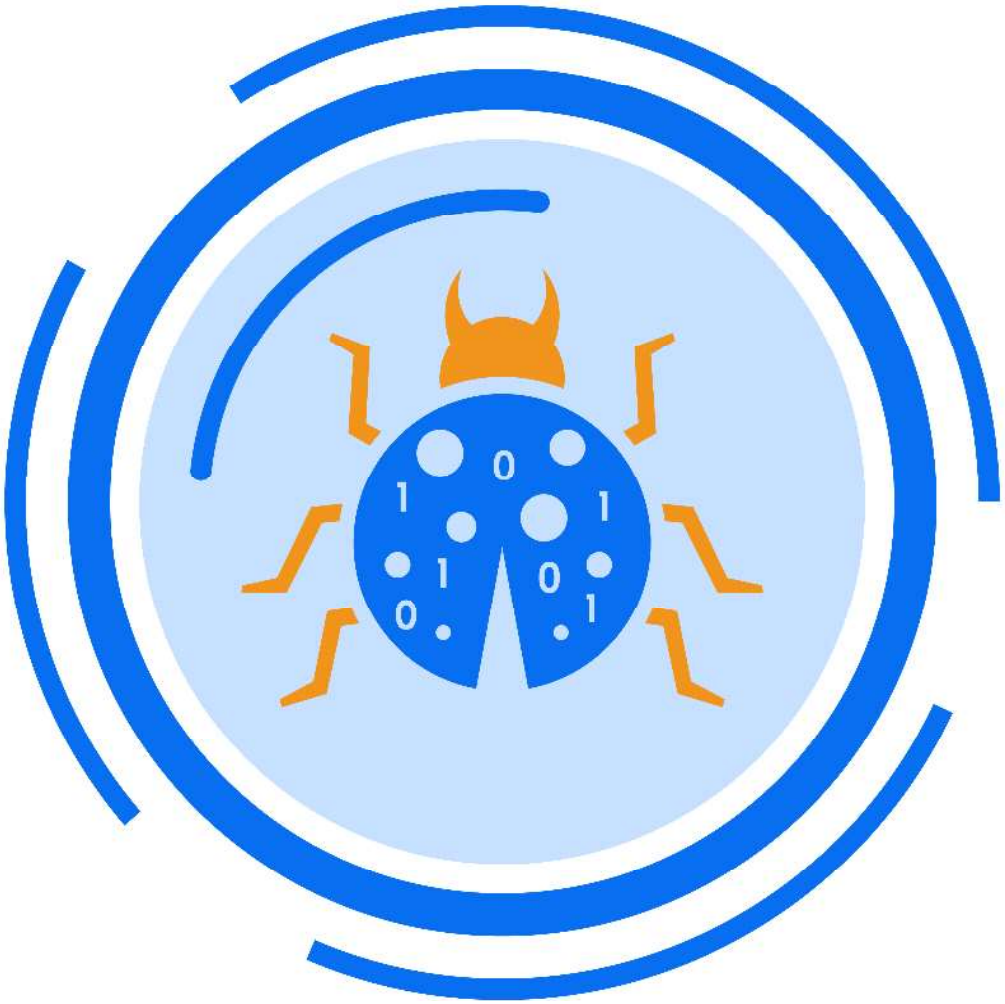


Practical Malware Analysis & Triage

Malware Analysis Report

Ransomware: WannaCry

by: scruffylord

Executive Summary

SHA256 hash	24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c

WannaCry is a widely known ransomware that was initially released in 2017.
Basically Wannacry will encrypt a device's files and also tries to traverse the network to encrypt other devices. WannaCry attempts to locate other hosts through the protocol SMB.

I have the WannaCry malware located in a safe and secure lab. Below will be the full analysis of the malware and the results I have discovered.

Enjoy !

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 is a ransomware cryptoworm, which targets computers running the Microsoft Windows operating system by encrypting (locking) data and demanding ransom

payments in the Bitcoin cryptocurrency. The worm is also known as WannaCrypt, Wana Decrypt0r 2.0, WanaCrypt0r 2.0, and Wanna Decryptor.

Initial Malware detonation

I first wanted to test what exactly happens at the ground level whenever the malware is detonated (without a network connection).

Here are the findings:

After the initial detonation you will see a few icons and files appear on the desktop.

Then only after a few seconds the main message/image appear demanding payment. This payment is demanding $300 worth of bitcoin. After this point all the files are encrypted. Also tools can longer be accessed.

Static Analysis

The following information was received by a variety of tools. Tools utilized in this part of the analysis are **PEview, PEstudio, CAPA**. The tool **FLOSS** was also used to take a look at the strings.

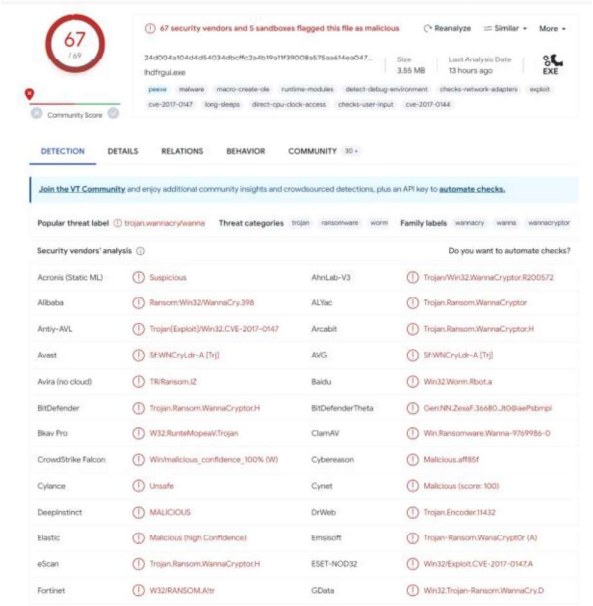
FileName*:*

Ransome.WannaCry.exe

SHA25 hash : ***24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c***

VirusTotal Results:

https://www.virustotal.com/gui/file/24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c



