Icon

Description automatically generated

Practical Malware Analysis & Triage

Malware Analysis Report

Wannacry.exe - Malware

July 04 | Alex Macenas | v1.0

Table of Contents

[Executive Summary 3](#__RefHeading___Toc1263_410516257)

[High-Level Technical Summary 4](#__RefHeading___Toc1265_410516257)

[Malware Composition 5](#__RefHeading___Toc1267_410516257)

[Basic Static Analysis 7](#__RefHeading___Toc1269_410516257)

[Basic Dynamic Analysis 8](#__RefHeading___Toc1271_410516257)

[Advanced Static/Dynamic Analysis 10](#__RefHeading___Toc1273_410516257)

[Indicators of Compromise 13](#__RefHeading___Toc1275_410516257)

[Network Indicators 13](#__RefHeading___Toc1277_410516257)

[Host-based Indicators 14](#__RefHeading___Toc1279_410516257)

[Rules & Signatures 15](#__RefHeading___Toc1281_410516257)

[Appendices 16](#__RefHeading___Toc1283_410516257)

[A. Yara Rules 16](#__RefHeading___Toc1285_410516257)

[B. Callback URLs 16](#__RefHeading___Toc1287_410516257)

# Executive Summary

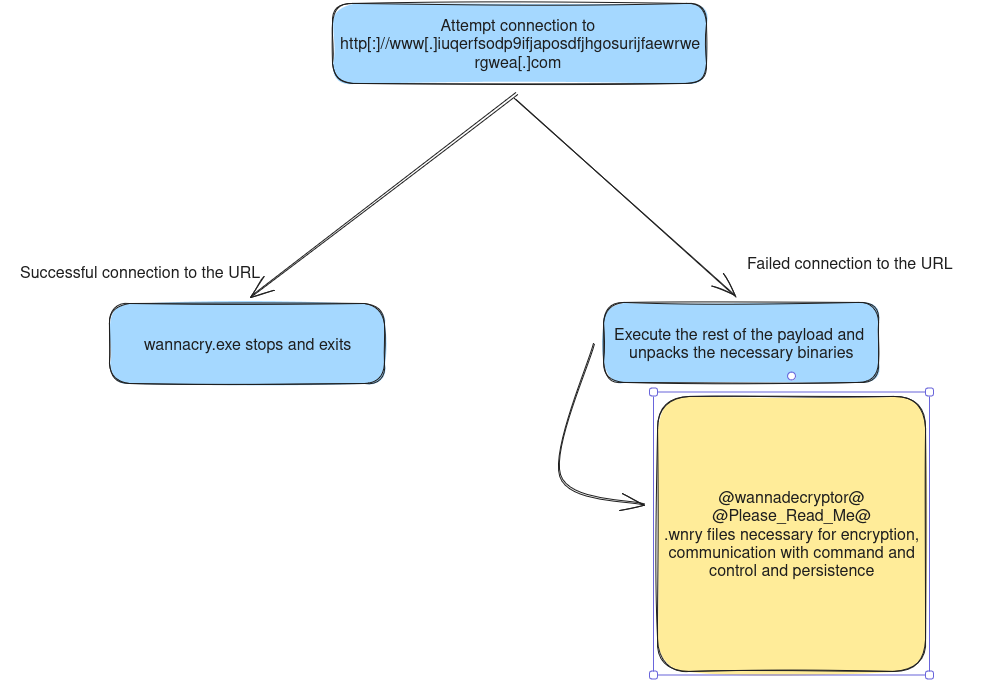
|  |  |
| --- | --- |
| SHA256 hash | 24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c |

Wannacry is a ransomware that appeared on the year 2017, It is specifically tailored for Windows operating system by abusing the Server Message Block protocol vulnerability called EternalBlue and propagates by abusing the same protocol. Symptoms of infections are files are locked and a .wncry extension, constant appearance of the wannadecrypto binary, and a persistence mechanism found on [C:\ProgramData](../../../../../C:/ProgramData)\<random\_folder\_name>.

