8Base ransomware stays unseen for a year

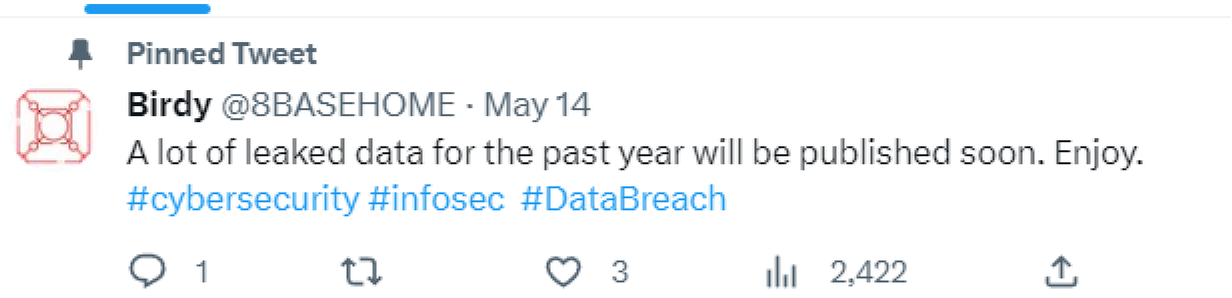
acronis.com/en-sg/cyber-protection-center/posts/8base-ransomware-stays-unseen-for-a-year/

Summary

- Comes to victims via SmokeLoader malware
- Sample is a PE32 file, written in C\C++
- Modified version of Phobos ransomware
- Encrypts users' files with AES-256-CBC cipher
- Writes IV and encrypted AES key to the end of encrypted files
- Data leak site shares similarities with the RansomHouse site

Introduction

8Base ransomware was first spotted in June 2023, with a massive number of targeted victims. It was later discovered that 8Base originated in March 2022 with the launch of an associated data leak site. 8Base also has a Twitter account, which was created in 2014. In the account’s pinned post, the threat actors announced the publication of leaked data from the past year's operation, indicating that in addition to encrypting user files, the group has also exfiltrated data to its own servers.



Pinned Tweet

Birdy @8BASEHOME · May 14

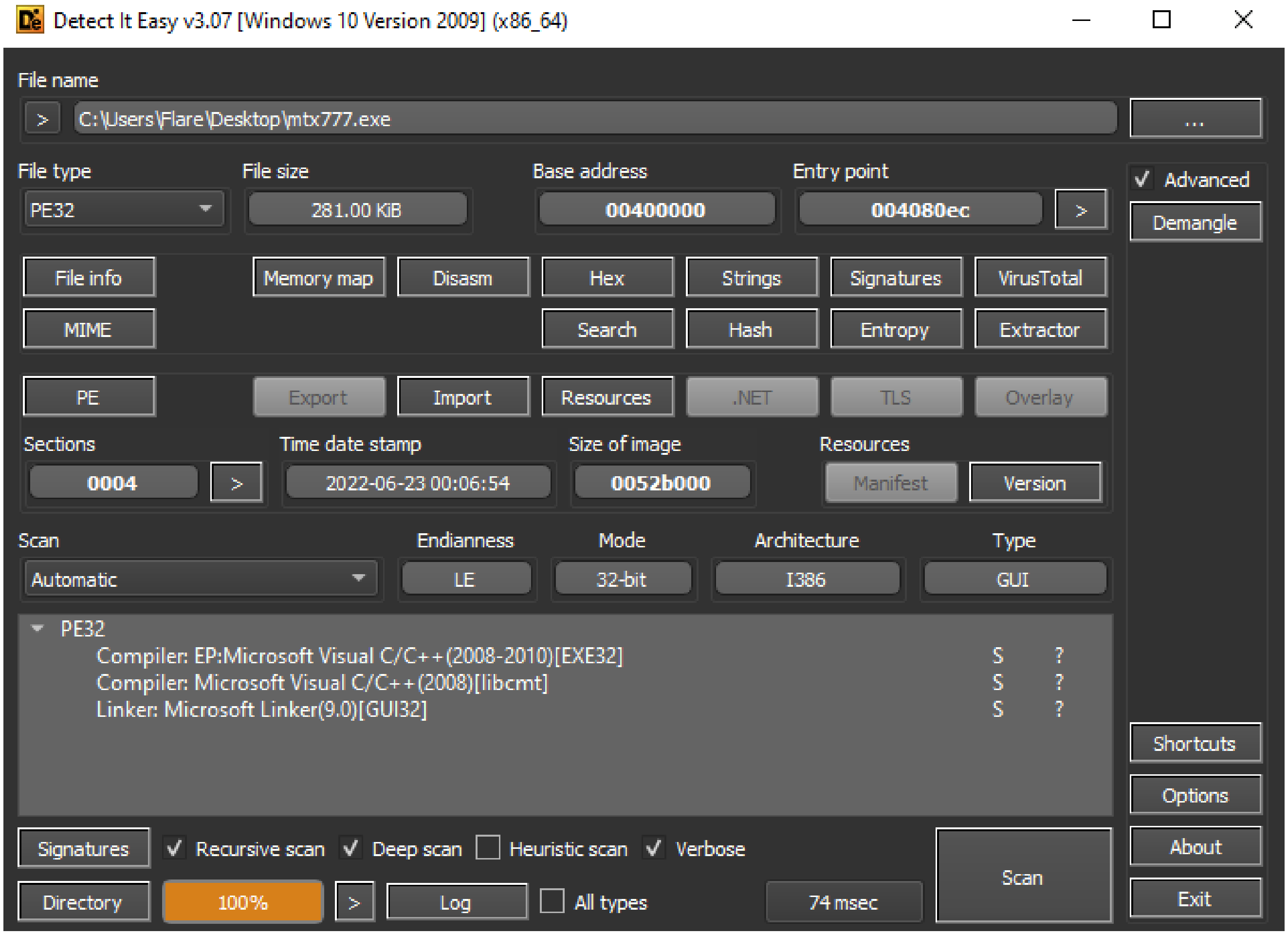
A lot of leaked data for the past year will be published soon. Enjoy. #cybersecurity #infosec #DataBreach

To deliver 8Base ransomware to the victims' machines, threat actors used SmokeLoader, a botnet that is very popular for ransomware attacks. In addition to malware downloading capabilities, SmokeLoader also has a backdoor function that allows threat actors to exfiltrate victims' data.

Technical details

Overview

The 8Base ransomware sample is a PE32 file, written in C\C++. The compilation timestamp ‘2022-06-23’ matches the start of gang operations. As was mentioned before, its activity was spotted only in June 2023, so this sample remained unseen until this moment.



Detect It Easy v3.07 [Windows 10 Version 2009] (x86_64)

File name: C:\Users\Flare\Desktop\mtx777.exe

File type: PE32
File size: 281.00 KiB
Base address: 00400000
Entry point: 004080ec

☑ Advanced

Sections: 0004
Time date stamp: 2022-06-23 00:06:54
Size of image: 0052b000

Scan: Automatic
Endianness: LE
Mode: 32-bit
Architecture: I386
Type: GUI

```
PE32
    Compiler: EP:Microsoft Visual C/C++(2008-2010)[EXE32]    S   ?
    Compiler: Microsoft Visual C/C++(2008)[libcmt]           S   ?
    Linker: Microsoft Linker(9.0)[GUI32]                     S   ?
```

