Identifying File Interaction Patterns in Ransomware Behaviour

Chapter · September 2018

DOI: 10.1007/978-3-319-92624-7_14

2 authors, including:

Simon Parkinson
University of Huddersfield
48 PUBLICATIONS **248** CITATIONS

SEE PROFILE

Identifying File Interaction Patterns in Ransomware Behaviour

Liam Grant and Simon Parkinson

Abstract Malicious software (Malware) has a rich history of causing significant challenges for both users and system developers alike. The development of different malware types is often resulting from criminal opportunity, and the monetisation of ransomware, coupled with the continuously growing importance of user data, is resulting in ransomware becoming one of the most prominent forms of malware. Detecting and stopping ransomware from executing is challenging due to the large verity of different types, as well as the speed of new instances being developed. This results in static approaches, such as using signatures, ineffective in many instances. This chapter investigates the behavioural analysis of ransomware, and in particular focussed on its interaction with the underlying file system. This study identifies that ransomware instances have unique behavioural patterns, which are significantly different from those of normal user interaction.

1 Ransomware

1.1 Introduction

Ransomware is a type of malicious software, or malware, that aims to cause damage to a single computer, server or computer network [1]. A key difference of ransomware over other forms of malware is that it aims to hold the user ransom by encrypting their data. The continuing increase in ransomware is as result of com-

Liam Grant
Department of Computer Science, University of Huddersfield, UK e-mail: u1470723@hud.ac.uk

Simon Parkinson
Department of Computer Science, University of Huddersfield, UK e-mail: simon.parkinson@hud.ac.uk

puting devices continuing to be a major part of users social and professional lives, thus increasing the dependency on data and the opportunity to monetise through crime. Malware is in a constant state of evolution, with new evasion techniques being introduced for every advancement in detection and prevention that is made in the security field [2].

The first known ransomware was created by Joseph L. Popp and was used at the World Health Organizations AIDS conference in 1988. It used very simple cryptography to encrypt files, due to this the files were easily decrypted. The virus was named the AIDS Trojan due to the attack [3]. Since then, ransomware has advanced with the general advancement of technology and now has several different families. Analysis by [4] shows the top 8 ransomware variants, with distribution dominated by *cerber* and *locky* variants in 2016.

Ransomware encrypts a users data using private key encryption or by preventing access to a computer. Using these methods, the ransomware holds the data on the machine ransom until the end user pays the requested amount, usually in Bitcoin or similar cryptocurrency [3]. Most of the time, end users do pay this ransom, especially if the encrypted data is valuable to them or their corporations. However, it is sometimes possible to get this data back with shared keys and applications from organizations such as No More Ransom [5].

When broken down, the distribution of different malware types is dominated by ransomware, which has been the ongoing trend for many years. In the first quarter of 2017, ransomware accounted for over 50% of recorded malware distribution [6]. In 2016, the global average of ransomware detections was 1,271 per day, which was an increase of 933 from the previous year [7]. However, [5], predicts that towards the end of 2017, the volume and effectiveness of ransomware attacks should drop due to anti-ransomware technologies and initiatives, and also the increased end-user awareness.

A common reason for this dominance and growth in the use of ransomware is that it is the most successful at extorting money from end users, which in turn makes it a more attractive method to current and future cyber-criminals. In 2016, the IC3 (Internet Crime Complaint Center) released a report showing that of the 2,673 complaints filed with their service, they accrued loses of over $2.4 million [8]. However, end users that are businesses arent only paying for a ransom; another report states that between April 2014 and June 2015, that the total loss for victims attacked by the CryptoWall ransomware variant was over $18 million, including the expenses for IT Services or legal fees [9].

Considering the recent increase in ransomware, and the necessity to prevent users from being exploited into paying ransom fees, there is strong motivation to better understand the behavioural habits of ransomware to provide new insights leading to stronger detection and prevention mechanisms. There are many behavioural characteristics of ransomware; however, in this work its interaction with the underlying file system is examined. As ransomware is programmed to perform systematic tasks (directory traversing, file identification, and encryption), it can be assumed that the software will have unique characteristics as to how these functions are performed.

In this Chapter, an empirical investigation is performed whereby a methodology is derived whereby samples of ransomware are executed whilst monitoring the underlying file system to identify any behavioural patters. The Chapter is structured as follows: in the first section, information is provided on the inner workings of ransomware as well as providing information on other common detection types. Related work is then discussed before a methodology of how to monitor file activity during a ransomware attack is presented. Key results are then presented and discussed, and finally, a conclusion lays out directions for future research.

1.2 How does Ransomware Work?

Core Functionality

Ransomware can be sorted in to two core types; (1) crypto ransomware and (2) locker ransomware. Locker ransomware is used to lock a computer or device, thus preventing the user from accessing it. Although this type does hold the data to ransom, it does not do so at a file level, meaning that if it is removed from a device, its data remains intact. Due to this locker ransomware is not as effective at extorting victims for ransom and other types [10].

The second type, crypto ransomware, is much more effective at extorting a payment. It essentially takes control of data stored on a system by encrypting it. Even if a victim can manage to remove the malware, the files will still be encrypted and unable to be accessed. Although it is possible to decrypt the data with the private key used, it is very hard and time consuming to do so and usually the victim will either pay the ransom or lose the data [10].

