

Spoil the Hunting : An Approach to Avoiding Ransomware Attacks

Suhyeon Lee^{1,2}, Huy Kang Kim², Kyounggon Kim²

¹ Agency for Defense Development, Seoul, Korea,
korea@add.re.kr

² Graduate School of Information Security, Korea university, Seoul, Korea,
{orion-alpha, cenda, anesra}@korea.ac.kr

Abstract. It has been several years since the introduction of 'Crypto Ransomware' (referred as ransomware), and it became a mainstream of cybercrime. Malicious code makers and distributors are making steady profits through ransomware attacks, and the structure is becoming more sophisticated. We looked at existing research on ransomware defense and found that most studies have targeted or intervened ransomware detection. In this paper, we focus on how we can abstract the process of ransomware attacks and protect files from ransomware attacks as well as detecting ransomware attacks. In this paper, we suggest a new defense point which randomizes target files to make it difficult to select valuable files for ransomware attacks. By implementing this methodology, we could completely avoid the data encryption in 141 of the 143 active ransomware samples in our experiment applied to the system. The rate of defense against ransomware samples tested in our study was 98.6%. Our research shows an approach that can effectively protect data against specific threats.

Keywords: ransomware, malware, moving target defense, protection

1 Introduction

Ransomware is a type of malware that encrypts a victim's files and threatens victims to pay for their description. Ransomware attacks have been a significant trend of cybercrimes for a long period. Recently, in May 2017, 'WannaCry Ransomware' hit the world with a powerful zero-day vulnerability related to Windows SMB services leaked from the Equation Group [17]. State-sponsored hackers are also using ransomware for monetary purposes. North Korea has been pointed out behind the WannaCry attack [22]. The damage of the WannaCry was estimated at 5 billion dollars [9]. Not only WannaCry, but also many other ransomware families such as CryptoLocker, CryptoWall, and Tesla are constantly active. One of the characteristics of this malicious software is not simple vandalism but service with a clear financial objective. Unlike other attacks, ransomware attacks are making a lot of money from infected victims by asking for money directly through cryptocurrency such as Bitcoin [18, 22, 23]. Ransomware attacks

require money directly to victims with strong encryption algorithms such as AES-128 and RSA-2048. As it is a good moneymaker, it highly motivates malicious code authors and highly developed in real time. In this regard, ransomware attacks, one of the most significant threats currently, is expected to continue to be a threat in the future, and countermeasures against this threat have been studied continuously.

We faced several challenging problems to deal with ransomware attacks. First of all, it is difficult to detect ransomware with pre-defined signatures. Because of 20,000 variants are created on average per month [24]. It means they are immune to traditional detection mechanism such as signature based detection. Second, some of the ransomware attacks are connected with social engineering techniques as an important attack vector [5, 10, 34]. If users execute ransomware through social engineering, it infects their system in high possibility. Therefore, we need to prepare the situation that ransomware attacks work on our system with high privilege in the end. Third, the activity of ransomware attacks in the system is to read and write files by accessing files, which is difficult to distinguish from normal software activities. The contents and contribution of this paper are as follows.

- From the ransomware defender’s point of view, we analyzed the ransomware attack process in four parts. It helps defenders decide which state-based defense they should use against ransomware attacks.
- Our method, which is randomizing targets of ransomware attacks is advantageous because it does not need to detect ransomware attacks compared to the existing method and it consumes little system resources. This method force ransomware attacks use inefficient ways to achieve their goal.
- Experiments showed and demonstrated the effectiveness of methods to protect data from ransomware attacks. We protected data (files) from 141 samples out of 143 infamous ransomware samples belonging to the four ransomware families we tested.

2 Background

2.1 Three Massive Outbreaks of Ransomware

According to the 2017 statistics of Kaspersky lab, ‘WannaCry’, ‘ExPetr’, and ‘BadRabbit’ are most massive outbreaks ransomware globally. WannaCry hit in May 2017, Expetr in June, and BadRabbit in October. WannaCry attack started on May 12, and it showed highly developed techniques in spreading. The ransomware automatically targeted all hosts in a same local subnet using severe SMB vulnerability, MS17-010 [3], as well as random IP ranges outside all around the world. ExPetr and Badrabbbit appeared later and used similar behaviors with WannaCry. They also massively hit the world [32].

Ransomware encrypts important files on the victim computer with symmetric key-based algorithm, for example, AES-128, and encrypts the symmetric key

value used for encryption again with asymmetric key based algorithm, for example, RSA-2048, making it difficult for the user to decrypt the file. WannaCry changes the extension of the encrypted file to .WCRY [31]. ExPetr, known as destructive ransomware, attacks mainly critical infrastructure facilities and, once infiltrated into one host, infiltrates the internal network using two Windows SMB service exploits: EternalBlue and EternalRomance [3]. Unlike WannCry, Expetr is known as a destructive ransomware that does not tell victims how to recover encrypted files [28]. 'BadRabbit Ransomware' requires 0.05 Bitcoin for infected computer users [29]. 'BadRabbit Ransomware', like WannCry and ExPetr, has the feature to penetrate into the internal network through the infected system to expand the infection. BadRabbit also uses the EternalRomance exploit to infect the internal network. The latest ransomware can continuously penetrate the same network using exploits when a single host is infected.

