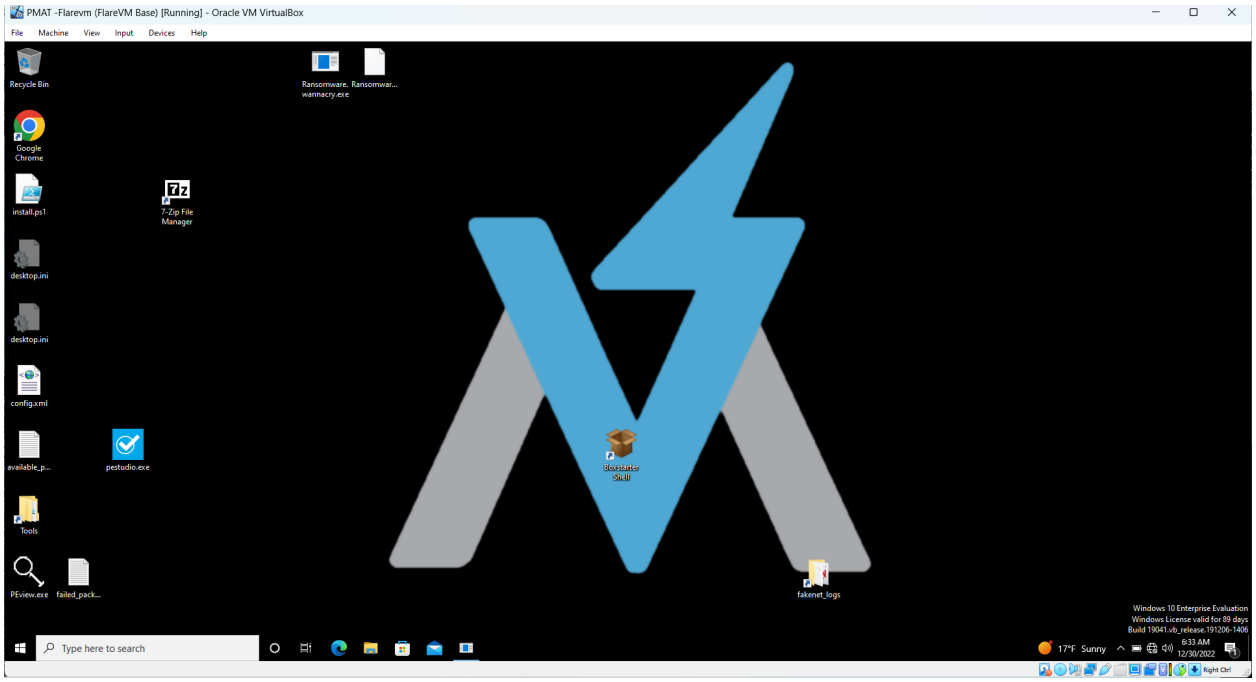
I started off by loading up my FlareVM sandbox on Virtualbox. In this analysis I use the tools of Remnux, 7Zip, Wireshark, PeStudio, PeView, Floss, Process Monitor(ProcMon), Capa and Cutter,



This is my base FlareVM install that I will use to revert back to once wannacry encrypts my computer.

```
VarFileInfo
Translation

----------------------------
| FLOSS STACK STRINGS (17) |
----------------------------
SMBu
/K__USERID__PLACEHOLDER__
__TREEPATH_REPLACE__
PIPE
SMBr
PC NETWORK PROGRAM 1.0
LANMAN1.0
Windows for Workgroups 3.1a
LM1.2X002
LANMAN2.1
NT LM 0.12
SMBs
SMB2
Windows 2000 2195
Windows 2000 5.0
\192.168.56.20\IPC$
http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com

----------------------------
| FLOSS TIGHT STRINGS (0) |
----------------------------
```

