

Mystic Stealer | Zscaler

 zscaler.com/blogs/security-research/mystic-stealer

Key Points

- *Mystic Stealer* is a new information stealer that was first advertised in April 2023
- Mystic steals credentials from nearly 40 web browsers and more than 70 browser extensions
- The malware also targets cryptocurrency wallets, Steam, and Telegram
- The code is heavily obfuscated making use of polymorphic string obfuscation, hash-based import resolution, and runtime calculation of constants
- Mystic implements a custom binary protocol that is encrypted with RC4

How do you know when something is in hot demand in the underground economy? The same way you do in the real world – the market becomes flooded. This is the story of information stealers today. "Stealers" are a kind of malware designed to run on an endpoint post-compromise, while their primary features center on the theft of user data. Oftentimes this is credential data, but it can be any data that may have financial value to an adversary; this includes paid online service accounts, cryptocurrency wallets, instant messenger, or email contacts lists, etc. Stealers also bridge the realms of criminal and nation-state focus. Many espionage-focused threat groups operate stealer families for pilfering information from target networks. Credential information can further increase access or penetration into an environment. Demand for compromised credentials to fuel criminal access to user accounts and target networks has resulted in a steady stream of newly developed information-stealing malware, keeping account markets stocked. With the amount of visibility we have at Zscaler, we are accustomed to encountering new threats on a daily basis. Enter Mystic Stealer, a fresh stealer lurking in the cyber sphere, noted for its data theft capabilities, obfuscation, and an encrypted binary protocol to enable it to stay under the radar and evade defenses. Together with our colleagues at [InQuest](#), we present a deep dive technical analysis of the malware. We also share indicators from an in-depth analysis of the infrastructure footprint of deployed Mystic Stealer controllers and countermeasures for detecting the client in your environment.

Note: the content of this blog is also hosted by [InQuest](#) [here](#).

The Data Heist Specialist

Mystic Stealer focuses on data theft, exhibiting capabilities that allow it to pilfer a wide array of information. For starters, it is designed to collect computer information such as the system hostname, user name, and GUID. It also identifies a likely system user geolocation using the locale and keyboard layout. But it doesn't stop there.

Key Mystic Stealer functions include its ability to extract data from web browsers and cryptocurrency wallets. Like many stealers, it collects auto-fill data, browsing history, arbitrary files, cookies, and information related to cryptocurrency wallets. Whether it's Bitcoin, DashCore, Exodus, or any other popular crypto wallet, Mystic Stealer has it covered. Mystic can also steal Telegram and Steam credentials.

Interestingly, the stealer does not require the integration of third-party libraries for decrypting or decoding target credentials. Some leading stealer projects download DLL files post-install to implement functionality to extract credentials from files on the local system. Instead, Mystic Stealer collects and exfiltrates information from an infected system and then sends the data to the command & control (C2) server that handles parsing. This is a different approach from many leading stealers and is likely an alternate design to keep the size of the stealer binary smaller and the intention less clear to file analyzers.

The Mystic Stealer crimeware is implemented in C for the client and Python for the control panel.

Technical analysis

Looking at the existing releases, it seems clear that the developer of Mystic Stealer is looking to produce a stealer on par with the current trends of the malware space while attempting to focus on anti-analysis and defense evasion.

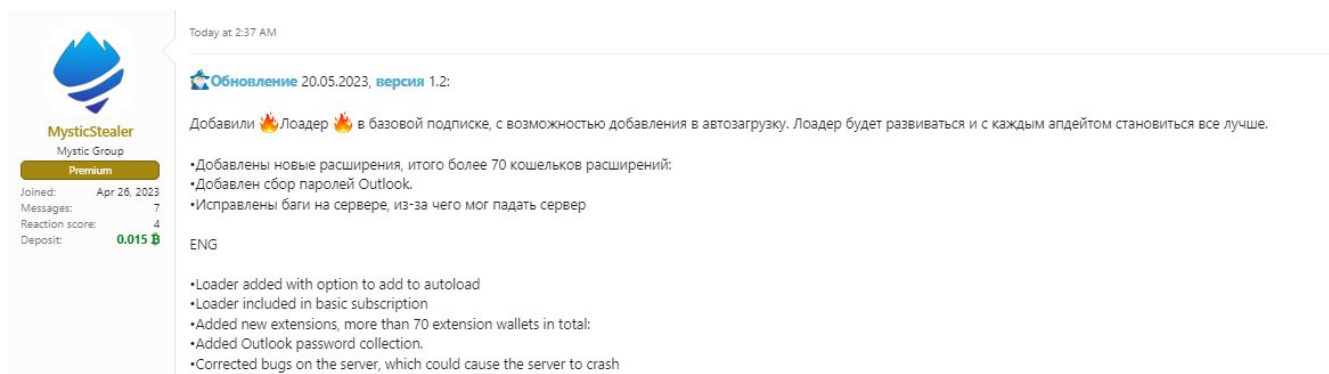
In terms of capabilities, it's a fairly standard set of functionality as seen with many stealers today. The malware collects system information which is packaged together for a check-in to the C2 server:

- Keyboard layout
- Locale
- CPU information
- Number of CPU processors
- Screen dimensions
- Computer name
- Username
- Running processes
- System architecture
- Operating system version

Key data theft functionality includes the ability to capture history and auto-fill data, bookmarks, cookies, and stored credentials from nearly 40 different web browsers. In addition, it collects Steam and Telegram credentials as well as data related to installed cryptocurrency wallets. The malware targets more than 70 web browser extensions for cryptocurrency theft and uses the same functionality to target two-factor authentication (2FA) applications. The approach used by Mystic Stealer is similar to what was [reported](#) for Arkei Stealer. Further details on targeted browsers, cryptocurrency plugins, and 2FA apps are available in the appendix.

Depending on a configuration provided by the C2 server, the malware will capture a screenshot of the desktop, which is exfiltrated to the C2 server.

On May 20, the Mystic Stealer seller posted updates that include loader functionality and a persistence capability to forums as shown in Figure 1. *Loader* refers to the ability to download and execute additional malware payloads. This is reflective of a continuing trend where loaders allow one threat actor to support the distribution of affiliate malware being loaded on compromised devices. This is already a notable risk for many organizations due to the use of malware distribution networks and initial access brokers for the distribution of high-severity payloads like ransomware. It underscores the need to take preventative steps to [ensure a security posture](#) that reduces the risk of malware delivery and footholds early on in attack campaigns.



