Practical Malware Analysis & Triage

Malware Analysis Report

Ransomware.wannacry.exe

Jan 2022 | Lanzo | v1.0

Table of Contents

Executive Summary

SHA256 hash	24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c

Wannacry is a ransomware malware compiled in C++ that runs on x64 and x86 Windows OS.
The sample consist of a main payload that upacks an additional payload, the malware then encrypts your files then demand ransom payments to unlock those files.
It also have worm capability and try to propagate itself using EternalBlue SMB Exploit.
Symptoms of infection include :

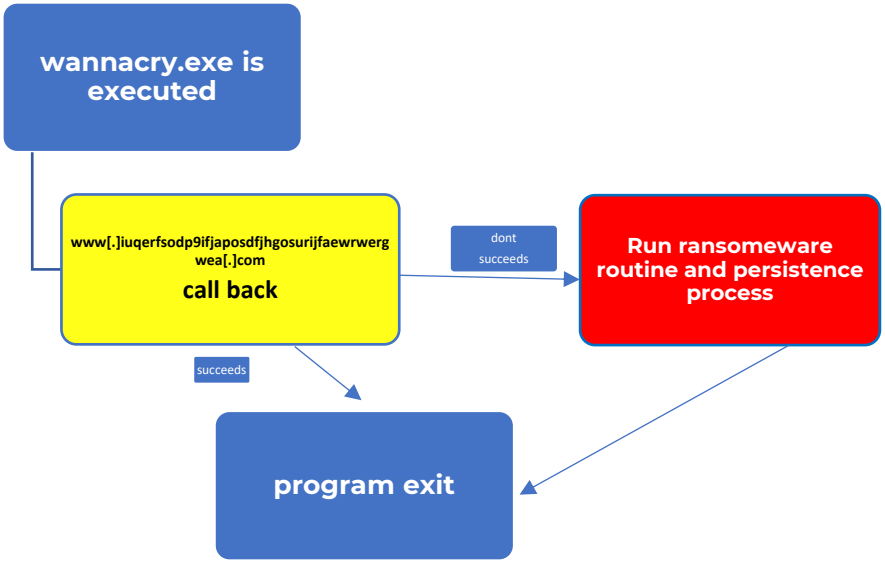
- Files are encrypted using the .WNRY extension
- Changed wallpaper on the infected host
- A Program windows explicitly telling the host is infected and the files encrypted and asking for a ransom with a countdown times and a payment link
- @WanaDecryptor@ executable and a @Please_Read_Me@ files on the desktop
- A hidden directory C:\ProgramData\ and a new service with the same name used for persistence
- A service listening on port 9050 taskche.exe

YARA signature rules are attached in Appendix A. Malware sample and hashes have been submitted to VirusTotal for further examination.

High-Level Technical Summary

Wannacry consists of two parts: an encrypted stage 0 dropper and an unpacked and decoded stage 2 command execution program. It first attempts to contact its callback URL (hxxp[://]www[.]iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea[.]com) as a kill switch if it succeeds the program just terminate otherwise it unpack the next stage, copy the files in the hidden directory, create the persistence process and run the crypto routine.

r



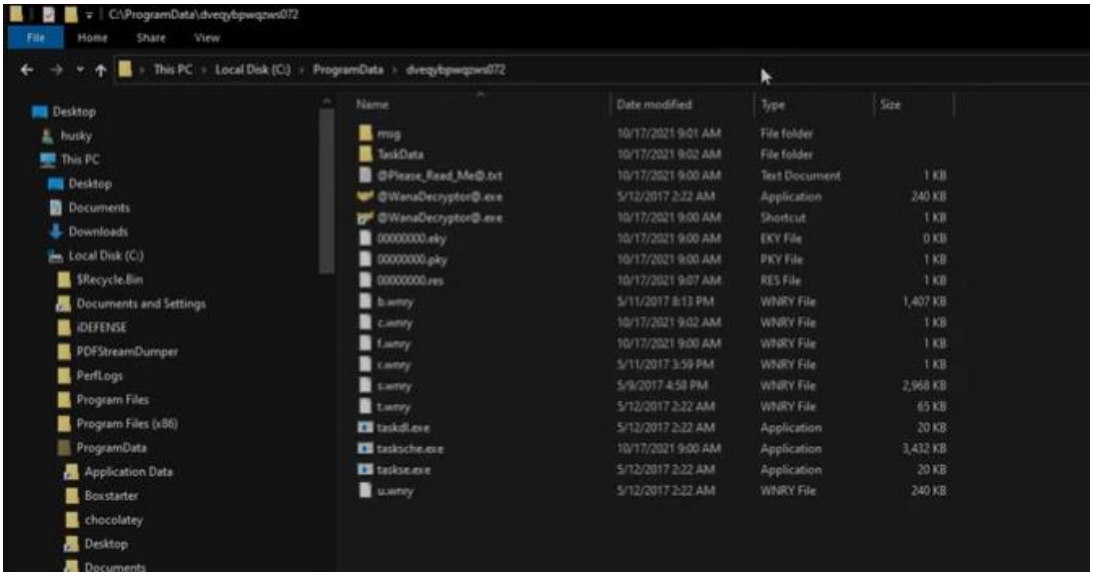
Malware Composition

DemoWare consists of the following components:

File Name	SHA256 Hash
Ransomware.wannacry.exe	24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c

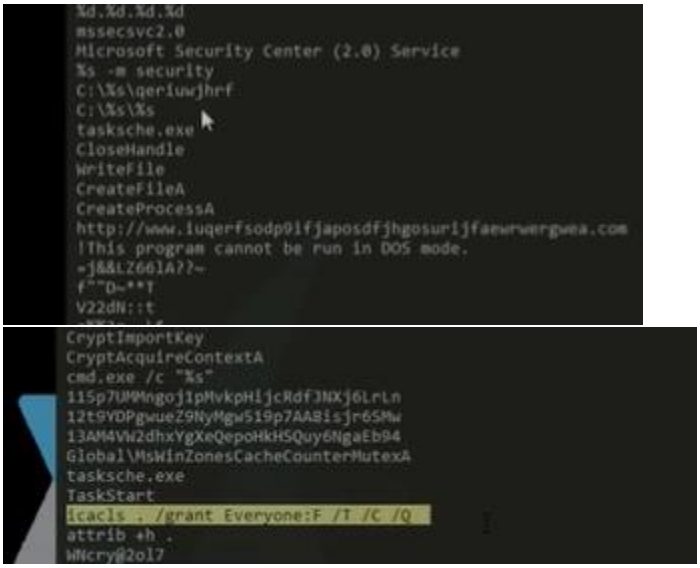
Ransomware.wannacry.exe The initial executable that runs if the callback URL fails.

Hidden files created by the second stage in C:\ProgramData\ with random name.

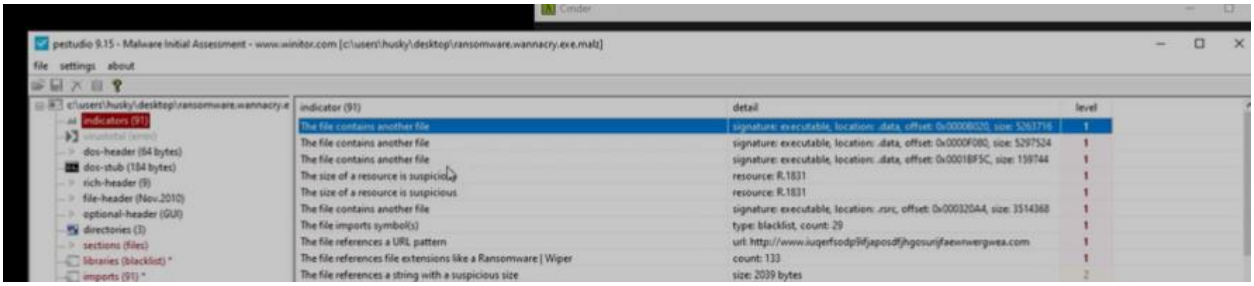


Basic Static Analysis

{Screenshots and description about basic static artifacts and methods}



We can see the URL, a path with %s string replacement, cmd command execution , directory attribute permission modifier and +h hidden attribute , some crypto API call and the suspicious‘look alike’ windows process tasksche.exe.



This sample contain another file packed, many encryption API call and as indicator we have “ The file references file extensions like a Ransomware | Wiper.

