ZAMBIA INFORMATION COMMUNICATION TECHNOLOGY (ICT) JOURNAL

*Volume 1 (Issue 1) (2017) Pages 35-40*

# Demystifying Ransomware Attacks: Reverse Engineering and Dynamic Malware Analysis of WannaCry for Network and Information Security

Aaron Zimba[1,2]
*Department of Computer Science and Technology*
*University of Science and Technology Beijing[1]*
*Beijing, China*
azimba@xs.ustb.edu.cn

Luckson Simukonda[2], Mumbi Chishimba[2]
*Department of Computer Science and Information Technology*
*Mulungushi University[2]*
*Kabwe, Zambia*
{thezo1992, chishimba.mumbi}@gmail.com

***Abstract*— Encryption has protected the Internet for some time now and it has come to raise user trust on the otherwise unsecure Internet. However, recent years have seen the use of robust encryption as stepping stone for cyber-criminal activities. Ransomware has not escaped the headlines even as it has attacked almost every sector of the society using a myriad of infection vectors. Mission critical data has been held to ransom and victims have had to part away with millions of dollars. The advent of the anonymous Bitcoin network has made matters worse where it's been virtually infeasible to trace the perpetrators. In this paper, we endeavor to perform dynamic analysis of WannaCry ransomware samples based on malware-free infection vectors. Further, we perform reverse-engineering to dissect the ransomware code for further analysis. Results show that despite the use of resilient encryption, the ransomware like other families in the wild uses the same attack structure and cryptographic primitives. Our analysis leads us to the conclusion that this ransomware strain isn't as complex as previously reported. This detailed practical analysis tries to raise awareness to the business community on the realities and importance of IT security whilst hinting on prevention, recovery and the limitations thereof.**

***Keywords-ransomware;encryption; malware; wannacry; infection vector***

## I. INTRODUCTION

The Internet today is plagued with a myriad of malware classes not limited to viruses, trojans, worms etc. Since it was not built with security in mind [1], the Internet has seen an incremental correlation between advancements in underlying technologies and malware sophistication – as technology advances, so does malware. Encryption, one of the pillars of secure technologies today, has likewise been integrated into the malware fraternity thus introducing a new form of cyber-attacks – crypto ransomware attacks [2]. Cyber-attacks are no longer the works of script kiddies or hacker-wannabes but rather organized cybercriminals such as Advanced Persistent Threat actors (APT) perpetuating all forms digital crimes [3]. Organized cybercrime groups attack networks for monetary gains which is a far stronger motivation absent in amateurs. This inherently implies that attackers are well organized and have at their disposal not only the technical knowhow but capable resources to attack networks than what a security administrator might have. The philosophy behind ransomware attacks is that of extortion, making the victim's data inaccessible via encryption until a ransom demand is met. The tragedy of ransomware is that it employs the most robust and resilient forms of encryption making it computationally infeasible [4] to decrypt a victim's data without consented efforts of distributed computing. WannaCry, one of the devastating ransomware attacks which plagued over 150 countries and traversed all continents [5] in May of 2017, spared no industry niche owing to the indiscriminate nature of the attack. It attacked universities, transport sector, health sector, telecoms sector etc implying that the ICT industry cannot burry its head in the sand but rather address the emerged new challenge. Figure 1 below shows the severity and distribution of reported WannaCry attacks world over.

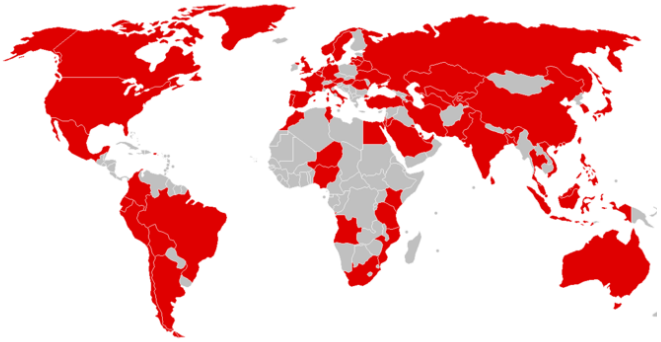


Figure 1. Distribution of initial WannaCry attacks [6]

What made WannaCry effective is not only the robust encryption schemes employed but the distribution mechanisms as well. A resilient encryption scheme is just one part needed for a successful ransomware attack but to reach all the aforementioned sectors, an effective infection mechanism is required. WannaCry used various forms of infections vectors [7] and employed network traversal by exploiting an SMB network vulnerability [8] to attack network devices on port 445 and any other physically connected devices. The media has not helped matters as it is flooded with a lot of inaccuracies and hearsay on the effect, infection vectors, prevention, mediation etc. History has however shown that the primitives used to effectuate ransomware attacks are not novel as cybercriminals tend to reuse malware code, ransomware inclusive [9]. Therefore, we in this paper, endeavor to demystify WannaCry ransomware attacks and operations. This gives insight not only into the inner workings of a particular malware but its collective strain. In light of this, we perform a full experimental dynamic and static analysis of WannaCry samples. Based on local and network behaviour of the malware samples, we suggest defense and mitigation measures for security purposes.