2.2 Ransomware as a Service

Ransomware is a business model as well as cybercrime that is very well automated and uses elaborate hybrid cryptosystem including both symmetric cryptography and asymmetric cryptography scheme. Attackers spread ransomware using phishing emails and websites those they infected. As they get paid by cryptocurrencies like Bitcoin, they can earn money without releasing their direct information. This way, attackers extorted billions of dollars from victims over the past fifteen years [35]. Their delivery model created the word Ransomware-as-a-Service (RaaS) [4]. In the past few years, RaaS business became a big trend of cybercriminals [25].

Business Model Ransomware has the obvious business model that can be showed as simple business model canvas as shown in Figure 1 [27]. As distributors are paid directly from victims, We made this model in the perspective of ransomware distributors, not developers. We can see this cybercrime makes good business sense. That is the reason why ransomware attacks have been advanced over constant. Attackers have partners (ransomware developers), know what service they have to provide (decryption), know what kinds of files victim users want (documents, photos, data), know how can they get paid (cryptocurrency) and other business elements. We highlight some of their business elements cannot be solved only with technical solutions because ransomware already became a business service. However it is not within the scope of this study to outline all business of ransomware. This paper will be limited to consideration of technical solutions related in user system.

Valuable Targets Ransomware encrypts valuable files for victims. The valuable files include database file for servers or document, photo files for ordinary users. In this paper, we treat ransomware in Windows for giving shape to our method and test even though not only Windows but also Linux is targeted by ransomware attacks [7]. Especially, Windows use file extensions for usability of applications

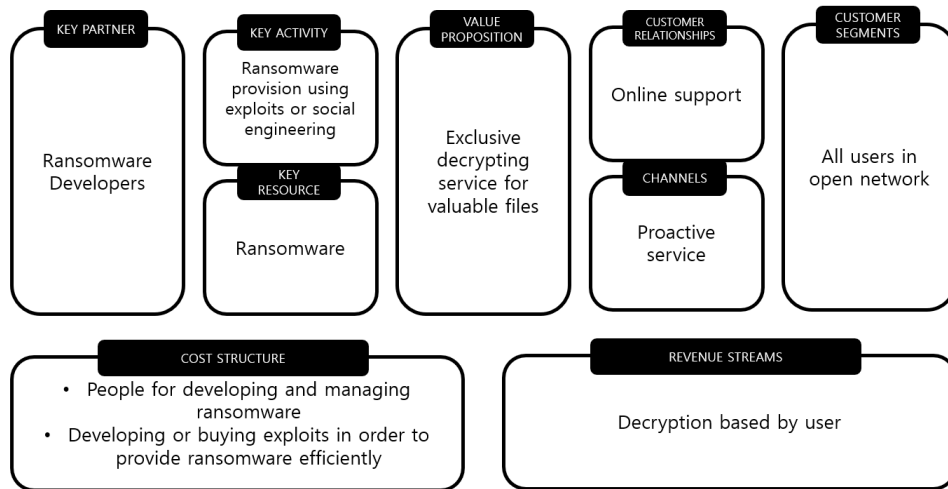


Fig. 1. Ransomware Business Model Canvas

and users [1]. Ransomware attackers distinguish between possibly valuable files and other files for users by file extensions. Fig 2 shows target extension list of one of WannaCry ransomware samples those target users who use Windows as their operating system. Encryption does not mean destruction, but it means restorable only with keys when the encryption algorithm is sound. According to Ransomware damage report, it shows ransomware attacks damaged a lot and some of the victims are willing to pay for their files [9]. Even one web hosting company paid 1 million dollars for decryption [7].

```

der pfx key crt csr p12 pem odt ott sxw stw uot 3ds max 3dm
ods ots sxc stc dif slk wb2 odp otp sxd std uop odg otg sxm mml
lay lay6 asc sqlite3 sqlitedb sql accdb mdb db dbf odb frm myd
myi ibd mdf ldf sln suo cs c cpp pas h asm js cmd bat ps1 vbs vb
pl dip dch sch brd jsp php asp rb java jar class sh mp3 wav swf
fla wmv mpg vob mpeg asf avi mov mp4 3gp mkv 3g2 flv wma
mid m3u m4u djvu svg ai psd nef tiff tif cgm raw gif png bmp
jpg jpeg vcd iso backup zip rar 7z gz tgz tar bak tbk bz2 PAQ
ARC aes gpg vmx vmdk vdi sldm sldx sti sxi 602 hwp snt onetoc2
dwg pdf wk1 wks 123 rtf csv txt vsdx vsd edb eml msg ost pst
potm potx ppam ppsx ppsm pps pot pptm pptx ppt xltm xltx xlc
xlm xlt xlw xlsb xlsx xls dotx dotm dot docm docb docx doc

```

Fig. 2. WannaCry Target Extension List

Hybrid Cryptography Most of ransomware use hybrid cryptography to encrypt files. Hybrid encryption has two sections. The one is the asymmetric sec-

tion, Key Encapsulation Mechanism (KEM), that encrypts the asymmetric key. The other is the symmetric section, Data Encapsulation Mechanism (DEM), that encrypts the message [19]. Formally, encryption scheme in ransomware is as follows.

