Practical Malware Analysis & Triage
Malware Analysis Report

Wannacry - Ransomware

Ian 2023 | cosmin-stan | v1.0

Table of Contents

Executive Summary

SHA256 hash	24D004A104D4D54034DBCFFC2A4B19A11F39008A575AA614EA04703480B1022C

WannaCry ransomware is a crypto-ransomware worm, compiled in C++, that targets hosts with Windows OS. WannaCry was first identified in May 2017 and spread panic across corporate networks worldwide as it quickly infected more than 200,000 computers in 150 countries.

WannaCry takes advantage of the Eternal Blue vulnerability to spread throughout the network, it is still one of the most commonly attempted exploits against SMB, accounting for over 91.88% of the attacks on port 445 (the SMB port).

#YARA signature rules are attached in Appendix 1

High-Level Technical Summary

WannaCry consists of a main payload that attempts to reach to a URL. If the connection to the URL is successfully, it unpacks its next stage files into *C:\ProgramData\[RANDOM STRING].*

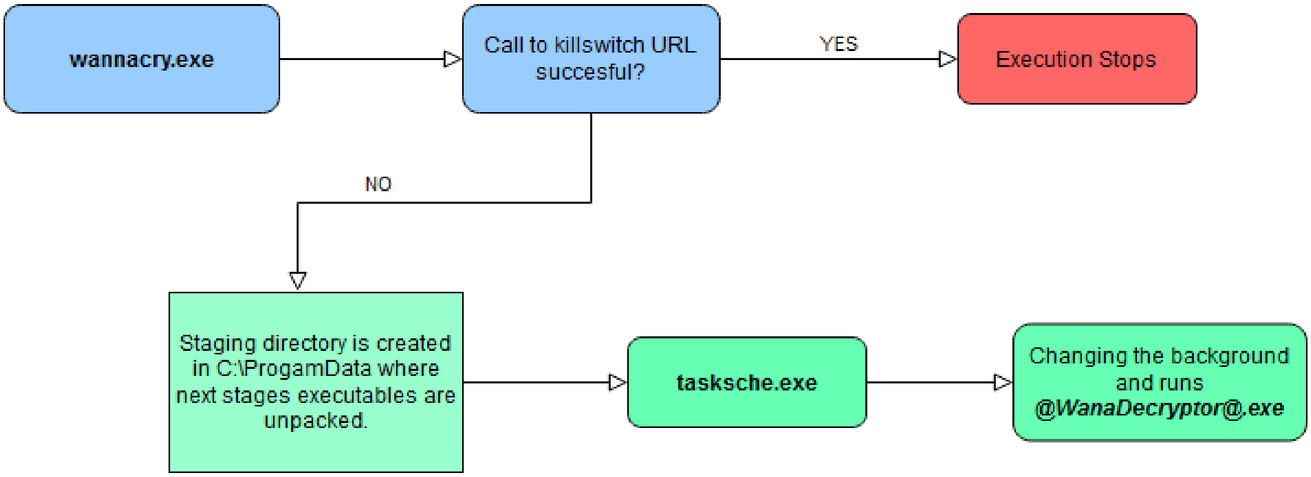

wannacry.exe
Call to killswitch URL succesful?
YES
Execution Stops
NO
Staging directory is created in C:\ProgamData where next stages executables are unpacked.
taksche.exe
Changing the background and runs @WanaDecryptor@.exe

Fig 1: Execution of WannaCry - Flowchart

Malware Composition

WannaCry consists of the following components:

File Name	SHA256 Hash
ransomware.wannacry.exe	24D004A104D4D54034DBCFFC2A4B19A11F39008A575AA614EA04703480B1022C
tasksche.exe	ED01EBFBC9EB5BBEA545AF4D01BF5F1071661840480439C6E5BABE8E080E41AA

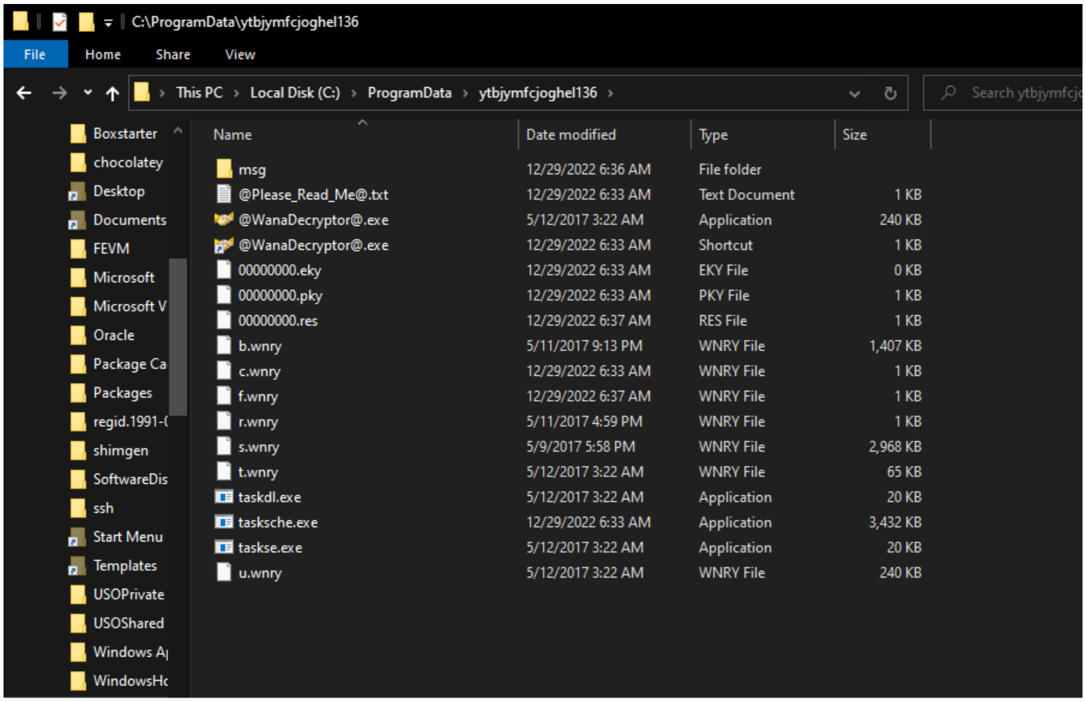
ransomware.wannacry.exe.exe

The initial executable succeeds if the call domain is not reachable.

tasksche.exe

The second stage executable is unpacked from the initial WannaCry executable and conducts most of the malicious operations on the host, encrypting the files.

Additional files:



C:\ProgramData\ytbjymfcjoghel136

Name	Date modified	Type	Size
msg	12/29/2022 6:36 AM	File folder	
@Please_Read_Me@.txt	12/29/2022 6:33 AM	Text Document	1 KB
@WanaDecryptor@.exe	5/12/2017 3:22 AM	Application	240 KB
@WanaDecryptor@.exe	12/29/2022 6:33 AM	Shortcut	1 KB
00000000.eky	12/29/2022 6:33 AM	EKY File	0 KB
00000000.pky	12/29/2022 6:33 AM	PKY File	1 KB
00000000.res	12/29/2022 6:37 AM	RES File	1 KB
b.wnry	5/11/2017 9:13 PM	WNRY File	1,407 KB
c.wnry	12/29/2022 6:33 AM	WNRY File	1 KB
f.wnry	12/29/2022 6:37 AM	WNRY File	1 KB
r.wnry	5/11/2017 4:59 PM	WNRY File	1 KB
s.wnry	5/9/2017 5:58 PM	WNRY File	2,968 KB
t.wnry	5/12/2017 3:22 AM	WNRY File	65 KB
taskdl.exe	5/12/2017 3:22 AM	Application	20 KB
tasksche.exe	12/29/2022 6:33 AM	Application	3,432 KB
taskse.exe	5/12/2017 3:22 AM	Application	20 KB
u.wnry	5/12/2017 3:22 AM	WNRY File	240 KB

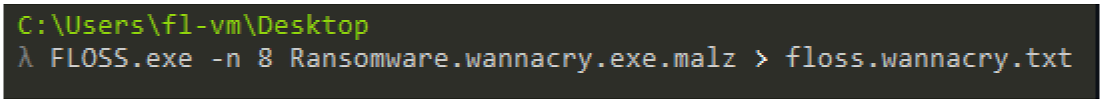
Fig 2: Staging directory for the WannaCry

Basic Static Analysis

{Screenshots and description about basic static artifacts and methods}

We used FLOSS to extract the PE strings and dump them into a txt file:

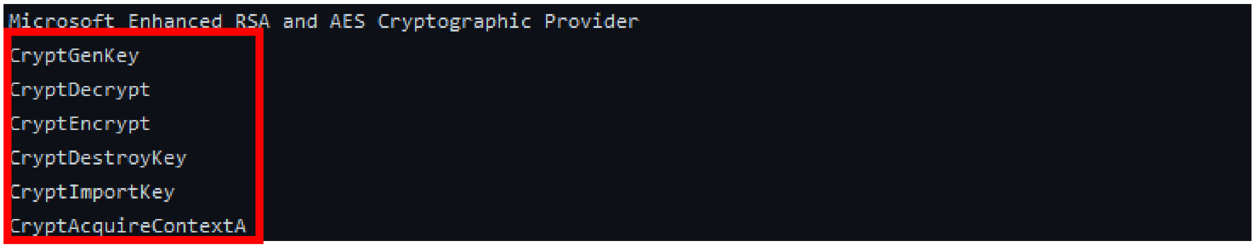
```
floss.exe -n 8 Ransomware.wannacry.exe.malz > wannacry_strings.txt
```



```
C:\Users\fl-vm\Desktop
λ FLOSS.exe -n 8 Ransomware.wannacry.exe.malz > floss.wannacry.txt
```

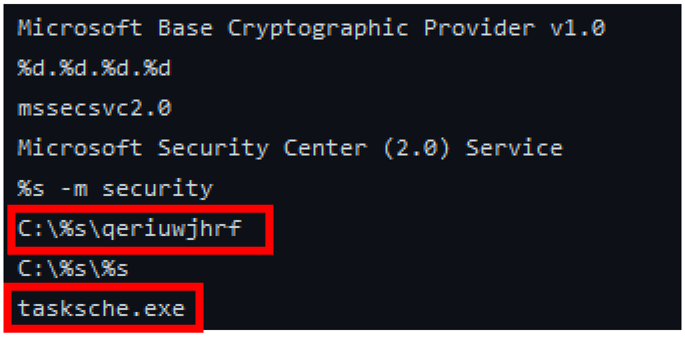
Suspicious strings:

Encryption imports:



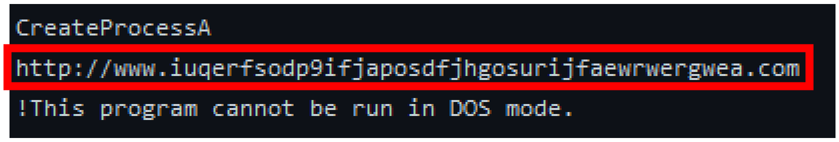
```
Microsoft Enhanced RSA and AES Cryptographic Provider
CryptGenKey
CryptDecrypt
CryptEncrypt
CryptDestroyKey
CryptImportKey
CryptAcquireContextA
```

