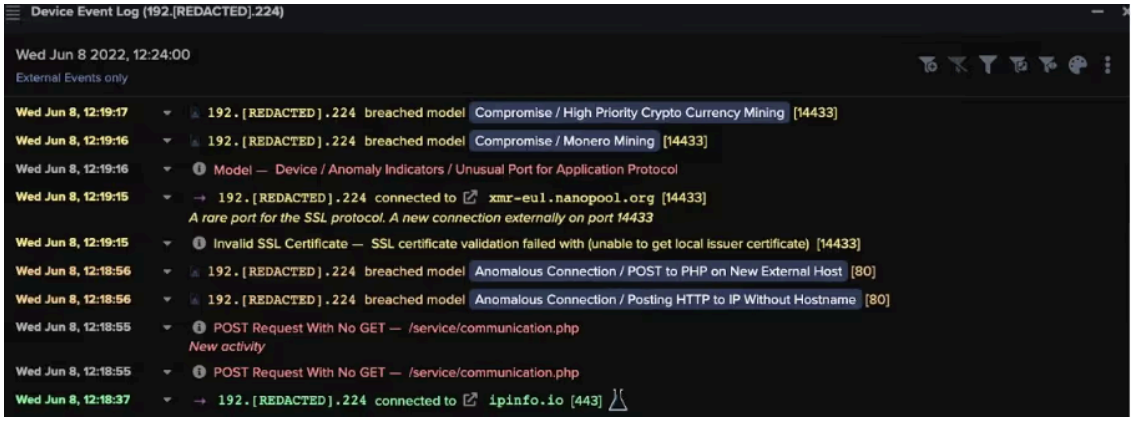


Figure 14: The event log for a PrivateLoader bot, obtained from the Threat Visualiser interface, shows a device making HTTP POST requests to '/service/communication.php' and connecting to the NanoPool mining pool, indicating successful execution of downloaded payloads

In several observed cases, PrivateLoader bots downloaded another malware downloader called 'SmokeLoader' (payloads named 'toolspab2.exe' and 'toolspab3.exe') from "Privacy Tools" endpoints [14], such as 'privacy-tools-for-you-802[.]com' and 'privacy-tools-for-you-783[.]com'. These "Privacy Tools" domains are likely impersonation attempts of the legitimate 'privacytools[.]io' website - a website run by volunteers who advocate for data privacy [15].

After downloading and executing malicious payloads, PrivateLoader bots were typically seen contacting crypto-mining pools, such as NanoPool, and making HTTP POST requests to external hosts associated with SmokeLoader, such as hosts named 'host-data-coin-11[.]com' and 'file-coin-host-12[.]com' [16]. In one case, a PrivateLoader bot went on to exfiltrate data over HTTP to an external host named 'cheapf[.]link', which was registered on the 14th March 2022 [17]. The name of the file which the PrivateLoader bot used to exfiltrate data

was ‘NOP8QIMGV3W47Y.zip’, indicating information stealing activities by Mars Stealer (Figure 15) [18]. By saving the HTTP stream as raw data and utilizing a hex editor to remove the HTTP header portions, the hex data of the ZIP file was obtained. Saving the hex data using a ‘.zip’ extension and extracting the contents, a file directory consisting of system information and Chrome and Edge browsers’ Autofill data in cleartext .txt file format could be seen (Figure 16).

Figure 15: A PCAP of a PrivateLoader bot’s HTTP POST request to cheapf[.]link, with data sent by the bot appearing to include Chrome and Edge autofill data, as well as system information

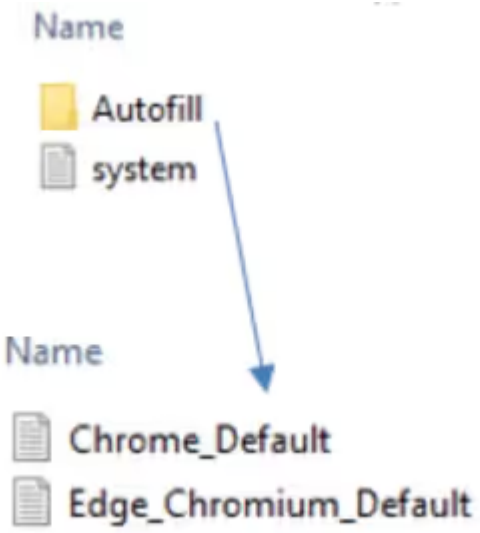


Figure 16: File directory structure and files of the ZIP archive

When left unattended, PrivateLoader bots continued to contact C2 infrastructure in order to relay details of executed payloads and to retrieve URLs of further payloads.

