Ransomware Detection Using Behavioural-Based Analysis

Ahmed Alyammahi, Yaser Khamayseh
College of Technological Innovation, Zayed University, UAE
M80008667@zu.ac.ae, Yaser.khamayseh@zu.ac.ae

Abstract

One of the main security issues today is ransomware attack which affects people, institutions as well as governments. Ransomware has now moved way past the traditional signature-based antivirus software that was used by hackers. Hence, more sophisticated detecting techniques for fighting this menace should be developed as it continuously becomes threatening. One of those methods, behavioural-based analysis, has been quite effective at discovering ransomware attacks. Therefore, behavioral-based analysis entails observing the actions of software or systems in order to determine if it is unusual for a possible ransomware infection. It is based on the premise that ransomware has particular actions that can be discerned by investigating its operations within the system. Such actions include encrypting files, network activities, and setting of systems. The study highlights the characteristics of ransomware attacks and the vulnerability of conventional signature-based anti-virus software to this threat. Additionally, we provide a summary of current literature concerning ransomware identification through behavior-based analysis and discuss its strengths and weaknesses. Lastly, we provide conclusions and suggestions for further research. Briefly, the case of behavioral-based analysis is very encouraging while looking at ways to discover ransomware intrusions. However, by examining the software and systems' conduct it becomes easier to identify ransomware attacks not detected through signatures-based anti-virus tools. Although this is an imperfect method with false positives and evasions used by attackers, the value of behavioral-based analysis would call for further explorations.

Introduction

The growth of ransomware attacks has been a major global problem as many people and corporations worldwide suffer huge losses due to this. The other type is ransomware, which encrypts the victim's files and demands a ransom in exchange for the decryption key (Kharraz, 2016). A well-executed ransomware attack would result in significant damages such as lost or leaked confidential information that would cost in terms of monetary compensation or a tarnished reputation. It becomes necessary to create more advanced detection means since many traditional signature-based antivirus software does not detect and prevent ransomware attacks. This includes behavioural-based analysis which has proven successful at detecting ransomware attacks. The second approach is about observing the behavior of software and systems for an unusual activity, which might point out an attack by ransomware. Behavioral-based analysis involves the analysis of the interactions between ransomware and the system (Graham, 2017).

This paper generally outlines the concept behind ransomware attacks as well as explains the weaknesses of traditional signatures-based antivirus. After that, we shall explain what behavioural-based analysis is all about and how it can help in detecting ransomware attacks. This article will look at some of the published papers regarding behavioral-based analysis for ransomware detection, focusing on the strengths and weaknesses of applying this method. Lastly, we will make a summary of the findings and recommendations for subsequent studies.

However, it is significant to explore the economic reasons for ransomware to work prior to covering countermeasures. The new era of ransomware is proving to be a lucrative venture for assailants as it requires little input yet offers a decent return, a trait that illegal and legitimate entities rarely possess. For instance, more than 97% of businesses in America pay up when asked to. Nonetheless, it is quite tricky to trace them as they usually conduct their activities anonymously and most of their income becomes lost or unaccounted for in tax revenues. However, it is easier to create ransomware leading to its wide dissemination. From this perspective, there are three key elements that make more and more ransomware attacks occur, presented below.

Another case is a rampaging ransomware attack that hit more than 150 countries. The modus operandi of the attack was simple: a virus infects a computer's disk and encrypts important data until the user pays a ransom for its decryption. This policy might not hold when it comes to cryptocurrencies. For example, in Bitcoin, a transaction is just a text linked to a user's wallet's private key. Every transaction is public but verifiable without any ID and includes a digital signature that cannot be altered. On the other hand, cryptocurrencies offer quick and anonymous financial transfers across the world, being untraceable by banks – a feature highly appreciated by the organizers of ransomware attacks. Recently, it was believed that four years back, cryptocurrency was questioned and its safety with ransom payments was doubted. Even though Cryptocurrencies have emerged and are accepted by the public, it has also changed negative public opinions towards gold (Cabaj, 2015). Consequently, ransomware perpetrators now prefer the use of cryptocurrencies for ransom payments.

