

SecurityScorecard Expands from Security Ratings to Supply Chain Detection and Response | Learn More



[Close](#) [X](#) [≡](#)



Resources ▾ A Deep Dive Into ALPHV/BlackCat Ransomware

RESEARCH

# A Deep Dive Into ALPHV/BlackCat Ransomware

Share



## Executive summary

ALPHV/BlackCat is the first widely known ransomware written in Rust. The malware must run with an access token consisting of a 32-byte value (-access-token parameter), and other parameters can be specified. The ransomware comes with an encrypted configuration that contains a list of services/processes to be stopped, a list of whitelisted directories/files/file extensions, and a list of stolen credentials from the victim environment. It deletes all Volume Shadow Copies, performs privilege escalation using the CMSTPLUA COM interface, and enables “remote to local” and “remote to remote” symbolic links on the victim’s machine.

The files are encrypted using the AES algorithm, with the AES key being encrypted using the RSA public key contained in the configuration. The extension of the encrypted files is changed to uhwuvzu by the malware.

## Analysis and findings

SHA256: 847fb7609f53ed334d5affbb07256c21cb5e6f68b1cc14004f5502d714d2a456

The malware can run with one of the following parameters:

### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

[Use necessary cookies only](#)

[Allow selection](#)

[Allow all cookies](#)

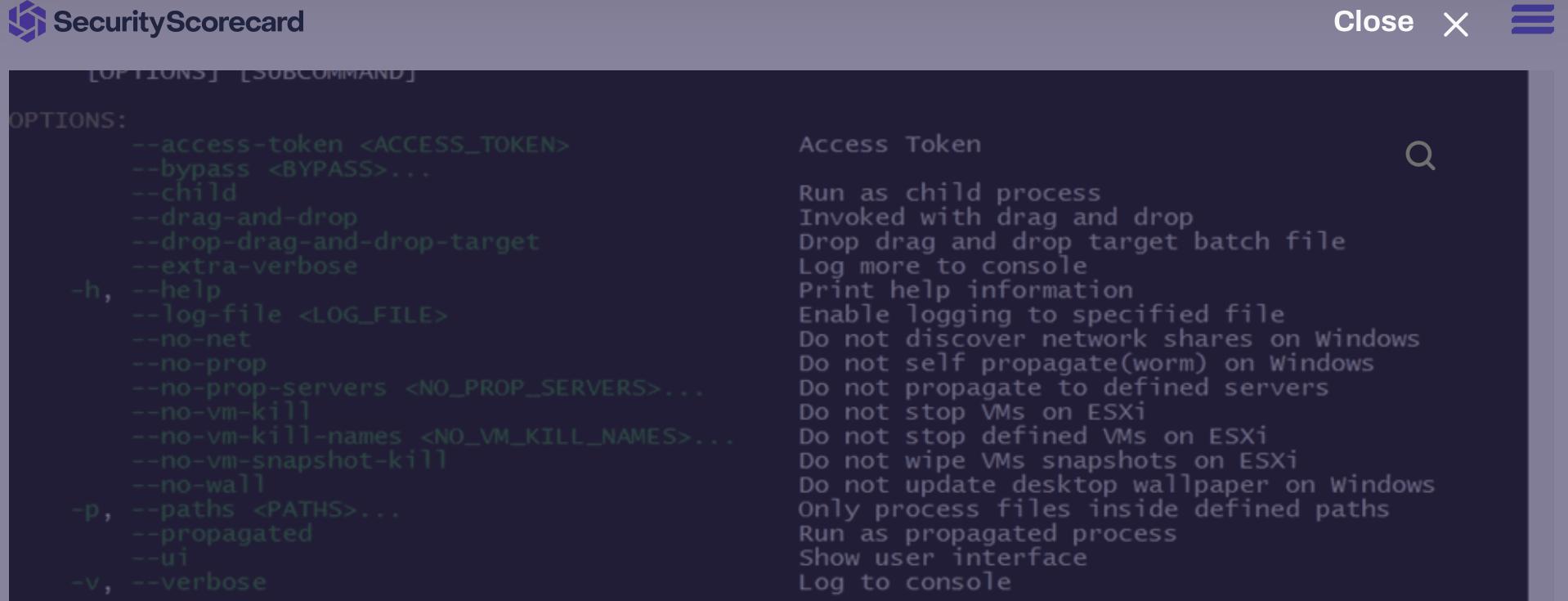


Figure 1

Whether the ransomware is running with no parameters or with an invalid access token, an error message is displayed:

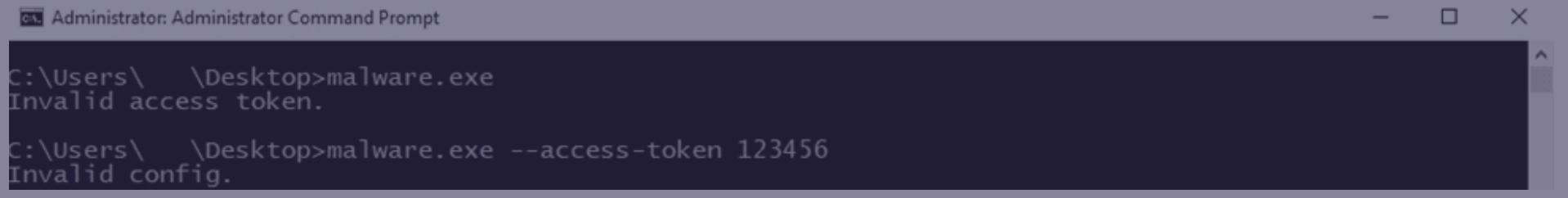


Figure 2

By performing the dynamic analysis, we've found that the access token must be a 32-byte value that is not unique.

The binary registers a new top-level exception handler via a function call to SetUnhandledExceptionFilter:

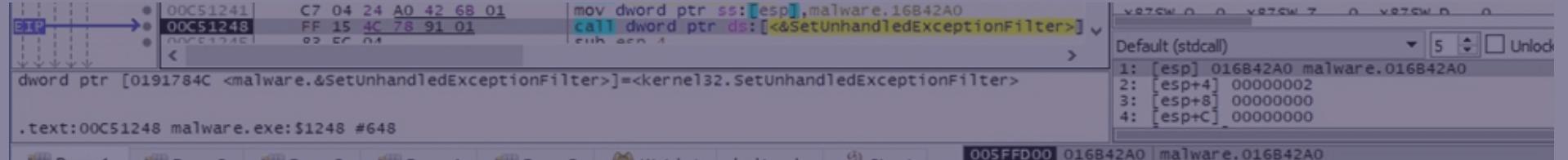
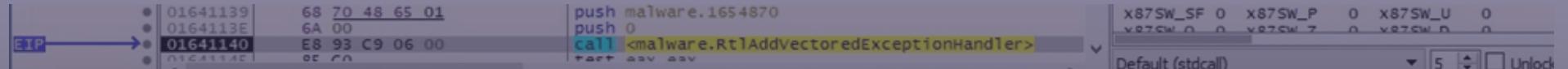


Figure 3

The AddVectoredExceptionHandler API is utilized to register a vectored exception handler:



#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

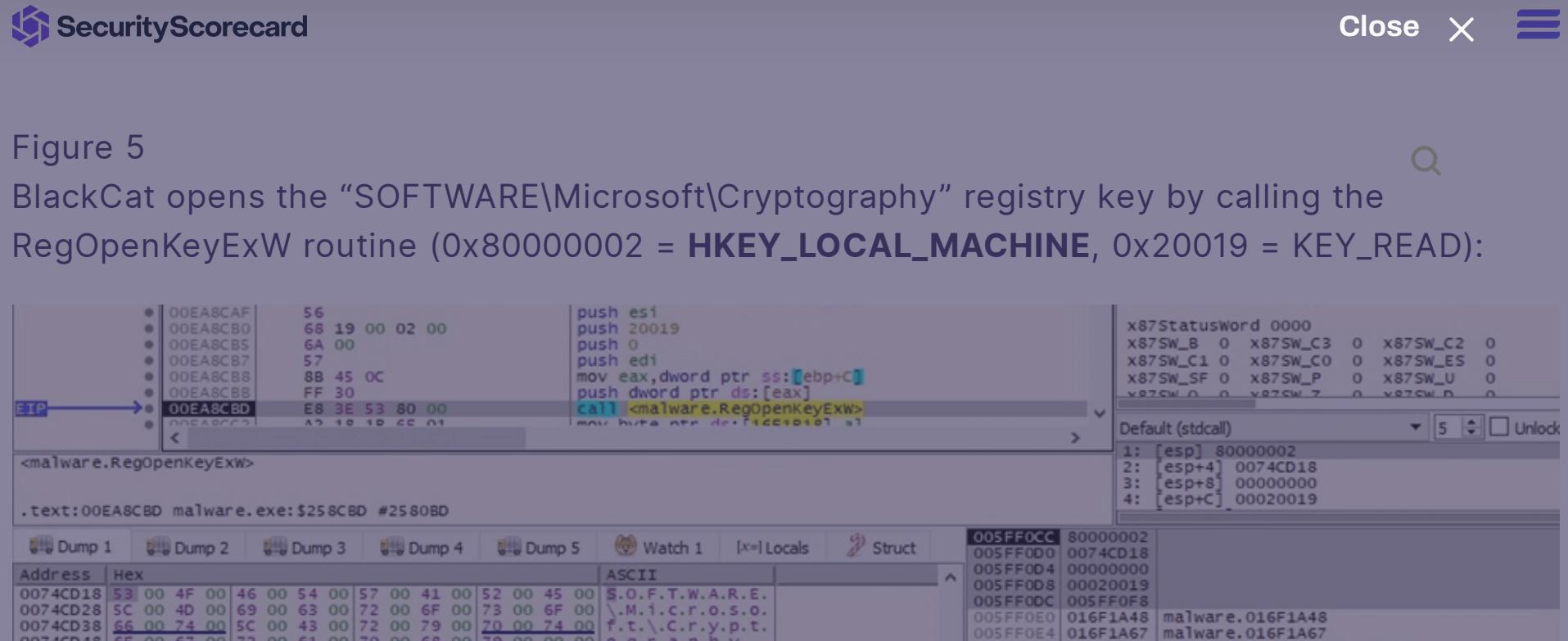
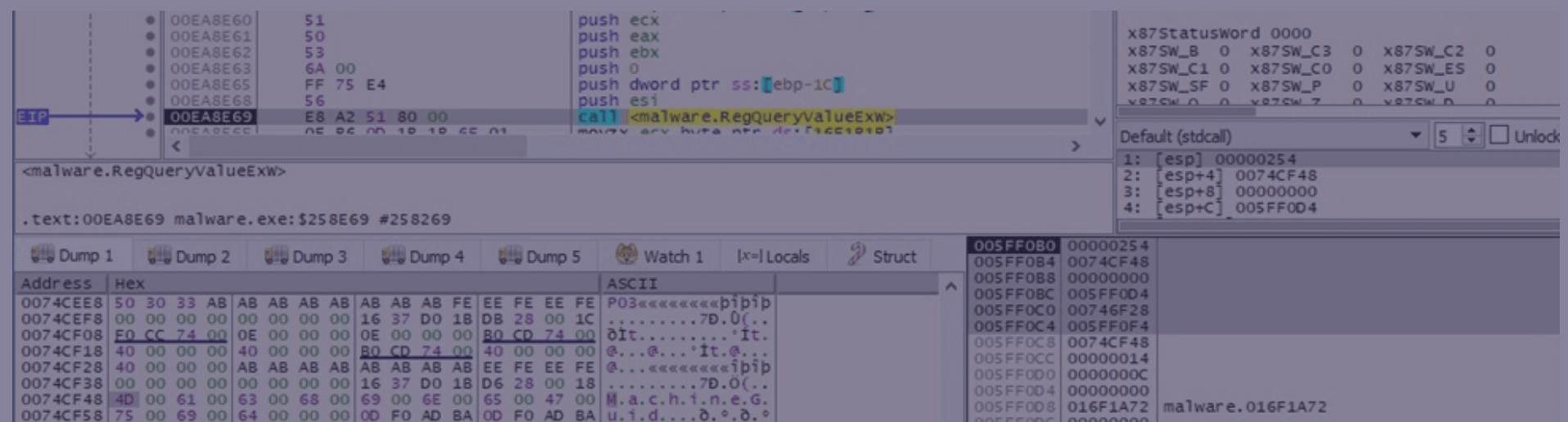


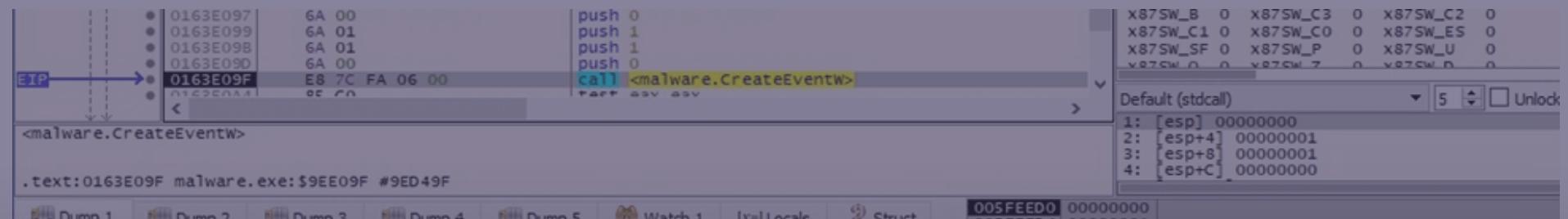
Figure 6

The binary extracts the MachineGUID value from the registry:



## Figure 7

The malicious process searches for cmd.exe in the current directory and then in the System32 directory via a function call to CreateFileW (0x7 = **FILE\_SHARE\_DELETE** | **FILE\_SHARE\_WRITE** | **FILE\_SHARE\_READ**, 0x3 = **OPEN\_EXISTING**, 0x2000000 = **FILE\_FLAG\_BACKUP\_SEMANTICS**):



This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services

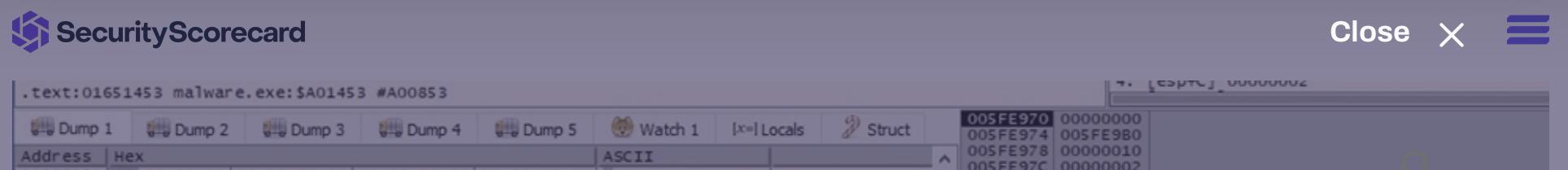


Figure 9

A named pipe whose name contains the current process ID and random bytes generated above is created using `CreateNamedPipeW` ( $0x40080001 = \text{FILE\_FLAG\_OVERLAPPED} | \text{FILE\_FLAG\_FIRST\_PIPE\_INSTANCE} | \text{PIPE\_ACCESS\_INBOUND}$ ,  $0x8 = \text{PIPE\_REJECT\_REMOTE\_CLIENTS}$ ):

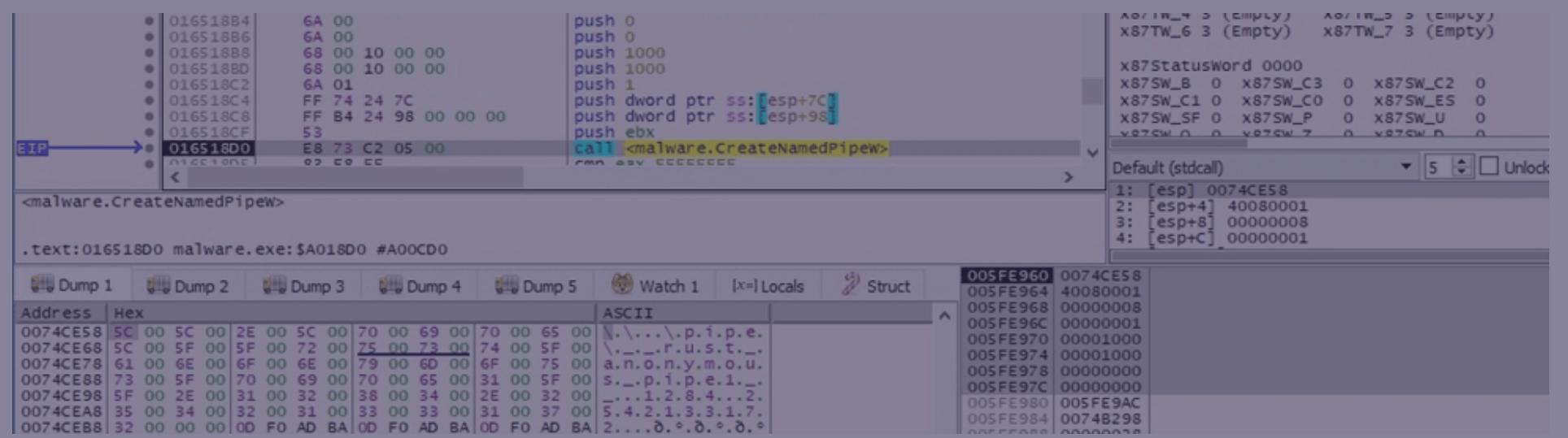


Figure 10

The process opens the named pipe for writing using the `CreateFileW` routine ( $0x40000000 = \text{GENERIC\_WRITE}$ ,  $0x3 = \text{OPEN\_EXISTING}$ ):

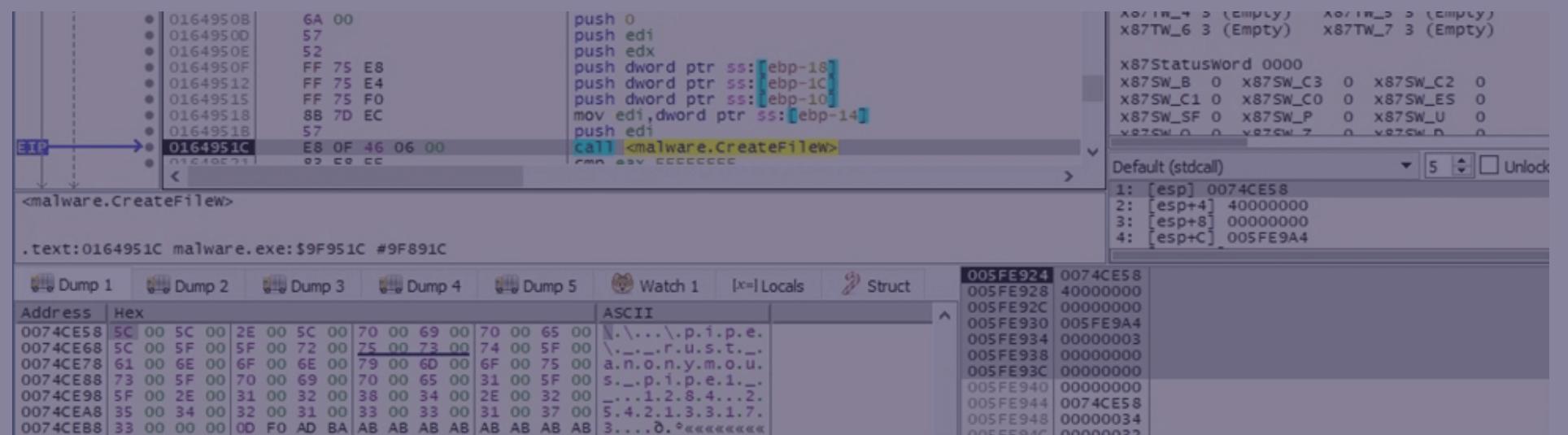


Figure 11

The ransomware creates a read and a write named pipe, respectively.