The image shows a forum post from the 'MysticStealer Premium' group. The post is dated 'Today at 2:37 AM' and is titled 'Обновление 20.05.2023, версия 1.2:'. The main text of the post is in Russian and states: 'Добавили 🔥 Лоадер 🔥 в базовой подписке, с возможностью добавления в автозагрузку. Лоадер будет развиваться и с каждым апдейтом становиться все лучше.' Below this, there is a bulleted list of updates: '•Добавлены новые расширения, итого более 70 кошелеков расширений;', '•Добавлен сбор паролей Outlook.', and '•Исправлены баги на сервере, из-за чего мог падать сервер'. The post also includes the text 'ENG' and a list of features in English: '•Loader added with option to add to autoloader', '•Loader included in basic subscription', '•Added new extensions, more than 70 extension wallets in total:', '•Added Outlook password collection.', and '•Corrected bugs on the server, which could cause the server to crash'. On the left side of the forum post, there is a profile card for 'MysticStealer' with a 'Premium' badge, showing 'Joined: Apr 26, 2023', 'Messages: 7', 'Reaction score: 4', and 'Deposit: 0.015 B'.

Figure 1. MysticStealer forum post advertising v1.2 update with loader support

As previously noted, there are several anti-analysis and evasion features additionally present in Mystic Stealer:

Binary expiration. The trojan will terminate execution if the running build is older than a specified date. This is likely an execution guardrail that attempts to prevent anti-malware researchers and sandboxes that analyze the sample much later than when it was intended to be distributed or executed on victim machines. Figure 2 shows a Mystic Stealer sample that retrieves the current system time and compares the value to 1685318914 (0x6473ED02), which when converted from an epoch to a timestamp translates to Sun May 28 17:08:34 2023.

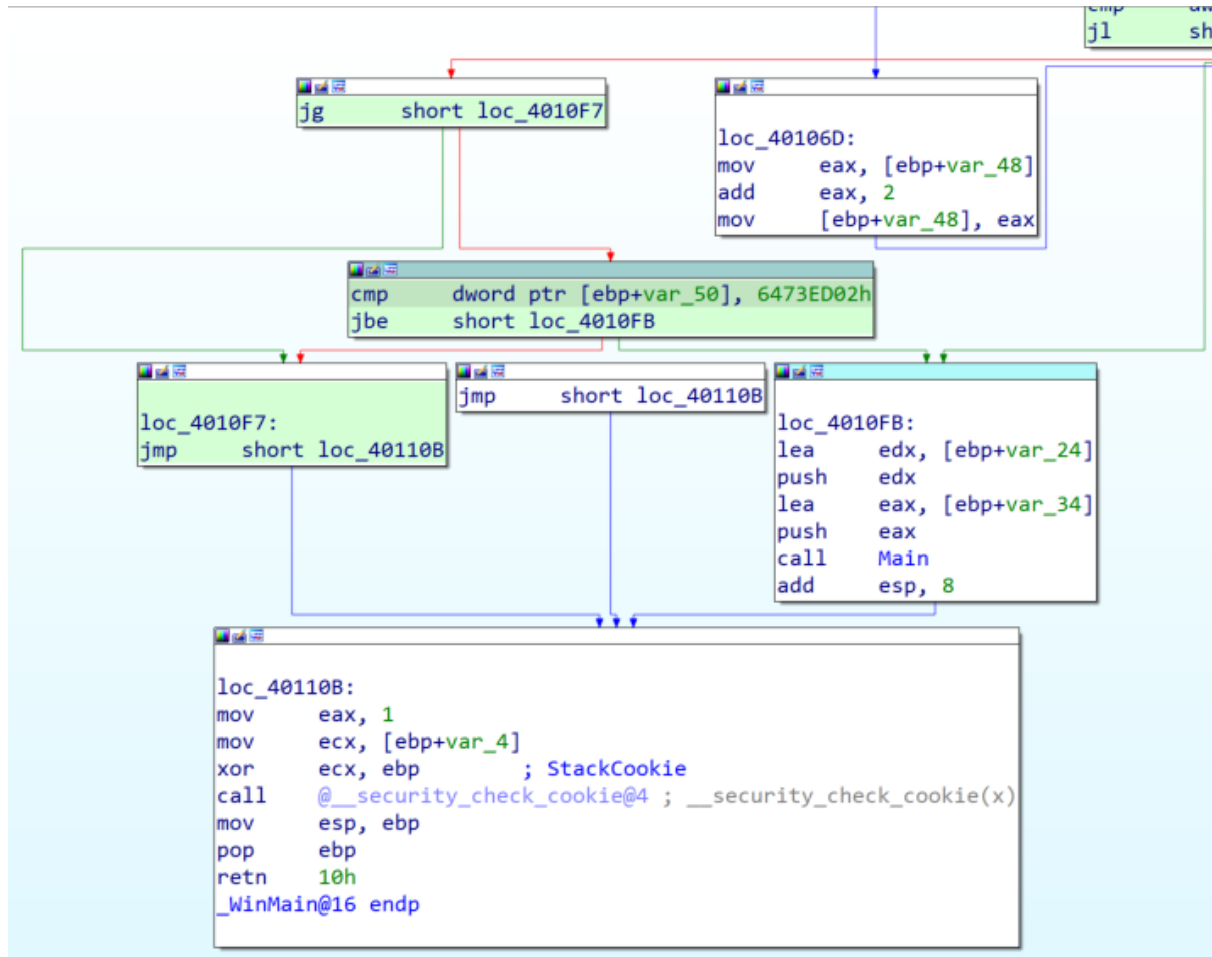


Figure 2. Example Mystic Stealer date expiration feature

Anti-virtualization. Mystic Stealer is configurable and some samples contain anti-VM features, detecting hypervisor runtime environments, and avoiding execution. This is helpful for avoiding execution in sandbox environments but it isn't always effective.

Mystic uses the CPUID assembly instruction to detect virtual environments by inspecting the result for specific values that are indicative of virtualization software. In particular, the code checks for the manufacturer ID string (with a length of 12 bytes) for the following values:

- “XenVMMXenVMM” (Xen HVM)
- “VMwareVMware” (VMware)
- “Microsoft Hv” (Microsoft Hyper-V)
- “KVMKVMKVM “ (KVM)
- “prl hyperv “ (Parallels)
- “VBoxVBoxVBox” (VirtualBox)

This detection code is likely derived from [Pafish](#).