Suspicious file path and possible second-stage executable



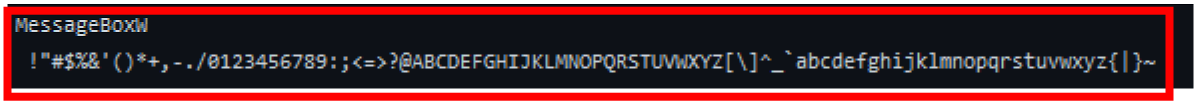
```
Microsoft Base Cryptographic Provider v1.0
%d.%d.%d.%d
mssecsvc2.0
Microsoft Security Center (2.0) Service
%s -m security
C:\%s\qeriuwjhrf
C:\%s\%s
tasksche.exe
```

Call back domain



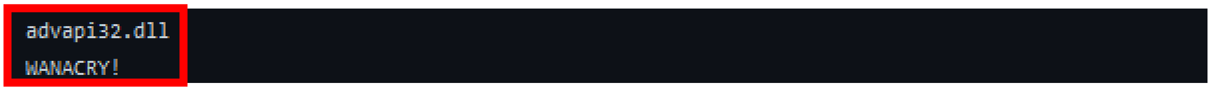
```
CreateProcessA
http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
!This program cannot be run in DOS mode.
```

Suspicious message box



```
MessageBoxW
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~
```

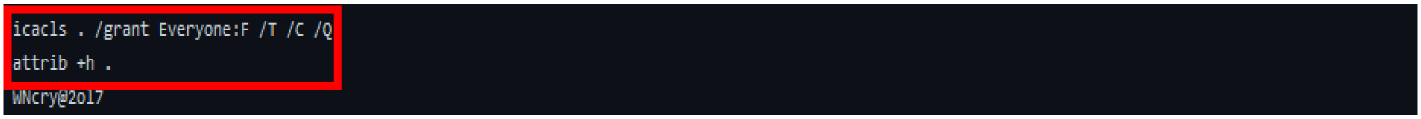
Suspicious string and dll



```
advapi32.dll
WANACRY!
```

Advapi32. dll is a part of the advanced API services library. It provides access to advanced functionality that comes in addition to the kernel. It is responsible for things like the Windows registry, restarting and shutting down the system, starting/stopping and creating Windows services, and managing user accounts!

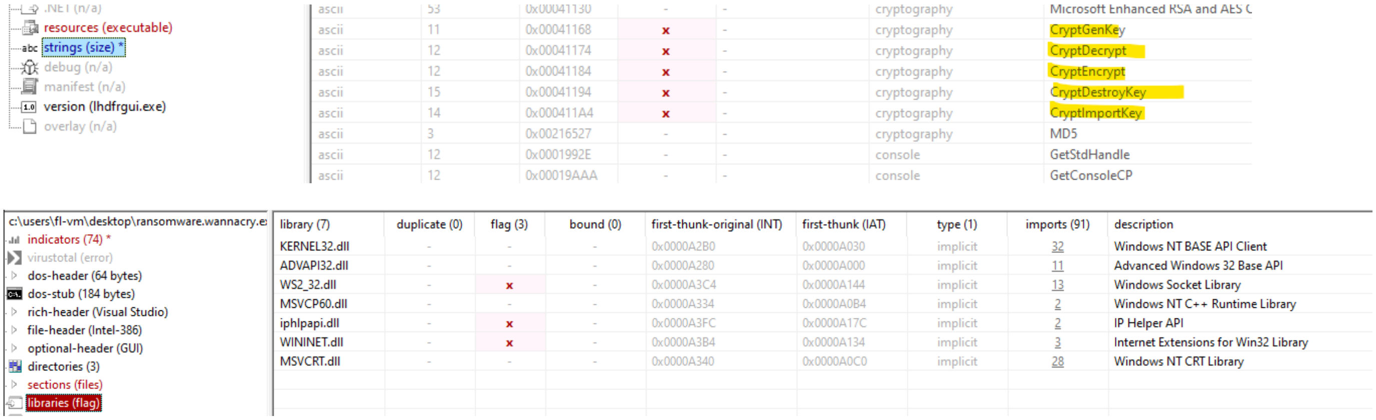
Suspicious Windows utility tool



```
icacls . /grant Everyone:F /T /C /Q
attrib +h .
WNcry@2o17
```

icacls is a Windows command-line utility that IT admins can use to change access control lists on files and folders.

I also used PEStudio to highlight some IOCs:



ascii	53	0x00041130	-	-	cryptography	Microsoft Enhanced RSA and AES C
ascii	11	0x00041168	x	-	cryptography	CryptGenKey
ascii	12	0x00041174	x	-	cryptography	CryptDecrypt
ascii	12	0x00041184	x	-	cryptography	CryptEncrypt
ascii	15	0x00041194	x	-	cryptography	CryptDestroyKey
ascii	14	0x000411A4	x	-	cryptography	CryptImportKey
ascii	3	0x00216527	-	-	cryptography	MD5
ascii	12	0x0001992E	-	-	console	GetStdHandle
ascii	12	0x00019AAA	-	-	console	GetConsoleCP

library (7)	duplicate (0)	flag (3)	bound (0)	first-thunk-original (INT)	first-thunk (IAT)	type (1)	imports (91)	description
KERNEL32.dll	-	-	-	0x0000A2B0	0x0000A030	implicit	32	Windows NT BASE API Client
ADVAPI32.dll	-	-	-	0x0000A280	0x0000A000	implicit	11	Advanced Windows 32 Base API
WS2_32.dll	-	x	-	0x0000A3C4	0x0000A144	implicit	13	Windows Socket Library
MSVCP60.dll	-	-	-	0x0000A334	0x0000A0B4	implicit	2	Windows NT C++ Runtime Library
iphlpapi.dll	-	x	-	0x0000A3FC	0x0000A17C	implicit	2	IP Helper API
WININET.dll	-	x	-	0x0000A3B4	0x0000A134	implicit	3	Internet Extensions for Win32 Library
MSVCRT.dll	-	-	-	0x0000A340	0x0000A0C0	implicit	28	Windows NT CRT Library

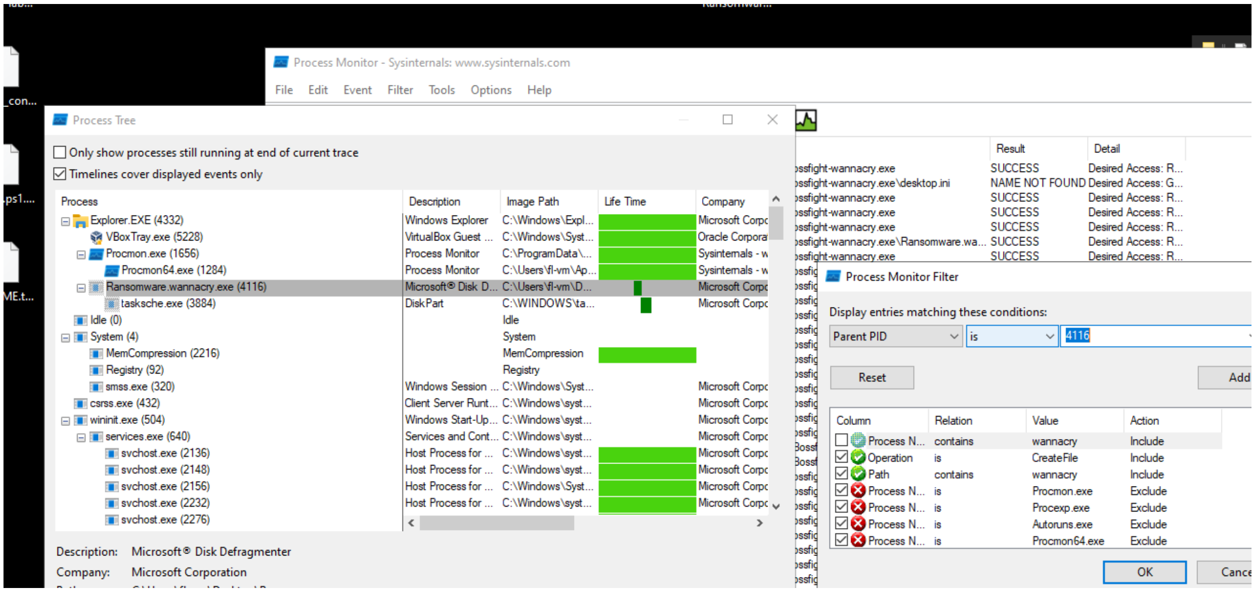
Fig 3: PEView

Basic Dynamic Analysis

{Screenshots and description about basic dynamic artifacts and methods}

I used Procmon to identify host indicators:

Procmon Filters:



Process Monitor - Sysinternals: www.sysinternals.com
File Edit Event Filter Tools Options Help
Process Tree

☐ Only show processes still running at end of current trace
☑ Timelines cover displayed events only

Process	Description	Image Path	Life Time	Company
Explorer.EXE (4332)	Windows Explorer	C:\Windows\Expl...	🟩	Microsoft Corpc
VBoxTray.exe (5228)	VirtualBox Guest ...	C:\Windows\Syst...	🟩	Oracle Corporat
Procmon.exe (1656)	Process Monitor	C:\ProgramData\...	🟩	Sysinternals - w
Procmon64.exe (1284)	Process Monitor	C:\Users\fl-vm\Ap...	🟩	Sysinternals - w
Ransomware.wannacry.exe (4116)	Microsoft® Disk D...	C:\Users\fl-vm\D...	🟩	Microsoft Corpc
taskche.exe (3884)	DiskPart	C:\WINDOWS\ta...	🟩	Microsoft Corpc
Idle (0)		Idle		
System (4)		System		
MemCompression (2216)		MemCompression	🟩	
Registry (92)		Registry		
smss.exe (320)	Windows Session ...	C:\Windows\Syst...		Microsoft Corpc
csrss.exe (432)	Client Server Runt...	C:\Windows\syst...		Microsoft Corpc
wininit.exe (504)	Windows Start-Up...	C:\Windows\syst...		Microsoft Corpc
services.exe (640)	Services and Cont...	C:\Windows\syst...		Microsoft Corpc
svchost.exe (2136)	Host Process for ...	C:\Windows\syst...	🟩	Microsoft Corpc
svchost.exe (2148)	Host Process for ...	C:\Windows\syst...	🟩	Microsoft Corpc
svchost.exe (2156)	Host Process for ...	C:\Windows\Syst...	🟩	Microsoft Corpc
svchost.exe (2232)	Host Process for ...	C:\Windows\syst...	🟩	Microsoft Corpc
svchost.exe (2276)				

Description: Microsoft® Disk Defragmenter
Company: Microsoft Corporation

	Result	Detail
ossfight-wannacry.exe	SUCCESS	Desired Access: R...
ossfight-wannacry.exe\desktop.ini	NAME NOT FOUND	Desired Access: G...
ossfight-wannacry.exe	SUCCESS	Desired Access: R...
ossfight-wannacry.exe	SUCCESS	Desired Access: R...
ossfight-wannacry.exe	SUCCESS	Desired Access: R...
ossfight-wannacry.exe\Ransomware.wa...	SUCCESS	Desired Access: R...
ossfight-wannacry.exe	SUCCESS	Desired Access: R...

