Digital Forensic Readiness Framework
for Ransomware Investigation

Avinash Singh(✉) , Adeyemi R. Ikuesan , and Hein S. Venter

University of Pretoria, Hatfield 0083, South Africa
{asingh,aikuesan,hventer}@cs.up.ac.za

Abstract. Over the years there has been a significant increase in the exploitation of the security vulnerabilities of Windows operating systems, the most severe threat being malicious software (malware). Ransomware, a variant of malware which encrypts files and retains the decryption key for ransom, has recently proven to become a global digital epidemic. The current method of mitigation and propagation of malware and its variants, such as anti-viruses, have proven ineffective against most Ransomware attacks. Theoretically, Ransomware retains footprints of the attack process in the Windows Registry and the volatile memory of the infected machine. Digital Forensic Readiness (DFR) processes provide mechanisms for the pro-active collection of digital footprints. This study proposed the integration of DFR mechanisms as a process to mitigate Ransomware attacks. A detailed process model of the proposed DFR mechanism was evaluated in compliance with the ISO/IEC 27043 standard. The evaluation revealed that the proposed mechanism has the potential to harness system information prior to, and during a Ransomware attack. This information can then be used to potentially decrypt the encrypted machine. The implementation of the proposed mechanism can potentially be a major breakthrough in mitigating this global digital endemic that has plagued various organizations. Furthermore, the implementation of the DFR mechanism implies that useful decryption processes can be performed to prevent ransom payment.

Keywords: Windows forensics · Digital forensic readiness
Ransomware forensics · Memory · Registry · Investigation

1 Introduction

Digital forensics involves the recovery and investigation of data acquired from digital devices related to computer crime [1]. Encrypted devices pose a major challenge in digital forensics, due to the difficulty of retrieving potential evidential information for litigation [2]. In digital forensics, the use of a cryptographic mechanism such as BitLocker[1], and advanced encryption standards to protect system/information is a major problem for an investigator. The Windows operating system (OS), being the most widely used OS [3, 4], is a central target for attackers who exploit the vulnerabilities of each version of the OS. Malicious software (malware) is a constantly

---

[1] https://docs.microsoft.com/en-us/windows/security/information-protection/bitlocker/bitlocker-overview.

growing threat with new variants surfacing exploiting new/undetected vulnerabilities [5]. Anti-viruses can only detect and remove malware with known signatures, deviant behaviour from a normally acceptable signature or based on unique behaviour [5]. However, one specific variant of malware that affects many organizations, companies, and individuals is ransomware [6]. Ransomware infects a machine and then starts encrypting all your files on the machine. It renders the system inaccessible until a ransom is paid to the attacker. In most cases, organizations ended up paying this ransom, due to the cost or lack of backups, reputation preservation, and the implication of prolonged downtime. New variants of ransomware are successfully circumventing existing anti-virus software protocols and other security apparatus [7]. As a process of mitigating ransomware attacks, this study proposed a Digital Forensic Readiness (DFR) mechanism. To the best of our knowledge, this is the first study that attempts to define a viable mechanism that can be used to potentially prevent the payment of ransom to attackers. A concise detail background of malware and a brief overview of DFR is given in the next section. This is then followed by the presentation and evaluation of the proposed mechanism. The paper concludes with related works and discussion of the implication of the proposed mechanism.

2 Background

The potential cost of cybercrime to the global economy, as shown in Fig. 1a could be as high as $500 billion. Furthermore, an approximate of 43% of cyber-attacks are targeted at small businesses [8].