The key generation algorithm, Gen , which takes as in input a security parameter 1^{k_1} , outputs a public/private key pair (pk, sk) for a deterministic asymmetric encryption algorithm $EncA$.

$$Gen(1^{k_1}) = (pk, sk) \quad (1)$$

The key generation algorithm, Gen' , which takes a security parameter 1^{k_2} as input, outputs a symmetric key K .

$$Gen'(1^{k_2}) = K \quad (2)$$

A deterministic symmetric encryption algorithm $EncS$, which takes an input file f of any length and encrypts it with the key K .

$$C = EncS_K(f) \quad (3)$$

The deterministic asymmetric encryption algorithm $EncA$, which takes previously encrypted file data C and the key K as input, outputs final encryption result D .

$$D = EncA_{pk}(C, K) \quad (4)$$

The first step of encryption scheme is operated in the side of attackers. By proceeding encryption from second steps of encryption scheme in a victim system, attackers can encrypt files without exposing their private key of the asymmetric encryption algorithm. The encrypted file data D in fourth equation cannot be decrypted without the private key sk if both symmetric and asymmetric cryptography algorithm are sound.

3 Our Approach

3.1 Abstraction of Process

To defend against ransomware attacks, we analyzed common processes of three major ransomware families and abstracted their process. The three major ransomware families are WannaCry, Locky, and Cerber. We derived them from 2017 statistics in [32]. We focus on their process only begin their binary to end of their encryption, not after encryption. As our purpose of abstraction is finding defending points against ransomware attacks, We did not focus on their specific notification for marketing. Abstracted pseudo code of their process is shown in Algorithm 1.

Algorithm 1 Pseudo Code of Ransomware Process

```
1: Soc = ServerConnection
2: if Soc = NULL then
3:   Exit()
4: pubkey = Soc.ReceiveKeyFromServer
5: while filename = OneOfTargets do
6:   pFile = ReadFile(filename)
7:   symKey = GenKey()
8:   cFile = Enrypt(symKey, pFile)
```

3.2 Division and Countermeasures

We categorize their process into several parts based on the code. The result of categorization is shown in Table 1. And then we list each part of processes by matching their process with existing researches. By comparing them, we can find what kinds of research against ransomware attacks exist before and what kinds of research are needed. We categorized the process into four phases (see Table 1). First, 'Key Receive from Server' includes 1 to 4 lines in the pseudo code. In this part, ransomware in system tries to get a public key of asymmetric encryption from its server. Second, 'Finding Target Files' includes 5 to 7 lines in the pseudo code. In this part, ransomware tries to find target files in victim system. As usual, in Windows, they use *FindFristFile* and *FindNextFile* functions recursively. When they find a file, they check if the file has one of their target extensions. If the file is one of the targets, it goes to next step. Third, 'Key Generation' includes 8 line in the pseudocode. In this part, ransomware generates a symmetric key for each target file. The last, 'File Encryption' includes line 9 in the pseudocode. In this part, the file is encrypted with the symmetric key. The symmetric key is encrypted by the public key sent from the server.

As we made our constitution of their process, we thought about their Countermeasure and related research about it. The two phases of four phases are about getting keys for encryption. The countermeasure for them is only getting their keys. Decryption tools [12, 13] by Anti-virus Companies and [24] are the results. As the fourth part 'File Encryption' makes a noticeable variation in the victim system, precedent studies deal with this point. In the middle of finding researches, we found There was little research on the second phase 'Finding Target Files'. The only work was the Ransomware Option of Windows 10 by the recent update [11].

We set big store by this second phase 'Finding Target Files'. If we can spoil their targeting, ransomware binary is not severe for victims. Because ransomware attacks are about business, they do not attempt severe attacks like Denial of Service (DoS) attacks or Vandalism except earning money. The easy and strong way to stop their finding our valuable files is access control. Recently, Windows 10 updated with anti-ransomware access control function [11]. If victims use this option, it will work against ransomware attacks as they cannot access to optional directories. However, when ransomware obtains all privileges, we cannot

guarantee they cannot access our valuable files. Thus, we thought another way. One is blocking them, for example, access control. The another way is avoiding them, for example, we can change file extensions of victim’s system. In next sections, we will discuss this strategy and test it.

Table 1. Phase and Countermeasure

Phase	Countermeasure	Related Research
Key Receive from Server	Key Acquisition	Decryption tools [12, 13]
Finding Target Files	Blocking (Access Control)	Anti Ransomware Option in Windows [11]
	Avoiding	None
Key Generation	Key Acquisition	[24]
File Encryption	Monitoring Files and Resources	HelDroid [16] Cutting [23], UNVEIL [22], Cryptolock [30], SeildFS [18]

4 Avoiding Targeting from Ransomware

In this section, we discuss what attackers try to find and how we can avoid their targeting on the victim system. Then we suppose one method to avoid targeting of ransomware with assuming the victim system is Windows Operating System. We use this method in next section and validate its effectiveness.

4.1 Form of Valuable Files

In section 2.2, we described what ransomware attackers try to encrypt and how they distinguish target files in victim system. Ransomware’s target is private user files cause if attackers encrypt the files can get easily from a public area, victims can restore files easily. Therefore, attackers’ first mission is finding random private files. Fortunately for attackers, in Windows, random private files have common formats those are file extensions in Figure 2. That is why ransomware checks a file has target extension or not. We use this characteristic to find our methodology.

