SEMARAK ILMU
PUBLISHING
202103268166(003316878-P)

Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:
https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

Early Detection of Windows Cryptographic Ransomware Based on Pre-Attack API Calls Features and Machine Learning

Wira Zanoramy A. Zakaria[1,*], Nur Mohammad Kamil Mohammad Alta[1], Mohd Faizal Abdollah[2], Othman Abdollah[2], S.M. Warusia Mohamed S.M.M Yassin[2]

[1] MyCERT, Cybersecurity Malaysia, Menara Cyber Axis, Jalan Impact, 63000 Cyberjaya, Selangor, Malaysia
[2] Fakulti Teknologi Maklumat dan Komunikasi, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia

ARTICLE INFO

Article history:
Received 9 August 2023
Received in revised form 26 December 2023
Accepted 11 January 2024
Available online 12 February 2024

ABSTRACT

Ransomware attacks are currently one of cybersecurity's greatest and most alluring threats. Antivirus software is frequently ineffective against zero-day malware and ransomware attacks; consequently, significant network infections could result in substantial data loss. Such attacks are also becoming more dynamic and capable of altering their signatures, resulting in a race to the bottom regarding weaponry. Cryptographic ransomware exploits crypto-viral extortion techniques. The malware encrypts the victim's data and demands payment in exchange. The attacker would release the data decryption key after accepting payment. After data encryption, the user has two options: pay the ransom or lose the data. Cryptographic ransomware causes damage that is nearly impossible to undo. Detection at an early stage of a ransomware attack's lifecycle is vital for preventing unintended consequences for the victim. Most ransomware detection technologies concentrate on detection during encryption and post-attack stages. Due to the absence of early behaviour signs, it is challenging to detect ransomware before it begins the unwanted process of mass file encryption. This study examines the relationship between API calls pattern and their nature to determine whether it is ransomware early behaviour. The purpose of this paper is to determine whether this technique can be used to early detect the presence of ransomware activity on a Windows endpoint. 582 ransomware samples that consist of ten ransomware families and 942 benign software samples were analysed. This study proposed RENTAKA, a novel framework for the early detection of cryptographic ransomware. It makes use of characteristics acquired from ransomware behaviour and machine learning. This study presented an algorithm to generate a ransomware pre-encryption dataset. This study, which includes six machine-learning models, gives satisfactory results in detecting cryptographic ransomware. The features used in this research were among the 232 features identified in Windows API calls. Five standard machine learning classifiers were employed in this experiment: Naive Bayes, k-nearest neighbours (kNN), Support Vector Machines (SVM), Random Forest, and J48. In our tests, SVM fared the best, with an accuracy rate of 93.8% and an area under the curve (AUC) of 0.979, respectively. The results indicate that we can distinguish ransomware from benign applications with low false-positive and false-negative rates.

Keywords:
Ransomware; Crypto-Ransomware; Early Detection; Ransomware Lifecycle

[*] *Corresponding author.*
E-mail address: wira@cybersecurity.my

https://doi.org/10.37934/araset.39.2.110131

1. Introduction

Malicious software, also known as malware, is created, and utilized by cybercriminals to disrupt computer systems, collect sensitive data, or gain unauthorized computer access. Computer viruses, worms, Trojan horses, spyware, adware, and ransomware are examples of malware. Ransomware is malicious software designed to harm a single computer, server, or computer network. It is designed to lock computers and encrypt specific files. The primary objective of ransomware is extortion by imposing a denial of service on the system or system resources, such as files, until the ransom is paid. The victim must pay a ransom to regain access to the system or files, typically delivered in cryptocurrency or bitcoin. If the victim pays the demanded ransom, he or she will regain access to the files [1-3].

Ransomware is malware that encrypts sensitive or credentialed data and demands a ransom in exchange for system control. From 2018 to 2020, ransomware attacks on industrial control systems (ICS) increased by approximately 500%. There are ransomware attacks in every region of the globe. Governments, hospitals, banks, commercial organizations, power grids, universities, and numerous individuals have been affected [4-7].

Unless a ransom is paid, ransomware prevents or restricts users from accessing their system by locking the system’s screen or the user's files [8,9]. Modern ransomware families, such as cryptographic ransomware, encrypt specific files on infected systems and compel users to pay the ransom via online payment methods to receive the decryption key. It is one of the numerous types of attacks that organizations have struggled to counter in recent years. Ransomware gains access to the servers of a business or individual and encrypts the data. Instead of recovering the data through infrastructure safeguards, hackers typically demand a ransom, usually a small amount that a company can afford to pay. Two types of ransomwares exist. The first type is locker ransomware, which locks the victim out of the operating system and prevents access to the desktop and files without encrypting them [10-13]. The second type is cryptographic ransomware, which encrypts system files and requires the user to pay a ransom to regain access to the data [3,14].

In 2021, the Colonial Pipeline experienced a ransomware attack that affected the pipeline's computerized management equipment. The company contributed 4,400,000 Bitcoin to the decryption tool. Financially motivated ransomware attackers are expected to increasingly target critical power system infrastructures such as substations, wind/solar farms, and charging infrastructures [15].

This dramatic increase, coupled with the continued evolution of ransomware strains, poses several threats to the smart grid, including the leaking and selling of confidential data, resulting in future cyberattacks, damage to field system control processes, resulting in the creation of public safety hazards, significant disruptions to the power grid operation, and economic loss [16-18].

Consequently, smart grid cybersecurity and engineering teams must prepare now for an increase in ransomware attacks with severe consequences in the coming years. Existing industry defence strategies for the power grid have emphasized tamper-resistant data backup and network-based security techniques such as network segmentation, encryption (transport layer security), moving target defence, and network intrusion detection systems to reduce the vulnerabilities of utility and industry communications standards [5,19-21].

However, encrypted ransomware files in TLS network protocols can bypass firewall security mechanisms and network IDS, and human risks are always present, posing a threat to administrative access to smart grid devices and servers. Methods for ransomware detection generally fall into two categories: static analysis and dynamic analysis. Ransomware is examined by static malware analysis without executing the actual binary files. This technique utilizes static data such as file header information, hashes, and URLs, which can be fed to open-source analysis tools such as VirusTotal [22]. Although static malware analysis methods are simple to detect and implement for known ransomware, they are mainly ineffective against highly sophisticated ransomware attacks.

Recently, machine learning-based static malware detection methods have been proposed to improve detection accuracy. As ransomware evolves, new malware must also be identified. Dynamic analysis techniques detect ransomware attacks by analysing abnormal behavioural data resulting from ransomware compilation or ransomware events perpetrated by adversaries on the target system [23-26].

Ransomware prevents users from accessing their computers by encrypting files or locking the system. The victim must pay the ransom demanded by the attacker for system restoration. The payment for the ransom must be made using a cryptocurrency transaction. The encrypted data cannot be decrypted unless the attacker provides the encryption key [27,28]. Two varieties of ransomware exist locker and cryptographic. Until the ransom is paid, Locker ransomware restricts user access to the infected system. Locker is ransomware that does not encrypt any files. During this time, cryptographic ransomware encrypts all targeted files on the compromised computer. Ransomware is packaged and transmitted into the network through various distribution mechanisms, including email attachments, drive-by downloads, compromised strategic websites, and exploiting network vulnerabilities [29-31]. Figure 1 shows the categories of ransomware.

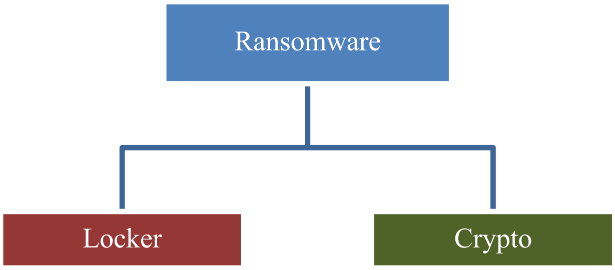

Fig. 1. Categories of ransomware

Cryptographic ransomware seeks to search for and encrypt users' files quietly and then demands a ransom in exchange for the decryption keys. Frequently, crypto ransomware does not encrypt the entire hard drive but instead searches for specific file extensions, such as.doc, .ppt, .jpg, and .pdf, often of files containing text documents, presentations, and images, which typically contain valuable and personal user data. These data would have the most significant impact on the users if lost. Asymmetric encryption and hybrid mechanisms have also been implemented, such as using a symmetric key to encrypt the files and a public key to encrypt the symmetric keys [18,32-34].

