WannaCry Malware Analysis

The crypto-worm ransomware WannaCry is known by several names, including WannaCrypt,WannaCryptor, and WannaDecryptor. It became known worldwide in May 2017 and affected 230,000 computers in 150 countries — notably, the NHS in the UK — encrypting files on infected Windows computers. (TechTarget 2021; CSO Online 2018; Kaspersky, n.d.). Victims were told that their files would be permanently deleted if they did not pay the ransom within 3 days (Kaspersky, n.d.).

After infecting and encrypting files, a $300 ransom payment was demanded in bitcoin, later increased to $600. If the ransom was not paid within three days, victims were told that their files would be permanently deleted (Kaspersky, n.d.). Although the amount paid by victims was negligible — $144,000 — the impact on organizations like the NHS was far greater (Ghafur et al. 2019). The financial impact on the NHS alone was £5.9 m, thousands of appointments had to be canceled and rescheduled, and ambulances had to travel to unaffected hospitals (Ghafur et al. 2019).

Luckily, the attack was inadvertently halted by Marcus Hutchins and Jamie Hankins, who registered a web domain found in the disassembled code (Whittaker 2019). This domain is known as the “killswitch” domain: from looking at WannaCry in Ghidra and x64dbg we can see that the malware sends a DNS request to the URL. If it succeeds, nothing happens, but if it fails WannaCry starts the process of encryption. By registering the domain, the.

WannaCry was able to propagate rapidly throughout networks and infect Windows computers through the use of the EternalBlue exploit. The NSA developed after they discovered a vulnerability in the Windows Server Message Block (SMB) protocol. EternalBlue came into the hands of the Shadow Brokers, who released it. Although Microsoft had patched the vulnerability, many systems continued to be vulnerable, andcould consequently be impacted by Wannacry (Kaspersky, n.d)

Using Pestudio and HxD, we can gather information on the program hashes, strings used, and DLLs invoked by WannaCry.

Worm/dropper component	
MD5:	db349b97c37d22f5ea1d1841e3c89eb4
SHA1:	e889544aff85ffaf8b0d0da705105dee7c97fe26

DLLs used by worm/dropper component	
Import library	**Import number**
Kernel32.dll	32
Advapi32.dll	11
Ws2_32.sll	13
Msvcp60.dll	2
Wininet.dll	3
Msvcrt.dll	28

Using Bintext we can have a look at the strings invoked by the worm component. The most interesting are below. Evidently, this resource is related to the encryption of files.

Notable Strings in worm component	
String	**Location**
CryptAcquireContextA	0xa63a
CryptGenRandom	0xa652
StartServiceA	0xa664
GetCurrentThread	0xa53c

With PeStudio, we can see there is an executable resource within the worm component, there is an executable resource. It can be unpacked with ResourceHacker, and we can find hashes and DLLs invoked by it. This resource will be explored in more detail later.

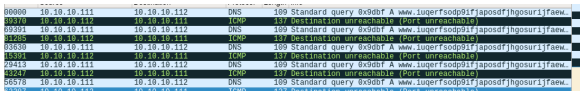
Encryption component	
Md5:	84c82835a5d21bbcf75a61706d8ab549
Sha1:	5ff465afaabcbf0150d1a3ab2c2e74f3a4426467

DLLs used by encryption component	
Import library	**Import number**
Kernel32.dll	54
Advapi32.dll	10
Msvcrt.dll	49
User32.dll	1

Notable Strings in encryption component	
String	**Location**
OpenMutexA	0xda84
CreateFileA	0xd922
CreateServiceA	0xdc2a
CryptReleaseContext	0xdc14

Using two VMs connected with an internal network, we performed a quick detonation of WannaCry so we could monitor the process of the infection and any network activity with Process Monitor, Process Explorer, and Wireshark.

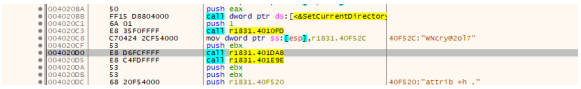
After launching wannacry.exe we can see an interesting DNS query to a domain: www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com.This is the “killswitch” domain.