Recent statistics have shown a growing trend of malware-based cybercrime, as depicted in Fig. 1b. According to Symantec, one in every 131 emails contains malware which is one of the biggest promoters of system infection [9]. Suggestively, with the growing trend of software applications and the relatively poor user education, cyber-crime through malicious software will continue to increase as seen in Fig. 1b. The common types of malware often encountered include virus, trojan, spyware, worms, adware, botnets, rootkits, and more recently, ransomware. A virus is a limited form of malware and can easily be detected by anti-virus, using signature-based or deviant-based logic and the effects easily reversed [10]. Trojans is a kind of malware that seems legitimate or is a part of the legitimate software that has been tampered with. The main purpose of a trojan is to gain backdoor access to the system. Given that trojan malware is usually undetected by the user, it has the potential to breach a given security apparatus for other serious attacks [11]. Similarly, spyware malware is designed to surreptitiously gather information or assert control over a given system without the knowledge of the user. Thus, it runs in the background to track the activity, and operational/dormant processes in the computer [11]. Adware malware spreads through an advertising medium usually through downloading free games and/or software. Adware may not necessarily be malicious but it can create backdoors or vulnerabilities in systems that other malware can exploit [10]. Similarly, a rootkit; considered one of the most dangerous and advanced forms of malware, gives an attacker full adminis-trator access to the system [10]. Ransomware is a form of malware that affects a huge number of systems mostly in organizations and institutions. This form of malware
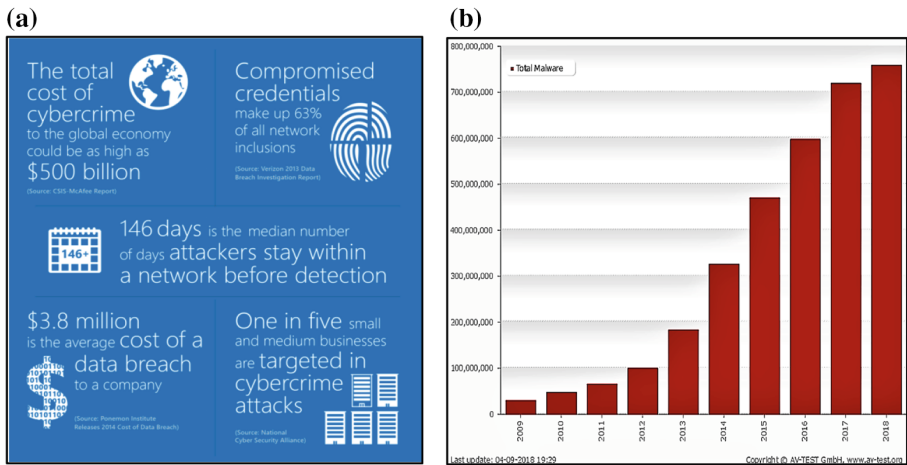
Fig. 1. Microsoft advanced threat analytics infographic (b) A decade statistic of Malware (https://www.av-test.org/en/statistics/malware/)

encrypts the user files while withholding the decryption key as a ransom for huge amounts of untraceable money, usually paid through Bitcoin [12, 13].

2.1 Method of Propagation

Different malware has specific methods of propagation or replicates itself to perform or cause maximum damage as intended. Some of the most common methods of propagation include social engineering [10], wired/wireless networks [14], file sharing [10], virtualized systems [12] and email [12, 15].

2.2 Adaptive Technique of Malware

Over the years, malware started to get more advanced by adapting and counteracting security mechanisms that prevent them from propagating. These techniques adopted by malware make it hard to detect as each technique brings in a new aspect to consider. Some of the common techniques used include polymorphism, metamorphism, obfuscation, DDNS [16] and fast flux [17]. Some of these techniques are further discussed.

- Polymorphism – this technique employs a modification mechanism to avoid signature-based detection. The malware simply changes itself without completely changing the code change its structure [16, 18]. However, some parts of the malware remain the same making it identifiable using adaptive detection algorithms [19].
- Metamorphism – this technique completely rewrites the malware such that it is extremely difficult to identify by anti-malware software. With each propagation this malware is changed, further adding to its unique behaviour making it almost impossible for anti-malware software to identify [16, 19].

- Obfuscation – by using archive files such as (zip, rar, tar, cab). The malware itself pretends to be an archive. This method encrypts the core (malicious) code such that it cannot be detected through an anti-virus. For example, base64 encoding commonly used to sneak malware into the system using HTTP/HTTPS channels [19].
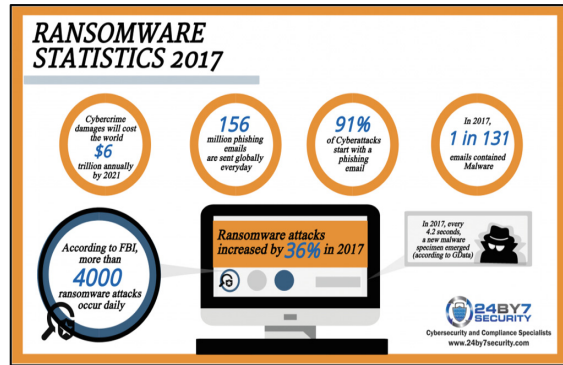
2.3   Ransomware

One of the fastest and widespread propagation of malware is ransomware. Ransomware uses a combination of different malware like a worm to replicate and transfer itself over a network and can be attached to a trojan or an adware to enter the system. As highlighted in Fig. 2, global ransomware attacks increased by 36% in 2017 with more than 100 more variants used by hackers. A total of 34% of entities globally were willing to pay the ransom and about 64% of such entities consisting of Americans [20]. The FBI estimates about 4000+ ransomware attacks globally every day since 2016 [9]. The amount of ransom demanded per attack has increased to an average of $1077 which is an increase of 266% [20]. Ransomware uses scare tactics to trick people to pay using threatening messages and having a time limit to pay before the ransom increases. Ransomware can appear in different shapes and sizes, some being more harmful than others, however, all have the same goal. Common types of ransomware include crypto malware, lockers, scareware, RaaS, and leakware. These types are further discussed.

