Practical Malware Analysis & Triage
Malware Analysis Report

WannaCry Ransomware

Feb 2024

Table of Contents

Executive Summary

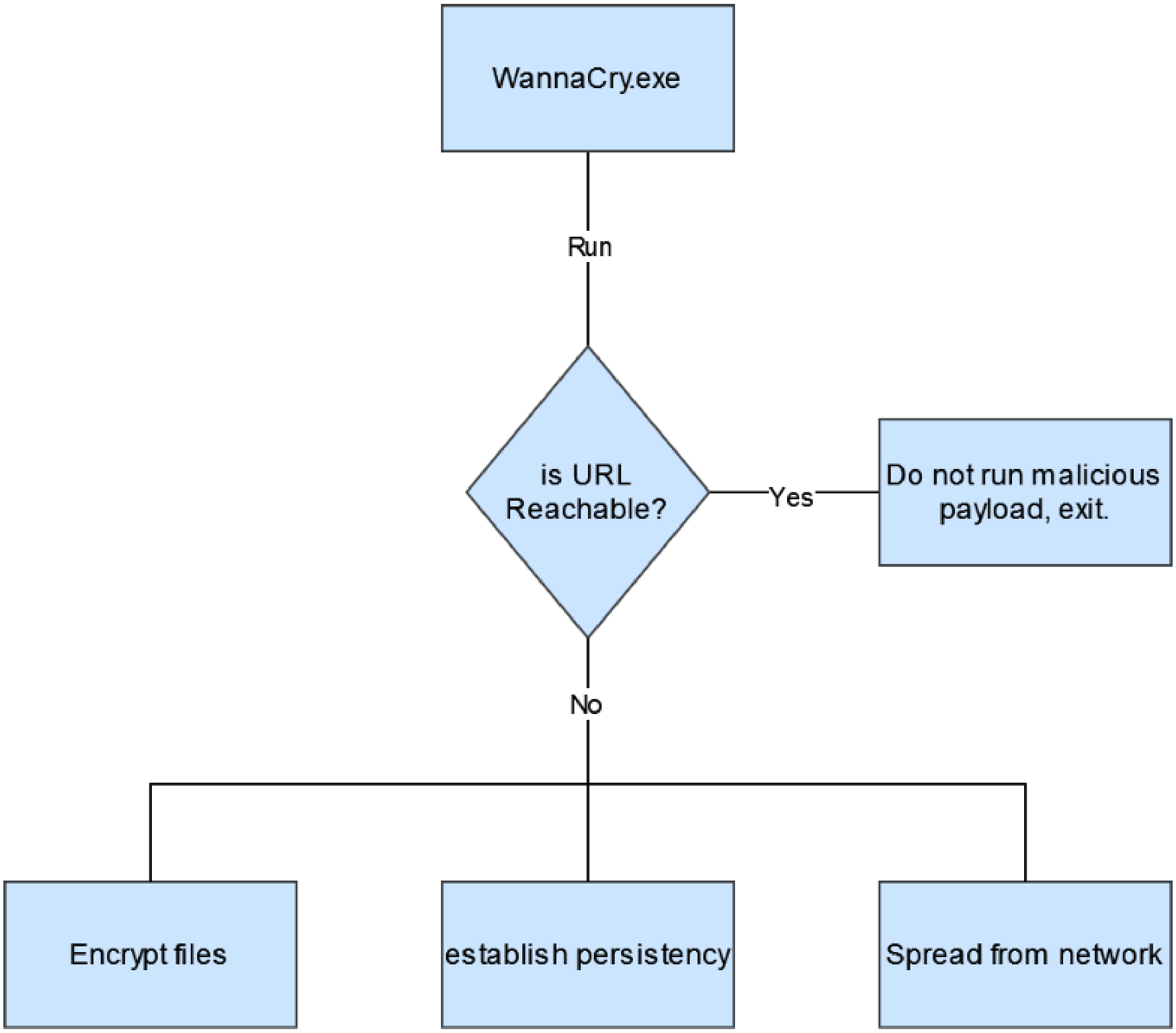
SHA256 hash	24D004A104D4D54034DBCFFC2A4B19A11F39008A575AA614EA04703480B1022C

WannaCry is a crypto ransomware used to extort money that became a global pandemic in May 2017. This ransomware spread through computers with Microsoft Windows OS. The user’s files were held hostage, and a Bitcoin Ransom was demanded for their return.

This file encrypts all files on the victim and uses it as pivot to spread itself on other computers on the network. Additionally, it presents persistence capabilities, keeping the files encrypted and the malware active after reboot.

High-Level Technical Summary

WannaCry presents itself with a killswitch mechanism that won't trigger the encryption if a certain URL is available and returns a 200OK response.
When not available, it proceeds to create persistence via a Registry Key and encrypting all the files, starting from the local directory of the file.
Also, the file has worm capabilities, spreading through the network with specific Source and Destination Ports.



Static Analysis

Information extracted without executing the sample. Tools used:
CFF Explorer, FLOSS, PEStudio,

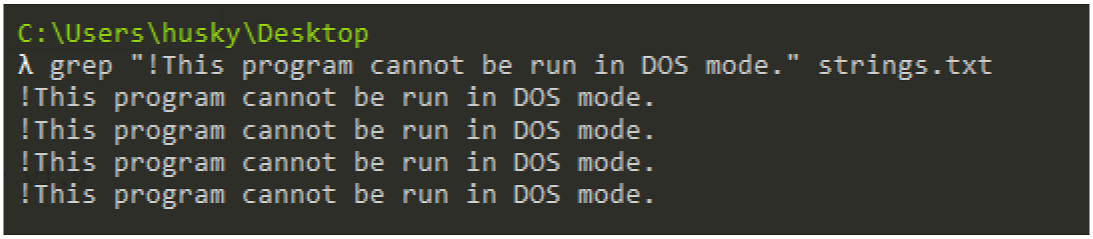
Original name	Ransomware.wannacry.exe
Written Language	C++
Architecture	32 Bits

Extracted Strings

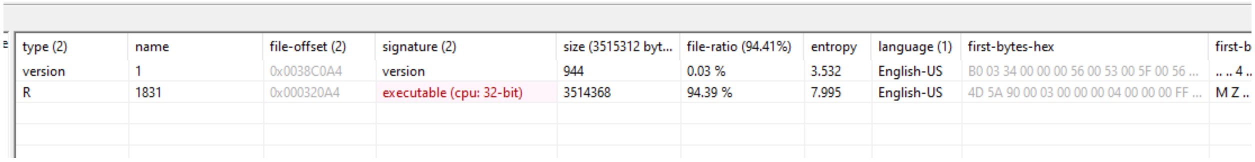
String	Information
hxxp://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com	URL
cmd.exe /c "%s"	Executes command on variable %s
diskpart.exe	Command interpreter helps manage the computer's drives
CreateServiceA	Create a service object and add it to the service control manager database. This function is commonly used by malware for persistence.
StartServiceA	
CreateServiceA	
InternetOpenA	
InternetOpenUrlA	
icacls . /grant Everyone:F /T /C /Q	
GetStartupInfoA	
mssecsvc.exe	

tasksche.exe	
Ihdfrgui.exe	
CryptEncrypt	
CryptDestroyKey	
C:\%s\qeriuwjhrf	
WanaCrypt0r	

