IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received July 22, 2018, accepted August 30, 2018, date of publication October 1, 2018, date of current version October 19, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2868885

Ransomware Automatic Data Acquisition Tool

LUIS JAVIER GARCÍA VILLALBA[1], (Senior Member, IEEE), ANA LUCILA SANDOVAL OROZCO[1], ANTONIO LÓPEZ VIVAR[1], ESTEBAN ALEJANDRO ARMAS VEGA[1], AND TAI-HOON KIM[2], (Member, IEEE)

[1]Grupo de Análisis, Seguridad y Sistemas, Departamento de Ingeniería del Software e Inteligencia Artificial, Facultad de Informática, Universidad Complutense de Madrid, 28040 Madrid, Spain
[2]Department of Convergence Security, Sungshin Women's University, Seoul 136-742, South Korea

Corresponding author: Luis Javier García Villalba (javiergv@fdi.ucm.es)

This research work was supported by Sungshin Women's University. In addition, this project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 700326.

ABSTRACT Ransomware attacks reported to authorities face the technical difficulty of local police units in gathering information and executing proper forensic analysis. This paper proposes a forensic analysis tool that acts during the final stage of the ransomware infection cycle to provide a quick and easy option to acquire valuable information for the forensic analyst in order to facilitate the subsequent classification of ransomware. The proposed tool combines pop-up window capture showing the ransomware and through the optical character recognition techniques, obtaining the rescue message along with the payment address and value. In addition, it extracts the files generated by the ransomware and dumps the virtual memory of the system for analysis by the forensic technician. To evaluate the accuracy of the tool, experiments were conducted with different samples of ransomware on a real computer, under a controlled environment.

INDEX TERMS Bitcoin, crypto currency, forensic analysis, Internet, memory dump, optical character recognition, pattern recognition, ransomware.

I. INTRODUCTION

In the beginning, Internet use was minimal. It was mainly used by the industrial, military and research sectors, but gradually its use became popular in society. The continuous development of technology and its ease of acquisition attracts a large number of consumers, facilitating the expansion of the Internet. Technological devices became essential storage and communication media for daily use, reaching the point of having private information on devices with Internet access. This caused a minority group of Internet users to become interested in stealing such information for profit. As a result, malicious applications emerged to attack Internet-enabled devices.

Among the different types of malicious software that can be found on the Internet, one of the most dangerous and in recent years widely used by cybercriminals, is called ransomware. Attack campaigns of this type of malware have been seen in different entities, whether public or private. In 2016, there was a campaign targeting hospitals such as the Hollywood Presbyterian Medical Center in Los Angeles, where ransomware-type malware blocked access to the system until a $17,000 ransom payment was made. FBI spokeswoman Laura Miller said they took over the investigations. However, police sources explained to The Time that the hospital had paid the ransom before requesting police assistance [1].

In addition, some hospitals in Germany were targeted, affecting the operation of their computer systems. This was the case with the X-ray system, which was unable to access the data it needed because it was encrypted. Fortunately, the hospitals didn't pay the ransom because they recovered the data from their backups [2].

Other notable targets were large and important media companies such as *The New York Times* [3]. Also, reputable universities such as the University of Calgary where they paid $16,000 to retrieve one week's worth of encrypted mail [4]. The attack on the train ticket vending machines in San Francisco, allowing people to travel without paying for the tickets, was a major attack on the train ticket machines in San Francisco [5].

Most malware is intended to obtain confidential information from businesses and Internet users in general. This is because the storage of personal data on the network is increasingly used. Hence the importance that servers acquire in the operation of technology companies. This makes them the main target for cybercriminals. One example is a massive ransomware attack on MongoDB, where 32,000 servers were hijacked and ransomed with bitcoins to retrieve the information. Several companies, including Telefónica and eBay, as well as governments using the service, were affected [6].

No one is exempt from undergoing a computer attack and with increasingly complex cyber-attacks occurring every day, forensic techniques based on memory analysis are becoming key tools in cyber-crime investigations. Forensic analysts can decipher what happened on a system by acquiring and inspecting the information in the virtual memory. However, the basis of this analysis may be invalidated if the memory acquisition has been altered or poorly executed.

In this context, the police in different countries have seen how the number of complaints related to Ransoware's attacks has grown exponentially, which has led them to seek tools to be able to process all these complaints more quickly without the need for highly specialized personnel and to return their equipment to the complainants as soon as possible. While there are many separate tools that can help in this task, we consider the proposed tool, as it is a single application, it is a more convenient and easy-to-deploy solution.

The rest of the work is structured as follows: The types of malware that exist, the methods of forensic malware analysis and a detailed analysis of ransomware from its appearance to the present day are shown in the section II. Section III presents the work related to malware analysis and classification. The proposed automatic ransomware classification tool is presented in section IV. Section V shows the results obtained in the experiments performed with the tool. Finally, section VI includes the main conclusions drawn from the work and future work.

II. RANSOMWARE AND THE MALWARE EVOLUTION

Malicious software is becoming increasingly sophisticated, its propagation methods have also improved, adapted and become more diverse and ingenious. Thus increasing the number of users who are victims of this type of software. By definition, malware is malicious software that seeks to infect a device or data without the user's knowledge by performing unwanted or harmful actions. This type of software is designed to steal information or disable devices that infect your computer [7].

Any software that harms a user, computer or network can be considered malware. As can be seen in Figure 1, the complexity and sophistication of malware has continued to grow, ranging from simple viruses that infected video games to complex rootkits, with rapid infection capabilities and highly advanced kernel-level stealth techniques [8].