Display entries matching these conditions:
Parent PID is 4116

	Column	Relation	Value	Action
☐	Process N...	contains	wannacry	Include
☑	Operation	is	CreateFile	Include
☑	Path	contains	wannacry	Include
☑	Process N...	is	Procmon.exe	Exclude
☑	Process N...	is	Procexp.exe	Exclude
☑	Process N...	is	Autoruns.exe	Exclude
☑	Process N...	is	Procmon64.exe	Exclude

Fig 4: Procmon Filters

We executed our sample (Ransomware.wannacry.exe.malz) on the FLARE-VM.

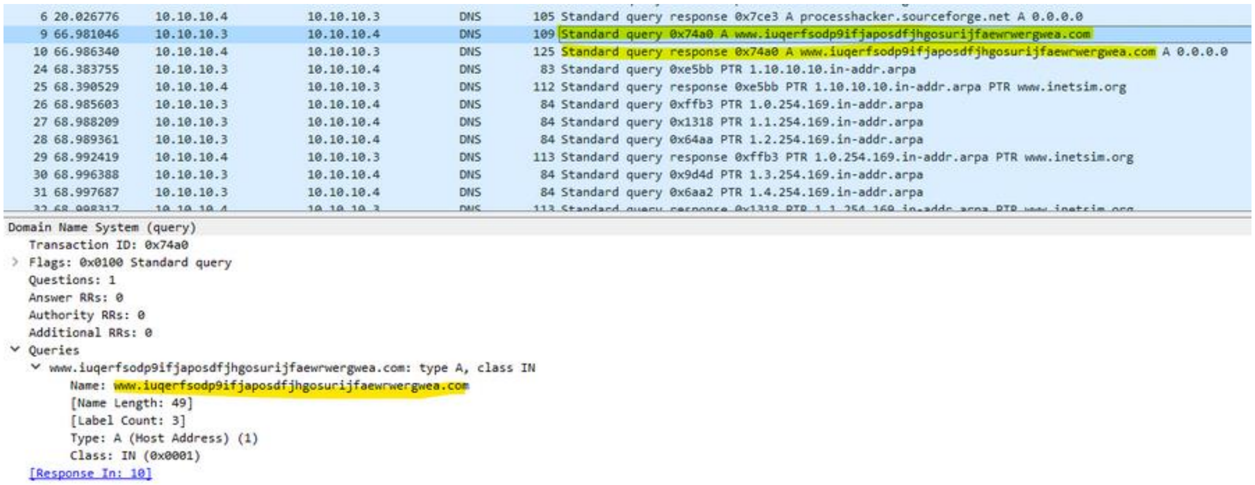
The malware starts by attempting to connect to the following domain with InternetOpenUrl:

```
www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
```

NOTE: If this succeeds, the malware immediately exits.

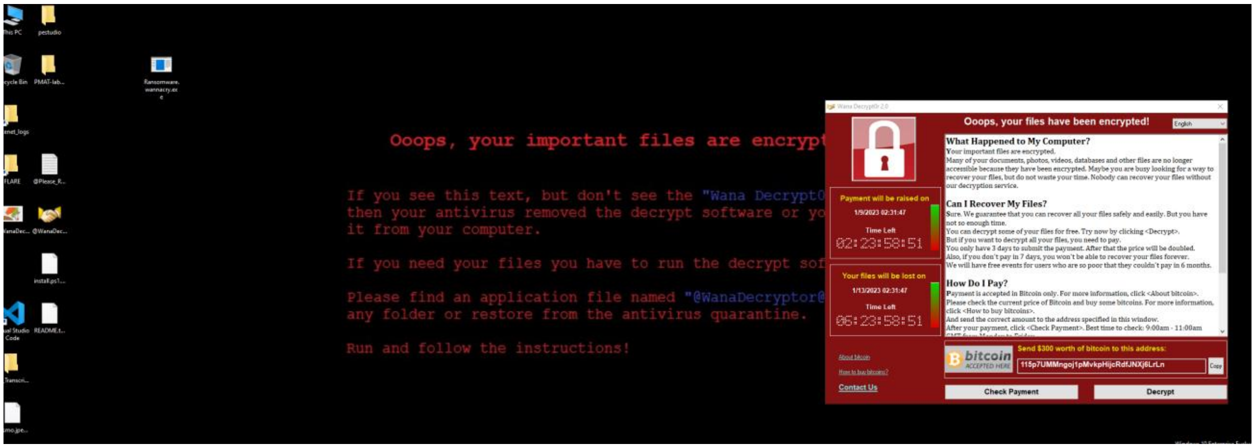
If the connection fails, the malware checks the number of arguments passed to the program. If zero, the malware continues with installation; otherwise it enters service mode.

Wireshark:



```
 6 20.026776   10.10.10.4   10.10.10.3   DNS   105 Standard query response 0x7ce3 A processhacker.sourceforge.net A 0.0.0.0
 9 66.981046   10.10.10.3   10.10.10.4   DNS   109 Standard query 0x74a0 A www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
10 66.986340   10.10.10.4   10.10.10.3   DNS   125 Standard query response 0x74a0 A www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com A 0.0.0.0
24 68.383755   10.10.10.3   10.10.10.4   DNS    83 Standard query 0xe5bb PTR 1.10.10.10.in-addr.arpa
25 68.390529   10.10.10.4   10.10.10.3   DNS   112 Standard query response 0xe5bb PTR 1.10.10.10.in-addr.arpa PTR www.inetsim.org
26 68.985603   10.10.10.3   10.10.10.4   DNS    84 Standard query 0xffb3 PTR 1.0.254.169.in-addr.arpa
27 68.988209   10.10.10.3   10.10.10.4   DNS    84 Standard query 0x1318 PTR 1.1.254.169.in-addr.arpa
28 68.989361   10.10.10.3   10.10.10.4   DNS    84 Standard query 0x64aa PTR 1.2.254.169.in-addr.arpa
29 68.992419   10.10.10.4   10.10.10.3   DNS   113 Standard query response 0xffb3 PTR 1.0.254.169.in-addr.arpa PTR www.inetsim.org
30 68.996388   10.10.10.3   10.10.10.4   DNS    84 Standard query 0x9d4d PTR 1.3.254.169.in-addr.arpa
31 68.997687   10.10.10.3   10.10.10.4   DNS    84 Standard query 0x6aa2 PTR 1.4.254.169.in-addr.arpa
32 68.998317   10.10.10.4   10.10.10.3   DNS   113 Standard query response 0x1318 PTR 1.1.254.169.in-addr.arpa PTR www.inetsim.org

Domain Name System (query)
   Transaction ID: 0x74a0
 > Flags: 0x0100 Standard query
   Questions: 1
   Answer RRs: 0
   Authority RRs: 0
   Additional RRs: 0
 v Queries
   v www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com: type A, class IN
        Name: www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
        [Name Length: 49]
        [Label Count: 3]
        Type: A (Host Address) (1)
        Class: IN (0x0001)
   [Response In: 10]
```

Fig 5: Wireshark logs for the Call Domain



Ooops, your important files are encrypt

If you see this text, but don't see the "Wana Decrypt0
then your antivirus removed the decrypt software or yo
it from your computer.

If you need your files you have to run the decrypt sof

Please find an application file named "@WanaDecryptor@
any folder or restore from the antivirus quarantine.

Run and follow the instructions!

Ooops, your files have been encrypted!

Payment will be raised on
1/9/2023 02:31:47
Time Left
02:23:58:51

Your files will be lost on
1/13/2023 02:31:47
Time Left
06:23:58:51

What Happened to My Computer?

Your important files are encrypted.
Many of your documents, photos, videos, databases and other files are no longer accessible because they have been encrypted. Maybe you are busy looking for a way to recover your files, but do not waste your time. Nobody can recover your files without our decryption service.

Can I Recover My Files?

Sure. We guarantee that you can recover all your files safely and easily. But you have not so enough time.
You can decrypt some of your files for free. Try now by clicking <Decrypt>.
But if you want to decrypt all your files, you need to pay.
You only have 3 days to submit the payment. After that the price will be doubled.
Also, if you don't pay in 7 days, you won't be able to recover your files forever.
We will have free events for users who are so poor that they couldn't pay in 6 months.

How Do I Pay?

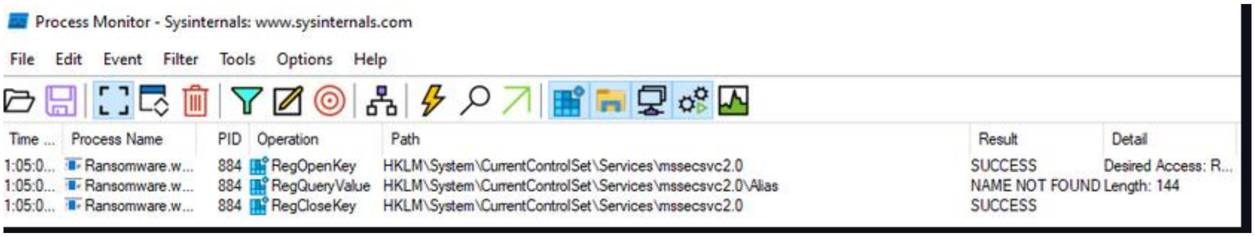
Payment is accepted in Bitcoin only. For more information, click <About bitcoin>.
Please check the current price of Bitcoin and buy some bitcoins. For more information, click <How to buy bitcoins>.
And send the correct amount to the address specified in this window.
After your payment, click <Check Payment>. Best time to check: 9:00am - 11:00am

Send $300 worth of bitcoin to this address:

115p7UMMngoj1pMvkpHijcRdfJNXj6LrLn

Fig 6: Desktop view after the initial detonation

The malware continues by creating a service named mssecsvc2.0.