The string "**!This program cannot be run in DOS mode.**" Appears 4 times, which suggests more than 1 executable in the file. Confirmed resources with PEStudio

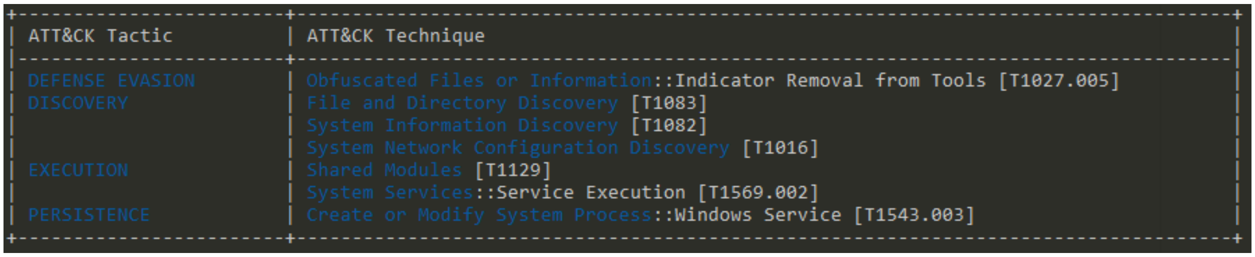


```
C:\Users\husky\Desktop
λ grep "!This program cannot be run in DOS mode." strings.txt
!This program cannot be run in DOS mode.
!This program cannot be run in DOS mode.
!This program cannot be run in DOS mode.
!This program cannot be run in DOS mode.
```

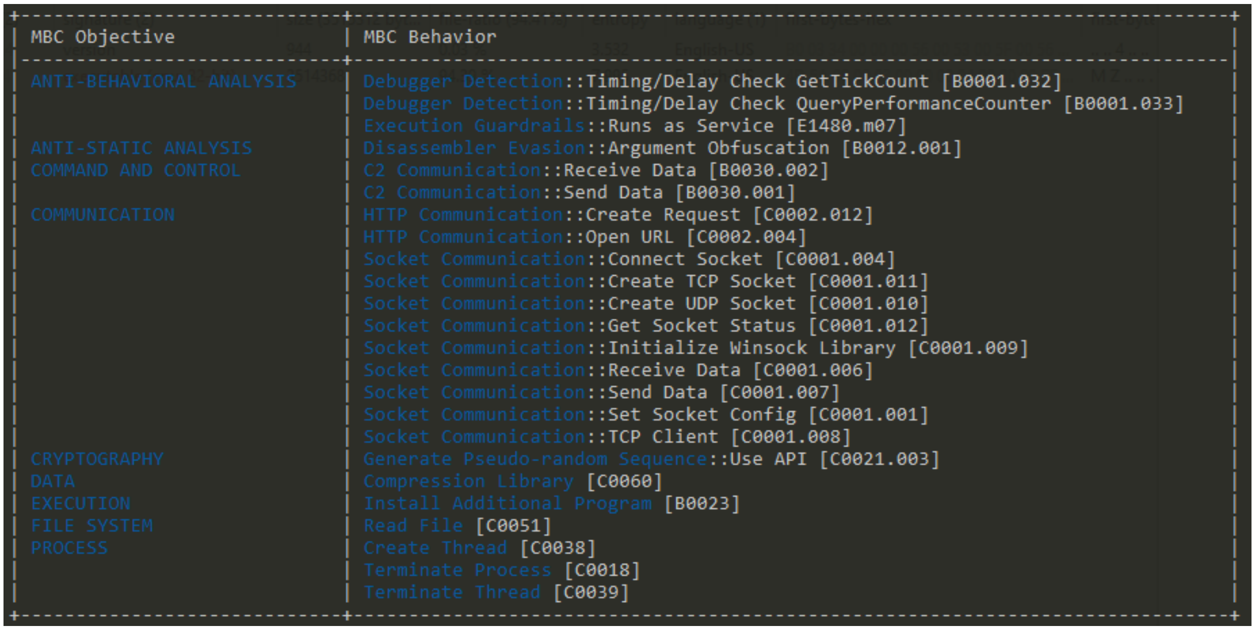


type (2)	name	file-offset (2)	signature (2)	size (3515312 byt...	file-ratio (94.41%)	entropy	language (1)	first-bytes-hex	first-b
version	1	0x0038C0A4	version	944	0.03 %	3.532	English-US	B0 03 34 00 00 00 56 00 53 00 5F 00 56 ...	.. .. 4 ..
R	1831	0x000320A4	executable (cpu: 32-bit)	3514368	94.39 %	7.995	English-US	4D 5A 90 00 03 00 00 00 04 00 00 00 FF ...	M Z ..

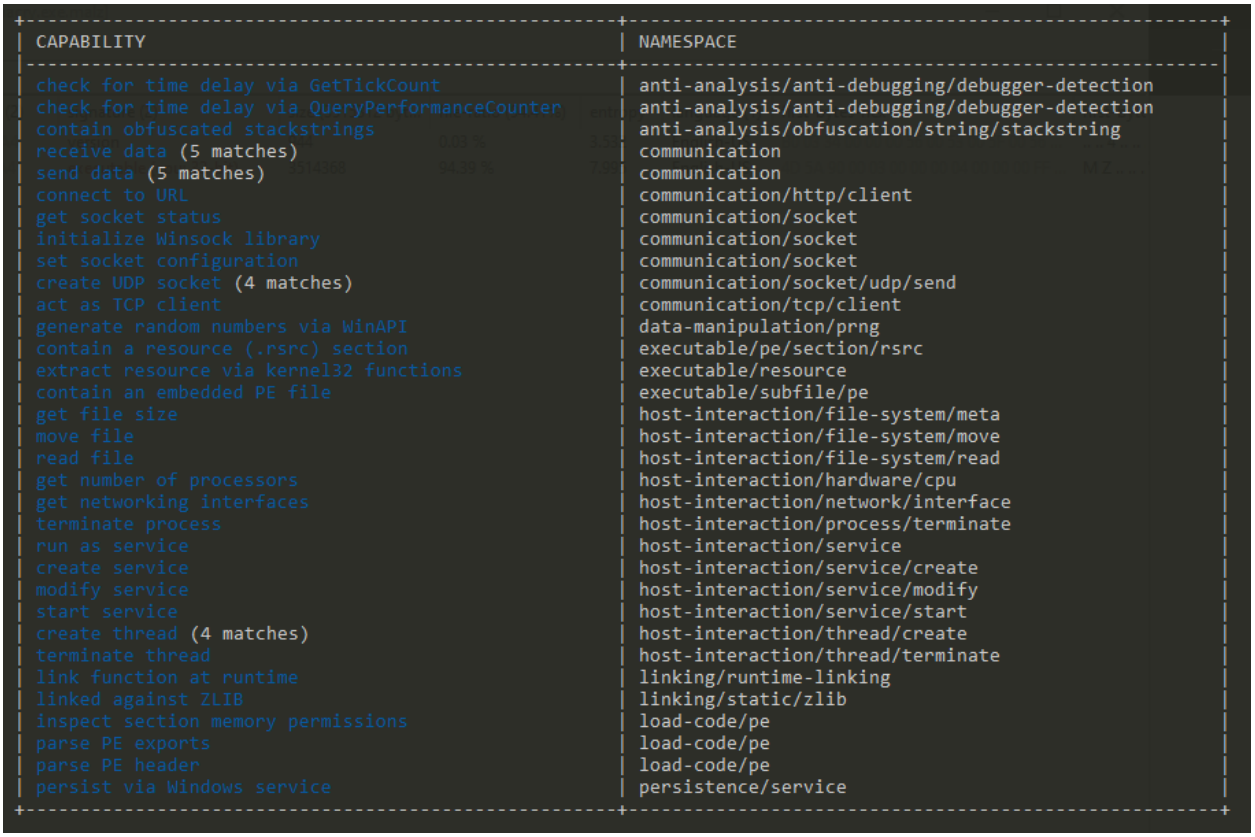
Capabilities



```
+--------------------------+-------------------------------------------------------------------------------+
| ATT&CK Tactic            | ATT&CK Technique                                                              |
|--------------------------+-------------------------------------------------------------------------------|
| DEFENSE EVASION          | Obfuscated Files or Information::Indicator Removal from Tools [T1027.005]     |
| DISCOVERY                | File and Directory Discovery [T1083]                                          |
|                          | System Information Discovery [T1082]                                          |
|                          | System Network Configuration Discovery [T1016]                                |
| EXECUTION                | Shared Modules [T1129]                                                        |
|                          | System Services::Service Execution [T1569.002]                                |
| PERSISTENCE              | Create or Modify System Process::Windows Service [T1543.003]                  |
+--------------------------+-------------------------------------------------------------------------------+
```



