# Prevention of Widespread Ransomware Cyber-Attacks through the SEAP Platform

**Eduard-Ștefan SANDU**
Faculty of Applied Sciences, University POLITEHNICA of Bucharest, Romania
edy.eminem@yahoo.com

**Abstract**

*This scientific study aims to explore the potential for launching a cyber-attack through SEAP platform, particularly in light of the increasing use of ransomware as a tool to cause widespread damage to critical infrastructure. The study focuses on the methodology of a ransomware attack on a critical infrastructure, with a specific emphasis on the analysis of the infection process, persistence mechanism, encryption process, recovery prevention, and propagation mechanisms, as well as the communication with command and control servers.*

**Index terms:** critical infrastructures, cyber-attack, cybersecurity, ransomware, SEAP

## 1. Introduction

The Electronic Public Procurement System, commonly referred to as S.E.A.P., is a web-based platform designed to facilitate electronic public procurement in Romania [1]. As the official website operated by the Authority for the Digitization of Romania (A.D.R.), S.E.A.P. serves as a public utility IT system accessible through the internet, which enables public procurement by electronic means. As the operator of the electronic system, A.D.R. is a legal entity under public law responsible for providing technical support and establishing the specific operating framework required to award public procurement contracts to contracting authorities through the electronic procedure, in accordance with relevant legislation.

S.E.A.P. was developed to increase the efficiency and transparency of public procurement processes in Romania, and to enable wider participation in the procurement process. By using electronic means to carry out procurement, S.E.A.P. aims to reduce the administrative burden associated with traditional paper-based processes, and to promote fair competition among suppliers. Plus, the platform aims to enhance the quality and reliability of procurement data, thereby supporting the development of evidence-based policies and practices.

The use of S.E.A.P. is mandatory for all public procurement procedures in Romania, regardless of the value of the contract being awarded. The platform provides a comprehensive set of tools and features for managing the entire procurement process, from publishing tender notices and receiving bids, to evaluating proposals and awarding contracts. Contracting authorities can use S.E.A.P. to create, publish, and manage procurement procedures, and to communicate with potential bidders throughout the process.

All in all, S.E.A.P. represents a significant step forward in the digitization of public procurement processes in Romania. By providing a secure and efficient means of conducting procurement procedures online, S.E.A.P. has the potential to increase the speed and effectiveness of public procurement, while also improving transparency and accountability.

The use of computers and electronic devices has become an integral part of daily life and their widespread use has resulted in a growing concern for cybersecurity. Malware is a type of malicious software that can cause significant harm to computer systems and electronic devices. Malware can take on various forms, ranging from a simple pop-up window to a sophisticated application that tricks the user into giving away sensitive information. This article will provide an overview of malware, its different types, and the impact it can have on computer systems and electronic devices. In particular, the focus will be on how malware can affect the security and functionality of these systems.

The term "ransomware" is used to describe a type of malicious software that demands a ransom from the victim in exchange for the release of compromised data. The name is derived from the combination of "ransom" and "software". Payment of the ransom is often requested in cryptocurrencies like Bitcoin, but there is no guarantee that the data will be restored even after payment. Regrettably, as ransomware continues to evolve and the anonymity of the internet provides ample opportunities for exploitation, cyber attackers are able to take advantage of legal loopholes and evade detection and retribution, transforming ransomware into a highly attractive and low-risk criminal enterprise.

A significant challenge in combating ransomware attacks is the intricacy involved in identifying the specific variant of ransomware utilized. The attack method utilized by ransomware entails the deletion of the original files during a data-level attack, and custom ransomware systems have been developed to incorporate an executable component. The said executable component comprises the malware's payload, expressed as lines of code that execute on the victim's virtual machine to gain control of the victim's files. The malware's behavior is regulated by a remote server that issues instructions and/or controls to the executable, facilitating the execution of its primary function.

Ransomware, a type of malicious software, infiltrates computer systems and restricts users' ability to access internal data stored on virtual machines. Data recovery in the aftermath of a ransomware attack is a precarious process, often involving the targeted individuals paying a ransom imposed by the attacker. However, the efficacy of such payments in ensuring file recovery remains uncertain.

The diversity of ransomware cyberattacks are manifested in several aspects, including but not limited to the encryption methodology used to restrict access to victim files, the complexity and intricacy of the ransomware's architecture, the dissemination channels employed to propagate the malicious code, and the strategies utilized for data recovery.

Ransomware, as a type of malware, exhibits diverse potentialities for propagation through various vectors, including but not limited to:

- one of the prevalent methods for distributing ransomware is through traffic redirection, where victims are lured into accessing specific websites that present themselves as exploit kits. When a user downloads the program, the malware payload exploits vulnerabilities in the virtual machine, leading to the encryption or locking of systems and files.
- a prevalent vector for ransomware propagation is through e-mail attachments or links that entice the target to access web portals containing malware. This tactic often employs the use of social engineering techniques to entice the recipient to access attachments or follow links that lead to websites containing malware.
- botnets are a pervasive method for ransomware propagation, which involves the distribution of malware through infected programs and legitimate programs with infected code. The botnet infects a large number of devices, which can then be used to distribute the ransomware to other devices, causing widespread damage.