4.2 Methodology

We can find methods to avoid based on the fact that major ransomware finds scores kinds of target files by their extension. They only encrypt files those have extensions in their hard-coded list, that is whitelist strategy. Therefore the first simple idea is randomizing extensions of our file system. We were sure that could not expect random enough extensions. We will test this method in next section.

The advanced method is randomizing the file signature in the file header. For now, major ransomware just checks file extensions, however, few ransomware

checks signature in files to check it is a valuable file of them. Therefore, we should think of this method. This advanced method will need difficult support as general applications in the system cannot recognize their randomized files with no support. If it is possible, this method will be powerful against ransomware attacks.

4.3 Effectiveness

Base on theories Our methods that are randomizing some parts of files to avoid targeting can be Moving Target Defense (MTD). Currently, many defense techniques are applying MTD. For example, We logically move our targets makes an attack take a longer time to prove our targets grounds for this theory. For example, Address Space Layout Randomization (ASLR) and The Mutable Networks (MUTE) are this kinds of techniques [26]. Even though our method is not conducting MTD [20], but we can consider our method has effect against ransomware attacks by the similar principle of MTD. We are sure that our method will make ransomware take a longer time to encrypt victims than before. When a victim system adapts our method, attackers will not be able to encrypt valuable files by the easy way like comparing extensions. They should check headers or encrypt all files. However, both two ways will have bigger overhead costs.

Base on strategies Considering in attackers view, it is not hard to switch their whitelist strategy to the blacklist strategy, that is encrypting all files they can find. The problem is this makes their evil works very inefficient. First, as they encrypt all files, the work will take longer time. This means they expose their work longer for anti-virus or users. Second, if we can make decoy files in the victim system, they cannot avoid it. The decoy files have plausible content and random names. And they are located in the path that users do not access in ordinary. Anti-virus monitors decoy files than monitoring all system. It is efficient, especially for ransomware attacks those use blacklist strategy [6].

4.4 File Extension Randomization

At present, we confirm that many ransomware is target based on the extension. Therefore we selected and implemented a method to randomize the extension. The overall structure of the program we designed is the same as the pseudo code below. A detailed description follows in Algorithm 2.

Set Target File Extensions to Protect We choose seven file extensions(.docx, .hwp, .pdf, .pptx, .txt, .xlsx, and .zip) those are frequently targeted by ransomware attacks.

In our study, we selected only seven file extensions. The reason is limited to seven because the purpose of the study is to see how effectively the extension randomization can defend against ransomware. WannaCry encrypts files with extensions ranging from dozens to hundreds. Based on the results of this study,

Algorithm 2 Pseudocode of Extension Randomization to Avoiding Targeting

```
1: Soc = ServerConnection
2: if Soc = NULL then
3:   Exit()
4: pubkey = Soc.ReceiveKeyFromServer
5: while
6:   if then filename = OneOfTargets
7:     pFile = ReadFile(filename)
8:     symKey = GenKey()
9:     cFile = Enrypt(symKey, pFile)
```

Table 2. Changed File Extension

Original File Extension	Changed File Extension	Document Type
.docx	.juev	Microsoft Word
.hwp	.tzqi	Hangul Office File
.pdf	.ohiz	Adobe Acrobat
.pptx	.nhru	Microsoft Powerpoint
.txt	.umbc	Text Document
.xlsx	.qooi	Microsoft Excel
.zip	.imko	Compressed File

if a user randomizes a file that ransomware encrypts, it will be able to defend against all extensions that ransomware attacks.

Generate Cryptographically Secure Pseudo-Random Extensions The new extensions corresponding to seven extensions we chose at former section are created in three steps.

Firstly, we use a cryptographically secure pseudo-random lowercase alphabet string of length 4. We used the Python `os.urandom` module for getting randomness. This module is suitable for cryptographic use [14]. As our randomness relies on CSPRNG, it is computationally impossible to know which random extensions will be used for attackers. Therefore, it is very difficult for ransomware attackers to predetermine a new extension as a target. As Window system extension has length 3 or 4. Our generator creates a 4-length random string that has a larger space available as an extension.

Secondly, it confirms whether the newly created extension is already used by the system through the registry query. If the extension value is already in use, create a new value and check again.

And lastly, if the new extension is not used, the registry key value of the extension to be replaced is read and registered in the new extension key value. Thus, even if the extension of file changes to a newly created extension, the OS recognizes the format of the file and can guarantee the usability of the system user [2]. Figure 3 shows the registry key of pdf file extension and Figure 4 shows

the registry key of the ohiz file extension, that is the generated random extension corresponding to the pdf file.