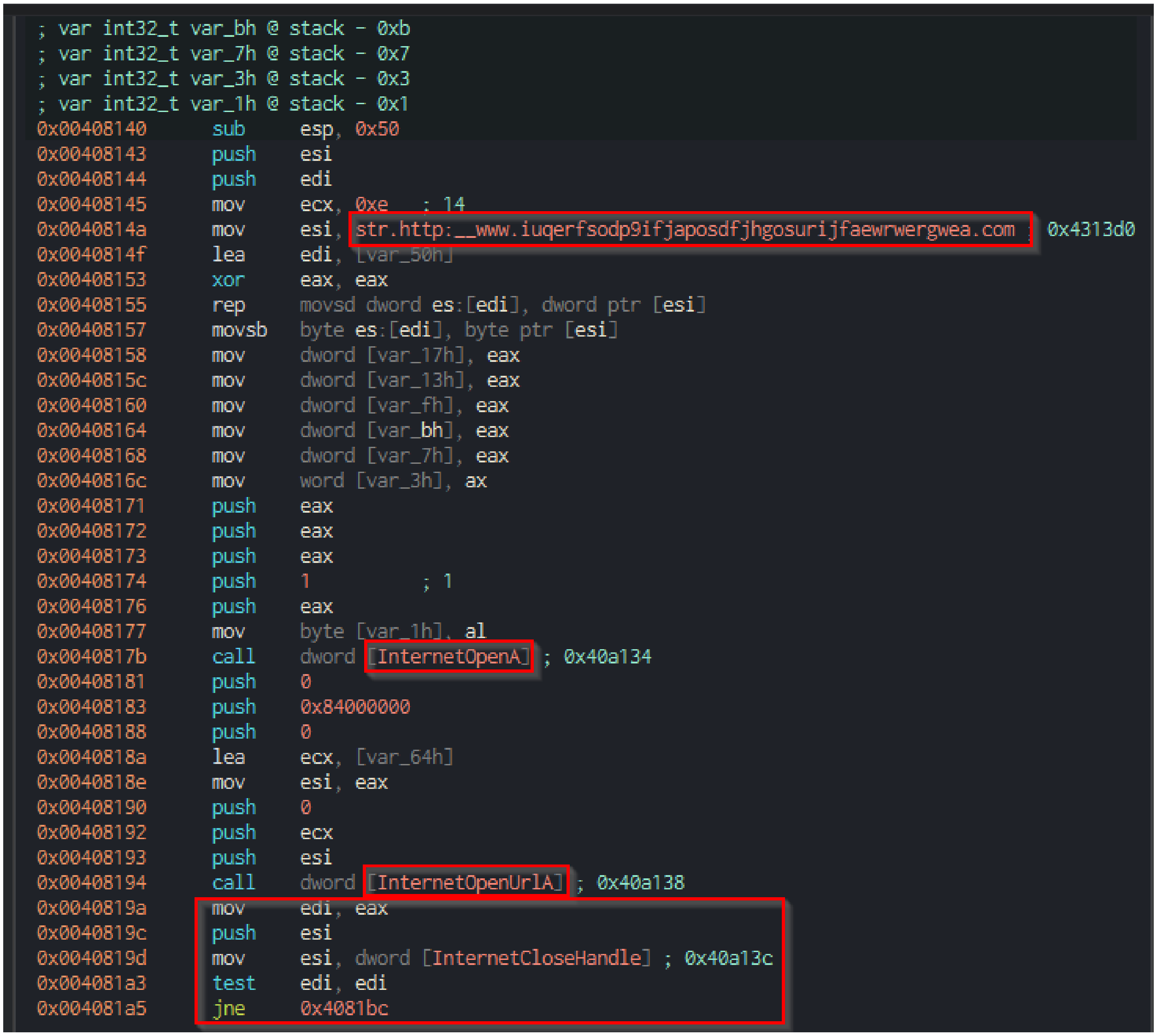
```
+------------------------------+------------------------------------------------------------------------------+
| MBC Objective                | MBC Behavior                                                                 |
|------------------------------+------------------------------------------------------------------------------|
| ANTI-BEHAVIORAL ANALYSIS     | Debugger Detection::Timing/Delay Check GetTickCount [B0001.032]              |
|                              | Debugger Detection::Timing/Delay Check QueryPerformanceCounter [B0001.033]   |
|                              | Execution Guardrails::Runs as Service [E1480.m07]                            |
| ANTI-STATIC ANALYSIS         | Disassembler Evasion::Argument Obfuscation [B0012.001]                       |
| COMMAND AND CONTROL          | C2 Communication::Receive Data [B0030.002]                                   |
|                              | C2 Communication::Send Data [B0030.001]                                      |
| COMMUNICATION                | HTTP Communication::Create Request [C0002.012]                               |
|                              | HTTP Communication::Open URL [C0002.004]                                     |
|                              | Socket Communication::Connect Socket [C0001.004]                             |
|                              | Socket Communication::Create TCP Socket [C0001.011]                          |
|                              | Socket Communication::Create UDP Socket [C0001.010]                          |
|                              | Socket Communication::Get Socket Status [C0001.012]                          |
|                              | Socket Communication::Initialize Winsock Library [C0001.009]                 |
|                              | Socket Communication::Receive Data [C0001.006]                               |
|                              | Socket Communication::Send Data [C0001.007]                                  |
|                              | Socket Communication::Set Socket Config [C0001.001]                          |
|                              | Socket Communication::TCP Client [C0001.008]                                 |
| CRYPTOGRAPHY                 | Generate Pseudo-random Sequence::Use API [C0021.003]                         |
| DATA                         | Compression Library [C0060]                                                  |
| EXECUTION                    | Install Additional Program [B0023]                                           |
| FILE SYSTEM                  | Read File [C0051]                                                            |
| PROCESS                      | Create Thread [C0038]                                                        |
|                              | Terminate Process [C0018]                                                    |
|                              | Terminate Thread [C0039]                                                     |
+------------------------------+------------------------------------------------------------------------------+
```



```
+------------------------------------------------------+--------------------------------------------------+
| CAPABILITY                                           | NAMESPACE                                        |
|------------------------------------------------------+--------------------------------------------------|
| check for time delay via GetTickCount                | anti-analysis/anti-debugging/debugger-detection  |
| check for time delay via QueryPerformanceCounter     | anti-analysis/anti-debugging/debugger-detection  |
| contain obfuscated stackstrings                      | anti-analysis/obfuscation/string/stackstring     |
| receive data (5 matches)                             | communication                                    |
| send data (5 matches)                                | communication                                    |
| connect to URL                                       | communication/http/client                        |
| get socket status                                    | communication/socket                             |
| initialize Winsock library                           | communication/socket                             |
| set socket configuration                             | communication/socket                             |
| create UDP socket (4 matches)                        | communication/socket/udp/send                    |
| act as TCP client                                    | communication/tcp/client                         |
| generate random numbers via WinAPI                   | data-manipulation/prng                           |
| contain a resource (.rsrc) section                   | executable/pe/section/rsrc                       |
| extract resource via kernel32 functions              | executable/resource                              |
| contain an embedded PE file                          | executable/subfile/pe                            |
| get file size                                        | host-interaction/file-system/meta                |
| move file                                            | host-interaction/file-system/move                |
| read file                                            | host-interaction/file-system/read                |
| get number of processors                             | host-interaction/hardware/cpu                    |
| get networking interfaces                            | host-interaction/network/interface               |
| terminate process                                    | host-interaction/process/terminate               |
| run as service                                       | host-interaction/service                         |
| create service                                       | host-interaction/service/create                  |
| modify service                                       | host-interaction/service/modify                  |
| start service                                        | host-interaction/service/start                   |
| create thread (4 matches)                            | host-interaction/thread/create                   |
| terminate thread                                     | host-interaction/thread/terminate                |
| link function at runtime                             | linking/runtime-linking                          |
| linked against ZLIB                                  | linking/static/zlib                              |
| inspect section memory permissions                   | load-code/pe                                     |
| parse PE exports                                     | load-code/pe                                     |
| parse PE header                                      | load-code/pe                                     |
| persist via Windows service                          | persistence/service                              |
+------------------------------------------------------+--------------------------------------------------+
```