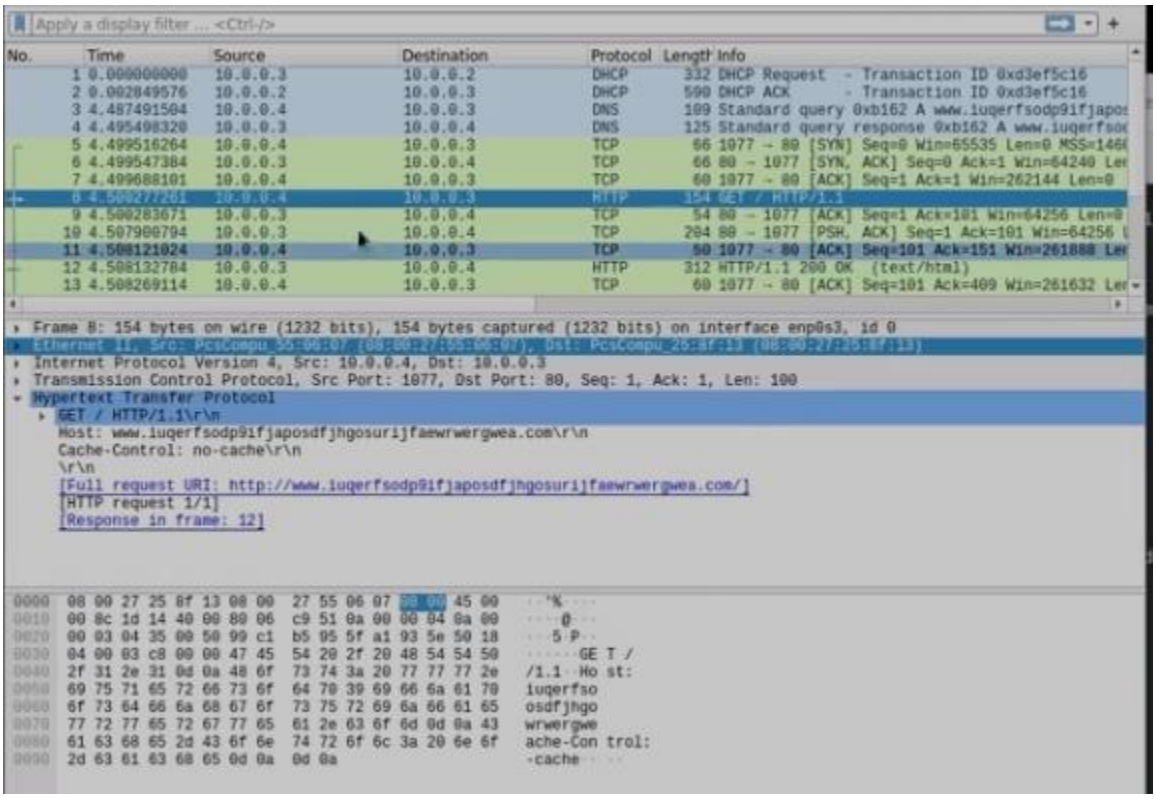
One of the API call is InternetOpenA, probably the API used to reach the killswitch URL.

Basic Dynamic Analysis

{Screenshots and description about basic dynamic artifacts and methods}

Detonation with Remnux as Internet simulator

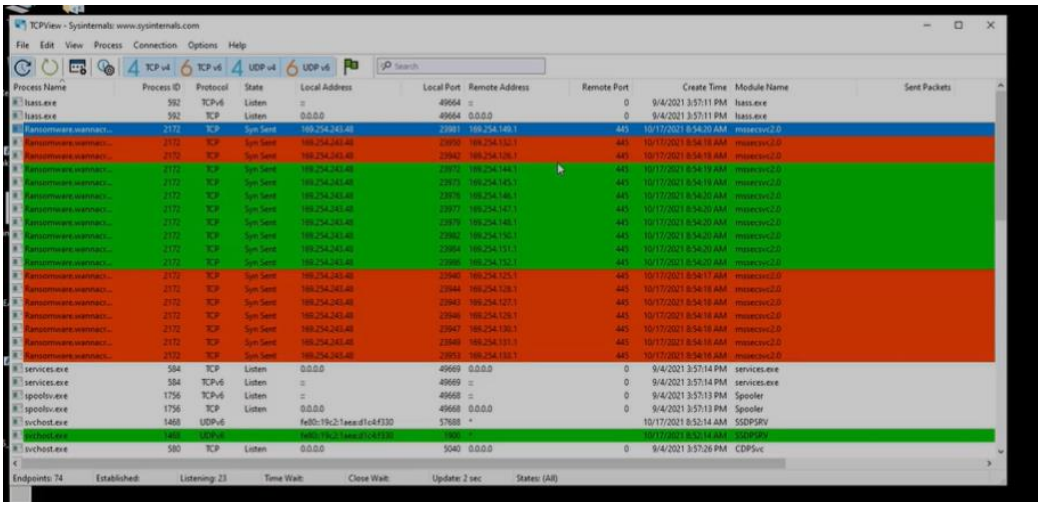
Running the sample with Administrator Privilege with Remnux as internet service emulator using Wireshark and Procmon as soon as the URL respond the sample just terminated execution.