Windows APIs imported by hash. The stealer resolves and dynamically loads Windows APIs using a custom XOR based [hashing algorithm](#) represented in the Python snippet shown below:

```
i = 0
for val in function_name:
    i = ord(val) ^ ((0x240CE91 * i) & 0xffffffff)
print(hex(i))
```

Note that the constant value (e.g., 0x240CE91) changes between Mystic samples. The malware walks the export tables for the following Windows DLLs and hashes each export name until a match is found:

- Kernel32.dll
- Advapi32.dll
- Kernel32.dll
- Gdiplus.dll
- Crypt32.dll
- User32.dll
- Ws2_32.dll
- Ole32.dll
- Gdi32.dll
- Ntdll.dll

Dynamic constant calculation. Constant values in the code are obfuscated and dynamically calculated at runtime. For example, the API hashing algorithm shown above uses the constant 0x240CE91. However, this constant does not directly exist in the code. Instead, the value 0x240CEA6 is present and the code performs an XOR operation with the value 0x37 to produce the actual constant 0x240CE91 as shown in Figure 3.

```
.text:0041ACF7 ; -----
.text:0041ACF7
.text:0041ACF7 loc_41ACF7: ; CODE XREF: HashImportFunctionName+C2↓j
.text:0041ACF7 5F pop edi ; HashImportFunctionName+D3↓j
.text:0041ACF8
.text:0041ACF8 loc_41ACF8: ; CODE XREF: HashImportFunctionName+71↑j
.text:0041ACF8 0F BE 06 movsx eax, byte ptr [esi]
.text:0041ACFB 33 D2 xor edx, edx
.text:0041ACFD 3B 45 FC cmp eax, [ebp+var_4]
.text:0041AD00 74 2B jz short loc_41AD2D
.text:0041AD02
.text:0041AD02 loc_41AD02: ; CODE XREF: HashImportFunctionName+A7↓j
.text:0041AD02 C7 45 DC 7E F7 00 00 mov [ebp+var_24], 0F77Eh
.text:0041AD09 81 75 DC B9 00 00 00 xor [ebp+var_24], 0B9h
.text:0041AD10 C7 45 F4 A6 CE 40 02 mov [ebp+var_C], 240CEA6h
.text:0041AD17 83 75 F4 37 xor [ebp+var_C], 37h
.text:0041AD1B 0F AF 55 F4 imul edx, [ebp+var_C]
.text:0041AD1F 0F BE 06 movsx eax, byte ptr [esi]
.text:0041AD22 33 D0 xor edx, eax
.text:0041AD24 46 inc esi
.text:0041AD25 0F BE 0E movsx ecx, byte ptr [esi]
.text:0041AD28 3B 4D FC cmp ecx, [ebp+var_4]
.text:0041AD2B 75 D5 jnz short loc_41AD02
.text:0041AD2D
.text:0041AD2D loc_41AD2D: ; CODE XREF: HashImportFunctionName+7C↑j
.text:0041AD2D 8B C2 mov eax, edx
.text:0041AD2F 5E pop esi
.text:0041AD30 C9 leave
.text:0041AD31 C3 retn
.text:0041AD32 ; -----
```

Figure 3. Example Mystic Stealer constant obfuscation technique

Encrypted binary custom protocol. The client communicates with the C2 server using a custom protocol over TCP, which we discuss in more depth later.

Polymorphic string obfuscation. We identified that the malware obfuscates strings using a library that is very similar to [ADVobfuscator](#). The obfuscator generates code at compile time that builds strings on the stack, which are then decrypted at runtime. The obfuscation is polymorphic, and therefore, every sample will contain strings that are uniquely encrypted with simple mathematical operations such as addition, subtraction, and XOR. As a result, this technique may bypass static antivirus signatures and complicate malware reverse engineering.

The Mystic Stealer seller refers to this obfuscation as a *morpher* that obfuscates builds with full undetectability (FUD) in sales threads. In one forum, the seller advertised that the project's morpher enabled the bypass of [SmartScreen](#), which members identified as a dubious claim based on the operation of obfuscators and SmartScreen. Some forum users suspected the use of an open-source obfuscator. This ended up as a point of contention in the forum, lowering the perception and trust of the project with some users.

C2 Communication

Mystic Stealer communicates with its command and control (C2) servers using a custom binary protocol over TCP.

1. The client sends a hello message containing a constant 4 byte value (0x946F19B5) to the C2 server.
2. The C2 responds with 256 bytes of binary data that is used as an RC4 key for all subsequent communications.
3. The client obtains the machine GUID from the registry value SOFTWARE\Microsoft\Cryptography\MachineGuid.
4. The client encrypts the GUID value (along with this GUID length) using RC4 and sends it to the C2 server.
5. The format of packets received from the server consists of a 4 byte big endian data size value followed by the data buffer. All data is encrypted with RC4.
6. The C2 server responds back with a binary configuration of the actions to perform (steal credentials, take screenshots, steal cryptocurrency wallets, etc). This configuration is structured by 1's and 0's representing whether to enable or disable a feature, respectively.
7. Data stolen from the infected system is labeled with specific binary tags that identify the type of information when it is sent to the C2 server.
8. Unlike most stealers that will harvest data in full and then exfiltrate it to a C2 server with a single request, Mystic Stealer will collect various types of information and immediately send the data to a C2 server on-the-fly without storing or writing data to the disk, which may be detected by EDR/antivirus applications.

The builder enables operators to specify up to four C2 endpoints. This is often used in crimeware to provide resiliency in case some servers are offline or blocklisted. In Mystic Stealer binaries, there are two arrays consisting of 4 DWORDs each that are encrypted with a modified XTEA-based algorithm. Thus, each sample can configure up to 4 IP addresses and ports. A Python-based implementation of the [decryption algorithm](#) for Mystic C2s is shown below:

```

def uint32(val):
    return val & 0xffffffff

def decrypt(block, key):
    sum = 0xC6EF3720
    delta = 0x61C88647

    key0, key1, key2, key3 = struct.unpack("<4L", key)

    block_size = 8
    num_blocks = len(block) // block_size
    blocks = struct.unpack("<2L" * num_blocks, block)

    v0 = blocks[0]
    v1 = blocks[1]

    for i in range(32):
        v1 = v1 - ((v0 + sum) ^ (key2 + (v0 << 4)) ^ (key3 + (v0 >> 5)))
        v1 = uint32(v1)
        v7 = uint32(v1 + sum)
        sum = uint32(sum + delta)
        v8 = v7 ^ uint32(key0 + (v1 << 4)) ^ uint32(key1 + (v1 >> 5));
        v0 = uint32(v0 - v8);

    return struct.pack("<2L", v0, v1)