The rest of the paper is organized as follows: Section II discusses primitives of ransomware attack structures and components whilst the attack model is presented in Section III. The experimental test-bed and methodology are brought forth in Section IV while results and analyses are discussed in Section V and we conclude the paper in Section VI.

## II. Attack Structure and Components

Ransomware attacks come mainly in two forms; locker and crypto ransomware attacks [10]. WannaCry attacks identify with the latter which employ encryption to effectuate a denial of service (DOS) attack on victim data. Crypto ransomware further subdivides into Private-key Crypto Ransomware (PrCR) and Public-key Crypto Ransomware (PuCR). PrCR inherits the challenge of symmetric key distribution and management which has subsequently led attackers to employ custom crafted classical substitution stream or block cipher. In light of the aforementioned, PrCR attacks are crackable through cryptanalysis. This has consequently led to the widespread implementation of PuCR against PrCR [11]. The diagram below in figure 2 illustrates the generic structure of crypto ransomware payload common in both PrCR and PuCR.

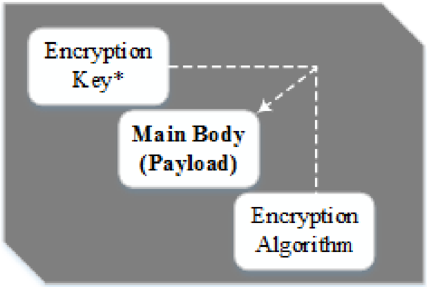


Figure 2. General structure of crypto ransomware

Depending on the attack structure, the encryption key* can be generated from within the ransomware payload after a successful attack or can be downloaded from a Command and Control server (C2). Regardless of the attack structure, crypto ransomware attacks rest on three main components; encryption methodology, C2 servers and infection vectors.

### A. Encryption Methodology

Encryption is the backbone component of the ransomware business model. Therefore, attackers have sought to employ the most resilient encryption algorithms not limited to RSA, AES, ECC etc. Symmetric encryption methodologies have the advantage of speed but do suffer from encryption/decryption key management whilst the resilient asymmetric encryption tends to be slower. Attackers have employed the advantages of both worlds to deploy a hybrid encryption methodology which when correctly implemented is deemed uncrackable [12]. Figure 3 below illustrates the attack structure of hybrid crypto ransomware.

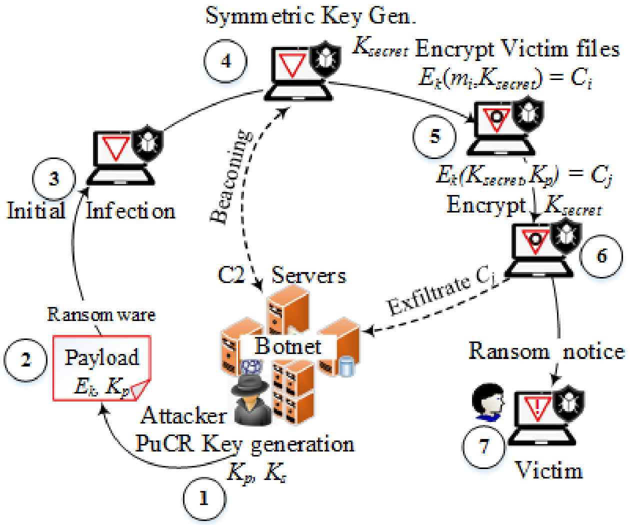


Figure 3. Hybrid PuCR attack structure

In the above attack structure, the public key $K_p$ generated from the PuCR key pair $\{K_p, K_s\}$ and implanted into the payload is used to encrypt the symmetric key $K_{secret}$ which actually encrypts the victim's files. This is denoted by the process $E_k(K_{secret}, K_p) = C_j$ . In this approach, the key $K_{secret}$ for decrypting user data, having been encrypted by $K_p$, can only be decrypted by the private key $K_s$ residing on the C2 servers. User data encryption, which is the actual ransomware attack is denoted by the process $E_k(m_i, K_{secret}) = C_i$ . In other attack structures, the ransomware payload generates an asymmetric key pair of which the public key is used to encrypt user data whilst the ransomware seeks to exfiltrate the private key to the C2 servers for future data decryption.

### B. C2 Servers

At the centre of operation of ransomware attacks lies Command and Control (C2) servers. C2 infrastructure may be owned by the attacker or could be a botnet controlled by the attacker. C2 are cardinal infrastructure and coordinating resources that the attacker harnesses to communicate with the ransomware payload once an infection is successful. Furthermore, C2 are also used to handle encryption key management and ransom payments via Bitcoin [13] and may also house the ransomware payload before it's delivered via different infection vectors. When a ransomware payload is successfully delivered to a victim, it usually beacons back to the C2 for further instructions. Earlier families of ransomware gave priority to confidentiality when communicating with C2 thus hinting on the cardinality of this component. Newer family versions however leverage the victim's system resources such as SSL to secure C2 communications. C2 may handle management of both the private and public key depending on the attack structure.