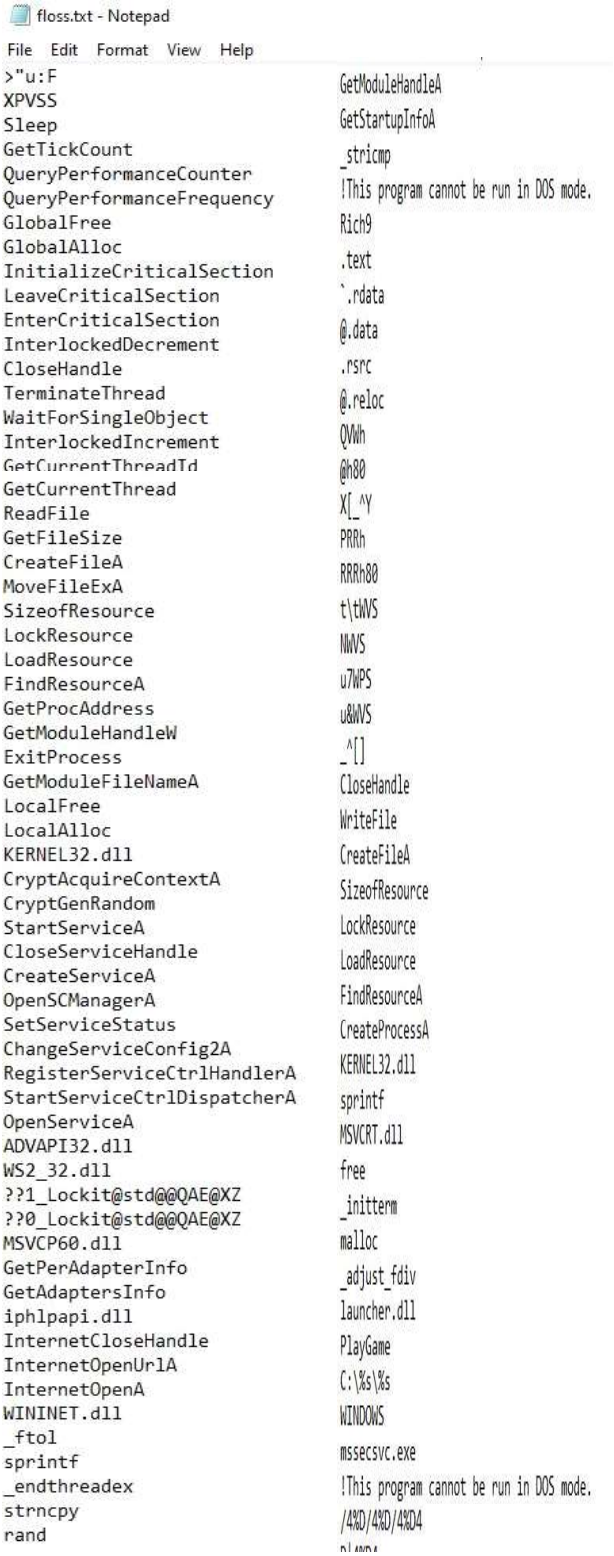
67 vendors have identified this to be Ransmoeware. Including Crowdstrike, Bitdefender and Fortinet. Who are very reliable vendors in the cybersecurity space.

Interesting/Suspicious strings:

- hxxp://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea[.]com (defanged url)
- tasksche.exe
- cmd.exe /c " %s "
- %s -m security
- WanaCrypt0r
- diskpart.exe
- ₩192.168.56.20₩IPC$
- ₩172.16.99.5₩IPC$
- Microsoft Security Center (2.0) Service
- This Program cannot be run in DOS Mode

(This string indicates that there were more executables packed into the binary.)

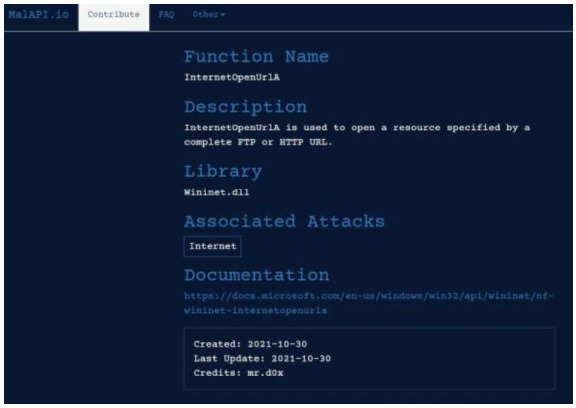
Here is are snippets from the Floss Results :