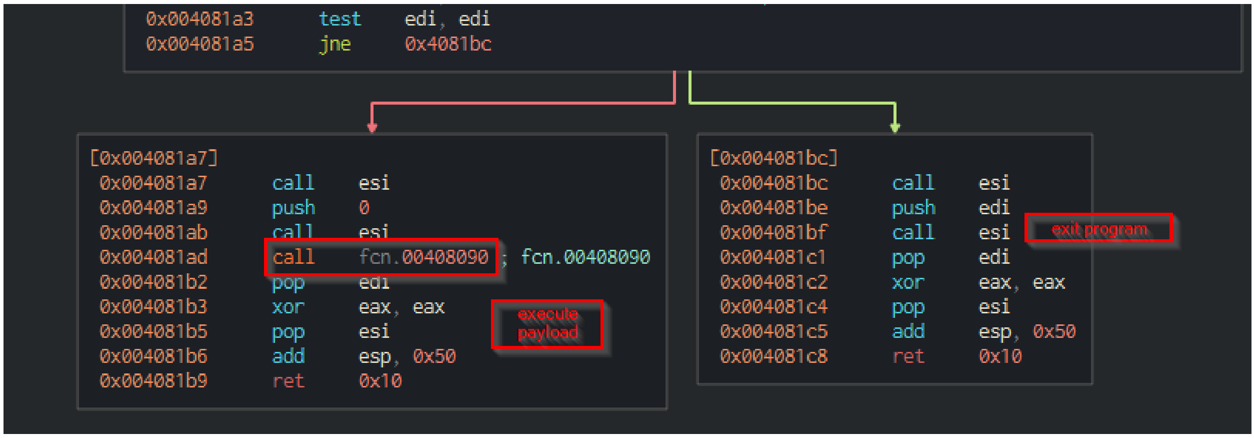
Without executing the file, we use cutter to access an attempted decompiled version of the source code.

Main function



```
; var int32_t var_bh @ stack - 0xb
; var int32_t var_7h @ stack - 0x7
; var int32_t var_3h @ stack - 0x3
; var int32_t var_1h @ stack - 0x1
0x00408140      sub     esp, 0x50
0x00408143      push    esi
0x00408144      push    edi
0x00408145      mov     ecx, 0xe    ; 14
0x0040814a      mov     esi, str.http:__www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com ; 0x4313d0
0x0040814f      lea     edi, [var_50h]
0x00408153      xor     eax, eax
0x00408155      rep     movsd dword es:[edi], dword ptr [esi]
0x00408157      movsb   byte es:[edi], byte ptr [esi]
0x00408158      mov     dword [var_17h], eax
0x0040815c      mov     dword [var_13h], eax
0x00408160      mov     dword [var_fh], eax
0x00408164      mov     dword [var_bh], eax
0x00408168      mov     dword [var_7h], eax
0x0040816c      mov     word [var_3h], ax
0x00408171      push    eax
0x00408172      push    eax
0x00408173      push    eax
0x00408174      push    1           ; 1
0x00408176      push    eax
0x00408177      mov     byte [var_1h], al
0x0040817b      call    dword [InternetOpenA] ; 0x40a134
0x00408181      push    0
0x00408183      push    0x84000000
0x00408188      push    0
0x0040818a      lea     ecx, [var_64h]
0x0040818e      mov     esi, eax
0x00408190      push    0
0x00408192      push    ecx
0x00408193      push    esi
0x00408194      call    dword [InternetOpenUrlA] ; 0x40a138
0x0040819a      mov     edi, eax
0x0040819c      push    esi
0x0040819d      mov     esi, dword [InternetCloseHandle] ; 0x40a13c
0x004081a3      test    edi, edi
0x004081a5      jne     0x4081bc
```

Program stores the URL and attempts an internet connection through Windows API calls. Depending on the result it jumps to different sections of the code.



