WannaCry Analysis & Triage Report

Summary

In May 2017, the global cybersecurity landscape was shaken by WannaCry, a ransomware that rapidly spread worldwide. Exploiting the EternalBlue vulnerability in Windows systems, it infected over 200,000 computers across 150 countries.

WannaCry operated with a straightforward yet devastating approach: it encrypted files on infected machines and demanded ransom payments in Bitcoin for decryption.

High-profile entities such as the UK's National Health Service and corporations like FedEx found themselves grappling with encrypted data and the looming threat of permanent loss.

Written primarily in C++, WannaCry comprises two main executables: mssecsvc2.0.exe and tasksche.exe. Upon activation, it first verifies internet connectivity to its callback domain. If communication fails, it proceeds to encrypt files on the compromised system, periodically displaying ransom demands in exchange for decryption keys.

SHA256:
2F3FC51546ADA848DFC8E775554C0DE3689D6FAE7BA4BF3D40E3C8DEC68B277B

Technical Overview

WannaCry operates through several stages:

It attempts to establish connectivity with its callback domain hxxp://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com. If successful, it terminates.

Upon failure to connect, it proceeds to create and initiate a new service named mssecsvc2.0, masquerading as the "Microsoft Security Center 2.0 Service." This service covertly scans the local network on port 445 to identify systems vulnerable to the EternalBlue exploit.

It unloads all malicious payloads into the directory "C:\ProgramData[random hash]."

Static Analysis

Import Address Table

Win32 APIs related to cryptography, sockets, and services

imports (91)	flag (28)
StartServiceCtrlDispatcherA	x
ChangeServiceConfig2A	x
CreateServiceA	x
QueryPerformanceFrequency	x
3 (closesocket)	x
16 (recv)	x
19 (send)	x
8 (htonl)	x
14 (ntohl)	x
115 (WSAStartup)	x
12 (inet_ntoa)	x
10 (ioctlsocket)	x
18 (select)	x
9 (htons)	x
23 (socket)	x
4 (connect)	x
11 (inet_addr)	x
GetAdaptersInfo	x
InternetOpenA	x
InternetOpenUrlA	x
InternetCloseHandle	x
MoveFileExA	x
GetCurrentThreadId	x
GetCurrentThread	x
CryptGenRandom	x
CryptAcquireContextA	x
rand	x
srand	x

Strings

http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com
Microsoft Enhanced RSA and AES Cryptographic Provider

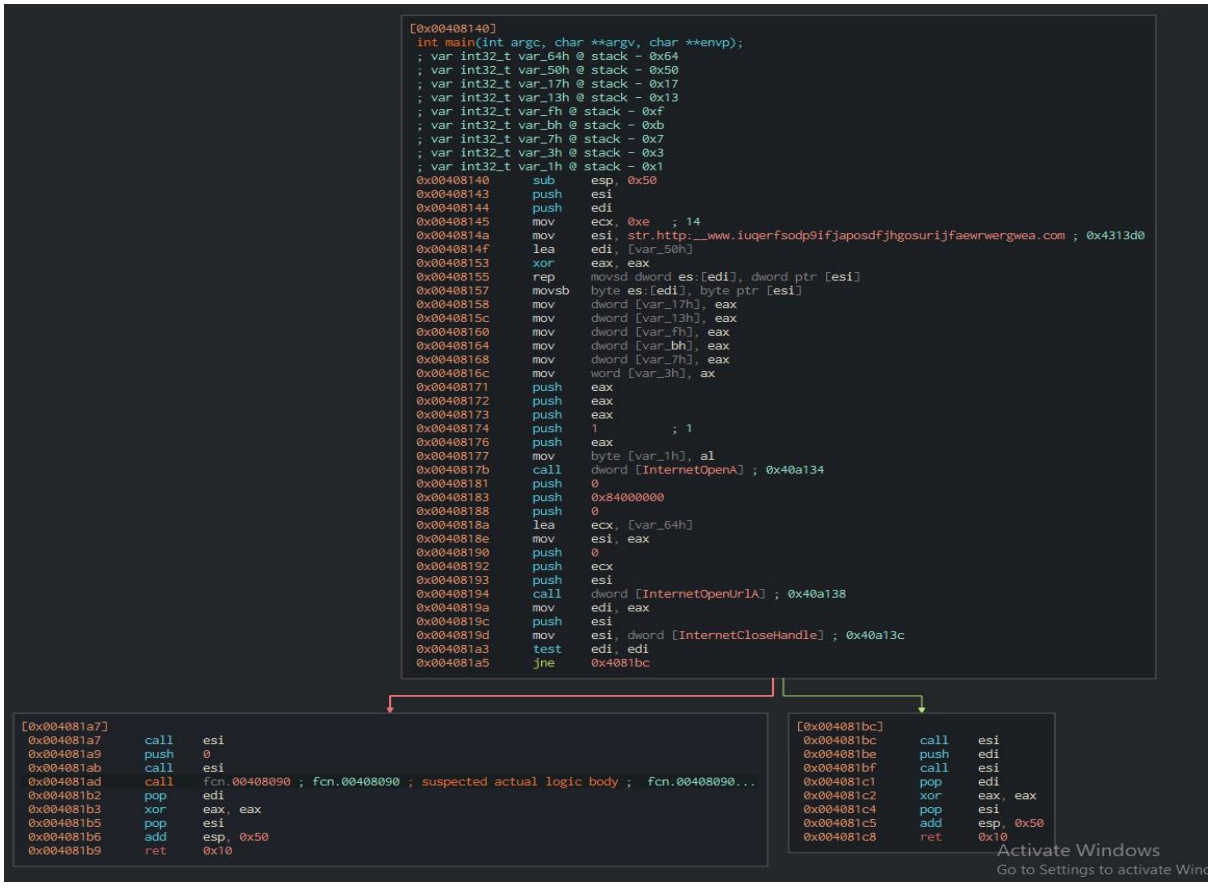
CryptGenKey
CryptDecrypt
CryptEncrypt
CryptDestroyKey
CryptImportKey
CryptAcquireContextA