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 dropper malware it unpacks all necessary binaries and dependencies for encrypting the target files, binary responsible for communicating back to command and control and setting up persistence. It first attempts to contact it’s callback URL (http[:]//www[.]iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea[.]com) then once a successful connection is made the malware will stop running and exits, most likely checking if it is in a sandbox, if not then the rest of the payload is executed.



# Malware Composition

Wannacry consists of the following components:

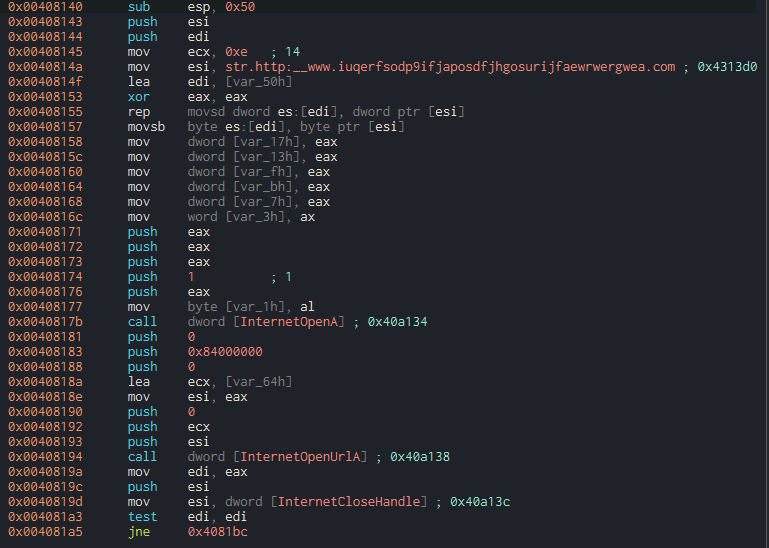
***The dropper malware***

*This is the initial program that is sent to the victim.*

***The kill switch mechanism***

*It is a code that is responsible for checking if the environment is a sandbox environment by testing and establishing a HTTP connection to the said URL:*

*asdasdasd*

Figure 1: Assembly view of binary establishing connection

**The Ransomware Mechanism**

It is a code that is responsible for encrypting the files of the victim and as well as the decryption module that proves that the files can be decrypted

**The Propagation Mechanism**

It is a code that is responsible for propagating to other devices on the network by exploiting the SMBv1 protocol.

# Basic Static Analysis

**Notable Strings and Libraries:**

- **http[:]//www[.]iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea[.]com –** the callback URL to determine whether to run the rest of the payload or not

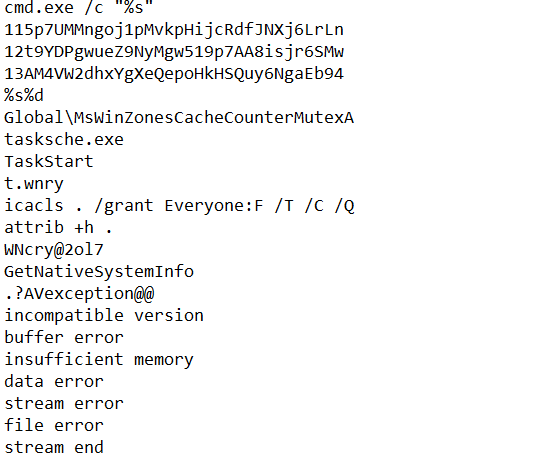
- **.wnry binaries** – the necessary files for encryption, decryption, communication with command and control and establishing persistence

- **InternetOpenURL –** the library or function that is used to establish connection to a URL.