```

A few generations of the C2 servers seem to utilize a default port of 16287/tcp as seen in Figure 4 of the control panel builder dialog posted in a sales thread on underground forums. We have not observed file samples where this port was utilized for the configured C2 servers. The following C2 ports have been observed in identified samples, providing some clustering by build configurations:

- 15555/tcp
- 15556/tcp
- 13219/tcp

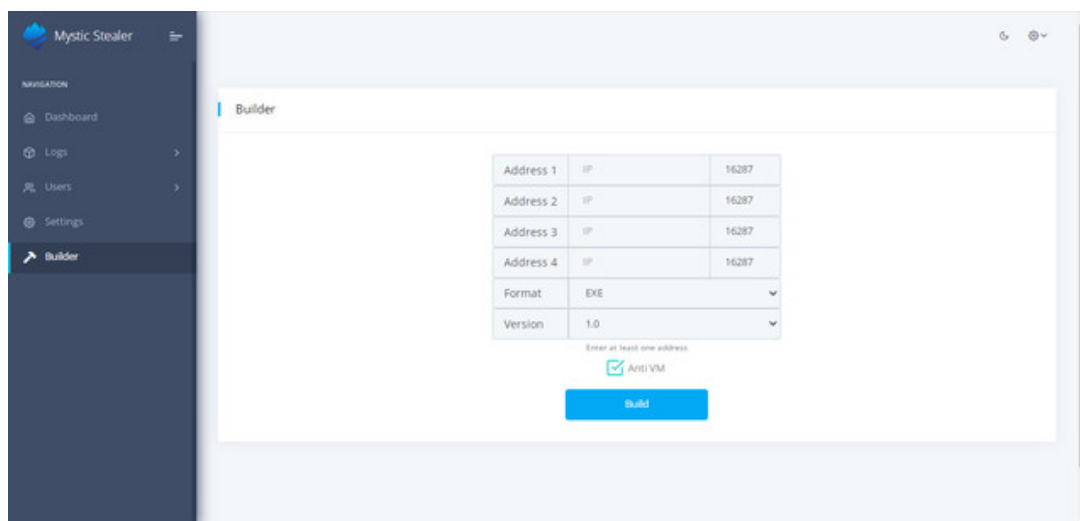


Figure 4. Mystic Stealer control panel builder dialog

C2 server footprint

The stealer has been linked to multiple server-hosting IP addresses across a diverse geographic spectrum, including but not limited to registrations in France, Germany, Russia, the United States, and China. We list C2 servers identified by the hosting panel and C2 callbacks in the appendix. Large commercial hosting provider Hetzner (AS24940) accounts for nearly half of the hosts in addition to a number at OVH (AS16276). However, we also note a number of servers within the Latvian, Bulgarian and Russian hosting spheres. These include:

- Aeza Group Ltd (AS210644)
- GIR-AS (AS207713)
- Partner-AS / LetHost LLC (AS204603)
- Scalaxy B.V. (AS58061)
- Sukhoi Su-57 LLC (AS46308)
- WAICORE-TRANSIT (AS202973)

Some of these providers stand out as potential contenders in the realm of bulletproof hosting, a term that sets off alarm bells in the cybersecurity world. Bulletproof hosting providers are entities that offer services with a particular appeal to individuals and groups engaged in nefarious activities, due to the providers' lax enforcement of legal norms and frequent protection and misdirection efforts that they take on behalf of criminal clientele. These services are often used to host malware, command and control servers, phishing campaigns, and other illicit digital operations. InQuest and Zscaler note a particular tendency of operators of credential stealers and other malware as a service (MaaS) systems to utilize protected backend hosting in the underground services space. This strategy often affords greater capabilities in blocklist avoidance as well as the reduced impact of takedown efforts and law enforcement reach.

The "Grand" cluster

One particular cluster of C2 servers sticks out when searching for hosted control panels. We have labeled this the "Grand" cluster based on WHOIS artifacts seen with some domains. We have included a list of these domains in the appendix. This group of domains is noted to share the following attributes:

- Cloudflare nameservers and CDN fronting
Nameservers: meadow, jimmy
- Registration details:
 - Domains registered mid-late 2022
 - Registrar: Public Domain Registry (PDR Ltd.)
 - Registrant State/Province: Novosibirskaya oblast
 - Registrant Country: RU
 - Registrant: Grand (grand.bbs[[@](mailto:)]yandex.ru)

We note that while the majority of domains follow the above registration convention, a few outliers exist. For example, the domain **alchemistwallet[.]io** is registered with NetEarth One Inc., and one or more domains use different authoritative nameserver pairs (amit, jacqueline; roselyn, stan). One or more domains were additionally registered in 2023.

Several of these domains were mentioned in a note by [FalconFeedsio](#). We believe that these domains were likely picked up from domain aftermarket resale, a tactic that can yield tangible value for an adversary. Already-registered domains carry established reputation attributes based on past usage, and we note that some of these domains carry reputation scores in various datasets indicating that they had relatively high rankings. For example, looking at **gujaratstudy[.]in**, we can see that the domain was most recently registered on 2022-10-07. Prior to this date, in 2021, the domain was registered and hosted by a previous owner, with DNS resolution observed through October of 2021. After the new DNS registration by the Grand persona, the domain was initially live via authoritative DNS in regway.com on 2023-10-08, and then migrated to Cloudflare DNS on 2023-

10-11. This pattern is fairly consistent through domains in the Grand cluster. Another domain, **bhandarapolice[.]org**, appears to have previously been used for the official website of an Indian district police department. The domain's category labels on VirusTotal still reflect a positive reputation: *government, public information, top-1M*. A WHOIS record showing the registration details of a representative domain from this set is available in the appendix.