#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

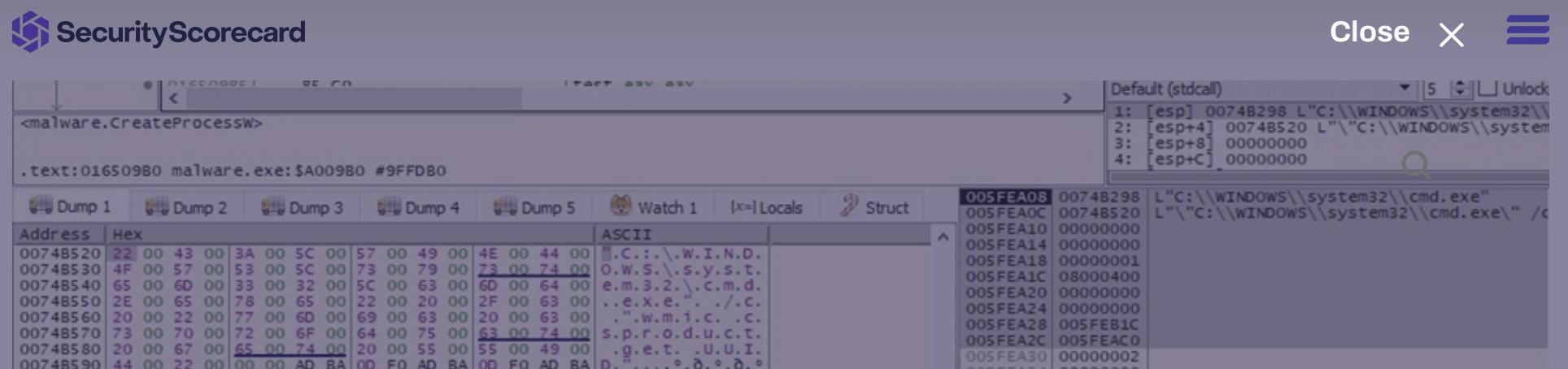


Figure 12

The CreateEventW API is utilized to create two unnamed event objects:

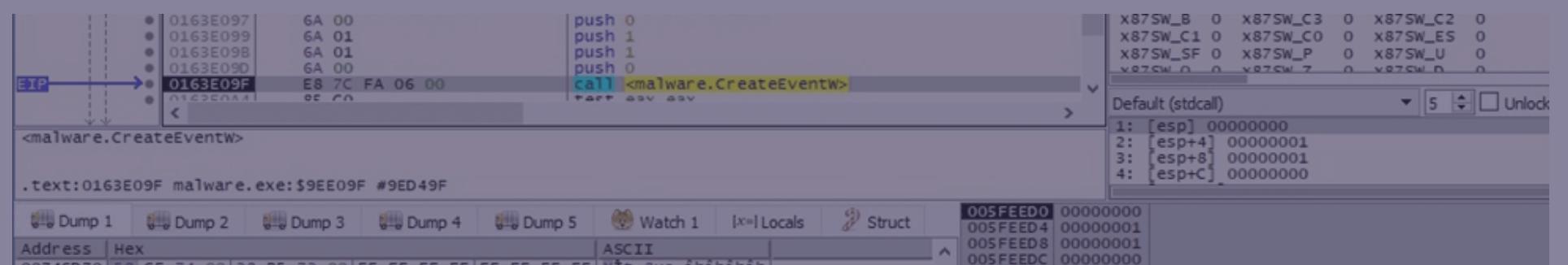


Figure 13

The binary waits until the event objects are in the signaled state by calling WaitForMultipleObjects:

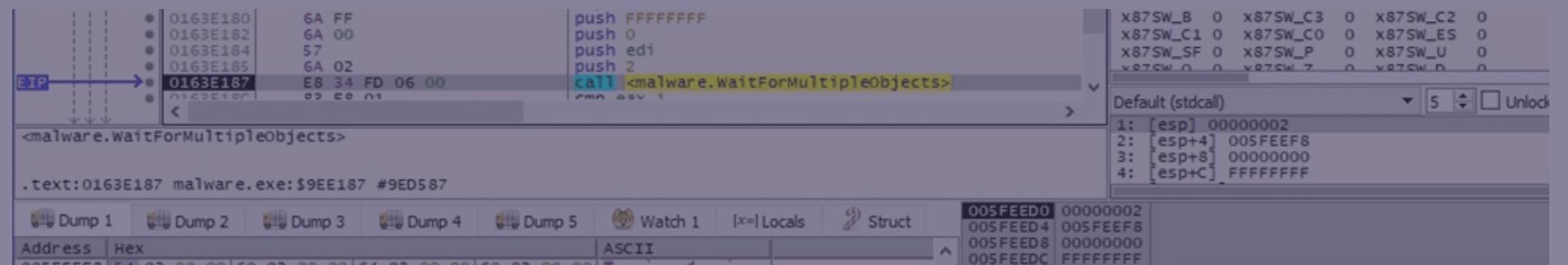
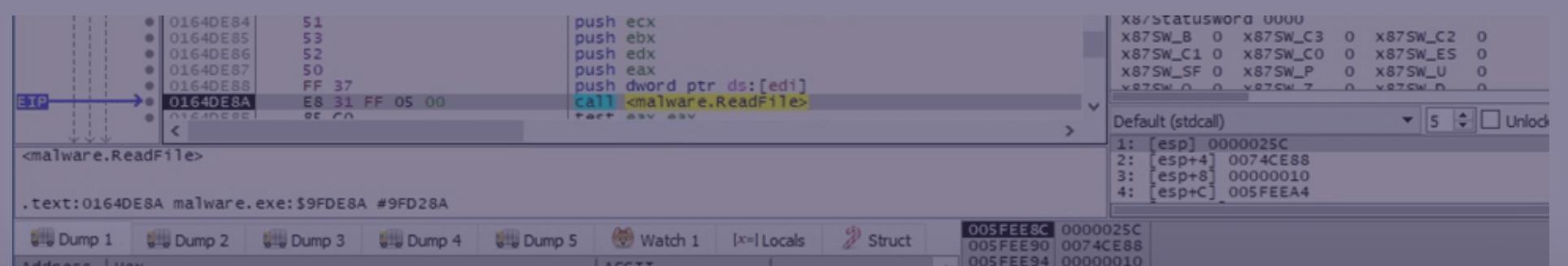


Figure 14

The output of the above process is read from the named pipe using the ReadFile routine:



#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

Figure 16

The content of the ransom note and the text that will appear on the Desktop Wallpaper are decrypted by the ransomware:

Address	Hex	ASCII
0074CF48	3E 3E 20 57 68 61 74 20 68 61 70 70 65 6E 65 64	>> What happened
0074CF58	3F 0A 0A 49 6D 70 6F 72 74 61 6E 74 20 66 69 6C	?..Important fil
0074CF68	65 73 20 6F 6E 20 79 6F 75 72 20 6E 65 74 77 6F	es on your netwo
0074CF78	72 68 20 77 61 73 20 45 4E 43 52 59 50 54 45 44	rk was ENCRYPTED
0074CF88	20 61 6E 64 20 6E 6F 77 20 74 68 65 79 20 68 61	and now they ha
0074CF98	76 65 20 22 75 68 77 75 76 7A 75 22 20 65 78 74	ve "uhwuvzu" ext
0074CFA8	65 6E 73 69 6F 6E 2E 0A 49 6E 20 6F 72 64 65 72	ension..In order
0074CFB8	20 74 6F 20 72 65 63 6F 76 65 72 20 79 6F 75 72	to recover your
0074FC8	20 66 69 6C 65 73 20 79 6F 75 20 6E 65 65 64 20	files you need
0074CFD8	74 6F 20 66 6F 6C 6C 6F 77 20 69 6E 73 74 72 75	to follow instru
0074CFE8	63 74 69 6F 6E 73 20 62 65 6C 6F 77 2E 0A 0A 3E	ctions below...>
0074CFF8	3E 20 53 65 6E 73 69 74 69 76 65 20 44 61 74 61	> Sensitive Data
0074D008	0A 0A 53 65 6E 73 69 74 69 76 65 20 64 61 74 61	..Sensitive data
0074D018	20 6F 6E 20 79 6F 75 72 20 6E 65 74 77 6F 72 6B	on your network
0074D028	20 77 61 73 20 44 4F 57 4E 4C 4F 41 44 45 44 2E	was DOWNLOADED.
0074D038	0A 49 66 20 79 6F 75 20 44 4F 4E 27 54 20 57 41	.If you DON'T WA
0074D048	4E 54 20 79 6F 75 72 20 73 65 6E 73 69 74 69 76	NT your sensitiv
0074D058	65 20 64 61 74 61 20 74 6F 20 62 65 20 50 55 42	e data to be PUB
0074D068	4C 49 53 48 45 44 20 79 6F 75 20 68 61 76 65 20	LISHED you have

Figure 17

Address	Hex	ASCII
0074AA98	49 6D 70 6F 72 74 61 6E 74 20 66 69 6C 65 73 20	Important files
0074AAA8	6F 6E 20 79 6F 75 72 20 6E 65 74 77 6F 72 6B 20	on your network
0074AA88	77 61 73 20 44 4F 57 4E 4C 4F 41 44 45 44 20 61	was DOWNLOADED a
0074AAC8	6E 64 20 45 4E 43 52 59 50 54 45 44 2E 0A 53 65	nd ENCRYPTED..Se
0074AAD8	65 20 22 52 45 43 4F 56 45 52 2D 75 68 77 75 76	e "RECOVER-uhwuv
0074AAE8	7A 75 2D 46 49 4C 45 53 2E 74 78 74 22 20 66 69	zu-FILES.txt" fi
0074AAF8	6C 65 20 74 6F 20 67 65 74 20 66 75 72 74 68 65	le to get furthe
0074AB08	72 20 69 6E 73 74 72 75 63 74 69 6F 6E 73 2E AB	r instructions.«

Figure 18

The malicious binary obtains information about the current system via a function call to GetSystemInfo:

#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

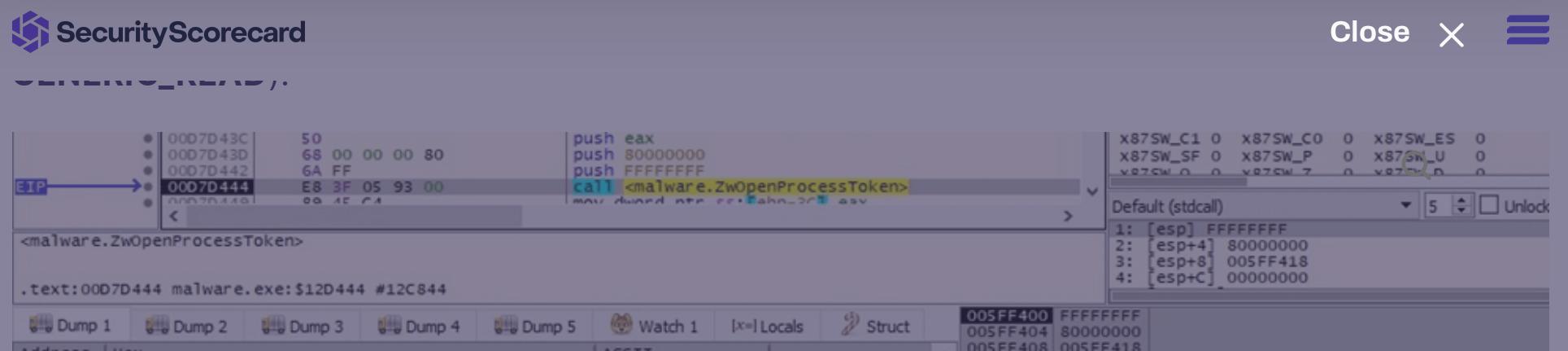


Figure 21

BlackCat extracts a TOKEN\_GROUPS structure containing the group accounts associated with the above token using the NtQueryInformationToken function (0x2 = **TokenGroups**):

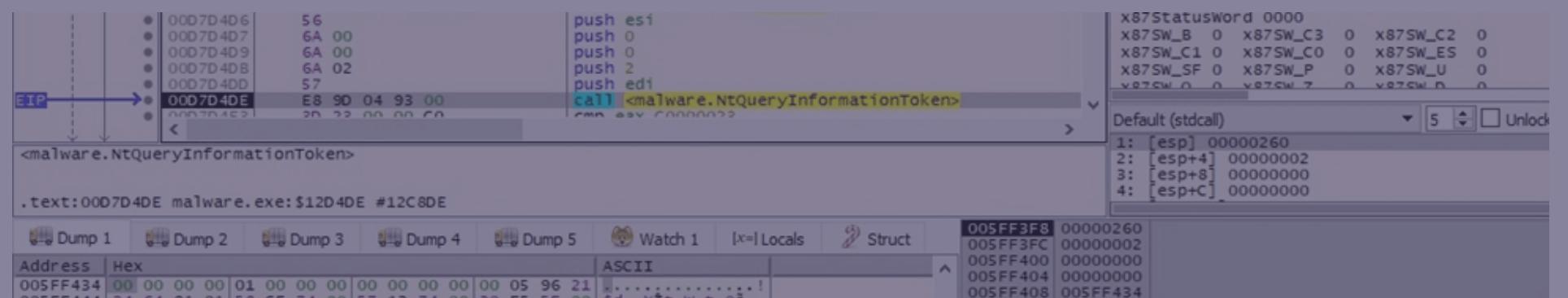


Figure 22

The OpenProcess API is utilized to open a local process object (0x438 = **PROCESS\_QUERY\_INFORMATION | PROCESS\_VM\_WRITE | PROCESS\_VM\_READ | PROCESS\_VM\_OPERATION**):

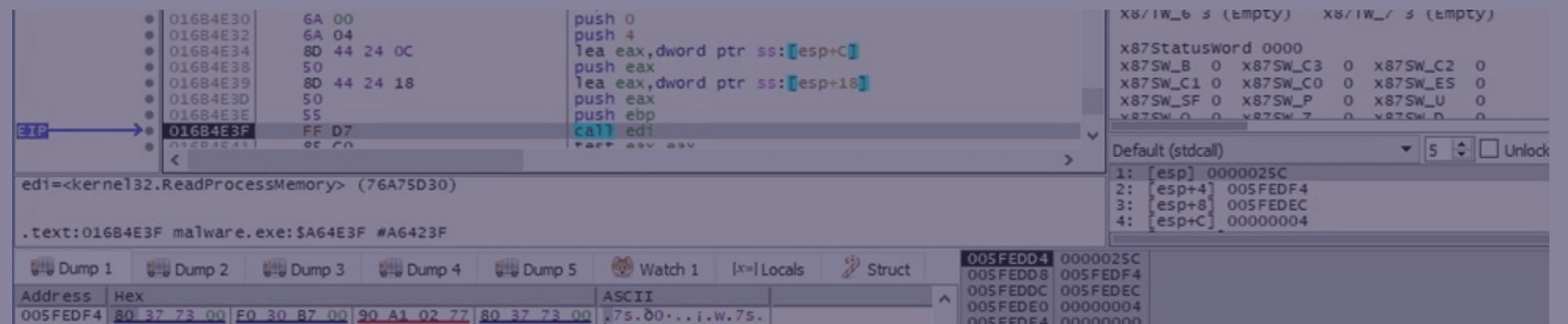


Figure 23

The malicious binary retrieves a pointer to a PEB structure using the ZwQueryInformationProcess routine (0x0 = **ProcessBasicInformation**):