- Crypto malware/encryptors – is the most common form of ransomware with the capability to cause significant damage within a short duration. This form of ransomware simply encrypts all the files on a machine and extorts money in return to decrypt the files. An example of this is the latest outbreak of the devastating WannaCry ransomware [21, 22].
- Lockers – infect the operating system such that the legitimate user is locked out until they pay the ransom. This is achieved by modifying the bootloader of the OS, such that the malware is loaded instead of the OS making the computer inaccessible. The early forms of this ransomware are not generally used because reloading the bootloader is a simple solution to overcome this ransomware [13].
- Scareware – is a form of ransom disguised as a genuine application. It claims to have discovered security vulnerabilities in a system but demands money to fix them. When the user refuses to pay, the software will display ads and pop-ups, causing the user to think the computer is infected and eventually paying [23].
- RaaS (Ransomware as a Service) – is like a middle-man for an attack. The RaaS provides a ransomware service where malware is hosted and distributed anonymously, managing payments and decryption keys [5, 12].
- Leakware/Doxware – is a form of ransomware that steals personal images and/or information from a computer and then demands a ransom as a form of blackmail.

Each variant leverages a different method or builds on the flaw of another variant to make it more harmful and widespread. A descriptive summary of existing ransomware is presented in Table 1.

The summary in Table 1 reveals that most ransomware exploits the lack of user education and the unpatched security vulnerabilities that exist in Microsoft Windows. Furthermore, it shows that AES-128 encryption is the most common encryption

Fig. 2. Ransomware statistics compiled by 24by7 security (https://24by7security.com/have-you-scheduled-your-first-cybersecurity-task-in-2018-here-are-some-interesting-2017-statistics/)

Table 1. A summary of trending ransomware.

Name	Encryption algorithm	Method of propagation	Vulnerability exploited
WannaCry	AES-128, RSA-2048	EternalBlue	Windows Server Message Block (SMB) protocol
	Remark: WannaCry exploited the SMB protocol by using the EternalBlue exploit which was developed by NSA. This exploits the way Microsoft Windows mishandles specifically crafted packets which enable the execution of certain code from the payload. WannaCry encrypts each file with a different AES-128 key which is further encrypted with an RSA key pair and then added to the file header of each file. In order to get the decryption key, the private key of the Command and Control (C2) server is needed in order to decrypt the encrypted AES-128 decryption key		
(Not)Petya	AES-128, RSA-2048	EternalBlue, Ukrainian tax software update	SMB, Master boot record
	Remark: Similar to WannaCry, however, more harmful. The infection process does not stop upon infecting system files but also changes the bootloader to load the malware. This bypasses the booting of the OS. This is done through the CHKDSK process where instead of loading the OS, it loads the Petya ransomware. While this message is shown it begins spawning processes in the background to encrypt the user files. This ransomware does not need administrator privileges to encrypt protected files since the OS is not loaded managing access control		
Locky	AES-128, RSA-2048	Phishing Email	Microsoft Word Macro
	Remark: Locky ransomware infects the system through social engineering in the form of a malicious Word macro. This macro then runs the trojan binary to start the encryption. The encryption used here is the same approach as that of WannaCry and Petya creating a new trend. This method of encryption is secure if the private key of the C2 server is not globally known. Thus, this method is unbreakable due to the mathematics involved in RSA encryption algorithm		

(*continued*)

Table 1. (*continued*)

Name	Encryption algorithm	Method of propagation	Vulnerability exploited
Cerber	RC4, RSA-2048	Spam Emails and Ads	Microsoft Office Documents
	Remark: Cerber is a RaaS that provides a toolkit which even works if you do not have an active internet connection. This ransomware enters systems through infected office documents that load the malware through a VBScript. This form of malware is well controlled since there are specific hacker groups working together to provide this service to less experienced hackers. This is how they manage to propagate the ransomware faster using affiliate programs and using social engineering techniques. This service is for those criminals who lack the technical expertise to execute such attacks and looking for quick profits		
Crysis	AES-128	Remote Desktop Service, VM environment	Weak or leaked accounts
	Remark: This attack is based on a user-oriented attack where remote desktops services are hacked. This gives attackers control of the machine, allowing them to manually install the ransomware. Crysis also makes use of a C2 which is used to manage and carry out the attack on a larger scale		

algorithm used by ransomware. Given that ransomware infects the system while hindering access to the system, post-mortem forensics (forensics performed after an incident has occurred) is not feasible. Therefore, a more pro-active approach is required to identify, and potentially acquire any cryptographic evidence from a system. The integration of digital forensic readiness into an organization can potentially provide a higher probability of decrypting a ransomware-attacked system as well as providing crucial information/evidence about the attack.