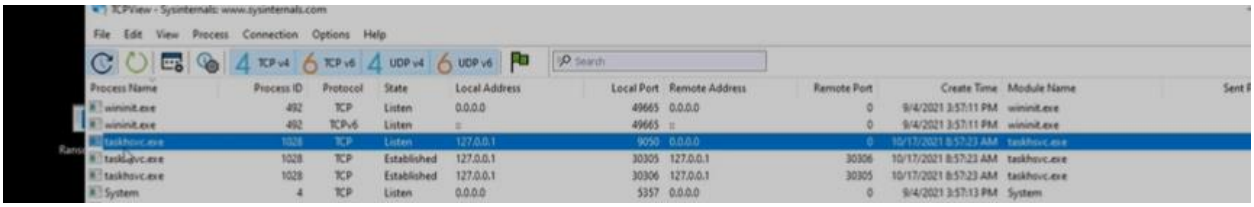
Ransomware.wannacry.exe
Jan 2022
v1.0

Detonation without Internet simulator

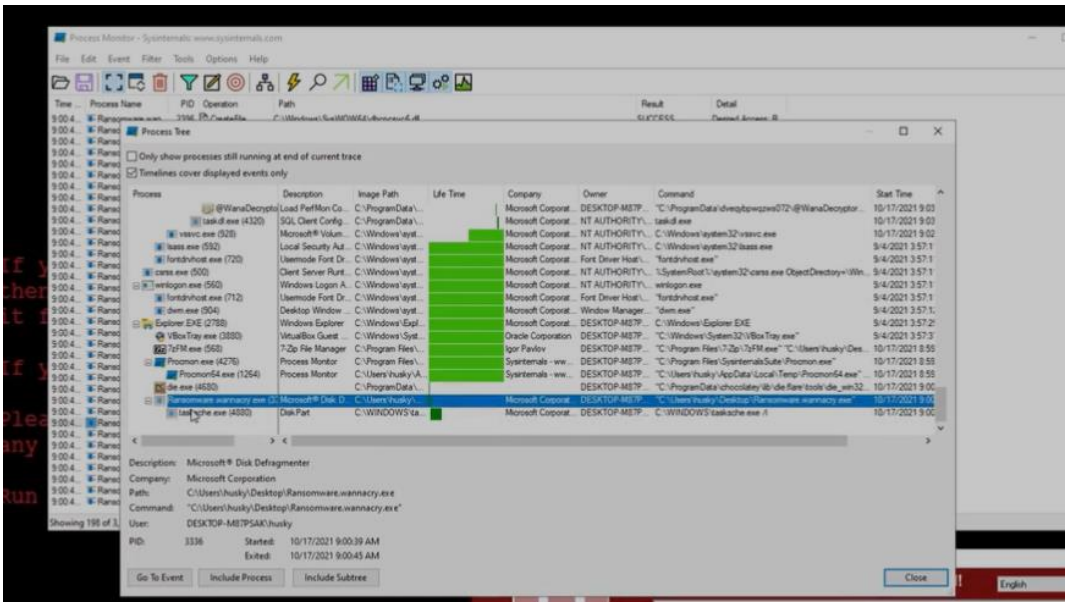
Detonating the sample without an internet simulator we can see the DNS request fail and the malware continue the execution.

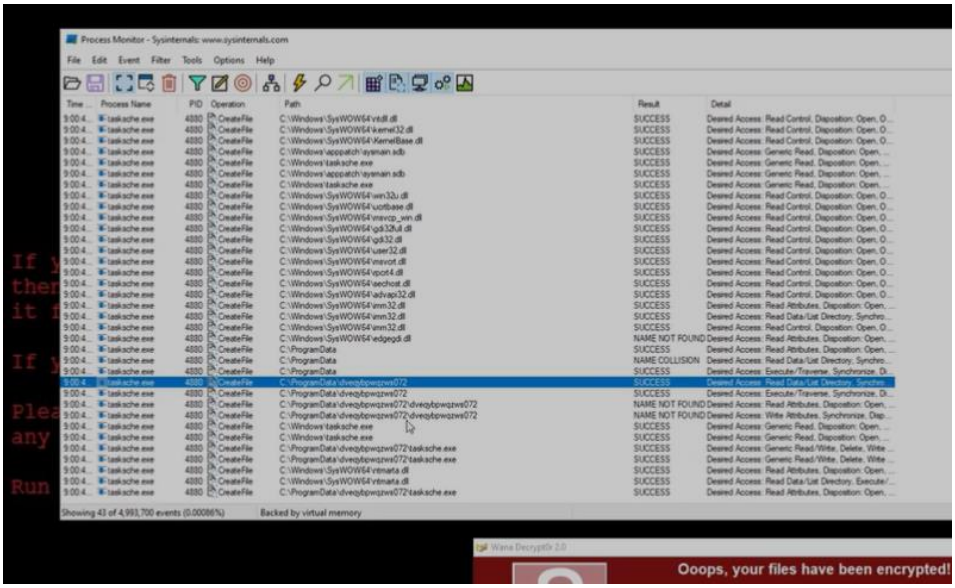


Many call on SMB port 445 to different local host addresses (EternalBlue exploit)