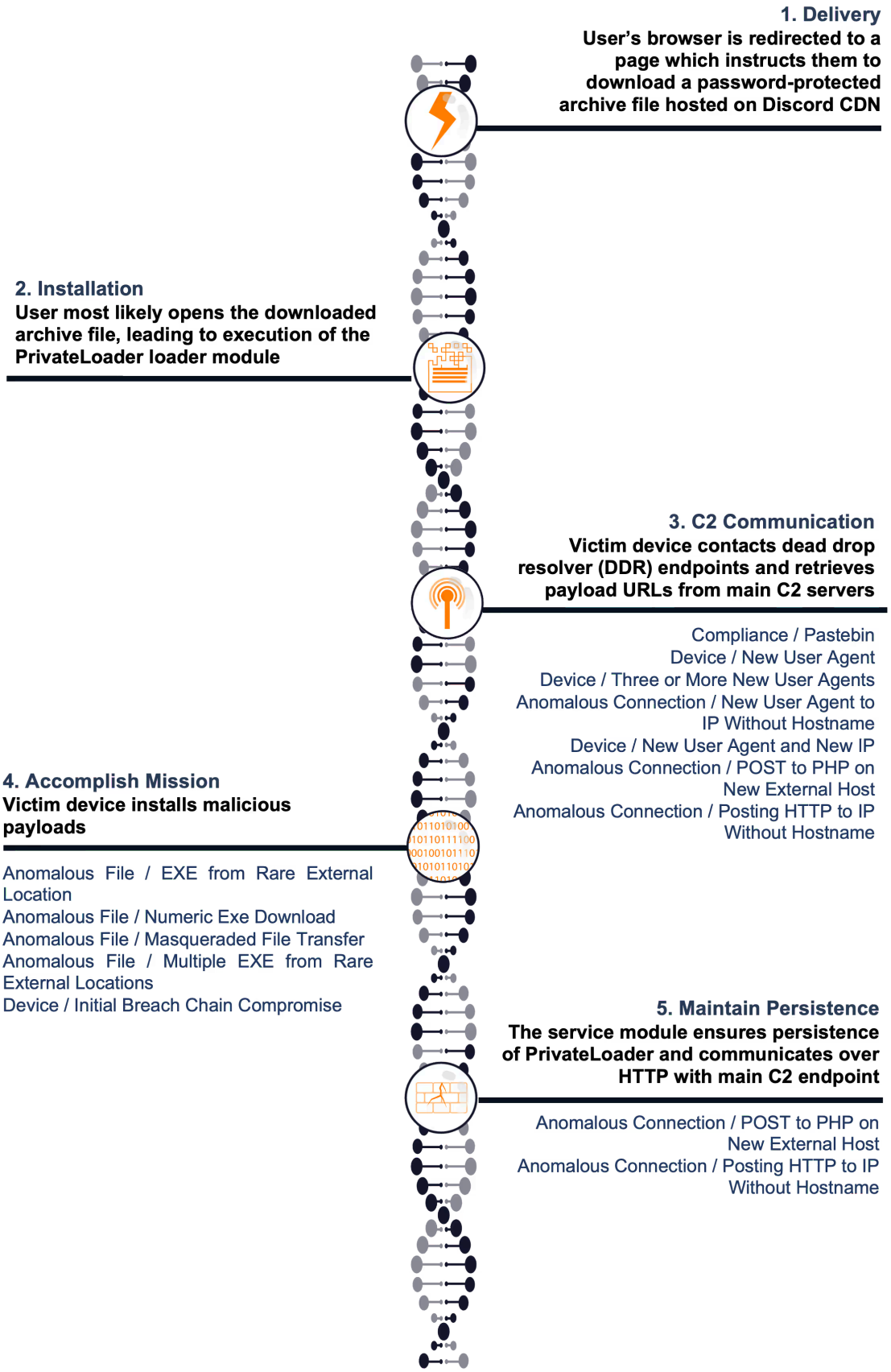


Figure 17: Timeline of the attack

Darktrace Coverage

Most of the incidents surveyed for this article belonged to prospective customers who were trialling Darktrace with RESPOND in passive mode, and thus without the ability for autonomous intervention. However in all observed cases, Darktrace DETECT was able to provide visibility into the actions taken by PrivateLoader bots. In one case, despite the infected bot being disconnected from the client's network, Darktrace was still able to provide visibility into the device's network behaviour due to the client's usage of Darktrace/Endpoint.