The following domains and registration dates are samples of some domains found in this cluster:

- HANOIGARDEN[.]NET (2022-07-19)
- BHANDARAPOLICE[.]ORG (2022-07-20)
- ENGTECHJOURNAL[.]ORG (2022-07-20)
- MARISOLBLOOMS[.]COM (2022-07-20)
- WORDCZARMEDIA[.]COM (2022-08-07)
- COLORADOTRUCKIE[.]COM (2022-08-14)
- BABYPICTURESULTRASOUND[.]COM (2022-09-08)
- SACREDSpace-SF[.]COM (2022-09-08)
- TEAMMSOLUTIONS[.]COM (2022-09-08)
- AFRICAHELP[.]ORG (2022-09-13)
- BAYSWATERHOLDING[.]COM (2022-09-20)
- ASHRAYAKRUTIFOUNDATION[.]ORG (2022-10-07)
- GUJARATSTUDY[.]IN (2022-10-07)

The nature of the Grand cluster is not completely known at this time. Until recently, the domains have been live and serving Mystic Stealer control panels as shown in Figure 5 below.

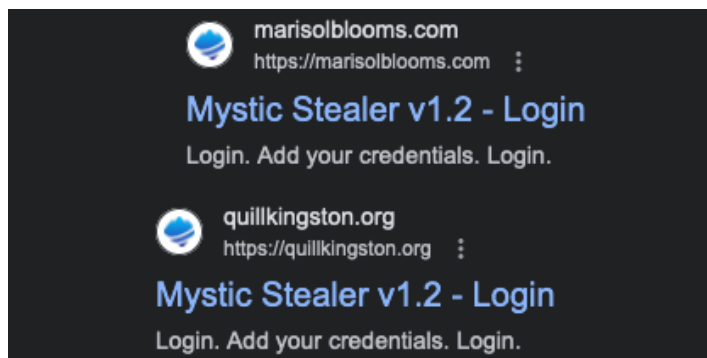


Figure 5. Example Mystic Stealer control panel domains cached in Google Search cache related to the Grand cluster

While possible that they are simply C2 servers, we did not identify file samples associated with them. Recently, many of the sites appear to have gone offline with the upstream CDN reporting connection failures. It may be possible that the domains are part of a traffic distribution or frontend proxy and traffic service.

Control panel

The Mystic Stealer developers provide a web-based admin control panel as shown in Figure 6.

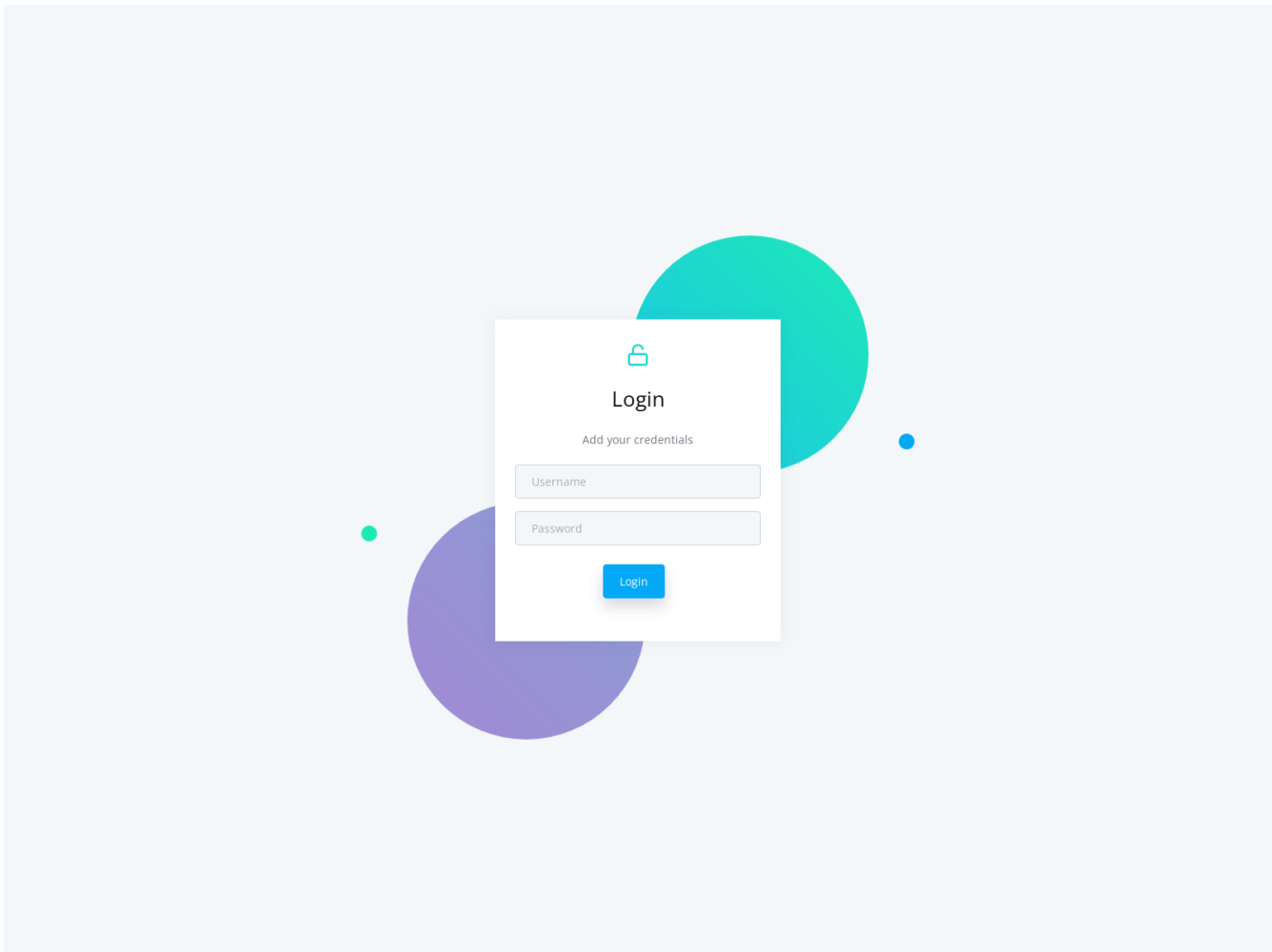


Figure 6. Mystic Stealer web admin control panel login page

Crimeware control panels allow operators to configure settings and access data collected from deployed malware and typically serve as the interface for criminal users to interact with the software. Common functions include statistics dashboards, malware builders, controlling options and features, credential log and data access, integration configurations, and more. The Mystic Stealer control panel operates out of band on a separate exposed service port than the malware utilizes for C2 communications. The developers utilize the Python Django web framework for the control panel. While not exclusive, the use of Python frameworks in crimeware development, typically dominated by PHP applications, is somewhat rare. As a historical example, another crimeware project implemented on Django was the Nice Pack exploit kit.