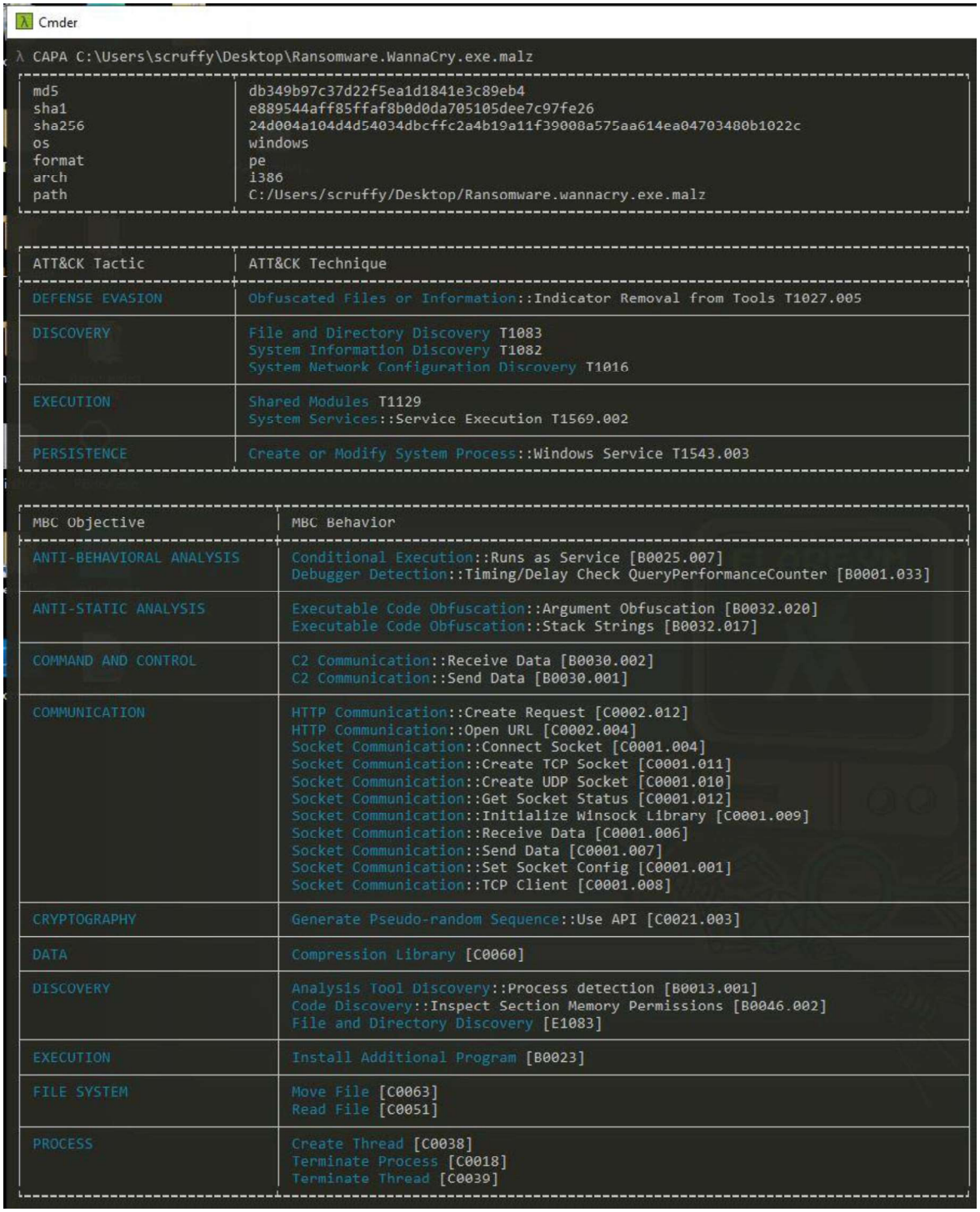
Suspicious API calls:

- IsDebuggerPresent
- StartService
- CreateService
- RegCreateKey
- RegSetValue
- InternetOpenA
- InternetOpenURL

Some of these API calls were compared on the this website: https://malapi.io/



CAPA Results during Static Analysis :