Like malware, ransomware uses multiple attack vectors (such as spam emails, malicious advertisements, and social engineering) to infect a computer [35-38]. The malware will lock the system or encrypt the victim’s data. The victim will ultimately be required to pay a ransom to unlock the system or obtain the decryption key.

The concept of ransomware is not novel. It has existed in the form of AIDS malware since 1989 [39]. The category of crimeware includes ransomware. This is a category of malicious software designed to aid in online crime and cyber-extortion [21,31,40-42]. Ransomware remains one of the deadliest cyber threats years after year. It threatens all IT users, whether they employ desktop or mobile devices. Numerous individuals, governments, and businesses were affected by ransomware attacks across the globe. It has gained in popularity in recent years. It is utilized by cybercriminals to gain notoriety and enormous profits. The effects of ransomware on manufacturing, the economy, and social trust in affected organizations is all detrimental.

The number of machines infected with ransomware grows daily. Ransomware continues to be one of the most severe cyber threats and endangers IT and users worldwide. In recent years, it has become a phenomenon and a terrifying threat to individuals, nations, and organizations. In addition to penalizing computational tasks, ransomware requires victims to pay extortionate amounts to regain access to the system and its contents. Cybercriminals generate millions of dollars in profits and continue to distribute new ransomware variants [40,43,44].

Cybercriminals have expanded their online moneymaking strategies. In addition to promoting fake services on online retail websites and luring consumers into fake mobile banking applications, cybercriminals use ransomware to extort victims' money. The emergence of e-currencies like Bitcoin contributes to the increase in ransomware attacks. Today, ransomware is the weapon of choice for cybercriminals seeking financial gain.

In recent years, ransomware has become a significant threat to IT users and worsens annually. According to Symantec, ransomware dominated the global threat landscape in 2016, as its prevalence rose by 267%. It is believed that the CryptoWall 3.0 ransomware family is the most lucrative. The global cost of the damages is estimated at USD350 million [45-48]. According to a 2016 Kaspersky analysis, ransomware attacks on individuals and businesses occurred every 40 seconds on average. Ransomware has infected individuals, universities, government agencies, financial institutions, healthcare facilities, and businesses. In May 2017, WannaCry infected approximately 400 thousand systems in 150 countries. Figure 2 shows the list of sectors affected by ransomware attacks.

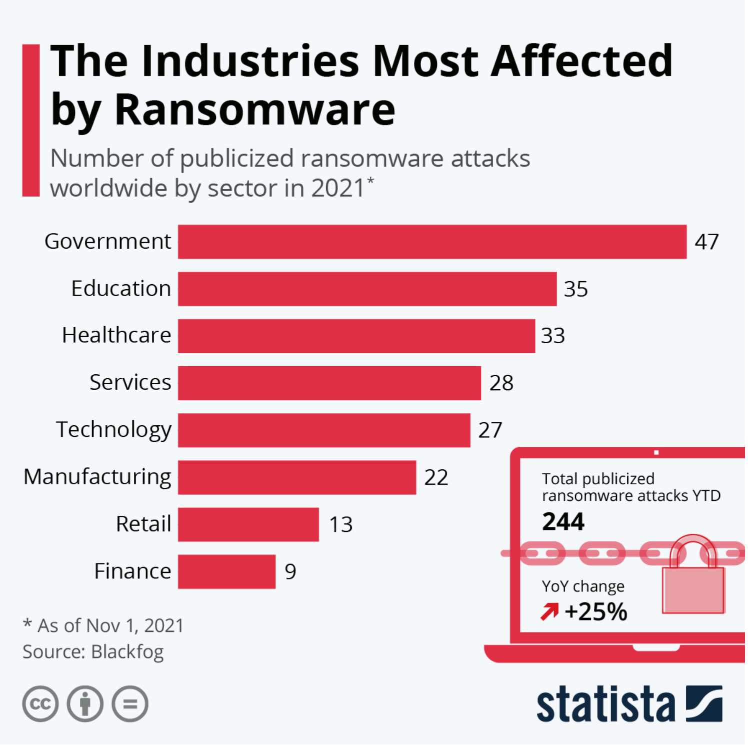

Fig. 2. List of sectors most affected by ransomware (source: statista)

The WannaCry and NotPetya attacks of 2017 are estimated to have cost the global economy more than $8 billion [49-52]. Over 50,000 systems were infected with the GandCrab ransomware during the first quarter of 2018. In addition to ransom payments, the expenses included the cessation of commercial activities, the impact on the public image of the affected organizations, and the insurance repercussions. In 2017, ransomware affected not only desktop systems but also mobile devices.

Comparing the first quarter of 2017 to the previous quarter, antivirus vendors observed a threefold increase in ransomware installation packages. Since 2018, malware developers have employed "droppers", seemingly harmless trojan software that, when activated, downloads and installs the actual malware on a computer [53]. On March 29, 2019, MyFitnessPal, a popular fitness application, notified its users of a system breach that allowed attackers to obtain data from its servers [7]. In September 2018, cybercriminals exploited a Facebook network flaw to gain access to more than 30 million accounts. Huawei reported in October 2019 that its computers and networks are subjected to approximately one million cyberattacks daily. Indeed, these attacks provide evidence of the ever-expanding cyber threat landscape, which various malicious software types, such as adware, viruses, spyware, rootkits, worms, and ransomware, may cause. Ransomware has become an increasingly destructive form of malware that typically locks or encrypts the victims' computers and forces them to pay a ransom to regain access to their files. According to Cisco, ransomware is quickly becoming the most lucrative form of malware ever and is on its way to becoming a $1 billion market annually [54]. Its explosion was caused by the rise of cryptocurrencies, which preserve user anonymity, and the development of nearly impossible-to-crack encryption techniques. Recent evidence indicates that ransomware attacks target critical and sensitive data to coerce victims into paying the ransom. For instance, on July 25, 2019, an attack struck the South African electric utility city power. The victims of a subsequent attack in Laporte County, Indiana, and Lake City, Florida, had to pay $132,000 and $462,000 to decrypt encrypted government data. Even though these notable targets had a line of defence, the attacks were deceptive and intelligent enough to avoid detection and infecting their targets. Another significant member of the malware family is ransomware. It restricts the user's access to their files until the ransom is paid. This is achieved by encrypting and locking the victim's desktop file. It is one of the greatest threats to system security. Over 200,000 computers in 150 countries were recently infected with WannaCry ransomware, resulting in billions of dollars in losses [45,55-57]. Ransomware prevents users from accessing their files until a ransom is paid. The number of hackers who use this type of malware to generate revenue has increased. RaaS is a business model that allows ransomware applications to be purchased for a fee. The price may be paid in full or in accordance with a profit-sharing agreement. This suggests that criminals are cooperating. One party can organize an infection or attack campaign, followed by the distribution of ransomware created by another party. Each side gains from an effective attack. This will eventually lead to a rise in specialized criminals who will be challenging to combat by law enforcement.

Ransomware is malicious software that infects, locks, or takes control of a user's computer and then demands a ransom to undo the damage. The detection of ransomware is crucial to the security of computer systems. However, signature-based methods have difficulty detecting zero-day attacks and polymorphic viruses. Consequently, machine learning-based detection becomes necessary.

1.1 Variations in Ransomware Assaults

There are primarily two types of ransomware attacks: locker ransomware, which prevents the victim from logging in, and crypto ransomware, which encrypts specific file types, rendering them inaccessible.

i. Locker ransomware
   Locker ransomware prevents its victim from logging into the system. In most cases, the system can be restored by restarting or operating safely. Consequently, this ransomware is less harmful and can be quickly resolved.

ii. Crypto ransomware

Crypto ransomware encrypts file types that the victim considers valuable, such as documents, spreadsheets, images, and databases. It can utilize symmetrical, asymmetrical, or hybrid encryption. The encryption process can be divided into three categories based on the steps involved: Class A, the file is encrypted but not renamed or relocated; class B, the file is encrypted and renamed but not relocated; and class C, the file is encrypted, renamed, and relocated, making tracking, and restoring the file more challenging.