John Von Neumann was the first to theoretically postulate the concept of a computer virus in 1949. It was the first concept of malware and took several years until the first instantiation of the Von Neumann concept appeared [9]. In 1959 a game was developed in Bell Computer's labs called CoreWar consisting of two programs competing to occupy the opponent's memory [10]. The Creeper program, developed in 1971, was considered the first implementation of the Von Neumann concept and expanded through ARPARNET. However, Creeper was not a malware because it was not designed to do harm. Creeper can be considered the father of all worms and viruses. A decade later Fred Cohen in 1984 developed the first malware to gain access to a Wuchner2016 computer [11]. The virus Friday the 13th (1988), also known as "'Jerusalem''. It got its name because it was activated when it matched on the calendar on Friday and the 13th [12].

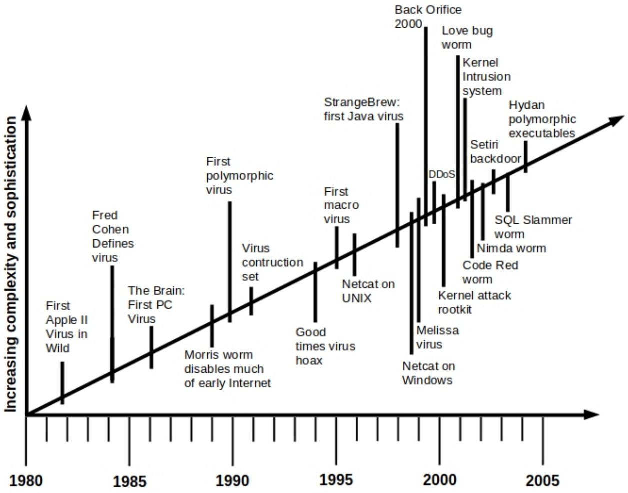

FIGURE 1. Malware evolution [8].

Melissa was one of the first malware to use social engineering to spread by sending emails with an attachment with a *.doc extension containing the malicious code. In 2000, the LoveLetter virus with characteristics similar to those of Melissa, but with greater complexity, was classified as a Trojan and worm.

It modifies the files on the infected computer and is transmitted via e-mail when the user opens the files where it is hosted [13]. Each of these malware have been the most important predecessors of modern malware.

As noted, over time, technology has evolved and malware has adapted to these changes. These adaptations generate variations and increase their scope. There are different types of malware, but they can all be classified into three main groups: Viruses, Worms and Trojans.

- Viruses are replicated on computer resources, are destructive, infect and take control of vulnerable systems.
- Worms try to get the addresses of computers on the network, slowing it down and blocking their communications.
- Trojans perform actions without the user noticing, obtaining or modifying data and send it to criminals.

The Trojans in turn divide into: Exploit: takes advantage of security failures in communications to enter the equipment. Rootkits: are malicious programs that hide certain objects or activities on the system. ransomware: encrypts data so that it cannot be used by the user until a ransom is paid.

Another classification focuses on those that are developed and distributed by legitimate companies, but sometimes include threats to users:

- Riskware: are legitimate programs that can cause harm if used by malicious users. You can delete, block, modify or copy data on computers or networks.
- Pornware: are programs that display pornographic material on a device. One of its purposes is to publish pornographic websites and services subject to fees.
- Adware: is an advertising software, which opens up pop-up windows that can be dangerous.

A. RANSOMWARE

The term ransomware comes from the union of two words ''ransom'' which means rescue and ''ware'' which comes from the term software. Therefore, ransomware refers to malicious software that demands a ransom in exchange for returning control of either the device or the data contained on the infected computer [14]. There are many versions, but most of them run an encryption of the device data, thus denying the user access to them and asking for an economic ransom to retrieve the information again, which rarely happens.

There are different versions of ransomware start dates. How it began to spread, how much money was required to get the data back and where it came from. Glassberg [15] referred to the fact that it has existed for many years, but did not provide an exact date. Kharraz *et al.* [16] stated that it could appear from 2004 onwards, but it was not more significant until ten years later. One of the best known versions dates back to the late 1980s, when malware was propagated to replace the autoexec.bat file with a different file that, after 90 computer restarts, informed the user that he had been infected by a malware that blocked access to the data contained on the computer until a cash ransom of $189 was paid [17].

During the 1990s, ransomware went almost unnoticed until it reappeared in 2005 using new and more powerful encryption schemes. However, as of 2009 when the turning point for ransomware occurs with the birth of the Bitcoin cryptomoney. Cyber-criminals found in her the answer to the problem of anonymity when collecting ransom [18].

With regard to origin, Glassberg suggests that it originated in Russia and Eastern Europe. O'Gorman and McDonald [19] started in Russia in 2009, and spread to Europe and the United States in 2010.

Table 1 shows the most representative families that have been developed since their origins, as well as their evolution over time [17].

- Archiveus: it encrypts the user's files and is required to purchase products from an online pharmacy in exchange for obtaining the password.
- GPCode: is a Trojan that encrypts files and propagates via email as an attachment.
- RansomLock: when the computer is turned on, it displays a window with the message ''blocking access to the data'' and then asks for a ransom to unblock it.

TABLE 1. Ransomware evolution.