A majority of crypto ransomware software uses Elliptic Curve Diffie Hellman (ECDH) or RSA, named after Ron Rivest, Adi Shamir and Leonard Adleman, combined with AES (Advanced Encryption Standard) encryption [11]. Using a RSA public key to encrypt files, and separate private, or secret, key to encrypt the AES keys which were used in the encryption of data [12].

At the core of almost every modern ransomware attack, the is a C&C (command and control) server that the software requires communication with [13]. The C&C usually acts as the holder of one or more keys, which is required to encrypt the data, or to encrypt the keys that have encrypted the victims data [14]. Without communication with the server, the ransomware is unable to get the resources required to encrypt files on a machine. However, due to the importance of the internet in our modern lives, this is an increasingly easy problem for attackers to get around. Some ransomware can be stopped from the use of a blacklist/whitelist to block communication with known malware C&C server IP addresses; however, this is easy to get around, by either changing IP or domain names.

Over the past few years, ransomware and other malware have started to have more advanced evasion features built into them. Theses evasion techniques are used

to either bypass security software, or to avoid being analysed or run in test environments. [15], classifies these into three broad categories:

- **Anti-security techniques:** The avoidance of detection by anti-malware engines, firewalls or application containment.
- **Anti-sandbox techniques:** The ability to detect what kind of environment the ransomware is running in by checking registry key, files or processes related to virtual environments.
- **Anti-analyst techniques:** The ability to detect monitoring processes, such as Process explorer or Wireshark. In some cases, attempting to confuse analysts by obfuscating data or creating false traffic.

Targeting

Some cyber criminals select their targets in advance, preparing ransomware in bespoke software created to cause the maximum amount of damage to a victim. [16] In these cases, the victim almost always is an organisation rather than individuals, as the attackers are spending this time to effectively get a load ransom pay-out from the victim [15]. Attacks like this are hard to do, and generally need a lot of technical competence to pull off, usually requiring a selection of planned and perfectly executed stages. These methods generally involve using network administration software to gain access and move throughout a network system, stealing of credentials for the system itself and performing advanced reconnaissance to fully understand the network and how best to affect it [16].

Distribution

Ransomware does not just appear on a system; it requires some method of distribution. There are many methods for distributing malware, with some more effective than others. The main types can be broken into these categories:

- **Email** is one of the main types of successful distribution for ransomware, often made possible through large-scale malicious spam campaigns. These campaigns and usually spread via a botnet-network of hundreds to millions of infected PCs. These botnets send out hundreds of thousands of emails daily, targeting individuals using social engineering techniques to trick users into think the mail is important (e.g. fake invoices, infected attachments, etc.) [7].
- **Exploit kits** are the second most common distribution method, which work by exploiting vulnerabilities in other software which can then be used to infect machines. These kits are spread using spam email and malvertising, advert links that link to sites. Once a exploit kit has been setup on a victims computer, it works as a back door into the system for malware to infect [5].
- **Self-propagation** is not as common as other techniques; however some ransomware families and variants a build with some level of self-propagation. Copy-

ing itself onto removable drives or networked computers. Some Android ransomwares can spread themselves through a list of contacts using SMS messages [16].

- **Other** There are many other types of distribution for ransomware, but are just not that common compared to the three types mentioned above. Malvertising, is one of these techniques, misleading victims by seeming like an advert which usually links to a compromised website [16]. This technique is more common in the spreading of exploit kits which attackers can then use to infect a victims computer. Another method used in the brute forcing of passwords, generally for server applications, which can then be used to infect machine connected to the core server [15].

Cryptolocker: An Example of Crypto Ransomware

Cryptolocker is a family of crypto ransomware that was used in a large scale cyber-attack in late 2013 to mid-2014. It used Bitcoin transactions for the ransom currency and Liao et al. [17], found 795 payments adding up to 1,128.40 BTC. CryptoLocker was spread via email using social engineering techniques to trick recipients into running infected attachments in emails claiming to be from a logistics company[18].

CryptoLocker uses a combination of public keys for distribution, and private keys for large-scale encryption. Using these two methods together, the software attempts to connect to a C&C server which then creates a RSA public/private key pair and sends the public key back to the infected machine [12]. CryptoLocker then generates AES private keys and starts to encrypt data on the victims machine, usually using different keys for each file extension type. Once it has finished encrypting file, the RSA public key, obtained from the C&C server, is then used to encrypt the AES private keys. It then generates a page, informing the victim that the files have been encrypted and how to pay the ransom [12].

1.3 Current State of Ransomware

WannaCry

There have been many large-scale ransomware attacks over the past few years, most recently being the WannaCry attack in May 2017. It affected more than 70,000 computers around the world after its first few hours, totaling over 200,000 computers [19]. WannaCry targeted 176 different file types, which were appended with `.WCRY` after encryption. It asked for a ransom in Bitcoin to the value of roughly 300 US Dollars, and claim that each all the files would be deleted if the ransom was not paid in 7 days [20]. WannaCry caused massive damage to several companies and organisation across the world.