1.2 Facilitators of the Ransomware Assault

Due to the actions of multiple enablers, ransomware attacks have increased in frequency and variety. These enablers arose primarily because of technological progress and changes in lifestyle.

i. Encryption Engineering

Encryption is used to ensure privacy. Due to today's reliance on the internet, massive amounts of information are transmitted electronically. On the contrary, these data are easily imperceptible. As a result, encryption technology was developed to ensure that only authorized individuals can access the data. This technology has proven to have both advantages and disadvantages. Ransomware has utilized this technology to encrypt victims' files for extortion purposes. Primarily, ransomware uses symmetrical encryption, asymmetric encryption, and hybrid encryption. Symmetrical encryption employs the same key for both encryption and decryption. Its benefit is that the encryption process can be completed quickly. However, it has the disadvantage of being less secure. Asymmetrical encryption employs a single key for encryption (the "public key") and a second key for decryption (the "private key"). The encryption process is more secure but slower. The hybrid encryption algorithm combines symmetrical and asymmetric encryption. The victim's data is initially encrypted using symmetrical encryption, followed by asymmetrical encryption of the key. This expedites the encryption process and increases security.

ii. Digital Currency

Digital currency such as Bitcoin is the primary method of ransom payment. This is primarily because such a currency allows the recipient to remain anonymous to the government. A digital currency such as Bitcoin is widely accepted. This is true, especially considering the prevalence of online stores accepting the virtual currency. Using a one-way hash function, blockchain technology is an alternative form of encryption technology. This is the primary technology used by the cyber currency payment method to validate the currency.

iii. Accessibility for Ransomware

With the availability of RaaS, it is simple to acquire ransomware codes. Unskilled individuals can also obtain free development kits, such as Torlocker, TOX, and Hidden Tear. This significantly lowers the barrier to entry for ransomware.

1.3 Ransomware Lifecycle

Before we can build solutions for early detection, it is crucial to understand the phases of a ransomware lifecycle [19,37,58]. Most literature described ransomware phases as depicted in Figure 3. Understanding ransomware behaviour at each level would permit the creation of a mitigation strategy at the desired stage.

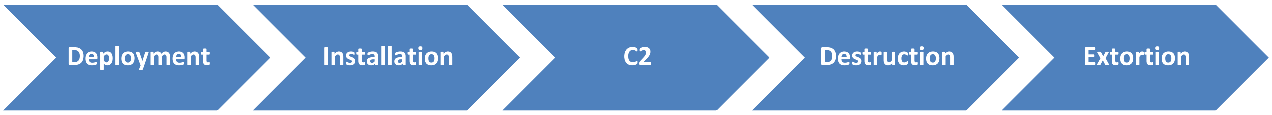

Fig. 3. Lifecycle of a ransomware attack

i. Deployment
Through email, the most popular method of transmitting ransomware, cybercriminals can execute dangerous software because they have persuaded the recipient to believe the communication is legitimate. Some of the most popular social engineering approaches are Microsoft Office files with macros, phishing, and executable files with icons. It is known that some ransomware may be downloaded from malicious websites and exploit kits like the Angler EK.

ii. Installation
The ransomware will automatically install once it has reached the host. In the case of Windows-based systems, changes are made to the system during the installation process, such as the setting of specific registry values that will guarantee the malicious malware starts up each time the host is rebooted. At this point, several valid subprocesses were formed, and a few dynamic-link libraries were run. The attacker starts to control the machine when the ransomware is installed. The malicious components may occasionally be divided into a few scripts, processes, batch files, and other tools. To prevent being discovered by antivirus scanners that rely on signatures, this is done.

iii. Command and Control (C2)
The ransomware contacts the server to obtain the encryption keys and instructions that must be followed moving forward, such as platform mapping and identification of network shared drives. There are many inconsistent ways that ransomware communicates with its controlling server. In some circumstances, communication can take place over a plain HTTP channel without encryption, or they can access the controller server using a complicated channel like the TOR network.

iv. Destruction
The specified files are now being encrypted by ransomware. The crucial element that sets ransomware apart from other malware and makes it challenging to eradicate is encryption. The encryption phase begins after successfully connecting to the victim's machine. The encryption key may occasionally be created on the victim's computer.

v. Extortion
The next step is to inform the user that the data have been fully encrypted. The ransom note that contains the instructions to be followed to send the cash and decrypt the data is presented in a window to let the victim know this. All ransomware uses extortion, although there are several ways it does it.

1.4 The Behaviours of Ransomware Installation

After ransomware has been successfully uploaded to a victim's system, the setup procedure is essential for a successful infection. One or more of the precautionary measures listed below may be utilized against ransomware.

i. Payload Persistence
This action ensures that the attack will remain persistent after a system reboot. Standard methods include placing an executable file in the start-up directory, adding a new registry key, and configuring a scheduled task.

ii. Limit System Recovery
This action is intended to prevent the victim from reverting the system to its state before infection. Deleting scheduled backups, backup systems, and backup files is standard practice.

iii. Covert Mode
This action is intended to conceal the attack from the victim. Standard techniques include executing from the %AppData% directory and utilizing the same name as the standard system executable.

iv. Environment Mapping
This action is intended to confirm that the infection is present in the victim's system and not in a sandbox. The typical setup for the dynamic analysis of malware is a sandbox. Typical methods include checking the security settings and policies, geographic location, user language, file system architecture, and network drives.

v. Communication Coverup
This action will ensure that communication with the C2 server is successful. Using an algorithm to generate a domain name randomly will complicate the authority's tracking efforts.

vi. Privilege Escalation
This action is intended to grant the attacker administrative privileges. The administrator can only execute numerous system-related tasks; therefore, elevating to the administrator level will ensure that all functions can be executed without restriction.

1.5 Ransomware Analysis Methodologies

The purpose of ransomware analysis is to understand better how ransomware operates. Based on this knowledge, defensive measures can be designed to prevent future infections. There are two types of analysis: static analysis and dynamic analysis. Static analysis is based on the executable's source code. For dynamic analysis, ransomware is executed in a controlled environment, and all of its actions are recorded for analysis.

i. Static Analysis
Static analysis can be performed rapidly by examining the features of an executable piece of code and matching them to previously observed malicious code. Analysis of the malicious code is simple and quick. Successful detection in this case also means that the ransomware can be prevented from being executed. It can be subject to code obfuscation. Regular operation code addition can result in a mismatch with previously identified malicious code. In addition, when the code is encrypted, the analysis is ineffective. There is currently no efficient brute-force method for decrypting encryptions. Simply put, it is time-consuming. Multi-phase attacks are also ineffective against static analysis. The initial code may be merely a simple process to open a backdoor for additional codes to be downloaded and, as such, does not have a similarly malicious effect.

ii. Dynamic Analysis
The behaviour-based analysis is another name for dynamic analysis. Malicious code is executed in a controlled and monitored environment, usually a sandbox. Every action is recorded for analysis. This type of analysis is less susceptible to obfuscation, and it is possible to analyse encrypted code. Malicious action must be a part of the process for its objective to be attained. Encrypted code must be decrypted before malware can execute its intended function. This type of analysis requires an expensive and time-consuming setup. To accurately capture ransomware behaviour, the simulated environment must closely resemble the real world. As previously discussed, one of the installation behaviours of ransomware is environment mapping. If the analysis is conducted on a virtual machine, which can reduce costs and conserve resources, the ransomware may detect this and prevent itself from exhibiting all its behaviours.

1.6 Ransomware Detection Techniques

This section discusses the various ransomware detection and identification techniques.

i. Machine Learning
Machine learning is the process of discovering data patterns to create a model. This model is then able to predict the outcome when given new data. Finding the appropriate algorithm to match the type of data and the desired effect can be challenging when employing machine learning. Machine learning has the advantage of accurately predicting the outcome with sufficient training data. Diverse training data should be used to predict a balanced distribution of outcomes. Because machine learning involves discovering data patterns, it is less susceptible to concealment. Finding the proper algorithm is frequently tricky and may require trial and error. In addition, bias and overfitting may occur if proper precautions are not taken.

ii. Honeypot
Honeypot is the process of creating decoy files for ransomware to attack. After accessing these files, the ransomware can be identified. Once the traps or honeypot files have been established, they await an attack. Consequently, the technique requires little system maintenance or processing power. There is no assurance that ransomware will target the honeypot files. Consequently, it is essential to understand the characteristics of the files that ransomware will target. While it is possible to deploy honeypot-style fake folders containing tripwire files for ransomware to interact with, the nature of decoy folders makes it unlikely that ransomware would attempt to infiltrate them, thereby bypassing this defense. This limited view of a system is a drawback of honeypots, as the absence of attack alerts in a honeypot does not indicate that other areas are not being targeted. As malware is automated and can randomly target any location, placing a honeypot anywhere to detect activity is preferable to not monitoring. The honeypot principle of gathering information about an attack for defense is still applicable. According to the study, honeypots can identify both the user and the number of modified files, which can inform subsequent actions.