2.4   Digital Forensic Readiness

Digital Forensic Readiness (DFR) is the ability of an organization to maximize its potential to use digital evidence whilst minimizing the costs of an investigation [24]. In essence, DFR is a pre-investigation process which attempts to capture digital information from an identified system prior to the incident occurrence, which might not be available after the incident occurred. Therefore, with DFR, an organization would need to implement on-the-fly evidence collection processes. Thus, a DFR mechanism can be used to conduct an investigation, strengthen the security apparatus of the organization, or to prevent the occurrence of a known attack. With respect to ransomware, DFR can be used to perform analysis and potentially locate cryptographic keys within memory. Two probable methods of accessing the memory contents include targeting the memory allocation space in the running processes in the OS or by scanning through all processes on the system. Due to the lack of evidence collection prior to the incident occurrence, this research proposes a digital forensic readiness framework. This framework will provide a mechanism to securely collect and preserve potential digital evidence.

3   Digital Forensic Readiness Framework

The proposed DFR framework, Windows Registry and RAM Readiness Framework (W3RF), for this study, is presented in Fig. 3. This section describes the proposed framework including the necessary DFR processes defined in the ISO/IEC 27043 standard [25]. The proposed DFR framework attempts to define a method for secure communication, forensic soundness of potential evidence, and the process of decryption key extraction. The framework consists of four interconnected phases and sub-phases. Phase-1 (P1) deals with the process of identifying sources of potential digital evidence from a single computer or networked computer. Phase-2 (P2) considers the process of extracting potential evidential information from the identified component of the system/network. The Windows Registry and RAM are potential sources where ransomware information can be extracted. However, this extraction will occur near real-time using a trigger-based mechanism. The captured data is then securely transmitted to the storage process.
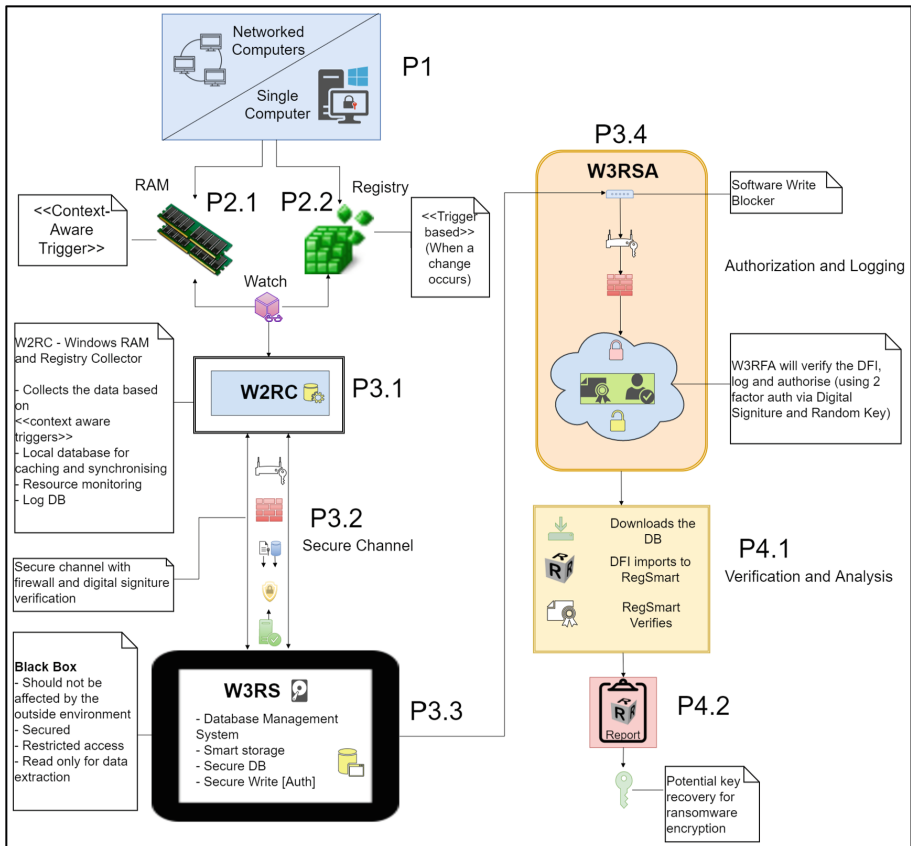

Fig. 3. Digital forensic readiness framework for ransomware investigation

The data storage process constitutes the Phase-3 (P3) of the proposed DFR framework. This phase contains a database management system (DBMS), where an investigator can extract this stored information. The input to this database is connected to the system/network through a one-way secured communication channel. However, the output-channel is connected to a forensic analysis tool which utilizes a forensically-sound communication channel. The forensic attribute of the output channel includes an access control mechanism (authorization, and continuous identity verification), integrity preservation and a verification process. The forensic analysis tool is capable of data-scavenging and the extraction of potential decryption keys. Phase-4 (P4) of the W3RF involves the reporting process of the action and events carried out by a forensic investigator and the system. This phase logs all the actions and processes performed by the authorized entities ensuring chain-of-evidence and chain-of-custody. A high-level overview of the functionality of each component is depicted in Fig. 3. These components are further discussed in detail in the proceeding subsections.