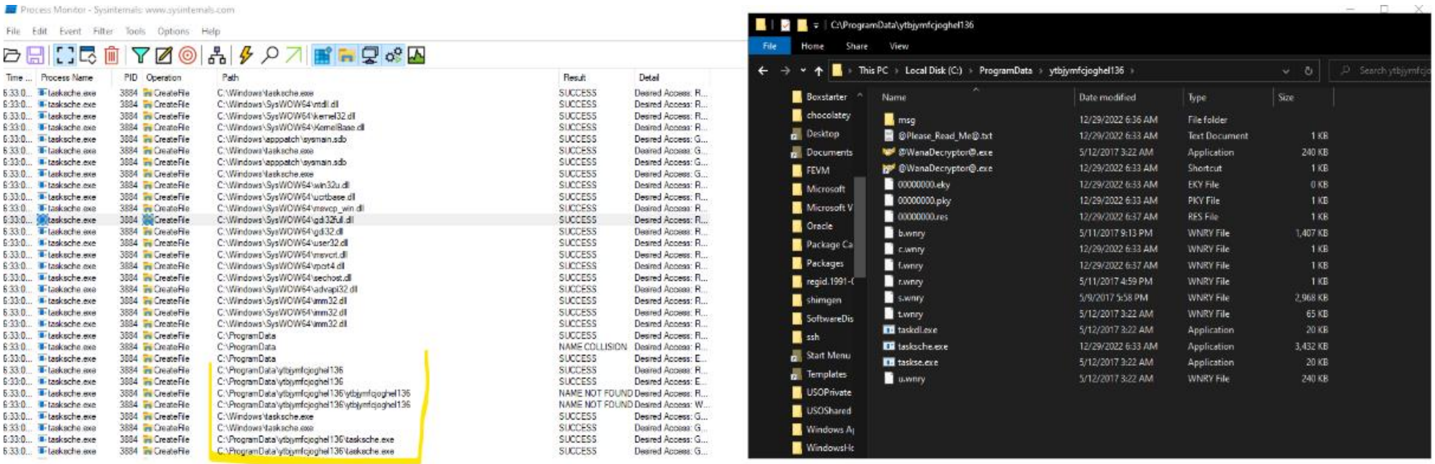
```
Time ...  Process Name      PID  Operation       Path                                                       Result           Detail
1:05:0... Ransomware.w...   884  RegOpenKey      HKLM\System\CurrentControlSet\Services\mssecsvc2.0         SUCCESS          Desired Access: R...
1:05:0... Ransomware.w...   884  RegQueryValue   HKLM\System\CurrentControlSet\Services\mssecsvc2.0\Alias   NAME NOT FOUND   Length: 144
1:05:0... Ransomware.w...   884  RegCloseKey     HKLM\System\CurrentControlSet\Services\mssecsvc2.0         SUCCESS
```

Fig 7: Procmon Results

Once created, the malware starts the service.

The malware writes the file C:\Windows\tasksche.exe. The malware executes and then writes a new hidden folder:

C:\ProgramData\ytbjymfcjoghel136\tasksche.exe



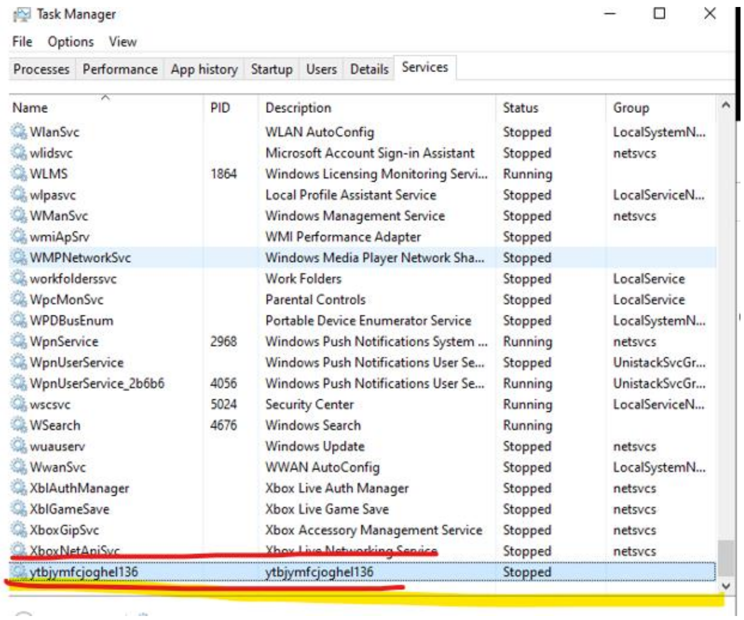
```
Process Monitor - Sysinternals: www.sysinternals.com
File  Edit  Event  Filter  Tools  Options  Help

Time ...  Process Name   PID   Operation   Path                                                      Result            Detail
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\tasksche.exe                                   SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\ntdll.dll                             SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\kernel32.dll                          SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\KernelBase.dll                        SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\apppatch\sysmain.sdb                           SUCCESS           Desired Access: G...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\tasksche.exe                                   SUCCESS           Desired Access: G...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\apppatch\sysmain.sdb                           SUCCESS           Desired Access: G...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\tasksche.exe                                   SUCCESS           Desired Access: G...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\win32u.dll                            SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\ucrtbase.dll                          SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\msvcp_win.dll                         SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\gdi32full.dll                         SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\gdi32.dll                             SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\user32.dll                            SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\msvcrt.dll                            SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\rpcrt4.dll                            SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\sechost.dll                           SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\advapi32.dll                          SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\imm32.dll                             SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\imm32.dll                             SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\SysWOW64\imm32.dll                             SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\ProgramData                                            SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\ProgramData                                            NAME COLLISION    Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\ProgramData                                            SUCCESS           Desired Access: E...
6:33:0... tasksche.exe   3884  CreateFile  C:\ProgramData\ytbjymfcjoghel136                          SUCCESS           Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\ProgramData\ytbjymfcjoghel136                          SUCCESS           Desired Access: E...
6:33:0... tasksche.exe   3884  CreateFile  C:\ProgramData\ytbjymfcjoghel136\ytbjymfcjoghel136        NAME NOT FOUND    Desired Access: R...
6:33:0... tasksche.exe   3884  CreateFile  C:\ProgramData\ytbjymfcjoghel136\ytbjymfcjoghel136        NAME NOT FOUND    Desired Access: W...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\tasksche.exe                                   SUCCESS           Desired Access: G...
6:33:0... tasksche.exe   3884  CreateFile  C:\Windows\tasksche.exe                                   SUCCESS           Desired Access: G...
6:33:0... tasksche.exe   3884  CreateFile  C:\ProgramData\ytbjymfcjoghel136\tasksche.exe             SUCCESS           Desired Access: G...
6:33:0... tasksche.exe   3884  CreateFile  C:\ProgramData\ytbjymfcjoghel136\tasksche.exe             SUCCESS           Desired Access: G...
```

```
C:\ProgramData\ytbjymfcjoghel136
File  Home  Share  View
This PC > Local Disk (C:) > ProgramData > ytbjymfcjoghel136 >        Search ytbjymfcjo

Boxstarter     Name                      Date modified         Type            Size
chocolatey     msg                       12/29/2022 6:36 AM    File folder
Desktop        @Please_Read_Me@.txt      12/29/2022 6:33 AM    Text Document       1 KB
Documents      @WanaDecryptor@.exe       5/12/2017 3:22 AM     Application       240 KB
FEVM           @WanaDecryptor@.exe       12/29/2022 6:33 AM    Shortcut            1 KB
Microsoft      00000000.eky              12/29/2022 6:33 AM    EKY File            0 KB
Microsoft V    00000000.pky              12/29/2022 6:33 AM    PKY File            1 KB
Oracle         00000000.res              12/29/2022 6:37 AM    RES File            1 KB
Package Ca     b.wnry                    5/11/2017 9:13 PM     WNRY File       1,407 KB
Packages       c.wnry                    12/29/2022 6:33 AM    WNRY File           1 KB
regid.1991-(   f.wnry                    12/29/2022 6:37 AM    WNRY File           1 KB
shimgen        r.wnry                    5/11/2017 4:59 PM     WNRY File           1 KB
SoftwareDis    s.wnry                    5/9/2017 5:58 PM      WNRY File       2,968 KB
ssh            t.wnry                    5/12/2017 3:22 AM     WNRY File          65 KB
Start Menu     taskdl.exe                5/12/2017 3:22 AM     Application        20 KB
Templates      tasksche.exe              12/29/2022 6:33 AM    Application     3,432 KB
USOPrivate     taskse.exe                5/12/2017 3:22 AM     Application        20 KB
USOShared      u.wnry                    5/12/2017 3:22 AM     WNRY File         240 KB
Windows A
WindowsHc
```

Fig 8: Staging directory

The folder acts as a staging folder for the ransomware malware. It includes all the files needed for the ransomware request, including the message for the users.

Checking Services from Task Manager, I noticed that WannaCry establish its persistence by installing a service in case of restart.



```
Task Manager
File  Options  View
Processes  Performance  App history  Startup  Users  Details  Services

Name                    PID    Description                              Status    Group
WlanSvc                        WLAN AutoConfig                          Stopped   LocalSystemN...
wlidsvc                        Microsoft Account Sign-in Assistant      Stopped   netsvcs
WLMS                    1864   Windows Licensing Monitoring Servi...    Running
wlpasvc                        Local Profile Assistant Service          Stopped   LocalServiceN...
WManSvc                        Windows Management Service               Stopped   netsvcs
wmiApSrv                       WMI Performance Adapter                  Stopped
WMPNetworkSvc                  Windows Media Player Network Sha...      Stopped
workfolderssvc                 Work Folders                             Stopped   LocalService
WpcMonSvc                      Parental Controls                        Stopped   LocalService
WPDBusEnum                     Portable Device Enumerator Service       Stopped   LocalSystemN...
WpnService              2968   Windows Push Notifications System ...    Running   netsvcs
WpnUserService                 Windows Push Notifications User Se...    Stopped   UnistackSvcGr...
WpnUserService_2b6b6    4056   Windows Push Notifications User Se...    Running   UnistackSvcGr...
wscsvc                  5024   Security Center                          Running   LocalServiceN...
WSearch                 4676   Windows Search                           Running
wuauserv                       Windows Update                           Stopped   netsvcs
WwanSvc                        WWAN AutoConfig                          Stopped   LocalSystemN...
XblAuthManager                 Xbox Live Auth Manager                   Stopped   netsvcs
XblGameSave                    Xbox Live Game Save                      Stopped   netsvcs
XboxGipSvc                     Xbox Accessory Management Service        Stopped   netsvcs
XboxNetApiSvc                  Xbox Live Networking Service             Stopped   netsvcs
ytbjymfcjoghel136              ytbjymfcjoghel136                        Stopped
```