Ransomware-as-a-Service

Ransomware has been a rising tool of cyber criminals over the past few years, Ransomware-as-a-service (RaaS) has attributed to this greatly. RaaS offers cyber criminals the ability to hold victims data for ransom, for only a small cut of the profit [21]. Allowing less technologically competent criminals to take advantage of ransomware. RaaS creators are able to host their code and systems on the dark web, where affiliates can subscribe to the service [22]. Affiliates can then configure any aspects that they need and deploy, allowing them to increase ransom. They can even look at an estimate of their earning potential before they subscribe [21].

These RaaS packages are usually free to deploy, but have a profit-sharing model, in which the earning from ransoms paid are split between the affiliates and the author [23]. When a victim pays to regain access to their files, this payment is delivered to the authors account or accounts and the author then distributes this back to the affiliates. These shares in profit ranges from 60 to 80 percent, making it worthwhile for an affiliate and creating a sizable profit for authors with more affiliates [22].

New Initiatives

On July 25th 2016, the Dutch National Police, Europol, Intel Security and Kaspersky Labs announced that they were joining together to create a new anti-ransomware initiative called “No More Ransom” [24]. This initiative was put together to form a new level of cooperation between public and private sectors against the threat of ransomware. They have created an online portal which aims to inform the public about the threats that ransomware pose and offers help to victims looking to recover their data without conceding to the attackers demands [25].

No More Ransom offers decryption tools for 104 known ransomware families. The cooperation between each private sector and law enforcement body increases this pool, using a combination of shared encryption keys and research into ransomware [26]. No More Ransom and grown since its initial announcement to be well over 100 partners all working towards a limiting the threat and stopping the rise in ransomware use [27].

However, there can be a negative effect from these types of initiatives. Providing applications and help to people to combat the use of ransomware also allows for the cybercriminals themselves to have access to this material and give them a helping hand in avoiding these methods of help.

Ransomware is still a major threat to users worldwide, and large-scale attacks, such as WannaCry and petya are still causing massive amounts of damage to organisations outside of ransom demands. With online supply of ransomware available to anyone willing to make use of it, the threat could grow even larger as a result. However, organisations such as No More Ransom can help to hinder these in the future, or at least to reduce the number of types of ransomware.

2 Detection of Ransomware on a System

As discussed in the previous section, ransomware is a constant and growing threat to computer systems. However, there are many ways of detecting ransomware on systems before and during an attack on a system.

Signature-based Detection

Signature based detection has been around since the inception of anti-virus software [28]. It works by searching through systems for any known malware signatures and flagging files that match these. These signatures are generally made up of byte code from the malicious software, when the pattern has been discovered it is then stored in a database of signatures and can then be searched through in order to match likely malware on a victim's computer or device. This method can also be referred to as static analysis [29].

Signatures are created by examining a software's byte code and extracting one or more sequences of bytes from the body of a specific strain of malware. This sequence is generally unique to that malware and shouldn't be likely to appear in other files [30]. Typically, this signature will match a variety of malware variants from the same family by matching them the the known sources store in the signature database [31].

Although this is the most common technique used by anti-virus softwares, signature based detection comes with it own disadvantages that can make them less likely to succeed in the detection malware such as ransomware:

- **Zero-day:** Signature-based detection uses the signatures of known malware, due to this new malware which has not been analysed and had its signature added to the database will not be detected by this method [30].
- **Evasion:** Malware creators could also change the signature of the software, making it unable to match to it's new signature. They can do this by applying polymorphic methods to their software or by using obfuscation techniques to re-order code [32].
- **False Positives:** Another downside of signature detection is false positives. This is when the software detects a file that has a matching signature pattern, but is not a malware file. This can cause issues, especially if the file is then removed by the anti-virus that is running [30].

Signature-based malware detection also has it's own strengths. It is a well known and understood method in the industry that has been used since the first anti-virus softwares have been in circulation [32]. It also has the benefit that malware signatures are widely available for use from online repositories and global research, protecting from all recorded threats [28].

Another method that is often used alongside standard signature-based anti-virus is the use of manual heuristics analysis for detecting malware on a system. Heuristic

detection uses rules and algorithms to look for specific commands that could be used for malicious purposes. This way, an anti-virus scanner can find malware without needing to find and match a signature[33].

To do this, a heuristics anti-virus would start scanning code for any suspicious methods and attributes of known malicious programs. Many malicious programs open files for other existing programs and modify them, a heuristic analyser can then examine the code for an application and then for each malicious command that application uses, the more suspicious the application is deemed [34]. Using a predetermined threshold, the analyser will compare it's finish analysis of the application and if it crosses the threshold it will then be marked as a malicious software or an infected file. This method of detection has a high performance rate and can be easy to implement, but like signature-based detection, it has a low detection rate for new malware. It can also produce more false positive than a signature-based method [33].

Behavior-based Detection

Behavioral detection uses the behavioral attributes of how a malware behaves on a given system. This ranges from systems calls, interaction with the file system, abnormal behavior to how a system is usually used [35]. Ransomware has core behavioral traits that help it to be easily defined in the malware category and these can be easily adopted into a detection methods to find [29].