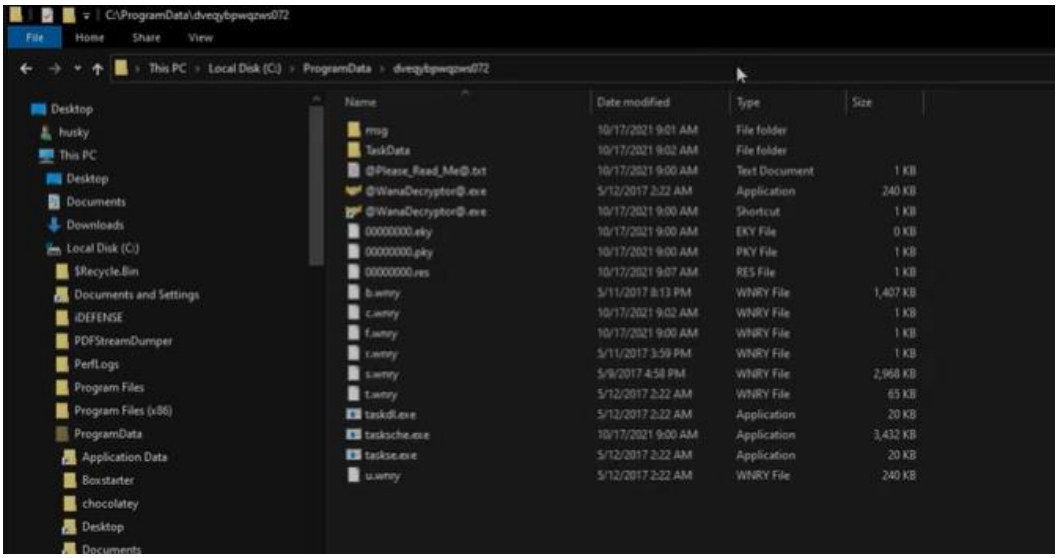
This process listening on port 9050





Ransomware.wannacry.exe
Jan 2022
v1.0

Following the process Tree we can see it create a folder with a random name with all the unpacked files.

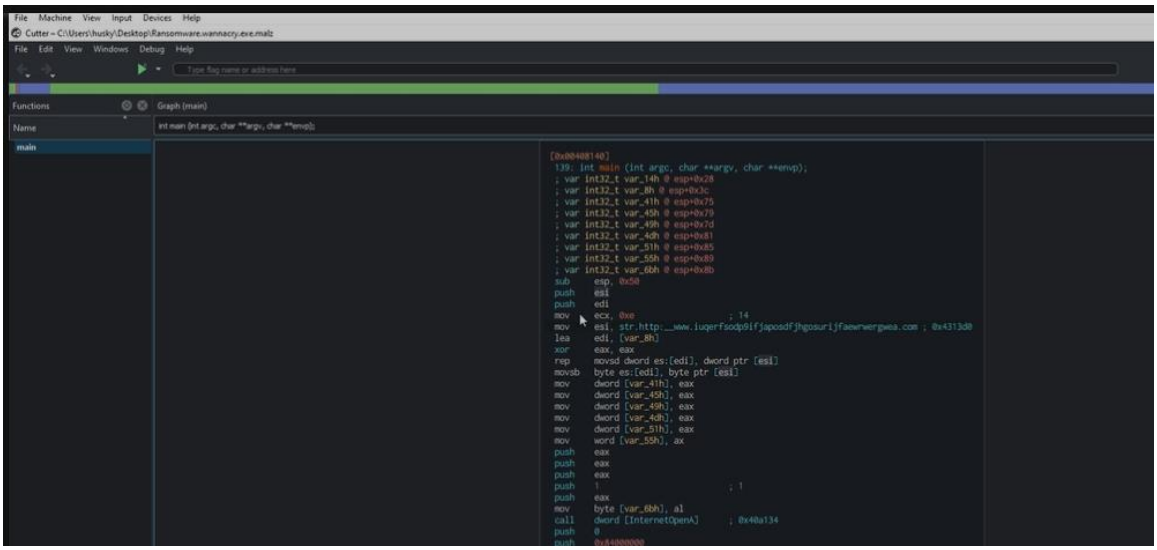


At the end we can see that our wallpaper changed and a program start saying that all our files are being encrypted and we have a limited time to pay the ransom to get them back.

Advanced Static Analysis

{Screenshots and description about findings during advanced static analysis}

In Cutter we can see the string with the URL bein loaded in ESI register and then used in InternetOpenA API call.