Year	Family Ransomware
1989	Cybor
2005	GPCode
2006	Archiveus
2011	Ransomlock
2012	Bootlocker
2013	Reveton
2014	Cryptolocker, Ophionlocker
2015	Teslacrypt
2016	Locky

- Bootlocker: it encrypts the data from the infected computer and installs a bootlocker on the hard disk so that when the user turns on the computer a message appears that the data has been encrypted.
- Reveton: it is known as the police trojan, he is shown a message that says he has broken a law and then locks the screen with the message.
- Cryptolocker: it encrypts files and demands a ransom. He derived a family of ransomware such as CryptDefense, TorrentLocker, CTB-Locker, Cryptowall, Teslacrypt and AlphaCrypt. It uses asymmetric encryption.
- Ophionlocker: if the ransom is not paid within three days the password will be deleted. Use the anonymous TOR network and charge in bitcoin.
- Teslacrypt: attacks files related to computer games such as saved games, user profiles.
- Locky: usually arrives as an unwanted message to the email with an attachment, usually a Word document with macros that will be executed when enabled, when the infection started.

Ransomware can be classified according to the actions it performs on the devices. They are [20]:

- Encrypters: encrypt files and folders. The user notices when he or she tries to open the files and cannot.
- Blockers: block the computer screen with a message and ask for a payment. The files are not encrypted.
- Master boot record (MBR) ransomware: is the part of the hard disk where the operating system boot is located, ransomware changes the boot state by displaying another type of message.
- Web server ransomware: they aim to attack web servers and encrypt their files.
- Mobile phone ransomware: they are usually in the applications they download.

B. FEATURES OF A RANSOMWARE ATTACK

The phases of a ransomware attack are:

- **Dissemination**: The most common way of spreading the message is through e-mail, cyber-criminals use various social engineering techniques to get the user to trust

the message and thus achieve their goal: the execution of malicious software. Some of the most commonly used social engineering techniques are [21]: Executable files with icons, Office files with macros, Use of the RLO (Right to Left Override) character and Phishing.
- **Infection**: Once the malicious payload has been delivered to the victim's system, infection begins. The malicious code is automatically installed in the system, adding new entries in the system logs to ensure that it runs permanently and automatically every time the computer is restarted.
- **Command & Control**: At this stage, the malware attempts to communicate with the server that controls it to obtain, in most cases, the encryption keys and instructions to be followed from this point onwards. The way ransomware communicates with its controlling server varies from family to family and is not always the same. In some cases communication can be via a simple http channel without encryption or they can use more complex channels such as the TOR network to access the controller server.
- **Encryption**: At this stage, ransomware starts encrypting the previously identified files using the instructions and keys sent by your controller server.
- **Blackmail**: Once the files have been fully encrypted, the next step is to let the user know about it. To communicate this to the victim, a window is displayed indicating instructions to be followed in order to consign the payment and release the encrypted data. The way in which extortion is carried out varies from family to family.

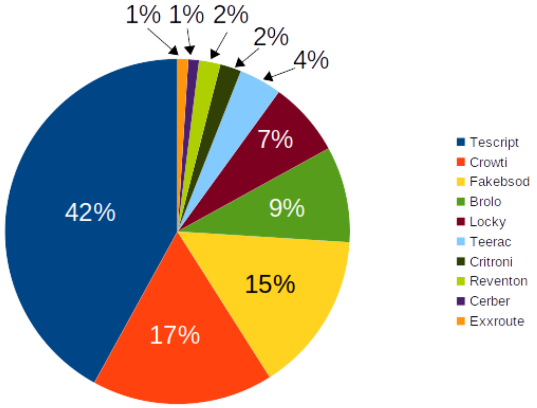

FIGURE 2. Top ten ransomware families.

In all families of ransomware the behavior is similar, the difference is in the payment method and the actions you perform on the victim. The growth of ransomware is due to the fact that the authors use polymorphism to create variations of each type. There are more than 50 families in circulation [22]. The 10 most popular families today are shown in the Figure 2

III. MALWARE FORENSIC ANALYSIS

The purpose of malware scanning is usually to provide the information needed to respond to an intrusion into a computer or network. To this end, there are two fundamental approaches to forensic malware analysis: static and dynamic. This section explains each of these approaches.

Static analysis classifies malware through its persistent representation such as binary files or other file formats that may contain malicious code. These files have a data structure that stores information about the malware code, such as: application type, used libraries, imported functions, exported functions, compilation date, sections, console or textGUI and resources used [11]. Antiviruses can also use this technique as they scan the solid disk in particular [23]. Static analysis is based on two types of searches: text string search and semantic search.

- **Text strings searching**: It operates on binaries without any abstraction or interpretation. It is based on the creation of rules that allow the detection of text strings, instruction sequences, and other patterns within the malicious file.
- **Semantic-based searching**: It covers the abstract of a particular binary, focuses on analyzing the behavior of programs such as flow control and graphs of system calls [11].

The libraries used by malware give out a lot of information about their behavior, whether they are linked statically, dynamically, or at runtime. Runtime links are common in malware, especially when they are obfuscated or packaged. Analysts are interested in dynamic linking because when the program calls the function it will be known so that it can be used [23].

Dynamic analysis is the second approach after static analysis of malware. In order to perform this type of analysis, it is necessary for the malware to be executed and to observe its behavior in order to classify it later. The execution of any malicious software must be carried out with care and within a controlled environment [23]. For this type of analysis, isolated environments are used as a security mechanism to run unreliable programs safely without fear of propagating or harming the real system. These tools generate network activity reports, files created, opened or deleted by each process and activity in the system logs [11].

Isolated environments are in some cases ineffective because many malware can detect that they are running on a virtual machine by changing their behavior and altering the results of the scan [21].