- **SMB –** most likely to connect and exploit vulnerable SMBv1

- **Cryptography libraries –** the library or function that is used to encrypt and decrypt files on the victim

- **recv and send –** most likely communication for the command and control



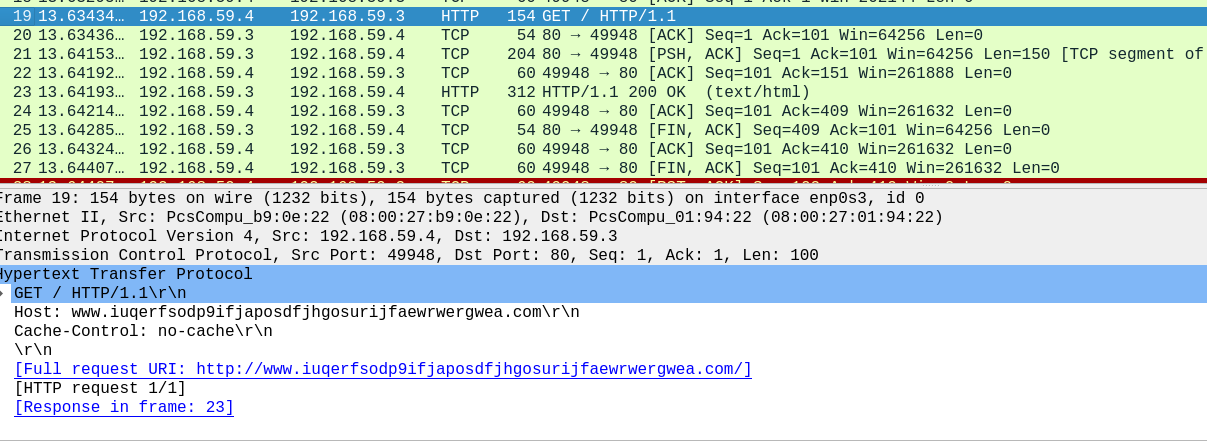
# Basic Dynamic Analysis

{Screenshots and description about basic dynamic artifacts and methods}

Initial detonation:

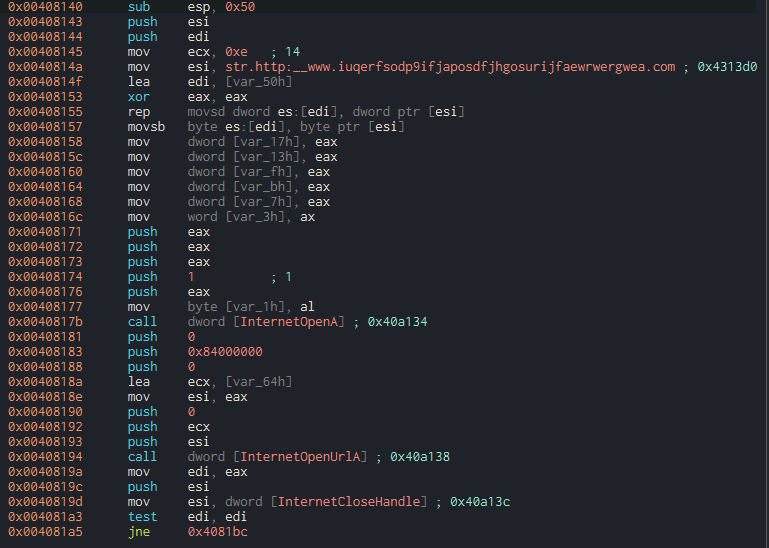
Figure 2: Initial detonation, encrypted files have a .wncry extension

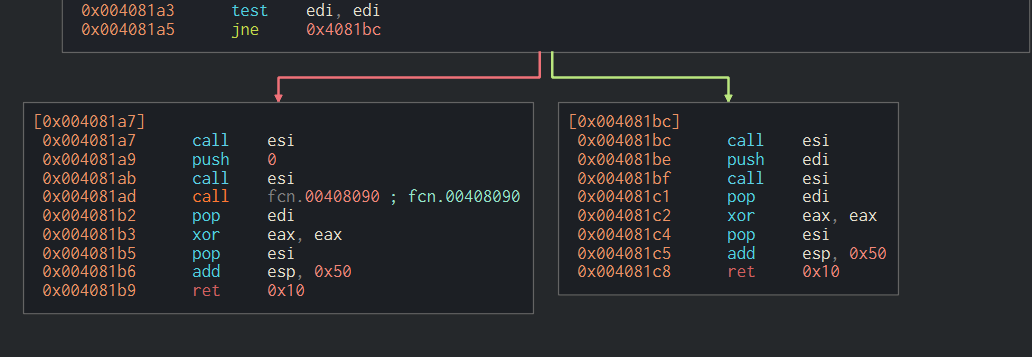
At **Figure 2** files are encrypted and the spawn of the @wannadecryptor@ binary, the change of wallpaper with a message that the victim files is encrypted and requires payment.