1.7 Related Works

UNVEIL demonstrates a system that can detect ransomware by creating an artificial but realistic execution environment capable of locating file and screen lockers. File lockers can be detected by constructing and analysing semantic patterns in opened or modified files. As for screen lockers, UNVEIL identifies them by comparing a snapshot of the system before and after the execution of the suspect software.

EldeRan is a framework for dynamically analysing and classifying ransomware based on machine learning. EldeRan monitors a set of actions performed by applications during their initial installation phases, searching for ransomware characteristics and indicators. Our tests with 582 ransomware samples from 11 families and 942 benign applications indicate that EldeRan achieves a 0.995 area under the ROC curve. Additionally, EldeRan does not require the availability of an entire ransomware family in advance. These findings suggest that dynamic analysis can aid in ransomware detection, as ransomware samples exhibit a set of family-specific characteristics at runtime, which facilitates the early detection of new variants.

Scaife *et al.,* proposed CryptoDrop in 2016 as an early-warning detection system that alerts users during suspicious file activity by analysing a series of actions required for ransomware to function. By capturing the entropy of the read/write operations, the system provides a metric to detect and calculate these entropy measures. This detection system can detect variant ransomware that may not be detected by signature scanning; however, monitoring to measure entropy requires significant overhead. CryptoDrop is an early-warning detection system designed to alert users when suspicious applications perform malicious activities. Using a combination of behavioural indicators, this method identifies processes that appear to be manipulating large amounts of data. As soon as it detects ransomware, it halts the process and prevents it from completing its mission. In addition, the authors of CryptoDrop have described three crypto-ransomware behaviour scenarios based on different file-related activities. These scenarios are overwriting files, in which malware overwrites files and encrypts them in place; changing files' locations, in which malware changes files' locations and probably renames them before encrypting and dropping them back in their original places; and creating new files, the most damaging scenario, in which malware creates new versions of files with encrypted contents and deletes the original ones. Considering this, CryptoDrop has implemented three primary indicators for detecting malicious executions. One indicator identifies file modifications based on byte value modifications. Using the SDHash function, another indicator measures the similarity between versions of the same file. The final indicator measures the entropy of the encrypted file using Shannon entropy. In addition, CryptoDrop employs two secondary indicators: deletion cases, triggered when files are detected after suspicious activity, and file type funnelling, which occurs when an application simultaneously reads and writes a disparate number of files.

PEDA is a proposed framework for early ransomware detection. The framework is divided into Phase 1 and Phase 2. PEDA gathers and evaluates Windows API calls produced by ransomware samples using a learning algorithm. The proposed Learning Algorithm (LA) uses API pattern recognition to determine whether the suspect program was ransomware. This method enables the most comprehensive detection of known and unknown ransomware but may generate many false positives. PEDA created a signature database for the samples and added them to the Phase 2 signature repository if ransomware was anticipated. Despite being accurate and quick, this method only detects known malware, despite its rigidity. This technique, however, was incapable of detecting ransomware that utilized its encryption code and inherited the drawbacks of signature-based solutions.

1.8 Early Detection Challenges

New strains exhibit enhanced code and behaviour obfuscation. CTBLocker, for instance, encrypts only a portion of a file, resulting in lower writing rates than previous ransomware; a detection based on disc activity may therefore fail. GandCrab obfuscates function call parameter strings [59]. Moreover, in the early stages of the crypto ransomware infection, data and evidence were scarce. As the encryption process has not yet begun during the pre-encryption phase, it is difficult to determine if the host is infected with crypto-ransomware or if it is simply a routine of a benign program [37,60-62]. Even with a limited number of post-attack indicators, such as encryption APIs, encrypted files, file deletions, shadow copy deletions, bulk file renaming, and CreateDesktop API desktop changes, it is possible to track ransomware-like activity [31,63,64]. Detecting crypto ransomware during the pre-encryption phase is advantageous. After mass file encryption, it is useless and irreversible that data has been lost. Crypto-ransomware can be distinguished from conventional malware by its irreversible effect [58,61,65]. That is, even after the attack has been neutralized, the encrypted files cannot be accessed without the decryption key. Therefore, it is essential to detect such a threat early on, prior to the encryption process. However, the lack of sufficient information during the early phases of an attack is the greatest obstacle to early detection, resulting in low detection accuracy and a high rate of false alarms. Unfortunately, most of the previous research focused on post-attack ransomware detection. At this point, files are either completely or partially encrypted, and it is impossible to recover them. Instead of using a complete activity log file after execution, ransomware activity must be predicted during execution. This capability will allow cyber security endpoint protection to be upgraded to use behavioural data for blocking malicious activity, as opposed to detecting it post-execution and repairing the damage [66]. Referring to Figure 4, the critical substages has been labeled as Setup, File Encryption and Post-encryption.

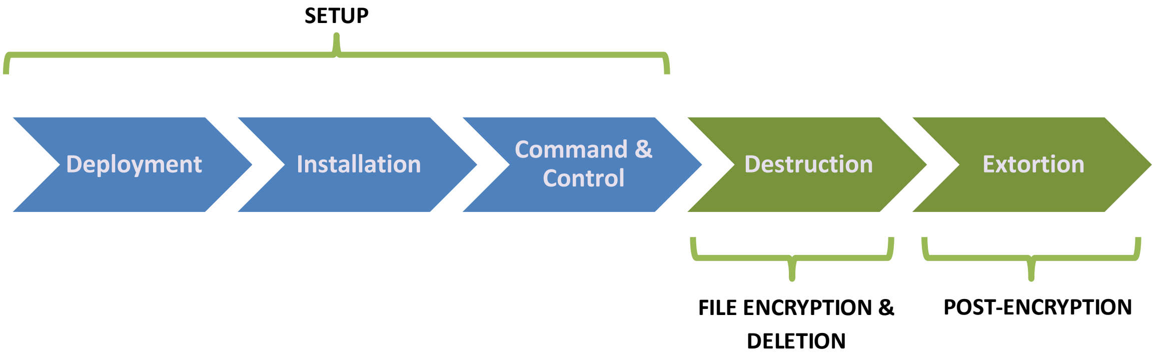

Fig. 4. Ransomware lifecycle with critical substages labelled

Based on what has been discussed so far, there are a few things that need to be done to find crypto ransomware during pre-encryption:

i. At this early stage, there is not enough proof to say for sure if this is related to a ransomware infection or not.
ii. There is no evidence that many files have been encrypted.
iii. There are no prominent and visible ransomware indicators such as weird file extensions, changed desktop wallpaper, ransom note, high CPU consumption, and system slowdown.
iv. Most research work on early detection uses API calls, which can be manipulated by the ransomware developer to defeat detection models. The detection model is based on the likelihood that API calls will be evaded.

1.9 Research Contributions

Listed below are the research contributions:

i. We proposed a framework to identify the most significant dynamic ransomware characteristics and classify ransomware using these characteristics.
ii. We proposed an algorithm to detect the encryption boundary and feature selection. This algorithm is developed to identify encryption boundary of a ransomware family.
iii. We extracted API calls that representing the pre-attack activities of ransomware.
iv. We feed the selected pre-encryption features and feed to classifiers. We tested how accurate the k-Nearest Neighbor, Naive Bayes, Support Vector Machine, and Random Forest algorithms are.

2. Methodology

This section discusses the research methodology for crypto-ransomware early detection framework. The detection framework consists of data collection, data preparation, feature selection and classification modules. Several experiments were conducted for three modules in this framework to concisely identify the most suitable technique to early detect ransomware attacks. The performance evaluation used in this research are True Positive (TP) and Accuracy (A).

Methodology is a strategy or plan of action that generates outcomes. In this chapter, each of the processes within methodology are explained to properly implement the detection framework. There are three (3) main phases in this research methodology model. The phases are:

i. Phase 1: Data Collection and Dataset Enhancement
ii. Phase 2: Framework Development and Experiment
iii. Phase 3: Testing and Validation (Refer to Figure 5).

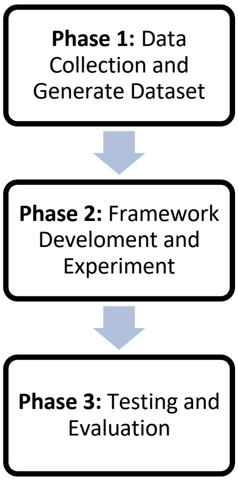
Phase 1: Data Collection and Generate Dataset
Phase 2: Framework Develoment and Experiment
Phase 3: Testing and Evaluation

Fig. 5. Research design

2.1 Proposed Framework

Having tools to identify unknown cryptographic ransomware attacks before unapproved mass file encryption occurs seems vital given the scope and diversity of cyber threats we face today.