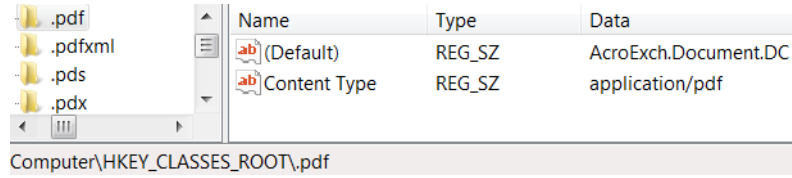


Fig. 3. File Extension Stored in Registry

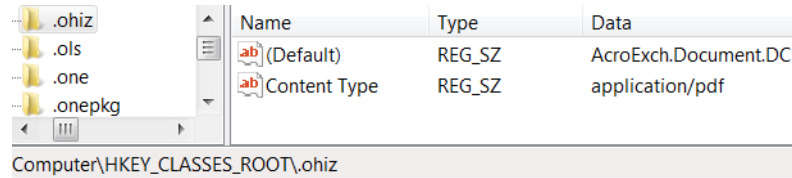


Fig. 4. Generated Random Extension Stored in Registry

Change All Target File’s Extension. We change the extension of the files by referring to the new random extensions those we generated before. Figure 5 shows the system recognize new extensions we generated and match the files to proper programs.

5 Experiment

In this section, we show how we design our experiment and demonstrate that our method is valid for ransomware. We point out this experiment is implemented under the assumption ransomware act like our abstraction of the process at the former section.

5.1 Experiment Setup

To set up a tempting system for our test, we selected Windows 7 with reference to the fact that Windows 7 shares the largest parts of the desktop operation market in [8]. For that reason, our experiment environment is Windows 7 SP1 x86








Name	Date modified	Type	Size
 Chilly_altruist.plaz	2017-08-09 오후 1...	한컴오피스 NEO ...	64 KB
 Hello.umbc	2017-11-05 오후 5...	Text Document	1 KB
 Huffman.imko	2017-11-05 오후 5...	압축(ZIP) 파일	11,684 KB
 IRPHooking.ohiz	2017-03-31 오전 1...	Adobe Acrobat D...	568 KB
 myself.juev	2017-09-24 오후 1...	Microsoft Word 문...	166 KB
 TTP.nhru	2017-09-24 오후 1...	Microsoft PowerP...	1,741 KB
 voice.qooi	2017-09-24 오후 1...	Microsoft Excel 워...	61 KB

Fig. 5. Victim file after extension change

virtualized in VMware Workstation. We used VMware Workstation to rollback our experiment system after ransomware infection testing.

We made random selections from the set of .docx, .hwp, .pdf, .pptx, .txt, .xlsx, and .zip files. We made a new subdirectory of C drive of our experiment environment. And we put them in the subdirectory we made. Three reasons contribute to the design. Firstly, we select seven of frequent target file extensions of ransomware attacks. Secondly, attackers target victim's own valuable files. If general files those are included in a lot of systems are encrypted, the victim user can bring the copy of them from other systems. Random files and random directory make them look like a set of customized files by attackers. Thirdly, main ransomware filter exiting directories before encrypting [15,21]. Therefore, we should make a new directory and put random files in here.

5.2 Data Set

We obtained 143 active ransomware samples out of whole 350 total ransomware samples from Virusshare.com using ransomware keyword searching. Figure 6 displays a pie chart that represents our full ransomware samples with denoting ransomware families. Our samples include 'WannaCry', 'Cerber', 'Locky', 'Tesla Ransomware', which get a lot of share of total ransomware attacks [32]. The active ransomware sample means ransomware that encrypts, in effect, the set of random files in our experiment environment. As several factors influence ransomware's behavior, they do not always encrypt the files of the victim. For example, C&C server connection is a necessary process to get public keys for ransomware [15,17]. Other factors can be a function of detecting virtual machines [33] or fault in development.

5.3 Experiment

The experiment procedure was as follows. We observed the set of random files and normal files after we executed each ransomware sample file to check during

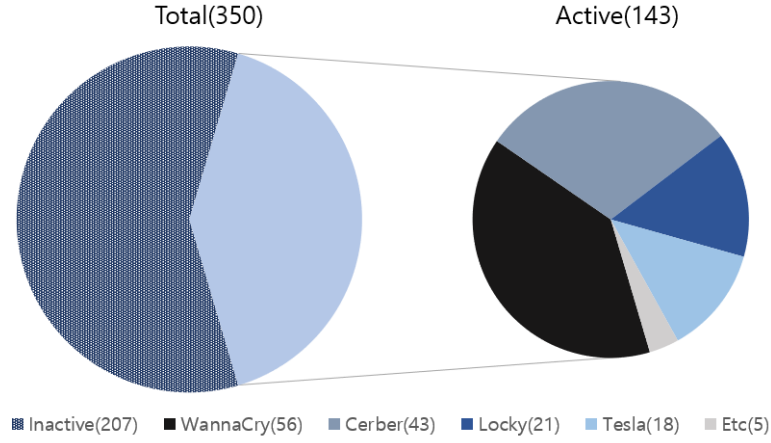


Fig. 6. Ransomware Samples for Test

5 minutes. We judged the 'defense' is successful if two conditions are satisfied. One is that ransomware encrypts all normal files. Another is all files in the set of random files are not modified. If we can check files are being encrypted, and it exceeds our time limit (5 minutes), we extended our time limit on our discretion to check the full behavior of ransomware.