Fig 9: Task Manager

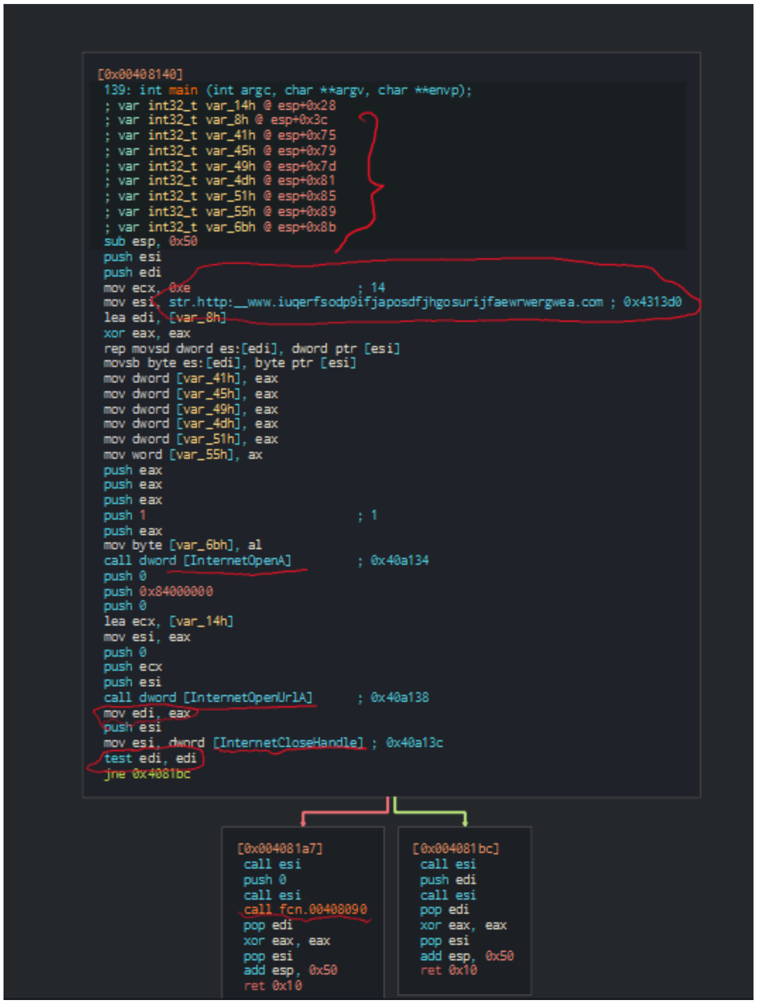
Advanced Static Analysis

{Screenshots and description about findings during advanced static analysis}

Part of WannaCry's infection routine involves sending a DNS request that checks for a live URL/domain. If its request returns showing that the URL is alive or online, it will activate the kill switch, prompting WannaCry to exit the system and no longer proceed with its propagation and encryption routines. Thus, even if the infected machine restarts, the kill switch will prevent WannaCry from performing its routines on it.

Decompiler Graph of the Wannacry:

The "killswitch" mechanism can be found in the disassembly of the main() function:



```
[0x00408140]
 139: int main (int argc, char **argv, char **envp);
 ; var int32_t var_14h @ esp+0x28
 ; var int32_t var_8h @ esp+0x3c
 ; var int32_t var_41h @ esp+0x75
 ; var int32_t var_45h @ esp+0x79
 ; var int32_t var_49h @ esp+0x7d
 ; var int32_t var_4dh @ esp+0x81
 ; var int32_t var_51h @ esp+0x85
 ; var int32_t var_55h @ esp+0x89
 ; var int32_t var_6bh @ esp+0x8b
 sub esp, 0x50
 push esi
 push edi
 mov ecx, 0xe                        ; 14
 mov esi, str.http:__www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com ; 0x4313d0
 lea edi, [var_8h]
 xor eax, eax
 rep movsd dword es:[edi], dword ptr [esi]
 movsb byte es:[edi], byte ptr [esi]
 mov dword [var_41h], eax
 mov dword [var_45h], eax
 mov dword [var_49h], eax
 mov dword [var_4dh], eax
 mov dword [var_51h], eax
 mov word [var_55h], ax
 push eax
 push eax
 push eax
 push 1                              ; 1
 push eax
 mov byte [var_6bh], al
 call dword [InternetOpenA]          ; 0x40a134
 push 0
 push 0x84000000
 push 0
 lea ecx, [var_14h]
 mov esi, eax
 push 0
 push ecx
 push esi
 call dword [InternetOpenUrlA]       ; 0x40a138
 mov edi, eax
 push esi
 mov esi, dword [InternetCloseHandle] ; 0x40a13c
 test edi, edi
 jne 0x4081bc
```

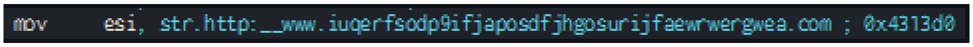
```
[0x004081a7]
 call esi
 push 0
 call esi
 call fcn.00408090
 pop edi
 xor eax, eax
 pop esi
 add esp, 0x50
 ret 0x10
```

```
[0x004081bc]
 call esi
 push edi
 call esi
 pop edi
 xor eax, eax
 pop esi
 add esp, 0x50
 ret 0x10
```

Fig 10: Graph view from the Cutter Decompiler

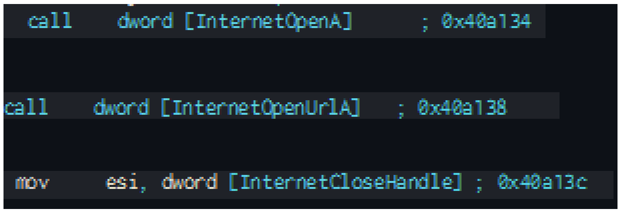
Noticed from the Decompiler Graph, the URL string containing the killswitch URL.

It moves the string into the esi Register:



```
mov     esi, str.http:__www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com ; 0x4313d0
```

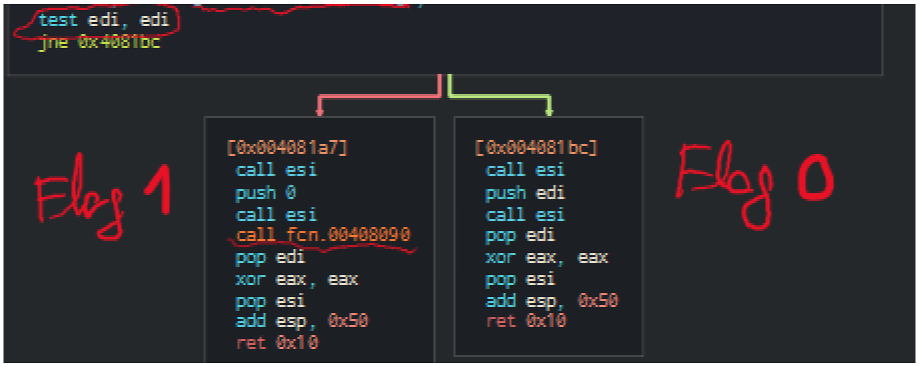
It can also be noticed some important API calls:



```
 call     dword [InternetOpenA]       ; 0x40a134

call     dword [InternetOpenUrlA]    ; 0x40a138

mov      esi, dword [InternetCloseHandle] ; 0x40a13c
```

Everything is moved into edi Register where it performs a test, if it has a flag value of 0, it will continue executing the malware, if it has a flag value of 1, it will return and kill the process. The flag value depends if the call to the killswitch domain succeeds or not.

```
test     edi, edi
```



```
test edi, edi
jne 0x4081bc

[0x004081a7]              [0x004081bc]
 call esi                  call esi
 push 0                    push edi
 call esi                  call esi
 call fcn.00408090         pop edi
 pop edi                   xor eax, eax
 xor eax, eax              pop esi
 pop esi                   add esp, 0x50
 add esp, 0x50             ret 0x10
 ret 0x10
```

For Flag 0, the malware manages to connect to the killswitch domain, and the malware immediately exits.

```
add esp, 0x50
ret 0x10
```

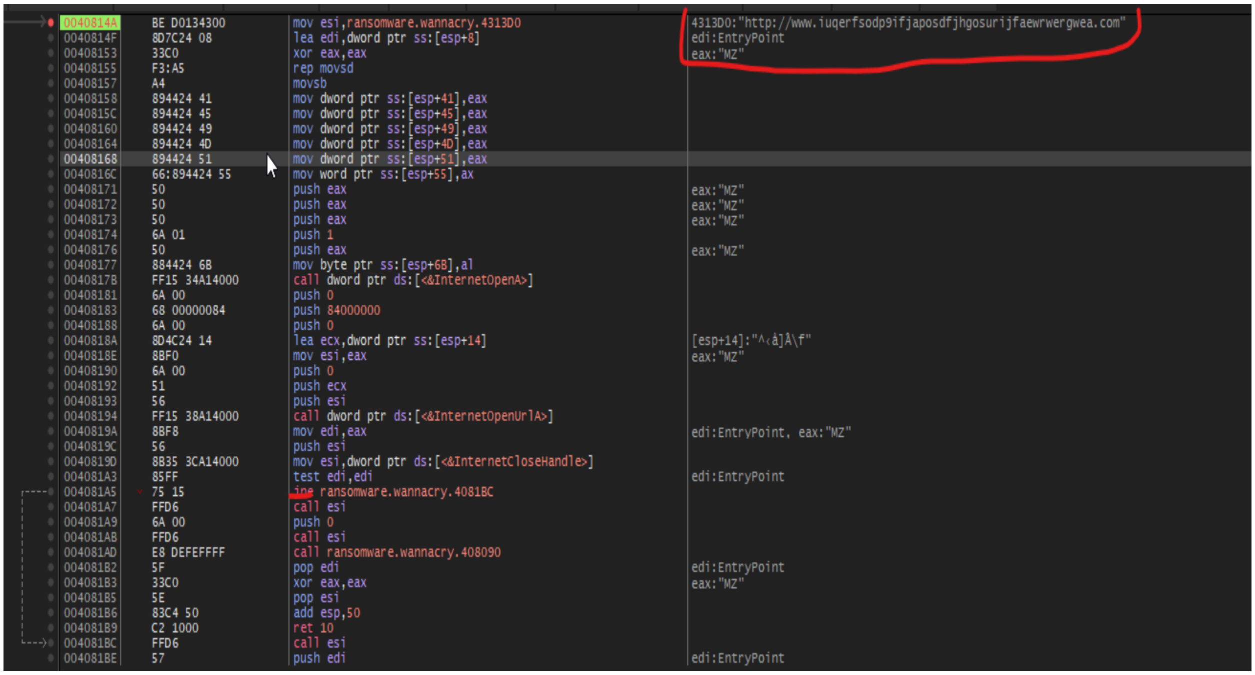
For Flag 1, the malware doesn't succeed in connecting to the killswitch Domain and it performs a call to another function:

```
 call     fcn.00408090
```

Advanced Dynamic Analysis

{Screenshots and description about advanced dynamic artifacts and methods}

We will start analyzing the WannaCry ransomware from that killswitch URL:

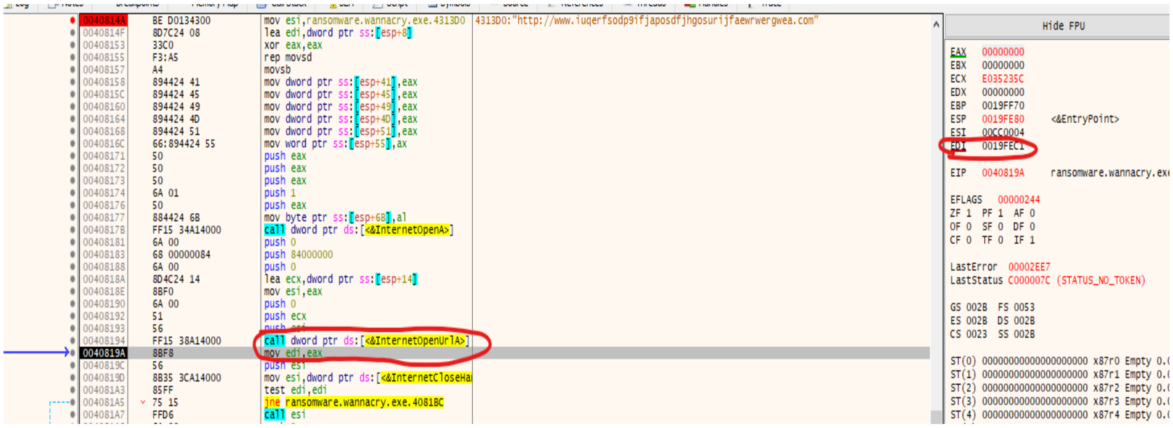


```
0040814A  BE D0134300     mov esi,ransomware.wannacry.4313D0          4313D0:"http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com"
0040814F  8D7C24 08       lea edi,dword ptr ss:[esp+8]                edi:EntryPoint
00408153  33C0            xor eax,eax                                 eax:"MZ"
00408155  F3:A5           rep movsd
00408157  A4              movsb
00408158  894424 41       mov dword ptr ss:[esp+41],eax
0040815C  894424 45       mov dword ptr ss:[esp+45],eax
00408160  894424 49       mov dword ptr ss:[esp+49],eax
00408164  894424 4D       mov dword ptr ss:[esp+4D],eax
00408168  894424 51       mov dword ptr ss:[esp+51],eax
0040816C  66:894424 55    mov word ptr ss:[esp+55],ax
00408171  50              push eax                                    eax:"MZ"
00408172  50              push eax                                    eax:"MZ"
00408173  50              push eax                                    eax:"MZ"
00408174  6A 01           push 1
00408176  50              push eax                                    eax:"MZ"
00408177  884424 6B       mov byte ptr ss:[esp+6B],al
0040817B  FF15 34A14000   call dword ptr ds:[<&InternetOpenA>]
00408181  6A 00           push 0
00408183  68 00000084     push 84000000
00408188  6A 00           push 0
0040818A  8D4C24 14       lea ecx,dword ptr ss:[esp+14]               [esp+14]:"^‹å]Å\f"
0040818E  8BF0            mov esi,eax                                 eax:"MZ"
00408190  6A 00           push 0
00408192  51              push ecx
00408193  56              push esi
00408194  FF15 38A14000   call dword ptr ds:[<&InternetOpenUrlA>]
0040819A  8BF8            mov edi,eax                                 edi:EntryPoint, eax:"MZ"
0040819C  56              push esi
0040819D  8B35 3CA14000   mov esi,dword ptr ds:[<&InternetCloseHandle>]
004081A3  85FF            test edi,edi                                edi:EntryPoint
004081A5  75 15           jne ransomware.wannacry.4081BC
004081A7  FFD6            call esi
004081A9  6A 00           push 0
004081AB  FFD6            call esi
004081AD  E8 DEFEFFFF     call ransomware.wannacry.408090
004081B2  5F              pop edi                                     edi:EntryPoint
004081B3  33C0            xor eax,eax                                 eax:"MZ"
004081B5  5E              pop esi
004081B6  83C4 50         add esp,50
004081B9  C2 1000         ret 10
004081BC  FFD6            call esi
004081BE  57              push edi                                    edi:EntryPoint
```

Fig 11: x32dbg Debugger

We will step over until we reach the *test edi,edi* function

The program tests the reachability to the killswitch URL using the API call InternetOpenUrlA.



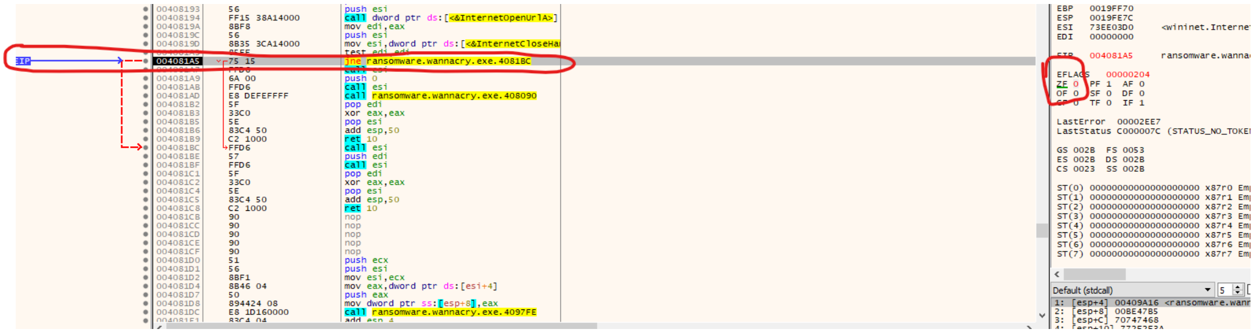
```
0040814A  BE D0134300      mov esi,ransomware.wannacry.exe.4313D0   4313D0:"http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com"
0040814F  8D7C24 08        lea edi,dword ptr ss:[esp+8]
00408153  33C0             xor eax,eax
00408155  F3:A5            rep movsd
00408157  A4               movsb
00408158  894424 41        mov dword ptr ss:[esp+41],eax
0040815C  894424 45        mov dword ptr ss:[esp+45],eax
00408160  894424 49        mov dword ptr ss:[esp+49],eax
00408164  894424 4D        mov dword ptr ss:[esp+4D],eax
00408168  894424 51        mov dword ptr ss:[esp+51],eax
0040816C  66:894424 55     mov word ptr ss:[esp+55],ax
00408171  50               push eax
00408172  50               push eax
00408173  50               push eax
00408174  6A 01            push 1
00408176  50               push eax
00408177  884424 6B        mov byte ptr ss:[esp+6B],al
0040817B  FF15 34A14000    call dword ptr ds:[<&InternetOpenA>]
00408181  6A 00            push 0
00408183  68 00000084      push 84000000
00408188  6A 00            push 0
0040818A  8D4C24 14        lea ecx,dword ptr ss:[esp+14]
0040818E  8BF0             mov esi,eax
00408190  6A 00            push 0
00408192  51               push ecx
00408193  56               push esi
00408194  FF15 38A14000    call dword ptr ds:[<&InternetOpenUrlA>]
0040819A  8BF8             mov edi,eax
0040819C  56               push esi
0040819D  8B35 3CA14000    mov esi,dword ptr ds:[<&InternetCloseHa
004081A3  85FF             test edi,edi
004081A5  75 15            jne ransomware.wannacry.exe.4081BC
004081A7  FFD6             call esi
```

```
Hide FPU

EAX    00000000
EBX    00000000
ECX    E035235C
EDX    00000000
EBP    0019FF70
ESP    0019FE80     <&EntryPoint>
ESI    00CC0004
EDI    0019FEC1

EIP    0040819A     ransomware.wannacry.exe

EFLAGS   00000244
ZF 1  PF 1  AF 0
OF 0  SF 0  DF 0
CF 0  TF 0  IF 1

LastError   00002EE7
LastStatus C000007C (STATUS_NO_TOKEN)

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

ST(0) 0000000000000000000 x87r0 Empty 0.0
ST(1) 0000000000000000000 x87r1 Empty 0.0
ST(2) 0000000000000000000 x87r2 Empty 0.0
ST(3) 0000000000000000000 x87r3 Empty 0.0
ST(4) 0000000000000000000 x87r4 Empty 0.0
```

Fig 12: InternetOpenUrlA API call in x32dbg

It can be observed that the value of the EDI is set at this point in the function. The debugger was run without inetsim running, so the EDI it is not 0.

If the program manages to reach the killswitch URL, it will have a flag of 0 and it will jump (jne) and exit the program:



```
00408193  56               push esi
00408194  FF15 38A14000    call dword ptr ds:[<&InternetOpenUrlA>]
0040819A  8BF8             mov edi,eax
0040819C  56               push esi
0040819D  8B35 3CA14000    mov esi,dword ptr ds:[<&InternetCloseHa
004081A3  85FF             test edi,edi
004081A5  75 15            jne ransomware.wannacry.exe.4081BC
004081A7  FFD6             call esi
004081A9  6A 00            push 0
004081AB  FFD6             call esi
004081AD  E8 DEFEFFFF      call ransomware.wannacry.exe.408090
004081B2  5F               pop edi
004081B3  33C0             xor eax,eax
004081B5  5E               pop esi
004081B6  83C4 50          add esp,50
004081B9  C2 1000          ret 10
004081BC  FFD6             call esi
004081BE  57               push edi
004081BF  FFD6             call esi
004081C1  5F               pop edi
004081C2  33C0             xor eax,eax
004081C4  5E               pop esi
004081C5  83C4 50          add esp,50
004081C8  C2 1000          ret 10
004081CB  90               nop
004081CC  90               nop
004081CD  90               nop
004081CE  90               nop
004081CF  90               nop
004081D0  51               push ecx
004081D1  56               push esi
004081D2  8BF1             mov esi,ecx
004081D4  8B46 04          mov eax,dword ptr ds:[esi+4]
004081D7  50               push eax
004081D8  894424 08        mov dword ptr ss:[esp+8],eax
004081DC  E8 1D160000      call ransomware.wannacry.exe.4097FE
004081E1  83C4 04          add esp,4
```

```
EBP   0019FF70
ESP   0019FE7C
ESI   73EE03D0     <wininet.Interne
EDI   00000000

EIP   004081A5     ransomware.wanna

EFLAGS   00000204
ZF 0  PF 1  AF 0
OF 0  SF 0  DF 0
CF 0  TF 0  IF 1

LastError  00002EE7
LastStatus C000007C (STATUS_NO_TOKE

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

ST(0) 0000000000000000000 x87r0 Em
ST(1) 0000000000000000000 x87r1 Em
ST(2) 0000000000000000000 x87r2 Em
ST(3) 0000000000000000000 x87r3 Em
ST(4) 0000000000000000000 x87r4 Em
ST(5) 0000000000000000000 x87r5 Em
ST(6) 0000000000000000000 x87r6 Em
ST(7) 0000000000000000000 x87r7 Em

Default (stdcall)   5

1: [esp+4] 00409A16 <ransomware.wann
2: [esp+8] 00BE47B5
3: [esp+C] 70747468
```

Fig 13: jne instruction (killswitch)

If the flag is set to 1, it will not jump (jne) and it will continue the program, infecting the host.