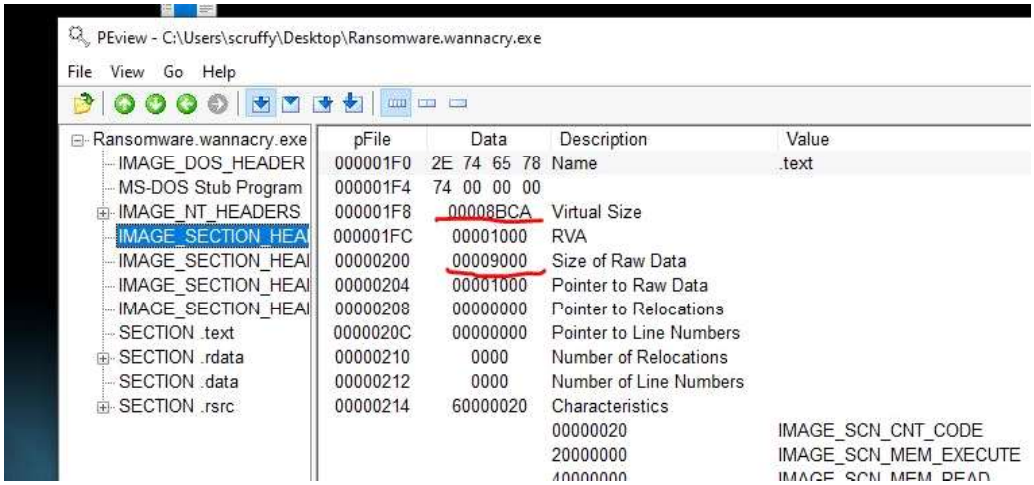

Cmder
λ CAPA C:\Users\scruffy\Desktop\Ransomware.WannaCry.exe.malz
md5 db349b97c37d22f5ea1d1841e3c89eb4
sha1 e889544aff85ffaf8b0d0da705105dee7c97fe26
sha256 24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c
os windows
format pe
arch i386
path C:/Users/scruffy/Desktop/Ransomware.wannacry.exe.malz
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
Executable Code Obfuscation::Argument Obfuscation [B0032.020]
Executable Code Obfuscation::Stack Strings [B0032.017]
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
Analysis Tool Discovery::Process detection [B0013.001]
Code Discovery::Inspect Section Memory Permissions [B0046.002]
File and Directory Discovery [E1083]
EXECUTION
Install Additional Program [B0023]
FILE SYSTEM
Move File [C0063]
Read File [C0051]
PROCESS
Create Thread [C0038]
Terminate Process [C0018]
Terminate Thread [C0039]

Capability	Namespace
reference analysis tools strings	anti-analysis
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
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 on Windows	host-interaction/file-system/read
get number of processors	host-interaction/hardware/cpu
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 on Windows	linking/runtime-linking
linked against ZLIB	linking/static/zlib
inspect section memory permissions	load-code/pe
persist via Windows service	persistence/service

CAPA provided tons of information on the capability of this malware. If you take a look at the ˝ namespace ˛ section you will see very suspicious items, just to name a few :

Anit-analysis
Anti-debugging
Persistence
Data-manipulation

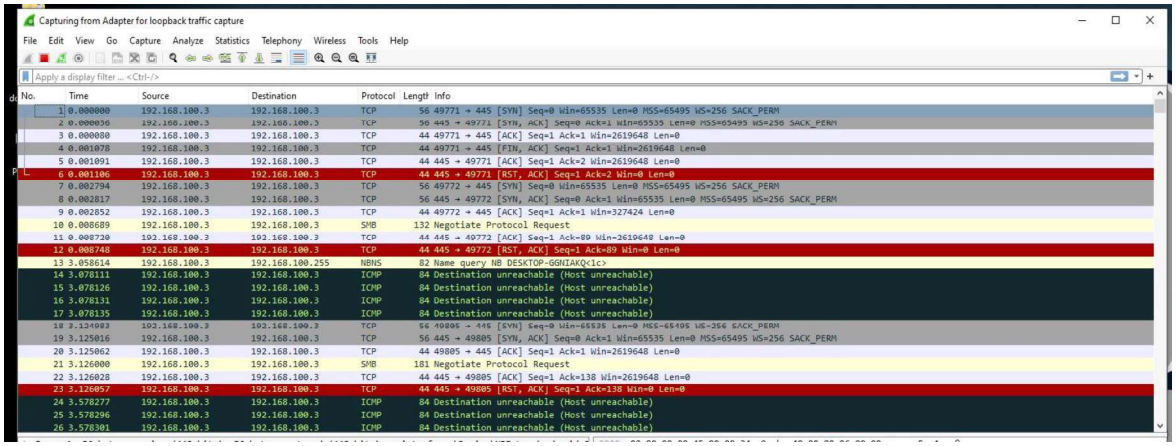
After review of PEview I was able to ascertain that this looks to be Packed Malware, In order to do this you need to compare the Virtual Size to the Size of Raw Data. Packed malware is used by threat actors in order to obfuscate programs so they cannot be analyzed by anti-virus softwares.