5.4 Result

In most cases, Our random files avoided ransomware's encryption. The result was shown in Table 2. We prevented encryption in 141 cases of 143 active ransomware samples. In our experiments, 56 out of 56 in WannaCry family were successful in defense, 43 out of 43 in the Cerber family, 21 out of 21 in the Locky family, and 18 out of 18 samples in the Tesla family succeeded in defending did. Our approach had difficulty defending two of the samples we tested. The two samples are ZeroLocker and Satan Ransomware, which encrypt all files except for a specific directory.

5.5 Evaluation

The evaluation of our experiments can be shown efficiently with three questions and answers.

- Q1. Did we protect our files from ransomware?
- Q2. Isn't it a defensive defense to certain ransomware families?
- Q3. Did we overuse your system's resources?

- About Q1: As you can see in Table2, we protected our set of randomized files from 141 ransomware samples from a total of 143 active ransomware samples.

Table 3. Experiment Result

Ransomware Family	Success count/ Total count	Defense Rate	Note
WannyCry	56/56	100%	
Cerber	43/43	100%	
Locky	21/21	100%	
Tesla	18/18	100%	
Etc	3/5	60%	ZeroLocker, SatanLocker
Total	141/143	98.6%	

Ransomware could not find and encrypt the file as our methodology targeted its general characteristic of targeting files. We can say that we have shown the right answer to this question.

- About Q2: Our strategy only depended on Ransomware’s general whitelist strategy. We protected our set of randomized files from WannaCry, Cerber, Locky, Tesla Ransomware, which get a lot of share of total ransomware attacks [32], in all cases. Our method is not related to that how detailed the technique is, whether ransomware uses a cryptographic method or a cryptographic function. Even if ransomware is doing some normal operation in a system with a powerful zero-day, our approach is useful if only the strategy is matched. That is why our result shows good defense rate from our ransomware samples. Therefore, unless ransomware uses a blacklist strategy or there is no overall strategy change, our approach will continue to be useful.
- About Q3: Our method requires a preprocessing process only in the process of randomizing the system. This technique does not change the file after modifying the registry, it only changes the extension, therefore it consumes little time. After that, this method does not consume any resources while it remains in the system. It is our unique advantage that cannot be found in existing signature-based, heuristic-based ransomware countermeasures.

6 Discussion and Limitation

Ransomware attacks generally took a whitelist strategy, and we got good results because we randomized the extensions. However, we understand the need for limitations and further research in four ways.

First, randomizing file extensions can make problems of randomizing system data. Randomizing data has a limitation. The encoding point of the data to be randomized is limited. Now, ransomware has simply looked for the target based on the extension, thus we only proceed with randomization using the extension, but other methods also exist. As extensions are randomized, users can get confused in using their system. Even users who get files from other randomized systems can get confused because their system cannot recognize randomized files. Therefore, it is necessary to apply randomization considering system and network usage.

Second, this method does not provide any guarantee that ransomware attacks will stick to whitelist strategy forever. We must keep an eye on how ransomware will go to targets. Ransomware attacks using a blacklist strategy, which encrypts all files in a system, exists in very small numbers. For example, our two exceptional results in Table 3 are them. It shows applying blacklist strategy is not a problem as ransomware. We guess the overhead that occurs encrypting all files makes ransomware developers choose whitelist strategy. If ransomware attacks using such a blacklist strategy become rampant, our method will be ineffective. Therefore, countermeasures against ransomware attacks using blacklist strategies should be studied. For example, blacklist strategy is very weak to decoy file. We can think of monitoring the attempt to modulate a randomized file as a decoy file. The blacklist strategy will try to access all the files that cannot be targeted, therefore if you throw a random file to the program and try to encrypt it, we can judge the system is under ransomware attacks. We expect if we use both randomizing and decoy, we can prevent both blacklist and whitelist ransomware.

Third, the concept of moving ransomware’s target files is related to the concept of MTD. Since ransomware attacks use malware with a clear target and business model, MTD is a concept that should be considered in defending against ransomware attacks. The technique applied in this paper is still difficult to see in the category of MTD because it is a logical move only once in non-target periodically moving experiments. We can get better results if we apply the model that applies the technique of moving the target continuously against ransomware with the existing MTD research.

Fourth, to defeat ransomware, we should break their business. Ransomware shows the business model as known as RaaS and we made their business model in Figure 1. This fact means countermeasure against ransomware is not only about its technique, like C&C connection and hybrid cryptography on user system but also it is about their business. Nevertheless we limited our research on ransomware in technical ways in this paper. There will be a considerable effect if their business is damaged.

7 Related Works

Ransomware has been a major threat for several years, and many researchers have been studying how to defend against Ransomware. There have been researches on how to protect Ransomware infected files and how to recover files after Ransomware infection. Table 4 summarizes previous research on Ransomware defense methods and our research. Most research suggests that Ransomware makes it difficult to encrypt files. Since Ransomware is malicious code, malicious code defense method is applied. The most basic method of malicious code defense is signature-based detection. Ransomware developers are creating many variants with different hash values within the same family by automatic modularizing and changing some code on its own to bypass this detection. Signature-based Ran-

somware detection is limited for this reason. Another method for Ransomware detection is behavior-based detection.