Figure 3: Attempts to communicate with the URL

# Advanced Static/Dynamic Analysis

**Figure 4** is the assembly view of the code responsible for establishing communication with the URL.

Figure 4: Attempt and establish with the specified URL

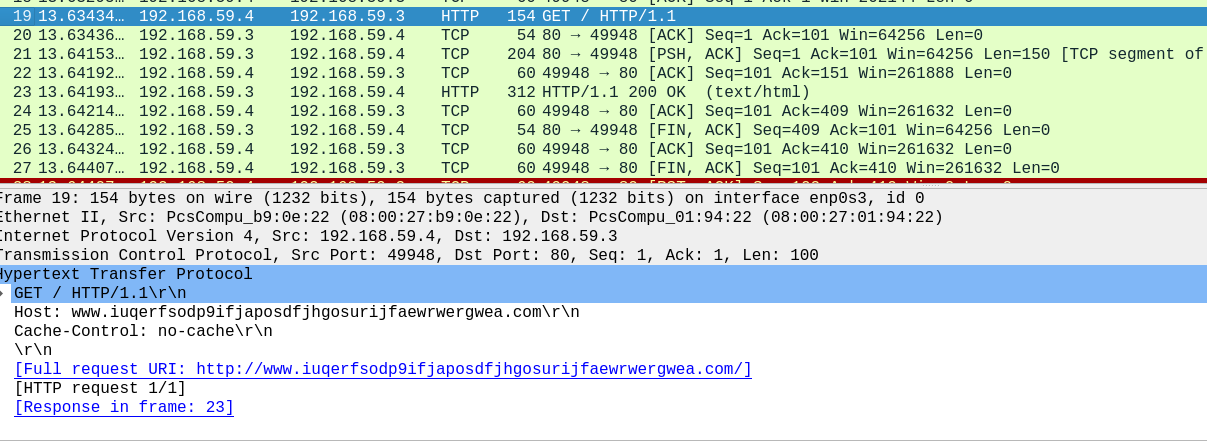
Figure 5: Continuation of Figure 4

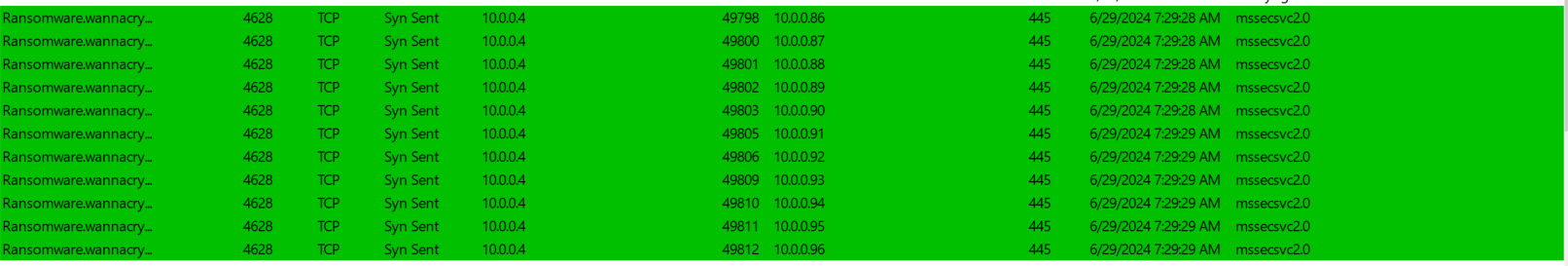
**Figure 5** is the assembly view of the code responsible for determining whether to continue with the rest of the payload or stop completely.