3.1 Memory

The dynamic and static memory (P2.1) of Windows OS has a vast amount of information pertaining to the current state of the machine. It can potentially provide an investigator with significant information about the active user, processes as well as any malicious processes that may be running in the background without the awareness of the user. Most damages that a malicious program can do to a system usually occur in the memory [1, 26]. Memory is a repository for data and program code that is systematically structured. This structure is similar to a linked-list, which consists of fixed block sizes where chunks of allocated data are slotted and stored. The location of these newly allotted data slots makes use of lookup tables to find and locate data within this structure, giving direct access to a specific address for faster access [26].

Given the limited capacity of the dynamic memory, virtual memory is created. Virtual memory is allocated memory on secondary storage, thus, expanding the amount of data that can be stored [27]. The swapping from virtual memory to main memory is called paging [27]. Cryptographic keys can potentially be found in memory as seen in [26, 28, 29], thus providing the opportunity to extract these keys for investigation. Since memory consists of pages, these cryptographic keys can be split over several non-contiguous pages making it more difficult for an investigator to manually scavenge these pages for the keys. For instance, findings in [30] leveraged memory structure to optimize search through a virtual address re-constructor. However, this will work effectively in a post-incident scenario. Therefore, a DFR process would require a near real-time data acquisition. Nevertheless, such a data acquisition process would require a method of continuous data collection.

A context-aware trigger based approach to data collection has been explored in a similar context such as crowd-sourcing [31]. This trigger can actively monitor the system using the least amount of processing power whilst minimizing the overhead cost of searching [26]. Furthermore, the trigger focuses more on newly created processes, thereby minimizing the search space. The signature of a known malware can also be added to the logic of the trigger mechanism. The criteria for the trigger mechanism focus on but is not limited to the entropy change in files, autorun entries

added to the registry, scanning through files and mounted drives, loading of the Windows crypto library and detecting the deletion of shadow volume copies. This criterion was reached through analysis and execution of WannaCry and Petya ransomware. This criterion can also be used for ransomware detection. From [26, 32, 33] volatile memory represents one of the most reliable sources of forensic evidence pertaining to cryptographic keys and active malicious processes.

3.2 Registry

The Windows Registry (P2.2) is a hierarchical database that consists of the systems configurations and user metadata [34]. The registry is a structured database of key-value pairs. This complex data structure allows for the storage of complex information in various formats [35]. It also provides an easier information management process. Ransomware leaves traces of itself in the registry as it uses the registry to modify some system configurations in order to control and manipulate the system [13]. In most cases, ransomware creates and modifies a few keys with one common key (Computer \HKEY_CURRENT_USER\Control Panel\Desktop\Wallpaper) where the ransomware can set the background image to elicit a prompt response for the ransom. Advanced ransomware also creates registry entries that can instruct the system to automatically run the malware if the system is rebooted. The registry can also point to the location of the malware. The registry, therefore, is a good non-volatile source of information to successfully identify ransomware metadata and other evidential information.

3.3 Windows Registry and RAM Readiness Collector (W3RC)

The W3RC (P3.1) is a tool that monitors the memory and registry using trigger-based mechanisms. The flagged data is securely transmitted to the storage process. This tool monitors the system and logs benign processes. The data collection entails any Registry changes and a deep-level process dump whenever a new process is created or modified. This dump can be used for further analysis by an investigator as the executable can be rebuilt. W3RC synchronises to W3RS while maintaining a local cache in the case of an interruption with the transfer process. This process uses a system-collector mechanism in order to get low-level access to the system, whilst ensuring unauthorized alteration. This is considered logical as the OS maintains access control and if a system is infected, it could infect and encrypt this process. This would not be a problem as the data is stored outside the collector in a forensically sound manner. However, one major component of the collection and storage process is the security of the transmission medium.

3.4 Secure Channel

The communication between W2RC and W3RS takes place over a secure encrypted channel (P3.2) over the network. The channel utilizes TCP/IP over SSL/TLS, and each interaction passes through a verification process. The digital signatures are compared and matched to the see if the incoming message is authentic. If the secure channel is compromised or if the connection is broken the W3RS system will capture such event

and send an alert to the system administrator, whilst going into a suspend state to prevent any infection/temperament.

3.5 Windows Registry and RAM Readiness Storage (W3RS)

W3RS (P3.3) is a system that captures and stores the data in a secure and forensically sound manner. This system acts as an isolated black box that ensures that the data is safe and secure. This is also to prevent any ransomware or malware from infecting the storage system. To achieve this, a software write blocker will be used. The system will also have a database management system so that redundant data is not stored. This will help decrease the amount of storage space required to keep the database. One major challenge in DRF is storage capacity [36]. Since data is collected on-the-fly, collecting data over extended periods of time can amount to a few gigabytes of storage space being used up. One way to mitigate this is to overwrite older data, as one would not require events to be stored for long.