Behavioral detection differs from signature-based detection in that rather than looking at specific sequences in byte code, it take a more abstract look at how a software works [29]. For example, Crypto Ransomware has a core behavioral trait of encryption, this means that in order for it to work as desired it must have some sort of encryption method. Knowing this is a core behavior a file system can be checked for unknown processes performing encryption behavior on a system and mark it as suspicious or protect files. In essence, behavioral detection finds suspicious activity through observing the behavior on a process [29].

2.1 Related Work

Ransomware Behavior Studies

Kharraz et al., [2] performed a study in which they analysed multiple samples of families and variants of known crypto ransomware and locker ransomware. The study used a data set of 3,921 samples of ransomware and analysis of the IRP's created when ransomware starts an attack on a system.

It was found for crypto ransomware that they can use both customised and standard encryption methods. Most customised methods of encryption were im-

plemented to decrease the chance of detect through common anti-virus analysis. However, it was found that some of the more modern crypto ransomware families make use of standard windows encryption calls using the `CryptoAPI` to use the `CryptEncrypt` functions. The analysis shows that when using these standard methods, the process creates a number of IRP packets, first calling a create method to open a file. It then reads the file to access the file contents and encrypt the data. Finally, the process writes the encrypted data back to the file [2]..

From this, Kharraz [2] suggests that using basic API call monitoring to identify an encryption attack as it is happening. This method would only work for a a less technical ransomware family that use standard encryption techniques, but could mitigate several low end attacks.

Gazet [36], finds that many ransomware family target specific file extensions, typically files such as `.doc, .odt, .zip` etc. Encrypting an entire disk would be a time consuming process and make a ransomware more liable to be detected before it has finished encrypting the drive. another reason is could be to encrypt files that the user will be familiar with the names of [36]. This can be used to to define better rules for filtering out potential threats to a system.

Scaife et al. [37], proposed a early warning protection system titled "CryptoDrop" in which through experimental analysis could stop ransomware in a average of 10 files lost. Showing that analysis of ransomware behaviors are a potential for stopping ransomware attacks on users. They achieved this through three main types of ransomware indicators.

- **File type changes:** As discussed above, ransomware typically target specific file types [36] and can often changes the type or extension of that file. Scaife et al. [37] use this to identify suspicious behavior in the file system. Although this does not inherently suggest malicious behavior from a process.
- **Similarity Measurements:** During encryption of a file, the content of that file changes, usually to the point where the file is completely unrecognizable from the original. Using this behavior pattern, Scaife et al [37] uses a similarity-preserving hash to compare a file's original content to the new content of that file after an edit, finding from a score of 0 to 100 how similar the two are.
- **File Entropy:** Entropy of encrypted or compressed files have a high entropy level due to the nature of how such actions occur. Entropy is in essence the measurement of "randomnes" of a file structure, using a prediction of letters that should appear after those that precede it [38]. Using this, it can be easily deciphered which files have been encrypted [37].

3 Methodology

In this Chapter, the investigated hypothesis is that different implementations of ransomware will have different behavioural characteristics on the file system level. This section details research undertaken to pursue this hypothesis.

3.1 File System Monitor

The main software development for this project is a file monitoring application. This application is used to monitor all interaction with files in specified file directories. The interaction will be recorded and saved in an .exe file and formatted as CSV, as this should assist to prevent the file from being affected by the ransomware. An assumption was made here that the ransomware is more likely to target data files than system executables (.exe) as encrypted an essential executable file may render the system unsustainable for displaying a ransom message. The main use of the software will be for it to run inside a virtual environment alongside crypto-ransomware samples to record file interaction data for future analysis. The software will need to run with extended privileges in the virtual environment to ensure that it has access to all files on the system. This is necessary as monitoring a file for activity (open, read, write) requires that the monitoring software has privileges to at least read the file's properties and contents.

The software has been developed in C#, and will use the `FileSystemWatcher` class, a tool built into the .NET framework that can be used to monitor file system events and handle them by type. This class can identify when files have been changed, renamed, created and deleted. The `FileSystemWatcher` tool is particularly helpful for this research as it is both efficient and reliable due to its utilisation of Windows core functionality. This application will focus on using the changed and renamed `FileSystemWatcher` features to log file system changes as a ransomware runs in the background. Once these events types have been triggered, the software will add the event data to a new line in the output file. This process is continued until the ransomware is confirmed to have stop its encryption processes.

The main functionality for the software is as followed:

- Monitor the file system for any changes to file content or renaming of files;
- Record this data is a CSV format so it easily portable in data analysis software;
- Output file is kept in a secure location so it can be safe from the ransomware malware instance; and
- Data stored on the files changes should be in the following format:
  `timestamp, New file Name, New directory, old file name, old directory, event type, file size`.

Ransomware can use a variety of methods to encrypt files, but mostly either performs an edit and rename, where it encrypts the data and the renames the file with a custom extension, or a delete and create, where the contents of the file are copied into another document with is then encrypted, and the original file will be deleted. Due to this, some of the information that is being created can be missing from the log file; however in these cases, these fields will be stored as empty.

3.2 Virtual Environment

The virtual environment will need to be set-up to have a file structure representative of a normal users file system. To do this, roughly 600 example files (PDF, word, ect.) will be stored in the environment.