Additionally, it is crucial to safeguard user data against all types of crypto-ransomware attacks with no data loss. Making an early detection system that can block crypto-ransomware assaults, even ones that use complicated encryption, is doable by keeping an eye on the Application Programming Interface (API) calls that crypto-ransomware makes. The proposed framework for crypto ransomware pre-encryption detection uses both Machine Learning (ML) and API calls to predict and stop ransomware from causing problems. The machine learning part of this framework is meant to look for patterns in a lot of data that usually point to a ransomware attack. It can find these patterns and learn from them using strong algorithms, which helps it predict future attacks more accurately.

API calls are used by this framework to keep an eye on suspicious activities in real time. It depends a lot on system-level interactions, especially those done through API calls, because they can often be used to spot ransomware activity early on. Usually, a sudden increase or strange pattern in file-level API calls can be a sign that ransomware is about to start encrypting files. By matching up the patterns of API calls found with the predictions of the ML model, the framework can successfully spot ransomware encryption before it happens. Once a possible threat is found, the system flags it right away and does what it needs to do to stop the encryption process. This makes it possible to respond quickly and proactively, limiting the damage and data loss that could come from crypto-ransomware attacks. To prevent user data and files from being encrypted, we proposed an early detection methodology for crypto-ransomware called RENTAKA. This framework consists of ransomware dynamic analysis, pre-encryption boundary identification, feature generation and selection and classifier. According to thorough studies of most incidents, there is a strong correlation between API calls for the Windows platform and ransomware-specific events and processes. System calls must be called for user-level malware, such as ransomware, to interface with the operating system (OS) and carry out its harmful deeds. The operations that software uses during execution are called API calls.

In other words, API calls are a collection of routines offered by the OS for the development of programmes, where each API call carries out a certain function. Using dynamic analysis, the API calls are retrieved after running the ransomware sample in a sandbox. In this study, we show that, when used against the dataset of ransomware families, the suggested method can achieve zero data loss and can identify crypto-ransomware in the early stage. The framework is depicted in Figure 6.

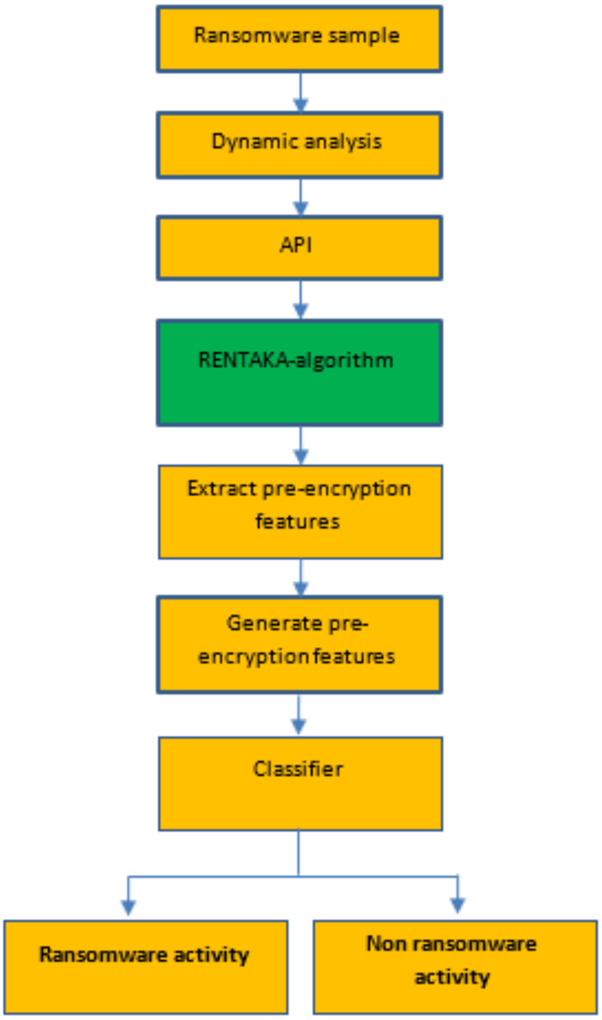

Fig. 6. RENTAKA framework

The framework consists of the RENTAKA-Algorithm to generate the features that are needed for this study (Figure 7). The features the collection of API calls that are used during the pre-attack stages of the ransomware lifecycle. This study aims to determine how feature selection affects classification methods when Cuckoo Sandbox is used. Evaluations were conducted on classification algorithms such as k-Nearest Neighbors, Naive Bayes, Support Vector Machines, and Random Forests. This study's dataset included over 1,584 ransomware samples from eleven families. The Cuckoo Sandbox executes these samples and observes their behaviour in real time. This study illustrates the improvement in accuracy obtained when selecting features using mutual information criteria.

```
Initialize a list of all malware samples as sampleList
FOR each sample in sampleList
Load the sample into the malware sandbox environment
Initialize dynamicAnalysisTool
START dynamicAnalysis on the loaded sample with dynamicAnalysisTool
If dynamicAnalysis is successful
Extract behavioral log and store it in behaviorLog
Locate APIstat cluster within behaviorLog and store in apiCluster
FOR each API in apiCluster
    a.  IF API matches pattern for encryption API
    b.  Flag API as "ENC"
    c.  Store the index of this API in the apiCluster to a variable called encIndex
    d.  BREAK
IF encIndex exists
    a.  Extract all APIs before the ENC flag (up to encIndex) from apiCluster
    b.  Store this extracted list into a file named as "{sample}_APIs.txt"
Terminate the sample execution in the sandbox
ELSE
PRINT "Dynamic analysis failed for the sample: {sample}"
END IF
END dynamicAnalysis
Move to the next sample in sampleList
END FOR
PRINT "Analysis process completed."
```

Fig. 7. The pseudo code for RENTAKA-Algorithm

2.2 Dataset

The dataset (Table 1 and 2) used in this study is from the Resilient Information System Security (RISS) research group from Imperial College London in 2016. This dataset was selected because it has API data for ten ransomware families and a good selection of benign applications. The dataset was created using a dynamic analysis approach for 582 samples of ransomware and 942 samples of benign programs. The data are captured in five main categories with 30,067 features. API calls have 232 features. These researchers successfully used the RISS dataset from different institutions and produced acceptable results.

Table 1
Total samples from each ransomware family in the dataset

No	Sample name	Count
1	Critroni	50
2	Cryptlocker	107
3	Cryptowall	46
4	Kollah	25
5	Kovter	64
6	Locker	97
7	Matsnu	59
8	Pgpcoder	4
9	Reveton	90
10	Teslacrypt	6
11	Trojan-ransom	34

Table 2
The total of data from each category in the dataset

Category	Count
API	232
Registration key	346
Dropped file	6622
Files and directory operation	7500
Embedded string	16267
Total	30967

3. Results

Table 3 below shows how well Random Forest, Naive Bayes, Support Vector Machines (SVM), k-Nearest Neighbours (kNN), and J48 (an implementation of the C4.5 decision tree algorithm) work for classifying data. Accuracy, True Positive Rate (TPR), and False Positive Rate (FPR) are used to measure how well something works.

i. Random Forest
This classifier is the second most accurate of the five. Its accuracy is 96.3934%. It gets true situations right 98.4% of the time (TPR), but 7.1% of the time it gets negative situations wrong (FPR). Overall, this is a good performer with a good balance between accuracy (high TPR) and speed (low FPR).

ii. Naïve Bayes
This is the least accurate of the five classifiers, with an accuracy of 80.9836%. It has the lowest TPR (0.781) and a fair FPR (0.142). This means that it doesn't do as well at both finding true cases and keeping from misclassifying negative cases.

iii. SVM
Out of the five classifiers, this one works the best, with a 97.0492% accuracy rate. It has a great TPR of 0.995, which means that it almost always gets positive cases right. It has the same FPR (0.071) as the Random Forest and kNN models, which means it is also good at avoiding false alarms.

This model is slightly less accurate than the Random Forest and SVM models, with an accuracy of 96.0656%. It has a TPR of 0.979, which is pretty high, but its FPR is 0.071, which is the same as the best models. So, even though it's not as accurate overall, it's about the same at finding real cases and avoiding false alarms.

J48: This classifier is the second worst of the five because it is only 94.7541% accurate. It has the same TPR (0.979) as the kNN model, but its FPR (0.106) is the highest of all the models, which means it mistakes more negative cases for positive ones.

The SVM model seems to have the best balance of accuracy, TPR, and FPR, making it the most effective classifier based on the data.