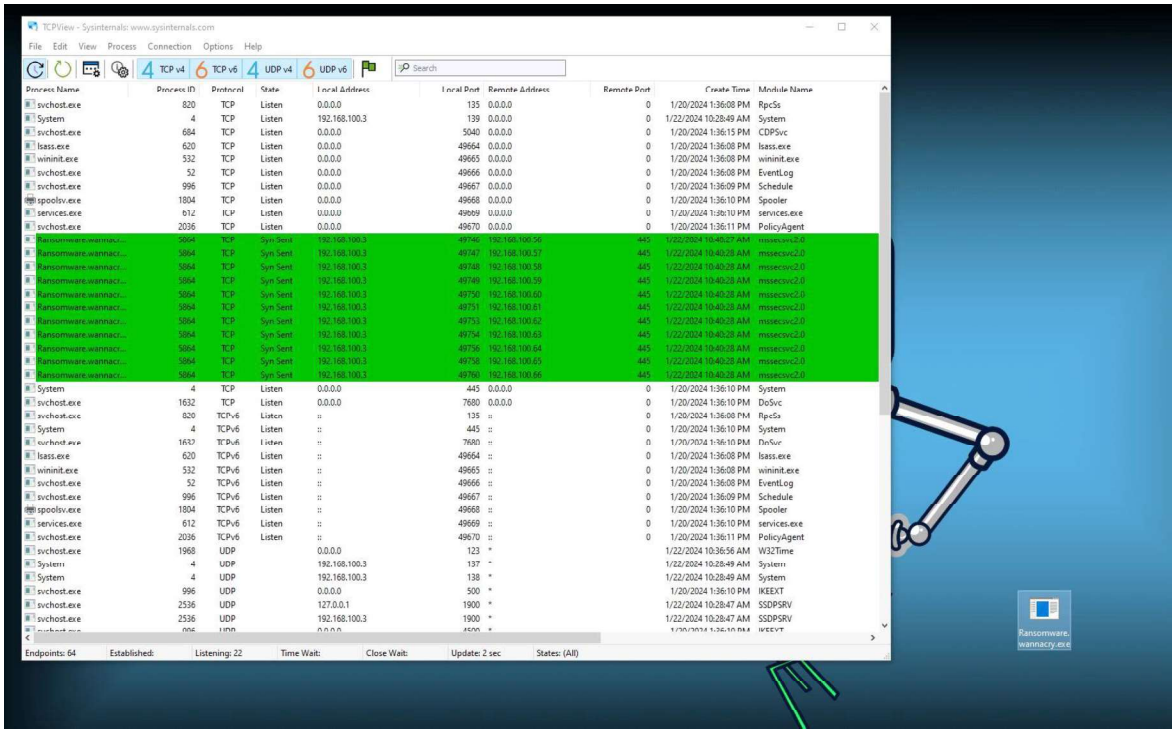


Dynamic Analysis

Now I wanted to detonate the malware with TCPview and Wireshark (Without Internet).

We can see the malware trying to reach out here:

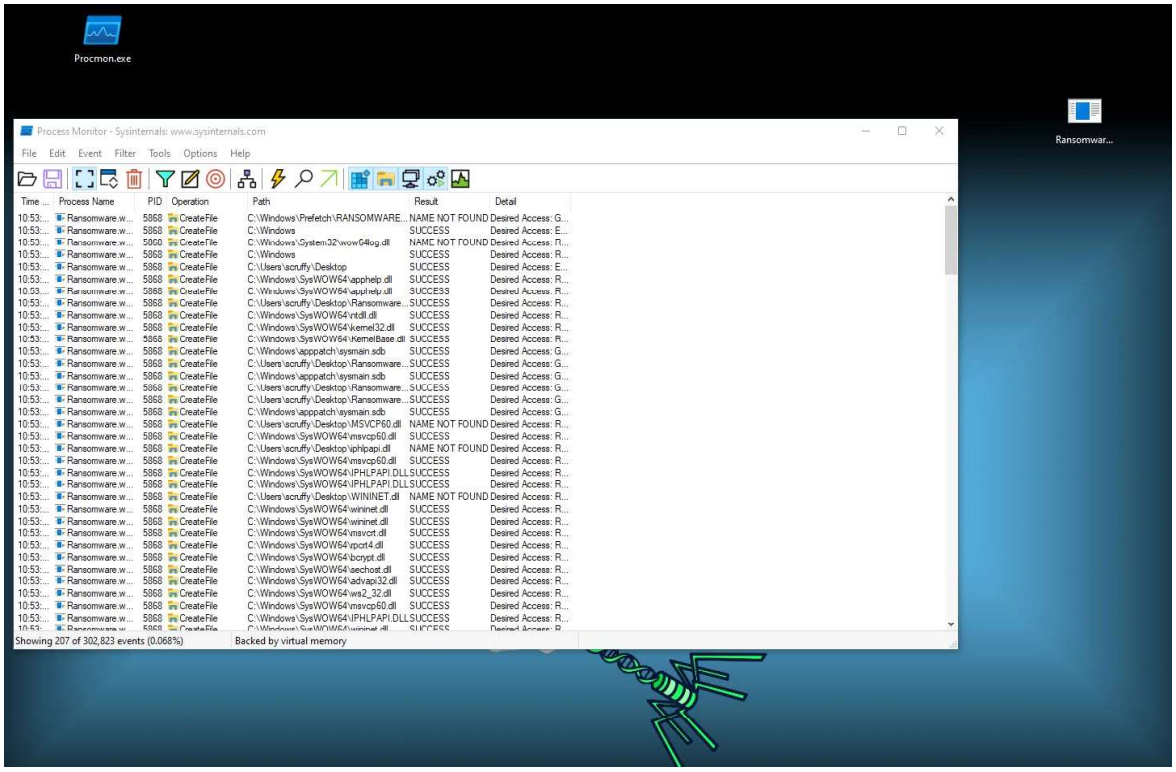




As soon as the malware was detonated it tried to reach out on port 445. The Server Message Block Protocol. This is a network file sharing protocol, and as implemented in Microsoft Windows.