Zheng *et al.* [24] present a ransomware detection tool called GreatEatlon. GreatEatlon uses static analysis techniques to recognize abuse of mobile device management and extracts data flow information needed to detect malicious uses of cryptographic APIs. In [25] a tool called Heldroid is developed that analyzes the three main characteristics of a ransomware: composed of a text, encryption, and blocking analyzer. The tool performs image processing to extract the

TABLE 2. Execution of the memory dump tools on ransomware samples.

Sample / Tool	Cerber	Locky	TeslaCrypt	TeslaVariant	Wannacry	Sage	Time (s)	Without user interaction
Dumpit	✓	✓	✓	✓	✓	✓	50	✓
RAMCapturer	✓	✓	✓	✓	✓	✓	58	×
FTK Imager	✓	✓	✓	✓	✓	✓	52	×
Winpmem	✓	✓	✓	✓	✓	✓	51	×

text from the rescue message. The extracted text is automatically corrected by a spelling checker to remove possible errors during OCR. Finally, depending on a set threshold, the text is classified as dangerous indicating that it probably comes from a ransomware. Experiments with a threatening image dataset analyzed the quality of the scanned image. The tool extracted and classified the text as threatening within a group of images with an original typeface. Other types of letters were tested and simple symbols were extracted, but he failed to recognize more complex symbols. However, it was concluded that thin fonts, handwritten or otherwise, are more difficult for users to read and that attackers often use readable fonts, which allows the tool to behave as expected.

In [26] the authors developed a modular framework called BitIodine, to analyze the blockchain of bitcoin. This tool groups together addresses belonging to the same user or group of users, as well as classifying and tagging them. The information extracted from the bitcoin network is then displayed. The design of the tool is based on several blocks: grouping block, tracking block, network block, classifier block and export block. The tool developed was tested in real cases to find out the relationship between two addresses belonging to the same person. Another utility was to consult a transaction made in a day with a specific value, returning the bitcoin address. The last experiment analyzed the Criptolocker ransomware in which they quantified the number of rescues paid and the information collected from the victim. Several addresses belonging to Criptolocker were collected. These addresses were entered into the tool and were found to belong to several user groups, which have 2118 addresses from which 771 rescues are identified.

In [27] an application based on the Volatility framework was developed with the aim of automating the RAM memory analysis tool and detecting the presence of possible malware by generating a report. The tool searches for image information that details the characteristics of the operating system and its architecture. Volatility calls this profile and is required to execute any commands in the tool. After having the profile, the next step is to analyze the malware: First, check the open connections at the time the volatile memory was removed. The details of open connections are very useful in analyzing malware that is network-based. Most of these malware need to connect to the command and control center to execute the next command or to send specific information, such as password details or files. From the previous steps, you get a list of the processes that communicate with an IP address and the ports used for communication. If an open connection is found, the process identifier, which initiated the communication, is easily retrieved. There is a possibility that such a process may not be legitimate and may engage in malicious activities. It is important to understand that it can be difficult for researchers to check and analyze each executed process, as each can connect to an IP address to confirm whether it is malware or not. The tool then automatically downloads the executable file that initiated the communication. Finally, the tool checks whether the executable file may be malware. To do this, it sends it to VirusTotal to check whether it is malware or not. With the result of VirusTotal ForMaLity it requests a report of the analysis and stores it in a file.

IV. PROPOSED TOOL

x In this work we propose a solution that allows to obtain, besides the memory of the affected system, detailed information of the responsible ransomware. Details such as the rescue message shown to the user, the bitcoin textitwallet address to which the payment should be directed and the value requested as ransom. This data allows the forensic analyst to classify and subsequently correlate different cases within a single ransomware campaign which can be used as evidence before a court to impute these attacks to a particular criminal organization.

To achieve these objectives, the tool, which as a forensic tool runs once the computer has been infected, must be able to perform a full memory dump of the affected system and at the same time perform a screenshot that allows you to obtain the additional information mentioned above for use in the classification and correlation of malware.

In general terms, the proposed technique consists of three phases:

1. Extract information from the ransomware pop-up window from the infected device.
2. Perform a search for files related to the attack.
3. Dump the RAM memory of the infected device.

A. SCREEN CAPTURE ACQUISITION AND ANALYSIS

The most common way to communicate to the user that he has been infected by ransomware is through pop-up windows where he is alerted that his data has been "hijacked". Therefore, analyzing the data contained in these windows allows the analyst to obtain more information to help guide the criminal investigation.

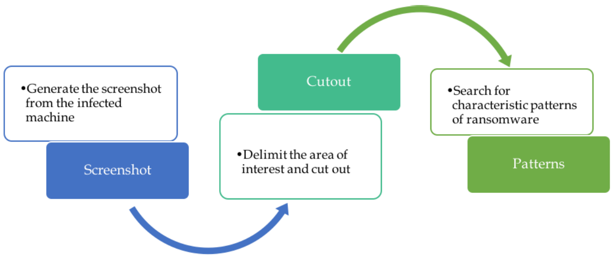
•Generate the screenshot from the infected machine
Screenshot
Cutout
•Delimit the area of interest and cut out
•Search for characteristic patterns of ransomware
Patterns

FIGURE 3. **Ransomware window information extraction.**

To do this, the tool proposed in this work must perform a series of steps to obtain a screenshot that contains only the window of the ransomware and from which the message shown here can be extracted. Each of the steps involved in obtaining the screenshot and extracting the information is shown in Figure 3.