The file structure created for this experiment will consist of five main folders, each named as ”folder” suffixed with a number from 1 - 5, keeping the folders in a clear order when viewed normally in the file system. Each of these five folders will contain 2 sub-folders, these sub-folders will also be named ”folder”, but suffixed with the number of the parent folder and the number order of that folder within it's parent folder, for example, the first sub-folder in folder1 would be named folder1-1 and the second sub-folder being folder1-2. Each sub-folder will also have two sub-folders and keep the same naming conventions, this continues down through until the depth of the folders reaches four folders deep, with the final folder for the top branch being titled folder1-1-1-1 (see fig 1).

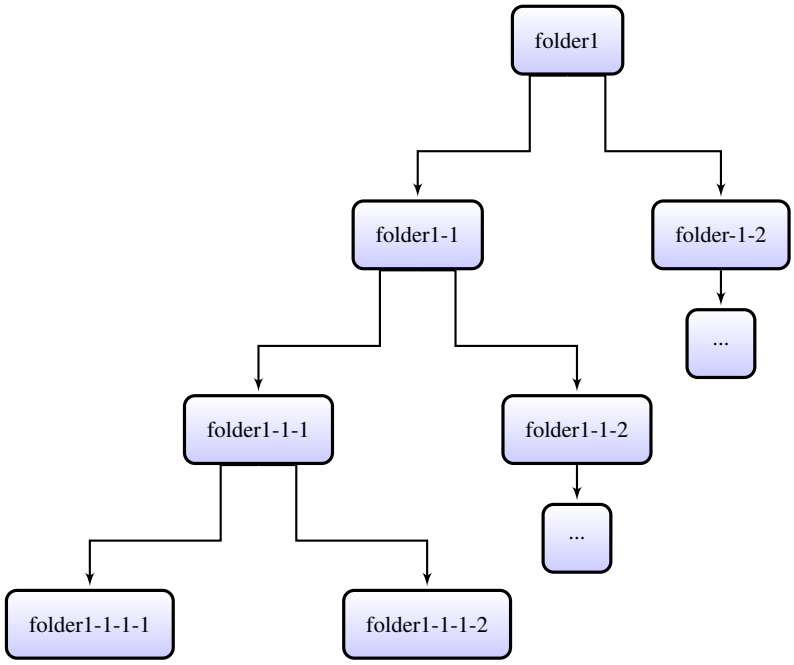

Fig. 1: Example of folder structure an naming conventions

In each of the folders there will be 8 files of different types, these being .doc, .docx, .txt, .csv, .pdf, .ppt, .xmls, .pub. These files have a mixed order in each of the folders and sub-folders, this has been designed this way to see if any behavioral patterns occur depending on file type, and can help identify if ransomware ignore

certain file types. There is also a change in file size, once again this is to see if there are differences in behavior when ransomware encrypt files of different sizes.

This virtual environment will have a snapshot (a backup of the system that can easily be restored) of the set file set-up and an installation of the software before ransomware infection. This is to make rerunning the analysis of ransomware activity easier. Using a virtual environment could potentially impede the study. This is because some ransomware families have inbuilt protections to identify if they are running in virtual environments. This may cause some ransomware families or variants to not be suitable for use with the presented methodology. This is a limitation that would need further research to overcome.

3.3 Ransomware

The ransomware executables were collected from VirusShare, an online repository for malware, shared by the anti-virus community. From this repository, ransomware samples will be collected, using some of the wider known ransomware, such as petya, locky or cryptowall. Ransomware like petya, and its variants could be more difficult to collect data for, as they more commonly known to make changes to the master boot record to stop users from accessing the machine. Some ransomware may not run on the virtual environment due to the issues discussed above, and others may be unable to connect to the C&C servers either due to location, firewall or server downtime.

4 Results

In this section, results are presented from executing five common ransomware types alongside the presented methodology for monitoring file system interaction.

Name	Traversal	File Order	Folder Order
GandCrab	Depth-first	Alphabetical order	Alphabetical order
TelsaCrypt	Depth-first	Alphabetical order	Alphabetical order
CryptXXX	Depth-first	Alphabetical order	Reverse alphabetical order
Osiris	Random order	Prioritised by extension, then alphabetical	Random
Sage2.2	Random	Single file iterations	Random

Table 1: Displaying the folder and file traversal methods

4.1 Traversal Methods

Gandcrab and TeslaCrypt showed very similar file system traversal methods (see Table 1), both adopting depth-first traversal following a systematic approach, encrypting files in each folder in a alphabetical order, going from the first parent folder and traversing through it's sub-directories, going through the first branch in each sub folder, until no further folders can be found (see Figure 2).

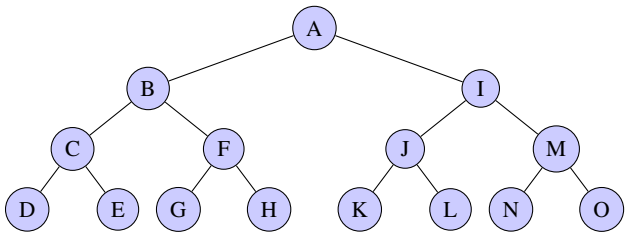

Fig. 2: File system traversal graph for GandCrab and TelsaCrypt. This graph shows the movement in a folder and it's sub-folders, moving from A to O.