# Indicators of Compromise

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

## Network Indicators

Figure 6: Wireshark capture of communication with the URL

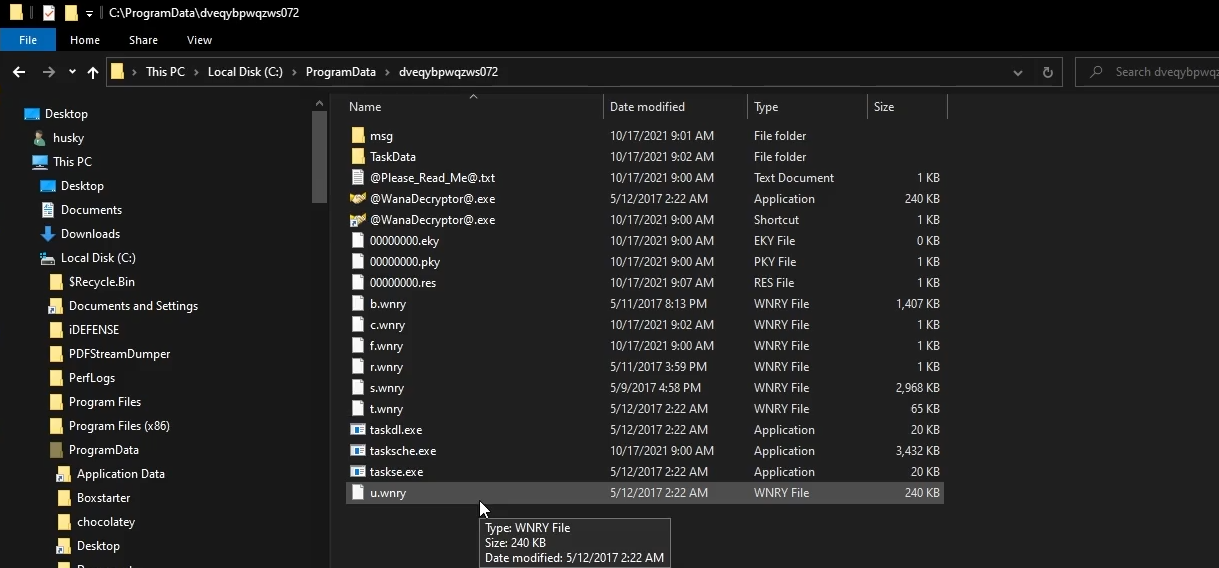
Figure 7: TCPview of wannacry propagating

**Figure 7** is after successful detonation, wannacry determines the subnet of the physical host and tries to attempt and propagate to any host that is vulnerable by contacting every possible host IP address on the subnet.

*Fig 4: WireShark Packet Capture of stage 2 executable download.*

## Host-based Indicators

All the unpacked binaries after the initial detonation, note the name of the directory, it can also be found as well running by using Task Manager → Services → <name\_of\_directory>

Figure 8: The entire payload unpacked

# Rules & Signatures

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

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

# Appendices

## Yara Rules

rule wannacry {

    meta:

*date\_created =* *"July 04, 2024"*

*author =* *"Alex Macenas"*

      description = "A rule for wannacry"

    strings:

*$host\_based\_indicators\_1* *=* *"@wannadecryptor@"*

*$host\_based\_indicators\_2* *=* *"@Please\_Read\_Me@"*

*$smb* *=* *"SMB"*

*$file\_extension* *=* *".wnry"*

    condition:

       any of (*$*host\_based\_indicators\_1, *$*host\_based\_indicators\_2, *$*smb, *$*file\_extension) and *$*PE\_magic\_byte

}

## Callback URLs

|  |  |
| --- | --- |
| **Domain** | **Port** |
| **http[:]//www[.]iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea[.]com** | 80 |
|  |  |
|  |  |