☑ Recursive scan ☑ Deep scan □ Heuristic scan ☑ Verbose

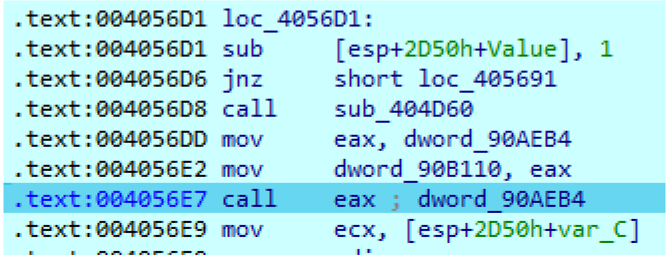
□ All types

100%

74 msec

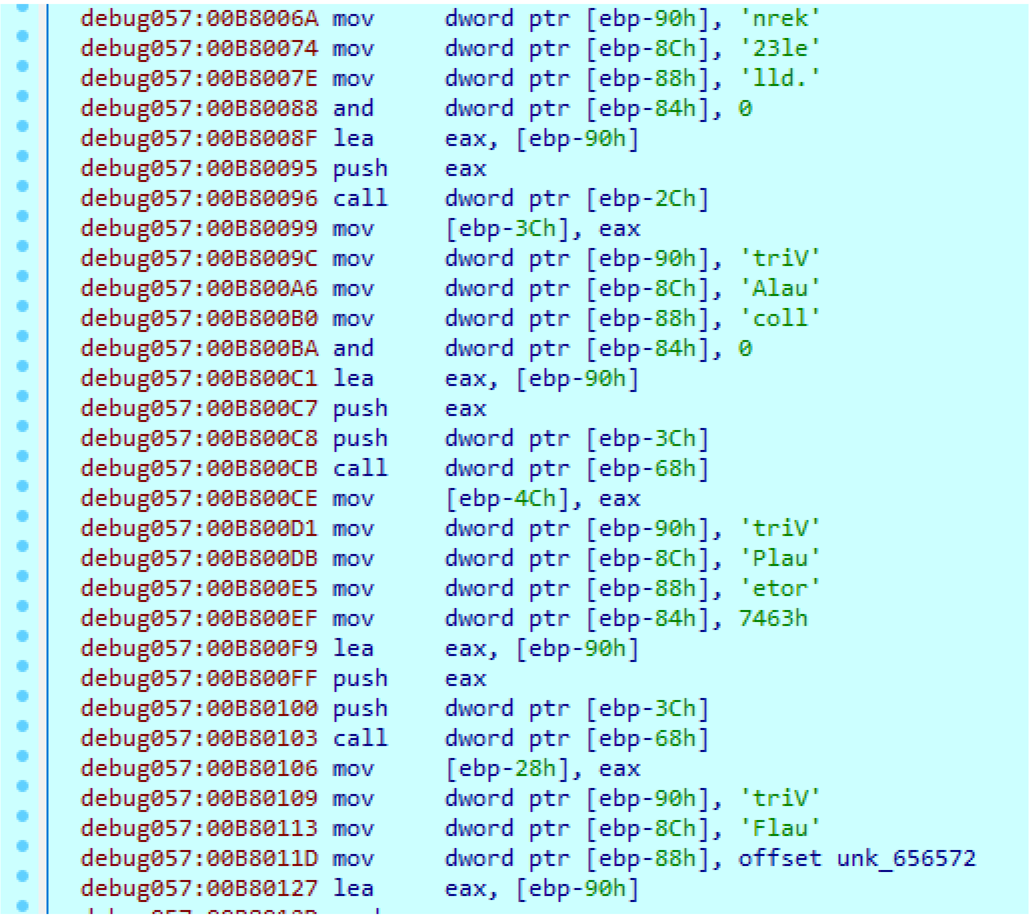
Execution

At the start of execution, 8Base decrypts some executable code, loads it to the 'eax' register, and calls it.



```
.text:004056D1 loc_4056D1:
.text:004056D1 sub     [esp+2D50h+Value], 1
.text:004056D6 jnz     short loc_405691
.text:004056D8 call    sub_404D60
.text:004056DD mov     eax, dword_90AEB4
.text:004056E2 mov     dword_90B110, eax
.text:004056E7 call    eax ; dword_90AEB4
.text:004056E9 mov     ecx, [esp+2D50h+var_C]
```

While the sample file doesn't have a lot of imports, during execution, it loads separated parts of import names and saves them to local variables for further use.

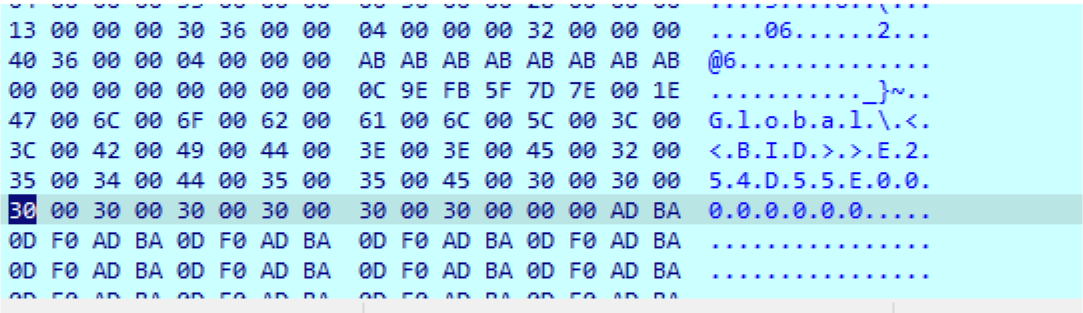


```
debug057:00B8006A mov     dword ptr [ebp-90h], 'nrek'
debug057:00B80074 mov     dword ptr [ebp-8Ch], '23le'
debug057:00B8007E mov     dword ptr [ebp-88h], 'lld.'
debug057:00B80088 and     dword ptr [ebp-84h], 0
debug057:00B8008F lea     eax, [ebp-90h]
debug057:00B80095 push    eax
debug057:00B80096 call    dword ptr [ebp-2Ch]
debug057:00B80099 mov     [ebp-3Ch], eax
debug057:00B8009C mov     dword ptr [ebp-90h], 'triV'
debug057:00B800A6 mov     dword ptr [ebp-8Ch], 'Alau'
debug057:00B800B0 mov     dword ptr [ebp-88h], 'coll'
debug057:00B800BA and     dword ptr [ebp-84h], 0
debug057:00B800C1 lea     eax, [ebp-90h]
debug057:00B800C7 push    eax
debug057:00B800C8 push    dword ptr [ebp-3Ch]
debug057:00B800CB call    dword ptr [ebp-68h]
debug057:00B800CE mov     [ebp-4Ch], eax
debug057:00B800D1 mov     dword ptr [ebp-90h], 'triV'
debug057:00B800DB mov     dword ptr [ebp-8Ch], 'Plau'
debug057:00B800E5 mov     dword ptr [ebp-88h], 'etor'
debug057:00B800EF mov     dword ptr [ebp-84h], 7463h
debug057:00B800F9 lea     eax, [ebp-90h]
debug057:00B800FF push    eax
debug057:00B80100 push    dword ptr [ebp-3Ch]
debug057:00B80103 call    dword ptr [ebp-68h]
debug057:00B80106 mov     [ebp-28h], eax
debug057:00B80109 mov     dword ptr [ebp-90h], 'triV'
debug057:00B80113 mov     dword ptr [ebp-8Ch], 'Flau'
debug057:00B8011D mov     dword ptr [ebp-88h], offset unk_656572
debug057:00B80127 lea     eax, [ebp-90h]
```

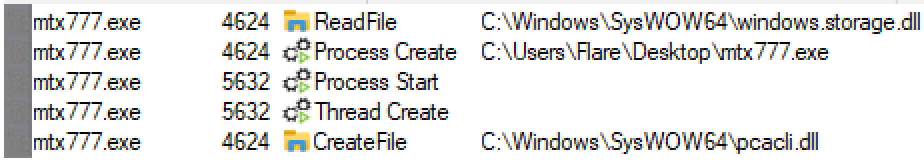
Here are some imports used to work with files, loaded during execution:

kernel32_FindClose
kernel32_FindNextFileW
kernel32_SystemTimeToFileTime
kernel32_FindFirstFileW
kernel32_MoveFileW
kernel32_GetFileSizeEx
kernel32_SetFilePointerEx
kernel32_SetEndOfFile
kernel32_SetFilePointer
kernel32_GetLogicalDrives
kernel32_CopyFileW
kernel32_GetFileAttributesW
kernel32_ReadFile
kernel32_WriteFile

8Base then loads the mutex name and checks if it already exists. If so, it will terminate execution; if not, it creates a mutex and a new process of itself with the 'CreateProcessW' function.



```
13 00 00 00 30 36 00 00  04 00 00 00 32 00 00 00  ....06......2...
40 36 00 00 04 00 00 00  AB AB AB AB AB AB AB AB  @6..............
00 00 00 00 00 00 00 00  0C 9E FB 5F 7D 7E 00 1E  ..........._}~..
47 00 6C 00 6F 00 62 00  61 00 6C 00 5C 00 3C 00  G.l.o.b.a.l.\.<.
3C 00 42 00 49 00 44 00  3E 00 3E 00 45 00 32 00  <.B.I.D.>.>.E.2.
35 00 34 00 44 00 35 00  35 00 45 00 30 00 30 00  5.4.D.5.5.E.0.0.
30 00 30 00 30 00 30 00  30 00 30 00 00 00 AD BA  0.0.0.0.0.0.....
0D F0 AD BA 0D F0 AD BA  0D F0 AD BA 0D F0 AD BA  ................
0D F0 AD BA 0D F0 AD BA  0D F0 AD BA 0D F0 AD BA  ................
```



```
mtx777.exe    4624  ReadFile        C:\Windows\SysWOW64\windows.storage.dll
mtx777.exe    4624  Process Create  C:\Users\Flare\Desktop\mtx777.exe
mtx777.exe    5632  Process Start
mtx777.exe    5632  Thread Create
mtx777.exe    4624  CreateFile      C:\Windows\SysWOW64\pcacli.dll
```

Before encrypting files, 8Base takes some preparatory steps. First, it copies itself to three different folders on the system:

C:\Users\Flare\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup
C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp\mtx777.exe
C:\Users\Flare\AppData\Local\mtx777.exe

Next, it creates new Registry keys to enable itself to auto-start:

HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\mtx777
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\mtx777

It modifies some keys, responsible for internet policy:

HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\ProxyBypass 1
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\IntranetName 1
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\UNCAsIntranet 1
HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\ZoneMap\AutoDetect 0

8Base then uses the ‘Wow64DisableWow64FsRedirection’ function to disable file system redirection.