#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

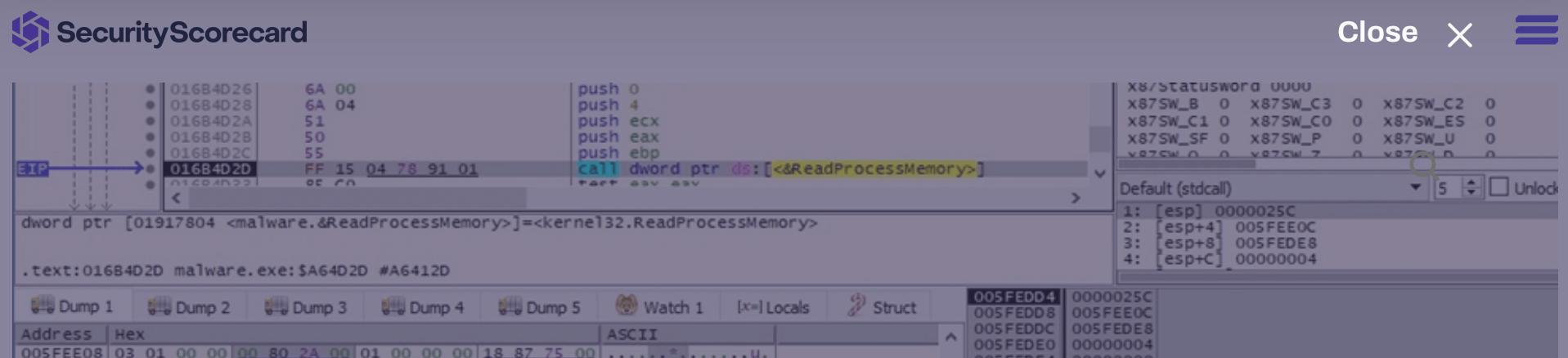


Figure 25

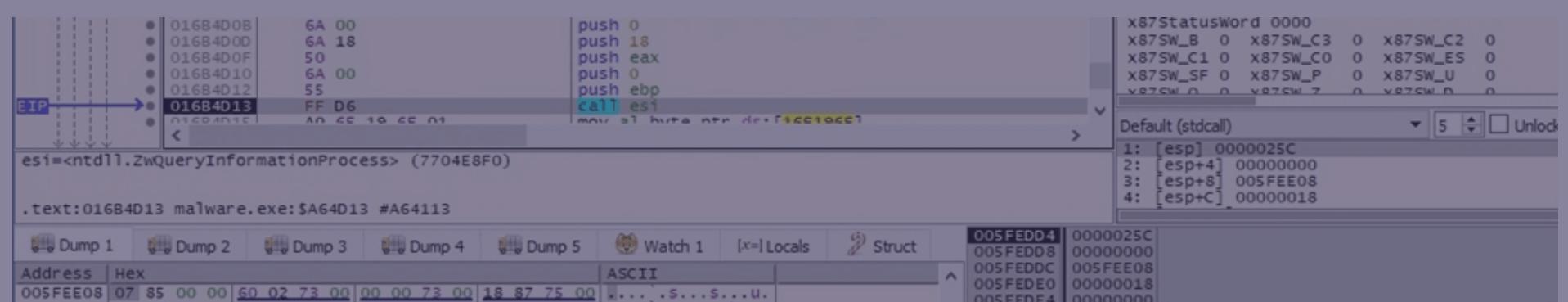


Figure 26

The path of the image file for the current process is retrieved using ReadProcessMemory:

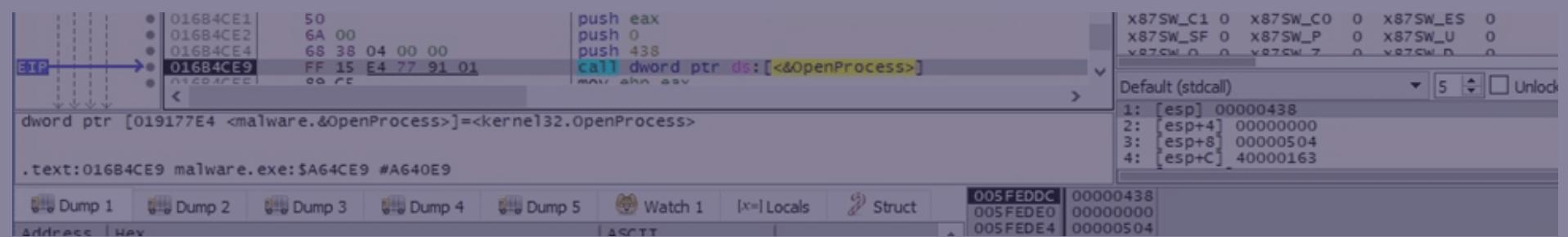
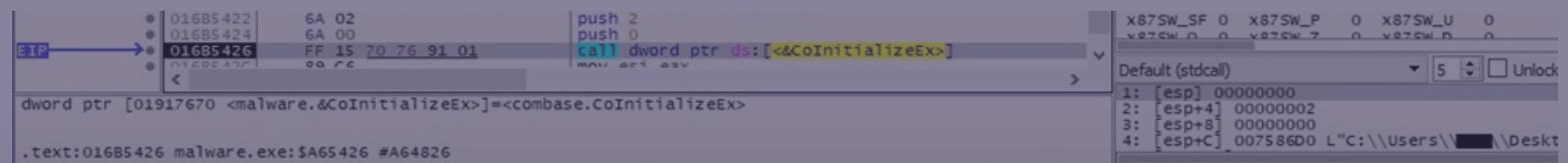


Figure 27

## Privilege escalation via UAC bypass using CMSTPLUA COM interface

The ransomware initializes the COM library for use by the current thread via a call to CoInitializeEx (0x2 = COINIT\_APARTMENTTHREADED):



### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

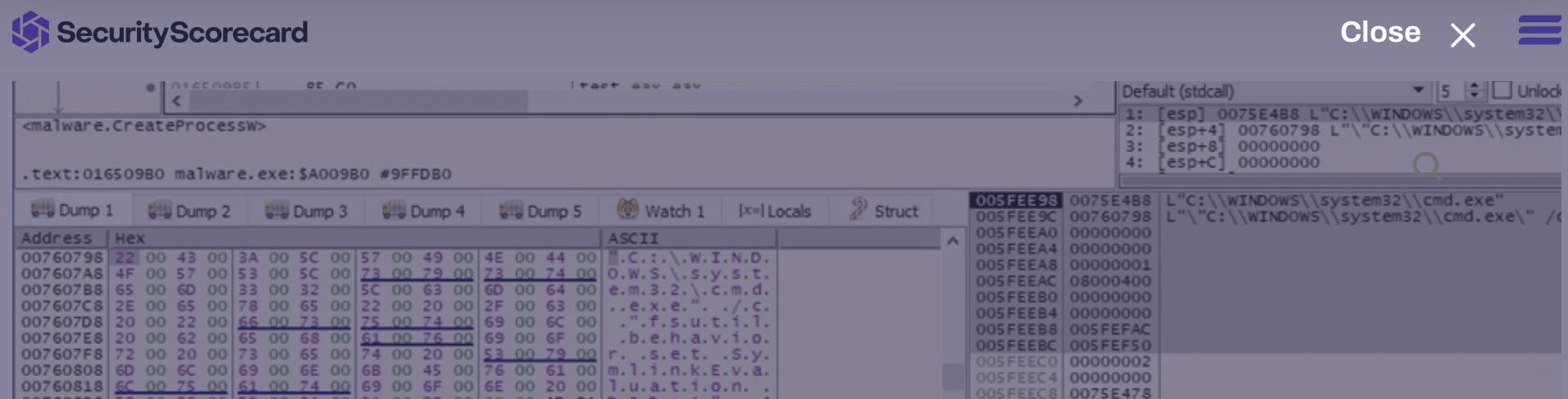


Figure 29

The initial executable is spawned with administrative privileges:

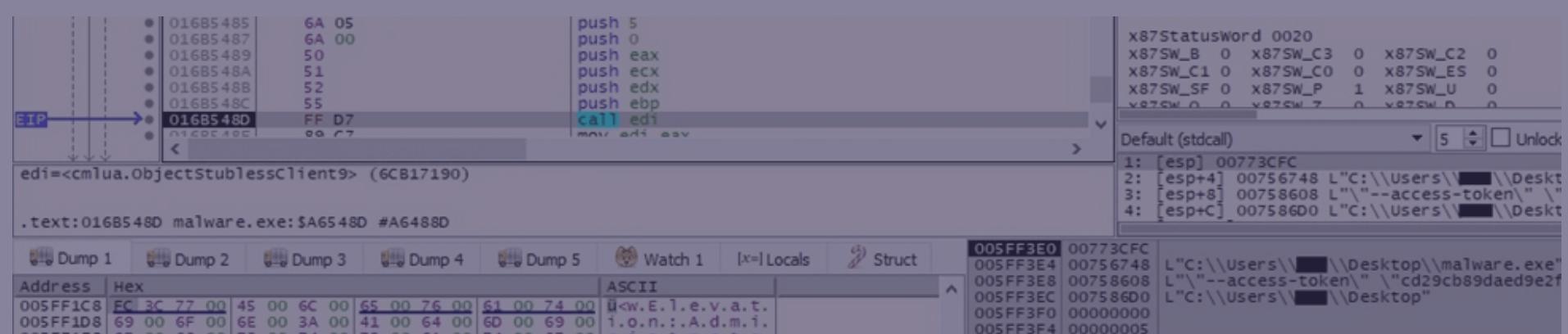


Figure 30

The `LookupPrivilegeValueW` routine is utilized to retrieve the locally unique identifier that represents the following privileges:

- `SeIncreaseQuotaPrivilege` `SeSecurityPrivilege` `SeTakeOwnershipPrivilege`
- `SeLoadDriverPrivilege` `SeSystemProfilePrivilege` `SeSystemtimePrivilege`
- `SeProfileSingleProcessPrivilege` `SeIncreaseBasePriorityPrivilege`
- `SeCreatePagefilePrivilege` `SeBackupPrivilege` `SeRestorePrivilege`
- `SeShutdownPrivilege` `SeDebugPrivilege` `SeSystemEnvironmentPrivilege`
- `SeChangeNotifyPrivilege` `SeRemoteShutdownPrivilege` `SeUndockPrivilege`
- `SeManageVolumePrivilege` `SeImpersonatePrivilege` `SeCreateGlobalPrivilege`
- `SeIncreaseWorkingSetPrivilege` `SeTimeZonePrivilege`
- `SeCreateSymbolicLinkPrivilege` `SeDelegateSessionUserImpersonatePrivilege`

#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

## SecurityScorecard

[Close](#) [X](#) [≡](#)

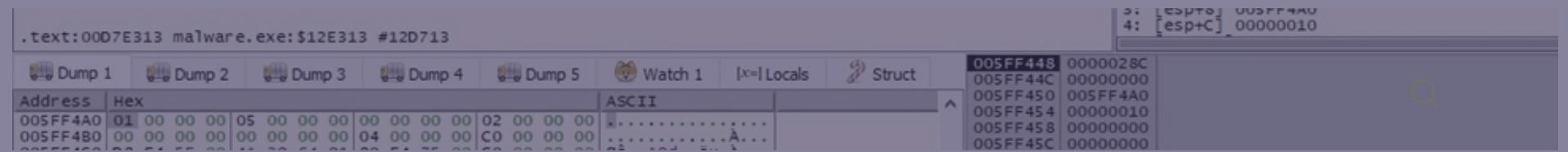


Figure 32

The binary creates the following processes that enable “remote to local” and “remote to remote” symbolic links on the local machine:

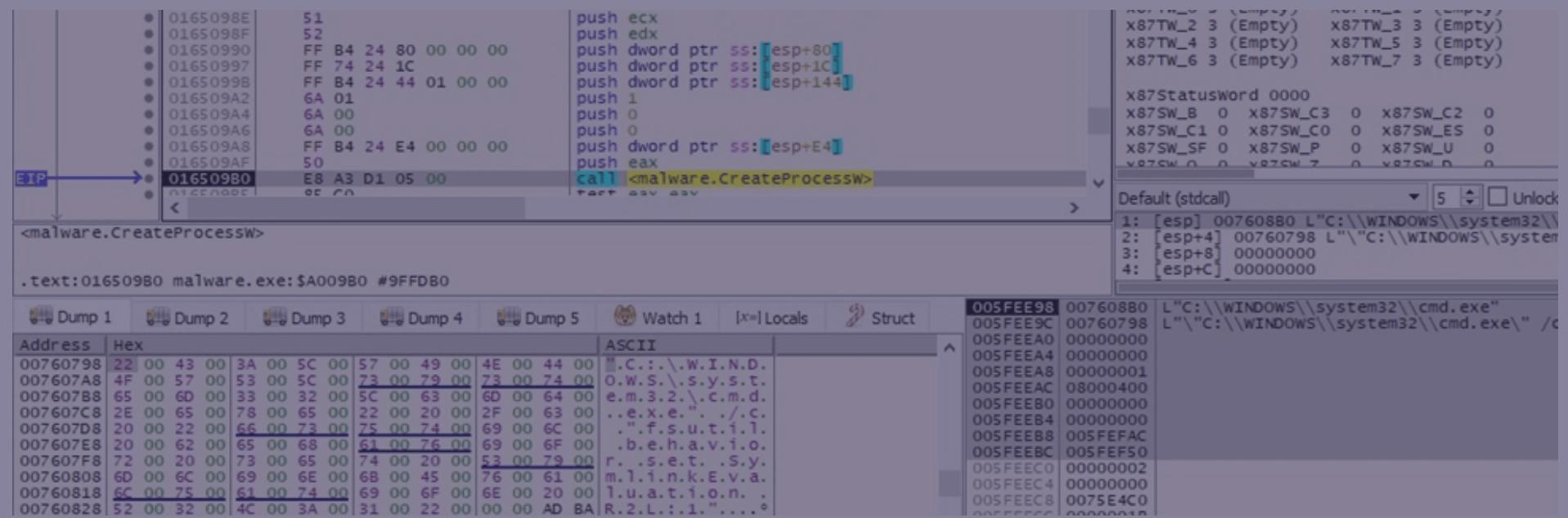


Figure 33

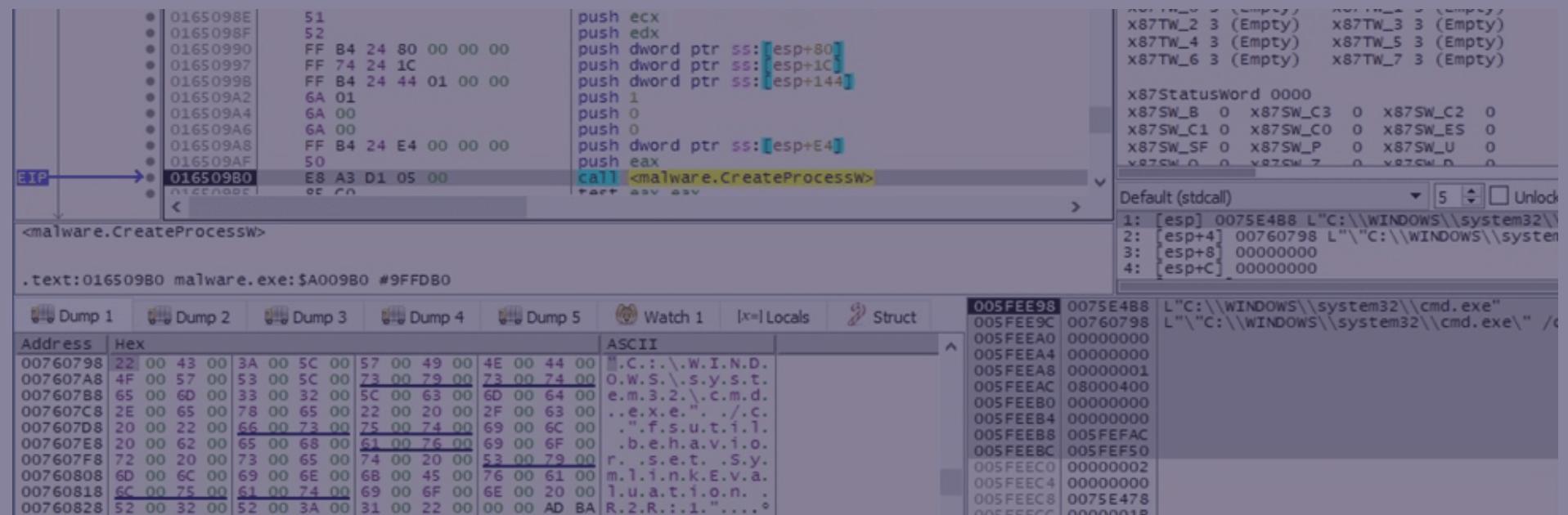


Figure 34

The malware tries to stop the Internet Information service (IIS) using IISReset.exe:



### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

The screenshot shows the Immunity Debugger interface. The assembly pane displays the following code snippet:

```

push ecx
push edx
push dword ptr ss:[esp+80]
push dword ptr ss:[esp+1C]
push dword ptr ss:[esp+144]
push 1
push 0
push 0
push dword ptr ss:[esp+E4]
push eax
call <malware.CreateProcessW>
ret esp