In fact, new ransomware families are released almost on a daily basis, part of which is related to the accessibility of constructing such type of malware. Encryption schemes via RSA/AES are easily programmed in common cryptographic libraries (4). Furthermore, others have revealed that simple ransomware can also work effectively for extortion, especially for the ones using weak encryptions. Moreover, today, any ordinary attacker can create tailored malware thanks to the recent increase in Ransomware-as-a-Service (RaaS). RaaS sells ready-to-use ransomware kits and ensures that once paid, all the money goes directly to its developers. In addition, open-source ransomware projects such as EDW and Invisible Tears have contributed greatly towards this growth. These projects were developed initially as learning tools but later became some of the most commonly used ransomware sample templates (Ami, 2018).

Attacks utilizing ransomware are among the most perilous and appealing dangers to current cyber security. Anti-virus software is often inadequate against ransomware as well as zero-day malware attacks, which could cause huge organization infections and critical information misfortune. Also, these assaults are getting more unique and ready to modify their marks, which is causing a weapons contest.

A behavioral detection system for ransomware must take into account a number of factors and be designed in a certain way. This method was developed in response to the shortcomings of conventional signature-based antivirus software in terms of its ability to identify ransomware outbreaks. The behavioral-based analysis presents a viable solution to these drawbacks. This method keeps an eye on how software and systems behave in order to spot unusual activity that could point to a ransomware assault.

Attacks utilizing ransomware have grown to be a serious security risk, harming people, businesses, and even governments. Antivirus software that relies on signatures is no longer enough to identify and stop ransomware assaults. This emphasizes the requirement for more sophisticated detection techniques, including behavioural-based analysis.

Literature Review

Ransomware is a brand of malicious software used to cipher files at victims' PCs and extort payment of ransom in return of a key or program unlocking code. However, with growing ransomware attacks becoming increasingly frequent and complex, it is difficult to identify them. Behavior-based detection may entail an identification of abnormal behavioral patterns that suggest an occurrence of a ransomware cyber-attack. Among other things, this literature reviews studies aimed at ransomware detection with behavior-based analysis (Graham, 2017).

Researchers suggested a machine learning method for detecting ransomware through a behavioral approach in research. The machine learning was fed with the dataset of ransomware behaviors as well as

normal user behaviors. The model had a 98.9 percent rate of detection of ransomware (Burguera, 2016). In one study, they also suggested a hybrid method for detecting ransomware using behavioral analysis. The authors combined an approach that was both static and dynamic to detect ransomware-characteristic behaviors. A prototype system for monitoring and detection of ransomware achieved a detection accuracy rate of 98.5% (Cabaj, 2015).

Amin Kharraz, et al., 2016. Behavioral-based Ransomware detection framework. This led to the identification of common ransomware behavioral patterns using system call traces by the authors. With this framework, it scored 99.3% detection rate for ransomware. A behavior-based approach of using deep learning for ransomware detection was suggested by Amin Kharraz, et al., 2016. The authors built and tested two separate models, one that involved CNNs and another where CNNs were paired with LSTM. The accuracy of the proposed method was found to be 99.7 percent (Kharraz, 2016). In a similar study, Or Ami, Y. E. et al., 2018, deep learning approach for ransomware detection based on behavioural-based analysis. Through using CNNs and LSTMs, the authors were able to discover unique features in the ransomware's behaviour. A detection rate of 99.4% was attained for ransomware using the approach. In general, these studies prove the usefulness of using a behavior-based analysis technique in detecting ransomware. Using machine learning and deep learning techniques for ransomware pattern detection has proven to be effective while combining static and dynamic analyses improves precision. Nevertheless, with ransomware continuing to advance, researchers and cybersecurity professionals will need to stay on top of their game developing new methods and technologies to identify ransomware (Ami, 2018).

Recently, behavioural-oriented analysis has been in the limelight as an appropriate mechanism for detecting ransomware among researchers and cyber security experts. Rather than seeking specific signatures or known malware, behavior-based analysis searches upon the anomalous behaviors. This method can easily identify unknown signatures of newer ransomware variants. Behavior–based analysis can discover ransomware in whatever file form, even if an advanced malware variant is involved. This also helps to prevent attackers from getting around the system through simple file substitution and modification of the malware signature (Cabaj, 2015).