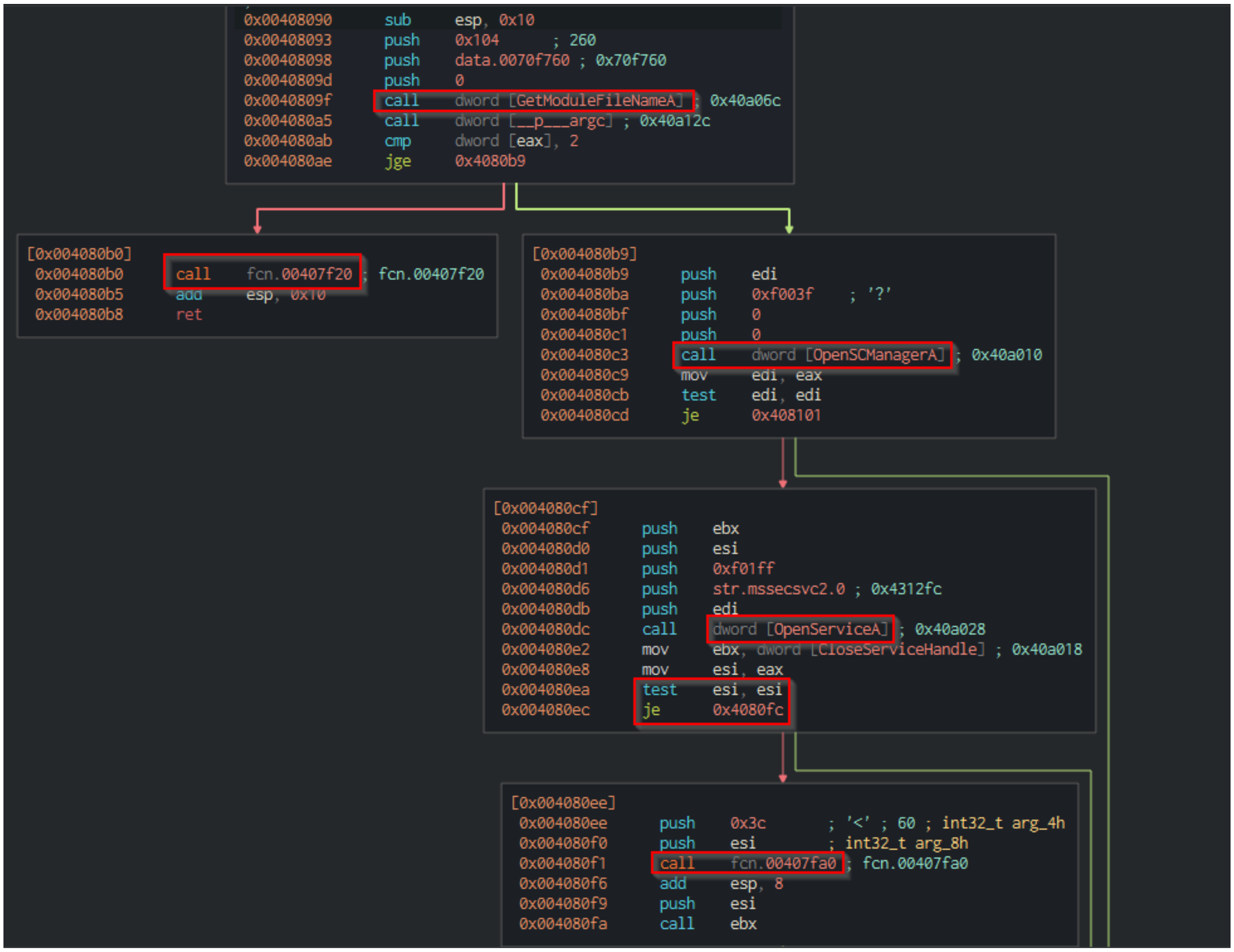
```
0x004081a3      test    edi, edi
0x004081a5      jne     0x4081bc
```

```
[0x004081a7]
 0x004081a7      call    esi
 0x004081a9      push    0
 0x004081ab      call    esi
 0x004081ad      call    fcn.00408090 ; fcn.00408090
 0x004081b2      pop     edi
 0x004081b3      xor     eax, eax
 0x004081b5      pop     esi
 0x004081b6      add     esp, 0x50
 0x004081b9      ret     0x10
```

```
[0x004081bc]
 0x004081bc      call    esi
 0x004081be      push    edi
 0x004081bf      call    esi
 0x004081c1      pop     edi
 0x004081c2      xor     eax, eax
 0x004081c4      pop     esi
 0x004081c5      add     esp, 0x50
 0x004081c8      ret     0x10
```

The left side shows the execution of the payload, calling on function allocated on 00408090. Otherwise, it will exit the program.

“Execute payload” (00408090)



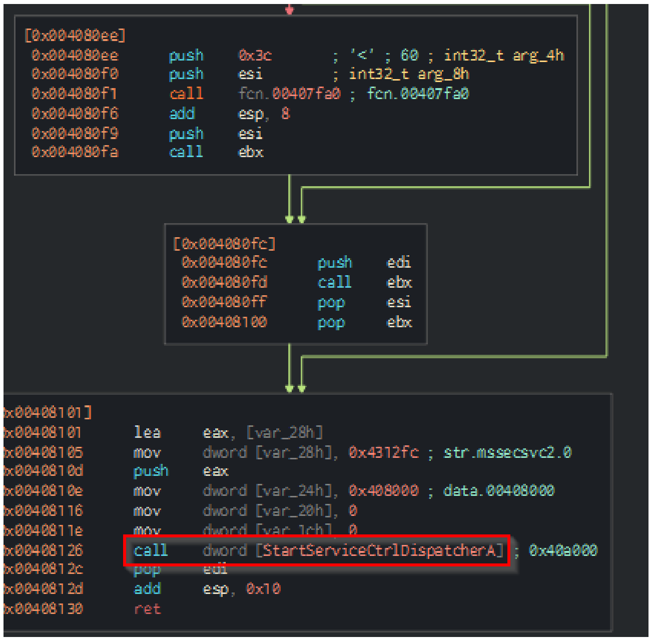
```
0x00408090      sub     esp, 0x10
0x00408093      push    0x104       ; 260
0x00408098      push    data.0070f760 ; 0x70f760
0x0040809d      push    0
0x0040809f      call    dword [GetModuleFileNameA] ; 0x40a06c
0x004080a5      call    dword [__p___argc] ; 0x40a12c
0x004080ab      cmp     dword [eax], 2
0x004080ae      jge     0x4080b9
```

```
[0x004080b0]
 0x004080b0      call    fcn.00407f20 ; fcn.00407f20
 0x004080b5      add     esp, 0x10
 0x004080b8      ret
```

```
[0x004080b9]
 0x004080b9      push    edi
 0x004080ba      push    0xf003f    ; '?'
 0x004080bf      push    0
 0x004080c1      push    0
 0x004080c3      call    dword [OpenSCManagerA] ; 0x40a010
 0x004080c9      mov     edi, eax
 0x004080cb      test    edi, edi
 0x004080cd      je      0x408101
```

```
[0x004080cf]
 0x004080cf      push    ebx
 0x004080d0      push    esi
 0x004080d1      push    0xf01ff
 0x004080d6      push    str.mssecsvc2.0 ; 0x4312fc
 0x004080db      push    edi
 0x004080dc      call    dword [OpenServiceA] ; 0x40a028
 0x004080e2      mov     ebx, dword [CloseServiceHandle] ; 0x40a018
 0x004080e8      mov     esi, eax
 0x004080ea      test    esi, esi
 0x004080ec      je      0x4080fc
```

```
[0x004080ee]
 0x004080ee      push    0x3c        ; '<' ; 60 ; int32_t arg_4h
 0x004080f0      push    esi         ; int32_t arg_8h
 0x004080f1      call    fcn.00407fa0 ; fcn.00407fa0
 0x004080f6      add     esp, 8
 0x004080f9      push    esi
 0x004080fa      call    ebx
```