If a system within an organization's network becomes infected with PrivateLoader, it will display a range of anomalous network behaviours before it downloads and executes malicious payloads. For example, it will contact Pastebin or make HTTP requests with new and unusual user-agent strings to rare external endpoints. These network behaviours will generate some of the following alerts on the Darktrace UI:

- Compliance / Pastebin
- Device / New User Agent and New IP
- Device / New User Agent
- Device / Three or More New User Agents
- Anomalous Connection / New User Agent to IP Without Hostname
- Anomalous Connection / POST to PHP on New External Host
- Anomalous Connection / Posting HTTP to IP Without Hostname

Once the infected host obtains URLs for malware payloads from a C2 endpoint, it will likely start to download and execute large volumes of malicious files. These file downloads will usually cause Darktrace to generate some of the following alerts:

- Anomalous File / EXE from Rare External Location
- Anomalous File / Numeric Exe Download
- Anomalous File / Masqueraded File Transfer
- Anomalous File / Multiple EXE from Rare External Locations
- Device / Initial Breach Chain Compromise

If RESPOND is deployed in active mode, Darktrace will be able to autonomously block the download of additional malware payloads onto the target machine and the subsequent beaconing or crypto-mining activities through network inhibitors such as 'Block matching connections', 'Enforce pattern of life' and 'Block all outgoing traffic'. The 'Enforce pattern of life' action results in a device only being able to make connections and data transfers which Darktrace considers normal for that device. The 'Block all outgoing traffic' action will cause all traffic originating from the device to be blocked. If the customer has Darktrace's Proactive Threat Notification (PTN) service, then a breach of an Enhanced Monitoring model such as 'Device / Initial Breach Chain Compromise' will result in a Darktrace SOC analyst proactively notifying the customer of the suspicious activity. Below is a list of Darktrace RESPOND (Antigena) models which would be expected to breach due to PrivateLoader activity. Such models can seriously hamper attempts made by PrivateLoader bots to download malicious payloads.

- Antigena / Network / External Threat / Antigena Suspicious File Block
- Antigena / Network / Significant Anomaly / Antigena Controlled and Model Breach
- Antigena / Network / External Threat / Antigena File then New Outbound Block

- Antigena / Network / Significant Anomaly / Antigena Significant Anomaly from Client Block
- Antigena / Network / Significant Anomaly / Antigena Breaches Over Time Block

In one observed case, the infected bot began to download malicious payloads within one minute of becoming infected with PrivateLoader. Since RESPOND was correctly configured, it was able to immediately intervene by autonomously enforcing the device's pattern of life for 2 hours and blocking all of the device's outgoing traffic for 10 minutes (Figure 17). When malware moves at such a fast pace, the availability of autonomous response technology, which can respond immediately to detected threats, is key for the prevention of further damage.

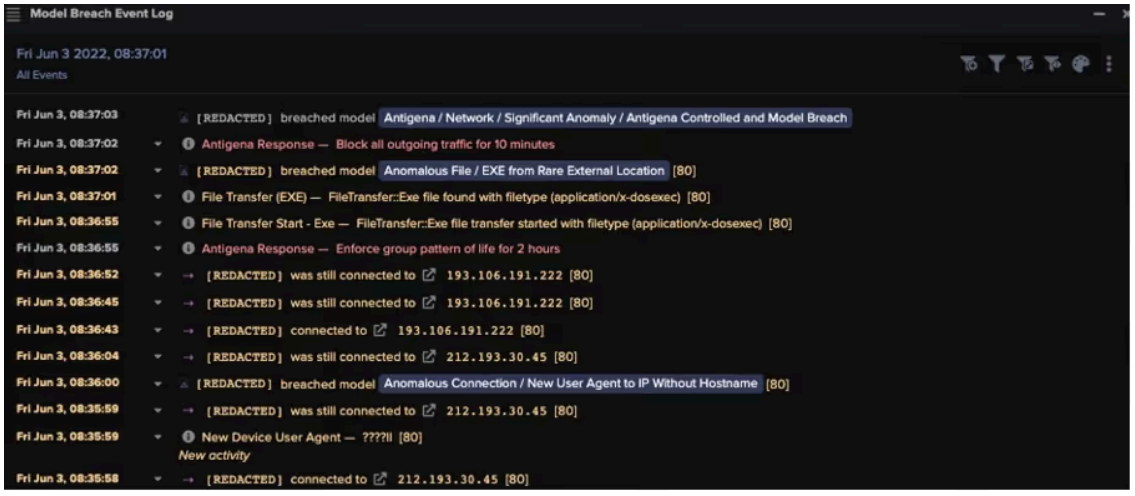


Figure 18: The event log for a Darktrace RESPOND (Antigena) model breach shows Darktrace RESPOND performing inhibitive actions once the PrivateLoader bot begins to download payloads