Some of them have suggested using behavioral-based analysis and machine learning-based strategies to detect ransomware. The machine learning techniques would be trained on a data set of ransomware behaviors and normal user behaviors in an attempt to discern features characterizing ransomware activity. Some have achieved very high detection rates for ransomware, e.g., up to 99.7%. Other studies recommend combining static and dynamic analysis together with machine learning-based approaches for ransomware detection. Static analysis is made up of examining the code of a source and the meta-data of a file, while dynamic analysis studies the process behavior of a system or a program during operations. This is achieved by amalgamating the above-mentioned approaches so as to determine known and unknown ransomware behaviors. The false positive problem of ransomware detection is based on behavioral analysis (Bhardwaj et. Al., 2016). There may be false positives that stem from legitimate users' activities or normal software such as anomalous behavior patterns due to an actual attack. Nevertheless, through enhanced analysis procedures and including additional contextual factors, scientists may significantly reduce misidentifications.

The other benefit of employing behavioral-based analysis towards ransomware detection is that it works in a real-time mode. This is crucial because ransomware attacks have the potential to inflict major harm in a brief period. By spotting the behavior characteristics of the ransomware as it occurs, cybersecurity experts are able to stop it from spreading and reduce the harm done in the process (O'Gorman and McDonald, 2012). Some recent studies that focused on deep learning-driven detection techniques, based solely on behavioural-based analysis, have been carried out as well. Machine learning includes a specific method called deep learning which provides training data to the neural network models. To mention a few, these methods are usually more effective when it comes to ransomware detection scoring around ninety-nine percent or above. A significant challenge with the use of deep learning-based techniques for ransomware detection involves the requirement of extensive datasets. Data collection is a daunting task, especially with regard to labeling the huge amount of ransomware behavioral data to train neural networks (Burguera, 2016).

Nevertheless, with the help of transfer learning and pre-trained models, it is possible to minimize the volume of training data needed. However, there are also challenges that accompany the use of behavioral-based analysis for ransomware detection such as the possibility of avoidance by attackers. Anti-analysis

methods used by an attacker or changes in the behavior pattern of ransomware can make it difficult for researchers who want to track and analyze them. Therefore, updating behavior-based analysis algorithms is essential as these help researchers in identifying any new and emerging ransomware behavior patterns (O'Gorman and McDonald, 2012).

This approach can detect ransomware attacks before they infect a victim's system and can help prevent the spread of the attack. In conclusion, behavioral-based analysis is a promising approach for ransomware detection that has shown high accuracy rates in various studies.

Methodology

Performing ransomware detection requires a decision-making algorithm used to understand the behavior of the operating system and make a simple binary decision: yes, whether it is safe/good or not, it is Ransomware. Due to the general nature of algorithmic decision-making, there are many ways to solve complex problems. We have found that machine-learning techniques work best for ransomware detection. The main purpose of machine learning is to be able to predict target text for unknown objects and is the process of optimizing the prediction model. Most describe the art from the product to the purpose label (Scaife et. al., 2016). Model fitting or model training are other requirements for good modeling. When used for ransomware detection, machine learning is the process of teaching decision-making algorithms to identify certain behaviors (data inputs) of active processes better than others resulting from malicious, non-malicious use of ransomware. Training decision-making algorithms requires three main elements: training data containing threat examples such as flagged ransomware and identified malicious applications; capturing good behavior; and a framework for developing training and prediction algorithms. "Applying APKs" to create a suitable system and then introducing machines and deep learning (DL) algorithms to identify malicious APKs based on results. These are the two main uses of ML for malware detection (Almomani et. al., 2021).

Proposed System

Our behavioral-based ransomware detection approach focuses on two specific Indicators of Compromise (IoCs). These IoCs are derived from observing the behavior of ransomware software on a machine. The first IoC is file modifications. Through our analysis of ransomware samples, we discovered that the ransomware adds an unknown file extension to encrypted files. Therefore, the presence of ransomware can be determined by the presence of such unknown file extensions.

The second IoC we considered is file entropy. Entropy is a measure of a file's unpredictability. Many types of ransomware encrypt system-resident files, and during this process, the file's plaintext is transformed into ciphertext with a high degree of entropy (Sharma et. al., 2021). By examining the entropy levels of files, we can identify potential ransomware activity.