CryptXXX also took a similar approach to traversing the folder structure; however doing this in a reversed alphabetical order, starting from folder 5 and it's sub-directories, first following the final branch in each sub-folder until it reaches the end (see Figure 3).

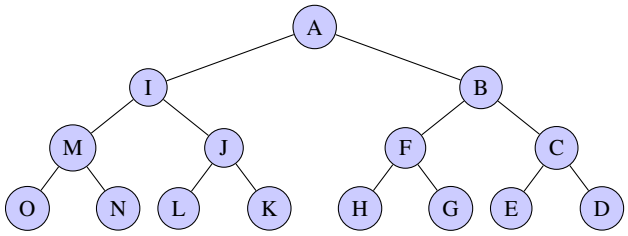

Fig. 3: Graph displaying traversal pattern for CryptXXX. The path taken by CryptXXX is displayed from A to O.

Timestamp	Folder	File	Event Type
201803141350386950	folder2/folder2-1/folder2-1-1/folder2-1-1-1	file2.ppt	Renamed
201803141350387266	folder2/folder2-1/folder2-1-1/folder2-1-1-1	OSIRIS-3925.htm	Created
201803141350389144	folder1	file1.doc	Renamed
201803141350389292	folder1	OSIRIS-89a0.htm	Created
201803141350389449	folder3/folder3-2/folder3-2-2/folder3-2-2-2	file2.ppt	Renamed
201803141350389607	folder3/folder3-2/folder3-2-2/folder3-2-2-2	OSIRIS-f4cc.htm	Created
201803141350389767	folder3/folder3-2/folder3-2-2/folder3-2-2-2	file3.xls	Renamed
201803141350389926	folder1	file5.ppt	Renamed
201803141350390229	folder1	file6.xls	Renamed
201803141350390388	folder5/folder5-1/folder5-1-1	file7.doc	Renamed
201803141350390703	folder5/folder5-1/folder5-1-1	OSIRIS-db55.htm	Created
201803141350391328	folder3/folder3-2/folder3-2-2/folder3-2-2-2	file6.doc	Renamed
201803141350391481	folder1/folder1-1	file4.ppt	Renamed
201803141350391639	folder1/folder1-1	OSIRIS-f3db.htm	Created
201803141350391798	folder1/folder1-1	file5.xls	Renamed
201803141350391950	folder5/folder5-2/folder5-2-1/folder5-2-1-1	file6.doc	Renamed
201803141350392105	folder5-2/folder5-2-1/folder5-2-1-1	OSIRIS-c0bd.htm	Created
201803141350392423	folder1/folder1-1	file8.doc	Renamed
201803141350392734	folder5/folder5-1/folder5-1-1/folder5-1-1-1	file2.ppt	Renamed
201803141350392891	folder5/folder5-1/folder5-1-1/folder5-1-1-1	OSIRIS-8f80.htm	Created

Table 2: Displaying the first twenty lines of the recorded log of Osiris activity in the file system.

Sage2.2 and Osiris also display similar behaviour to each other, but share no common behaviour with the ransomware discussed above. Both seemly approach each folder and subfolder in a random order, with no clear patterns on how it moves between folders in the directory. It was considered that this may be resulting from the use of multiple threads, each handling a portion of the directory structure. However, even after considering this possibility, it still was not possible to identify a systematic pattern. Osiris, is the only sample that iterates through the file system 3 times, each time encrypting different file types. As shown in Table 2, Osiris starts its file encryption in folder2-1-1-1 and moves onto folder1 and then onto folder3-2-2-2. This random order continues until the third iteration through the directories, where it starts to show a more structured traversal technique, using depth-first traversal.

4.2 Files and extensions

A majority of the ransomware samples used follow a alphabetical ordering to the files in a directory, always encrypting the first file it identifies in the file structure and moves systematically through the directory until no more files are available for processing. Gandcrab, TeslaCrypt and CryptXXX follow this pattern; however, it was noticed that TeslaCrypt did not encrypt .pub files.

Osiris encrypts files in the same alphabetical order as the three ransomware instances mentioned above; however it encrypts in three different stages, prioritising certain file extension on each run through the file system. In the first encryption stage, it encrypts all files with extensions of .doc, .ppt and .xls, these file are encrypted in an alphabetical order in each folder. The second stage of encryption, Osiris encrypts all .docx files, and then for the third stage of the encryption process, all .txt, .csv and .pdf files were encrypted.