3.6 Windows Registry and RAM Readiness Storage Authorization (W3RSA)

This is a subcomponent of the W3RS required for authorization (P3.4). It allows an investigator to retrieve the data stored in the W3RS in a secure manner, whilst performing the necessary logging and authorization processes. An identifier mechanism will be used by the system administrator in order to grant access to the investigator during an investigation. The information and processes are logged for transparency as the data acquired from the W3RS may contain some personal data. The access control mechanism will integrate a 2-phase authentication procedure, therefore making it secure and robust.

3.7 Verification and Analysis

The verification and analysis processes (P4.1) ensures the integrity of the stored data. Log data and hash signatures are attached to the data so that the analysis tool, RegSmart[2], can verify the authenticity of this data as well as the completeness. The tool will then comb through the data, find and aggregate all the data into one report (P4.2) as part of the analysis phase. From the report, an investigator would see the properties, signature, classification level, propagation method, and potential decryption keys of the ransomware.

4 Compliance with ISO/IEC 27043 Standard

The evaluation of the proposed framework in tandem with the standardized digital forensic readiness process, ISO/IEC 27043, is presented in this section. The ISO/IEC 27043 standard provides a generic framework for the implementation of DFR in an

---

[2] https://avinashsingh786.github.io/RegSmart/#regacquire.

organization. Typically, the planning process group of the ISO/IEC 27043 can be used to measure the level of compliance with a given readiness framework during a digital investigation. A direct mapping of the proposed framework with respect to the planning process group is presented in Fig. 4. The different phases of the proposed framework, as shown in Fig. 4, align with the ISO/IEC 27043 standard. More importantly, the proposed framework can be mapped directly with the pre-analysis phase of ISO/IEC 27043. Detail of this compliance is further shown in Table 2. The concurrent processes in the ISO/IEC 27043 standard include managing information flow, documentation, obtaining authorization, preserving the chain of custody, and preserving digital evidence.
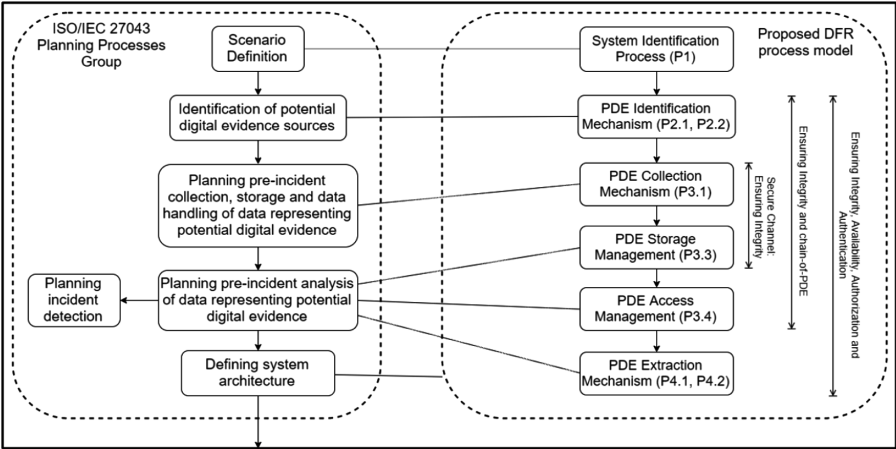
Fig. 4. Mapping of the proposed DFR framework to the ISO/IEC 27043 standard

Table 2. The evaluation process of the proposed W3RF framework

ISO/IEC 27043 planning processes group	Proposed DFR framework	
	Phase	Description
Scenario definition	System Identification Processes	Ransomware evidence collection based on the contents contained within RAM and Registry. Such information gives the ability to detect intrusion and potentially log the decryption keys. It can potentially be used to prevent further propagation of the ransomware
Identification of PDE sources	PDE Identification Mechanism	New registry keys and/or modifications to existing system related keys. Memory monitoring to seek out process memory that is performing some sort of encryption of multiple files

<div align="right">(continued)</div>

Table 2. (*continued*)

ISO/IEC 27043 planning processes group	Proposed DFR framework	
	Phase	Description
Planning, pre-incident collection, storage and data handling of data representing PDE	PDE Collection Mechanism	Using the secure channel to transfer and store the information collected in a secure and forensically sound manner to a black box environment to prevent any malicious attempt on the stored evidence as well as preventing it from being encrypted
Planning, pre-incident analysis of data representing PDE	PDE Storage, Access Management, and Extraction Mechanism	This is a database management system (DBMS) where potential evidence is safely stored per user. Access management is reliant on a verifiable authentication and authorization processes
Defining system architecture	Proposed DFR Process Model	The system architecture consists of the entire framework and respective process model. This can be implemented in either a one-computer-one-DBMS or a distributed system, centralized DBMS which leverages client-server architecture. These configurations depend on organizational policies, infrastructure, and funding

The proposed framework caters for this through W3RS(A) which logs all actions, events, and processes in the system. Furthermore, it provides chain of evidence, chain of custody as well as authorization to obtain the data for analysis thus preserving the digital evidence. The next section describes other related works on cryptographic key recovery from memory and digital forensic readiness.