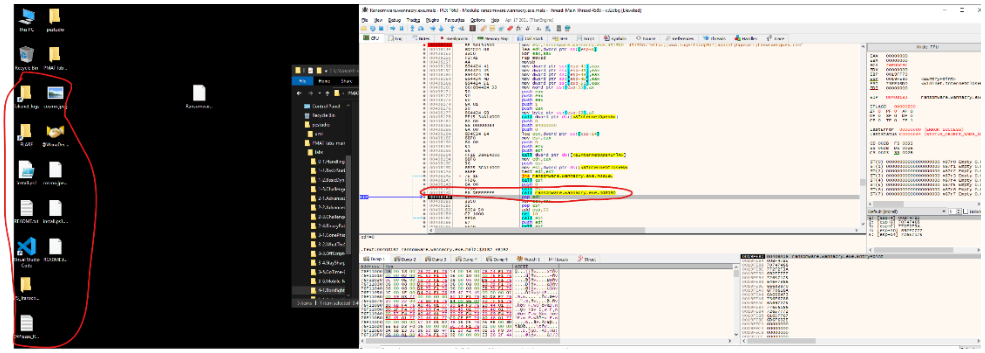


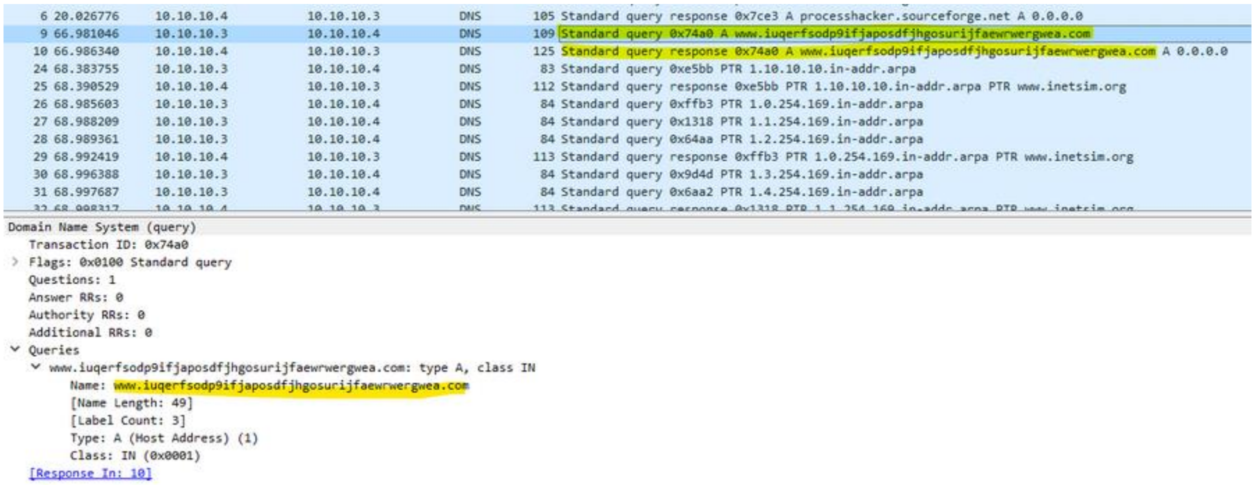
Fig 13: x32dbg execute the WannaCry ransomware from the Debugger

Indicators of Compromise

The full list of IOCs can be found in the Appendices.

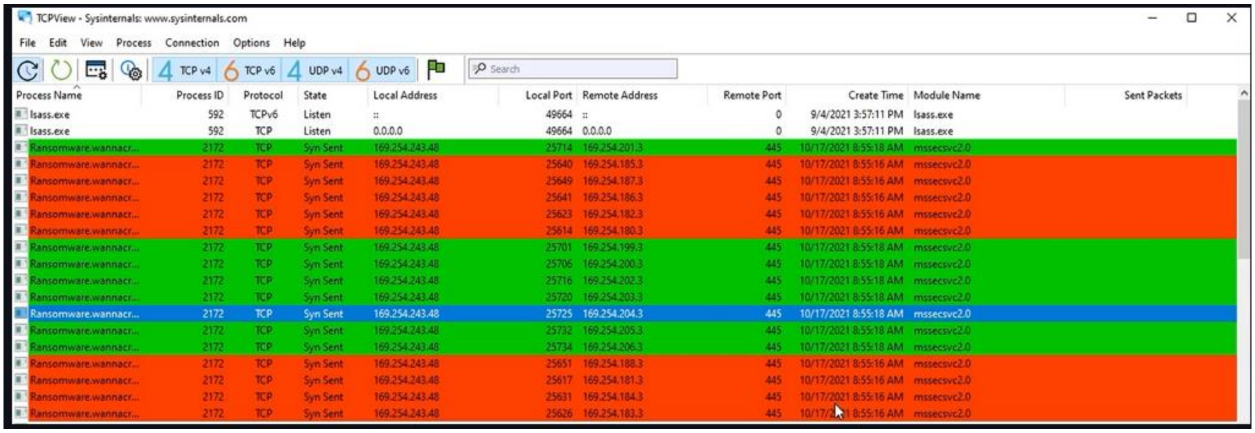
Network Indicators

{Description of network indicators}



```
 6 20.026776    10.10.10.4    10.10.10.3    DNS    105 Standard query response 0x7ce3 A processhacker.sourceforge.net A 0.0.0.0
 9 66.981046    10.10.10.3    10.10.10.4    DNS    109 Standard query 0x74a0 A www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
10 66.986340    10.10.10.4    10.10.10.3    DNS    125 Standard query response 0x74a0 A www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com A 0.0.0.0
24 68.383755    10.10.10.3    10.10.10.4    DNS     83 Standard query 0xe5bb PTR 1.10.10.10.in-addr.arpa
25 68.390529    10.10.10.4    10.10.10.3    DNS    112 Standard query response 0xe5bb PTR 1.10.10.10.in-addr.arpa PTR www.inetsim.org
26 68.985603    10.10.10.3    10.10.10.4    DNS     84 Standard query 0xffb3 PTR 1.0.254.169.in-addr.arpa
27 68.988209    10.10.10.3    10.10.10.4    DNS     84 Standard query 0x1318 PTR 1.1.254.169.in-addr.arpa
28 68.989361    10.10.10.3    10.10.10.4    DNS     84 Standard query 0x64aa PTR 1.2.254.169.in-addr.arpa
29 68.992419    10.10.10.4    10.10.10.3    DNS    113 Standard query response 0xffb3 PTR 1.0.254.169.in-addr.arpa PTR www.inetsim.org
30 68.996388    10.10.10.3    10.10.10.4    DNS     84 Standard query 0x9d4d PTR 1.3.254.169.in-addr.arpa
31 68.997687    10.10.10.3    10.10.10.4    DNS     84 Standard query 0x6aa2 PTR 1.4.254.169.in-addr.arpa
32 68.998317    10.10.10.4    10.10.10.3    DNS    113 Standard query response 0x1318 PTR 1.1.254.169.in-addr.arpa PTR www.inetsim.org
Domain Name System (query)
   Transaction ID: 0x74a0
 > Flags: 0x0100 Standard query
   Questions: 1
   Answer RRs: 0
   Authority RRs: 0
   Additional RRs: 0
 v Queries
   v www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com: type A, class IN
        Name: www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
        [Name Length: 49]
        [Label Count: 3]
        Type: A (Host Address) (1)
        Class: IN (0x0001)
   [Response In: 10]
```

Fig 14: WireShark Packet Capture of the initial call to the killswitch URL



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	9/4/2021 3:57:11 PM	lsass.exe	
lsass.exe	592	TCP	Listen	0.0.0.0	49664	0.0.0.0	0	9/4/2021 3:57:11 PM	lsass.exe	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25714	169.254.201.3	445	10/17/2021 8:55:18 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25640	169.254.185.3	445	10/17/2021 8:55:16 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25649	169.254.187.3	445	10/17/2021 8:55:16 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25641	169.254.186.3	445	10/17/2021 8:55:16 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25623	169.254.182.3	445	10/17/2021 8:55:16 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25614	169.254.180.3	445	10/17/2021 8:55:16 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25701	169.254.199.3	445	10/17/2021 8:55:18 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25706	169.254.200.3	445	10/17/2021 8:55:18 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25716	169.254.202.3	445	10/17/2021 8:55:18 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25720	169.254.203.3	445	10/17/2021 8:55:18 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25725	169.254.204.3	445	10/17/2021 8:55:18 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25732	169.254.205.3	445	10/17/2021 8:55:18 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25734	169.254.206.3	445	10/17/2021 8:55:18 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25651	169.254.188.3	445	10/17/2021 8:55:16 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25617	169.254.181.3	445	10/17/2021 8:55:16 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25631	169.254.184.3	445	10/17/2021 8:55:16 AM	mssecsvc2.0	
Ransomware.wannacr...	2172	TCP	Syn Sent	169.254.243.48	25626	169.254.183.3	445	10/17/2021 8:55:16 AM	mssecsvc2.0	

Fig 15: TCPView

The executable makes some TCP calls to an IP on port 445. (that IP from the image above it is not the actual malicious IP, it is generated automatically when the DHCP server is not reachable)

WannaCry takes advantage of the Eternal Blue exploit to spread throughout the network, it is still one of the most commonly attempted exploits against SMB, accounting for over 91.88% of the attacks on port 445 (the most common SMB port).

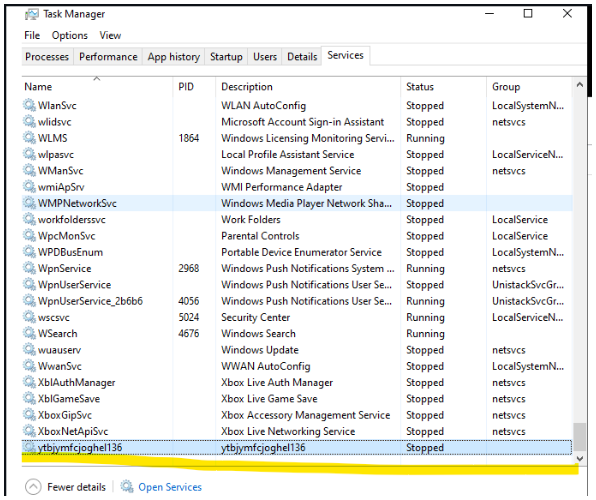
Host-based Indicators

{Description of host-based indicators}

1. The directory used for the staging area for WannaCry, Fig 1

Note: The directory may be different from host to host.