### C. *Infection Vectors*

Developing an effective crypto ransomware utilizing strong encryption techniques supported by a resilient C2 is only half the job for an effective ransomware attack. These two aforementioned components need to be supplemented by an impactful methodology that ensures that the ransomware is effectively delivered to the targeted victim. Infection vectors are the means through which attackers achieve this. Attackers use both benign and complex methodologies to deliver the ransomware payload to the victims. Malicious spam email tops the infection vector list as the most effective ransomware delivery mechanism [14]. The spam email usually carries the payload as an attachment in form of a Word macro, executable binary or even a dirty link pointing to some resource housing the ransomware.

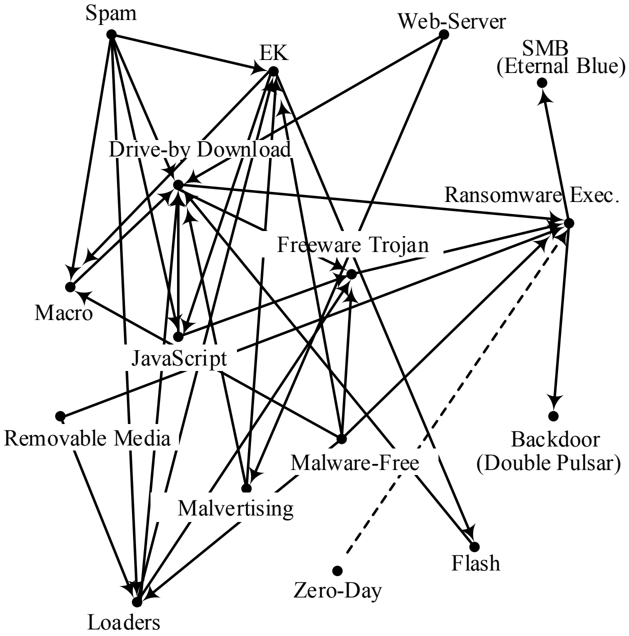


Figure 4. Bayesian network of various infection vectors

Attackers use a wide range of social engineering tactics to implore the victim to open the attachment or to follow the link which consequently results into installation of the ransomware and subsequent infection. However, spam mails are subject to filtering by email servers implying that not all sent spam mail will reach the intended victim. Attackers therefore use other infection vectors such as Exploit Kits (EK). The EternalBlue EK was the main infection vector used to propagate WannaCry [15] ransomware over the network on port 445 while the DoublePulsar EK ensued a backdoor [16]. Neutrino EK is known to ferry a wide range of ransomware including the famous Locky and Cryptowall [17]. We consider all these infection vectors and others in the construction of the infection Bayesian network of figure 4 as shown above and subsequent deduction of the attack model in the proceeding section. It's worth noting that the Bayes network above is not exhaustive and that some infection vectors harbor sub-infection vectors which can further extended the Bayesian network. These vectors tend to be interlinked in one way or the other.

## III. THE ATTACK MODEL

Figure 4 represents a directed infection vector network with various nodes sharing a relationship depicted by the associated edges. Depending on the infection source, the ransomware propagates through different nodes until it's executed on the victim thereby generating unique attack paths. The inter-dependence of nodes in a path can be captured by a Bayesian network in which the overall likelihood of executing the ransomware on the target can be expressed as a function of conditional probabilities in the associated attack path. The infection vector Bayesian network *(BiN)* is thus expressed as:

$$BiN^{in.v} = (\Gamma^{in.v}, \Omega^{in.v}) \tag{1}$$

where $\Gamma^{in.v}$ is a directed acyclic graph (DAG) with nodes $n_i \in N_i$ as discrete random variables and edges nodes $e \in E_i$ denoting casual relationships. $\Omega^{in.v}$ is a set of quantitative network parameters. Using Equation (1) and the network structure in figure 4, we deduce an attack graph for the attack model as illustrated below in figure 5.

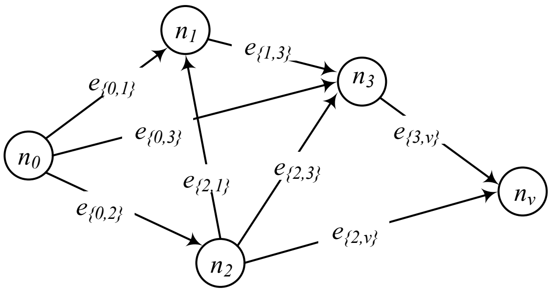


Figure 5. Illustrative attack graph

The attack model comprises: the *attacking agent* at source $n_0$ which is the ransomware itself; the *assets* which are nodes exploited in the course of reaching the target; the *goals* which are the sought after security breaches. We distinguish two *assets*; *pivot* assets and *critical* assets. Pivot assets, representative of the node set $\{n_0, n_1\}$ are not directly connected to the target whereas the critical asset e.g. $n_3$ is connected directly to the target. Each node $n_i$ casts a conditional probability distribution $Pr(n_i \mid Parents(N_i))$ quantifying the influence imposed by the parent's sample space, where the full joint probability distribution is given as:

$$Pr(n_1, \dots, N_i) = \prod_{i=1}^{n} Pr(n_i \mid parents\ (N_i)) \tag{2}$$

Therefore, the probability of compromising the target $n_v$ given the incoming edges $e_{\{2,v\}}$ and $e_{\{3,v\}}$ can be expressed as:

$$\begin{aligned} \Pr(n_v) &= \Pr(n_v | n_2, n_3) \\ &= \Pr(n_v | n_2) \cdot \Pr(n_v | n_3) \end{aligned} \tag{3}$$