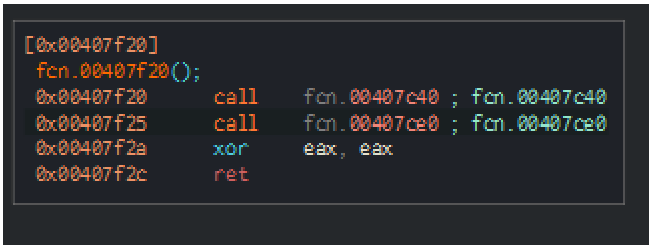
```
[0x004080ee]
 0x004080ee      push    0x3c        ; '<' ; 60 ; int32_t arg_4h
 0x004080f0      push    esi         ; int32_t arg_8h
 0x004080f1      call    fcn.00407fa0 ; fcn.00407fa0
 0x004080f6      add     esp, 8
 0x004080f9      push    esi
 0x004080fa      call    ebx

[0x004080fc]
 0x004080fc      push    edi
 0x004080fd      call    ebx
 0x004080ff      pop     esi
 0x00408100      pop     ebx

x00408101]
x00408101      lea     eax, [var_28h]
x00408105      mov     dword [var_28h], 0x4312fc ; str.mssecsvc2.0
x0040810d      push    eax
x0040810e      mov     dword [var_24h], 0x408000 ; data.00408000
x00408116      mov     dword [var_20h], 0
x0040811e      mov     dword [var_1ch], 0
x00408126      call    dword [StartServiceCtrlDispatcherA] ; 0x40a000
x0040812c      pop     edi
x0040812d      add     esp, 0x10
x00408130      ret
```

Calling the GetModuleFileNameA with an empty argument returns the path of the directory of the executable file. Then calls on the __p__argc function, which we currently don't know its purpose.

One flow of the program calls on the Service Control Manager and opens a service, then calls on function 00407fa0, finishing on a StartService.

The other flow calls on function 00407f20.

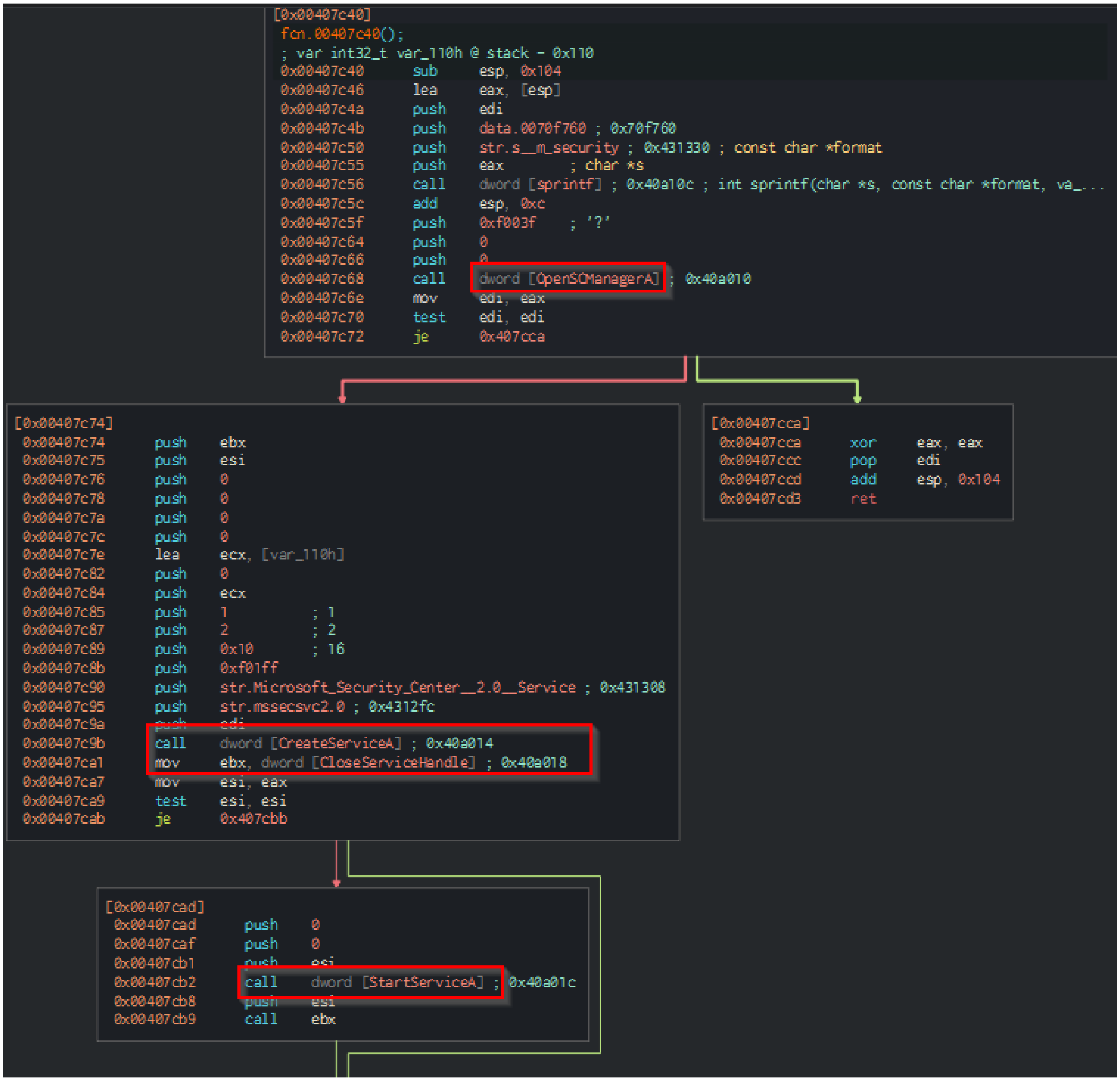
00407f20



```
[0x00407f20]
 fcn.00407f20();
 0x00407f20      call    fcn.00407c40 ; fcn.00407c40
 0x00407f25      call    fcn.00407ce0 ; fcn.00407ce0
 0x00407f2a      xor     eax, eax
 0x00407f2c      ret
```

Simply calls 2 functions

00407c40



```
[0x00407c40]
 fcn.00407c40();
 ; var int32_t var_110h @ stack - 0x110
 0x00407c40      sub     esp, 0x104
 0x00407c46      lea     eax, [esp]
 0x00407c4a      push    edi
 0x00407c4b      push    data.0070f760 ; 0x70f760
 0x00407c50      push    str.s__m_security ; 0x431330 ; const char *format
 0x00407c55      push    eax         ; char *s
 0x00407c56      call    dword [sprintf] ; 0x40a10c ; int sprintf(char *s, const char *format, va_...
 0x00407c5c      add     esp, 0xc
 0x00407c5f      push    0xf003f     ; '?'
 0x00407c64      push    0
 0x00407c66      push    0
 0x00407c68      call    dword [OpenSCManagerA] ; 0x40a010
 0x00407c6e      mov     edi, eax
 0x00407c70      test    edi, edi
 0x00407c72      je      0x407cca
```

```
[0x00407c74]
 0x00407c74      push    ebx
 0x00407c75      push    esi
 0x00407c76      push    0
 0x00407c78      push    0
 0x00407c7a      push    0
 0x00407c7c      push    0
 0x00407c7e      lea     ecx, [var_110h]
 0x00407c82      push    0
 0x00407c84      push    ecx
 0x00407c85      push    1           ; 1
 0x00407c87      push    2           ; 2
 0x00407c89      push    0x10        ; 16
 0x00407c8b      push    0xf01ff
 0x00407c90      push    str.Microsoft_Security_Center__2.0__Service ; 0x431308
 0x00407c95      push    str.mssecsvc2.0 ; 0x4312fc
 0x00407c9a      push    edi
 0x00407c9b      call    dword [CreateServiceA] ; 0x40a014
 0x00407ca1      mov     ebx, dword [CloseServiceHandle] ; 0x40a018
 0x00407ca7      mov     esi, eax
 0x00407ca9      test    esi, esi
 0x00407cab      je      0x407cbb
```