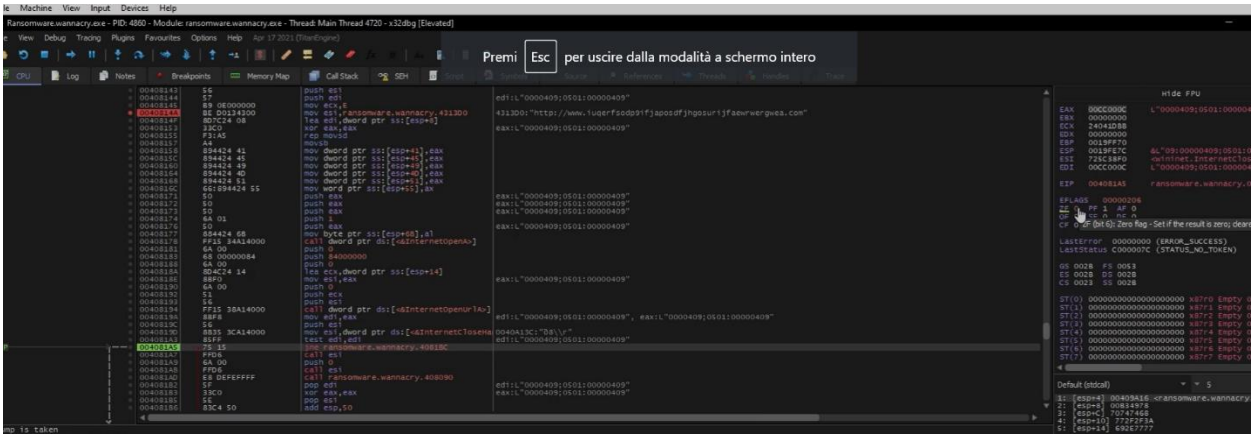


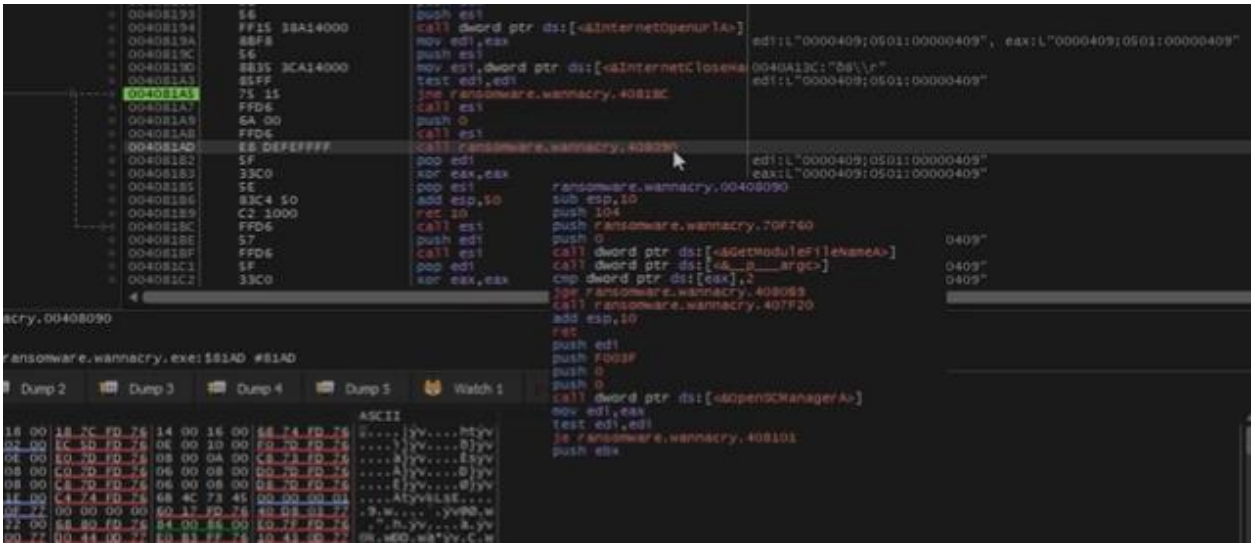
The result of the API Call is tested before taking the JNE in " test edi, edi" before the killer switch, if it is true the program terminate (0x004081bc) otherwise the program before quit call the function 00408090 and run the rest of the program (0x004081a7)

Advanced Dynamic Analysis

{Screenshots and description about advanced dynamic artifacts and methods}

Using x32dbg we can change the execution modifying the ZF zero flag set before the JNE, this way even if the URL is reached and ZF is not set we can change this right before the JNE e execute the rest of the program.