Following from Equation (3), we assume Markov assumption [18] that a child node depends only its parents and not the history thereof. Thus the order of the attack events prior to access of the parent node is not significant in our attack model.

In light of the above, the attack scenarios of our experiments resume from the *pivot* nodes. Further, we use malware-free intrusions [19] as the infection vector.

## IV. Experiment Testbed and Methodology

The experiment setup for dynamic analysis is illustrated in figure 6 below comprising the server-side component for polling behavioral features from the client-side component where the WannaCry ransomware runs. The server-side runs Cuckoo sandbox and Volatility on Linux and we follow the best practices [20] for malware containment. Our ransomware samples are collected from Malwr and VirusTotal. We test the ransomware samples on Windows virtual hosts (Windows XP, Windows 7 and Windows 8).

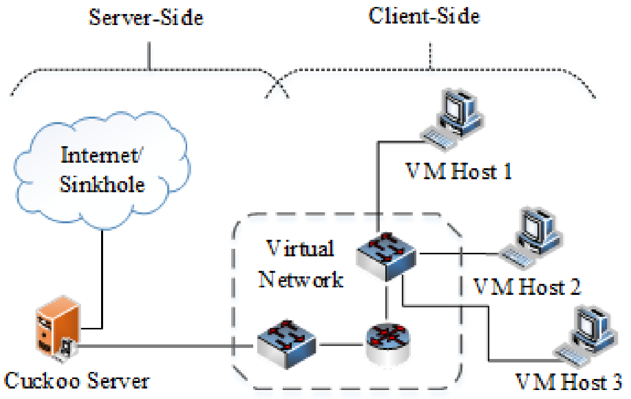


Figure 6. Ransomware dynamic analysis test-bed setup

To acquire the pivot and critical assets depicted in the attack model, we launch a reconnaissance attack using Nmap on the target network. The results are shown in Table I below.

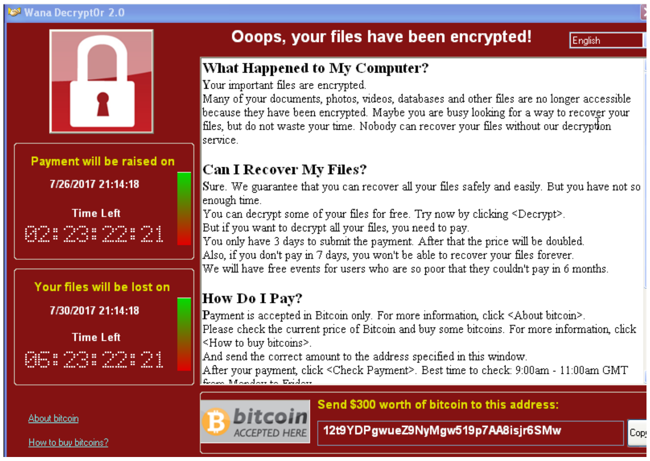
TABLE I. Reconnaissance Probe Results

| Host | Open Ports | Protocol | Service |
|---|---|---|---|
| VM Host 1 | 135 | TCP | Msrpc |
|  | 139 | TCP | Netbios-ssn |
|  | 3389 | TCP | RDP |
|  | 123 | UDP | Ntp |
| VM Host 2 | 3389 | TCP | Ms-wbt-server |
|  | 445 | TCP | Microsoft-ds |
|  | 5357 | TCP | Wsdapi |
| VM Host 3 | 554 | TCP | Rtsp |
|  | 2869 | TCP | Icslap |
|  | 3389 | TCP | RDP |
|  | 445 | TCP | Microsoft-ds |

With conditions (*cf.* Equation 3) satisfied that actualize the pursued infection vector [19], we implant the ransomware on the targeted victim and perform dynamic analysis. For reverse engineering the ransomware code, we perform static code dissection on the binary using an interactive disassembler IDA Pro and a debugger Ollydbg. We discuss the results of both analyses in the proceeding section.

## V. Results and Analyses

WannaCry upon execution, unlike other ransomware strains does not employ hibernation as a sandbox evasion technique. In a couple of seconds, the ransomware encrypts all directory contents on the system except those in the SystemRoot and Program Files. It does not encrypt the *.exe or *.dll file extensions. This is only logical considering that WannaCry is not a locker ransomware. The product of the encryption process are files with the *.WNCRY extension. The ransomware note with a Bitcoin address of *{12t9YDPgwueZ9NyMgw519pAA8isjr6Mw}* is shown in figure 7 below.