```

The memory dump pane shows the command being executed:

```

L"C:\\\\WINDOWS\\\\system32\\\\cmd.exe\" /c del /s /q *.*"

```

Figure 36

There is also a second process that is responsible for deleting all volume shadow copies with wmic:

The screenshot shows the Immunity Debugger interface. The assembly pane displays the same code snippet as Figure 36:

```

push ecx
push edx
push dword ptr ss:[esp+80]
push dword ptr ss:[esp+1C]
push dword ptr ss:[esp+144]
push 1
push 0
push 0
push dword ptr ss:[esp+E4]
push eax
call <malware.CreateProcessW>
ret esp

```

The memory dump pane shows the command being executed:

```

L"C:\\\\WINDOWS\\\\system32\\\\cmd.exe\" /c wmic shadow get

```

Figure 37

Interestingly, the malware runs the following command that is incomplete and returns an error:

The screenshot shows the Immunity Debugger interface. The assembly pane displays the same code snippet as previous figures:

```

push ecx
push edx
push dword ptr ss:[esp+80]
push dword ptr ss:[esp+1C]
push dword ptr ss:[esp+144]
push 1
push 0
push 0
push dword ptr ss:[esp+E4]
push eax
call <malware.CreateProcessW>
ret esp

```

The memory dump pane shows the command being executed:

```

L"C:\\\\WINDOWS\\\\system32\\\\cmd.exe\" /c l.e.t.e.\"....\\o

```

#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

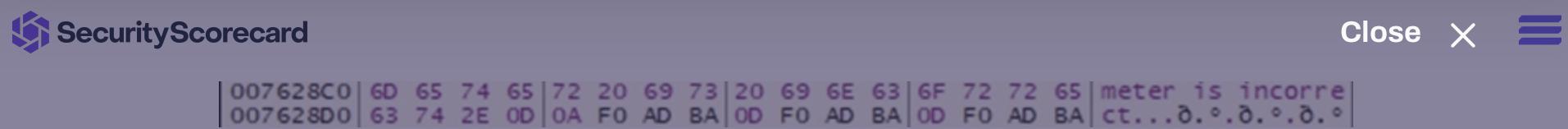


Figure 39

The binary disables Automatic Repair using the bcdedit tool:

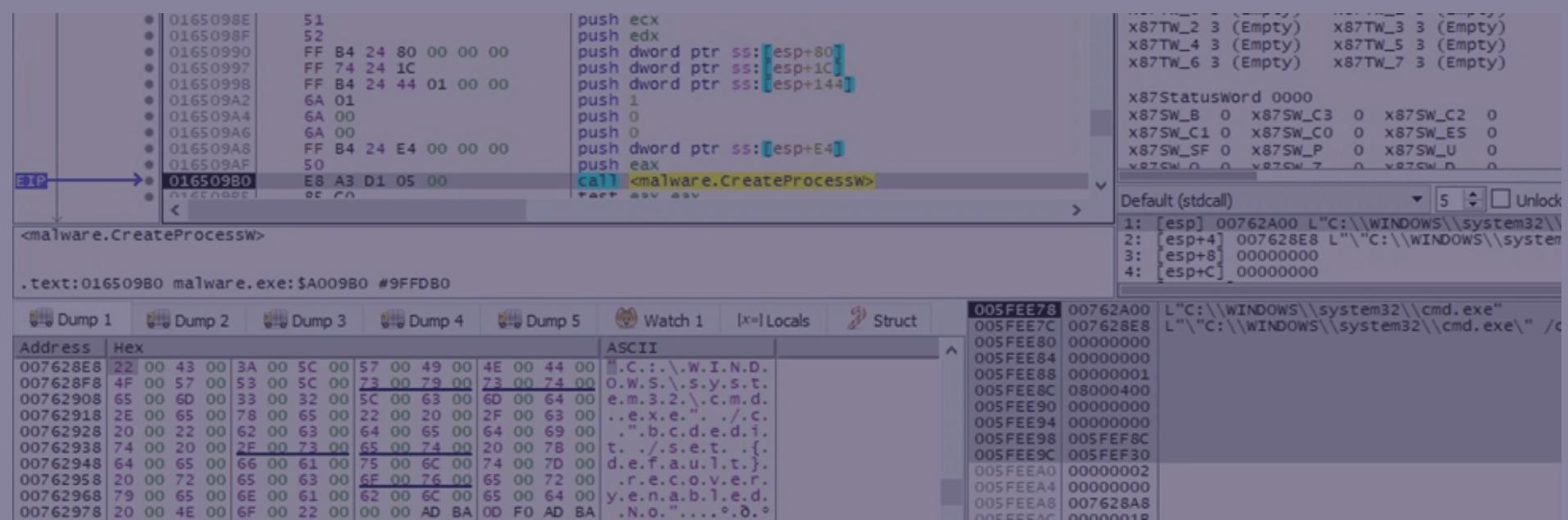


Figure 40

The ransomware tries to clear all event logs, however, the command is incorrect and returns an error, as highlighted below:

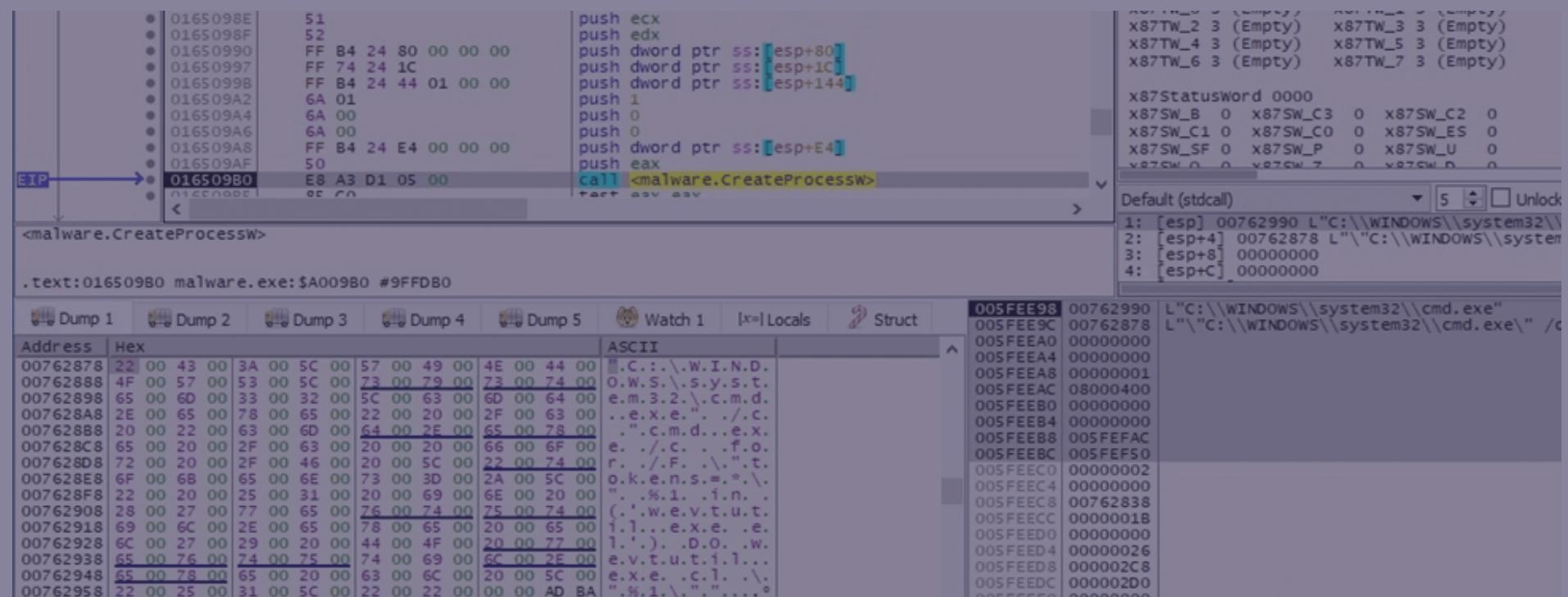


Figure 41

Address	Hex	ASCII
007628B8	15 37 D0 18 DB 28 00 18	.7D.0(..\\\"tokens
007628C8	3D 2A 5C 22 20 77 61 73	=*\" was unexpect
007628D8	74 65 64 20 61 74 20 74	ted at this time
007628E8	2E 0D 0A BA 0D F0 AD BA	...d...d...d...

#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.



Figure 43

The process obtains a list of active services using `EnumServicesStatusExW` (`0x30 = SERVICE_WIN32, 0x1 = SERVICE_ACTIVE`):

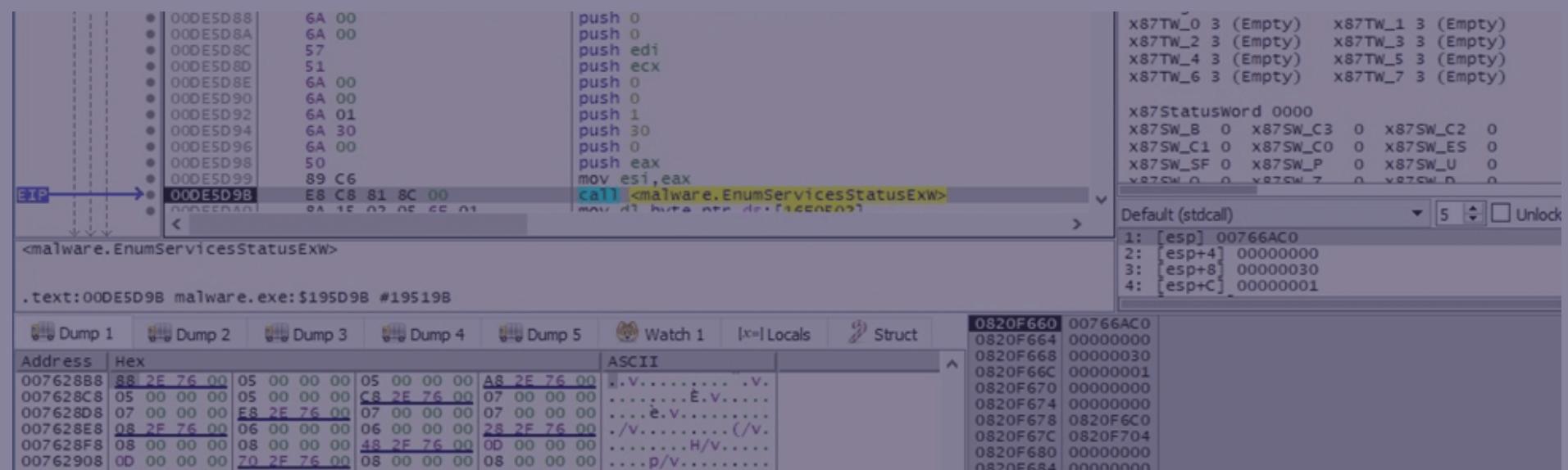


Figure 44

The malware targets the list of services from the `kill_services` element in the BlackCat configuration.

A targeted service is opened by calling the `OpenServiceW` routine (`0x2c = SERVICE_STOP | SERVICE_ENUMERATE_DEPENDENTS | SERVICE_QUERY_STATUS`):

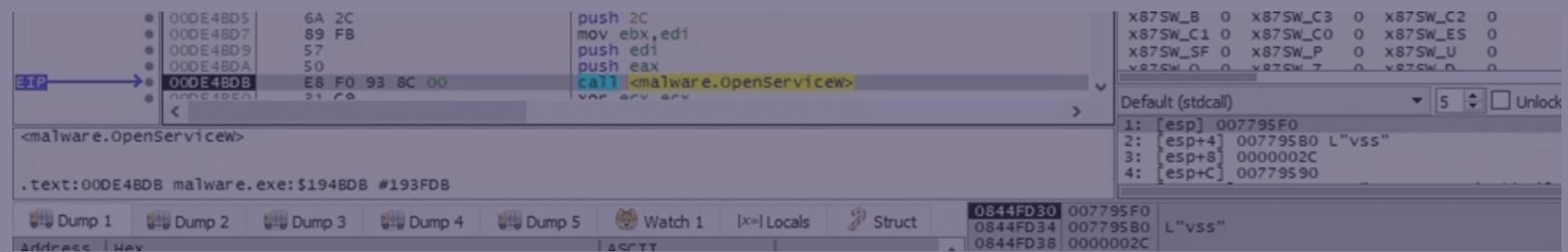
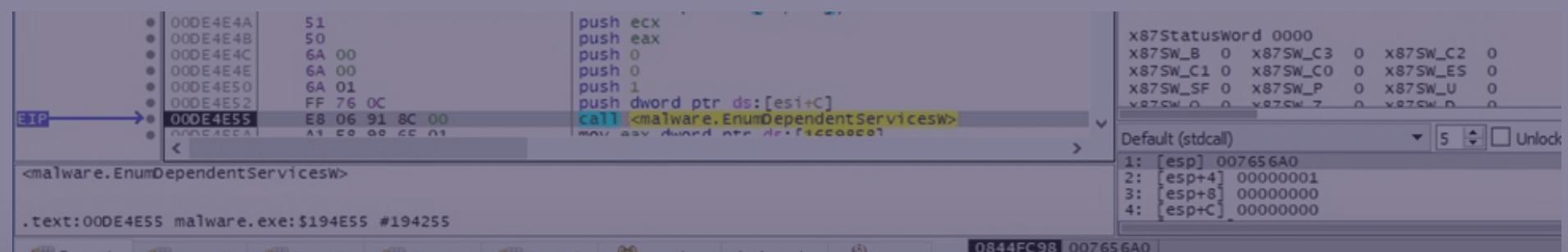


Figure 45

`EnumDependentServicesW` is utilized to retrieve the active services that depend on the targeted service (`0x1 = SERVICE_ACTIVE`):



#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

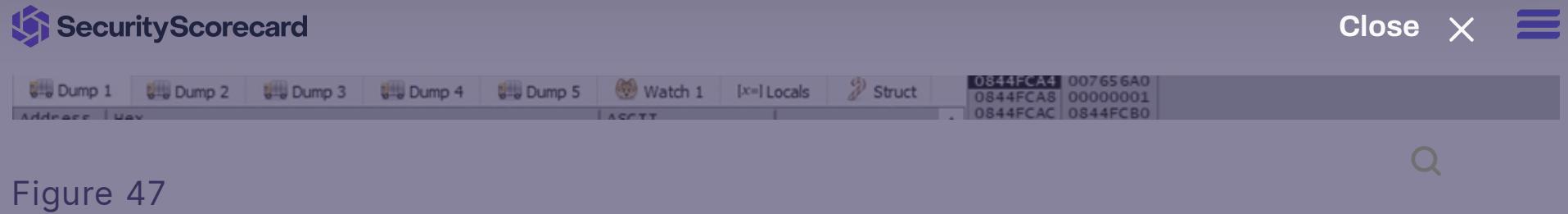


Figure 47

# Killing targeted processes

The executable takes a snapshot of all processes and threads in the system (0xF = TH32CS\_SNAPALL):

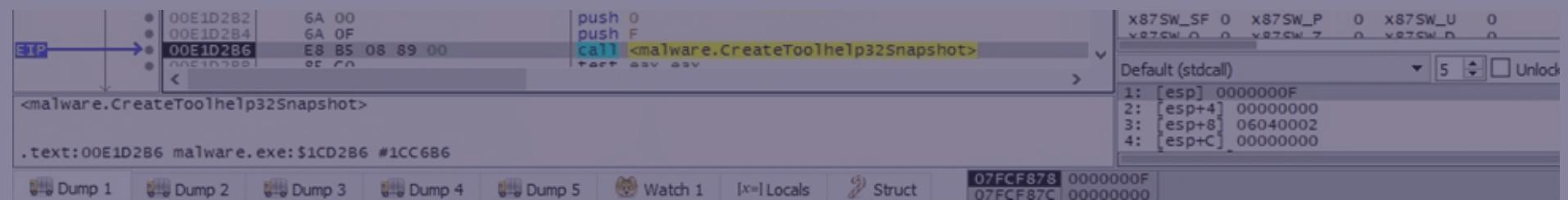


Figure 48

The processes are enumerated using the Process32FirstW and Process32NextW APIs:

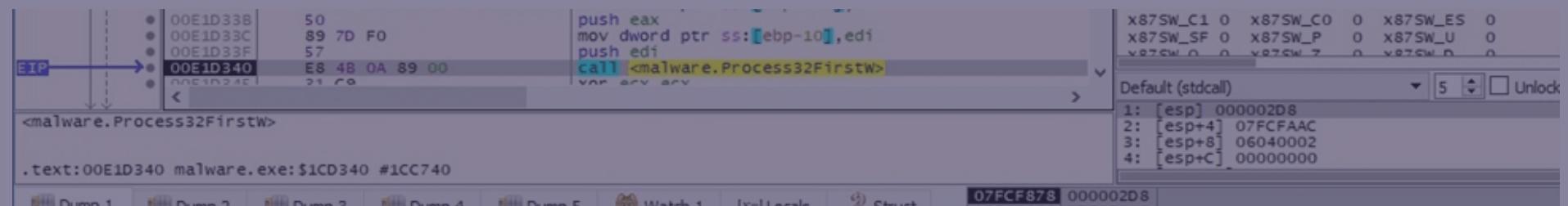


Figure 49

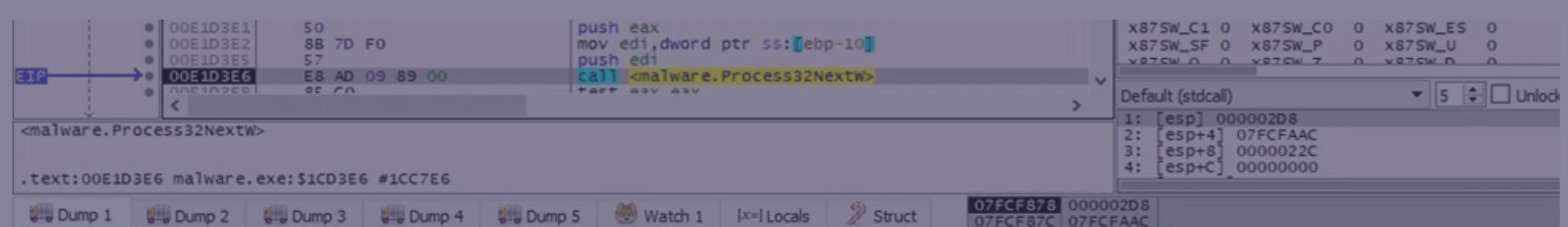
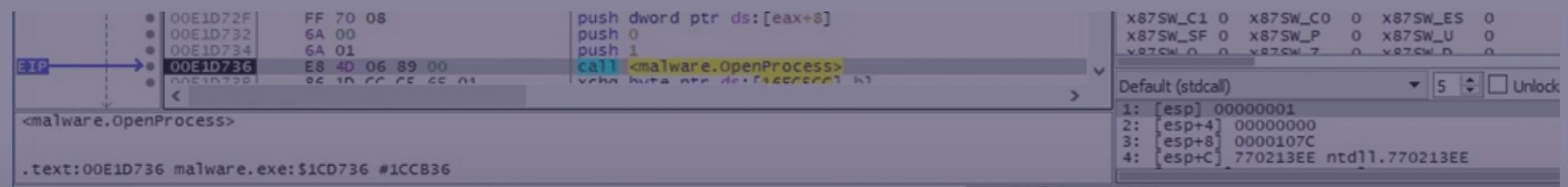


Figure 50

The malware targets the list of processes from the kill\_processes element in the BlackCat configuration.