The control panel is deployed on a customer's server. The commonly observed service port for deployed panels is 443/tcp. An earlier observed deployment in March 2023 utilized 8005/tcp.

A number of community members have shared information identifying IP addresses of hosting panels. A number of these are also identified and archived on urlscan.io:

2023-03-22 <https://urlscan.io/result/535841c6-ea4a-4e8c-85b7-e19bd5ad68e5>

Control panel - hXXp://164.132.200[.]171:8005/login/

2023-03-22 <https://urlscan.io/result/7b2e16cb-9b66-4192-8b69-98fb89fa12ea/>

Control panel - hXXp://164.132.200[.]171:8005/login/

2023-05-02 <https://urlscan.io/result/3fdaf5e7-a741-4cb8-8fa9-dedb00b1672b>

Control panel - hXXp://135.181.47[.]95/login/

2023-05-02 <https://urlscan.io/result/5d326ed9-3bcc-40f3-9fd2-2bdea6fd800f>

Control panel - hXXp://95.216.32[.]74/login/

2023-05-04 <https://urlscan.io/result/882d8d05-1523-41eb-892f-ba58d6656512/>

Control panel - hXXp://185.252.179[.]18/

2023-05-04 <https://urlscan.io/result/cc6be796-ee37-4cc4-a37f-c9abb9bf17bc/>

Django admin control panel - hXXp://185.252.179[.]18/admin/

2023-05-15 <https://urlscan.io/result/16f972cb-adb8-486a-9bff-3bebb673792e/>

Control panel - hXXp://212.113.106[.]114/login/

2023-05-25 <https://urlscan.io/result/b5224ba6-1b50-42b0-b453-46204ebd1358/>

Django admin control panel - hXXp://www.coloradotruckie[.]com/admin/

2023-06-05 <https://urlscan.io/result/016de1c6-cb24-4e3a-9ffa-5f8c21edf2c5/>

Control panel - hXXp://213.142.147[.]235/login/

Tracking an installation of a control panel for the month of May, we've seen the version of the deployed panel change, likely reflecting upgrades by the customer:

- 2023-05-03: Mystic Stealer - Login
- 2023-05-08: Mystic Stealer v1.1 - Login
- 2023-05-31: Mystic Stealer v1.2 - Login

We also note that the utilized page style is not exclusive to Mystic Stealer, appearing to be borrowed from or relating to a more broadly accessible template seen with other applications. The control panel UI kit appears to be based on [Datta Able for Django](#). We do not believe there is any connection between this project and Mystic Stealer. It is likely that the Mystic Stealer developer is simply using the publicly available open-source UI kit.

Presence on Underground Forums

Mystic Stealer made its public debut on underground forums in late April 2023, several weeks after initial samples were known to surface. A seller named *Mystic Stealer* joined the WWH (WWH-Club) and BHF (Best Hack Forums, using the name *MysticStealer*) forums just a couple of days before posting, and, the stealer was listed for rent at a price of \$150 per month. The seller later advertised Mystic Stealer on the XSS forum. Information-stealing trojans are a hot commodity in the underground economy, underscoring the level of emphasis the criminal community places on the collection of credentials to drive initial access into target user accounts and network environments. With its comprehensive data collection capabilities, it's no surprise that Mystic Stealer has caught the attention of members of these forums. According to observed advertisements, this seller also operates a Telegram account named [@mysticstealer](#) and the channel [t\[.\]me/+ZjiasReCKmo2N2Rk](#) (Mystic Stealer News).

Conclusion

As Mystic Stealer is a new player, it's hard to predict its trajectory. What's clear, however, is that it's a sophisticated threat with the potential for widespread damage. Over the past few weeks, we've observed a fascinating dance of panels appearing and disappearing. Yet, amidst this volatility, a number of these elusive entities have maintained their persistent presence. These patterns could be attributed to a range of factors: perhaps a surge in fresh sales, the relentless pursuit of takedowns, or the unpredictable behavior of the customers themselves.

This was a joint research collaboration between Zscaler ThreatLabz and InQuest. Special thanks to all of those involved from InQuest Labs.

Cloud sandbox

The screenshot displays a Zscaler Cloud Sandbox report for a file named 'Win32.Downloader.Smokeloader'. The report is dated 6/14/2023 2:24:47 PM. The threat score is 90, indicating a high risk. The report is categorized as 'Malicious' and 'Malware & Botnet Detected'. The report includes sections for Classification, Mitre ATT&CK, Virus and Malware, Security Bypass, Networking, Stealth, Spreading, Information Leakage, Exploiting, Persistence, System Summary, and Download Summary.

Section	Details
CLASSIFICATION	Class Type: Malicious Threat Score: 90 Category: Malware & Botnet Detected Win32.Downloader.Smokeloader
MITRE ATT&CK	This report contains 5 ATT&CK techniques mapped to 3 tactics
VIRUS AND MALWARE	Troj/Krypt-XU Win32.Downloader.Smokeloader
SECURITY BYPASS	May Try To Detect The Virtual Machine To Hinder Analysis
NETWORKING	URLs Found In Memory Or Binary Data
STEALTH	Disables Application Error Messages
SPREADING	No suspicious activity detected
INFORMATION LEAKAGE	No suspicious activity detected
EXPLOITING	Known MD5
PERSISTENCE	Creates Temporary Files
SYSTEM SUMMARY	One Or More Processes Crash Uses 32bit PE Files PE File Has An Executable .Text Section And No Other Executable Section Reads Software Policies Classification Label Sample May Be VM Or Sandbox-Aware. Try Analysis On A Native Machine Spawns Processes
DOWNLOAD SUMMARY	Original file: 250 KB Dropped files: 2 MB Packet capture: 186 KB

In addition to sandbox detections, Zscaler's multilayered cloud security platform detects indicators related to Mystic Stealer at various levels with the following threat names:

[Win32.Trojan.Mystic.KV](#)

Appendix

C2 server endpoints observed in recent bot configurations

- 194.169.175[.]123:13219
- 185.252.179[.]18:13219
- 142.132.201[.]228:13219
- 135.181.47[.]95:13219
- 94.130.164[.]47:13219
- 94.23.26[.]20:13219
- 91.121.118[.]80:13219