I start off by running floss.exe. FLOSS analyzes compiled programs, identifies functions that may decode data, and automatically deobfuscates hidden strings. I picked up a few interesting strings from my floss output. The most interesting in my opinion is the URL along with an IP address.

```
C:\Users\Jalen\Desktop
λ capa Ransomware.wannacry.exe
loading : 100%|
matching: 100%|
+------------------------+------------------------------------------------------------------------------+
| md5                    | db349b97c37d22f5ea1d1841e3c89eb4                                             |
| sha1                   | e889544aff85ffaf8b0d0da705105dee7c97fe26                                     |
| sha256                 | 24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c             |
| os                     | windows                                                                      |
| format                 | pe                                                                           |
| arch                   | i386                                                                         |
| path                   | Ransomware.wannacry.exe                                                      |
+------------------------+------------------------------------------------------------------------------+

+------------------------+------------------------------------------------------------------------------+
| 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   | Conditional Execution::Runs as Service [B0025.007]                    |
|                            | Debugger Detection::Timing/Delay Check QueryPerformanceCounter [B0001.033]  |
| 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]                                           |
| DISCOVERY                  | Code Discovery::Inspect Section Memory Permissions [B0046.002]        |
| EXECUTION                  | Install Additional Program [B0023]                                    |
| FILE SYSTEM                | Move File [C0063]                                                     |
|                            | Read File [C0051]                                                     |
| PROCESS                    | Create Thread [C0038]                                                 |
```

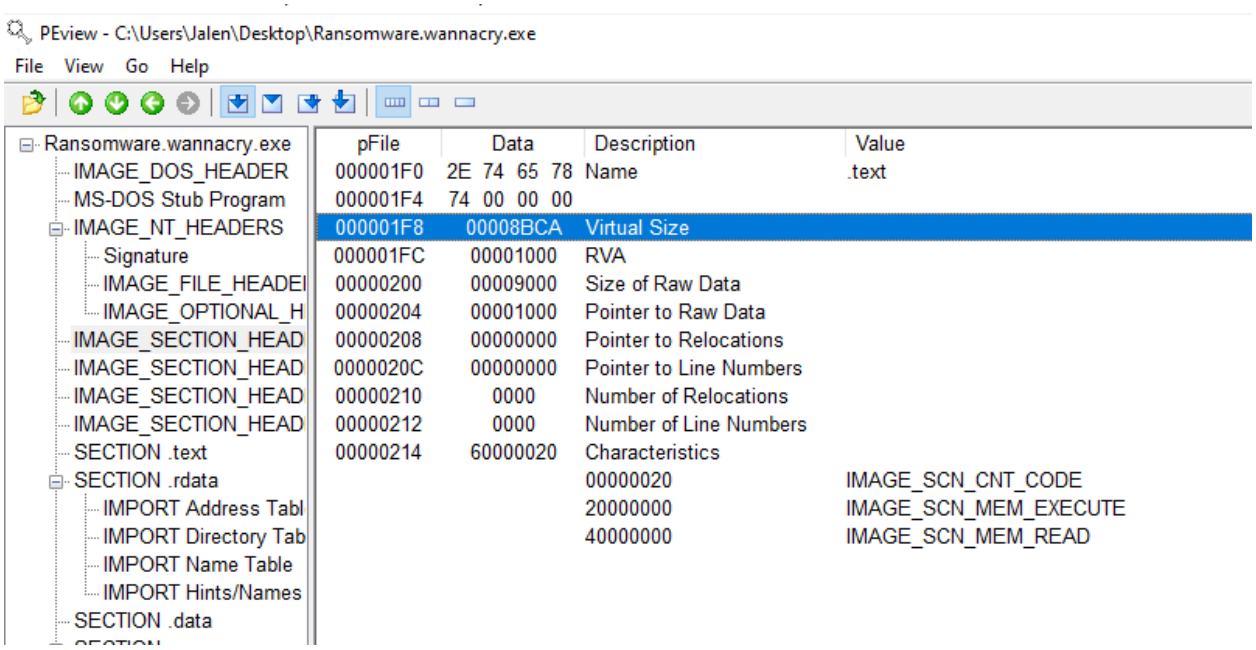
pestudio 9.46 - Malware Initial Assessment - www.winitor.com - [c:\users\jalen\desktop\ransomware.wannacry.exe]