It opens a targeted process using OpenProcess (0x1 = **PROCESS\_TERMINATE**):



This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

The screenshot shows the Immunity Debugger interface. The assembly pane displays the following code:

```
push eax  
push esi  
push 0  
push 0  
push 400  
mov esi,dword ptr ss:[ebp-20]  
push esi  
push 0  
push 2  
push dword ptr ss:[ebp-4C]  
push dword ptr ss:[ebp-50]  
push dword ptr ss:[ebp-54]  
call <malware.CreateProcessWithLogonW>
```

The instruction at address `00D7FB6A` is highlighted in yellow. The memory dump pane shows the following memory dump:

Address	Hex	ASCII
00768648	22 00 43 00 3A 00 5C 00 55 00 73 00	%.C.:.\U.s.e.r.
00768658	73 00 5C 00 [REDACTED] 5C 00	s.\.n.u.l.D.e.
00768668	73 00 6B 00 74 00 6F 00 70 00 5C 00	s.k.t.o.p.\m.a.
00768678	6C 00 77 00 61 00 72 00 65 00 2E 00	l.w.a.r.e..e.x.
00768688	65 00 22 00 20 00 2D 00	68 00 69 00 e."
00768698	6C 00 64 00 20 00 20 00 2D 00 2D 00	l.d. . .-.a.c.
007686A8	63 00 65 00 73 00 73 00 2D 00 74 00	c.e.s.s.-.t.o.k.
007686B8	65 00 6E 00 20 00 63 00 64 00 32 00	e.n..c.d.2.9.c.
007686C8	62 00 38 00 39 00 64 00 61 00 65 00	b.s.9.d.a.e.d.9.
007686D8	65 00 32 00 66 00 33 00 39 00 38 00	e.2.f.3.9.8.7.e.
007686E8	39 00 30 00 64 00 37 00 61 00 63 00	9.0.d.7.a.c.9.7.
007686F8	34 00 38 00 38 00 32 00 34 00 31 00	4.8.8.2.4.1.d.3.
00768708	66 00 38 00 37 00 65 00 32 00 31 00	f.8.7.e.2.1.a.5.
00768718	37 00 34 00 65 00 33 00 62 00 34 00	7.4.e.3.b.4.6.5.
00768728	65 00 31 00 35 00 35 00 34 00 34 00	e.1.5.5.4.4.e.0.
00768738	35 00 30 00 63 00 20 00 00 00 AB AB	5.0.c. ....<<<<<<

The registers pane shows the following register values:

Register	Value
EIP	00D7FB6A
ESP	00D7F9E8
EBP	00D7F9E0
ECX	00000000
EDX	00000000
ECB	00000000

The stack dump pane shows the following stack dump:

Address	Hex	ASCII
0790F9F8	007632E8	L"Administrator"
0790F9FC	007DCE88	
0790FA00	007DCF68	
0790FA04	00000002	
0790FA08	00000000	
0790FA0C	00768648	L"\"C:\\\\Users\\\\[REDACTED]\\\\Desktop\\\\malware.exe"
0790FA10	00000400	
0790FA14	00000000	
0790FA18	00000000	
0790FA1C	0790FA4C	
0790FA20	0790FA24	
0790FA24	01806B00	malware.01806B00
0790FA28	00000000	
0790FA2C	01684391	return to malware.01684391 from malware
0790FA30	00000001	
0790FA34	000002F0	
0790FA38	0000000C	
0790FA3C	00000000	
0790FA40	00000001	
0790FA44	0000025C	

Figure 53

The number of network requests the Server Service can make is set to the maximum by modifying “HKLM\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters\MaxMpxCt” Registry value:

Assembly pane:

```

● 0165098E 51 push ecx
● 0165098F 52 push edx
● 01650990 FF B4 24 80 00 00 00 push dword ptr ss:[esp+80]
● 01650997 FF 74 24 1C push dword ptr ss:[esp+1C]
● 01650998 FF B4 24 44 01 00 00 push dword ptr ss:[esp+144]
● 016509A2 6A 01 push 1
● 016509A4 6A 00 push 0
● 016509A6 6A 00 push 0
● 016509A8 FF B4 24 E4 00 00 00 push dword ptr ss:[esp+E4]
● 016509AF 50 push eax
● 016509B0 E8 A3 D1 05 00 call <malware.CreateProcessW>
● nicecode 0F C0 test ax, ax

```

Registers pane (Registers tab selected):

- EIP: 016509B0
- Call stack: 016509B0, 016509A8, 016509A4, 016509A2, 01650998, 01650997, 01650990, 0165098F, 0165098E

Stack dump pane (Registers tab selected):

Address	Hex	ASCII
007652C0	22 00 43 00	3A 00 5C 00 57 00 49 00 4E 00 44 00
007652D0	4F 00 57 00	53 00 5C 00 73 00 79 00 73 00 74 00
007652E0	65 00 6D 00	33 00 32 00 5C 00 63 00 6D 00 64 00
007652F0	2E 00 65 00	78 00 65 00 22 00 20 00 2F 00 63 00
00765300	20 00 22 00	72 00 65 00 67 00 20 00 61 00 64 00
00765310	64 00 20 00	48 00 4B 00 45 00 59 00 5F 00 4C 00
00765320	4F 00 43 00	41 00 4C 00 5F 00 4D 00 41 00 43 00
00765330	48 00 49 00	4E 00 45 00 5C 00 53 00 59 00 53 00
00765340	54 00 45 00	4D 00 5C 00 43 00 75 00 72 00 72 00
00765350	65 00 6E 00	74 00 43 00 6F 00 6E 00 74 00 72 00
00765360	6F 00 6C 00	53 00 65 00 74 00 5C 00 53 00 65 00
00765370	Z2 00 76 00	69 00 63 00 65 00 73 00 5C 00 4C 00
00765380	61 00 6E 00	6D 00 61 00 6E 00 53 00 65 00 72 00
00765390	76 00 65 00	72 00 5C 00 50 00 61 00 72 00 61 00
007653A0	6D 00 65 00	74 00 65 00 72 00 73 00 20 00 2F 00
007653B0	76 00 20 00	4D 00 61 00 78 00 4D 00 70 00 78 00
007653C0	43 00 74 00	20 00 2F 00 64 00 20 00 36 00 35 00
007653D0	35 00 33 00	35 00 20 00 2F 00 74 00 20 00 52 00
007653E0	45 00 47 00	5F 00 44 00 57 00 4F 00 52 00 44 00

Memory dump pane (Registers tab selected):

Address	Hex	ASCII
07B4F5A8	007654D8	L"C:\\\\WINDOWS\\\\system32\\\\cmd.exe"
07B4F5AC	007652C0	L"\"C:\\\\WINDOWS\\\\system32\\\\cmd.exe\" /c
07B4F580	00000000	
07B4F5B4	00000000	
07B4F5B8	00000001	
07B4F5BC	08000400	
07B4F5C0	00000000	
07B4F5C4	00000000	
07B4F5C8	07B4F6BC	
07B4F5CC	07B4F660	
07B4F5D0	00000002	
07B4F5D4	00000000	
07B4F5D8	007656E0	
07B4F5DC	00000018	
07B4F5E0	00000000	
07B4F5E4	00000026	
07B4F5E8	000002C8	
07B4F5EC	000002D4	
07B4F5F0	00000000	
07B4F5F4	007654D8	L"C:\\\\WINDOWS\\\\system32\\\\cmd.exe"
07B4F5F8	00000020	
07B4F5FC	0000001C	
07B4F5FD	00000000	

Figure 54

The malicious process obtains the ARP table using the arp command, as shown below:

This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

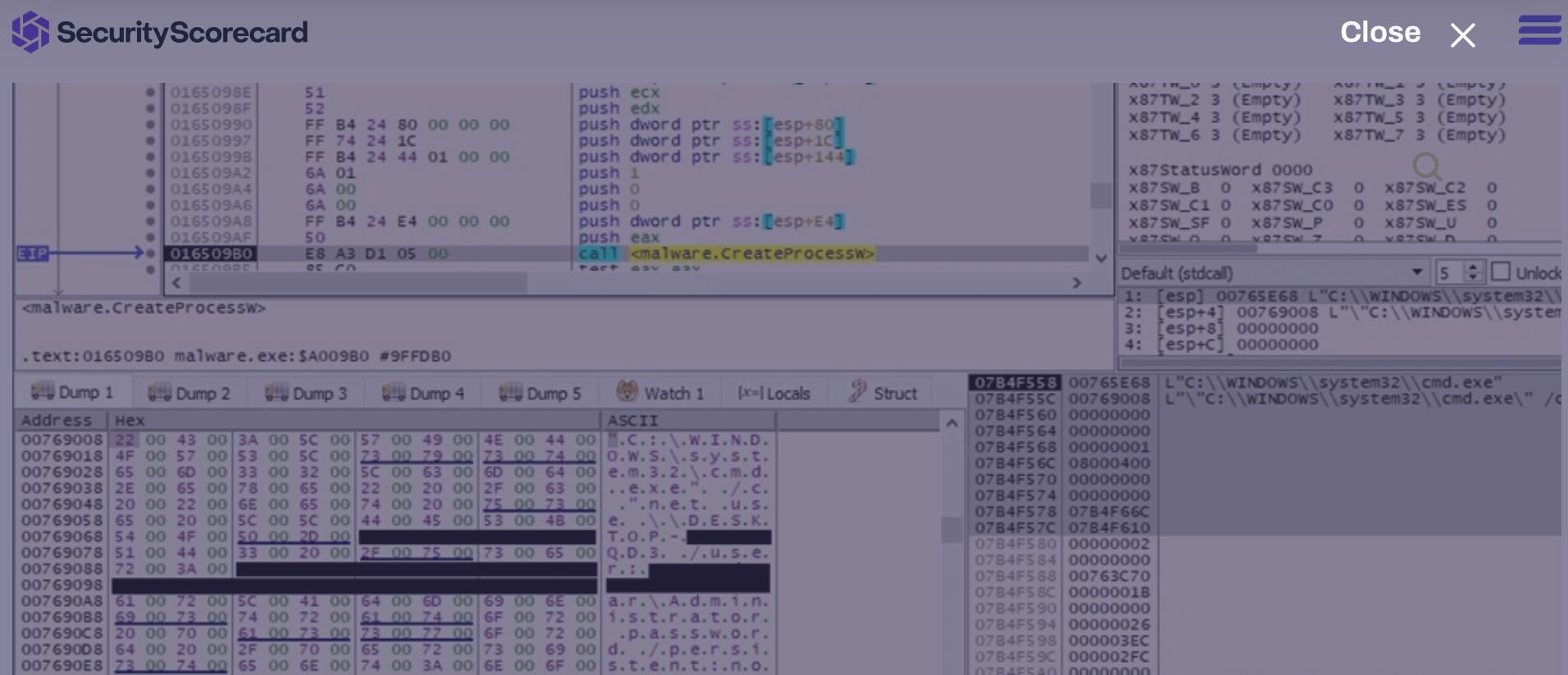


Figure 56

The malware retrieves the currently available disk drives by calling the GetLogicalDrives routine:

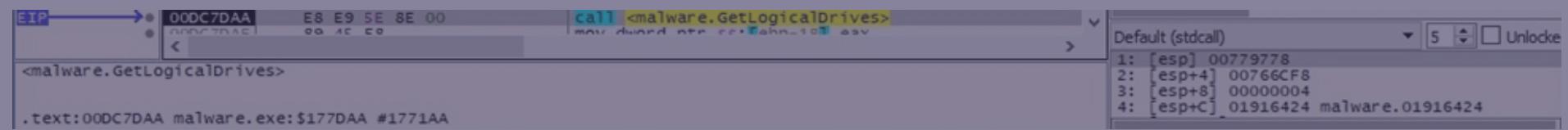


Figure 57

The GetDriveTypeW API is utilized to obtain the drive type:

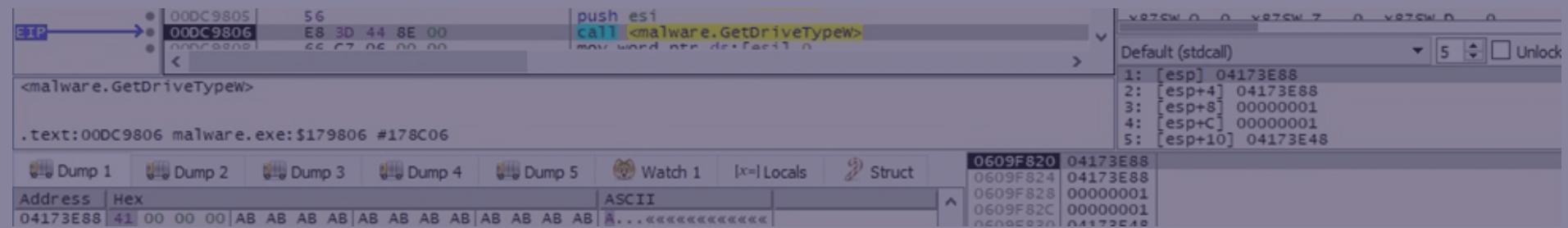


Figure 58

The ransomware starts scanning the volumes on the local machine using FindFirstVolumeW:

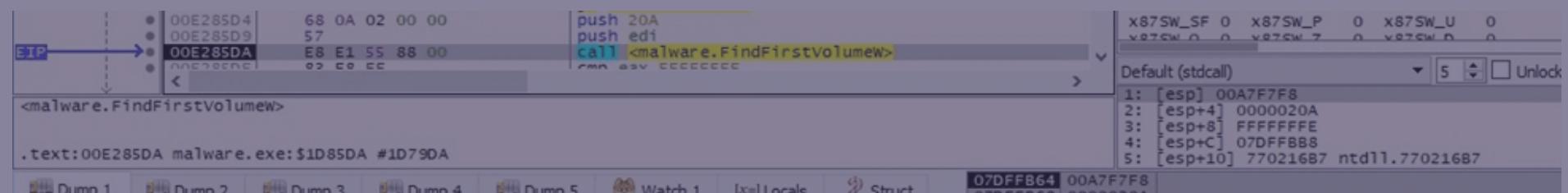


Figure 59

#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

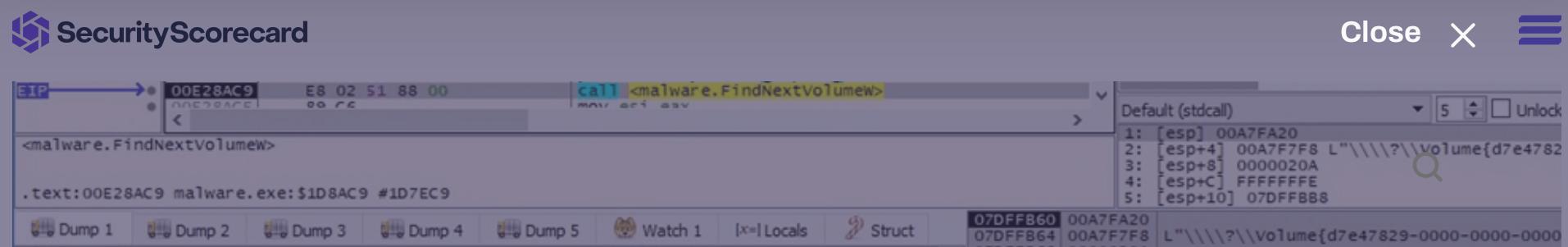


Figure 61

All unmounted volumes are mounted via a function call to SetVolumeMountPointW:

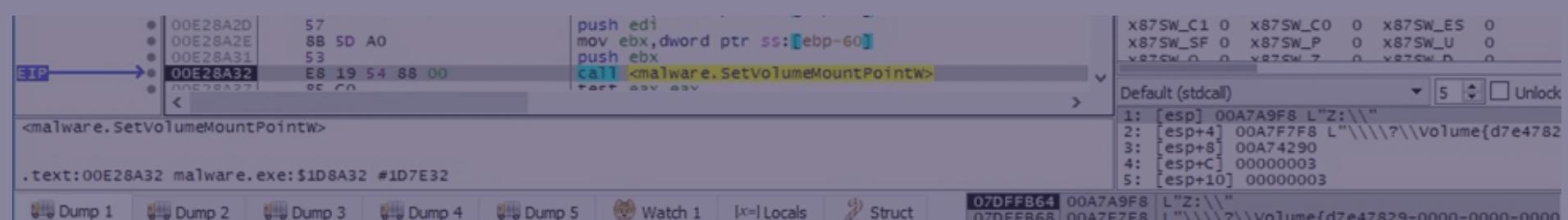


Figure 62

BlackCat traverses the file system using the FindFirstFileW and FindNextFileW APIs:

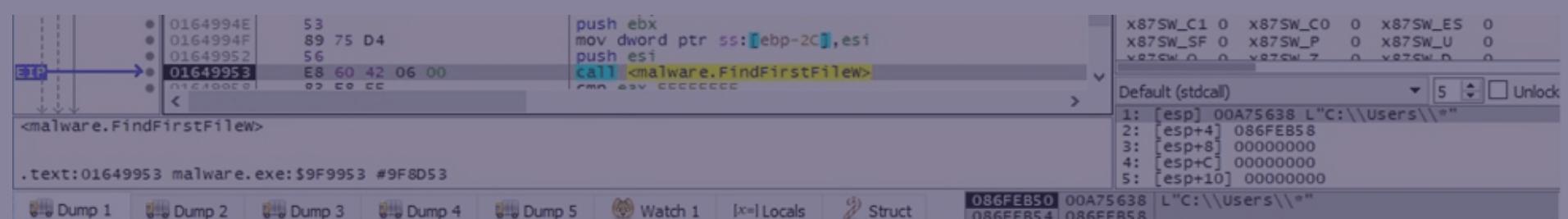


Figure 63

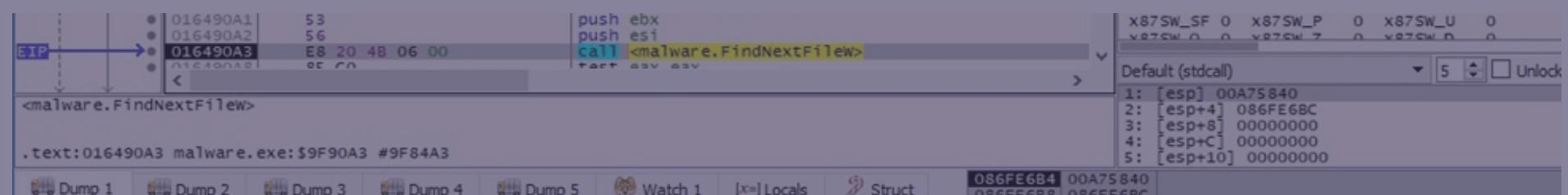


Figure 64

The BlackCat configuration is stored in JSON form and is decrypted at runtime. It contains:

- the extension appended to the encrypted files
- RSA public key that is used to encrypt the AES encryption key
- ransom note name and content
- stolen credentials specific to the victim's environment
- encryption cipher: AES

#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

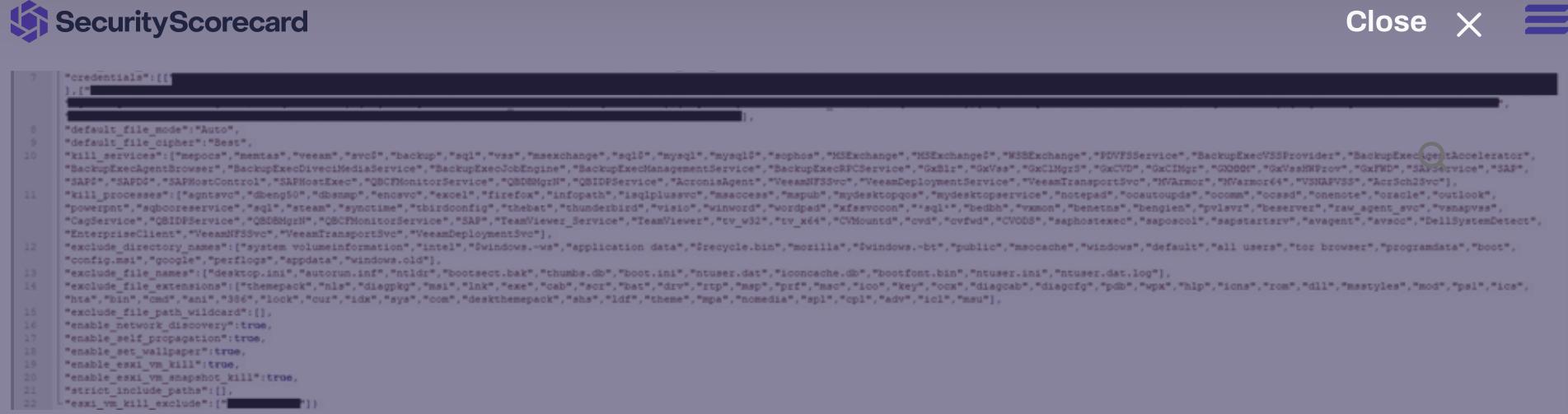


Figure 65

# Files encryption

The CreateFileW API is used to open a targeted file (0xC0000000 = **GENERIC\_READ** | **GENERIC\_WRITE**, 0x7 = **FILE\_SHARE\_DELETE** | **FILE\_SHARE\_WRITE** | **FILE\_SHARE\_READ**, 0x3 = **OPEN\_EXISTING**):

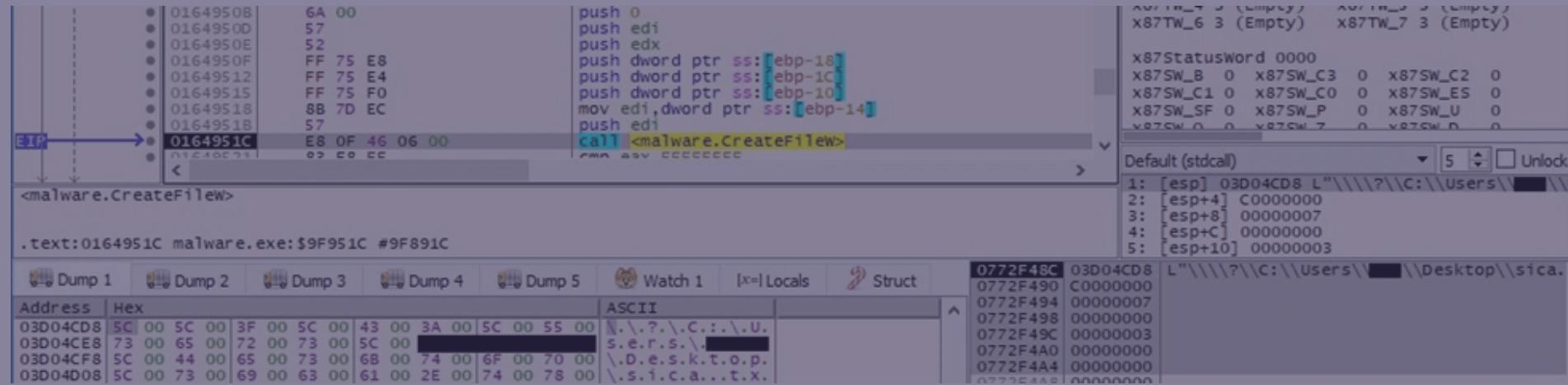
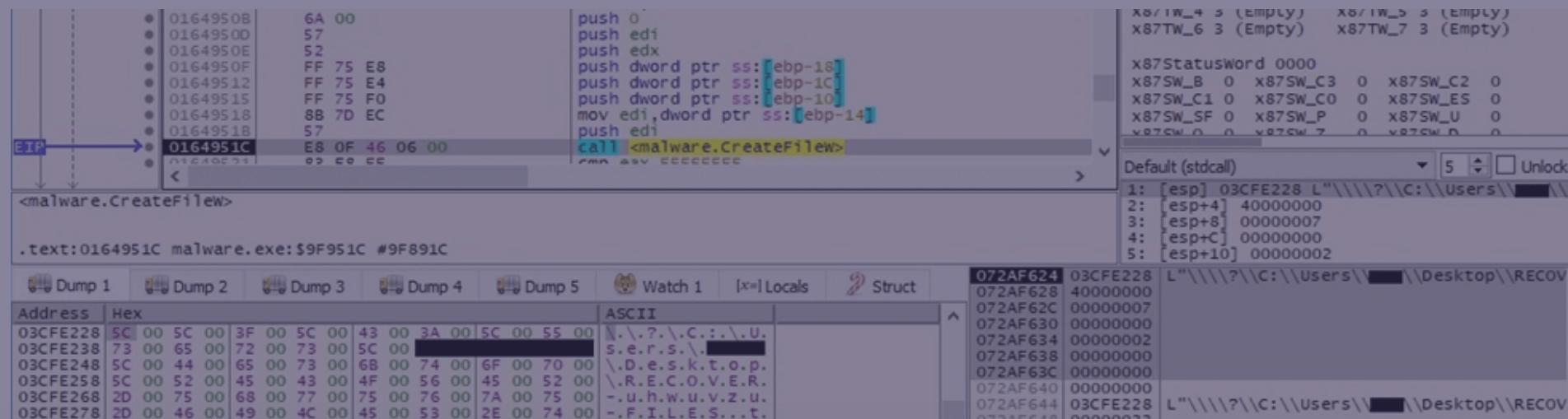


Figure 66

The ransom note is created in every traversed directory (0x40000000 = **GENERIC\_WRITE**, 0x7 = **FILE\_SHARE\_DELETE** | **FILE\_SHARE\_WRITE** | **FILE\_SHARE\_READ**, 0x2 = **CREATE\_ALWAYS**):



This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

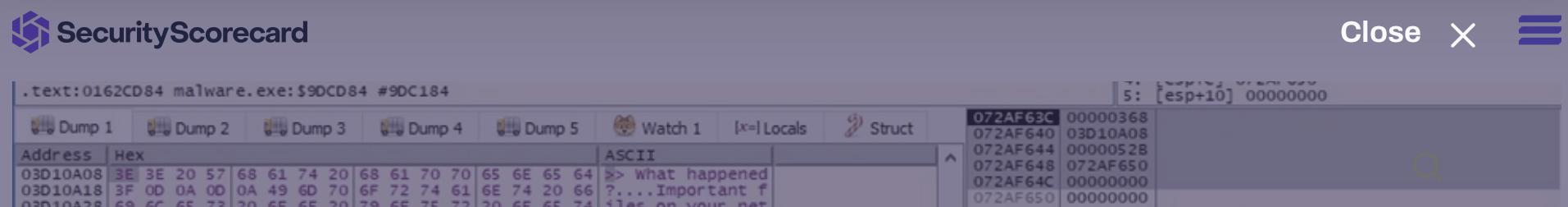


Figure 68

```

1 >> What happened?
2
3 Important files on your network was ENCRYPTED and now they have ".uhuvzu" extension.
4 In order to recover your files you need to follow instructions below.
5
6 >> Sensitive Data
7
8 Sensitive data on your network was DOWNLOADED.
9 If you DON'T WANT your sensitive data to be PUBLISHED you have to act quickly.
10
11 Data includes:
12 - Accounting;
13 - Finance;
14 - Confidential data;
15 - Projects;
16 - Blueprints;
17 - Database;
18 - Contracts and agreements;
19 - Personal data;
20 - Reports;
21 - And more...
22
23 Samples are available on your personal web page linked below.
24
25 >> CAUTION
26
27 DO NOT MODIFY ENCRYPTED FILES YOURSELF.
28 DO NOT USE THIRD PARTY SOFTWARE TO RESTORE YOUR DATA.
29 YOU MAY DAMAGE YOUR FILES, IT WILL RESULT IN PERMANENT DATA LOSS.
30
31 >> What should I do next?
32
33 1) Download and install Tor Browser from: https://torproject.org/
34 2) Navigate to:
http://nacp5njssx1h2doctut3utoch4773jq2pb16mgs3rjhyzumydonkqyd.onion/?access-key=o8ibWml57j7vnvH2YLIjOCmEhKrqAnl0lnvUDm9CeQ42BSChZ8J3KMc85Fr9WN1HTjjp1BPMZK49WODG2VyyjXk6A%2F2q0vRTbGyqRXe2HtQayvtR8o%2B04f37qp%2Fax1E0zQKnZLNrComYeeItA53N01xS2Cm%2FqxiIVIS3HPIq4bfUw6u2A3Id80mQe0d0K6qInGgK8H13pmQ%3D43C

```

Figure 69

The file's extension is changed using the MoveFileExW function. The renamed file is opened using CreateFileW (0x7 = FILE\_SHARE\_DELETE | FILE\_SHARE\_WRITE | FILE\_SHARE\_READ, 0x3 = OPEN\_EXISTING, 0x02000000 = FILE\_FLAG\_BACKUP\_SEMANTICS):

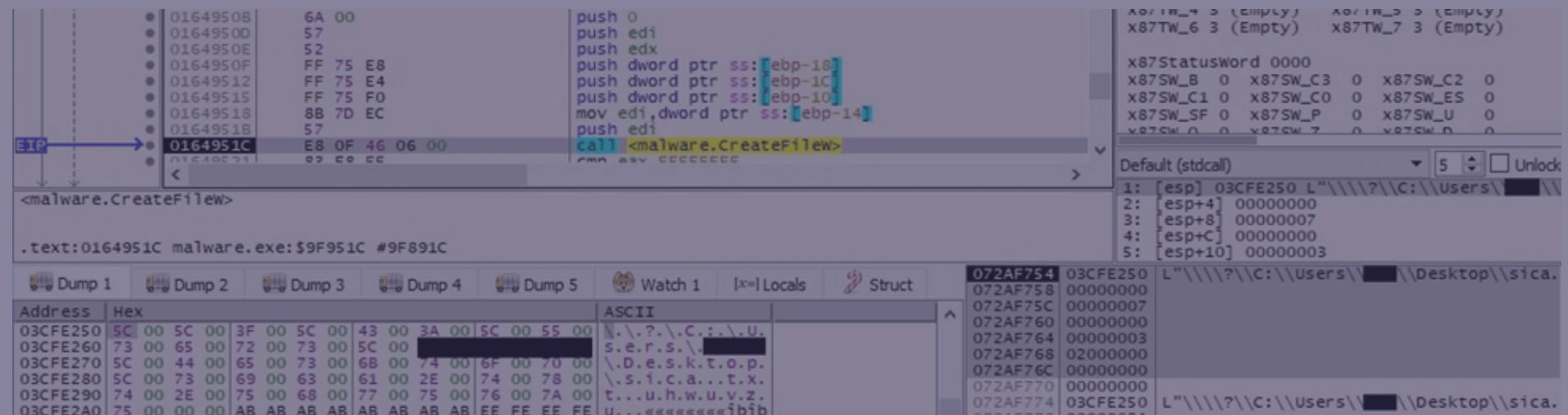
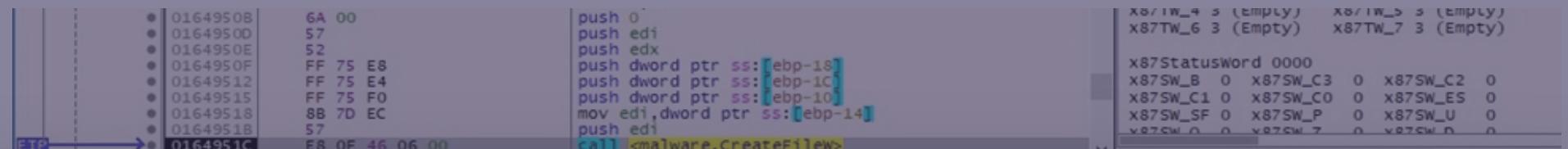


Figure 70

Interestingly, BlackCat creates intermediary files called "checkpoints-<encrypted file name>" during the encryption process:



#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

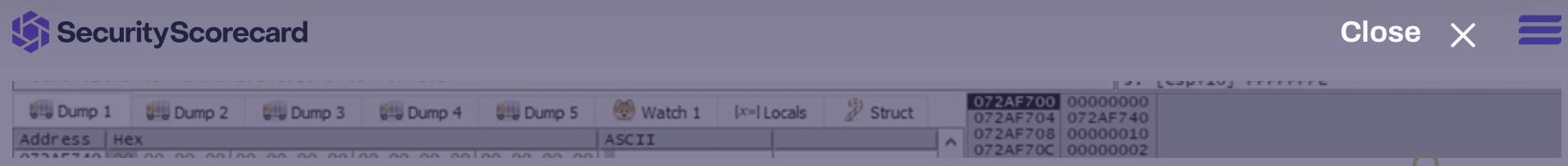


Figure 72

The ransomware moves the file pointer to the beginning of the file by calling the SetFilePointerEx API (0x0 = FILE\_BEGIN):