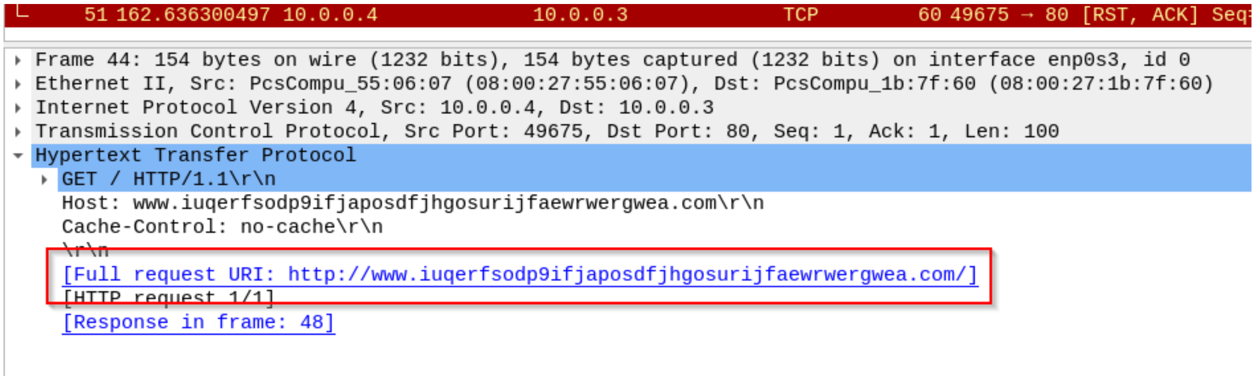
```
[0x00407cca]
 0x00407cca      xor     eax, eax
 0x00407ccc      pop     edi
 0x00407ccd      add     esp, 0x104
 0x00407cd3      ret
```

```
[0x00407cad]
 0x00407cad      push    0
 0x00407caf      push    0
 0x00407cb1      push    esi
 0x00407cb2      call    dword [StartServiceA] ; 0x40a01c
 0x00407cb8      push    esi
 0x00407cb9      call    ebx
```

First function Opens a service manager, creates, and starts a service

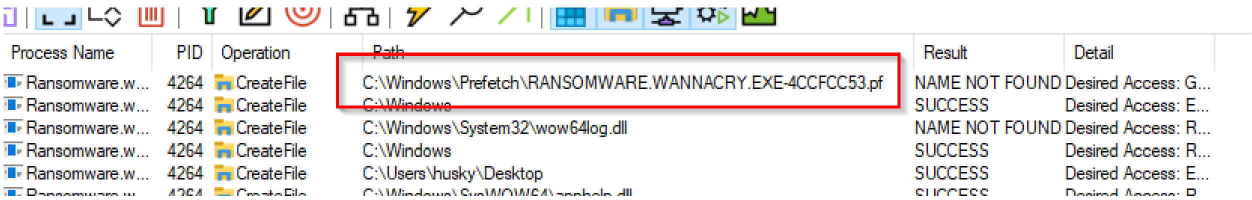
Dynamic Analysis

First, we execute the file with INetSim running on a REMnux machine with Wireshark analyzing the traffic. Nothing seems to happen, which means the part of the code seen whether or not to execute the payload depends on if the URL is reachable. If it is, it won't execute.



```
└    51 162.636300497 10.0.0.4            10.0.0.3          TCP         60 49675 → 80 [RST, ACK] Seq=

▸ Frame 44: 154 bytes on wire (1232 bits), 154 bytes captured (1232 bits) on interface enp0s3, id 0
▸ Ethernet II, Src: PcsCompu_55:06:07 (08:00:27:55:06:07), Dst: PcsCompu_1b:7f:60 (08:00:27:1b:7f:60)
▸ Internet Protocol Version 4, Src: 10.0.0.4, Dst: 10.0.0.3
▸ Transmission Control Protocol, Src Port: 49675, Dst Port: 80, Seq: 1, Ack: 1, Len: 100
▾ Hypertext Transfer Protocol
  ▸ GET / HTTP/1.1\r\n
    Host: www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com\r\n
    Cache-Control: no-cache\r\n
    \r\n
    [Full request URI: http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com/]
    [HTTP request 1/1]
    [Response in frame: 48]
```

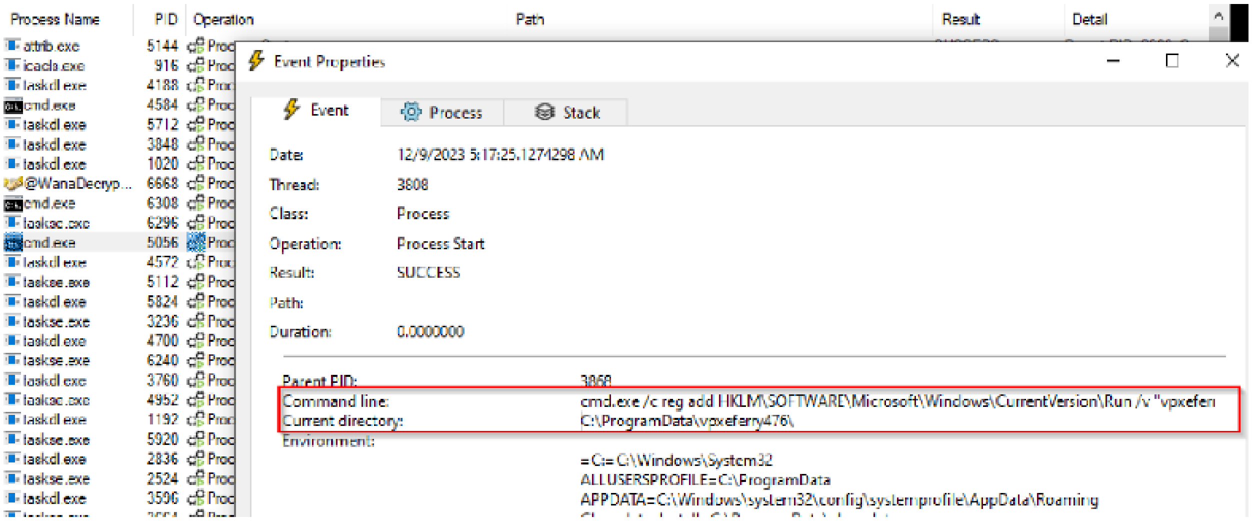
We see the HTP request for the URL seen on the Static Analysis.