file settings about

- c:\users\jalen\desktop\ransomware.wannacry.ex
  - indicators (wait...)
  - virustotal (error)
  - dos-header (64 bytes)
  - dos-stub (wait...)
  - rich-header (Visual Studio)
  - file-header (Intel-386)
  - optional-header (GUI)
  - directories (3)
  - sections (wait...)
  - libraries (wait...)
  - imports (wait...)
  - exports (n/a)
  - exceptions (n/a)
  - tls-callback (n/a)
  - relocations (n/a)
  - .NET (wait...)
  - resources (executable)
  - strings (wait...)
  - debug (n/a)
  - manifest (n/a)
  - version (lhdfrgui.exe)
  - overlay (wait...)

property	value
md5	DB349B97C37D22F5EA1D1841E3C89EB4
sha1	E889544AFF85FFAF8B0D0DA705105DEE7C97FE26
sha256	24D004A104D4D54034DBCFFC2A4B19A11F39008A575AA614EA04703480B1022C
first-bytes-hex	4D 5A 90 00 03 00 00 00 04 00 00 00 FF FF 00 00 B8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00 00 00
first-bytes-text	M Z . . . . . @ . . . . .
file-size	3723264 bytes
entropy	7.964
imphash	n/a
signature	Microsoft Visual C++ v6.0
tooling	wait...
entry-point	55 8B EC 6A FF 68 A0 A1 40 00 68 A2 9B 40 00 64 A1 00 00 00 00 50 64 89 25 00 00 00 00 83 EC 68 53
file-version	6.1.7601.17514 (win7sp1_rtm.101119-1850)
description	Microsoft® Disk Defragmenter
file-type	executable
cpu	32-bit
subsystem	GUI
compiler-stamp	Sat Nov 20 09:03:08 2010 | UTC
debugger-stamp	n/a
resources-stamp	0x00000000
import-stamp	0x00000000
exports-stamp	n/a

I then load up PEStudio to see what is on the inside of the wannacry executable. Pestudio is a free tool that allows you to perform an initial assessment of a malware without even infecting a system or studying its code.



PEview provides a quick and easy way to view the structure and content of 32-bit Portable Executable (PE) and Component Object File Format (COFF) files. This PE/COFF file viewer displays header, section, directory, import table, export table, and resource information within EXE, DLL, OBJ, LIB, DBG, and other file types.

Then I open up PEView.

To compare the virtual size and the size of raw data and to determine whether the executable is packed or not packed. What is a PE viewer?

pFile	Data	Description	Value
000001F0	2E 74 65 78	Name	.text
000001F4	74 00 00 00		
000001F8	00008BCA	Virtual Size	
000001FC	00001000	RVA	
00000200	00009000	Size of Raw Data	
00000204	00001000	Pointer to Raw Data	
00000208	00000000	Pointer to Relocations	
0000020C	00000000	Pointer to Line Numbers	
00000210	0000	Number of Relocations	
00000212	0000	Number of Line Numbers	
00000214	60000020	Characteristics	
		00000020	IMAGE_SCN_CNT_CODE
		20000000	IMAGE_SCN_MEM_EXECU
		40000000	IMAGE_SCN_MEM_READ

Calculator

Programmer

8BCA

HEX 8BCA

DEC 35,786

OCT 105 712

BIN 1000 1011 1100 1010

pFile	Data	Description	Value
000001F0	2E 74 65 78	Name	.text
000001F4	74 00 00 00		
000001F8	00008BCA	Virtual Size	
000001FC	00001000	RVA	
00000200	00009000	Size of Raw Data	
00000204	00001000	Pointer to Raw Data	
00000208	00000000	Pointer to Relocations	
0000020C	00000000	Pointer to Line Numbers	
00000210	0000	Number of Relocations	
00000212	0000	Number of Line Numbers	
00000214	60000020	Characteristics	
		00000020	IMAGE_SCN_CNT_CODE
		20000000	IMAGE_SCN_MEM_EXECU
		40000000	IMAGE_SCN_MEM_READ