Looking at the capture events in Procmon, tasche.exe is created. Tasche.exe then launches attrib.exe, iscalcs.exe, and taskdl.exe. It also starts the mssecsvs2.0 service, and we can see it attempting to spread through the SMB functionality. Shortly after this, the desktop background is changed and the WannaCry program launches, presenting the ransom note. Then it makes a hidden folder in a hidden holder. This is where the executable resource file is. The ransomware employs several

persistent mechanisms, writing itself into the registry and creating itself as a service so it cannot be stopped.

After unpacking the executable resource it asks for a password to be unzipped. Using ResourceHacker we can see that it is a PK zip file. To find the password we load the resource into x64dbg and have a look at the intermodular calls. From there, we can see FindResourceA, LoadResource, and LockResource. The resource is a self extracting zip file and the password is hardcoded, very close to the FindResourceA call. The resource successfully unzips when we input the password: wNcry@2o17.



The contents of the ZIP file are interesting was obtained with HxD:

- A “msg” folder that contains the ransom note translated in different languages.
- “B.wnry” is a bitmap image which displays the instructions to decrypt the encrypted files.
- “C.wnry” has several Tor addresses and the Tor browser.
- “R.wnry” has more decryption instructions in English.
- “S.wnry” is a ZIP archive which has the Tor software.
- “T.wnry” is a file with WANACRY! Encryption format.
- “U.wnry” is an executable file which contains the decryption component.
- “Taskdl.exe” is a cleanup routine that deletes files with the .wncry extension.
- “Taskse.exe” is a tool that helps spread and execute the malware further.

In Ghidra, we can see that the “killswitch” domain is hard-coded in the malware. So, when opening WannaCry in x64dbg we go straight to the URL using the string references. In that subroutine there is an API call to InternetOpenA, InternetOpenUrlA and InternetCloseHandle. Essentially, those calls pass some parameters when they call the domain and if they are successful the handle of the internet connection is returned to the EAX register. We breakpoint at the URL and run to that point. Interestingly, after stepping through the API calls there is a jne (jump if not equal) conditional. Upon looking at the graph view, there are two branches it can take — one that does nothing, and one that will call the WannaCry ransomware to start. We will take the branch that calls the ransomware to start. We are using Ghidra to examine the functions in more detail because of the handy decompiler. The first function after taking that specific branch is to check the arguments. If there are no arguments, it creates the mssecsvc service. Then, it writes the encryption component to C:\Windws\taskche, runs it with a /i argument that creates a hidden directory. It copies itself into the directory, then creates a service and launches a hidden copy of itself there.

References

CSO Online. 2018. “What is WannaCry ransomware, how does it infect, and who was responsible?” https://www.csoonline.com/article/3227906/what-is-wannacry-ransomware-how-does-it-infect-and-who-was-responsible.html.

Ghafur, S., S. Kristensen, G. Martin, P. Aylin, K. Honeyford, and A. Darzi. 2019. “A retrospective impact analysis of the WannaCry cyberattack on the NHS.” *npj Digit. Med.* 2 (98). https://doi.org/10.1038/s41746-019-0161-6.

Gibbs, Samuel. 2017. “WannaCry: hackers withdraw £108,000 of bitcoin ransom | Malware.” The Guardian. https://www.theguardian.com/technology/2017/aug/03/wannacry-hackers-withdraw-108000-pounds-bitcoin-ransom.

Hardy, Colin. 2019. “WannaCry Ransomware - Revisited. Behavioural and Static Analysis Techniques.” https://www.youtube.com/watch?v=AwouoQ802fA&ab_channel=ColinHardy.

Kaspersky. n.d. “Ransomware WannaCry: All you need to know.” Kaspersky. Accessed January 28, 2022. https://www.kaspersky.com/resource-center/threats/ransomware-wannacry.

techtargert. 2021. “WannaCry ransomware.” TechTarget. https://www.techtarget.com/searchsecurity/definition/WannaCry-ransomware.

Whittaker, Zack. 2019. “The sinkhole that saved the internet.” TechCrunch. https://techcrunch.com/2019/07/08/the-wannacry-sinkhole/.