It executes some commands to delete shadow copies, backup catalogs, change BootStatusPolicy and disable Recovery Mode.

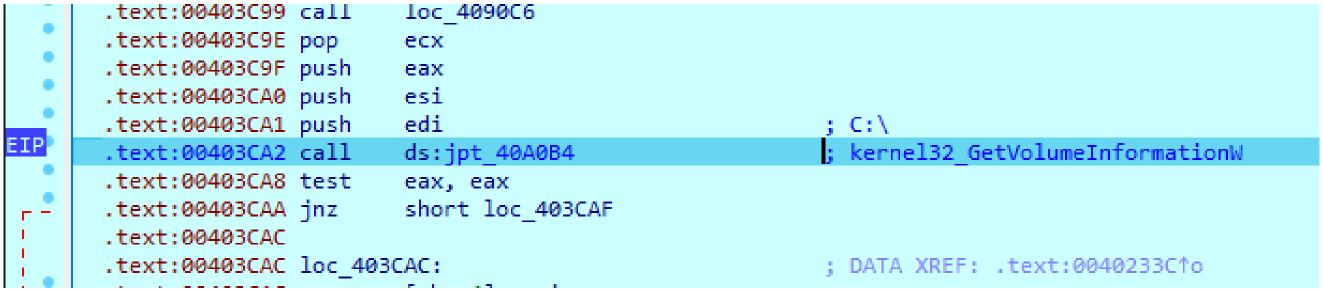
vssadmin delete shadows /all /quiet
wmic shadowcopy delete
bcdedit /set {default} bootstatuspolicy ignoreallfailures
bcdedit /set {default} recoveryenabled no
wbadmin delete catalog -quiet

It also executes the following commands to disable the firewall:

netsh advfirewall set currentprofile state off
netsh firewall set opmode mode=disable

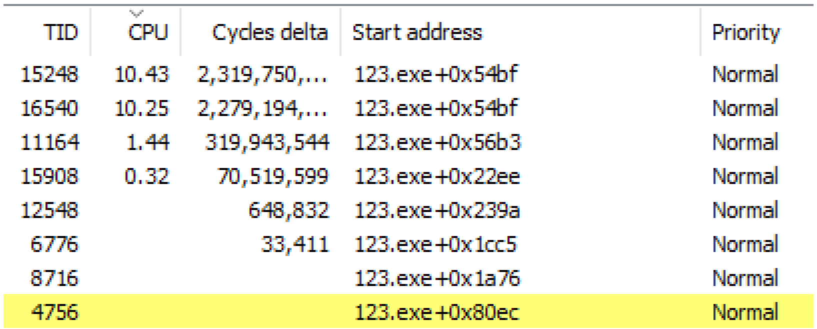
File encryption

8Base begins searching for available drives on the system with ‘GetLogicalDrives’ and obtains information about them.



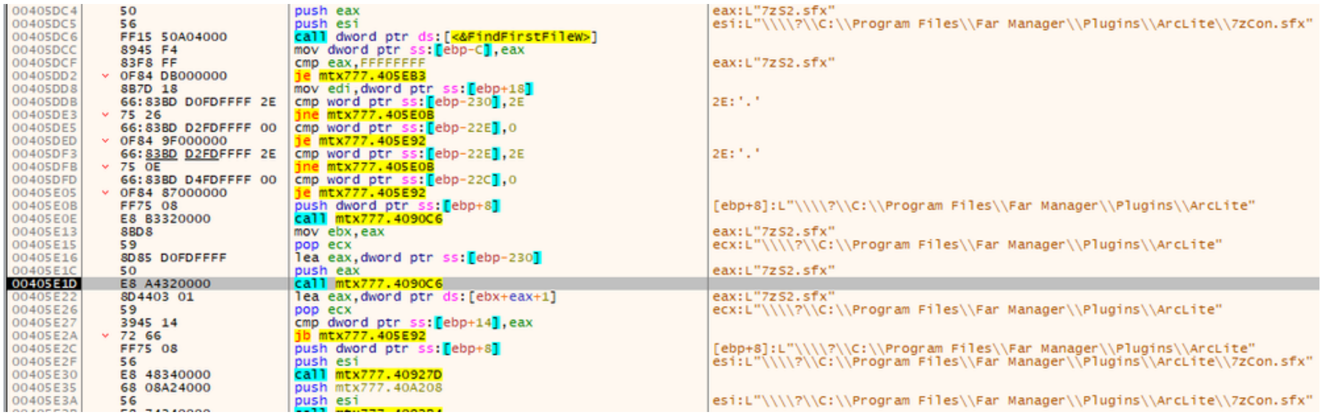
```
.text:00403C99 call    loc_4090C6
.text:00403C9E pop     ecx
.text:00403C9F push    eax
.text:00403CA0 push    esi
.text:00403CA1 push    edi                          ; C:\
.text:00403CA2 call    ds:jpt_40A0B4                ; kernel32_GetVolumeInformationW
.text:00403CA8 test    eax, eax
.text:00403CAA jnz     short loc_403CAF
.text:00403CAC
.text:00403CAC loc_403CAC:                          ; DATA XREF: .text:0040233C↑o
```

Then it starts creating encryption threads:



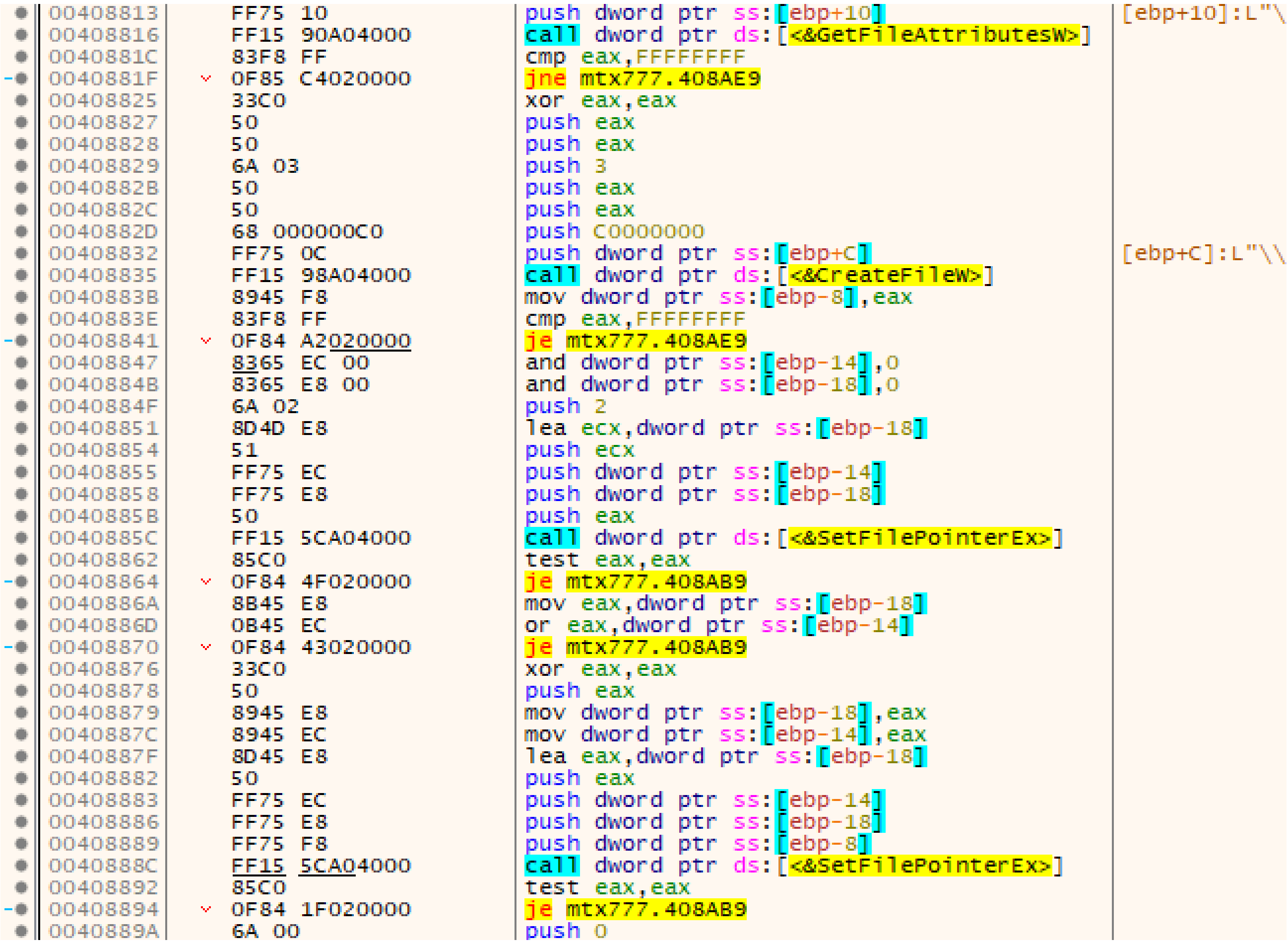
TID	CPU	Cycles delta	Start address	Priority
15248	10.43	2,319,750,...	123.exe+0x54bf	Normal
16540	10.25	2,279,194,...	123.exe+0x54bf	Normal
11164	1.44	319,943,544	123.exe+0x56b3	Normal
15908	0.32	70,519,599	123.exe+0x22ee	Normal
12548		648,832	123.exe+0x239a	Normal
6776		33,411	123.exe+0x1cc5	Normal
8716			123.exe+0x1a76	Normal
4756			123.exe+0x80ec	Normal