Scaife *et al.* [30] emphasized that the most important point in dealing with ransomware attacks is realizing the problem is all about data. Thus, their research did not limit to specific ransomware families but instead applies to a wide range of Ransomware. As with their research, this study also focused on how to protect data from ransomware attacks. Additionally, behavioral based detection may involve overloading the system cause they should monitor system resources at all times. Andronio *et al.* [16] study focused on Android mobile Ransomware. In Kharraz *et al.*'s study [23], they researched how to protect Ransomware by investigating file I/O mainly in Ransomware from 2006 to 2014. Kharraz *et al.*'s work explored how to detect Ransomware through dynamic analysis of system calls [22].

There was also research on how to recover data after infection by Ransomware. Most of the methods for recovering files from Ransomware is file backup approach. Continella *et al.* researched how to protect data from Ransomware using Windows native filesystem immune.

8 Conclusion

WannyCry, ExPetr, Bad Rabbit Ransomware, which hit the world in 2017, infected many systems and demanded cryptocurrency such as Bitcoin, from a large number of people. Recently, Ransomware has become more sophisticated and compact combined with zero-day attacks as well as encrypting important files. The damage of Ransomware is continuously increasing, and the security measures against it are also being researched steadily. Our research was based on an elementary idea. To receive the money for decrypting a file from the user due to the characteristics of Ransomware, all the files are encrypted so as not to destroy the system but to encrypt only the extension of the specific file. We thought about changing the extension of important files so that Ransomware could not encrypt the files. And Microsoft has been able to interoperate with programs that can open the file using the registry even if the user changes the extension. In this way, we made changes to the seven major extensions and created an ordinarily available Windows operating system environment. And after testing 143 recently discovered Ransomware samples, we succeeded in defending all 141 except Ransomware, which encrypts all files except certain folders. This experiment was a straightforward approach, but it showed strong security. We hope that many further studies will be conducted based on the results of this experiment.

References

1. Application registration. [https://msdn.microsoft.com/kokr/library/windows/desktop/ee872121\(v=vs.85\).aspx](https://msdn.microsoft.com/kokr/library/windows/desktop/ee872121(v=vs.85).aspx), [Online; accessed 11-February-2018]

Table 4. Ransomware Countermeasure Categorization

Countermeasure Categories	Sub Categories	Papers	Ransomware Families
Making Access to Data Difficult	Detecting Ransomware	Cryptolock (and Drop It): Stopping Ransomware Attacks on User Data [30]	CryptoDefense
			CryptoFortress
			CryptoTorLocker
			CryptoWall
			CTB-Locker
			Filecoder
			GPcode
			MBL advisory
			PostCoder
			Ransom-FUE
		HelDroid: Dissecting and Detecting Mobile Ransomware [16]	TeslaCrypt
			Virlock
			Xorist
			Android Malware
			Revention
			Cryptolocker
			CryptoWall
			Tobfy
			Seftad
			Winlock
		Cutting the Gordian Knot: A Look Under the Hood of Ransomware Attacks [23]	Loktrom
			Calelk
			Urausy
			Rotten
			BlueScreen
			Kovter
			CryptoLocker
			CryptoWall
			CTB-Locker
			CryptVault
		UNVEIL: A Large-Scale, Automated Approach to Detecting Ransomware [22]	CoinVault
			FileCoder
			TeslaCrypt
			Tox
			VirLock
			Revention
			Tobfy
			Urausy
			Windows Anti-Ransomware Function
			TelsaCrypt
	Extension Randomizaiton	Our Approach	WannyCrypt
			Cerber
			CryptoLocker
			CryptoWall
			Crowti
			CryptoDefense
			Critroni
Recover Data After Encryption	Detecting Ransomware	ShieldFS: A Self-healing, Ransomware-aware FileSystem [18]	