5   Related Work

Owing to the emergence of new variants of ransomware, and the degree of complexity of the emerging cryptographic mechanism, ransomware detection is difficult [26, 32]. This makes it difficult for investigators to trace where the source of the ransomware and how it propagates. The vulnerability that the malware exploited has been the main focus of security researchers. However, the capability of DFR with regard to memory and registry presents a high potential for the mitigation of ransomware. Several DFR frameworks exist in other fields of forensics such as behavioural biometrics [24, 37], public key infrastructure systems [38], cloud [39] and IoT [40]. The development of readiness framework for these areas presents a milestone for conducting an investigation.

Over the years, the acquisition and analysis of volatile memory have significantly increased [30, 41]. One of the many uses for RAM is to find decryption keys, in a case that the hard drive/device is encrypted. There are many ways to extract potential decryption keys from snapshots/dumps of RAM [2, 26, 30, 41]. These studies have developed new methods and algorithms to extract cryptographic keys. These findings can be leveraged as a mechanism for forensic analysis. A brute force evaluation approach [26] is however considered ineffective. Other studies suggested searching through high-entropy regions, structural properties of the cryptographic key, programming constructs (e.g. C structs), and key schedules. However, due to the increase in security advancements, some of these methods may no longer be viable or could be ineffective. Furthermore, such integration would consider the near-real-time data acquisition process, as these methods were tested and executed in a post-incident scenario on older operating system versions.

6   Conclusion and Future Work

This study proposed a digital forensic readiness framework that can be deployed for ransomware investigation. The implementation of such a framework within a system can significantly produce near real-time potential evidence in contrast to a post-mortem evidence (after the incident has occurred, when potential evidence may have been deleted or encrypted). However, such implementation could be hindered by the security mechanisms of the modern OS. For instance, the OS manages the running processes on a system as well as access control to these processes. Therefore, getting process information may be limited to the access rights provided by the OS. This induces limitation to the amount of information that can be collected in real-time. Moreover, in comparison to post-mortem forensics, the proposed framework can potentially generate more evidential information during a ransomware incident. A preliminary investigation into Windows-10 OS supports this assertion. A reliable and faster method of data discovery, as well as a contextual trigger mechanism for potential data collection, will be explored in the future work. This mechanism will integrate data storage optimization and reduce CPU overhead. From a forensic perspective, this framework presents a mechanism to maximize the cost of legal prosecution and protection against ransom payment. Furthermore, this framework also provides a mechanism to better understand how ransomware works and propagates on a granular level.

References

1. Logen, S., Höfken, H., Schuba, M.: Simplifying RAM forensics: a GUI and extensions for the volatility framework. In: Proceedings of the 2012 7th International Conference on Availability, Reliability and Security, ARES 2012, pp. 620–624 (2012)
2. Hargreaves, C., Chivers, H.: Recovery of encryption keys from memory using a linear scan. In: Proceedings of the 3rd International Conference on Availability, Reliability and Security, ARES 2008, pp. 1369–1376, March 2008