msg/m_bulgarian.wnry
msg/m_bulgarian.wnry
msg/m_chinese (traditional).wnry
...

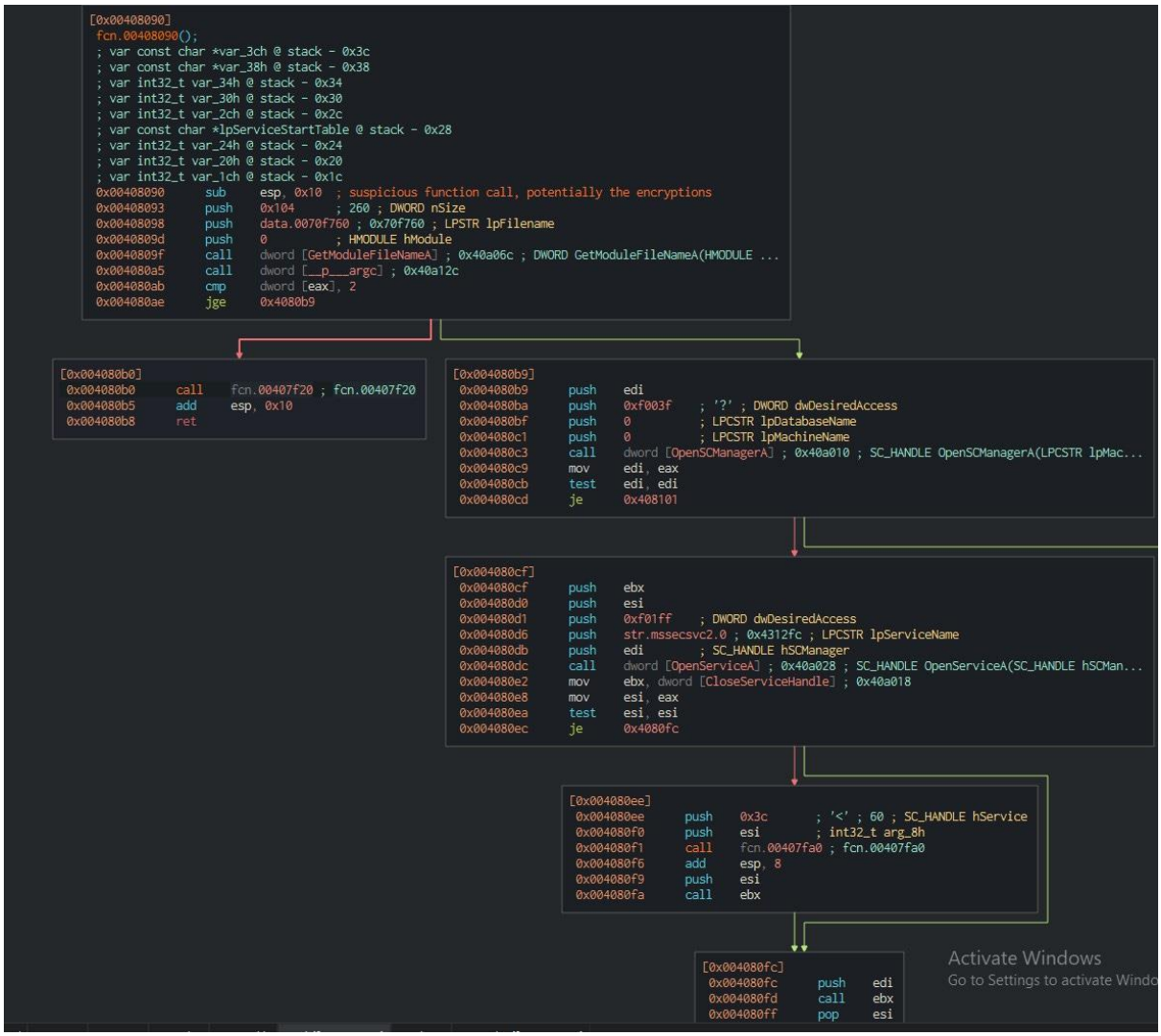
\\172.16.99.5\IPC$
\\192.168.56.20\IPC$

Main Flow

KillSwitch Mechanism: The program exits if the function call InternetOpenUrlA to the callback URL succeeds. If it fails, the main function executes (at fcn.00408090).



Next, it proceeds to fcn.00408090 and executes fcn.00407f20 (left-hand side branch).



fcn.00407f20 is straightforward, comprising two functions:

- fcn.00407c40 creates a service tasked with scanning LAN IPs for SMB EternalBlue vulnerabilities.
- fcn.00407ce0 encrypts all files within the local filesystem.

- fcn.00407c40 resolves to the following function call:

SC_HANDLE CreateServiceA(
  OpenSCManagerA(null, null, SC_MANAGER_ALL_ACCESS),
  lpServiceName="mssecsvc2.0",
  lpDisplayName="Microsoft Security Center 2.0 Service",
  dwDesiredAccess=SERVICE_ALL_ACCESS,
  dwServiceType=SERVICE_WIN32_OWN_PROCESS,
  dwStartType=SERVICE_AUTO_START,
  dwErrorControl=SERVICE_ERROR_NORMAL,
  lpBinaryPathName="C:\Users\ME\Desktop\Ransomware.wannacry.exe -m security",
  lpLoadOrderGroup=null // service does not belong to a group,
  lpdwTagId=null // not changing the existing tag,
  lpDependencies=null // service has no dependencies,
  lpServiceStartName=null // CreateServices uses the LocalSystem account,
  lpPassword=null,
);

- In fcn.00407ce0, tasksche.exe employs numerous filesystem APIs from Kernel32.dll to facilitate file encryption.