Timestamp	Folder	File	Event Type
201804020637243459	folder2/folder2-1/folder2-1-2	file7.doc...	Created
201804020637243459	folder5/folder5-2/folder5-2-2	file7.doc...	Created
201804020637243459	folder2/folder2-1/folder2-1-1/folder2-1-1-2	file6.doc...	Created
201804020637243616	folder2	file1.doc...	Created
201804020637243616	folder2/folder2-1/folder2-1-1	file7.doc...	Created
201804020637243616	folder5/folder5-2/folder5-2-2/folder5-2-2-1	file6.doc...	Created
201804020637243616	folder2/folder2-1	file8.doc...	Created
201804020637243616	folder2/folder2-2/folder2-2-1/folder2-2-1-1	file6.doc...	Created
201804020637243616	folder5/folder5-2/folder5-2-1/folder5-2-1-1	file6.doc...	Created
201804020637243616	folder2/folder2-2/folder2-2-2	file7.doc...	Created
201804020637243616	folder2/folder2-1/folder2-1-2	file7.doc...	Renamed
201804020637243616	folder2/folder2-2/folder2-2-1/folder2-2-1-2	file6.doc...	Created
201804020637243616	folder5/folder5-2/folder5-2-1	file7.doc...	Created
201804020637243616	folder2/folder2-1/folder2-1-2	file7.doc	Deleted
201804020637243616	folder2/folder2-1/folder2-1-2	!HELP_SOS.hta	Created
201804020637243616	folder2	file1.doc...	Renamed
201804020637243616	folder2	file1.doc	Deleted
201804020637243616	folder2	!HELP_SOS.hta	Created
201804020637243616	folder2/folder2-1/folder2-1-1	file7.doc...	Renamed
201804020637243616	folder2/folder2-1/folder2-1-1	file7.doc,,,	Deleted

Table 3: Displaying the first twenty lines of the recorded log of Sage2.2 activity in the file system.

Sage2.2 encrypts files by iterating through the file system for each individual file type, starting with .doc files and ending with .pdf file. Due to this behaviour and the single use of each file type in the monitor file structure, no other behavioural patterns, such as alphabetical ordering, could be discerned. In addition, due to the limited number of file extensions, this ransomware could iterate through many other file extension types that were not recorded. Sage2.2 did not encrypt .pub files.

4.3 Ransom Notes

Each of the ransomware sample run created ransom notes in each directory it affected, with each sample using different behaviour to create these notes. A majority

of the samples created one file, with TeslaCrypt being the only one to create 3 different files, these being .png, .txt and .htm. As shown in Figure 4, TeslaCrypt creates these files once it reaches the end of a folder path, meaning that the creation method must run after exploration of a folders direct sub-folders.

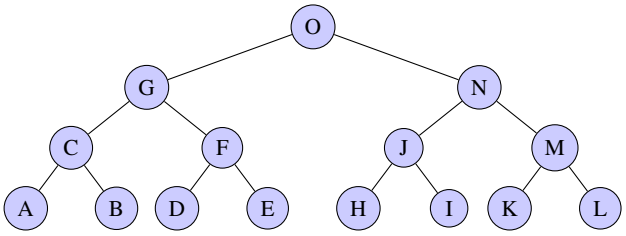

Fig. 4: TelsaCrypt's creation of ransom note files in each directory. This graph displays the movement pattern for the creation of these notes, from A to O.

CryptXXX and Osiris create ransom notes after the successful encryption of a file in a directory. They encrypt the first file, chosen in alphabetical order, create the ransom note, and then continue through the directory. Sage2.2 also displays this behaviour, creating the ransom note after deletion of the original file. This behaviour is repeated through the first iteration of the file system, but does not continue after, as all the files have been created. GandCrab created it's ransom notes upon entering a directory, before any encryption in the folder had been done.

5 Conclusion

In this section, a conclusion is drawn based on the results of the previous section, discussing the main findings of this experiment and possible future research to expand on this work.

Each of the investigated ransomware samples displayed unique behavioural traits in regard to file system activity. They either had their own behaviour pattern, showing behaviour that only itself possesses, or a number of behavioural patterns that it shared with other ransomware samples. The samples investigated in this research identified that two have common behavioural features (GandCrab and TelsaCrypt) and the other three have unique characteristics. These behaviour patterns show that ransomware can be seen working in a file system clearly and, importantly, show a distinctive difference from normal user file interactions.

This work has facilitated further research in this topic, showing that there are specific behavioural patterns that can be observed in ransomware file system interaction and shows that ransomware can be identified using individual or shared patterns. This data could also be used to create systems that could potentially halt ransomware processes in progress should they activate on a system.