To search files on the drive, 8Base uses the ‘FindFirstFileW’ and ‘FindNextFileW’ functions. During encryption, it skips the ‘C:\Windows’ folder, files with its own extension, and ransom note files. Other found files are given to the encryption thread.



```
00405DC4   50                     push eax                              eax:L"7zS2.sfx"
00405DC5   56                     push esi                              esi:L"\\\\?\\C:\\Program Files\\Far Manager\\Plugins\\ArcLite\\7zCon.sfx"
00405DC6   FF15 50A04000          call dword ptr ds:[<&FindFirstFileW>]
00405DCC   8945 F4                mov dword ptr ss:[ebp-C],eax
00405DCF   83F8 FF                cmp eax,FFFFFFFF                      eax:L"7zS2.sfx"
00405DD2   0F84 DB000000          je mtx777.405EB3
00405DD8   8B7D 18                mov edi,dword ptr ss:[ebp+18]
00405DDB   66:83BD D0FDFFFF 2E    cmp word ptr ss:[ebp-230],2E          2E:'.'
00405DE3   75 26                  jne mtx777.405E0B
00405DE5   66:83BD D2FDFFFF 00    cmp word ptr ss:[ebp-22E],0
00405DED   0F84 9F000000          je mtx777.405E92
00405DF3   66:83BD D2FDFFFF 2E    cmp word ptr ss:[ebp-22E],2E          2E:'.'
00405DFB   75 0E                  jne mtx777.405E0B
00405DFD   66:83BD D4FDFFFF 00    cmp word ptr ss:[ebp-22C],0
00405E05   0F84 87000000          je mtx777.405E92
00405E0B   FF75 08                push dword ptr ss:[ebp+8]             [ebp+8]:L"\\\\?\\C:\\Program Files\\Far Manager\\Plugins\\ArcLite"
00405E0E   E8 B3320000            call mtx777.4090C6
00405E13   8BD8                   mov ebx,eax                           eax:L"7zS2.sfx"
00405E15   59                     pop ecx                               ecx:L"\\\\?\\C:\\Program Files\\Far Manager\\Plugins\\ArcLite"
00405E16   8D85 D0FDFFFF          lea eax,dword ptr ss:[ebp-230]
00405E1C   50                     push eax                              eax:L"7zS2.sfx"
00405E1D   E8 A4320000            call mtx777.4090C6
00405E22   8D4403 01              lea eax,dword ptr ds:[ebx+eax+1]      eax:L"7zS2.sfx"
00405E26   59                     pop ecx                               ecx:L"\\\\?\\C:\\Program Files\\Far Manager\\Plugins\\ArcLite"
00405E27   3945 14                cmp dword ptr ss:[ebp+14],eax
00405E2A   72 66                  jb mtx777.405E92
00405E2C   FF75 08                push dword ptr ss:[ebp+8]             [ebp+8]:L"\\\\?\\C:\\Program Files\\Far Manager\\Plugins\\ArcLite"
00405E2F   56                     push esi                              esi:L"\\\\?\\C:\\Program Files\\Far Manager\\Plugins\\ArcLite\\7zCon.sfx"
00405E30   E8 48340000            call mtx777.40927D
00405E35   68 08A24000            push mtx777.40A208
00405E3A   56                     push esi                              esi:L"\\\\?\\C:\\Program Files\\Far Manager\\Plugins\\ArcLite\\7zCon.sfx"
```

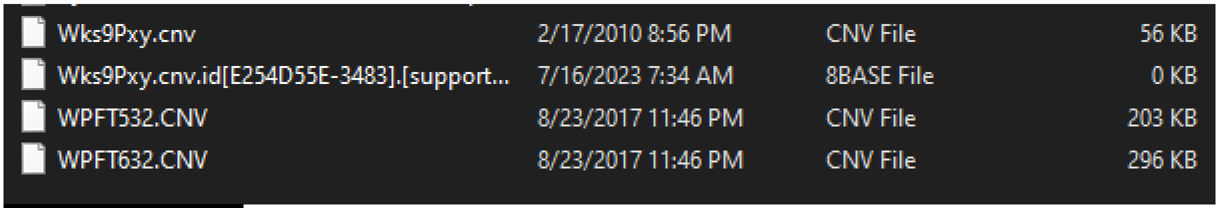
The encryption thread opens the file, gets its attributes, and reads its context.



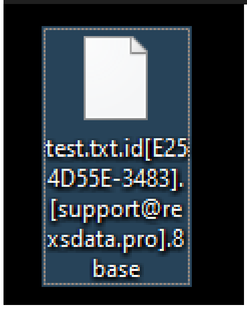
```
00408813   FF75 10         push dword ptr ss:[ebp+10]                   [ebp+10]:L"\
00408816   FF15 90A04000   call dword ptr ds:[<&GetFileAttributesW>]
0040881C   83F8 FF         cmp eax,FFFFFFFF
0040881F   0F85 C4020000   jne mtx777.408AE9
00408825   33C0            xor eax,eax
00408827   50              push eax
00408828   50              push eax
00408829   6A 03           push 3
0040882B   50              push eax
0040882C   50              push eax
0040882D   68 000000C0     push C0000000
00408832   FF75 0C         push dword ptr ss:[ebp+C]                    [ebp+C]:L"\\
00408835   FF15 98A04000   call dword ptr ds:[<&CreateFileW>]
0040883B   8945 F8         mov dword ptr ss:[ebp-8],eax
0040883E   83F8 FF         cmp eax,FFFFFFFF
00408841   0F84 A2020000   je mtx777.408AE9
00408847   8365 EC 00      and dword ptr ss:[ebp-14],0
0040884B   8365 E8 00      and dword ptr ss:[ebp-18],0
0040884F   6A 02           push 2
00408851   8D4D E8         lea ecx,dword ptr ss:[ebp-18]
00408854   51              push ecx
00408855   FF75 EC         push dword ptr ss:[ebp-14]
00408858   FF75 E8         push dword ptr ss:[ebp-18]
0040885B   50              push eax
0040885C   FF15 5CA04000   call dword ptr ds:[<&SetFilePointerEx>]
00408862   85C0            test eax,eax
00408864   0F84 4F020000   je mtx777.408AB9
0040886A   8B45 E8         mov eax,dword ptr ss:[ebp-18]
0040886D   0B45 EC         or eax,dword ptr ss:[ebp-14]
00408870   0F84 43020000   je mtx777.408AB9
00408876   33C0            xor eax,eax
00408878   50              push eax
00408879   8945 E8         mov dword ptr ss:[ebp-18],eax
0040887C   8945 EC         mov dword ptr ss:[ebp-14],eax
0040887F   8D45 E8         lea eax,dword ptr ss:[ebp-18]
00408882   50              push eax
00408883   FF75 EC         push dword ptr ss:[ebp-14]
00408886   FF75 E8         push dword ptr ss:[ebp-18]
00408889   FF75 F8         push dword ptr ss:[ebp-8]
0040888C   FF15 5CA04000   call dword ptr ds:[<&SetFilePointerEx>]
00408892   85C0            test eax,eax
00408894   0F84 1F020000   je mtx777.408AB9
0040889A   6A 00           push 0
```

Before starting encryption, 8Base creates a new file with a new extension:

<Original file name and extension>.id[<Unique victim ID>].[<Threat actors email>].8base