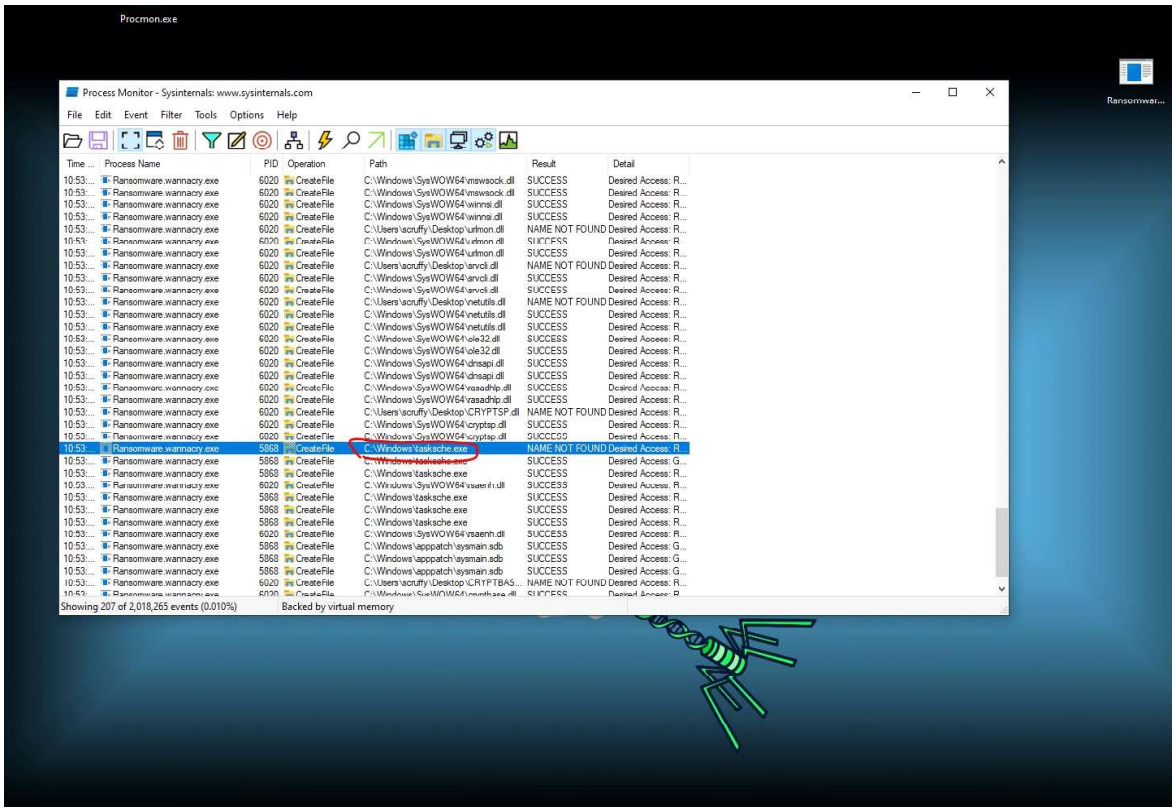
Also in this tool we see a new task created called tasksvc.exe on listening port 9050:

Results and findings from Procmon:



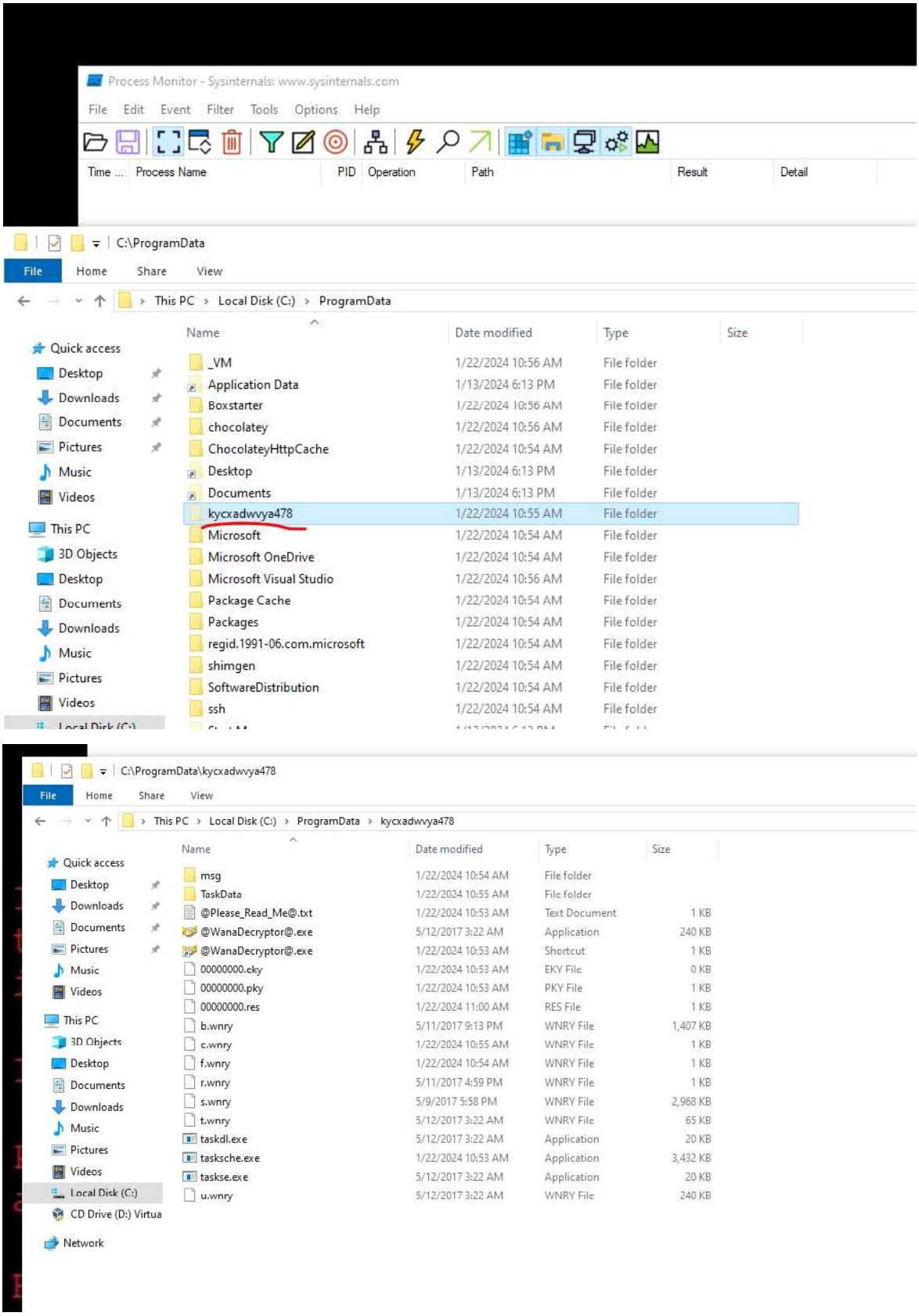
When the malware is detonated we see tons of processes being created and starting to run.