- **Screenshot**: In this first stage the tool takes a full screenshot of the computer. Including all open windows.
- **Cropping**: After the screenshot, the tool searches for patterns that allow you to determine the area of the window related to a ransomware. In this way the image obtained will not contain any information other than the window of the ransomware, ensuring the privacy and anonymity of the user.
- **Message extraction**: Once the image of the ransomware window is obtained, an optical character recognition is performed on the image in order to extract all the text of the message and mainly the address data of the bitcoin wallet as well as the value to be paid for the ransom. At the end of this stage, this final image is stored on the external USB device.

For the screen capture and cropping phase, an artificial vision algorithm was used to obtain the invariant points of interest. Two candidates were chosen: SIFT and SURF Although SURF was slightly faster in detecting these points of interest, SIFT was chosen as the latter outperforms the former in terms of the number of features detected (extracting twice the number of points of interest) and because of the lower number of false positives of SIFT compared to SURF [28]. As for the OCR phase, the Python Pytesseract library was chosen for its versatility compared to other similar options such as Texttract or Pyocr.

B. SEARCH OF RELATED ATTACK FILES

Despite the great variety of ransomware available, the vast majority of them make use of a file as an additional means of communication with the victim. This file contains a warning message, as well as instructions on how to retrieve the information. As the objective of this tool is to obtain as much data and information as possible from the infected computers, the search functionality of these files containing this information was established for future analysis.

This will be done through the recursive directory path, searching for files with specific extensions such as *.txt and *.html, the latter being common in the most recent versions of the different ransomware families.

Searching for relevant files as well as memory dump are utilities that will always be performed.

C. MEMORY DUMP

In memory acquisition, the system memory is collected as an image. This image is then examined by the coroner. The accuracy of memory analysis is based on the correct acquisition of system memory. To this end, we carried out the analysis of different tools that allow us to obtain an adequate report to verify if their characteristics are adequate to the needs of the tool proposed in this work.

The tools evaluated in this paper were:

- **Dumpit**:Memory dump tool developed by MoonSols, which runs easily and extracts the data in memory. It is a tool that remains in continuous development, being its latest version Dumpit v3.0. Its execution generates two files, one is the memory dump and the other with a *.json extension that contains information about the architecture of the host machine. This information is necessary to make it easier for tools specialized in memory analysis to know the architecture of the computer. Dumpit works for most versions of Windows and runs by command console with several options to automate the process without the user interfering.
- **RAMCapturer**: It is a small free forensic tool that allows you to extract the contents of your computer's volatile memory, even if it is protected by an active anti-purge or anti-tipping system. Developed by the company Belkasoft. Separate 32-bit and 64-bit versions are available. It is compatible with all versions and editions of Windows, including XP, Vista, Windows 7, 8 and 10, 2003 and 2008 Server.
- **Winpmem**: It is an open source framework that serves for the extraction of volatile memory. It can be found in GitHub and is written in Python, the first versions run perfectly. But the most current version generates an AFF4 file format which compresses the files since the volatile memory can be very large depending on the amount of memory the computer has. Memory analysis tools such as Volatility are not able to examine dumps with AFF4 extension at the time of doing this work.
- **FTK imager**: It is a virtual memory imaging and data preview tool used to acquire information (memory dumps) in a forensic way by creating copies without making changes in the state of the original evidence. it is a tool widely used for both extraction and memory analysis, thanks to its graphical environment that facilitates its use for the user.

In order to select the best memory dump tool, several memory dumps were run on a computer infected with different samples of ransomware.

TABLE 3. Results of screenshot first tests.

Capture ID	Ransomware	Resolution	Suitable	Matches	Time (secs.)
1	Locky	880x600	Yes	13	48.76
2	TeslaCrypt	1280x800	Yes	12	90.16
3	WannaCry	1280x800	Yes	17	87.92
4	CryptoLocker	1600x900	Yes	18	118.99
5	Cerber	1280x800	Yes	7	69.64
6	Variante Tesla	1280x800	Yes	7	112.56
7	CTB-Locker	1600x900	No	9	127.19
8	Sage	1280x800	Yes	9	68.48

These tests were conducted on a 32-bit architecture laptop with 2GB RAM and running Windows 7 as the base operating system.

The samples of ransomware executed on the computer were: Cerver, Locky, TeslaCrypt, VarianteTesla, Wannacry and Sage, 6 of the most representative of the last years.

The table below 2 summarizes the behavior and total time it took for each tool to extract the virtual memory and store it on a USB device connected to the computer.

Although outside the scope of this work, a possible line of action would be the one carried out in [27] where a tool based on the Volatility framework was developed in order to automate the analysis of the memory dump. The tool loads the memory dump file and searches for open network connections at the time the memory dump was removed. This is very useful as many malwares need to connect to their command and control center. From the list of connections, a list of processes is extracted and with this list all the executables present in the dump file are extracted and analyzed by means of the web tool VirusTotal in order to check if it is a malware. The result of the analysis is stored in a log file.

The end result is a tool capable of performing a memory dump, making a screenshot and analyzing it to extract the window from the ransomware and from it obtain the message information and other relevant details. In addition, we added the functionality of searching in the analyzed computer, files that contain more details or instructions that many of the families of ransomware usually leave once they have entered the system.

V. EXPERIMENTS AND RESULTS

In these experiments, the tool with all the automated functionalities was used to extract relevant information from ransomware, as well as to dump memory and search for files related to the attack within the system.