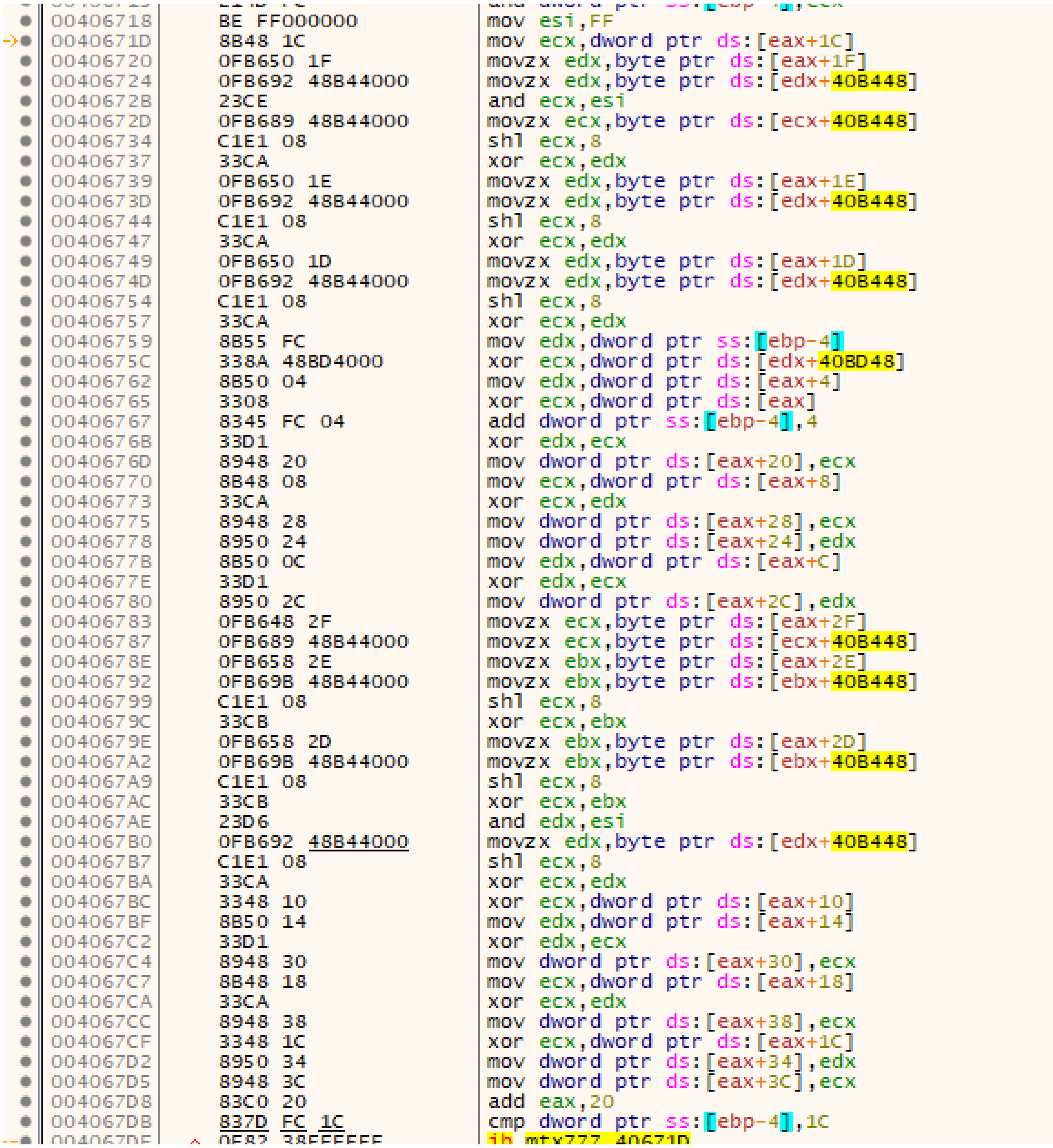


```
Wks9Pxy.cnv                                   2/17/2010 8:56 PM     CNV File      56 KB
Wks9Pxy.cnv.id[E254D55E-3483].[support...    7/16/2023 7:34 AM     8BASE File     0 KB
WPFT532.CNV                                   8/23/2017 11:46 PM    CNV File     203 KB
WPFT632.CNV                                   8/23/2017 11:46 PM    CNV File     296 KB
```



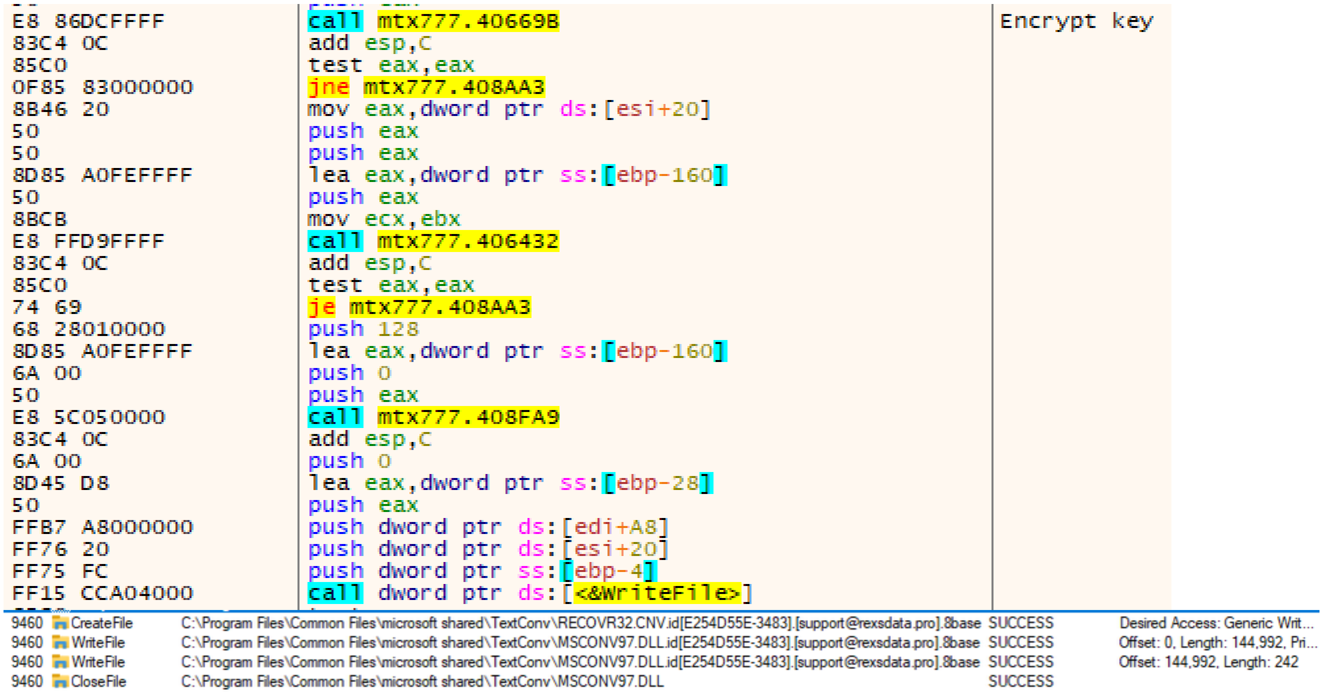
test.txt.id[E254D55E-3483].[support@rexsdata.pro].8base