One process that really stands out is C:₩Windows₩tasksche.exe . Most likely this is a second stage payload.



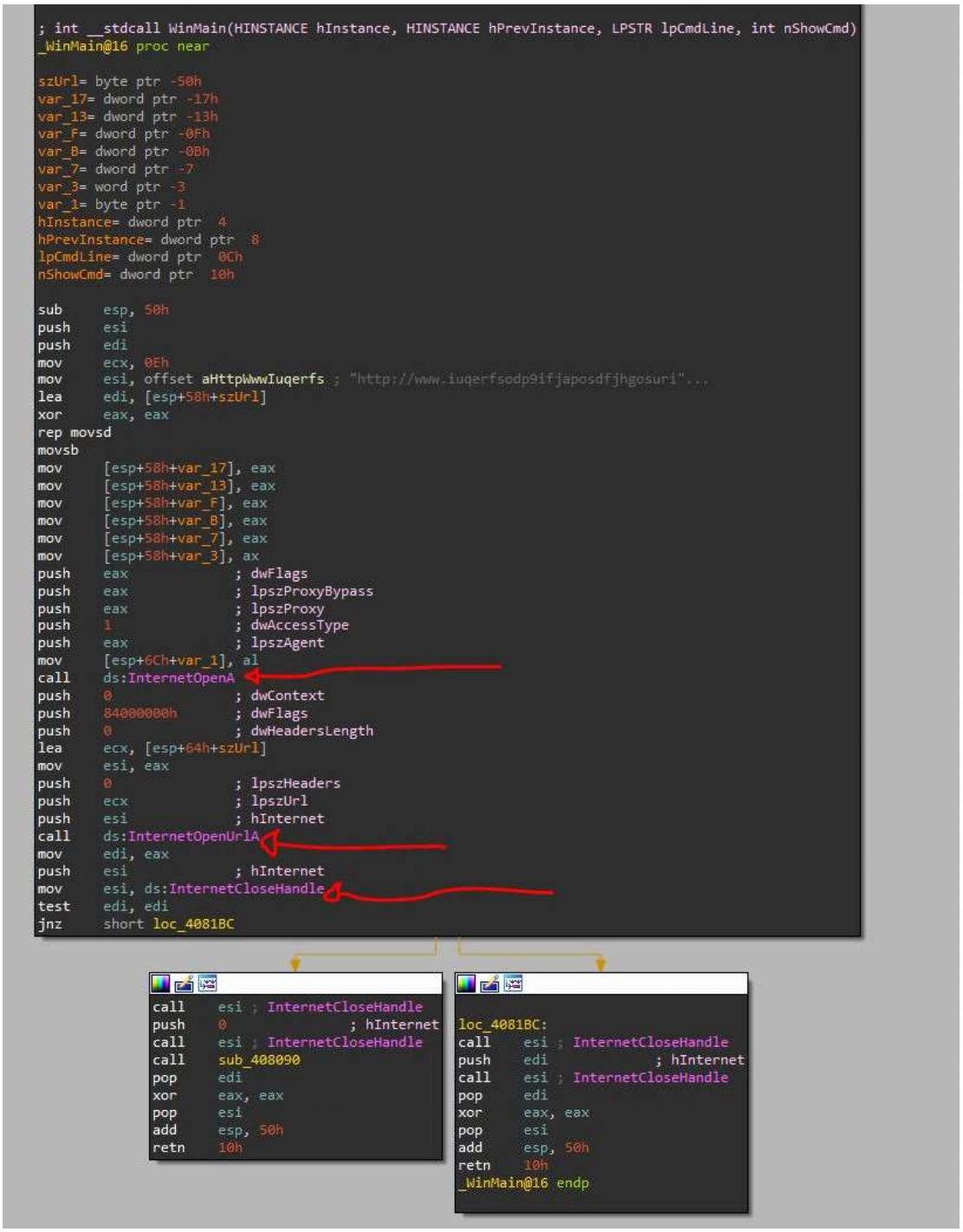
I was able to locate the staging area by finding a strange name directory in procmon:

This is the hidden directory found in the ProgramData directory.