2. Establish persistence by installing a service in case of restart.

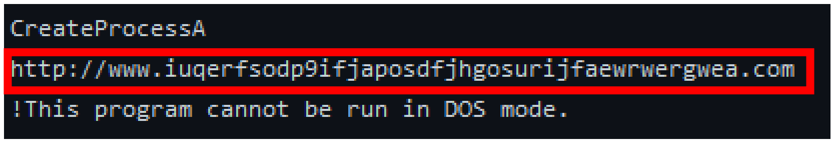


Task Manager
File Options View
Processes Performance App history Startup Users Details Services
Fewer details
Open Services

Name	PID	Description	Status	Group
WlanSvc		WLAN AutoConfig	Stopped	LocalSystemN...
wlidsvc		Microsoft Account Sign-in Assistant	Stopped	netsvcs
WLMS	1864	Windows Licensing Monitoring Servi...	Running	
wlpasvc		Local Profile Assistant Service	Stopped	LocalServiceN...
WManSvc		Windows Management Service	Stopped	netsvcs
wmiApSrv		WMI Performance Adapter	Stopped	
WMPNetworkSvc		Windows Media Player Network Sha...	Stopped	
workfolderssvc		Work Folders	Stopped	LocalService
WpcMonSvc		Parental Controls	Stopped	LocalService
WPDBusEnum		Portable Device Enumerator Service	Stopped	LocalSystemN...
WpnService	2968	Windows Push Notifications System ...	Running	netsvcs
WpnUserService		Windows Push Notifications User Se...	Stopped	UnistackSvcGr...
WpnUserService_2b6b6	4056	Windows Push Notifications User Se...	Running	UnistackSvcGr...
wscsvc	5024	Security Center	Running	LocalServiceN...
WSearch	4676	Windows Search	Running	
wuauserv		Windows Update	Stopped	netsvcs
WwanSvc		WWAN AutoConfig	Stopped	LocalSystemN...
XblAuthManager		Xbox Live Auth Manager	Stopped	netsvcs
XblGameSave		Xbox Live Game Save	Stopped	netsvcs
XboxGipSvc		Xbox Accessory Management Service	Stopped	netsvcs
XboxNetApiSvc		Xbox Live Networking Service	Stopped	netsvcs
ytbjymfcjoghel136		ytbjymfcjoghel136	Stopped	

Fig 16: Task Manager – Service for persistence

3. *Call to the malicious URL*



```
CreateProcessA
http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
!This program cannot be run in DOS mode.
```

4. The malware changes the Desktop wallpaper and adds the decryptor and other files on the desktop.

Rules & Signatures

A full set of YARA rules is included in Appendix A.

The WannaCry ransomware malware has very apparent signature,

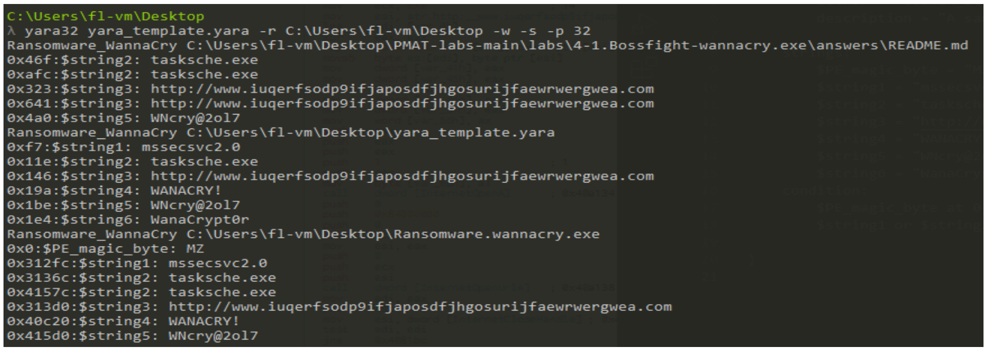
- various obvious strings can be found in the binary: “wnry”, “WANACRY!”, “WNcry@2017”
- callback to that well-known URL used for WannaCry Ransomware

Appendices

A. Yara Rules

Full Yara repository located at:
https://github.com/cosmin-stan/Malware-Analysis/tree/main/Malware/Ransomware/Wannacry

```
rule Ransomware_WannaCry {

    meta:
        last_updated = "2021-01-08"
        author = "cosmin_stan"
        description = "A sample Yara rule to detect WannaCry Ransomware"

    strings:
        $PE_magic_byte = "MZ"
        $string1 = "mssecsvc2.0" ascii
        $string2 = "tasksche.exe" ascii
        $string3 = "http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com"
ascii
        $string4 = "WANACRY!" ascii
        $string5 = "WNcry@2ol7" ascii
        $string6 = "WanaCrypt0r" ascii
    condition:
        $PE_magic_byte at 0 and
        $string1 or $string2 or $string3 or $string4 or $string5 or $string6

}
```



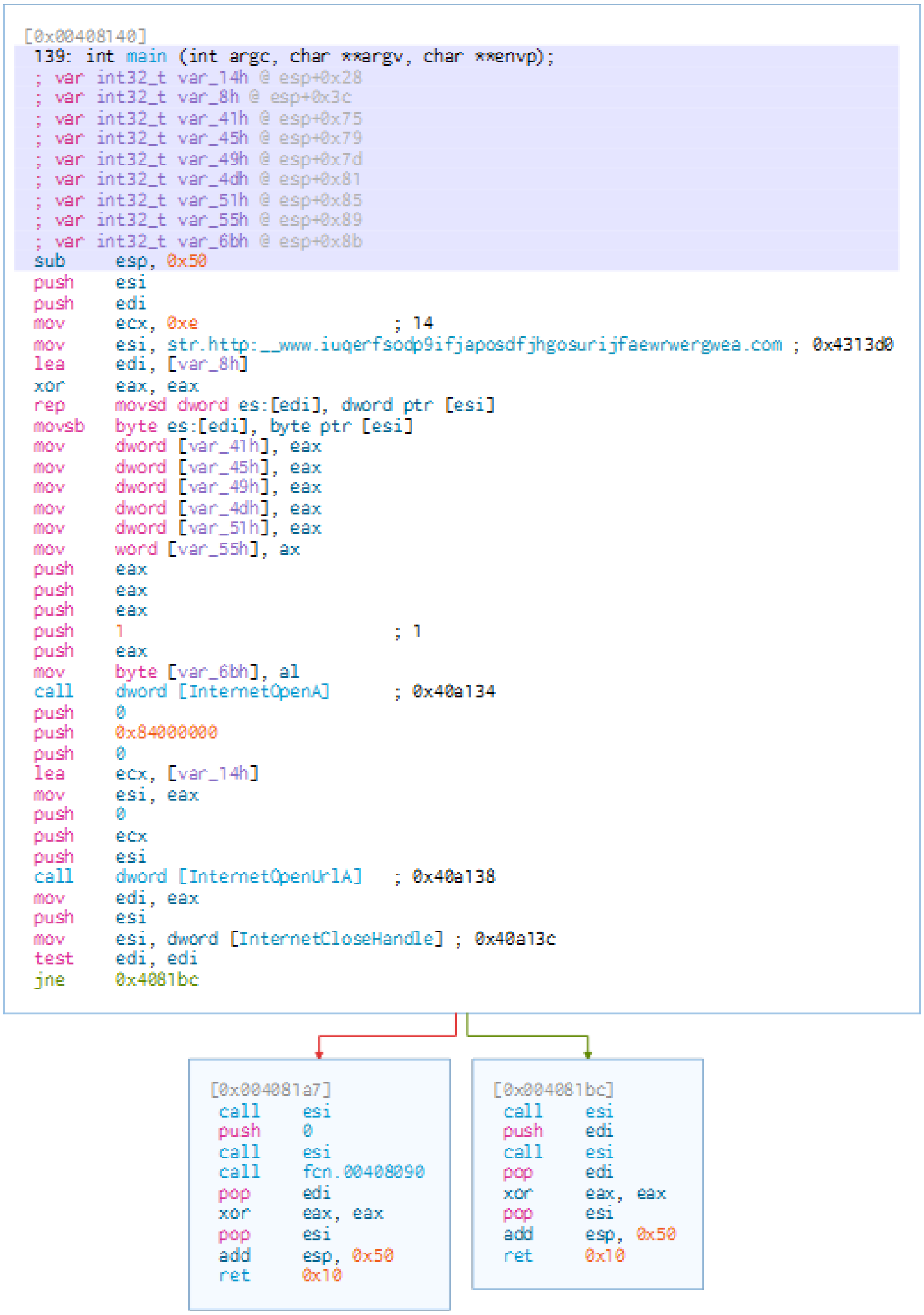
```
C:\Users\fl-vm\Desktop
λ yara32 yara_template.yara -r C:\Users\fl-vm\Desktop -w -s -p 32
Ransomware_WannaCry C:\Users\fl-vm\Desktop\PMAT-labs-main\labs\4-1.Bossfight-wannacry.exe\answers\README.md
0x46f:$string2: tasksche.exe
0xafc:$string2: tasksche.exe
0x323:$string3: http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
0x641:$string3: http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
0x4a0:$string5: WNcry@2ol7
Ransomware_WannaCry C:\Users\fl-vm\Desktop\yara_template.yara
0xf7:$string1: mssecsvc2.0
0x11e:$string2: tasksche.exe
0x146:$string3: http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
0x19a:$string4: WANACRY!
0x1be:$string5: WNcry@2ol7
0x1e4:$string6: WanaCrypt0r
Ransomware_WannaCry C:\Users\fl-vm\Desktop\Ransomware.wannacry.exe
0x0:$PE_magic_byte: MZ
0x312fc:$string1: mssecsvc2.0
0x3136c:$string2: tasksche.exe
0x4157c:$string2: tasksche.exe
0x313d0:$string3: http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
0x40c20:$string4: WANACRY!
0x415d0:$string5: WNcry@2ol7
```

Fig 17: Testing the Yara rule

B. Callback URL

Domain	Port
hxxp://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com	443

C. Decompiled Code Snippets



```
[0x00408140]
 139: int main (int argc, char **argv, char **envp);
 ; var int32_t var_14h @ esp+0x28
 ; var int32_t var_8h @ esp+0x3c
 ; var int32_t var_41h @ esp+0x75
 ; var int32_t var_45h @ esp+0x79
 ; var int32_t var_49h @ esp+0x7d
 ; var int32_t var_4dh @ esp+0x81
 ; var int32_t var_51h @ esp+0x85
 ; var int32_t var_55h @ esp+0x89
 ; var int32_t var_6bh @ esp+0x8b
 sub     esp, 0x50
 push    esi
 push    edi
 mov     ecx, 0xe                      ; 14
 mov     esi, str.http:__www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com ; 0x4313d0
 lea     edi, [var_8h]
 xor     eax, eax
 rep     movsd dword es:[edi], dword ptr [esi]
 movsb   byte es:[edi], byte ptr [esi]
 mov     dword [var_41h], eax
 mov     dword [var_45h], eax
 mov     dword [var_49h], eax
 mov     dword [var_4dh], eax
 mov     dword [var_51h], eax
 mov     word [var_55h], ax
 push    eax
 push    eax
 push    eax
 push    1                             ; 1
 push    eax
 mov     byte [var_6bh], al
 call    dword [InternetOpenA]         ; 0x40a134
 push    0
 push    0x84000000
 push    0
 lea     ecx, [var_14h]
 mov     esi, eax
 push    0
 push    ecx
 push    esi
 call    dword [InternetOpenUrlA]      ; 0x40a138
 mov     edi, eax
 push    esi
 mov     esi, dword [InternetCloseHandle] ; 0x40a13c
 test    edi, edi
 jne     0x4081bc
```

```
[0x004081a7]
 call    esi
 push    0
 call    esi
 call    fcn.00408090
 pop     edi
 xor     eax, eax
 pop     esi
 add     esp, 0x50
 ret     0x10
```

```
[0x004081bc]
 call    esi
 push    edi
 call    esi
 pop     edi
 xor     eax, eax
 pop     esi
 add     esp, 0x50
 ret     0x10
```

Fig 17: Process Injection Routine in Cutter