Next, it transfers data to the encryption function, which uses the AES-256 algorithm in CBC mode. The IV keys are generated randomly during execution and will later be written to the encrypted file. To encrypt the AES key, it uses the RSA algorithm, making this encryption pretty strong. The encryption algorithms are hardcoded and don't use any crypto imports.



```
00406718    BE FF000000        mov esi,FF
0040671D    8B48 1C            mov ecx,dword ptr ds:[eax+1C]
00406720    0FB650 1F          movzx edx,byte ptr ds:[eax+1F]
00406724    0FB692 48B44000    movzx edx,byte ptr ds:[edx+40B448]
0040672B    23CE               and ecx,esi
0040672D    0FB689 48B44000    movzx ecx,byte ptr ds:[ecx+40B448]
00406734    C1E1 08            shl ecx,8
00406737    33CA               xor ecx,edx
00406739    0FB650 1E          movzx edx,byte ptr ds:[eax+1E]
0040673D    0FB692 48B44000    movzx edx,byte ptr ds:[edx+40B448]
00406744    C1E1 08            shl ecx,8
00406747    33CA               xor ecx,edx
00406749    0FB650 1D          movzx edx,byte ptr ds:[eax+1D]
0040674D    0FB692 48B44000    movzx edx,byte ptr ds:[edx+40B448]
00406754    C1E1 08            shl ecx,8
00406757    33CA               xor ecx,edx
00406759    8B55 FC            mov edx,dword ptr ss:[ebp-4]
0040675C    338A 48BD4000      xor ecx,dword ptr ds:[edx+40BD48]
00406762    8B50 04            mov edx,dword ptr ds:[eax+4]
00406765    3308               xor ecx,dword ptr ds:[eax]
00406767    8345 FC 04         add dword ptr ss:[ebp-4],4
0040676B    33D1               xor edx,ecx
0040676D    8948 20            mov dword ptr ds:[eax+20],ecx
00406770    8B48 08            mov ecx,dword ptr ds:[eax+8]
00406773    33CA               xor ecx,edx
00406775    8948 28            mov dword ptr ds:[eax+28],ecx
00406778    8950 24            mov dword ptr ds:[eax+24],edx
0040677B    8B50 0C            mov edx,dword ptr ds:[eax+C]
0040677E    33D1               xor edx,ecx
00406780    8950 2C            mov dword ptr ds:[eax+2C],edx
00406783    0FB648 2F          movzx ecx,byte ptr ds:[eax+2F]
00406787    0FB689 48B44000    movzx ecx,byte ptr ds:[ecx+40B448]
0040678E    0FB658 2E          movzx ebx,byte ptr ds:[eax+2E]
00406792    0FB69B 48B44000    movzx ebx,byte ptr ds:[ebx+40B448]
00406799    C1E1 08            shl ecx,8
0040679C    33CB               xor ecx,ebx
0040679E    0FB658 2D          movzx ebx,byte ptr ds:[eax+2D]
004067A2    0FB69B 48B44000    movzx ebx,byte ptr ds:[ebx+40B448]
004067A9    C1E1 08            shl ecx,8
004067AC    33CB               xor ecx,ebx
004067AE    23D6               and edx,esi
004067B0    0FB692 48B44000    movzx edx,byte ptr ds:[edx+40B448]
004067B7    C1E1 08            shl ecx,8
004067BA    33CA               xor ecx,edx
004067BC    3348 10            xor ecx,dword ptr ds:[eax+10]
004067BF    8B50 14            mov edx,dword ptr ds:[eax+14]
004067C2    33D1               xor edx,ecx
004067C4    8948 30            mov dword ptr ds:[eax+30],ecx
004067C7    8B48 18            mov ecx,dword ptr ds:[eax+18]
004067CA    33CA               xor ecx,edx
004067CC    8948 38            mov dword ptr ds:[eax+38],ecx
004067CF    3348 1C            xor ecx,dword ptr ds:[eax+1C]
004067D2    8950 34            mov dword ptr ds:[eax+34],edx
004067D5    8948 3C            mov dword ptr ds:[eax+3C],ecx
004067D8    83C0 20            add eax,20
004067DB    837D FC 1C         cmp dword ptr ss:[ebp-4],1C
004067DF    0F82 38FFFFFF      jb mtx777.40671D
```

After encrypted data is written, 8Base takes one further step — it encrypts the AES key and writes it to the end of the file with the IV key.

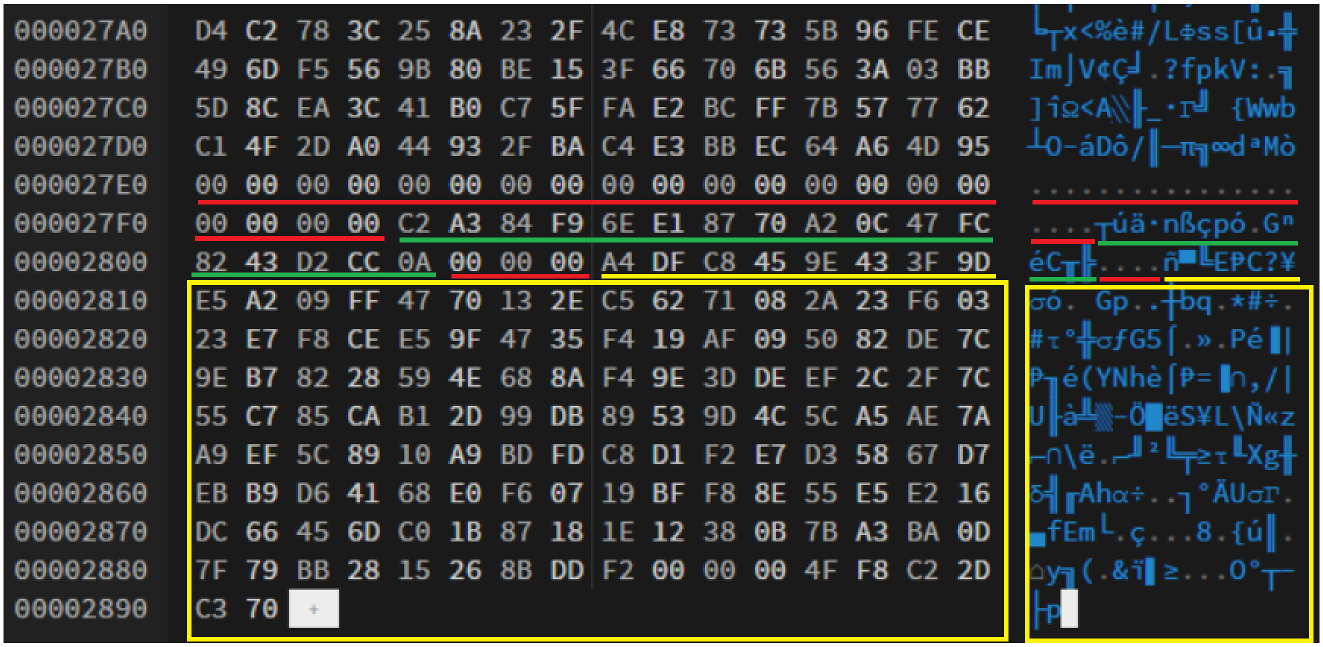


```
E8 86DCFFFF        call mtx777.40669B                       Encrypt key
83C4 0C            add esp,C
85C0               test eax,eax
0F85 83000000      jne mtx777.408AA3
8B46 20            mov eax,dword ptr ds:[esi+20]
50                 push eax
50                 push eax
8D85 A0FEFFFF      lea eax,dword ptr ss:[ebp-160]
50                 push eax
8BCB               mov ecx,ebx
E8 FFD9FFFF        call mtx777.406432
83C4 0C            add esp,C
85C0               test eax,eax
74 69              je mtx777.408AA3
68 28010000        push 128
8D85 A0FEFFFF      lea eax,dword ptr ss:[ebp-160]
6A 00              push 0
50                 push eax
E8 5C050000        call mtx777.408FA9
83C4 0C            add esp,C
6A 00              push 0
8D45 D8            lea eax,dword ptr ss:[ebp-28]
50                 push eax
FFB7 A8000000      push dword ptr ds:[edi+A8]
FF76 20            push dword ptr ds:[esi+20]
FF75 FC            push dword ptr ss:[ebp-4]
FF15 CCA04000      call dword ptr ds:[<&WriteFile>]
```

```
9460 CreateFile  C:\Program Files\Common Files\microsoft shared\TextConv\RECOVR32.CNV.id[E254D55E-3483].[support@rexsdata.pro].8base  SUCCESS  Desired Access: Generic Writ...
9460 WriteFile   C:\Program Files\Common Files\microsoft shared\TextConv\MSCONV97.DLL.id[E254D55E-3483].[support@rexsdata.pro].8base  SUCCESS  Offset: 0, Length: 144,992, Pri...
9460 WriteFile   C:\Program Files\Common Files\microsoft shared\TextConv\MSCONV97.DLL.id[E254D55E-3483].[support@rexsdata.pro].8base  SUCCESS  Offset: 144,992, Length: 242
9460 CloseFile   C:\Program Files\Common Files\microsoft shared\TextConv\MSCONV97.DLL                                                SUCCESS
```

With the encryption process completed, we can analyze the file structure.