The machine used has the following features: Intel Pentium 64-bit Architecture Processor, 4GB RAM and Windows 7 Operating System. Also, with the intention of maintaining a real environment, several applications for everyday use were installed for a user such as: Microsoft Office, Adobe Reader, in addition to having a wide variety of text files, images and videos with different formats stored in the different folders of the computer.

As for the USB drive that will contain the tool and store the memory dump, it has a 32GB capacity in NTFS format.

The first group of tests evaluated the technique's ability to identify the ransomware pop-up window and analyze how it affects different operating systems and configurations. In the tests, 62 samples representing characteristic features of a window belonging to a ransomware were used and 8 screenshots were analyzed.

Table 3 presents the configurations and results of the analysis performed. The first three columns show the characteristics of the screenshots. The fourth column shows whether the analysis of the screenshot and the image obtained has been good enough to search for patterns. The Matches column shows the number of matches that were obtained with the pop-up window for each sample. Once the pattern search is complete, it is checked whether the cumulative total of match points reaches the set threshold and also belongs to more samples than the three used. Finally, the time taken by the tool to carry out all the processes was considered.

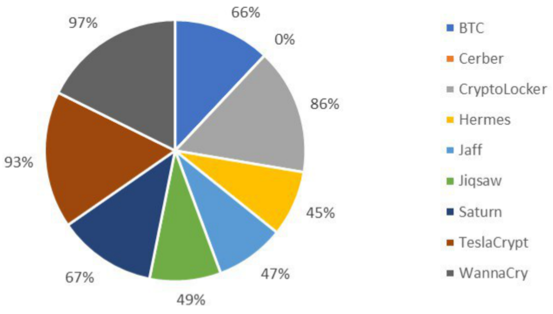

FIGURE 4. OCR recognition result.

To evaluate the effectiveness of OCR, this functionality was applied to the pop-up windows of different samples of ransomware, focusing on key features: Email, URL, Bitcoin address and ransom money obtaining the result that can be observed in Figure 4

Finally, the behavior of the technique is analyzed automatically. The experiments were conducted using 13 samples of ransomware. The tests were conducted in real environments to analyze their scalability.

TABLE 4. Results of running the tool.

Functionality Sample	Screenshot	Cropping	OCR	Files Suplementary	Dump	Matches
CryptoLocker	✓	✓	✓	✓	✓	✓
TeslaCrypt	✓	✓	✓	✓	✓	✓
Cerber	✓	✓	✓	✓	✓	✓
JigSaw	✓	✓	✓	✓	✓	✓
Hermes	✓	✓	✓	✓	✓	✓
BTCware	✓	✓	✓	✓	✓	✓
Saturn	✓	✓	✓	✓	✓	✓
Jaff	✓	✓	✓	✓	✓	✓
GrandCrab	×	×	×	×	×	×
WannaCry	✓	✓	✓	✓	✓	✓
Locky	✓	×	×	✓	✓	✓
Crysis	×	×	×	×	×	×
Vipassana	✓	×	×	✓	✓	✓

In the Table 4 you can see the functionalities that the tool managed to perform inside the infected machine.

After a study of the performance of the tool, we came to the conclusion that there is a need to improve several aspects, such as avoiding storage drive encryption and solving problems by adding checks to ensure that the tool does not fail during execution. In order to solve the problem of data encryption, we carried out future experiments based on the previously obtained data capture, in order to test the different functionalities of the tool in a controlled environment. In the event of a failure in implementation, several checks were carried out which had not been taken into account before. How can be the verification that the cut has been successfully made before trying to do both pattern recognition and OCR, thus getting the rest of the features can be executed. In the following table we can see the result of the experiments under a controlled environment and applying the corresponding evolutionary processes.

The tool worked correctly except for 4 that were not able to perform all the functionalities for the following reasons:

- In the case of Vipassana and Locky, no image cropping or optical character recognition (OCR) was performed, because these samples do not have a pop-up window as is usual in ransomware, but rather change the wallpaper showing the information there.
- When capturing, the tool does not find a region of interest that does not affect sensitive user information, so cropping and recognition is ruled out.
- In the case of Crysis and GrandCrab no test can be performed because in the previous experiment it was not even feasible to get the image capture, because I completely encrypt the storage unit where the tool is stored, not allowing its execution.

VI. CONCLUSIONS AND FUTURE WORK

Today, ransomware analysis has taken on great relevance due to the growth and diversity of this type of malware, as well as the need to understand and analyze its behavior.

The tool proposed in this work allows to obtain relevant information from a ransomware, within a real environment without the need for complex configurations, automating functionalities studied in different sources such as memory dump.

Most of the work researched deals with malware in a controlled environment, which can alter its actual behavior. In many of the studies, the creation of the environments is carried out using tools that require a specific configuration in each case. Limiting these tasks to a group of users with sufficient knowledge to perform the task. In this work the search for patterns in images was carried out in order to obtain sufficient information to determine if it is a ransomware. If the information is sufficient, optical character recognition and extraction in plain text are used to facilitate manipulation. In any case, it will do the memory dump and search for relevant files. All these functionalities are executed from a USB stick where the information obtained is also stored. (In no case does the tool remain resident in the memory). To observe the behavior of the tool, experiments were conducted on ten different samples of ransomware in a real environment. Where it was concluded that the tool has an optimal performance, in the different environments that was executed obtaining the expected results. It was also noted that the tool runtime increases depending on the speed of the USB and the size of the RAM memory. Finally, the following lines of future work could be mentioned:

- Extension of the OCR capabilities of the tool.
- Improved detection of the ransomware message window as well as its processing and cropping.
- Add a database of more samples to increase pattern recognition.
- Implement online storage of results to avoid dependence on USB and protect the tool against ransomware such as Crysis or GrandCab.
- Explore the possibility of a version of the tool that works on other operating systems.