Calculator

Programmer

9000

HEX 9000

DEC 36,864

OCT 110 000

BIN 1001 0000 0000 0000

QWORD MS

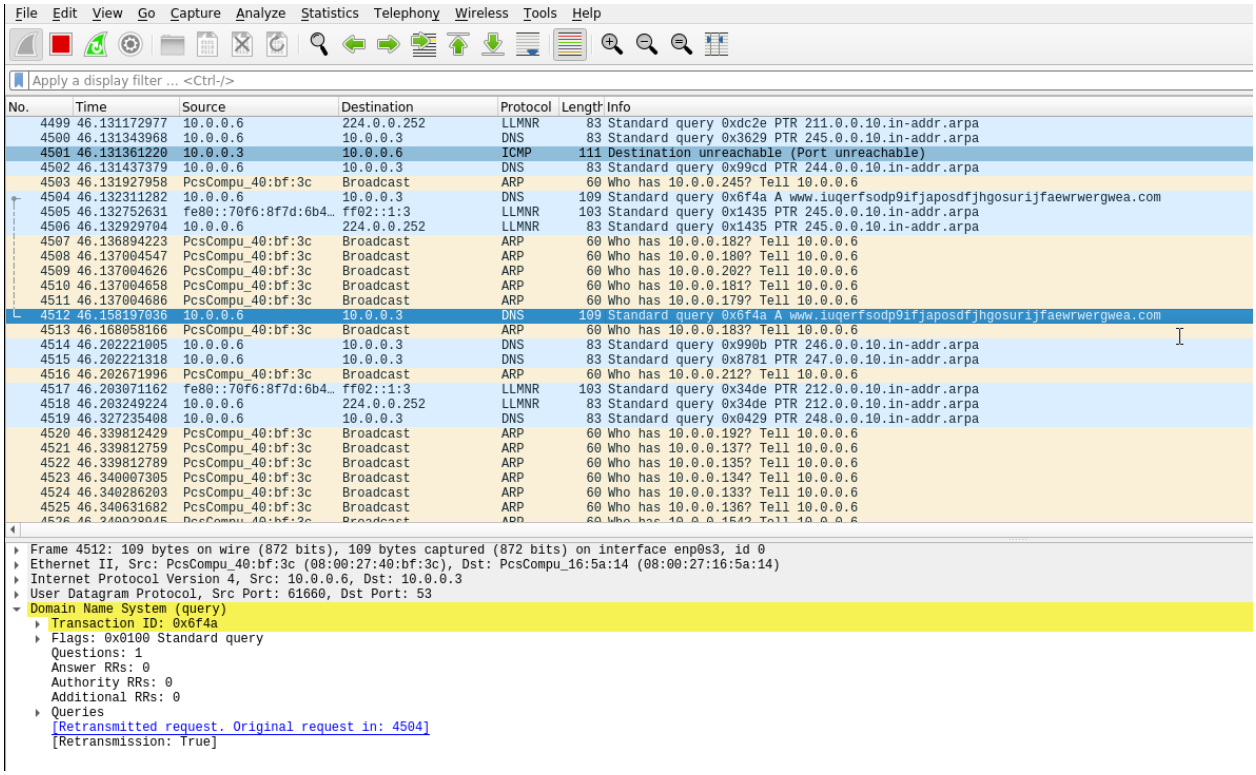
Bitwise Bit shift

I then take out my programming calculator to plug in the numbers to see if theyre close in size. The hexadecimals were close so there was no information to be gathered there.

```
remnux@remnux:~$ inetsim
INetSim 1.3.2 (2020-05-19) by Matthias Eckert & Thomas Hungenberg
Using log directory:      /var/log/inetsim/
Using data directory:     /var/lib/inetsim/
Using report directory:   /var/log/inetsim/report/
Using configuration file: /etc/inetsim/inetsim.conf
Parsing configuration file.
Configuration file parsed successfully.
=== INetSim main process started (PID 1659) ===
Session ID:     1659
Listening on:   10.0.0.3
Real Date/Time: 2022-12-30 11:15:38
Fake Date/Time: 2022-12-30 11:15:38 (Delta: 0 seconds)
 Forking services...
  * https_443_tcp - started (PID 1665)
  * smtp_25_tcp - started (PID 1666)
  * dns_53_tcp_udp - started (PID 1663)
  * pop3_110_tcp - started (PID 1668)
  * smtps_465_tcp - started (PID 1667)
  * pop3s_995_tcp - started (PID 1669)
  * ftp_21_tcp - started (PID 1670)
  * http_80_tcp - started (PID 1664)
  * ftps_990_tcp - started (PID 1671)
 done.
Simulation running.
```

I spawn my remnux VM to help with this analysis and I run iNetsim and wireshark. INetSim is a software suite for simulating common internet services in a lab environment

I detonate the wannacry.exe and to my surprise nothing is happening. But then I check wireshark.



No.	Time	Source	Destination	Protocol	Length	Info
4499	46.131172977	10.0.0.6	224.0.0.252	LLMNR	83	Standard query 0xdc2e PTR 211.0.0.10.in-addr.arpa
4500	46.131343968	10.0.0.6	10.0.0.3	DNS	83	Standard query 0x3629 PTR 245.0.0.10.in-addr.arpa
4501	46.131361220	10.0.0.3	10.0.0.6	ICMP	111	Destination unreachable (Port unreachable)
4502	46.131437379	10.0.0.6	10.0.0.3	DNS	83	Standard query 0x99cd PTR 244.0.0.10.in-addr.arpa
4503	46.131927958	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.245? Tell 10.0.0.6
4504	46.132311282	10.0.0.6	10.0.0.3	DNS	109	Standard query 0x6f4a A www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