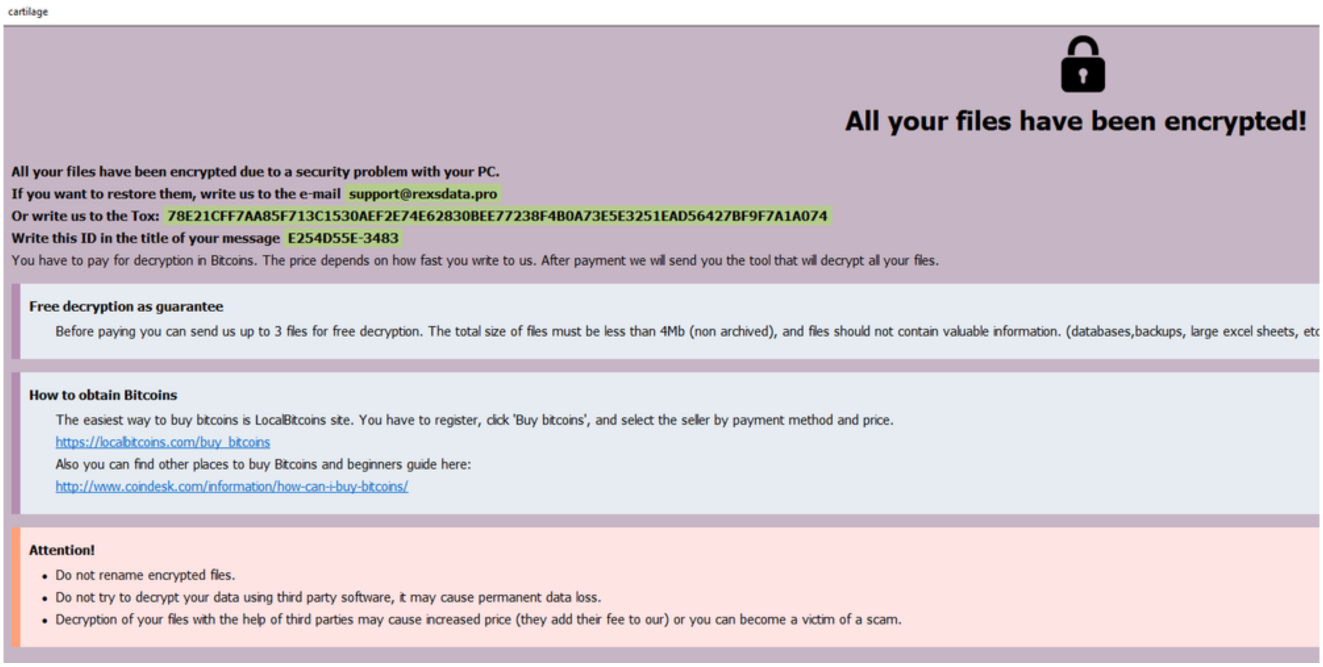
The first written data in the file is encrypted data. Next, there is a block of data, which is typical for Phobos family ransomware. First, there are 20 bytes of '00' (red line), which are used as a separator between encrypted data and this block. Then there are 16 bytes of IV key, which is different for each encrypted file (green line). Finally, the last block (yellow lines) is an encrypted AES key, which is similar for all files, encrypted in one session.



```
000027A0   D4 C2 78 3C 25 8A 23 2F 4C E8 73 73 5B 96 FE CE   └┬x<%è#/Lėss[û·╉
000027B0   49 6D F5 56 9B 80 BE 15 3F 66 70 6B 56 3A 03 BB   Im⌡V¢Ç╛.?fpkV:.╗
000027C0   5D 8C EA 3C 41 B0 C7 5F FA E2 BC FF 7B 57 77 62   ]îΩ<A░╟_·Γ╝ {Wwb
000027D0   C1 4F 2D A0 44 93 2F BA C4 E3 BB EC 64 A6 4D 95   ┴O-áDô/║─π╗∞dªMò
000027E0   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
000027F0   00 00 00 00 C2 A3 84 F9 6E E1 87 70 A2 0C 47 FC   ....┬úä·nßçpó.Gⁿ
00002800   82 43 D2 CC 0A 00 00 00 A4 DF C8 45 9E 43 3F 9D   éC╥╠....ñ▀╚E₧C?¥
00002810   E5 A2 09 FF 47 70 13 2E C5 62 71 08 2A 23 F6 03   σó. Gp..┼bq.*#÷.
00002820   23 E7 F8 CE E5 9F 47 35 F4 19 AF 09 50 82 DE 7C   #τ°╬σƒG5⌠.».Pé▐|
00002830   9E B7 82 28 59 4E 68 8A F4 9E 3D DE EF 2C 2F 7C   ₧╖é(YNhè⌠₧=▐∩,/|
00002840   55 C7 85 CA B1 2D 99 DB 89 53 9D 4C 5C A5 AE 7A   U╟à╩▒-Ö█ëS¥L\Ñ«z
00002850   A9 EF 5C 89 10 A9 BD FD C8 D1 F2 E7 D3 58 67 D7   ⌐∩\ë.⌐╜²╚╤≥τ╙Xg╫
00002860   EB B9 D6 41 68 E0 F6 07 19 BF F8 8E 55 E5 E2 16   δ╣╓Ahα÷..┐°ÄUσΓ.
00002870   DC 66 45 6D C0 1B 87 18 1E 12 38 0B 7B A3 BA 0D   ▄fEm└.ç...8.{ú║.
00002880   7F 79 BB 28 15 26 8B DD F2 00 00 00 4F F8 C2 2D   ⌂y╗(.&ï▌≥...O°┬-
00002890   C3 70                                             ├p
```

Ransom note

The ransom note files 'info.hta' and 'info.txt' are dropped after the completed encryption process in 'C:\' and 'C:\User\User\Desktop.'



All your files have been encrypted!

All your files have been encrypted due to a security problem with your PC.
If you want to restore them, write us to the e-mail support@rexsdata.pro
Or write us to the Tox: 78E21CFF7AA85F713C1530AEF2E74E62830BEE77238F4B0A73E5E3251EAD56427BF9F7A1A074
Write this ID in the title of your message E254D55E-3483
You have to pay for decryption in Bitcoins. The price depends on how fast you write to us. After payment we will send you the tool that will decrypt all your files.

Free decryption as guarantee

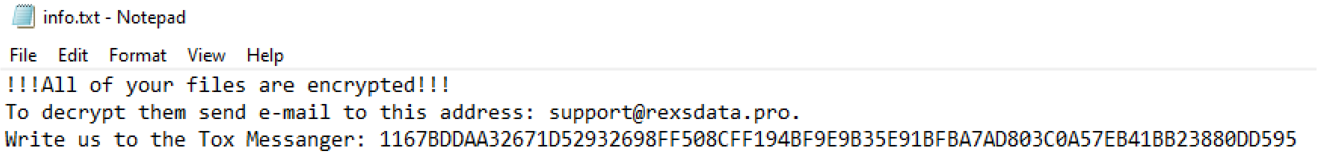
Before paying you can send us up to 3 files for free decryption. The total size of files must be less than 4Mb (non archived), and files should not contain valuable information. (databases,backups, large excel sheets, etc

How to obtain Bitcoins

The easiest way to buy bitcoins is LocalBitcoins site. You have to register, click 'Buy bitcoins', and select the seller by payment method and price.
https://localbitcoins.com/buy_bitcoins
Also you can find other places to buy Bitcoins and beginners guide here:
http://www.coindesk.com/information/how-can-i-buy-bitcoins/

Attention!

- Do not rename encrypted files.
- Do not try to decrypt your data using third party software, it may cause permanent data loss.
- Decryption of your files with the help of third parties may cause increased price (they add their fee to our) or you can become a victim of a scam.



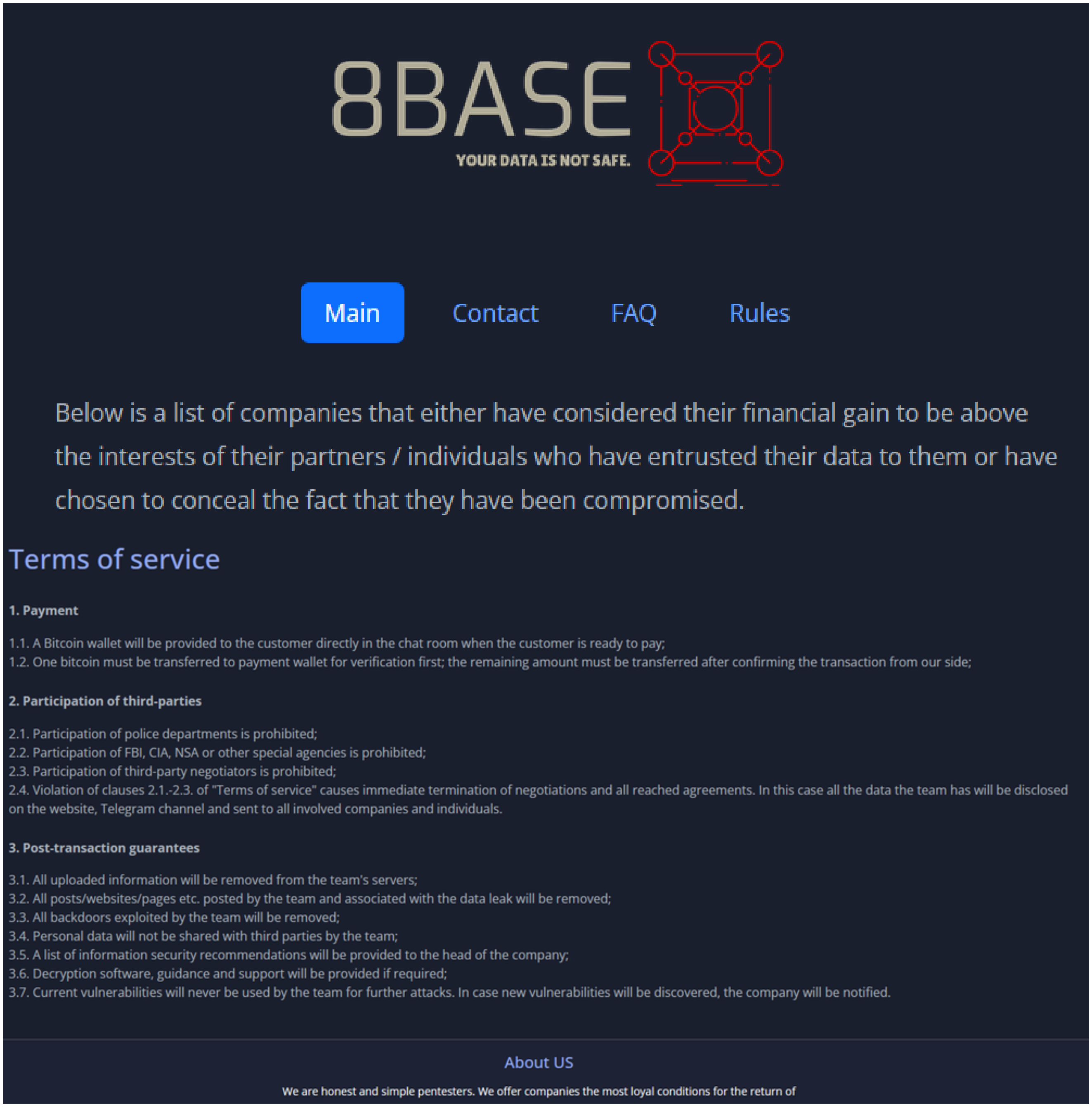
```
!!!All of your files are encrypted!!!
To decrypt them send e-mail to this address: support@rexsdata.pro.
Write us to the Tox Messanger: 1167BDDAA32671D52932698FF508CFF194BF9E9B35E91BFBA7AD803C0A57EB41BB23880DD595
```