Ooops, your files have been encrypted!

Payment will be raised on
7/26/2017 21:14:18
Time Left
02:23:22:21

Your files will be lost on
7/30/2017 21:14:18
Time Left
06:23:22:21

About bitcoin
How to buy bitcoins?

**What Happened to My Computer?**
Your important files are encrypted.
Many of your documents, photos, videos, databases and other files are no longer accessible because they have been encrypted. Maybe you are busy looking for a way to recover your files, but do not waste your time. Nobody can recover your files without our decryption service.

**Can I Recover My Files?**
Sure. We guarantee that you can recover all your files safely and easily. But you have not so enough time.
You can decrypt some of your files for free. Try now by clicking <Decrypt>.
But if you want to decrypt all your files, you need to pay.
You only have 3 days to submit the payment. After that the price will be doubled.
Also, if you don't pay in 7 days, you won't be able to recover your files forever.
We will have free events for users who are so poor that they couldn't pay in 6 months.

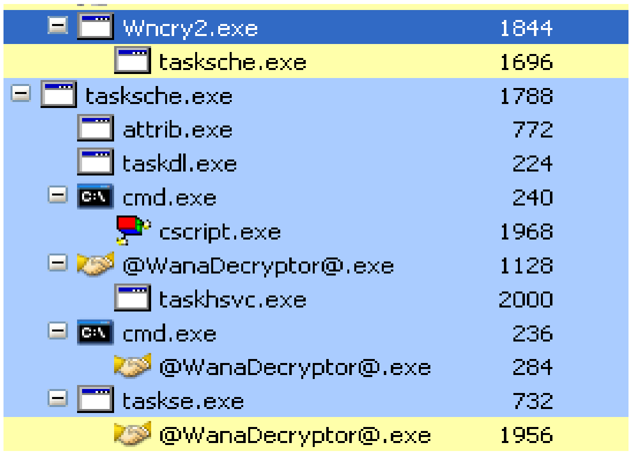
**How Do I Pay?**
Payment is accepted in Bitcoin only. For more information, click <About bitcoin>.
Please check the current price of Bitcoin and buy some bitcoins. For more information, click <How to buy bitcoins>.
And send the correct amount to the address specified in this window.
After your payment, click <Check Payment>. Best time to check: 9:00am - 11:00am GMT from Monday to Friday.

bitcoin ACCEPTED HERE
Send $300 worth of bitcoin to this address:
12t9YDPgwueZ9NyMgw519p7AA8isjr6SMw
Copy

Figure 7. WannaCry ransom note after encryption

### A. Dynamic Analysis

The main process *Wncry2* PID 1844 spawns 3 child processes; *tasksche* PID 1788, *cmd* PID 240, *taskdl* PID 224 and a couple more which terminate upon task completion. The spawning activity is shown in figure 8 below.



```
⊟ Wncry2.exe                 1844
    tasksche.exe             1696
⊟ tasksche.exe               1788
    attrib.exe                772
    taskdl.exe                224
  ⊟ cmd.exe                   240
      cscript.exe            1968
  ⊟ @WanaDecryptor@.exe      1128
      taskhsvc.exe           2000
  ⊟ cmd.exe                   236
      @WanaDecryptor@.exe     284
  ⊟ taskse.exe                732
      @WanaDecryptor@.exe    1956
```

Figure 8. WannaCry process tree decomposition

The Wncry2 process masquerades internally as the Microsoft utility diskpart.exe. The process tree likewise executes VB and batch scripts used to achieve a persistence mechanism. The *icacls.exe* is used to grant global permissions (*777* Linux equivalent) to the directory in contention. Loaded libraries at runtime include but not limited to kernel32, shell32.dll, user32 etc after which calls are made.

The sample comes with an implanted *master* RSA public key whose corresponding private key is retained by the attacker. Upon infection, the ransomware uses a secure PRNG function *CryptGenRandom* from the operating system *CryptoAPI* to generate a 2048-bit sub-RSA pair for use by the cryptographic service provider (CSP). The public key from this sub-pair is exported to *00000000.pky* in unencrypted form. The private key thereof is exported and written to *00000000.eky* after being encrypted by the implanted master RSA public key using the *CryptEncrypt* function. Further, a 128-bit AES key is generated in Cipher Block Chaining (CBC) mode for encryption of the victim's target files, with a unique key per file. These symmetric keys are then encrypted by the earlier public key from the sub-pair which was exported to *00000000.pky.* The diagram below in figure 9 illustrates the encryption process flow.