Targeted web browsers

- Opera
- K-Meleon
- Mozilla icecat
- Mozilla Firefox
- Comodo IceDragon
- 8pecxstudios Cyberfox
- NETGATE Technologies BlackHawk
- Torch
- Chedot
- Kometa
- liebao
- Comodo
- Iridium
- Vivaldi
- Orbitum
- K-Melon
- Chromium
- QIP Surf
- Maxthon3
- Nichrome
- Chromodo
- Amigo
- 7Star
- CentBrowser
- Mail.Ru Atom
- Google Chrome
- Coowon
- uCozMedia Uran
- CocCoc Browser
- Microsoft Edge
- Sputnik
- Elements Browser
- 360Browser
- Epic Privacy Browser
- CatalinaGroup Citrio
- YandexBrowser
- MapleStudio ChromePlus
- Brave-Browser
- Fenrir Inc Sleipnir5 ChromiumViewer

Targeted MFA and cryptocurrency wallet browser extensions

Extension ID	Browser Extension Name
lbnejdfjmmkpcnlpebklmknkoeiohofec	TronLink

fhbohimaelbohpbjblcdcngcnapndodjp	BinanceChain
ffnbelfdoeiohenkjbmadjehjhajb	Yoroi
jbdaocneiiimjbjlgalhcelgbejmnid	Nifty Wallet
afbcbjbpfadlkmhmcLhkeeodmamcflc	Math Wallet
hnfanknocfeofbddgcijnmhnfnkdnaad	Coinbase Wallet
hpglfhghfnhbgpjdenjgmdgoeiappafln	Guarda
blnieiiffboillknjnegoghkgnoapac	EQUAL Wallet
cjelfplplebdjjenllpjcbmljkfcffne	Jaxx Liberty
fihkakfobkmkjojpchpfgcmhfjnmnfpj	BitApp Wallet
kncchdigobghenbbaddojjnaogfppfj	iWallet
amkmjimmflddogmhpjloimipbofnfjih	Wombat
nlbmnnijcnlegkjjpcfjclmcfggfefdmd	MEW CX
nanjmdknhkinifnkgdcggcfnhdaammj	GuildWallet
nkddgncdjgjfcdamfgcmfnlhccnimig	Saturn Wallet
fnjhmkhhmkbjkkabndcnnogagobneec	Ronin Wallet
cphhlgmgameodnhkjdmkpanelnlohao	NeoLine
nhnkbkgjkgcigadomkphalanndcapjk	Clover Wallet
kpfopkelmapcoipemfendmdcghnegimn	Liquidity Wallet
aiifbnfbobpmeekipheeiijmdpnlpgpp	Terra Station
dmkamcknogkgcdfhbbddcghachkejeap	Keplr
fhmfendgdocmcbmfikdcogofphimnkno	Sollet

cnmamaachppnkjgnildpdmkaakejnhae	Auro Wallet
jojhfloedkpkglbfimdfabpdfjaoolaf	Polymesh Wallet
flpiciilemghbmfalicajoolhkkenfel	ICONex
nknhiehlklippafakaeklbeglecifhad	Nabox Wallet
hcfpincpppdclinealmandijcmnkbgn	KHC
ookjlbkiijnhpmnjffcofjonfbgaoc	Temple
mnfifekajgofkckjemidiaecocnkjeh	TezBox
lodccjbbdhfakaekdiahmedfbieldgik	DAppPlay
ljmpgkjfbfhoebgogflfebnejmfbml	BitClip
lkcjlnjfbikmcmcbachjpd bijejflpcm	Steem Keychain
nkbihfbeogaeaoehlefnkodbefgpgknn	MetaMask
bcopgchhojmggmffilplmbdicgaihkp	Hycon Lite Client
klnaejjgibimbhlephnhpmaofohgkpgkd	ZilPay
aeachknmefphepccionboohckonoemg	Coin98 Wallet
bhghoamapcdpbohphigooaddinpkbai	Authenticator
dkdedlpgdmmkkfjabffeganieamfklkm	Cyano Wallet
nlgbhdfgdhgbiamfdmbikcdghidoadd	Byone
onofpnbbkehpmmoabgpcpmigafmmnjhl	Nash Extension
cihmoadaigncejobammfbdmdekcje	Leaf Wallet
gaedmjdmmahhbjeafbgaolhanlaolb	Authy 2FA
oeljdldpnmdbchonielidgobddffflal	EOS Authenticator

ilgcnhelpchnceepijajklblbcobl	GAuth Authenticator
imloifkgjagghnncjkhhggdhalmcnfklk	Trezor Password Manager
infeboajghbjppbeppbkgnabfdkdaf	OneKey
cgeeodpfagjceefieflmdfphplkenlfk	EVER Wallet
pdadjkfkcgafgbceimcpbkalfnfnepbnk	KardiaChain Wallet
acmacodkjbdgmoleebolmdjonilkdbch	Rabby Wallet
bfnaelmomeimhlpmgjnphhpkkoljpa	Phantom
fhilaheimglignddkjgofkcbgekhenbh	Oxygen - Atomic Crypto Wallet
mgffkfbidihjpoaomajlbgchddlicgpn	Pali Wallet
hmeobnfnfcmkdcmlblgagmfpfboieaf	XDEFI Wallet
lpfcbjknijpeeillifnkikgncikgfhdo	Nami
dngmlblcodfobpdpecaadgfbcgjfnm	MultiversX DeFi Wallet
bhhhlbepdkbapadjdnnojkbgioidbic	Solflare Wallet
jnkelfanjkeadonecabehalmbgpfodjm	Goby
jhgnbkkipaallpehbohjmkbjofjdmeid	SteemKeychain
jnlgamecbpmbajffhmmmlhejkemejdma	Braavos Smart Wallet
kkpllkodjeloidieedojogacfhpaihoh	Enkrypt: Ethereum, Polkadot & RSK Wallet
mcohilncbfahbmgdjkbpemcciolgcge	OKX Wallet
gjagmgiddbbciopjhllkdnddhcglnemk	Hashpack
kmhchipebfmpgmihbkipmjlmioameka	Eternal
phkbamefinggmgakglpklijmgibohnba	Pontem Aptos Wallet

lpilbniiabackdjcionkobgImdddfbcjo	Keeper Wallet
cjmknjdjhnagcfbpiemnkdpomccnjblmj	Finnie
aijcbedoijmgnlmjeegjaglmepbmpkpi	Leap Terra Wallet
fdjamakpfbdddfjaoaikfcpapjohcfmg	Dashlane — Password Manager
foolghllnmhmmndgjiamiodkpenpbb	NordPass® Password Manager & Digital Vault
pnlccmojcmehlpggmfnbbiapkmbliob	RoboForm Password Manager
hdokiejnpimakedhajhdlcegeplioahd	LastPass: Free Password Manager
naepdomgkenhinolocfifgehiddafch	Browserpass
bmikpgodpkclnkgmnppehdgcimmided	MYKI Password Manager & Authenticator
efbglgofoipbgcjepnhiblaibcnclgk	Martian Wallet for Sui & Aptos