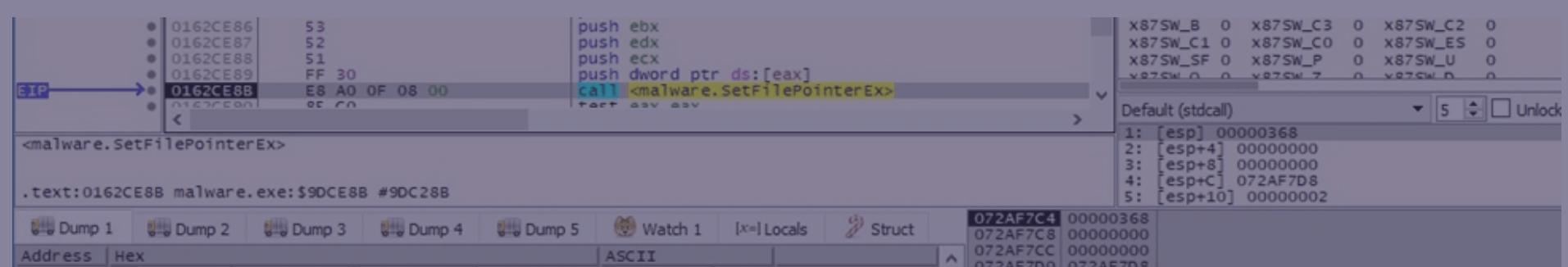


Figure 73

The process reads 4 bytes from the beginning of the file using ReadFile:

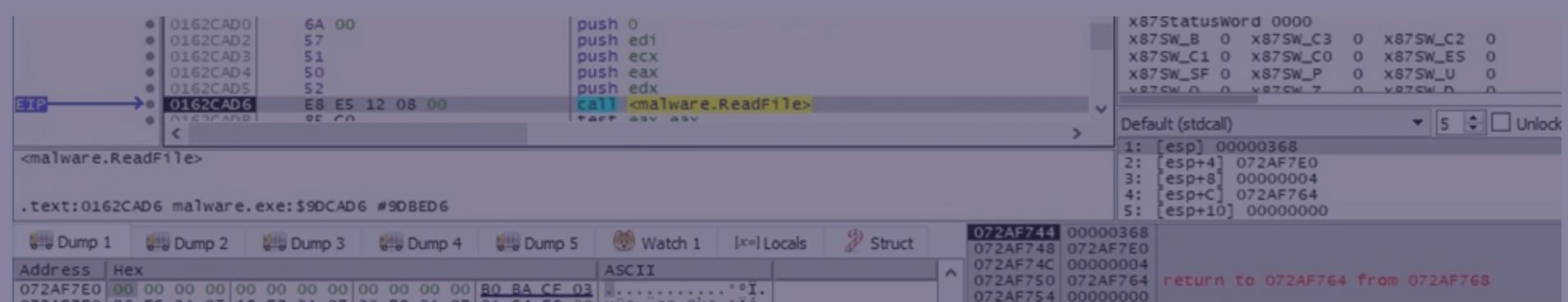


Figure 74

A JSON form containing the encryption cipher (AES), the AES key used to encrypt the file, the data, and the chunk size, is constructed in the process memory:

Address	Hex	ASCII
03D13F80	7B 22 76 65 72 73 69 6F 6E 22 3A 30 2C 22 6D 6F	{"version":0,"mo
03D13F90	64 65 22 3A 22 46 75 6C 6C 22 2C 22 63 69 70 68	de":"Full","ciph
03D13FA0	65 72 22 3A 22 41 65 73 22 2C 22 70 72 69 76 61	er":"Aes","priva
03D13FB0	74 65 5F 68 65 79 22 3A 5B 31 38 34 2C 31 32 39	te_key":[184,129
03D13FC0	2C 31 34 37 2C 31 31 36 2C 34 32 2C 32 31 31 2C	,147,116,42,211,
03D13FD0	35 35 2C 38 31 2C 34 33 2C 31 39 37 2C 31 35 2C	55,81,43,197,15,
03D13FE0	32 34 33 2C 31 31 30 2C 32 33 33 2C 32 33 36 2C	243,110,233,236,
03D13FF0	31 37 35 5D 2C 22 64 61 74 61 5F 73 69 7A 65 22	175],"data_size"
03D14000	3A 31 30 30 30 2C 22 63 68 75 6E 6B 5F 73 69 7A	:1000,"chunk_siz
03D14010	65 22 3A 32 35 33 36 32 38 31 36 2C 22 66 69 6E	e":25362816,"fin
03D14020	69 73 68 65 64 22 3A 66 61 6C 73 65 7D F0 AD BA	ished":false}]}.

Figure 75

The binary generates 0x50 (80) random bytes that are used to border the JSON form. The resulting buffer has a size of 256 bytes and is rotated using instructions such as pshuflw:

#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

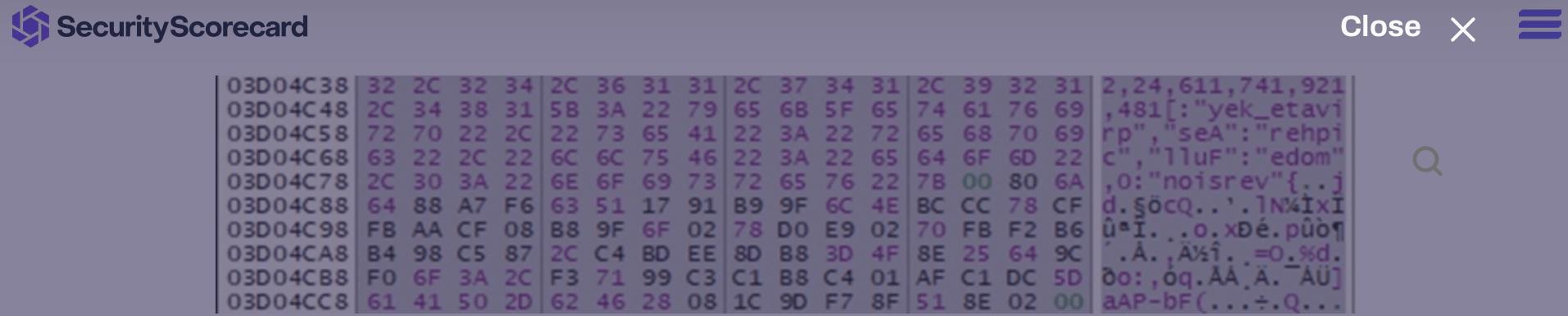


Figure 77

A 4-byte border “19 47 B2 CE” that separates the encrypted file content from the encrypted AES key is written to the file:

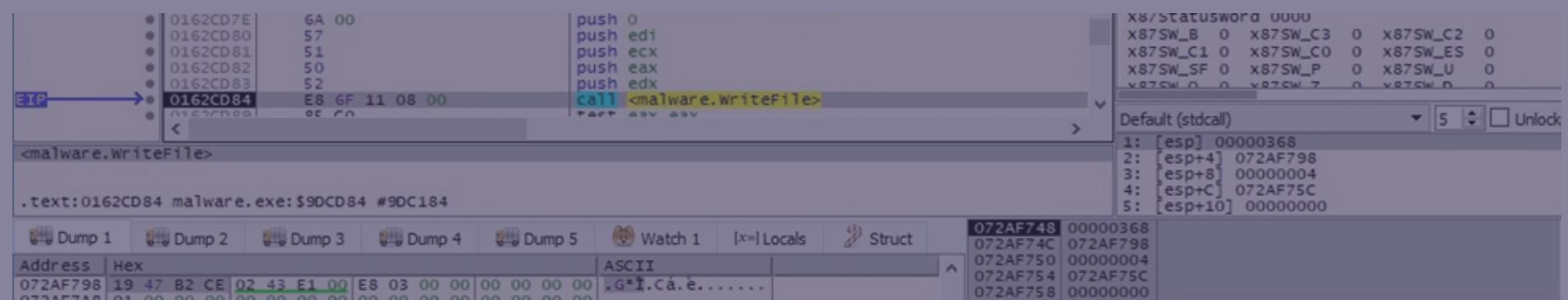


Figure 78

The buffer that contains the AES key presented in figure 77 is encrypted with the RSA public key from the BlackCat configuration. The result is written to the file using WriteFile:

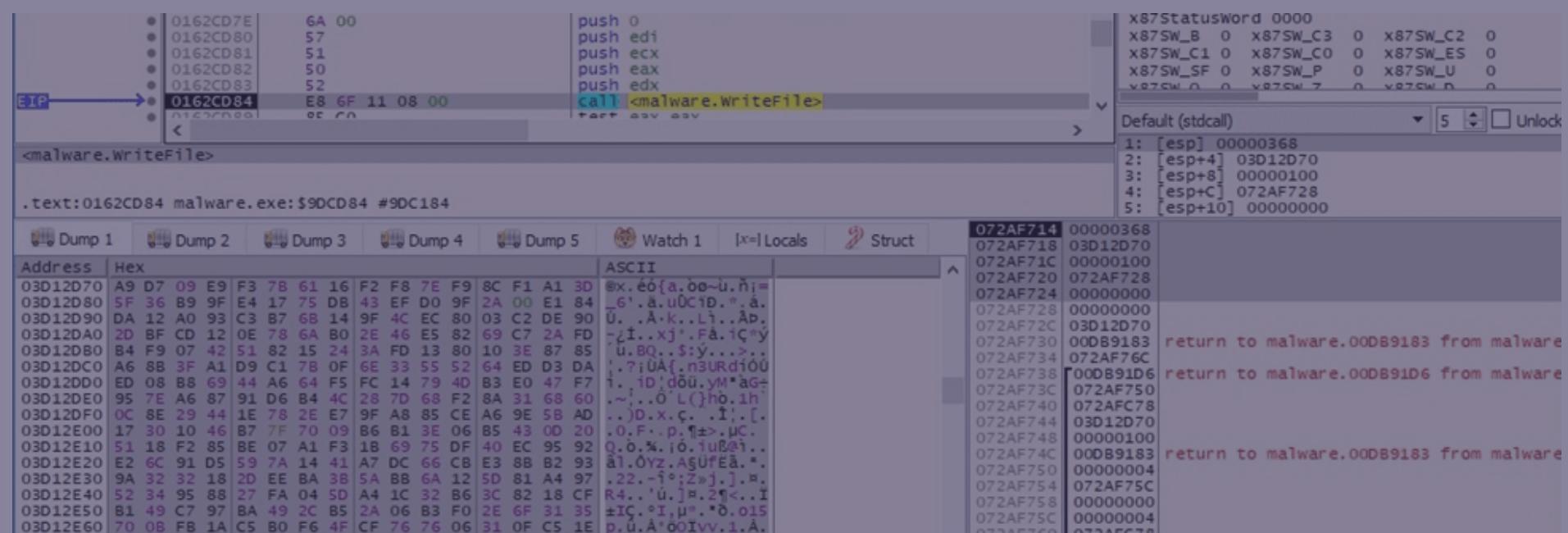
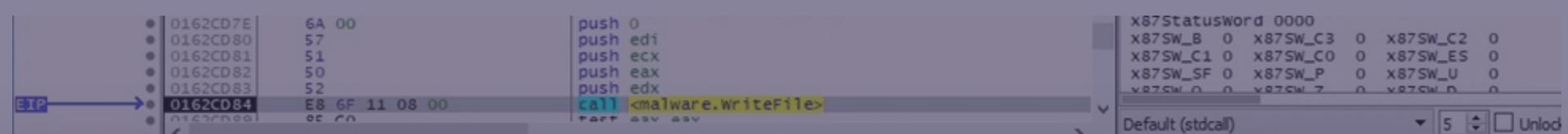


Figure 79

The size of encrypted key (0x100) is written to the file:



#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

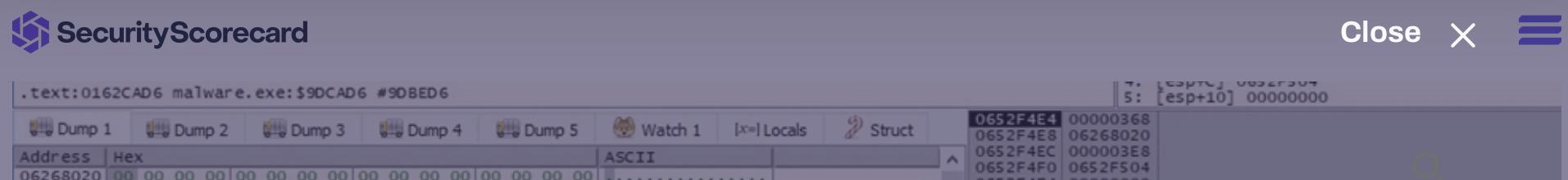


Figure 81

The file content is encrypted using the AES-128 algorithm. The malware uses the `aesenc` and `aesenclast` instructions for this purpose:

00DD84C7	66 OF 38 DC D0	aesenc xmm2,xmm0
00DD84CC	66 OF 38 DC D8	aesenc xmm3,xmm0
00DD84D1	66 OF 38 DC E0	aesenc xmm4,xmm0
00DD84D6	66 OF 38 DC E8	aesenc xmm5,xmm0
00DD84DB	66 OF 38 DC F0	aesenc xmm6,xmm0
00DD84E0	66 OF 38 DC F8	aesenc xmm7,xmm0
00DD84E5	66 OF 38 DC C8	aesenc xmm1,xmm0
00DD84EA	66 OF 7F 14 24	movdqa xmmword ptr ss:[esp],xmm2
00DD84EF	66 OF 6F 54 24 10	movdqa xmm2,xmmword ptr ss:[esp+10]
00DD84F5	66 OF 38 DC D0	aesenc xmm2,xmm0
00DD84FA	66 OF 6F 41 20	movdqa xmm0,xmmword ptr ds:[ecx+20]
00DD84FF	66 OF 7F 54 24 10	movdqa xmmword ptr ss:[esp+10],xmm2
00DD8505	66 OF 6F 14 24	movdqa xmm2,xmmword ptr ss:[esp]
00DD850A	66 OF 38 DC D8	aesenc xmm3,xmm0
00DD850F	66 OF 38 DC E0	aesenc xmm4,xmm0
00DD8514	66 OF 38 DC E8	aesenc xmm5,xmm0
00DD8519	66 OF 38 DC F0	aesenc xmm6,xmm0
00DD851E	66 OF 38 DC F8	aesenc xmm7,xmm0
00DD8523	66 OF 38 DC C8	aesenc xmm1,xmm0
00DD8528	66 OF 38 DC D0	aesenc xmm2,xmm0
00DD852D	66 OF 7F 14 24	movdqa xmmword ptr ss:[esp],xmm2
00DD8532	66 OF 6F 54 24 10	movdqa xmm2,xmmword ptr ss:[esp+10]
00DD8538	66 OF 38 DC D0	aesenc xmm2,xmm0
00DD853D	66 OF 6F 41 30	movdqa xmm0,xmmword ptr ds:[ecx+30]
00DD8542	66 OF 7F 54 24 10	movdqa xmmword ptr ss:[esp+10],xmm2
00DD854A	66 OF 6F 14 24	movdqa xmm2,xmmword ptr ss:[esp]

Figure 82

EIP	Address	Hex	Assembly
00DD87A0	55		push ebp
00DD87A1	89 E5		mov ebp,esp
00DD87A3	53		push ebx
00DD87A4	88 45 08		mov eax,dword ptr ss:[ebp+8]
00DD87A7	66 OF 6F 02		movdqa xmm0,xmmword ptr ds:[edx]
00DD87AB	B3 F1		mov bl,F1
00DD87AD	66 OF EF 00		pxor xmm0,xmmword ptr ds:[eax]
00DD87B1	66 OF 38 DC 42 10		aesenc xmm0,xmmword ptr ds:[edx+10]
00DD87B7	66 OF 38 DC 42 20		aesenc xmm0,xmmword ptr ds:[edx+20]
00DD87BD	66 OF 38 DC 42 30		aesenc xmm0,xmmword ptr ds:[edx+30]
00DD87C3	66 OF 38 DC 42 40		aesenc xmm0,xmmword ptr ds:[edx+40]
00DD87C9	66 OF 38 DC 42 50		aesenc xmm0,xmmword ptr ds:[edx+50]
00DD87CF	66 OF 38 DC 42 60		aesenc xmm0,xmmword ptr ds:[edx+60]
00DD87D5	66 OF 38 DC 42 70		aesenc xmm0,xmmword ptr ds:[edx+70]
00DD87DB	66 OF 38 DC 82 80 00 00 00		aesenc xmm0,xmmword ptr ds:[edx+80]
00DD87E4	86 1D FE 8B 6E 01		xchg byte ptr ds:[16E8BFE],bl
00DD87EA	66 OF 38 DC 82 90 00 00 00		aesenc xmm0,xmmword ptr ds:[edx+90]
00DD87F3	66 OF 7F 00		movdqa xmmword ptr ds:[eax],xmm0
00DD87F7	66 OF 38 DD 82 A0 00 00 00		aesenclast xmm0,xmmword ptr ds:[edx+A0]
00DD8800	66 OF 7F 01		movdqa xmmword ptr ds:[ecx],xmm0

Figure 83

The encrypted file content is written back to the file using `WriteFile`:

EIP	0162CD7E	6A 00	push 0
	0162CD80	57	push edi
	0162CD81	51	push ecx
	0162CD82	50	push eax
	0162CD83	52	push edx
EIP	0162CD84	E8 6F 11 08 00	call <malware.WriteFile>

#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.