Table 3
The result from five classifiers

Classifier	Accuracy	TPR	FPR
Random Forest	96.3934%	0.984	0.071
Naïve Bayes	80.9836%	0.781	0.142
SVM	97.0492%	0.995	0.071
kNN	96.0656%	0.979	0.071
J48	94.7541%	0.979	0.106

The confusion matrix (shown in Table 4) for the SVM model used for identifying ransomware gives information about how well the classifier works. The matrix shows how many times the model's predictions were right and how many times they were wrong. It shows how many times ransomware was correctly identified (True Positives), how many times benign software was correctly identified (True Negatives), how many times ransomware was mistakenly identified as benign software (False Negatives), and how many times benign software was mistakenly identified as ransomware (False Positives).

Table 4
Confusion matrix for SVM classifier

	true 0	true 1	class precision
pred. 0	254	12	95.49%
pred. 1	15	155	91.18%
class recall	94.42%	92.81%	

An ROC curve (receiver operating characteristic curve) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters true positive rate (TPR) and false positive rate (FPR). Figure 8 shows the ROC for all machine learning models involved in this study. Based on the figure, the SVM classification model produce the best result.

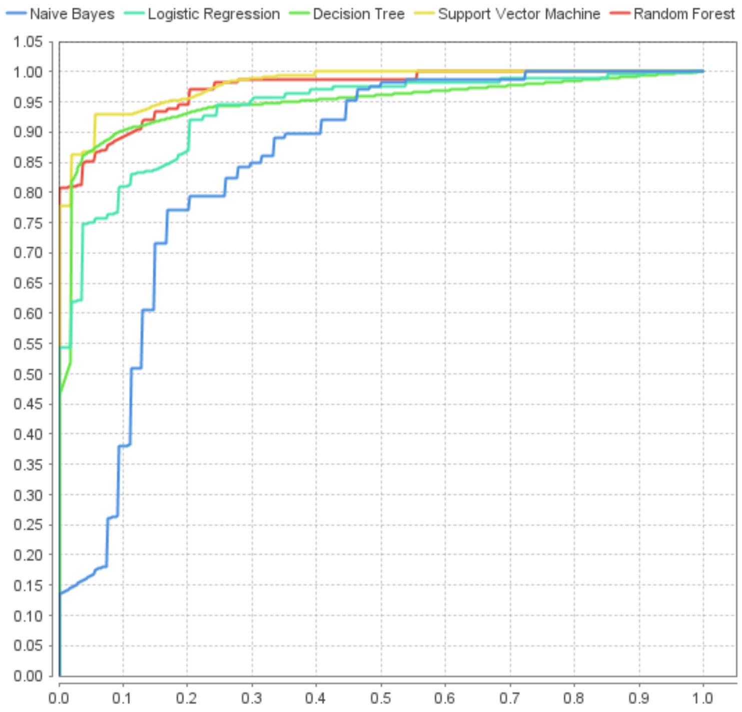

Fig. 8. The ROC comparison of all classifiers

4. Conclusions and Future Works

The security of data belonging to people and organisations is severely threatened by modern cryptographic ransomware. Because the attackers manually infiltrated the target system and conducted reconnaissance, targeted ransomware attacks feature more sophisticated attack paths. Due to the impenetrable locking of many corporate hosts, fraudsters are able to demand large ransoms. A deep understanding of implementation specifics is necessary to develop effective solutions to halt ransomware when preventative measures have failed since more than just relying on prevention measures is needed.

The types of ransomwares, the attack lifecycle, analytical techniques, detection tactics, and related efforts in its detection were all examined in this paper. This work also included a summary of the challenges in early crypto-ransomware detection. Ransomware frequently uses the abstraction that the Windows APIs provide to its advantage. The standard API calls that ransomware uses are in-depth examined in this study. Machine learning classification models enable early detection of ransomware infections, and the suggested technique identifies the characteristics of the ransomware setup. This is necessary to stop attackers from kidnapping the data and stealing it. Additionally, the operation of RENTAKA does not require the preceding availability of a ransomware family as a whole. These results suggest that dynamic analysis can aid in the early detection of novel variations of ransomware, as ransomware samples exhibit a set of identifiable qualities during execution shared by all families.

We proposed a machine learning classifier-based approach to ransomware detection. In our tests, the supervised machine learning technique called support vector machine (SVM) showed the highest degree of accuracy. Crypto-ransomware attacks are dynamic and on the point of becoming a particular kind of attack. Therefore, early detection systems with classification techniques based on machine learning are needed to decrease crypto-ransomware attacks. We will test with more samples and improve the pre-encryption boundary technique in subsequent work. An essential element of this research is the encryption boundary identification algorithm. It details the number of features that will be used to build the machine learning model. Feature engineering is very important for a machine learning model to work well. Most studies are focusing on API calls, system events, network traffic, and other system-level behaviours at the moment. In the future, researchers could look into more complex ways to extract features, such as taking into account correlations between different types of events or higher-level semantic features. Traditional algorithms are used by many machine learning models that look for ransomware.

However, there is growing interest in using deep learning approaches, which can automatically learn features from raw data. Recurrent Neural Networks (RNNs), especially networks with Long Short-Term Memory (LSTM), could be used to model the order of API calls or system events. Many of the current ways to find ransomware using machine learning involve offline analysis, but for practical uses, real-time or near-real-time detection is needed. In the future, work could be done to make models and systems work better in real time, maybe using online learning or incremental learning.

Acknowledgement
This research was not funded by any grant.

References

[1] Ren, Amos, Chong Liang, Im Hyug, Sarfraz Broh, and N. Z. Jhanjhi. "A three-level ransomware detection and prevention mechanism." *EAI Endorsed Transactions on Energy Web* 7, no. 26 (2020).

[2] Noorbehbahani, Fakhroddin, and Mohammad Saberi. "Ransomware detection with semi-supervised learning." In *2020 10th International Conference on Computer and Knowledge Engineering (ICCKE)*, pp. 024-029. IEEE, 2020. https://doi.org/10.1109/ICCKE50421.2020.9303689

[3] Hampton, Nikolai, Zubair Baig, and Sherali Zeadally. "Ransomware behavioural analysis on windows platforms." *Journal of information security and applications* 40 (2018): 44-51. https://doi.org/10.1016/j.jisa.2018.02.008

[4] Hagen, Christoph, Alexandra Dmitrienko, Lukas Iffländer, Michael Jobst, and Samuel Kounev. "Efficient and effective ransomware detection in databases." In *Annu. Comput. Secur. Appl. Conf.(ACSAC)*. 2018.

[5] Sittig, Dean F., and Hardeep Singh. "A socio-technical approach to preventing, mitigating, and recovering from ransomware attacks." *Applied clinical informatics* 7, no. 02 (2016): 624-632. https://doi.org/10.4338/ACI-2016-04-SOA-0064

[6] Spence, Nikki, M. B. B. S. Niharika Bhardwaj, and David P. Paul III. "Ransomware in healthcare facilities: a harbinger of the future?." *Perspectives in Health Information Management* (2018): 1-22.

[7] Kao, Da-Yu, and Shou-Ching Hsiao. "The dynamic analysis of WannaCry ransomware." In *2018 20th International conference on advanced communication technology (ICACT)*, pp. 159-166. IEEE, 2018. https://doi.org/10.23919/ICACT.2018.8323681

[8] Jethva, Brijesh, Issa Traoré, Asem Ghaleb, Karim Ganame, and Sherif Ahmed. "Multilayer ransomware detection using grouped registry key operations, file entropy and file signature monitoring." *Journal of Computer Security* 28, no. 3 (2020): 337-373. https://doi.org/10.3233/JCS-191346

[9] Furnell, Steven, and David Emm. "The ABC of ransomware protection." *Computer Fraud & Security* 2017, no. 10 (2017): 5-11. https://doi.org/10.1016/S1361-3723(17)30089-1

[10] Anghel, Mihail, and Andrei Racautanu. "A note on different types of ransomware attacks." *Cryptology ePrint Archive* (2019).