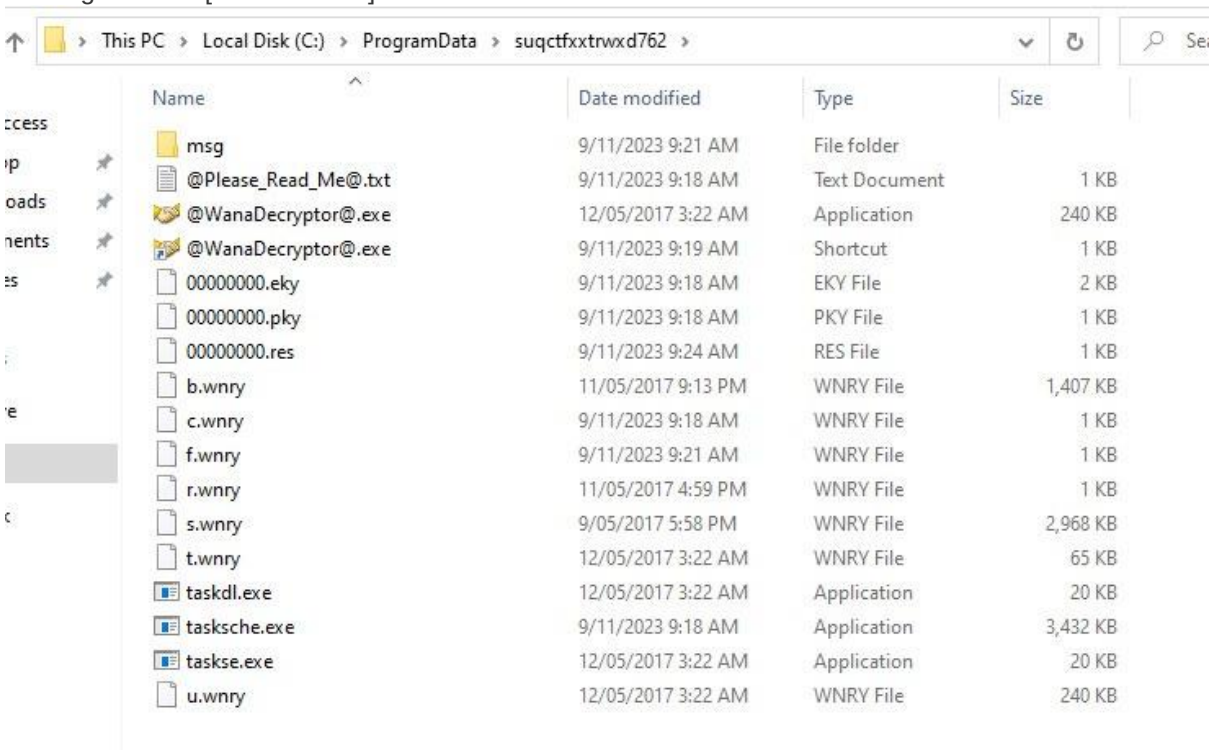
Decompiler (fcn.00407ce0)

```
int32_t var_2a4h;
int32_t var_2a0h;
LPVOID var_29ch;
int32_t var_28ch;
int32_t var_258h;
LPCSTR lpExistingFileName;
LPCSTR lpNewFileName;
eax = GetModuleHandleW ("kernel32.dll", edi, esi, ebp);
esi = eax;
ebx = 0;
if (esi == ebx) {
    goto label_0;
}
edi = imp.GetProcAddress;
eax = void (*edi)(uint32_t, char*) (esi, "CreateProcessA");
*(data.00431478) = eax;
eax = void (*edi)(uint32_t, char*) (esi, "CreateFileA");
*(data.00431458) = eax;
eax = void (*edi)(uint32_t, char*) (esi, "WriteFile");
*(data.00431460) = eax;
eax = void (*edi)(uint32_t, char*) (esi, "CloseHandle");
ecx = *(data.00431478);
*(data.0043144c) = eax;
if (ecx == ebx) {
    goto label_0;
}
if (*(data.00431458) == ebx) {
    goto label_0;
}
if (*(data.00431460) == ebx) {
    goto label_0;
}
if (eax == ebx) {
    goto label_0;
}
eax = FindResourceA (ebx, 0x727, data.0043137c);
esi = eax;
if (esi == ebx) {
    goto label_0;
}
eax = LoadResource (ebx, esi);
if (eax == ebx) {
    goto label_0;
}
eax = LockResource (eax);
var_29ch = eax;
if (eax == ebx) {
    goto label_0;
}
eax = SizeofResource (ebx, esi);
if (ebp == ebx) {
```

Indicators of Compromise

Paths

C:\ProgramData\[random hash]



Files

mssecsvc2.0.exe: db349b97c37d22f5ea1d1841e3c89eb4 (MD5)

tasksche.exe: 84c82835a5d21bbcf75a61706d8ab549 (MD5)

Network traffic Pattern

GET hxxp://iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com/

TCP Port 445 SMB Scan on LAN IPs

Rules & Signatures

YARA Rules

```
rule WannaCry {
    strings:
        $pe = "MZ"
        $callback = "iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com"
        $ipc = "\\IPC$"

    condition:
        $pe at 0 and $callback and $ipc
}
```