Close X

```
C:\Program Files (x86)\qemu\qapp-firmware.rndj.uhwuvzu
C:\Program Files (x86)\qemu\edk2-x86_64-secure-code.fd.uhwuvzu
C:\Program Files (x86)\qemu\index.html.uhwuvzu
C:\Program Files (x86)\qemu\openbios-ppc.uhwuvzu
C:\Program Files (x86)\qemu\openbios-sparc32.uhwuvzu
C:\Program Files (x86)\qemu\openbios-sparc64.uhwuvzu
C:\Program Files (x86)\qemu\palcode-clipper.uhwuvzu
C:\Program Files (x86)\qemu\petalogix-m1605.dtb.uhwuvzu
C:\Program Files (x86)\qemu\qemu.svg.uhwuvzu
```



Figure 85

The ransomware creates a PNG image called “RECOVER-uhwuvzu-FILES.txt.png”:

```
C:\Users\...\Desktop>malware.exe --access-token cd29cb89daed9e2f3987e90d7ac97488241d3f87e21a574e3b465e15544e050c --verbose
13:28:07 MASTER Locker::core::stack: Starting Supervisor
13:28:07 MASTER Locker::core::stack: Starting Discoverer
13:28:07 MASTER Locker::core::stack: Starting File Unlockers
13:28:07 MASTER Locker::core::pipeline::chunk_workers_supervisor: spawned_workers=2
13:28:07 MASTER Locker::core::pipeline::file_worker_pool: spawned_file_dispatchers=2
13:28:07 MASTER Locker::core::pipeline::file_worker_pool: spawned_chunk_work_infrastructure=2
13:28:07 MASTER Locker::core::stack: Detecting Other Instances
13:28:07 MASTER Locker::core::stack: Starting Cluster Service
13:28:07 MASTER Locker::core::stack: Connecting to Cluster
13:28:07 MASTER Locker::core::stack: This is a Master Process
13:28:07 MASTER Locker::core::stack: Starting Platform
13:28:07 MASTER Locker::core::os::windows::privilegeEscalation: win7_plus=true
13:28:07 MASTER Locker::core::os::windows::privilegeEscalation: token_is_admin=true
13:28:07 MASTER Locker::core::os::windows::privilegeEscalation: token_is_domain_admin=true
13:28:07 MASTER encryptLib::windows: strict_include_paths={}
13:28:07 MASTER encryptLib::windows: strict_include_paths::local={}
13:28:07 MASTER encryptLib::windows: strict_include_paths::remote={}
13:28:07 MASTER locker::core::os::system_info: domain_name=
13:28:07 MASTER encryptLib::windows: initializing Networking Routine
13:28:07 MASTER locker::core::os::windows::system_info: username=
13:28:07 MASTER encryptLib::windows: IIS stop
13:28:07 MASTER locker::core::os::windows::privilegeEscalation: impersonate_spawn_trying::Administrator, ,password
13:28:07 MASTER locker::core::os::windows::privilegeEscalation: impersonate_spawn_trying::"C:\Users\...\Desktop\malware.exe" --child --access-token cd29cb89daed9e2f3987e90d7ac974882
11d3f87e21a574e3b465e15544e050c --verbose
13:28:07 MASTER locker::core::os::windows::privilegeEscalation: CreateProcessWithLogonW=success,1700
```

Figure 86

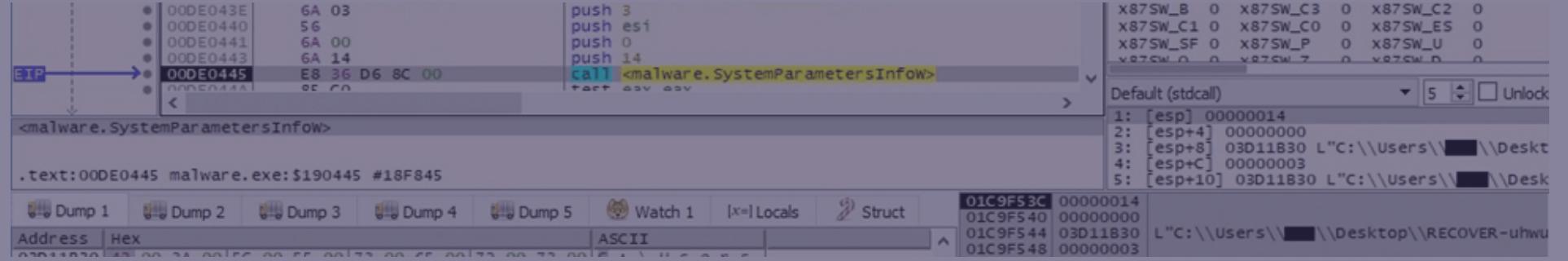
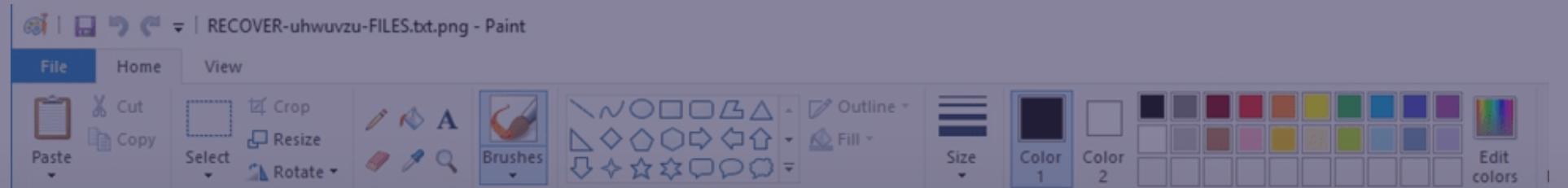
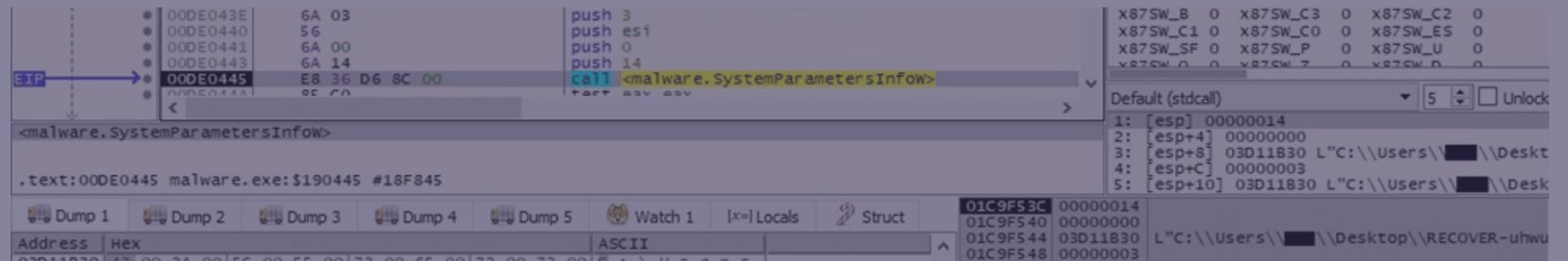


Figure 87

The Desktop wallpaper is changed to the above image by calling the SystemParametersInfoW API (0x14 = SPI\_SETDESKWALLPAPER, 0x3 = SPIF\_UPDATEINIFILE | SPIF\_SENDCHANGE):



#### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

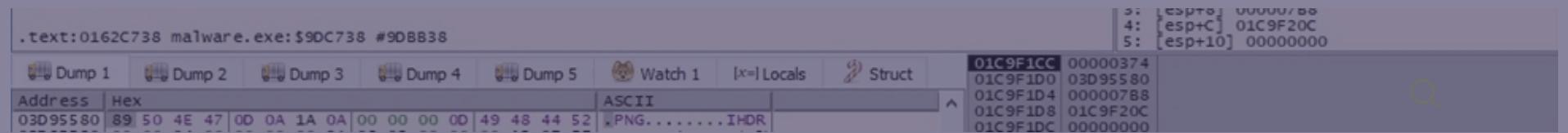


Figure 89

## Running with the `-extra-verbose -ui` parameters

The malware presents the relevant information in the following window:

```
sica.txt.uhwuvzu

Offset(h) 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
000001B0 DD 63 AC D7 EB 79 F6 CA 2E 3C CE 98 12 84 78 C0  Ýc-×ëyöÈ.<Î~..xÀ
000001C0 4B 8C BB F0 38 76 CD CE AA CA 46 4D 87 56 59 08  KE»ësvíí=ÉFM#VY.
000001D0 E7 51 F8 C7 3A 9F C4 4E 88 8A F2 1E 41 5F B7 7D  çQøÇ:ÝÄN~Šò.A_·}·
000001E0 C0 69 43 E9 05 09 30 29 40 B9 95 CB 78 EE D9 A8  ÄiCé..O)@··ÉxiÙ·
000001F0 91 26 2A 78 2E 38 6E 91 5A 43 E9 4F B1 A7 AA CB  '·&x.8n'ZCéO±S=È
00000200 68 31 A7 1C 32 2C 59 16 9C 53 C8 2A 06 2D B6 0F  h1$2,Y.œSÈ*.-¶.
00000210 9E 51 9B 53 1B D8 19 50 EE FA 61 2F 41 0F CE 84  žQ>S.Ø.Piúa/A.Í.
00000220 27 66 F4 E8 1A F1 E7 8E 9E C8 1A D0 B9 83 C0 51  'fðè.ñçžžÈ.Ð¹fÀQ
00000230 69 84 2C 8B 20 0C E8 79 FD 01 17 42 24 AA 61 CD  i,,,<.èyý..B$=áÍ
00000240 55 01 4E C4 D5 B2 85 52 63 03 F0 B9 1D 8D E8 A1  U.NÄÖ=...Rc.ð¹..è;
00000250 69 A1 E4 2C F3 30 7E B1 87 F8 7B 9D AD 4D 8A 75  i;ä,60~±#ø{..MŠu
00000260 84 E2 B6 2A 66 18 03 3A FB 0A 1C 66 69 56 E8 75  ..å¶*f...:ù..fiVèu
00000270 AC 7B 83 B7 46 05 19 63 2B A4 8D B4 A9 27 18 22  -{f·F..c+h.'@'.".
00000280 42 EE 8E 76 B3 D4 67 8D B4 7E 70 3C C9 51 04 10  Bižv·Ôg.'~p<ÉQ..
00000290 08 0A 99 9B 6A 33 5C A2 DF 9F 81 3E 4A 2F 7E 7A  .."m>j3\«BÝ.>J/~z
000002A0 8C 48 8C B7 C8 16 D3 95 8A 83 BB 90 15 7E EC 20  ØHE-E.Ó·Šf»..~í
000002B0 A7 CF C5 BC 09 7E C8 19 83 ED CD 2C 42 76 1E 94  SÍÅ¾.~È.fíí,Bv."
000002C0 5B 77 C3 D9 C5 32 19 D3 91 6A 7A 16 F2 AE D1 94  [wÄÜÅ2.Ó'jz.ØoÑ"
000002D0 55 23 C6 FC F6 AA FA 22 1A 8A EB CF DB E9 71 47  U#Æuö=ú".ŠëIÚéqG
000002E0 A2 CF 37 C8 33 BC C9 2D 75 1F 2F A4 E1 3D 02 A9  cÏ7È3¾É-u./ñá=.Ø
000002F0 EE F9 0A 7E 6C 3A 1D 51 D8 8A 53 E4 2D 2A A4 C1  iù..~1:.QØSSä-*ñÁ
00000300 32 38 D6 15 D4 3C 94 2A C8 1E 55 82 71 D8 63 10  28ö.Ö<"*È.U,qØc.
00000310 C2 A6 60 D9 6F 9B 55 63 A1 3B A7 39 DB A3 C6 00  Á: `Ùo>Uc;;S9Û£Æ.
00000320 87 1E E8 7F EA 8E 31 83 1C D1 1B B3 EF 3C 90 27  ±.è.èžlf.Ñ.·i<·
00000330 B0 A1 05 9A 92 30 F1 60 50 E7 25 C9 25 50 BA 37  ·i.š'Øñ`Pç%É%P°7
00000340 9C 08 35 D6 66 A0 CB 1F CE 36 5B E1 0B 1C FE 09  ø.5Öf È.Í6[á..p.
00000350 CA 41 AC D4 8C 8C E5 52 71 1F B6 11 EA 2F 5B C9  ÈA-ØŒŒåRq.¶.è/[É
00000360 70 7C 63 88 7C AC 84 F0 EB A0 25 13 56 63 91 F6  p|c^|~,,ðë % .Vc'Ø
00000370 EC 00 DF EF A1 OC 60 71 4C C4 10 82 44 D7 9A 0F  i.Bí..`qLÄ.,Dxš.
00000380 04 31 09 56 59 49 CD 1F 0D 65 33 34 0E F7 99 13  .1.VYIIÍ..e34.-"m.
00000390 14 F5 22 14 F1 0D CA C5 35 8D 2A 70 B6 60 AA A3  .ö".ñ.ÈÅ5.*p¶`*£
000003A0 B6 4E 07 83 C8 83 73 25 37 42 65 87 FA 91 09 81  ¶N.fÈfs%7Be#ú'..
000003B0 04 69 6D 4A 5A F3 83 94 1D 58 E0 FE 2C E1 FD 89  .imJZóf".Xàþ,áý%
000003C0 E7 9C E4 72 5E AA CB 76 74 27 F1 99 28 98 7D 13  çœär^=Èvt'ñ"m(").
000003D0 E2 A5 EF 53 1A 57 B6 BA 1F 38 12 07 EF BD 0E 71  åWiS.W¶°.8..i¾.q
000003E0 0E 0E 5A 09 64 F8 06 93 19 47 B2 CE AF 69 00 0B  ..Z.dø..".G=í-i..
000003F0 A6 20 03 E7 79 0C 0D AC FD 3B 98 6B DD 1B 90 BE  i..çy..~ý; "kÝ..%
00000400 97 30 3B 44 D2 43 1D 0C 04 2D 34 74 8A F8 6E FD  -0;DØC...-4tŠøný
00000410 48 19 DF AE 9E 74 A0 F7 B5 8B CF 8C 8C 44 8E 9E  H.Øøzt ÷u<ïGEDžž
00000420 EA 70 EE 36 75 80 A0 2D DD F6 BB B2 CA 05 BB 03  èpiëu€ -Ýö»=È.».
00000430 A4 F2 A0 78 13 35 89 95 EF 6F 86 26 99 F3 66 9F  ñò x.5%·iot&mófý
00000440 FE 7F BE 14 28 20 90 F9 5C 28 B2 0C 7F 63 38 5B  b.%.( .ù\(^..c8[
00000450 F0 5F 03 98 7D F8 87 75 9A 23 EA 3A CD 75 94 18  ð..~}øtuš#è:íu".
00000460 55 C3 61 9A F8 5C 69 94 AD 1A E7 69 5B 02 9A F5  UÅašø\i".."çí[.šõ
00000470 53 B3 D0 F0 34 0F C9 F2 76 57 A8 66 B6 17 AA E5  S·Ð84.ÉòvW" f¶.·å
00000480 6F 29 3E DE 02 60 0A 5C A6 19 37 D7 E7 45 B1 CE  o)>P.·.\\!.7×çE±í
00000490 76 DC E4 DC DA 2B 9A 1D 16 A3 CC AE E6 64 0B B0  vÜäÜÜ+š..£ìØæd.º
000004A0 BD 2C 27 E0 2B 17 13 88 6F 80 7E 1C 3F 66 9B 75  ª, 'à+..^o€~..?f>u
000004B0 CO 0C 3F 9A 4C 6C 04 7B 1B 18 89 29 46 AD 25 DD  Å.?šLl.(..%)F.%Ý
```

### This website uses cookies

We use cookies to personalize content and ads, to provide social media features and to analyze our traffic. We also share information about your use of our site with our social media, advertising and analytics partners who may combine it with other information that you've provided to them or that they've collected from your use of their services.

## Files created

checkpoints-<Filename>.uhwuvzu

RECOVER-uhwuvzu-FILES.txt.png

## Processes spawned

cmd.exe /c "wmic csproduct get UUID"

cmd.exe /c "fsutil behavior set SymlinkEvaluation R2L:1"

cmd.exe /c "fsutil behavior set SymlinkEvaluation R2R:1"

cmd.exe /c "iisreset.exe /stop"

cmd.exe /c "vssadmin Delete Shadows /all /quiet"

cmd.exe /c "wmic.exe Shadowcopy Delete"

cmd.exe /c "bcdedit /set {default}"

cmd.exe /c "bcdedit /set {default} recoveryenabled No"

cmd.exe /c for /F "tokens=\*" %1 in ('wevtutil.exe el') DO wevtutil.exe cl %1

cmd.exe

/c "reg add

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters  
/v MaxMpxCt /d 65535 /t REG\_DWORD /f"

cmd.exe /c "arp -a"

Platform	Why <b>SecurityScorecard?</b>	Services	Partners	Resources	Company
Supply Chain Cyber Risk		Digital Forensics & Incident Response	Locate a Partner	Blog	Leadership
Threat Landscape	Security Ratings	Advisory Services	Value-Added Resellers	Research	Press
Security & Risk Operations	Customer Stories	Penetration Testing	Managed Service Providers	Learning Center	Events
Cyber Insurance	Trust & Collaboration	Red Team	ISAC Partner Program	Webinars	Policy Insights
MAX	Marketplace	Tabletop Exercises	Technology Alliances	Tools & Documentation	Careers
Pricing & Packages			SCORE Portal Login	Public Scorecards	Contact Us
					Patents