Our behavioral-based ransomware detection approach includes another Indicator of Compromise (IoC) called Canary file manipulation. Canary files, also known as sparse files, are filler files that are irrelevant to the user. This comprises many harmless files, which are deliberately misplaced in random parts of the user's computer. When one of them changes, however, an alert is sent either to the user or a security system indicating that it has been infected by malware. This concept is straightforward yet efficient because watching selected canary files doesn't need a lot of resources than monitoring all the files of the file-system (Qaddoura et. al., 2021).

In isolation, any of these IoCs might raise the likelihood of a false positive. For example, depending only on entropy measurements will result in a false positive as some of the files with high entropy include the PE format. Therefore, we apply read the "magic bytes" of a file and entropy measurement together. Magic bytes constitute the first couple bytes in a file and help identify the extension of the file (Sheen and Gayathri, 2022). To do this with encrypted files, a generic file extension such as "unknown" is added. Therefore, by considering the entropy values as well as a search of an unknown file extension, an encrypted file detection is possible with less possibility of a Ransomware Attack. Our behavior-based ransomware detection approach uses more IoCs to make it accurate and also minimizes false positives (Sheen and Gayathri, 2022) (Manavi and Hamzeh, 2022).

FILE MONITORING

The implementation of the canary files IoCs and file modifications involved two phases: setup and detection. During the setup phase, canary files were placed on the real physical system disk to detect potential ransomware activity. However, the success of this implementation depended on the location of the canary files on the disk and the sequence in which the ransomware accessed files. By the time a canary file was examined and abnormalities were detected, there was a high likelihood that a significant portion of files would already be encrypted (Qaddoura et. al., 2021).

Two fictitious network drives were made, one before and one after the letter allocated to the real drives, so order to lessen the likelihood that the ransomware would encrypt the user's actual disk. After the fictitious network drives were created, canary files were added to them. More sophisticated ransomware, like WannaCry and Cerber, was seen to use canary file detection algorithms. To learn more about the variables affecting this identification, including file size or matching hash signatures, more investigation is required (Sheen and Gayathri, 2022). The canary files were created larger than 1KB using the user's original files in order to evade ransomware detection. Only these files were present on the fictitious disks. This implementation approach reduced the resource-intensive nature of continuously monitoring canary files installed within the computer device. Additionally, in the event of detecting ransomware, only the encrypted canary files would be affected, not personal files.

During the detection phase, two events were generated and recorded. The first event was a timer event that periodically tracked the number of files renames occurring within each five-second interval. The rationale behind this event was that ransomware typically renames files repeatedly during the encryption process. The five-second interval was chosen to minimize the chances of false detection, as legitimate programs, including games, often change file names as well. The second event utilized the Filesystem Watcher filter provided by Windows, which monitored file activities. The filter specifically watched over the two fake network drives. If a file rename was detected on those drives, a counter was incremented to alert the user of the file change. Three file modifications within the designated time frame resulted in the disabling phase (Manavi and Hamzeh, 2022).

The methods used for deactivating were modular and interoperable with the other two scripts. As part of the disabling procedure, files and folders' Access Control Levels were changed using Microsoft's Cacls command-line tool. These tactics allowed the detection system to monitor canary files and file alterations efficiently, allowing it to spot possible ransomware activity and take the necessary steps to deactivate it.

FILE ENTROPY

In our implementation, we incorporated the use of file entropy as a measure of unpredictability. We relied on the Shannon Entropy algorithm, which is widely recognized as an effective and accurate method for calculating entropy levels. This algorithm assigns a value between 1 and 8 to each file, with 1 representing low entropy and 8 representing high entropy. During our tests, we evaluated a set of files that included ciphertexts from various encryption systems, cryptographic hash outputs, and common system files. The objective was to evaluate how well the Shannon Entropy technique worked for figuring out the entropy levels in various file formats. The results of our experiments are displayed in Figures 1 and 2. These graphics show the differences in entropy levels among the files in the collection. Files that have a high entropy content, such as encrypted data, are more unpredictable. However, the entropy levels of files with text and alphanumeric data are often lower (Abdullah et. al., 2020) (Alsoghyer and Almomani, 2020). By leveraging the Shannon Entropy algorithm, we were able to effectively measure file entropy and utilize it as a valuable indicator in our behavior-based ransomware detection system. The ability to differentiate between files with high entropy (potentially indicating ransomware activity) and files with lower entropy levels provided valuable insights for identifying and detecting potential ransomware attacks (Abdullah et. al., 2020).