[11] Al-rimy, Bander Ali Saleh, Mohd Aizaini Maarof, and Syed Zainuddin Mohd Shaid. "A 0-day aware crypto-ransomware early behavioral detection framework." In *Recent Trends in Information and Communication Technology: Proceedings of the 2nd International Conference of Reliable Information and Communication Technology (IRICT 2017)*, pp. 758-766. Springer International Publishing, 2018. https://doi.org/10.1007/978-3-319-59427-9_78

[12] Maigida, Abdullahi Mohammed, Shafi'I. Muhammad Abdulhamid, Morufu Olalere, John K. Alhassan, Haruna Chiroma, and Emmanuel Gbenga Dada. "Systematic literature review and metadata analysis of ransomware attacks and detection mechanisms." *Journal of Reliable Intelligent Environments* 5 (2019): 67-89. https://doi.org/10.1007/s40860-019-00080-3

[13] Fernando, Damien Warren, Nikos Komninos, and Thomas Chen. "A study on the evolution of ransomware detection using machine learning and deep learning techniques." *IoT* 1, no. 2 (2020): 551-604. https://doi.org/10.3390/iot1020030

[14] Molina, Ricardo Misael Ayala, Sadegh Torabi, Khaled Sarieddine, Elias Bou-Harb, Nizar Bouguila, and Chadi Assi. "On ransomware family attribution using pre-attack paranoia activities." *IEEE Transactions on Network and Service Management* 19, no. 1 (2021): 19-36. https://doi.org/10.1109/TNSM.2021.3112056

[15] Shangting, Miao, and Pan Quan. "Industrial cyber range based on QEMU-IOL." In *2021 IEEE international conference on power electronics, computer applications (ICPECA)*, pp. 671-674. IEEE, 2021. https://doi.org/10.1109/ICPECA51329.2021.9362692

[16] Paquet-Clouston, Masarah, Bernhard Haslhofer, and Benoit Dupont. "Ransomware payments in the bitcoin ecosystem." *Journal of Cybersecurity* 5, no. 1 (2019): tyz003. https://doi.org/10.1093/cybsec/tyz003

[17] Ye, Yanfang, Tao Li, Donald Adjeroh, and S. Sitharama Iyengar. "A survey on malware detection using data mining techniques." *ACM Computing Surveys (CSUR)* 50, no. 3 (2017): 1-40. https://doi.org/10.1145/3073559

[18] Zimba, Aaron, and Mumbi Chishimba. "On the economic impact of crypto-ransomware attacks: The state of the art on enterprise systems." *European Journal for Security Research* 4, no. 1 (2019): 3-31. https://doi.org/10.1007/s41125-019-00039-8

[19] Patyal, Manveer, Srinivas Sampalli, Qiang Ye, and Musfiq Rahman. "Multi-layered defense architecture against ransomware." *International Journal of Business and Cyber Security* 1, no. 2 (2017).

[20] Zhang, Zhiyu, Guohang Lu, Chengwei Zhang, Yayu Gao, Yajun Wu, and Guohui Zhong. "Cyfrs: A fast recoverable system for cyber range based on real network environment." In *2020 information communication technologies conference (ICTC)*, pp. 153-157. IEEE, 2020. https://doi.org/10.1109/ICTC49638.2020.9123273

[21] Kharraz, Amin, and Engin Kirda. "Redemption: Real-time protection against ransomware at end-hosts." In *Research in Attacks, Intrusions, and Defenses: 20th International Symposium, RAID 2017, Atlanta, GA, USA, September 18–20, 2017, Proceedings*, pp. 98-119. Springer International Publishing, 2017. https://doi.org/10.1007/978-3-319-66332-6_5

[22] Maiorca, Davide, Francesco Mercaldo, Giorgio Giacinto, Corrado Aaron Visaggio, and Fabio Martinelli. "R-PackDroid: API package-based characterization and detection of mobile ransomware." In *Proceedings of the symposium on applied computing*, pp. 1718-1723. 2017. https://doi.org/10.1145/3019612.3019793

[23] Zhao, Hongwei, Mingzhao Li, Taiqi Wu, and Fei Yang. "Evaluation of supervised machine learning techniques for dynamic malware detection." *International Journal of Computational Intelligence Systems* 11, no. 1 (2018): 1153-1169. https://doi.org/10.2991/ijcis.11.1.87

[24] Ferrante, Alberto, Miroslaw Malek, Fabio Martinelli, Francesco Mercaldo, and Jelena Milosevic. "Extinguishing ransomware-a hybrid approach to android ransomware detection." In *Foundations and Practice of Security: 10th International Symposium, FPS 2017, Nancy, France, October 23-25, 2017, Revised Selected Papers 10*, pp. 242-258. Springer International Publishing, 2018. https://doi.org/10.1007/978-3-319-75650-9_16

[25] Ninyesiga, Allan. *Improved Heuristics for Malware Detection*. Vol. 307570435. A Working Paper on Research Gate accessible via https://www. researchgate. net/publication, 2016.

[26] Tasnim, Nowshin, Khandaker Tayef Shahriar, Hamed Alqahtani, and Iqbal H. Sarker. "Ransomware family classification with ensemble model based on behavior analysis." In *Machine Intelligence and Data Science Applications: Proceedings of MIDAS 2021*, pp. 609-619. Singapore: Springer Nature Singapore, 2022. https://doi.org/10.1007/978-981-19-2347-0_48

[27] Kiraz, Mehmet Sabir, Ziya Alper Genç, and Erdinç Öztürk. "Detecting large integer arithmetic for defense against crypto ransomware." *Cryptology ePrint Archive* (2017).

[28] Rosli, Muhammad Safwan, Abdullah, Raihana Syahirah and Yassin, Warusia. "Discovering Ransomware Behavior by Host-based Approach," *Journal of Theoretical and Applied Information Technology*, vol. 31, no. 14, (2019).

[29] Tsuda, Yu, Junji Nakazato, Yaichiro Takagi, Daisuke Inoue, Koji Nakao, and Kenjiro Terada. "A lightweight host-based intrusion detection based on process generation patterns." In *2018 13th Asia Joint Conference on Information Security (AsiaJCIS)*, pp. 102-108. IEEE, 2018. https://doi.org/10.1109/AsiaJCIS.2018.00025

[30] Mansfield-Devine, Steve. "Ransomware: taking businesses hostage." *Network Security* 2016, no. 10 (2016): 8-17. https://doi.org/10.1016/S1353-4858(16)30096-4

[31] Weckstén, Mattias, Jan Frick, Andreas Sjöström, and Eric Järpe. "A novel method for recovery from Crypto Ransomware infections." In *2016 2nd IEEE International Conference on Computer and Communications (ICCC)*, pp. 1354-1358. IEEE, 2016. https://doi.org/10.1109/CompComm.2016.7924925

[32] Kolodenker, Eugene, William Koch, Gianluca Stringhini, and Manuel Egele. "Paybreak: Defense against cryptographic ransomware." In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pp. 599-611. 2017. https://doi.org/10.1145/3052973.3053035

[33] Herati, Darwish Ahmad, A. M. Bojamma, and MP Indira Gandhi. "Countermeasures to Ransomware Threats." In *5th Annual Conference on Cyber-security, Banglalore, India*. 2018.

[34] Kolodenker, Eugene, William Koch, Gianluca Stringhini, and Manuel Egele. "Paybreak: Defense against cryptographic ransomware." In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pp. 599-611. 2017. https://doi.org/10.1145/3052973.3053035

[35] Connolly, Lena Y., and David S. Wall. "The rise of crypto-ransomware in a changing cybercrime landscape: Taxonomising countermeasures." *Computers & Security* 87 (2019): 101568. https://doi.org/10.1016/j.cose.2019.101568

[36] Beaman, Craig, Ashley Barkworth, Toluwalope David Akande, Saqib Hakak, and Muhammad Khurram Khan. "Ransomware: Recent advances, analysis, challenges and future research directions." *Computers & security* 111 (2021): 102490. https://doi.org/10.1016/j.cose.2021.102490

[37] Zakaria, Wira ZA, Mohd Faizal Abdollah, Othman Mohd, SM Warusia Mohamed SM M. Yassin, and Aswami Ariffin. "RENTAKA: A Novel Machine Learning Framework for Crypto-Ransomware Pre-encryption Detection." *International Journal of Advanced Computer Science and Applications* 13, no. 5 (2022). https://doi.org/10.14569/IJACSA.2022.0130545

[38] Malin, C. H., T. Gudaitis, T. J. Holt, and M. Kilger. "Phishing, watering holes, and scareware." *Deception in the Digital Age* (2017): 149-166. https://doi.org/10.1016/B978-0-12-411630-6.00005-0

[39] Giri, Babu Nath, Nitin Jyoti, and M. Avert. "The emergence of ransomware." *AVAR, Auckland* (2006).