3. Vaughan-Nichols, S.J.: Today's most popular operating systems (2017). http://www.zdnet.com/article/todays-most-popular-operating-systems/. Accessed 12 Apr 2018
4. Statista, Global market share held by the leading mobile operating systems from 2010 to 2015 (2015). https://www.statista.com/statistics/218089/global-market-share-of-windows-7/. Accessed 12 Apr 2018
5. Kaspersky, Overall Statistics for 2017, Kaspersky (2017). https://kasperskycontenthub.com/securelist/files/2017/12/KSB_statistics_2017_EN_final.pdf. Accessed 4 Apr 2018
6. Tailor, J.P., Patel, A.D.: A comprehensive survey: ransomware attacks prevention, monitoring and damage control. Int. J. Res. Sci. Innov. **4**, 2321–2705 (2017)
7. Bromium Labs, Understanding Crypto-Ransomware, p. 35 (2015). Bromium.com
8. Matt Mansfield, Cyber Security Statistics (2017). https://smallbiztrends.com/2017/01/cyber-security-statistics-small-business.html. Accessed 23 Apr 2018
9. United States Government, How to Protecting Your Networks from Ransomware (2016). https://www.justice.gov/criminal-ccips/file/872771/download. Accessed 26 Apr 2018
10. Damshenas, M., Dehghantanha, A., Mahmoud, R.: A survey on malware propagation, analysis and detection. Int. J. Cyber-Security Digit. Forensics **2**(4), 10–29 (2013)
11. Gandotra, E., Bansal, D., Sofat, S.: Malware threat assessment using fuzzy logic paradigm. Cybern. Syst. **48**(1), 29–48 (2017)
12. O'Brien, D.: Internet Security Threat Report - Ransomware 2017. In: Symantec, p. 35 (2017)
13. Savage, K., Coogan, P., Lau, H.: Information resources. Res. Manag. **54**(5), 59–63 (2011)
14. Stone-Gross, B., Cova, M., Gilbert, B., Kemmerer, R., Kruegel, C., Vigna, G.: Analysis of a Botnet takeover. IEEE Secur. Priv. **9**(1), 64–72 (2011)
15. United States Government, How to Protecting Your Networks from Ransomware, pp. 2–8 (2016)
16. Rad, B., Masrom, M., Ibrahim, S.: Camouflage in malware: from encryption to metamorphism. Int. J. Comput. Sci. Netw. Secur. **12**(8), 74–83 (2012)
17. Campbell, S., Chan, S., Lee, J.R.: Detection of fast flux service networks. Conf. Res. Pract. Inf. Technol. Ser. **116**, 57–66 (2011)
18. Spafford, E.H.: The internet worm incident. In: Ghezzi, C., McDermid, J.A. (eds.) ESEC 1989. LNCS, vol. 387, pp. 446–468. Springer, Heidelberg (1989). https://doi.org/10.1007/3-540-51635-2_54
19. Okane, P., Sezer, S., McLaughlin, K.: Obfuscation: the hidden malware. IEEE Secur. Priv. **9**(5), 41–47 (2011)
20. Symantec, 2017 Internet Security Threat Report, Istr (2017). https://www.symantec.com/security-center/threat-report. Accessed 27 Apr 2018
21. Ehrenfeld, J.M.: WannaCry, cybersecurity and health information technology: a time to act. J. Med. Syst. **41**(7), 104 (2017)
22. Kotov, V., Rajpal, M.S.: Understanding Crypto-Ransomware, p. 35 (2015). Bromium.com
23. Sophos, Stopping Fake Antivirus: How to Keep Scareware Off Your Network (2011)
24. Ikuesan, A.R., Venter, H.S.: Digital forensic readiness framework based on behavioral-biometrics for user attribution, vol. 1, pp. 54–59 (2017)
25. ISO 27043, International Standard ISO/IEC 27043: Information technology — Security techniques — Incident investigation principles and processes, vol. 2015 (2015)
26. Kaplan, B.: RAM is key: extracting disk encryption keys from volatile memory, p. 20 (2007)
27. Basu, A., Gandhi, J., Chang, J., Hill, M.D., Swift, M.M.: Efficient virtual memory for big memory servers. In: Proceedings of the 40th Annual International Symposium on Computer Architecture, ISCA 2013, pp. 237–248 (2013)
28. Pomeranz, H.: Detecting malware with memory forensics why memory forensics? Everything in the OS traverses RAM, pp. 1–27 (2012)

29. Olajide, F., Savage, N.: On the extraction of forensically relevant information from physical memory. In: IEEE World Congress on Internet Security, pp. 248–252 (2011)
30. Maartmann-Moe, C., Thorkildsen, S.E., Årnes, A.: The persistence of memory: forensic identification and extraction of cryptographic keys. Digit. Investig. **6**, 132–140 (2009)
31. Adomavicius, G., Tuzhilin, A.: Context-aware recommender systems. In: Recommender Systems Handbook, 2nd edn., pp. 191–226 (2015)
32. Hausknecht, K., Foit, D., Burić, J.: RAM data significance in digital forensics. In: 2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2015, pp. 1372–1375, May 2015
33. Patil, D.N., Meshram, B.B.: Extraction of forensic evidences from windows volatile memory. In: 2017 2nd International Conference for Convergence in Technology (I2CT), pp. 421–425 (2017)
34. Alghafli, K., Jones, A., Martin, T.: Forensic analysis of the Windows 7 registry. J. Digit. Forensics Secur. Law **5**(4), 5–30 (2010)
35. Lallie, H.S., Briggs, P.J.: Windows 7 registry forensic evidence created by three popular BitTorrent clients. Digit. Investig. **7**(3–4), 127–134 (2011)
36. Reddy, K., Venter, H.S.: The architecture of a digital forensic readiness management system. Comput. Secur. **32**, 73–89 (2013)
37. Mohlala, M., Adeyemi, I.R., Venter, H.S.: User attribution based on keystroke dynamics in digital forensic readiness process. In: IEEE Conference on Applications, Information and Network Security (AINS), pp. 124–129 (2017)
38. Valjarevic, A., Venter, H.S.: Towards a digital forensic readiness framework for public key infrastructure systems. In: 2011 Information Security South Africa, pp. 1–10 (2011)
39. Kebande, V.R., Venter, H.S.: On digital forensic readiness in the cloud using a distributed agent-based solution: issues and challenges. Aust. J. Forensic Sci. **50**(2), 209–238 (2018)
40. Kebande, V.R., Karie, N.M., Venter, H.S.: Adding digital forensic readiness as a security component to the IoT domain. Int. J. Adv. Sci. Eng. Inf. Technol. **8**(1), 1 (2018)
41. Dolan-Gavitt, B.: Forensic analysis of the Windows registry in memory. Digit. Investig. **5**, 26–32 (2008)