Conclusion

By investigating PrivateLoader infections over the past couple of months, Darktrace has observed PrivateLoader operators making changes to the downloader's main C2 IP address and to the user-agent strings which the downloader uses in its C2 communications. It is relatively easy for the operators of PrivateLoader to change these superficial network-based features of the malware in order to evade detection [19]. However, once a system becomes infected with PrivateLoader, it will inevitably start to display anomalous patterns of network behaviour characteristic of the Tactics, Techniques and Procedures (TTPs) discussed in this blog.

Throughout 2022, Darktrace observed overlapping patterns of network activity within the environments of several customers, which reveal the archetypal steps of a PrivateLoader infection. Despite the changes made to PrivateLoader's network-based features, Darktrace's Self-Learning AI was able to continually identify infected bots, detecting every stage of an infection without relying on known indicators of compromise. When configured, RESPOND was able to immediately respond to such infections, preventing further advancement in the cyber kill chain and ultimately preventing the delivery of floods of payloads onto infected devices.

IoCs

Type	IOC	Description
Hostname	crackedpc[.]net crackright[.]com cracka2zsoft[.]com skidrowcpygames[.]com	Endpoints providing cracked/pirated software
URL	hxxps://hero-files[.]com/ hxxps://qd-file[.]com/ hxxps://pu-file[.]com/	Web pages with instructions to download PrivateLoader loader module
URL	hxxps://pastebin[.]com hxxp://212.193.30[.]45/proxies.txt hxxp://45.144.225[.]57/server.txt hxxp://45.144.225[.]57/server_p.txt	Dead drop resolver (DDR) endpoints contacted by loader module to obtain main C2 IP address
IP	212.193.30[.]21 85.202.169[.]116 2.56.56[.]126 2.56.59[.]42	Main C2 endpoints for PrivateLoader
URI	/base/api/statistics.php	URI used by PrivateLoader loader module to retrieve the URL of the core module
URI	/base/api/getData.php	URI used in HTTP POST requests from PrivateLoader core module
URI	/service/communication.php	URI used in HTTP POST requests from PrivateLoader service module

User-agent strings	????ll ???wll ????lla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.63 Safari/537.36 ??\xf2\xf7?ll \xac\xdb???lla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.63 Safari/537.36 \xd3^???lla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.63 Safari/537.36 \xb4a???lla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.63 Safari/537.36 ??????????????????????\xc5 ???????? Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/74.0.3729.169 Safari/537.36 Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.63 Safari/537.36	User-agent strings used in PrivateLoader's HTTP requests

URL	hxxps://cdn.discordapp[.]com/attachments/978284851323088960/986671030670078012/PL_Client.bmp	XOR decrypted (with one-byte binary key 0011101) URL for PrivateLoader core module
URL	hxxp://45.144.225[.]57/download/Service.bmp	URL for PrivateLoader service module
URL	hxxp://31.41.244[.]240/rrun.exe hxxp://31.41.244[.]240/rrmx.exe hxxp://45.144.225[.]57/Setup.exe hxxp://45.144.225[.]57/download/NiceProcessX64.bmp hxxp://45.144.225[.]57/download/Service.bmp hxxp://62.204.41[.]23:9080/13.php hxxp://80.92.206[.]135/tools.exe hxxp://91.241.19[.]231/Proxypub.exe hxxp://94.103.85[.]170/1111.exe hxxp://185.163.45[.]239/g_shock_casio_easy hxxp://193.106.191[.]78/SetupMEXX.exe hxxp://193.106.191[.]153/SetupRU.exe hxxp://193.106.191[.]190/SetupMEXX.exe hxxp://193.106.191[.]222/SetupMEXX.exe hxxp://193.233.48[.]90/sloa2.exe hxxp://193.233.48[.]98/avt.exe	URLs for malicious payloads