[40] Scaife, Nolen, Henry Carter, Patrick Traynor, and Kevin RB Butler. "Cryptolock (and drop it): stopping ransomware attacks on user data." In *2016 IEEE 36th international conference on distributed computing systems (ICDCS)*, pp. 303-312. IEEE, 2016. https://doi.org/10.1109/ICDCS.2016.46

[41] Shukla, Manish, Sutapa Mondal, and Sachin Lodha. "Poster: Locally virtualized environment for mitigating ransomware threat." In *proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 1784-1786. 2016. https://doi.org/10.1145/2976749.2989051

[42] Salvi, Miss Harshada U., and Mr Ravindra V. Kerkar. "Ransomware: A cyber extortion." *Asian Journal For Convergence In Technology (AJCT) ISSN-2350-1146* 2 (2016).

[43] Liao, Kevin, Ziming Zhao, Adam Doupé, and Gail-Joon Ahn. "Behind closed doors: measurement and analysis of CryptoLocker ransoms in Bitcoin." In *2016 APWG symposium on electronic crime research (eCrime)*, pp. 1-13. IEEE, 2016. https://doi.org/10.1109/ECRIME.2016.7487938

[44] Sami, Ashkan, Babak Yadegari, Hossein Rahimi, Naser Peiravian, Sattar Hashemi, and Ali Hamze. "Malware detection based on mining API calls." In *Proceedings of the 2010 ACM symposium on applied computing*, pp. 1020-1025. 2010. https://doi.org/10.1145/1774088.1774303

[45] Maurya, A. K., Neeraj Kumar, Alka Agrawal, and Raees Ahmad Khan. "Ransomware: evolution, target and safety measures." *International Journal of Computer Sciences and Engineering* 6, no. 1 (2018): 80-85. https://doi.org/10.26438/ijcse/v6i1.8085

[46] Ray, Oliver, Samuel Hicks, and Steve Moyle. "Using ILP to Analyse Ransomware Attacks." In *ILP (Short Papers)*, pp. 54-59. 2016.

[47] Basil Al Jawaheri, Husam, Mashael Al Sabah, and Yazan Boshmaf. "Measurement and analysis of bitcoin transactions of ransomware." In *Qatar Foundation Annual Research Conference Proceedings*, vol. 2018, no. 3, p. ICTPD1026. Qatar: HBKU Press, 2018. https://doi.org/10.5339/qfarc.2018.ICTPD1026

[48] Cabaj, Krzysztof, Marcin Gregorczyk, and Wojciech Mazurczyk. "Software-defined networking-based crypto ransomware detection using HTTP traffic characteristics." *Computers & Electrical Engineering* 66 (2018): 353-368. https://doi.org/10.1016/j.compeleceng.2017.10.012

[49] Mansfield-Devine, Steve. "Ransomware: the most popular form of attack." *Computer Fraud & Security* 2017, no. 10 (2017): 15-20. https://doi.org/10.1016/S1361-3723(17)30092-1

[50] Alotaibi, Fahad M., and Vassilios G. Vassilakis. "Sdn-based detection of self-propagating ransomware: the case of badrabbit." *Ieee Access* 9 (2021): 28039-28058. https://doi.org/10.1109/ACCESS.2021.3058897

[51] Piggin, Richard. "NIS DIRECTIVE AND THE SECURITY OF CRITICAL SERVICES." *ITNOW* 60, no. 1 (2018). https://doi.org/10.1093/itnow/bwy021

[52] Adamov, Alexander, and Anders Carlsson. "The state of ransomware. Trends and mitigation techniques." In *2017 IEEE East-West Design & Test Symposium (EWDTS)*, pp. 1-8. IEEE, 2017. https://doi.org/10.1109/EWDTS.2017.8110056

[53] Gupta, Sanchit, Harshit Sharma, and Sarvjeet Kaur. "Malware characterization using windows API call sequences." In *Security, Privacy, and Applied Cryptography Engineering: 6th International Conference, SPACE 2016, Hyderabad, India, December 14-18, 2016, Proceedings 6*, pp. 271-280. Springer International Publishing, 2016. https://doi.org/10.1007/978-3-319-49445-6_15

[54] F. Cohen, F. Cohen, F. Cohen, and F. Cohen, “Ransomware menace grows as new threats emerge,” *Network Security*, vol. 2016, no. 8, pp. 1–2, (2016). https://doi.org/10.1016/S1353-4858(16)30072-1

[55] Shao, Sicong, Cihan Tunc, Pratik Satam, and Salim Hariri. "Real-time irc threat detection framework." In *2017 IEEE 2nd International Workshops on Foundations and Applications of Self* Systems (FAS* W)*, pp. 318-323. IEEE, 2017. https://doi.org/10.1109/FAS-W.2017.166

[56] Duggan, Michael. "The Legal Corner (TLC): Ransomware Attacks Against Health Care IT." *Journal of Informatics Nursing* 2, no. 4 (2017).

[57] Chittooparambil, Helen Jose, Bharanidharan Shanmugam, Sami Azam, Krishnan Kannoorpatti, Mirjam Jonkman, and Ganthan Narayana Samy. "A review of ransomware families and detection methods." In *Recent Trends in Data Science and Soft Computing: Proceedings of the 3rd International Conference of Reliable Information and Communication Technology (IRICT 2018)*, pp. 588-597. Springer International Publishing, 2019. https://doi.org/10.1007/978-3-319-99007-1_55

[58] Urooj, Umara, Mohd Aizaini Bin Maarof, and Bander Ali Saleh Al-rimy. "A proposed adaptive pre-encryption crypto-ransomware early detection model." In *2021 3rd International Cyber Resilience Conference (CRC)*, pp. 1-6. IEEE, 2021. https://doi.org/10.1109/CRC50527.2021.9392548

[59] Berrueta, Eduardo, Daniel Morato, Eduardo Magaña, and Mikel Izal. "A survey on detection techniques for cryptographic ransomware." *IEEE Access* 7 (2019): 144925-144944. https://doi.org/10.1109/ACCESS.2019.2945839

[60] Alqahtani, Abdullah, Mazen Gazzan, and Frederick T. Sheldon. "A proposed crypto-ransomware early detection (CRED) model using an integrated deep learning and vector space model approach." In *2020 10th Annual Computing and Communication Workshop and Conference (CCWC)*, pp. 0275-0279. IEEE, 2020. https://doi.org/10.1109/CCWC47524.2020.9031182

[61] Al-Rimy, Bander Ali Saleh, Mohd Aizaini Maarof, Mamoun Alazab, Syed Zainudeen Mohd Shaid, Fuad A. Ghaleb, Abdulmohsen Almalawi, Abdullah Marish Ali, and Tawfik Al-Hadhrami. "Redundancy coefficient gradual up-weighting-based mutual information feature selection technique for crypto-ransomware early detection." *Future Generation Computer Systems* 115 (2021): 641-658. https://doi.org/10.1016/j.future.2020.10.002

[62] Kok, S. H., A. Azween, and N. Z. Jhanjhi. "Evaluation metric for crypto-ransomware detection using machine learning." *Journal of Information Security and Applications* 55 (2020): 102646. https://doi.org/10.1016/j.jisa.2020.102646

[63] Continella, Andrea, Alessandro Guagnelli, Giovanni Zingaro, Giulio De Pasquale, Alessandro Barenghi, Stefano Zanero, and Federico Maggi. "ShieldFS: The Last Word in Ransomware Resilient Filesystems." *Black Hat USA* (2017). https://doi.org/10.1145/2991079.2991110

[64] Mercaldo, Francesco, Vittoria Nardone, Antonella Santone, and Corrado Aaron Visaggio. "Ransomware steals your phone. formal methods rescue it." In *Formal Techniques for Distributed Objects, Components, and Systems: 36th IFIP WG 6.1 International Conference, FORTE 2016, Held as Part of the 11th International Federated Conference on Distributed Computing Techniques, DisCoTec 2016, Heraklion, Crete, Greece, June 6-9, 2016, Proceedings 36*, pp. 212-221. Springer International Publishing, 2016. https://doi.org/10.1007/978-3-319-39570-8_14

[65] Al-rimy, Bander Ali Saleh, Mohd Aizaini Maarof, and Syed Zainudeen Mohd Shaid. "Ransomware threat success factors, taxonomy, and countermeasures: A survey and research directions." *Computers & Security* 74 (2018): 144-166. https://doi.org/10.1016/j.cose.2018.01.001

[66] Rhode, Matilda, Pete Burnap, and Kevin Jones. "Early-stage malware prediction using recurrent neural networks." *computers & security* 77 (2018): 578-594. https://doi.org/10.1016/j.cose.2018.05.010