Data leak site

While the ransom notes don’t have a link to the data leak site, the threat actor’s Twitter account does:

http://basemmnnqwxevlymli5bs36o5ynti55xojzvn246spahniugwkff2pad.onion/

This site contains the main page with the most recent victims of 8Base ransomware, a page for contacting the threat actors, a FAQ, and a “Rules” page.



Below is a list of companies that either have considered their financial gain to be above the interests of their partners / individuals who have entrusted their data to them or have chosen to conceal the fact that they have been compromised.

Terms of service

1. Payment

1.1. A Bitcoin wallet will be provided to the customer directly in the chat room when the customer is ready to pay;
1.2. One bitcoin must be transferred to payment wallet for verification first; the remaining amount must be transferred after confirming the transaction from our side;

2. Participation of third-parties

2.1. Participation of police departments is prohibited;
2.2. Participation of FBI, CIA, NSA or other special agencies is prohibited;
2.3. Participation of third-party negotiators is prohibited;
2.4. Violation of clauses 2.1.-2.3. of "Terms of service" causes immediate termination of negotiations and all reached agreements. In this case all the data the team has will be disclosed on the website, Telegram channel and sent to all involved companies and individuals.

3. Post-transaction guarantees

3.1. All uploaded information will be removed from the team's servers;
3.2. All posts/websites/pages etc. posted by the team and associated with the data leak will be removed;
3.3. All backdoors exploited by the team will be removed;
3.4. Personal data will not be shared with third parties by the team;
3.5. A list of information security recommendations will be provided to the head of the company;
3.6. Decryption software, guidance and support will be provided if required;
3.7. Current vulnerabilities will never be used by the team for further attacks. In case new vulnerabilities will be discovered, the company will be notified.

About US

We are honest and simple pentesters. We offer companies the most loyal conditions for the return of

The data leak site shares a lot of similarities to the RansomHouse group site, but it is still not clear whether these two groups are connected to each other or whether the 8Base threat actors have simply borrowed their site design.

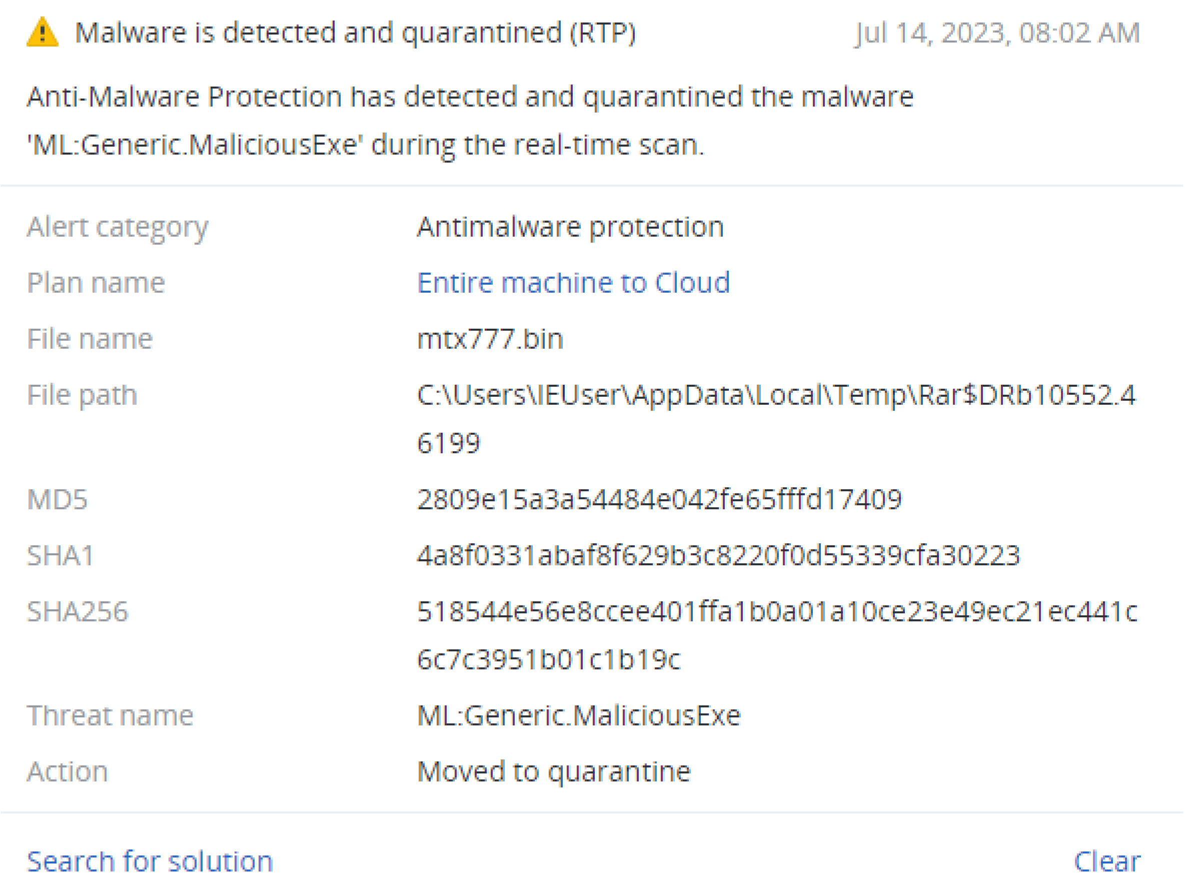
Conclusion

8Base ransomware successfully stayed unseen for almost a year before it was spotted with a large spike of targeted victims. On their Twitter account, the threat actors actively publish news, including info about recently breached victims.

The sample that was analyzed is a customized version of the Phobos ransomware, which encrypts users' files with AES-256-CBC algorithm, and utilizes SmokeLoader to bring malware to targeted systems.

The most interesting question here is about a potential connection between 8Base and another ransomware group (RansomHouse), as their data leak sites share a lot of similarities.

Detected by Acronis



⚠ Malware is detected and quarantined (RTP) Jul 14, 2023, 08:02 AM

Anti-Malware Protection has detected and quarantined the malware 'ML:Generic.MaliciousExe' during the real-time scan.

Alert category	Antimalware protection
Plan name	Entire machine to Cloud
File name	mtx777.bin
File path	C:\Users\IEUser\AppData\Local\Temp\Rar$DRb10552.46199
MD5	2809e15a3a54484e042fe65fffd17409
SHA1	4a8f0331abaf8f629b3c8220f0d55339cfa30223
SHA256	518544e56e8ccee401ffa1b0a01a10ce23e49ec21ec441c6c7c3951b01c1b19c
Threat name	ML:Generic.MaliciousExe
Action	Moved to quarantine

IoCs

Files

File name

SHA256

mtx777.exe

518544e56e8ccee401ffa1b0a01a10ce23e49ec21ec441c6c7c3951b01c1b19c

Network indicators

URL

Description

http://basemmnnqwxevlymli5bs36o5ynti55xojzvn246spahniugwkff2pad.onion/

Data leak site

https://twitter.com/8BASEHOME

Threat actor Twitter account

About Acronis

Acronis is a Swiss company, founded in Singapore. Celebrating two decades of innovation, Acronis has more than 2,000 employees in 45 locations. Acronis Cyber Protect solution is available in 26 languages in over 150 countries and is used by 18,000 service providers to protect over 750,000 businesses.