4505	46.132752631	fe80::70f6:8f7d:6b4…	ff02::1:3	LLMNR	103	Standard query 0x1435 PTR 245.0.0.10.in-addr.arpa
4506	46.132929704	10.0.0.6	224.0.0.252	LLMNR	83	Standard query 0x1435 PTR 245.0.0.10.in-addr.arpa
4507	46.136894223	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.182? Tell 10.0.0.6
4508	46.137004547	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.180? Tell 10.0.0.6
4509	46.137004626	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.202? Tell 10.0.0.6
4510	46.137004658	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.181? Tell 10.0.0.6
4511	46.137004686	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.179? Tell 10.0.0.6
4512	46.158197036	10.0.0.6	10.0.0.3	DNS	109	Standard query 0x6f4a A www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
4513	46.168058166	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.183? Tell 10.0.0.6
4514	46.202221005	10.0.0.6	10.0.0.3	DNS	83	Standard query 0x990b PTR 246.0.0.10.in-addr.arpa
4515	46.202221318	10.0.0.6	10.0.0.3	DNS	83	Standard query 0x8781 PTR 247.0.0.10.in-addr.arpa
4516	46.202671996	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.212? Tell 10.0.0.6
4517	46.203071162	fe80::70f6:8f7d:6b4…	ff02::1:3	LLMNR	103	Standard query 0x34de PTR 212.0.0.10.in-addr.arpa
4518	46.203249224	10.0.0.6	224.0.0.252	LLMNR	83	Standard query 0x34de PTR 212.0.0.10.in-addr.arpa
4519	46.327235408	10.0.0.6	10.0.0.3	DNS	83	Standard query 0x0429 PTR 248.0.0.10.in-addr.arpa
4520	46.339812429	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.192? Tell 10.0.0.6
4521	46.339812759	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.137? Tell 10.0.0.6
4522	46.339812789	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.135? Tell 10.0.0.6
4523	46.340007305	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.134? Tell 10.0.0.6
4524	46.340286203	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.133? Tell 10.0.0.6
4525	46.340631682	PcsCompu_40:bf:3c	Broadcast	ARP	60	Who has 10.0.0.136? Tell 10.0.0.6

I find the same URL from the floss strings I pulled earlier. It looks like the executable is trying to reach out to this URL.

```
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
```