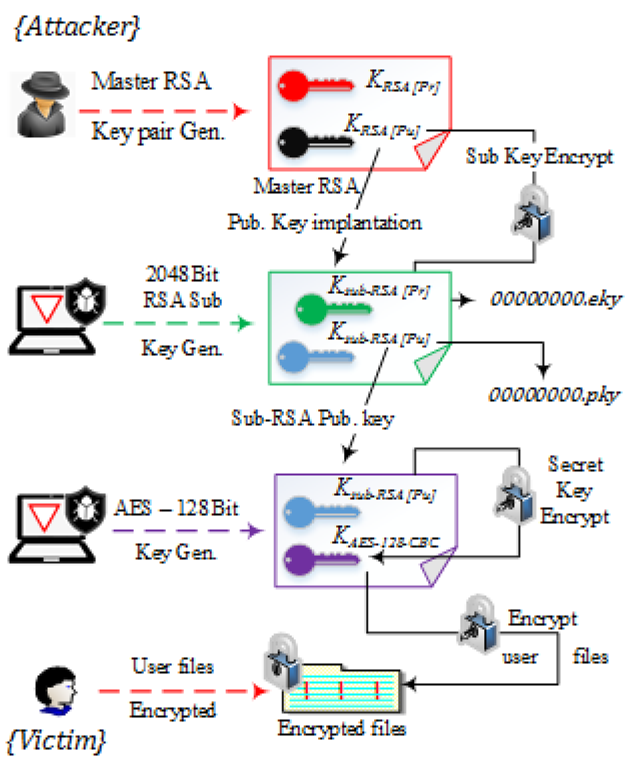
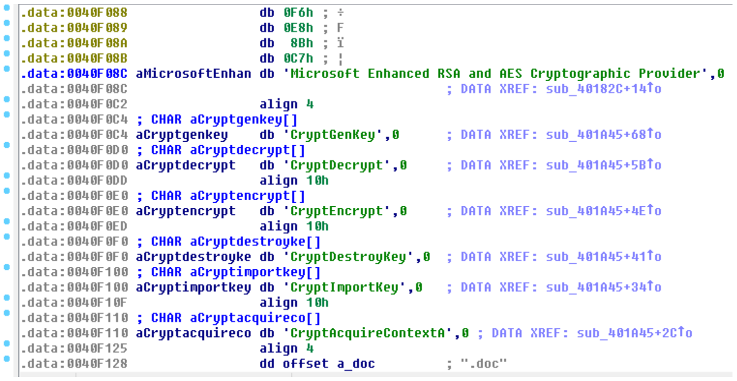


Figure 9. WannaCry encryption process

In total, the ransomware operates on four encryption keys: one RSA public key from the master key pair, two keys from the payload-generated sub-RSA pair and one AES symmetric key. The AES key is only encrypted by the payload-generated sub-RSA public key upon completion of encrypting the victim's targeted file extensions.

### B. Static Analysis

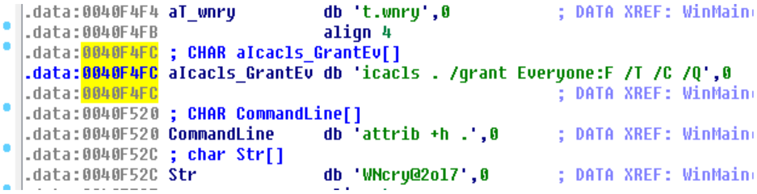
The encryption routines of WannaCry run from address 0040F08C to 0040F110 as shown in figure 10 below.



```
.data:0040F088                 db 0F6h ; ÷
.data:0040F089                 db 0E8h ; F
.data:0040F08A                 db  8Bh ; ï
.data:0040F08B                 db 0C7h ; ¦
.data:0040F08C aMicrosoftEnhan db 'Microsoft Enhanced RSA and AES Cryptographic Provider',0
.data:0040F08C                                         ; DATA XREF: sub_40182C+14↑o
.data:0040F0C2                 align 4
.data:0040F0C4 ; CHAR aCryptgenkey[]
.data:0040F0C4 aCryptgenkey    db 'CryptGenKey',0      ; DATA XREF: sub_401A45+68↑o
.data:0040F0D0 ; CHAR aCryptdecrypt[]
.data:0040F0D0 aCryptdecrypt   db 'CryptDecrypt',0     ; DATA XREF: sub_401A45+5B↑o
.data:0040F0DD                 align 10h
.data:0040F0E0 ; CHAR aCryptencrypt[]
.data:0040F0E0 aCryptencrypt   db 'CryptEncrypt',0     ; DATA XREF: sub_401A45+4E↑o
.data:0040F0ED                 align 10h
.data:0040F0F0 ; CHAR aCryptdestroyke[]
.data:0040F0F0 aCryptdestroyke db 'CryptDestroyKey',0  ; DATA XREF: sub_401A45+41↑o
.data:0040F100 ; CHAR aCryptimportkey[]
.data:0040F100 aCryptimportkey db 'CryptImportKey',0   ; DATA XREF: sub_401A45+34↑o
.data:0040F10F                 align 10h
.data:0040F110 ; CHAR aCryptacquireco[]
.data:0040F110 aCryptacquireco db 'CryptAcquireContextA',0 ; DATA XREF: sub_401A45+2C↑o
.data:0040F125                 align 4
.data:0040F128                 dd offset a_doc         ; ".doc"
```