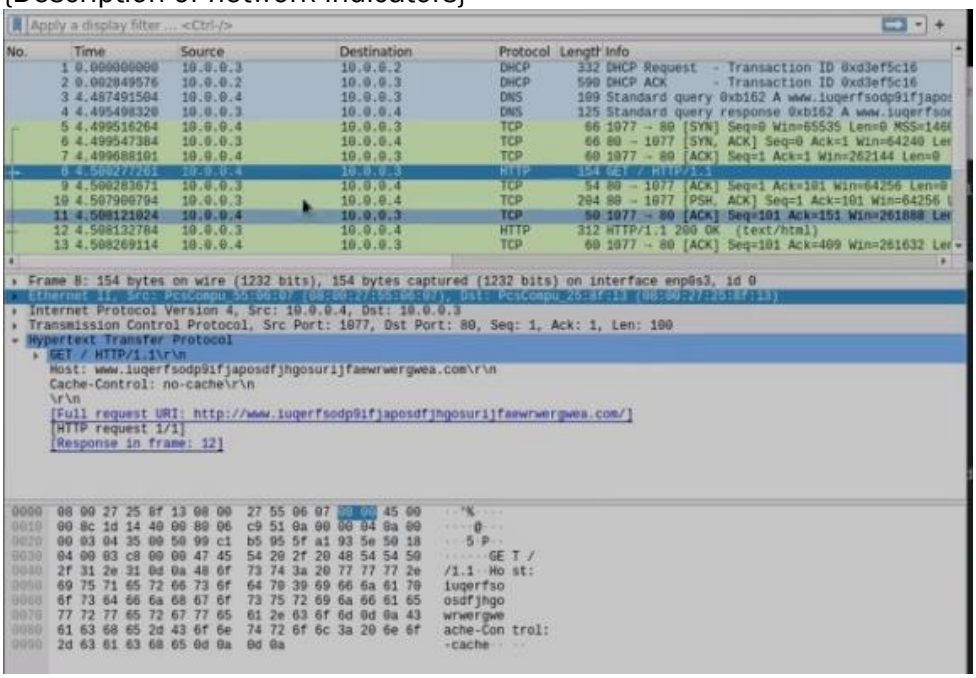
Ransomware.wannacry.exe
Jan 2022
v1.0

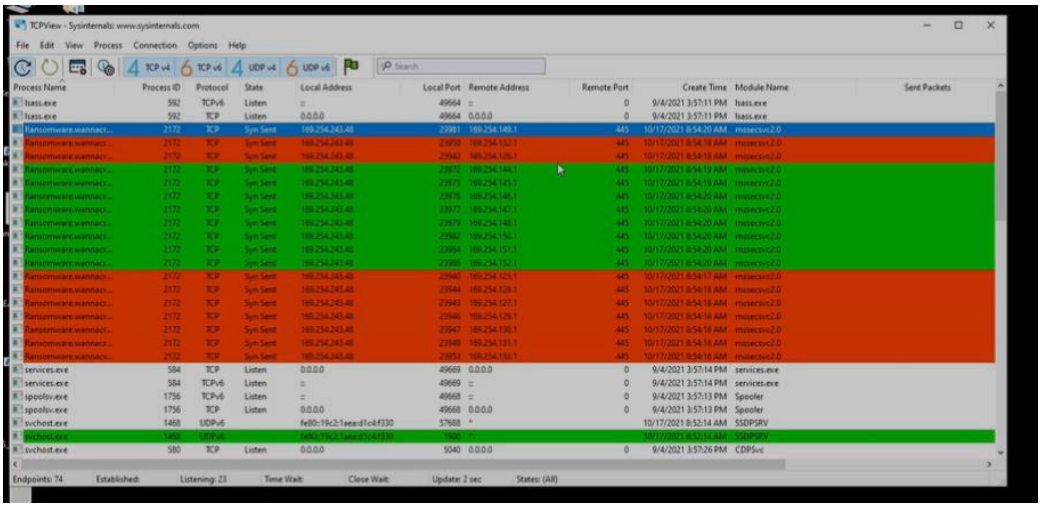
Indicators of Compromise

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

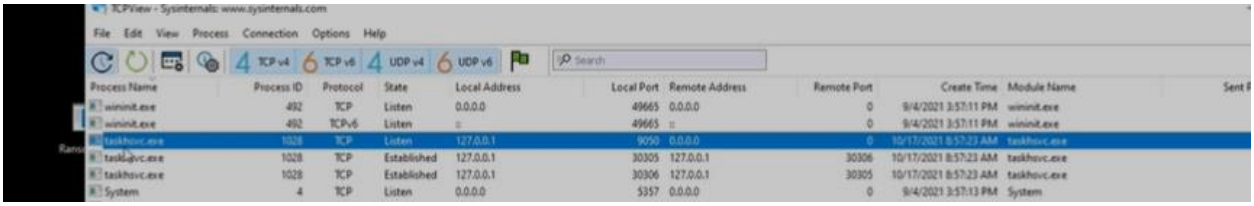
Network Indicators

{Description of network indicators}



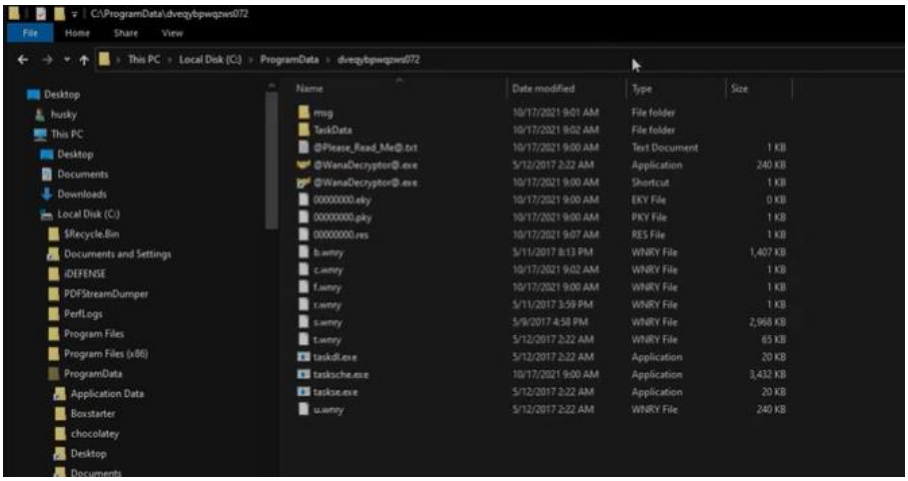


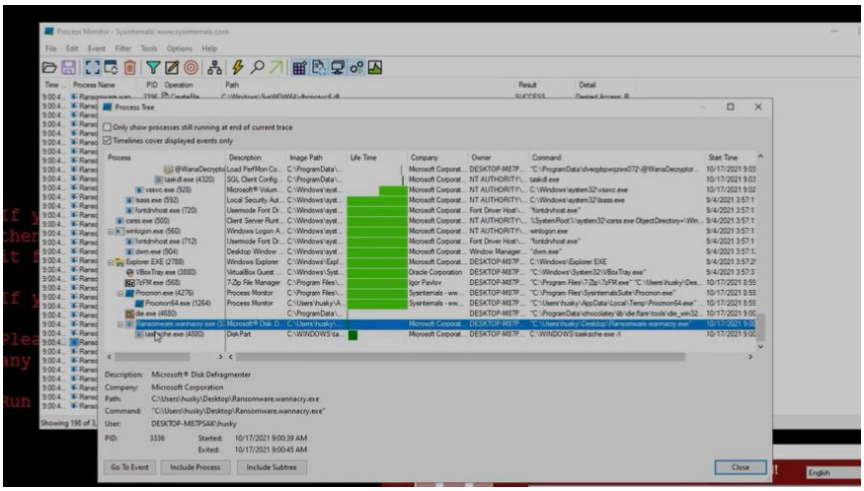
Ransomware.wannacry.exe
Jan 2022
v1.0

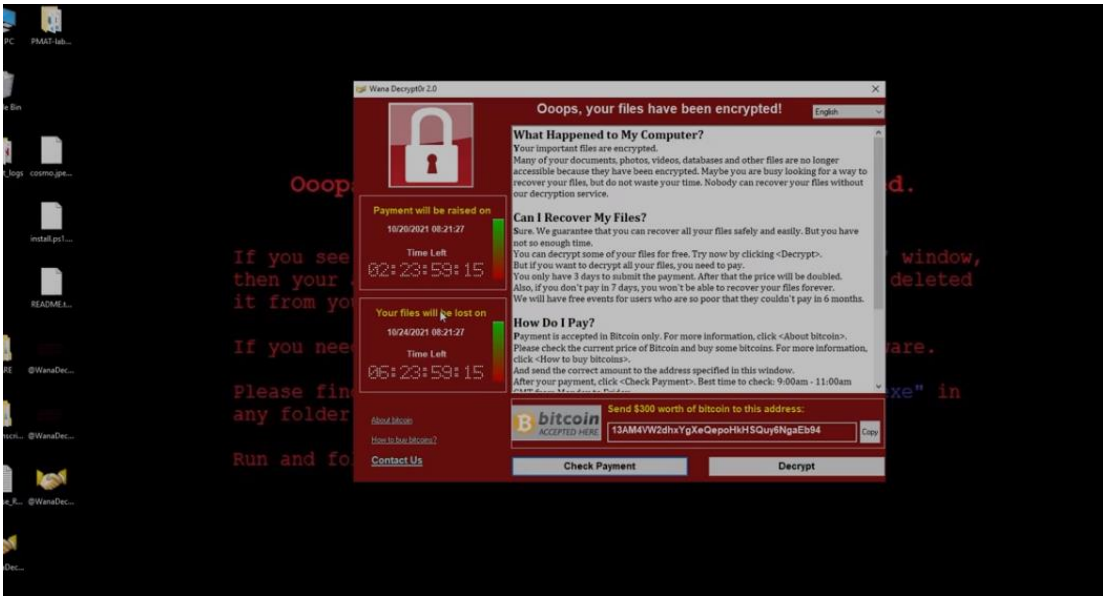


Host-based Indicators

{Description of host-based indicators}







Ransomware.wannacry.exe
Jan 2022
v1.0

Rules & Signatures

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

{Information on specific signatures, i.e. strings, URLs, etc}

%s -m security

C:\%s\qeriuwjhrf

tasksche.exe

icacls . /grant Everyone:F /T /C /Q

WNcry@2ol7

`.WNCRY`

- www[.]iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea[.]com

- - CryptGetRandom
- - CryptAcquireContextA
- - InternetOpenA
- - InternetOpenUrl
- - CreateServiceA
- - ChangeServiceConfig2A

Appendices

A. Yara Rules

```
rule Wannacry_rules {

    meta:
        last_updated = "2022-01-30"
        author = "Lanzo"
        description = "Wannacry rules"

    strings:
        // rules
        $string1 = "www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com" ascii
        $string2 = "tasksche.exe" ascii
        $PE_magic_byte = "MZ"


    condition:
        // Conditions
        $PE_magic_byte at 0 and
        ($string1 and $string2)
}
```

```
C:\Users\lanzo\Desktop
λ yara32 wannacry_rule.yara Ransomware.wannacry.exe.malz -s -w -p 32
Wannacry_rules Ransomware.wannacry.exe.malz
0x313d7:$string1: www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
0x3136c:$string2: tasksche.exe
0x4157c:$string2: tasksche.exe
0x0:$PE_magic_byte: MZ
```

B. Callback URLs

Domain	Port
www[.]iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea[.]com	80

C. Disassembled Code Snippets