References

1. Moir, R.: Defining malware: FAQ (2003). https://technet.microsoft.com/en-us/library/dd632948.aspx
2. 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 (2015). Springer
3. Richardson, R., North, M.: Ransomware: Evolution, mitigation and prevention. International Management Review **13**(1), 10 (2017)
4. Brenner, B.: InfoSec 2017: a look at the family album of ransomware (2017). https://nakedsecurity.sophos.com/2017/06/06/infosec-2017-a-look-at-the-family-album-of-ransomware/.
5. Beek, C.: McAfee Labs 2017 Threats Predictions (2017). www.mcafee.com/uk/resources/reports/rp-threats-predictions-2017.pdf
6. MalwareBytes: Cybercrime Tactics and Techniques (2017). https://www.malwarebytes.com/pdf/labs/Cybercrime-Tactics-and-Techniques-Q1-2017.pdf
7. Symantec: Internet Security Threat Report (2017). https://www.symantec.com/content/dam/symantec/docs/reports/istr-22-2017-en.pdf
8. FBI IC3: Internet Crime Report (2016). https://pdf.ic3.gov/2016_IC3Report.pdf
9. US Department of Justice: How to protect your networks from ransomware. Technical report (2016). https://www.justice.gov/criminal-ccips/file/872771/download
10. Savage, K., Coogan, P., Lau, H.: The evolution of ransomware. Symantec, Mountain View (2015)
11. Upadhyaya, R., Jain, A.: Cyber ethics and cyber crime: A deep dwelved study into legality, ransomware, underground web and bitcoin wallet. In: Computing, Communication and Automation (ICCCA), 2016 International Conference On, pp. 143–148 (2016). IEEE
12. Fischer, T.: Private and public key cryptography and ransomware. Technical report (2014)
13. Trend Micro: Command-and-control (C&C) server (2017). https://www.trendmicro.com/vinfo/us/security/definition/command-and-control-(c-c)-serve
14. Sophos: Ransomware: How an attack works (2016). https://community.sophos.com/kb/en-us/124699
15. Beek, C., Frosst, D., Greve, P., Gund, Y., Moreno, F., Peterson, E., Schmugar, C., Simon, R., Sommer, D., Sun, B., et al.: Mcafee labs threats report [internet]. McAfee Lab (April 2017). https://www. mcafee. com/us/resources/reports/rp-quarterly-threats-mar-2017. pdf, 49 (2017)
16. Symantec: ISTR Ransomware 2017. https://www.symantec.com/content/dam/symantec/docs/security-center/white-papers/istr-ransomware-2017-en.pdf
17. Liao, K., Zhao, Z., Doupé, A., Ahn, G.-J.: Behind closed doors: measurement and analysis of cryptolocker ransoms in bitcoin. In: Electronic Crime Research (eCrime), 2016 APWG Symposium On, pp. 1–13 (2016). IEEE
18. Panda Security: CryptoLocker: What Is and How to Avoid it. Panda Security (2015). https://www.pandasecurity.com/mediacenter/malware/cryptolocker/
19. McGoogan, C., Titcomb, J., Krol, C.: What is WannaCry and how does ransomware work? (2017). http://www.telegraph.co.uk/technology/0/ransomware-does-work/
20. Symantec Threat Intelligence: What you need to know about the WannaCry Ransomware (2017). https://www.symantec.com/blogs/threat-intelligence/wannacry-ransomware-attack
21. Joven, R., Yick Low, C.: MacRansom: Offered as Ransomware as a Servive (2017). https://blog.fortinet.com/2017/06/09/macransom-offered-as-ransomware-as-a-service

22. Barkly: Ransomware-as-a-Service is Booming (2017). https://blog.barkly.com/how-ransomware-as-a-service-works
23. Conner, B.: Ransomware-As-A-Service: The Next Great Cyber Threat? (2017). https://www.forbes.com/sites/forbestechcouncil/2017/03/17/ransomware-as-a-service-the-next-great-cyber-threat/#648c45d34123
24. Europol: No More Ransom: law enforcement and IT security companies join forces to fight ransomware (2016). https://www.europol.europa.eu/newsroom/news/no-more-ransom-law-enforcement-and-it-security-companies-join-forces-to-fight-ransomware
25. No More Ransom: About the Project (2016). https://www.nomoreransom.org/en/about-the-project.html
26. Osbourne, C.: No More Ransom project helps thousands of ransomware victims (2017). http://www.zdnet.com/article/no-more-ransom-project-unlocks-over-28000-devices/
27. KasperSky: No More Ransom: A very productive year (2017). https://www.kaspersky.com/blog/no-more-ransom-first-anniversary/17791/
28. Cloonan, J.: Advanced Malware Detection - Signatures vs. Behavior Analysis (2017). https://www.infosecurity-magazine.com/opinions/malware-detection-signatures/
29. Nieuwenhuizen, D.: A behavioural-based approach to ransomware detection (2017). https://labs.mwrinfosecurity.com/assets/resourceFiles/mwri-behavioural-ransomware-detection-2017-04-5.pdf
30. Ask, K.: Automatic malware signature generation. 2006-10-16]. http://citeseerx. ist. psu. edu/viewdoc/download (2006)
31. Hanel, A.: An Intro to Creating Anti-Virus Signatures (2011). http://hooked-on-mnemonics.blogspot.co.uk/2011/01/intro-to-creating-anti-virus-signatures.html
32. Shosha, A.F., Liu, C.-C., Gladyshev, P., Matten, M.: Evasion-resistant malware signature based on profiling kernel data structure objects. In: Risk and Security of Internet and Systems (CRiSIS), 2012 7th International Conference On, pp. 1–8 (2012). IEEE
33. Kaspersky: Heuristic analysis in Kaspersky Anti-Virus 2012 (2012). https://support.kaspersky.co.uk/6668
34. Ahmadi, M., Sami, A., Rahimi, H., Yadegari, B.: Malware detection by behavioural sequential patterns. Computer Fraud & Security **2013**(8), 11–19 (2013)
35. Naval, S., Laxmi, V., Gaur, M.S., Raja, S., Rajarajan, M., Conti, M.: Environment–reactive malware behavior: Detection and categorization. In: Data Privacy Management, Autonomous Spontaneous Security, and Security Assurance, pp. 167–182. Springer, ??? (2015)
36. Gazet, A.: Comparative analysis of various ransomware virii. Journal in computer virology **6**(1), 77–90 (2010)
37. 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 (2016). IEEE
38. Sorokin, I.: Comparing files using structural entropy. Journal in computer virology **7**(4), 259–265 (2011)