Process Name	PID	Operation	Path	Result	Detail
Ransomware.w...	4264	CreateFile	C:\Windows\Prefetch\RANSOMWARE.WANNACRY.EXE-4CCFCC53.pf	NAME NOT FOUND	Desired Access: G...
Ransomware.w...	4264	CreateFile	C:\Windows	SUCCESS	Desired Access: E...
Ransomware.w...	4264	CreateFile	C:\Windows\System32\wow64log.dll	NAME NOT FOUND	Desired Access: R...
Ransomware.w...	4264	CreateFile	C:\Windows	SUCCESS	Desired Access: R...
Ransomware.w...	4264	CreateFile	C:\Users\husky\Desktop	SUCCESS	Desired Access: E...
Ransomware.w...	4264	CreateFile	C:\Windows\SysWOW64\apphelp.dll	SUCCESS	Desired Access: R

We see a PF file was created on path *C:\Windows\Prefetch\RANSOMWARE.WANNACRY.EXE-4CCFCC53.pf*

Files with the file extension .pf can only be launched by certain applications. The file extension refers to encrypted files.

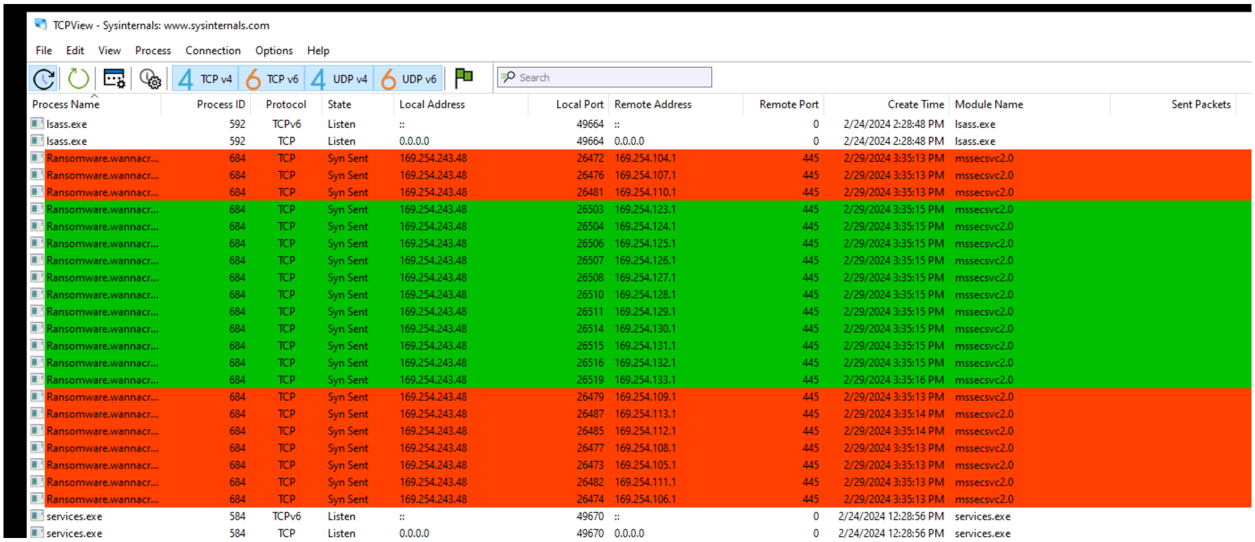


Process Name	PID	Operation
attrib.exe	5144	Proc
icacls.exe	916	Proc
taskdl.exe	4188	Proc
cmd.exe	4584	Proc
taskdl.exe	5712	Proc
taskdl.exe	3848	Proc
taskdl.exe	1020	Proc
@WanaDecryp...	6668	Proc
cmd.exe	6308	Proc
taskse.exe	6296	Proc
cmd.exe	5056	Proc
taskdl.exe	4572	Proc
taskse.exe	5112	Proc
taskdl.exe	5824	Proc
taskse.exe	3236	Proc
taskdl.exe	4700	Proc
taskse.exe	6240	Proc
taskdl.exe	3760	Proc
taskse.exe	4952	Proc
taskdl.exe	1192	Proc
taskse.exe	5920	Proc
taskdl.exe	2836	Proc
taskse.exe	2524	Proc
taskdl.exe	3596	Proc

Date: 12/9/2023 5:17:25.1274298 AM
Thread: 3808
Class: Process
Operation: Process Start
Result: SUCCESS
Path:
Duration: 0.0000000

Parent PID: 3868
Command line: cmd.exe /c reg add HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /v "vpxefen
Current directory: C:\ProgramData\vpxeferry476\
Environment:
=C:=C:\Windows\System32
ALLUSERSPROFILE=C:\ProgramData
APPDATA=C:\Windows\system32\config\systemprofile\AppData\Roaming

When shutting down InetSIM and re-detonating the malware, we see that it encrypts all available files, changes the Desktop Background and pops a GUI of the WannaCry de-cryptor, requesting a ransomware payment. Also, the malware makes multiple connection attempts throughout the network from port 684 to port 445.


TCPView - Sysinternals: www.sysinternals.com
File Edit View Process Connection Options Help
4 TCP v4
6 TCP v6
4 UDP v4
6 UDP v6
Search

Process Name	Process ID	Protocol	State	Local Address	Local Port	Remote Address	Remote Port	Create Time	Module Name	Sent Packets
lsass.exe	592	TCPv6	Listen	::	49664	::	0	2/24/2024 2:28:48 PM	lsass.exe	
lsass.exe	592	TCP	Listen	0.0.0.0	49664	0.0.0.0	0	2/24/2024 2:28:48 PM	lsass.exe	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26472	169.254.104.1	445	2/29/2024 3:35:13 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26476	169.254.107.1	445	2/29/2024 3:35:13 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26481	169.254.110.1	445	2/29/2024 3:35:13 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26503	169.254.123.1	445	2/29/2024 3:35:15 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26504	169.254.124.1	445	2/29/2024 3:35:15 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26506	169.254.125.1	445	2/29/2024 3:35:15 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26507	169.254.126.1	445	2/29/2024 3:35:15 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26508	169.254.127.1	445	2/29/2024 3:35:15 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26510	169.254.128.1	445	2/29/2024 3:35:15 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26511	169.254.129.1	445	2/29/2024 3:35:15 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26514	169.254.130.1	445	2/29/2024 3:35:15 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26515	169.254.131.1	445	2/29/2024 3:35:15 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26516	169.254.132.1	445	2/29/2024 3:35:15 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26519	169.254.133.1	445	2/29/2024 3:35:16 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26479	169.254.109.1	445	2/29/2024 3:35:13 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26487	169.254.113.1	445	2/29/2024 3:35:14 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26485	169.254.112.1	445	2/29/2024 3:35:14 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26477	169.254.108.1	445	2/29/2024 3:35:13 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26473	169.254.105.1	445	2/29/2024 3:35:13 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26482	169.254.111.1	445	2/29/2024 3:35:13 PM	mssecsvc2.0	
Ransomware.wannacr...	684	TCP	Syn Sent	169.254.243.48	26474	169.254.106.1	445	2/29/2024 3:35:13 PM	mssecsvc2.0	
services.exe	584	TCPv6	Listen	::	49670	::	0	2/24/2024 12:28:56 PM	services.exe	
services.exe	584	TCP	Listen	0.0.0.0	49670	0.0.0.0	0	2/24/2024 12:28:56 PM	services.exe	

Indicators of Compromise

Network Indicators

- URL hxxp://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
- Connections from port 684 to port 445

Host-based Indicators

- File *RANSOMWARE.WANNACRY.EXE-4CCFCC53.pf*
- Registry Key changed to autorun taskshe.exe
- Sha256 24D004A104D4D54034DBCFFC2A4B19A11F39008A575AA614EA04703480B1022C