Targeted cryptocurrency applications

- MyMonero
- Exodus
- Binance
- Raven
- Armory
- Dogecoin
- MultiBit
- Bitcoin
- DashCore
- Electrum
- Litecoin
- BitcoinGold
- WalletWasabi
- Atomic
- Guarda
- Electrum-LTC
- MyCrypto
- Bisq
- DeFi Blockchain
- Coinomi
- TokenPocket

Network signatures

The following Suricata signatures detect the initial C2 connection key exchange:

```
alert tcp $HOME_NET any -> $EXTERNAL_NET any (msg:"LOCAL Mystic Stealer C2 Client Hello Packet";  
flow:established,to_server; flowbits:set, mystic_stealer_conn_init; flowbits:noalert; dsizе:4; content:"|b5 19 6f  
94|"; fast_pattern; reference:md5,df80b1e50cfebb0c4dbf5ac51c5d7254; classtype:trojan-activity; sid:9999990;  
rev:1; metadata:created_at 2023_06_02, malware_family Mystic Stealer, signature_severity Major, updated_at  
2023_06_02;)
```

```
alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"LOCAL Mystic Stealer C2 Session Key Response  
Packet"; flow:established,to_client; flowbits:isset, mystic_stealer_conn_init; dsizе:256;  
reference:md5,df80b1e50cfebb0c4dbf5ac51c5d7254; classtype:trojan-activity; sid:9999991; rev:1;  
metadata:created_at 2023_06_02, malware_family Mystic Stealer, signature_severity Major, updated_at  
2023_06_02;)
```

Indicators of Compromise

Sample hashes

Hash	Notes
<u>47439044a81b96be0bb34e544da881a393a30f0272616f52f54405b4bf288c7c</u> Imphash: 8f2649698c183ba2b52e5e425852109d	1367.exe (2023-03-18) <ul style="list-style-type: none">• Communicates with 164.132.200[.]171:15555• Size ~234 KB• Compiler: EP:Microsoft Visual C/C++ (2017 v.15.5-6) [EXE32]• Early build
<u>5c0987d0ee43f2d149a38fc7320d9ffd02542b2b71ac6b5ea5975f907f9b9bf8</u> Imphash: d6d4965d7fe2d90a52736f0db331f81a	Mystic Stealer (2023-04-28) <ul style="list-style-type: none">• Communicates with 94.23.26[.]20:13219• Size ~211 KB• Compiler: EP:Microsoft Visual C/C++ (2017 v.15.5-6) [EXE32]
<u>acba3311b319a60192be2e29aa8038c863a794be39603a21ee8ee4ccc3ebfca6</u> Imphash: d6d4965d7fe2d90a52736f0db331f81a	update.exe (2023-05-01) <ul style="list-style-type: none">• Communicates with 185.252.179[.]18:13219• Size ~209 KB• Compiler: EP:Microsoft Visual C/C++ (2017 v.15.5-6) [EXE32]

<u>7c185697d3d3a544ca0cef987c27e46b20997c7ef69959c720a8d2e8a03cd5dc</u>	update.exe (2023-05-02)
Imphash: d6d4965d7fe2d90a52736f0db331f81a	<ul style="list-style-type: none"> • Communicates with 185.252.179[.]18:13219 • Size ~225 KB • Compiler: EP:Microsoft Visual C/C++ (2017 v.15.5-6) [EXE32]
<u>8592e7e7b89cac6bf4fd675f10cc9ba319abd4aa6eaa00fb0b1c42fb645d3410</u>	Mystic Stealer (2023-05-04)
Imphash: d6d4965d7fe2d90a52736f0db331f81a	<ul style="list-style-type: none"> • Communicates with 185.252.179[.]18:13219 • Size ~208 KB • Compiler: EP:Microsoft Visual C/C++ (2017 v.15.5-6) [EXE32]
<u>45d29afc212f2d0be4e198759c3c152bb8d0730ba20d46764a08503eab0b454f</u>	Mystic Stealer (2023-05-07)
Imphash: 9cd292d1fac1768b38a49bc6b288c67d	<ul style="list-style-type: none"> • Communicates with 135.181.47[.]95:13219 • Size ~180 KB • Compiler: EP:Microsoft Visual C/C++ (2017 v.15.5-6) [EXE32]
<u>30fb52e4bd3c4866a7b6ccedcfa7a3ff25d73440ca022986a6781af669272639</u>	qawsed.exe (2023-05-20)
Imphash: 9cd292d1fac1768b38a49bc6b288c67d	<ul style="list-style-type: none"> • Communicates with 142.132.201[.]228:13219 • Compiler: EP:Microsoft Visual C/C++ (2017 v.15.5-6) [EXE32]
<u>ce56e45ad63065bf16bf736dccb452c48327803b434e20d58a6fed04f1ce2da9</u>	Mystic Stealer (2023-05-22)
Imphash: 9cd292d1fac1768b38a49bc6b288c67d	<ul style="list-style-type: none"> • Communicates with 94.130.164[.]47:13219 • Size ~187 KB • Compiler: EP:Microsoft Visual C/C++ (2017 v.15.5-6) [EXE32]
<u>7ab8f9720c5f42b89f4b6fed21e7aa20334ba1230c3aef34b0e6481a3425681</u>	894d.exe (2023-05-23)
Imphash: 1c8b7141d44e96dcc8c22d3bfdac433c	<ul style="list-style-type: none"> • Communicates with 91.121.118[.]80:13219 • Size ~249 KB • Compiler: EP:Microsoft Visual C/C++ (2008-2010) [EXE32] • Sample is packed

fc4aa58229b6b2b948325f6630fe640c2527345ecb0e675592885a5fa6d26f03

Mystic Stealer (2023-05-25)

Imphash: baa93d47220682c04d92f7797d9224ce

- Communicates with 167.235.34[.]144:13219
- Size ~1.79 MB
- Sample is packed

References

Analysis resources

https://github.com/Microv/MysticStealer_HashResolver