Cryptographic operation	Entropy level
RSA 4096	6.01
AES 256	5.99
DES	5.92
SHA-1	3.76
MD5	3.59
XOR	3.15

Figure 1. Entropy level of ciphertexts and hashes

File type	Entropy level
dll	6.075
zip	7.913
exe	4.787
ps1	4.705
txt	1.837

Figure 2. Entropy level of other files

In our analysis, we observed that based on Figure 1, a suitable threshold for detecting encryption files using entropy was determined to be 5.5. However, Figure 2 revealed that certain file types, such as zip and dll files, could be misclassified as encrypted files due to their structure and high entropy levels. To address this challenge, we incorporated the examination of magic bytes in addition to entropy analysis. By considering both entropy and magic bytes, we were able to differentiate between valid files and encrypted files. A valid file, such as a dll or zip file, would exhibit high entropy with readable magic bytes. On the other hand, an encrypted file would display high entropy with encrypted magic bytes. To analyze the magic bytes, we utilized Microsoft SigCheck software or similar tools. Similar to the file monitoring approach, a timer and counter were employed to track suspicious files that required further action. If a file operation resulted in a high entropy value and an unreadable extension, indicating suspicious behavior, the file was tagged as a suspicious file. The blocking procedure would then be initiated when two or more suspicious files were identified within a short period (Alzahrani and Alghazzawi, 2019) (Sharma et. al., 2021). By incorporating entropy analysis, examination of magic bytes, and a threshold-based approach, we enhanced the accuracy of our behavior-based ransomware detection system. This multi-faceted approach allowed us to identify potentially encrypted files while minimizing false positives and effectively triggering the blocking procedure when suspicious behavior was detected.

Performance Evaluation

The evaluation had two main stages. Initially, we evaluated the efficiency of our algorithms for the detection of ransomware on some particular known strains. We then evaluated the resource use of every approach in a certain system. We used virtual machines running the Virtual Box to come up with a sandbox environment

for conducting these assessments. Isolation was ensured by disconnecting all networks and internet channels that used the testing operating system which was Windows 8.1. The strategies were implemented using PowerShell scripts, thus giving a peek into Windows system internals. In determining ransomware detection performance, the noteworthy ransomware samples WannaCry is used. We defined it as measuring the extent to which we could detect any ransomware behavior, and stop, or prevent such an attack.

WANNACRY

Initially, there was a test known as the notorious WannaCry ransomware. It seemed like all canary files located within the corrupt drives had been encrypted, but later, the file monitoring mechanism imposed the denial of access for files and folders. In the case of WannaCry, however, this was not the case because it was too much time and it had already disabled at least a few real files before its effectiveness was felt. The file entropy implementation gave satisfactory results because the setup files of WannaCry had very high entropy values. This kicked off our script right away so that we managed to cut the process before any of the files could be encrypted. It can be inferred that with WannaCry set-up files being required to start encryption, ACL Authorizations of the user by way of ACL identity were immediately taken out. Therefore, WannaCry did not encrypt any file because it had no rights to files located in the folders unauthorized to the operations manager. To date, we have been able to stop WannaCry five times using the ACL Authentication script during five independent tests (Lachtar et. al., 2019).

Future Work and Conclusion

Ransomware poses a constantly evolving danger due to hackers' constant creation of new and inventive strains. The rate of evolution of ransomware is so fast that signature-based detection cannot keep up with it. In this study, we showed that behavior-based approaches could be the way of the future. After applying them, we tested a number of simple detection methods against popular ransomware strains. More success will come from a ransomware detection system that has the predictive ability to infer plausible threats from unknown processes. This may be accomplished by treating all executables that are presently in use as unknowns and updating the threat level in real-time in response to the executable's actions. RansomFlare uses a technique similar to this, fusing machine learning and dynamic (behavior) analysis to provide predictive capabilities that can identify zero-day ransomware. Still, there are many unsolved issues that will need to be resolved in the future, such as how to handle sophisticated strains like Petya. Is it feasible to monitor real disks instead of fictitious drives without putting a lot of strain on the system? What other characteristics of the ransomware may be used to identify it, etc. Furthermore, despite the positive results of our early testing, a complete evaluation of the implementations across a variety of samples is still necessary.