	hxxp://193.233.48[.]74/rrmix.exe hxxp://194.87.31[.]175/PointerStick_2.exe hxxp://195.201.253[.]119/update.zip hxxp://212.193.30[.]29/WW/file1.exe hxxp://api.popsa-hueta[.]shop/ hxxp://api.popsa-hueta[.]xyz/ hxxp://colgefine[.]at/vento/6523.exe hxxp://data-coin-data-13[.]com/downloads/toolspab2.exe hxxp://file-hoster-cluster-1[.]com/files/kk.exe hxxp://f0681638.xsph[.]ru/Client.exe hxxp://h163012.srv12.test-hf[.]su/34.exe hxxp://h163012.srv12.test-hf[.]su/35.exe hxxp://hakhaulogistics[.]com/5/data64_1.exe hxxp://hakhaulogistics[.]com/5/data64_2.exe hxxp://hakhaulogistics[.]com/5/data64_4.exe hxxp://hakhaulogistics[.]com/5/data64_5.exe hxxp://hakhaulogistics[.]com/5/data64_6.exe hxxp://hakhaulogistics[.]com/6/data64_2.exe	

	hxxp://hakhaulogistics[.]com/6/data64_4.exe hxxp://hakhaulogistics[.]com/6/data64_5.exe hxxp://hakhaulogistics[.]com/6/data64_6.exe hxxp://ilialalagkou[.]com/forum/chrome.exe hxxp://jetij87d.beget[.]tech/Polution_v0.7b_windows_64.exe hxxp://jfklaskqkk[.]shop/sloki/uitowq/combobox3.exe hxxp://jsdkcd[.]link/MSN.exe hxxp://mysql.webtm[.]ru/ali.exe hxxp://mysql.webtm[.]ru/min.exe hxxp://opqwes[.]top/dl/buildp.exe hxxp://paneltraff.webtm[.]ru/user2.exe hxxp://privacy-tools-for-you-783[.]com/downloads/toolspab3.exe hxxp://privacy-tools-for-you-802[.]com/downloads/toolspab2.exe hxxp://privacy-tools-for-you-901[.]com/downloads/toolspab2.exe hxxp://rampl[.]at/index.php hxxp://stylesheet.faseaegasdfase[.]com/hp8/g1/rtst1058.exe hxxp://tengenuzui.s3.pl-waw.scw[.]cloud/pub-provider/poweroff.exe hxxp://theibaci[.]org/123/TrdngAnlzr1756.exe	

	hxxp://theibaci[.]org/123/TrdngAnlzr98262.exe hxxp://www.jhtuangou[.]com/tvstream66.exe hxxp://unknowndsl[.]com/file/UDLD.exe hxxp://usashit[.]com/77_1.exe hxxp://zerit[.]top/dl/buildz.exe hxxp://zerit[.]top/dl/build2.exe	

MITRE ATT&CK Techniques Observed

Tactic	Sub-Technique
Resource Development	T1583.006 Acquire Infrastructure: Web Services T1608.001 Stage Capabilities: Upload Malware T1608.005 Stage Capabilities: Link Target
Execution	T1204.002 User Execution: Malicious File
Defence Evasion	T1027 Obfuscated Files or Information
Command and Control	T1001 Data Obfuscation T1071.001 Application Layer Protocol: Web Protocols T1102.001 Web Service: Dead Drop Resolver T1105 Ingress Tool Transfer T1132 Data Encoding T1568 Dynamic Resolution T1573.001 Encrypted Channel: Symmetric Cryptography

References

[1], [8],[13] https://www.youtube.com/watch?v=Ldp7eESQotM

[2] https://news.sophos.com/en-us/2021/09/01/fake-pirated-software-sites-serve-up-malware-droppers-as-a-service/

[3] https://www.researchgate.net/publication/228873118_Measuring_Pay-per Install_The_Commoditization_of_Malware_Distribution

[4], [15] https://intel471.com/blog/privateloader-malware

[5] https://medium.com/walmartglobaltech/privateloader-to-anubis-loader-55d066a2653e

[6], [10],[11], [12] https://www.zscaler.com/blogs/security-research/peeking-privateloader

[7] https://www.trendmicro.com/en_us/research/22/e/netdooka-framework-distributed-via-privateloader-ppi.html

[9] https://www.gosecure.net/blog/2022/02/10/malicious-chrome-browser-extension-exposed-chromeback-leverages-silent-extension-loading/

[14] https://www.proofpoint.com/us/blog/threat-insight/malware-masquerades-privacy-tool

[16] https://asec.ahnlab.com/en/30513/

[17]https://twitter.com/0xrb/status/1515956690642161669

[18] https://isc.sans.edu/forums/diary/Arkei+Variants+From+Vidar+to+Mars+Stealer/28468

[19] http://detect-respond.blogspot.com/2013/03/the-pyramid-of-pain.html

---