2. Hkey classes root key. [https://msdn.microsoft.com/ko-kr/library/windows/desktop/ms724475\(v=vs.85\).aspx](https://msdn.microsoft.com/ko-kr/library/windows/desktop/ms724475(v=vs.85).aspx), [Online; accessed 11-February-2018]
3. Microsoft security bulletin ms17-010 - critical. <https://docs.microsoft.com/en-us/security-updates/securitybulletins/2017/ms17-010>, [Online; accessed 11-February-2018]
4. Taidoor campaign targets government agencies in taiwan. <https://www.trendmicro.com/vinfo/us/security/news/cyber-attacks/taidoor-campaign-targets-government-agencies-in-taiwan>, [Online; accessed 11-February-2018]
5. Alma ransomware: Analysis of a new ransomware threat (and a decrypter!). <https://info.phishlabs.com/blog/alma-ransomware-analysis-of-a-new-ransomware-threat-and-a-decrypter> (2016), [Online; accessed 24-August-2016]
6. V3 offers the best practical ransomware defense for pcs. <http://mglobal.ahnlab.com/site/securitycenter/securitycenterboard/securityInsightView.do?seq=2311> (2016), [Online; accessed 11-October-2016]
7. Erebus resurfaces as linux ransomware. <https://blog.trendmicro.com/trendlabs-security-intelligence/erebus-resurfaces-as-linux-ransomware/> (2017), [Online; accessed 11-February-2018]
8. Operating system share by version. <https://netmarketshare.com/operating-system-market-share.aspx> (2017), [Online; accessed 11-February-2018]
9. Ransomware damage report. <https://cybersecurityventures.com/ransomware-damage-report-2017-5-billion/> (2017), [Online; accessed 11-February-2018]
10. Ransomware faq. <https://www.microsoft.com/en-us/wdsi/threats/ransomware> (2017), [Online; accessed 19-December-2017]
11. Stopping ransomware where it counts: Protecting your data with controlled folder access. <https://cloudblogs.microsoft.com/microsoftsecure/2017/10/23/stopping-ransomware-where-it-counts-protecting-your-data-with-controlled-folder-access/?source=mmmpc> (2017), [Online; accessed 23-October-2017]
12. Downloading and using the trend micro ransomware file decryptor. <https://success.trendmicro.com/solution/1114221-downloading-and-using-the-trend-micro-ransomware-file-decryptor> (2018), [Online; accessed 31-January-2018]
13. Free ransomware decryptors. <https://noransom.kaspersky.com/> (2018), [Online; accessed 30-January-2018]
14. Wannacry malware profile. <https://docs.python.org/2/library/os.html> (2018), [Online; accessed 11-February-2018]
15. Alex Berry, Josh Homan, R.E.: Wannacry malware profile. <https://www.freeeye.com/blog/threat-research/2017/05/wannacry-malware-profile.html> (2017), [Online; accessed 11-February-2018]
16. Andronio, N., Zanero, S., Maggi, F.: Heldroid: Dissecting and detecting mobile ransomware. In: International Workshop on Recent Advances in Intrusion Detection. pp. 382–404. Springer (2015)
17. CERT-MU: The wannacry ransomware. Tech. rep. (May 2017)
18. Continella, A., Guagnelli, A., Zingaro, G., De Pasquale, G., Barengi, A., Zanero, S., Maggi, F.: Shieldfs: a self-healing, ransomware-aware filesystem. In: Proceedings of the 32nd Annual Conference on Computer Security Applications. pp. 336–347. ACM (2016)
19. Dent, A.W.: Hybrid cryptography. IACR Cryptology ePrint Archive 2004, 210 (2004)
20. Evans, D., Nguyen-Tuong, A., Knight, J.: Effectiveness of moving target defenses. In: Moving Target Defense, pp. 29–48. Springer (2011)

21. hasherezade: Cerber ransomware new, but mature. <https://blog.malwarebytes.com/threat-analysis/2016/03/cerber-ransomware-new-but-mature/> (2016), [Online; accessed 11-February-2018]
22. Kharraz, A., Arshad, S., Mulliner, C., Robertson, W.K., Kirda, E.: Unveil: A large-scale, automated approach to detecting ransomware. In: *USENIX Security Symposium*. pp. 757–772 (2016)
23. Kharraz, A., Robertson, W., Balzarotti, D., Bilge, L., Kirda, E.: 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 (2015)
24. Kolodenker, E., Koch, W., Stringhini, G., Egele, M.: Paybreak: defense against cryptographic ransomware. In: *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*. pp. 599–611. ACM (2017)
25. Lord, N.: What is ransomware as a service? learn about the new business model in cybercrime. <https://digitalguardian.com/blog/what-ransomware-service-learn-about-new-business-model-cybercrime>, [Online; accessed 11-February-2018]
26. Okhravi, H., Streilein, W.W., Bauer, K.S.: Moving target techniques: leveraging uncertainty for cyber defense. Tech. rep., MIT Lincoln Laboratory Lexington United States (2015)
27. Osterwalder, A., Pigneur, Y.: *Business model generation: a handbook for visionaries, game changers, and challengers*. John Wiley & Sons (2010)
28. Pankov, N.: Expetr targets serious business. Tech. rep. (June 2017), <https://www.kaspersky.com/blog/expetr-for-b2b/17343/>
29. Perekalin, A.: Bad rabbit: A new ransomware epidemic is on the rise. Tech. rep. (Oct 2017), <https://www.kaspersky.com/blog/bad-rabbit-ransomware/19887/>
30. Scaife, N., Carter, H., Traynor, P., Butler, K.R.: Cryptolock (and drop it): stopping ransomware attacks on user data. In: *Distributed Computing Systems (ICDCS), 2016 IEEE 36th International Conference on*. pp. 303–312. IEEE (2016)
31. Shakir, H.A., Jaber, A.N.: A short review for ransomware: Pros and cons. In: *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*. pp. 401–411. Springer (2017)
32. Sinitsyn, F.: Kaspersky security bulletin: Story of the year 2017. Tech. rep. (Dec 2017), <https://securelist.com/ksb-story-of-the-year-2017/83290/>
33. Sison, G.: Cerber starts evading machine learning. <https://blog.trendmicro.com/trendlabs-security-intelligence/cerber-starts-evading-machine-learning/>, [Online; accessed 11-February-2018]
34. Symantec: Internet security threat report. Tech. rep. (April 2017), <https://www.symantec.com/content/dam/symantec/docs/reports/istr-22-2017-en.pdf>
35. Vinod, P., Jaipur, R., Laxmi, V., Gaur, M.: Survey on malware detection methods. In: *Proceedings of the 3rd Hackers Workshop on computer and internet security (IITKHACK09)*. pp. 74–79 (2009)