For future work, we aim to develop new machine-learning techniques for ransomware detection. Further studies should focus on the creation of new machine-learning methods capable of identifying ransomware activity by means of behavioral-based analysis. For instance, deep learning approaches, including CNNs and RNNs, could help recognize ransomware’s temporal and spatial patterns.

Additionally, we consider the integration with other security technologies. Moreover, behavioral-based analysis can be used in conjunction with other security measures like IDPS or NSM to develop an all-enveloping anti-ransomware strategy. In the future, emphasis can be put on designing integrated systems using multitudes of security technologies for effective ransomware detection and remedy.

Finally, a real-world deployment and evaluation is needed. Behavioral-based analysis seems promising for ransomware detection; however, it would be prudent to study ways of employing and evaluating such models in practical environments. The next step lies in undertaking assessment studies to establish how well these configurations work in large-scale enterprise networks under varying sectors.

References

Amin Kharraz, S. A. (2016). A large-scale, automated approach to detecting ransomware. pp, 10-11.

Graham., C. (2017). Nhs cyber attack: Everything you need to know about ’biggest ransomware’ offensive in history. pp, 22-25.

Iker Burguera, U. Z.-T. (2016). behavior-based malware detection system for android. In Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices. pp, 3-11.

Krzysztof Cabaj, P. G. (2015). Network activity analysis of cryptowall ransomware. Przeglad Elektrotechniczny,. pp, 9-16.

Ami, Y. E. (2018). Ransomware prevention using application authentication-based file access control. pp, 5-9.

Akashdeep Bhardwaj, Dr. GVB Subrahmanyam, Dr. Vinay Avasthi, Dr. Hanumat Sastry (2016) Ransomware digital extortion: a rising new age threat, Indian Journal of Science and Technology, 9,14, 1-5

Gavin O'Gorman and Geoff McDonald (2012) Ransomware: A growing menace, Symantec Corporation

N. Scaife, H. Carter, P. Traynor, K.R. Butler, Cryptolock (and drop it): stopping ransomware attacks on user data, in 2016 IEEE 36th International Conference on Distributed Computing Systems (ICDCS) (IEEE, New York, 2016, June), pp. 303–312

Almomani, I., Qaddoura, R., Habib, M., Alsoghyer, S., Al Khayer, A., Aljarah, I., & Faris, H. (2021). Android ransomware detection based on a hybrid evolutionary approach in the context of highly imbalanced data. IEEE Access, 9, 57674–57691.

Alzahrani, N., & Alghazzawi, D. (2019, November). A review on android ransomware detection using deep learning techniques. In Proceedings of the 11th International Conference on Management of Digital EcoSystems (pp. 330–335).

Sharma, S., Krishna, C. R., & Kumar, R. (2021). RansomDroid: Forensic analysis and detection of Android Ransomware using unsupervised machine learning technique. Forensic Science International: Digital Investigation, 37, 301168.

Qaddoura, R., Aljarah, I., Faris, H., & Almomani, I. (2021). A classification approach based on evolutionary clustering and its application for ransomware detection. In Evolutionary Data Clustering: Algorithms and Applications (pp. 237–248). Springer, Singapore.

Sheen, S., & Gayathri, S. (2022). Early Detection of Android Locker Ransomware Through Foreground Activity Analysis. In Proceedings of Third International Conference on Communication, Computing and Electronics Systems (pp. 921–932). Springer, Singapore.

Manavi, F., & Hamzeh, A. (2022). A novel approach for ransomware detection based on PE header using graph embedding. Journal of Computer Virology and Hacking Techniques, 1–12.

Abdullah, Z., Muhadi, F. W., Saudi, M. M., Hamid, I. R. A., & Foozy, C. F. M. (2020, January). Android ransomware detection based on dynamic obtained features. In International Conference on Soft Computing and Data Mining (pp. 121–129). Springer, Cham.

Alsoghyer, S., & Almomani, I. (2020, March). On the effectiveness of application permissions for android ransomware detection. In 2020 6th conference on data science and machine learning applications (CDMA) (pp. 94–99). IEEE.

Nada Lachtar, Duha Ibdah, and Anys Bacha, "The Case for Native Instructions in the Detection of Mobile Ransomware", IEEE Letters of the Computer Society, vol. 2, issue 2, pp. 16–19, May 2019.