Figure 10. WannaCry encryption calls

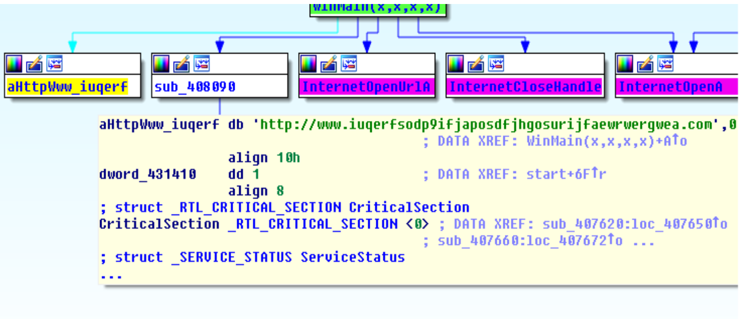
Granting of global permissions to the directory in contention by *icacls.exe* is shown at address 0040F4FC in figure 11 below.



```
.data:0040F4F4 aT_wnry         db 't.wnry',0           ; DATA XREF: WinMain
.data:0040F4FB                 align 4
.data:0040F4FC ; CHAR aIcacls_GrantEv[]
.data:0040F4FC aIcacls_GrantEv db 'icacls . /grant Everyone:F /T /C /Q',0
.data:0040F4FC                                         ; DATA XREF: WinMain
.data:0040F520 ; CHAR CommandLine[]
.data:0040F520 CommandLine     db 'attrib +h .',0      ; DATA XREF: WinMain
.data:0040F52C ; char Str[]
.data:0040F52C Str             db 'WNcry@2ol7',0       ; DATA XREF: WinMain
```

Figure 11. Permission allocation to current directory

It's worth noting that the current directory is set to *hidden* by the attribute *"attrib +h."* at address 0040F520. One of the samples we evaluated had a kill-switch which basically is used to detect sandboxing operations. The kill-switch domain is seen at address 004313D0 as shown in figure 12 below.



```
aHttpWww_iuqerf db 'http://www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com',0
                                        ; DATA XREF: WinMain(x,x,x,x)+A↑o
                align 10h
dword_431410    dd 1                    ; DATA XREF: start+6F↑r
                align 8
; struct _RTL_CRITICAL_SECTION CriticalSection
CriticalSection _RTL_CRITICAL_SECTION <0> ; DATA XREF: sub_407620:loc_407650↑o
                                        ; sub_407660:loc_407672↑o ...
; struct _SERVICE_STATUS ServiceStatus
...
```

Figure 12. WannaCry kill-switch domain

The ransomware variants without the kill-switch domain seem to have had been hex-edited without changing other parts of the code. This is to imply that encryption routines, their associated functions and other aspects remain unchanged. Summary characteristics of the analyzed samples is shown in Table II below.

***Remediation and Prevention:*** like all ransomware, WannaCry is best prevented than cured. Prevention should strongly be offline since the observed samples propagate on the network via port 445 using the exploit CVE-2017-0145 [21] against the SMB service. Since the samples overwrite the original files upon encryption, system restore efforts do not yield fruition.

TABLE II. WANNACRY SAMPLES CHARACTERISTICS

| Variant | Kill-Switch | CryptoAPI | Instant I/O Dev. Attack | Attack .exe/.dll |
|---|---|---|---|---|
| Sample 1 | ✓ | MS Base CSP | x | x |
| Sample 2 | x | MS Enhanced Crypto-Prov. | ✓ | x |
| Sample 3 | ✓ | MS Base CSP | x | x |
| Sample 4 | x | MS Enhanced Crypto-Prov. | ✓ | x |
| Sample 5 | ✓ | MS Base CSP | x | x |

All observed samples do not attack system files not limited to .exe and .dll extensions. It's however impractical and illogical to rename all user files to these extensions in an effort to avoid the attack as opposed to offline backup. The DoublePulsar backdoor and EternalBlue SMB propagation are countered by patching MS17-010 which affects all Windows versions prior to Windows 10. Since the sub-RSA key pair are generated on the host, it is possible retain the primes and modulus. WanaKiwi [22] uses such an approach to derive the decryption key and subsequent decryption of the affected files where the observed exponent in all the samples was *65537 (0x10001)*. It should be noted however that this method only works if the memory allocated to the WannaCry process is not overwritten of flushed, i.e. no system restart or reallocation of memory.

## VI. CONCLUSIONS

WannaCry ransomware is not so different from other ransomware families; it uses same encryption primitives and attack methodologies. However, unlike other ransomware, it generates a sub-RSA key pair which is used to encrypted the generated symmetric key. What made WannaCry spread fast and catch the attention of the world is the inclusion of the worm component which enabled it to self-propagate in networks with vulnerable SMB service. This infection vector is persistent and still valid for unpatched systems.

The inclusion of the kill-switch for sandbox evasion led to the demise of the initial variant of the ransomware. However, both the initial strain and the enhanced version which exclude the kill-switch are seen in the wild today on a daily basis [23]. Like other crypto ransomware, WannaCry does not encrypt system files and directories. Further, it does not check the file header before encryption but rather just the file extension. The presence of residual RSA primes in the memory address space of the WannaCry process makes it possible to derive a decryption key and subsequent decryption. Nevertheless, this recovery technique is only valid given the associated memory space is not overwritten or flushed.