- The phase of analysis of data dumped from memory could be improved by using the timestamp of the files.

ACKNOWLEDGEMENTS

This work could not have been done without the collaboration and development of Lenin Benavides Quintana and Carlos Roa Medina, undergraduate students of the Faculty of Computer Science of the Complutense University of Madrid.

REFERENCES

[1] R. Winton. (2016). *Hollywood Hospital Pays $17,000 in Bitcoin to Hackers; FBI Investigating*. [Online]. Available: http://www.latimes.com/business/technology/la-me-ln-hollywood-hospital-bitcoin-20160217-story.html

[2] R. Millman. *Ransomware Holds Data Hostage in two German Hospitals*. Accessed: Jun. 18, 2018. [Online]. Available: https://www.scmagazineuk.com/ransomware-holds-data-hostage-in-two-german-hospitals/article/530494/

[3] Ediciones EL País. (2016). *Lista de Empresas Afectadas por el Ciberataque*. [Online]. Available: https://elpais.com/internacional/2016/10/21/actualidad/1477081741_222586.html

[4] A. Ivanov, E. David, S. Fedor, and S. Pontiroli, "Kaspersky security bulletin 2016 story of the year: The ransomware revolution," Kaspersky Labs, Moscow, Russia, Tech. Rep., 2016.

[5] S. Gibbs. (2016). *Ransomware Attack on San Francisco Public Transit Gives Everyone a Free Ride*. [Online]. Available: https://www.theguardian.com/technology/2016/nov/28/passengers-free-ride-san-francisco-muni-ransomeware

[6] A. Martinez. (2017). *Un Enorme Ataque de Ransomware Secuestra 32.000 Servidores de MongoDB*. [Online]. Available: https://tinyurl.com/hs6flb5

[7] Avast. *Malware & Antimalware*. Accessed: Jun. 18, 2018. [Online]. Available: https://www.avast.com/es-es/c-malware

[8] E. Skoudis and L. Zeltser, *Malware: Fighting Malicious Code*. Upper Saddle River, NJ, USA: Prentice-Hall, 2004.

[9] Panda Security. *Classic Malware: Su Historia, su Evolución*. Accessed: Jun. 20, 2018. [Online]. Available: http://www.pandasecurity.com/spain/homeusers/security-info/classic-malware

[10] M. J. Erquiaga, "Botnets: Mecanismos de control y de propagacion," in *Proc. Actas XVII Congr. Argentino Ciencias Comput.*, La Plata, Argentina, Oct. 2011, p. 1.

[11] T. Wüchner, "Behavior-based Malware detection with quantitative data flow analysis," Ph.D. dissertation, Dept. Softw. Eng., Tech. Univ. Munich, Munich, Germany, 2016.

[12] Panda Security. *Los Virus Más Famosos de la Historia: Viernes 13*. Accessed: Jun. 20, 2018. [Online]. Available: http://www.pandasecurity.com/spain/mediacenter/malware/virus-viernes-13

[13] Anonymous, M. Burnett, C. Amaris, C. Doyle, L. J. Locher, and R. Morimoto, *Maximum Windows 2000 Security: A Hacker's Guide to Protecting Your Windows 2000 Server and Network*. Indianapolis, IN, USA: Sams, 2002.

[14] F. B. N. Oñate, "Plataformas de ejercicios de ciberseguridad," Ph.D. dissertation, Dept. Ing. Telemática Electrón., Tech. Univ. Madrid, Madrid, Spain, 2016.

[15] J. Glassberg. *Defending Against the Ransom Ware Threat*. Accessed: Jun. 20, 2018. [Online]. Available: http://www.elp.com/articles/powergrid_international/print/volume-21/issue-8/features/defending-against-the-ransom-war-threat.html

[16] A. Kharraz, W. Robertson, D. Balzarotti, L. Bilge, and E. Kirda, "Cutting the gordian knot: A look under the hood of ransomware attacks," in *Proc. 12th Int. Conf. Detection Intrusions Malware, Vulnerab. Assessment (DIMVA)*, Milan, Italy, Jul. 2015, pp. 3–24.

[17] H. Salvi and R. Kerkar, "Ransomware: A cyber extortion," *Asian J. Converg. Technol.*, vol. 2, no. III, p. 1, 2016.

[18] A. Ali, "Ransomware: A research and a personal case study of dealing with this nasty malware," *Issues Informing Sci. Inf. Technol.*, vol. 14, pp. 87–99, Jan. 2017.

[19] G. O'Gorman and G. McDonald, "Ransomware: A growing menace," Symantec Corp., Mountain View, CA, USA, Tech. Rep., 2012.

[20] *The no More Ransom Project*. (2017). [Online]. Available: https://www.nomoreransom.org/ransomware-qa.html

[21] C. V. Liţă, D. Cosovan, and D. Gavriluţ, "Anti-emulation trends in modern packers: A survey on the evolution of anti-emulation techniques in UPA packers," *J. Comput. Virol. Hacking Techn.*, vol. 14, no. 2, pp. 107–126, 2018.