Process Monitor - Sysinternals: www.sysinternals.com
File Edit Event Filter Tools Options Help
Time ... Process Name PID Operation Path Result Detail
C:\ProgramData
File Home Share View
This PC › Local Disk (C:) › ProgramData
Quick access
Desktop
Downloads
Documents
Pictures
Music
Videos
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Name Date modified Type Size
_VM 1/22/2024 10:56 AM File folder
Application Data 1/13/2024 6:13 PM File folder
Boxstarter 1/22/2024 10:56 AM File folder
chocolatey 1/22/2024 10:56 AM File folder
ChocolateyHttpCache 1/22/2024 10:54 AM File folder
Desktop 1/13/2024 6:13 PM File folder
Documents 1/13/2024 6:13 PM File folder
kycxadwvya478 1/22/2024 10:55 AM File folder
Microsoft 1/22/2024 10:54 AM File folder
Microsoft OneDrive 1/22/2024 10:54 AM File folder
Microsoft Visual Studio 1/22/2024 10:56 AM File folder
Package Cache 1/22/2024 10:54 AM File folder
Packages 1/22/2024 10:54 AM File folder
regid.1991-06.com.microsoft 1/22/2024 10:54 AM File folder
shimgen 1/22/2024 10:54 AM File folder
SoftwareDistribution 1/22/2024 10:54 AM File folder
ssh 1/22/2024 10:54 AM File folder
C:\ProgramData\kycxadwvya478
File Home Share View
This PC › Local Disk (C:) › ProgramData › kycxadwvya478
Quick access
Desktop
Downloads
Documents
Pictures
Music
Videos
This PC
3D Objects
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
CD Drive (D:) Virtua
Network
Name Date modified Type Size
msg 1/22/2024 10:54 AM File folder
TaskData 1/22/2024 10:55 AM File folder
@Please_Read_Me@.txt 1/22/2024 10:53 AM Text Document 1 KB
@WanaDecryptor@.exe 5/12/2017 3:22 AM Application 240 KB
@WanaDecryptor@.exe 1/22/2024 10:53 AM Shortcut 1 KB
00000000.eky 1/22/2024 10:53 AM EKY File 0 KB
00000000.pky 1/22/2024 10:53 AM PKY File 1 KB
00000000.res 1/22/2024 11:00 AM RES File 1 KB
b.wnry 5/11/2017 9:13 PM WNRY File 1,407 KB
c.wnry 1/22/2024 10:55 AM WNRY File 1 KB
f.wnry 1/22/2024 10:54 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 1/22/2024 10:53 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

Now I wanted to take a look at the program itself using the tool Tool ˜ IDA ˜

Here are the results:



Here we can see the main function and the APIs being called. We can see the URL hxxp://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea[.]com loaded into the beginning of the program as well.

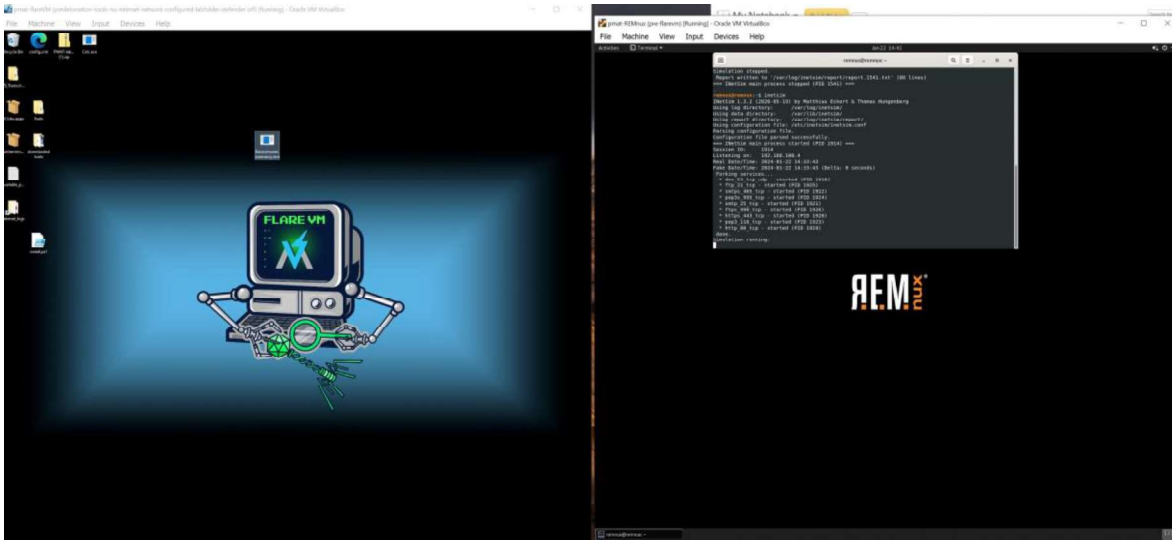
The outcome of the malware being executed depends on two things:

If the result of the API call from InternetOpenURL is successful then the malware will go to a specific location in the Memory.

If the result of the API call from InternetOpenURL is unsuccessful then the malware will go to a different location. This Function call will detonate the payload and still carry out every other part of the encryption. We have seen this side of the payload already.

Now I wanted to test the results of running the payload with an internet connection.

When the malware executable is run there are no changes to our host desktop. Unlike our first attempt we see files being created and the ransomware note pops up. Please review the screenshot below:



There are no created messages or files after detonation that we can see. Wanna cry will wait to access the URL hxxp://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea[.]com (this is the url defanged) before it begins the encryption payload.

Like discussed before, If the payload cannot reach that URL the malware will detonate and encrypt the device. Now after researching WannaCry it is not known for sure why the payload does not detonate after connecting to that URL. Though it is expected that the malware authors did this in order to hide the hide from analyst.

Appendices

- Yara Rules

```
rule RansomWare_WannaCry{
    meta:
        last_update = "2024-01-22"
        author = "scruffylord"
        description = "Yara rule for WannaCry"


    strings:
        $string1 = "%s -m security" ascii
        $string2 = "WNcry@2017" ascii
        $string3 = "wnry" ascii
        $string4 = "WanaCrypt0r" ascii
        $string5 = "%s -m security" ascii
        $PE_magic_byte = "MZ"
        $url = "www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com" ascii
        $exe = "tasksche" ascii
        $exe = "diskpart"
        $address = "\\192.168.56.20\\IPC$" fullword wide

    condition:
        $MZ_magic_byte at 0 and
        any of them
```