## REFERENCES

[1] M. Gallo and W.M. Hancock. "Networking explained." Digital Press. Dec 2001.

[2] A. Kharraz, W. Robertson, D. Balzarotti, L. Bilge, and E. Kirda. "Cutting the gordian knot: A look under the hood of ransomware attacks." In International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, pp. 3-24. Springer, Cham, 2015.

[3] J.V. Chandra, N. Challa, and S.K. Pasupuleti. "Advanced persistent threat defense system using self-destructive mechanism for cloud security." In Engineering and Technology (ICETECH), 2016 IEEE International Conference on, pp. 7-11. IEEE, 2016.

[4] A. Al Hasib and A.A.M. Mahmudul Haque. "A comparative study of the performance and security issues of AES and RSA cryptography." In Convergence and Hybrid Information Technology, 2008. ICCIT'08. Third International Conference on, vol. 2, pp. 505-510. IEEE, 2008.

[5] T. Webb and S. Dayal. "Building the wall: Addressing cybersecurity risks in medical devices in the USA and Australia." Computer Law & Security Review (2017).

[6] "Cyber-attack: Europol says it was unprecedented in scale." BBC News. (13th May 2017) [Online] Available: http://www.bbc.com/news/world-europe-39907965 [Accessed 17th June 2017]

[7] Adam McNeil. (19th May, 2017). "How did the WannaCry ransomworm spread?" [Online] Available: https://blog.malwarebytes.com/cybercrime/2017/05/how-did-wannacry-ransomworm-spread/

[8] S. Mansfield-Devine. "Leaks and ransoms–the key threats to healthcare organisations." Network Security 2017, no. 6 pp. 14-19. Elsevier. 2017.

[9] D. Formby, S. Durbha and R. Beyah. "Out of control: Ransomware for industrial control systems." (2017).

[10] A. Zimba. "Malware-Free Intrusion: A Novel Approach to Ransomware Infection Vectors." International Journal of Computer Science and Information Security 15, no. 2 (2017): 317.

[11] M.M. Ahmadian, H.R. Shahriari, and S.M. Ghaffarian. "Connection-monitor & connection-breaker: A novel approach for prevention and detection of high survivable ransomwares." In Information Security and Cryptology (ISCISC), 2015 12th International Iranian Society of Cryptology Conference on, pp. 79-84. IEEE, 2015.

[12] V. Palanisamy and A.M. Jeneba "Hybrid cryptography by the implementation of RSA and AES." International Journal of Current Research 33, no. 4 (2011): 241-244.

[13] K. Liao, Z. Zhao, A. Doupé and G.J. Ahn. "Behind closed doors: measurement and analysis of CryptoLocker ransoms in Bitcoin." In Electronic Crime Research (eCrime), 2016 APWG Symposium on, pp. 1-13. IEEE, 2016.

[14] A.W. Wijayanto, "Fighting cyber crime in email spamming: An evaluation of fuzzy clustering approach to classify spam messages." In Information Technology Systems and Innovation (ICITSI), 2014 International Conference on, pp. 19-24. IEEE, 2014.

[15] Spinellis Diomidis. "Software Reliability Redux." IEEE Software 34, no. 4 (2017): 4-7.

[16] M. Revankar. (23rd May, 2017). "WannaCry 2.0: Detect and Patch EternalRocks Vulnerabilities Now." [Online] Available: https://www.tenable.com/blog/wannacry-2-0-detect-and-patch-eternalrocks-vulnerabilities-now

[17] D. Sgandurra, L.M. González, R. Mohsen, and E.C. Lupu. "Automated Dynamic Analysis of Ransomware: Benefits, Limitations and use for Detection." arXiv preprint arXiv:1609.03020 (2016).

[18] Z. Ghahramani. "An introduction to hidden Markov models and Bayesian networks." International journal of pattern recognition and artificial intelligence 15, no. 01 (2001): 9-42.

[19] Aaron Zimba, Zhaoshun Wang,"Malware-Free Intrusions: Exploitation of Built-in Pre-Authentication Services for APT Attack Vectors", International Journal of Computer Network and Information Security(IJCNIS), Vol.9, No.7, pp.1-10, 2017.DOI: 10.5815/ijcnis.2017.07.01

[20] C. Rossow et al. "Prudent practices for designing malware experiments: Status quo and outlook." Security and Privacy (SP), 2012 IEEE Symposium on. IEEE, 2012.

[21] NVD - CVE-2017-0145. (16th March, 2017) [Online] Available: https://nvd.nist.gov/vuln/detail/CVE-2017-0145

[22] Wanakiwi. (May 2017). [online] Available: https://github.com/gentilkiwi/wanakiwi/releases [Accessed 13th June, 2017]

[23] "Note on WannaCrypt Infection Count Accuracy." Malware Intel Botnet Tracker. (June 2017).[Online] Available: https://intel.malwaretech.com/botnet/wcryp