[22] (2017). *Ransom Ware FAQ Windows Defender Security Intelligence*. [Online]. Available: https://www.microsoft.com/en-us/wdsi/threats/ransomware

[23] M. Sikorski and A. Honig, *Practical Malware Analysis: The Hands-on Guide to Dissecting Malicious Software*, 1st ed. San Francisco, CA, USA: No Starch Press, Mar. 2012.

[24] C. Zheng, N. Dellarocca, N. Andronio, S. Zanero, and F. Maggi, "Greateatlon: Fast, static detection of mobile ransomware," in *Proc. Int. Conf. Secur. Privacy Commun. Syst.*, 2016, pp. 617–636.

[25] N. Andronio, and S. Zanero, F. Maggi, "Heldroid: Dissecting and detecting mobile ransomware," in *Proc. Int. Workshop Recent Adv. Intrusion Detection*, 2015, pp. 382–404.

[26] M. Spagnuolo, F. Maggi, and S. Zanero, "Bitiodine: Extracting intelligence from the bitcoin network," in *Proc. 18th Int. Conf. Financial Cryptogr. Data Secur.*, in Lecture Notes in Computer Science, vol. 8437. Berlin, Germany: Springer, 2014, pp. 457–468.

[27] P. H. Rughani, "ForMaLity: Automated forensic malware analysis using volatity," *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 3, p. 177, 2017.

[28] J. Li, C. Li, T. Yang, and Z. Lu, "Cross-domain co-occurring feature for visible-infrared image matching," *IEEE Access*, vol. 6, pp. 17681–17698, 2018.

LUIS JAVIER GARCÍA VILLALBA received the Degree in telecommunication engineering from the Universidad de Málaga, Spain, in 1993, and the Ph.D. degree in computer science from the Universidad Politécnica de Madrid, Spain, in 1999. In 2000, he joined the Department of Electrical Engineering, Faculty of Engineering, Katholieke Universiteit Leuven, Belgium, as a Visiting Scholar in computer security and industrial cryptography. In 2001 and 2002, he was a Visiting Scientist with the IBM Research Division, IBM Almaden Research Center, San Jose, CA, USA. He is currently an Associate Professor with the Department of Software Engineering and Artificial Intelligence, Universidad Complutense de Madrid (UCM). He is also the Head of the Complutense Research Group of Analysis, Security and Systems, Faculty of Computer Science and Engineering, UCM. His professional experience includes the management of both national and international research projects and both public (Spanish Ministry of Research and Development, Spanish Ministry of Defence, and Horizon 2020–European Commission) and private financing (Hitachi, IBM, Nokia, Safelayer Secure Communications, and TB Solutions Security). He has authored or co-authored numerous international publications. He is an editor or a guest editor of numerous journals, such as *MPDI Entropy*, *Future Generation Computer Systems*, *MDPI Future Internet*, the IEEE Latin America Transactions, *IET Communications*, *IET Networks*, *IET Wireless Sensor Systems*, the *International Journal of Ad Hoc and Ubiquitous Computing*, the *International Journal of Multimedia and Ubiquitous Engineering*, *The Journal of Supercomputing*, and *MDPI Sensors*.

ANA LUCILA SANDOVAL OROZCO was born in Chivolo, Colombia, in 1976. She received the Degree in computer science engineering from the Universidad Autónoma del Caribe, Colombia, in 2001, the Degree in computer networks from the Universidad del Norte, Colombia, in 2006, and the M.Sc. in Research degree in computer science and the Ph.D. degree in computer science from the Universidad Complutense de Madrid, Spain, in 2009 and 2014, respectively. She is currently a Post-Doctoral Researcher with the Universidad Complutense de Madrid. Her main research interests are coding theory and information security and its applications.

ANTONIO LÓPEZ VIVAR received the Degree in computer engineering from the Universidad Carlos III of Madrid in 2011 and the master's degree in security of information and communication technologies from the Universidad Europea of Madrid in 2015. He is currently pursuing the Ph.D. degree with the Department of Software Engineering and Artificial Intelligence, Faculty of Computer Science and Engineering, Universidad Complutense de Madrid. He is currently a member with the Complutense Research Group of Analysis, Security and Systems, Universidad Complutense de Madrid, where he is also a Research Support Staff. His research interests are blockchain, cryptocurrencies, computer forensics, and cybersecurity.

ESTEBAN ALEJANDRO ARMAS VEGA received the Degree in computer science from the Polytechnic Institute "José Antonio Echeverría," Havana, Cuba, in 2009, and the M.Sc. degree in computer science from the Universidad Complutense de Madrid, Spain, in 2016, where he is currently pursuing the Ph.D. degree with the Department of Software Engineering and Artificial Intelligence, Faculty of Computer Science and Engineering. He is currently a member with the Complutense Research Group of Analysis, Security and Systems. His research interests include computer networks and computer security.

TAI-HOON KIM (M'17) received the M.S. and Ph.D. degrees in electrics, electronics and computer engineering from Sungkyunkwan University, South Korea, and the Ph.D. degree in computer engineering from Bristol University, U.K. He was a Researcher with the Technical Institute of Shindoricoh for two years. He was a Senior Researcher with the Korea Information Security Agency for two years and six months. He was with Defense Security Command for about two years. He was an Associate Professor with Hannam University for four years and six months. He is currently an Associate Professor of GVSA, Australia, and a fellow of UTAS, Australia. He has authored or co-authored 17 books about the software development, OS such as Linux and Windows 2000, and computer hacking and security. He has authored about 200 papers by 2012. His research interests include computer networks and computer security. He is a member of ACM, KIIT, and SERSC. He is a general chair or a program committee chair of more than 20 international conferences.