## 2. Crypto-ransomware

To classify the encryption methodology of victim files, crypto-ransomware can be classified into three distinct types:

- which overwriting the data of the encrypted file as metadata of the original file and this method can be considered a type of file obfuscation, as the encrypted data is hidden within the original file, making it difficult to detect.
- which changing the original file's location from the source directory to the directory where the encryption takes place, along with renaming the encrypted file and aims to further conceal the encrypted data and make the detection process more challenging.
- which creating a new file and entirely replacing the original file and is the most challenging to detect using a data or system-centric detection model that typically looks for bulk deletion behavior of running processes.

There exist diverse techniques for detecting and mitigating ransomware attacks, which possess strengths and limitations that may be leveraged by cybercriminals to render the malware more elusive. Given that the code underlying ransomware is perpetually evolving in terms of complexity, inventiveness, and targeted objectives, effective detection and mitigation of such attacks necessitate continued research and development of novel countermeasures.

In the event of a ransomware attack targeting multiple critical infrastructures that lack a robust cyber defense plan, the probability of successful data recovery is substantially reduced. In such cases, relinquishing the stolen data may be the only viable option. However, if the attacker has made an error in the development of the ransomware source code, another approach could be to attempt to crack the decryption key, which involves solving a mathematical puzzle.

Often, the targeted infrastructures opt to pay the ransom in order to retrieve the decryption key and regain access to their data. However, this course of action does not always guarantee the successful recovery of the stolen data or the attacker's compliance with the agreement. The use of cryptocurrency as the preferred method of payment presents a further layer of complexity for decision-makers who may be unfamiliar with the technology.

I shall present a tabulated account of the most massive ransomware attacks to date, along with their respective dates of detection and the classification of the encryption algorithms employed.

**Table 1.** Encryption algorithms for ransomware attacks [2]

| # | **Malware** | **First known appearance** | **Encryption algorithms** |
|---|---|---|---|
| 1 | GPcoder | 2004 | AES - ECB |
| 2 | Crypto Locker (Gameover ZeuS) | 2013 | AES |
| 3 | Crypto Wall | 2014 | AES - CBC |
| 4 | CTB Locker | | AES - ECB |
| 5 | Torrent Locker | | AES - CTR \| CBC |
| 6 | Tesla Crypt | 2015 | AES - ECB \| CBC |
| 7 | Crypt Vault | | RSA - OAEP |
| 8 | Locky | 2016 | AES - CTR \| EBC / AES RSA + ECB |
| 9 | Petya | | Salsa20 |
| 10 | Not Petya | | MFT - Salsa20 - AES |
| 11 | WannaCry | | AES - RSA |
| 12 | SamSam | | RSA |
| 13 | Hermes | 2017 | RSA - AES - CBC |
| 14 | Ryuk | 2018 | RSA - AES - CBC |

The data presented in Table 1 suggests that ransomware attackers frequently exploit the encryption algorithms that are trusted by critical infrastructures to safeguard their important data. It is additionally noted that ransomware developers tend to replicate established encryption processes,

with the majority of their efforts directed towards the development of new methods for infiltration and infection.

### 2.1. Analyzing the classic methodology employed by ransomware

The initial step in the ransomware attack process involves the downloading of the ransomware onto the targeted virtual system. However, this download alone does not necessarily trigger the destructive consequences associated with ransomware. Instead, the download serves as the first phase of a complex process informally referred to as "infection". The downloaded payload, known as a binary, contains the code that directs the actions of the ransomware and is typically transmitted through a variety of infection methods that are tailored to the specificities of the attack.

The second step comprises the code that governs the actions of the ransomware according to the instructions provided by the attacker. The execution instructions differ widely based on the specifics of the attack, making it especially challenging to develop an accurate ransomware execution model. Still, the execution of ransomware can be separated into three general steps, as outlined below [3]:

- **stealth operations**. Before initiating an attack, ransomware must become acquainted with the victim's system and cybercriminals aim to keep their ransomware undetected during this preliminary step to avoid prematurely aborting the attack.
- **suspicious activities**. Ransomware launches the malevolent component of the attack covertly without disclosing its presence and has the ability to evade detection. In the case of locker-ransomware, this phase involves impeding the user interface, while in crypto-ransomware, it entails encrypting the targeted data.
- **obvious actions**. Upon completion of the encryption process, ransomware typically presents a ransom note, which serves as a form of communication to the victim regarding the ransom demand and the method of payment. The ransom note can be displayed as a pop-up window, a file or a wallpaper, and usually includes instructions on how to access the decryption key.

In the absence of a successful defense against the ransomware attack, the critical infrastructure is left with the decision of whether or not to acquiesce to the attacker's demands. If the organization chooses to comply, payment is typically made in the form of digital currency, as these transactions are unregulated and allow for a degree of anonymity. The ransom note will often provide instructions for purchasing digital coins from online exchanges and transferring them to the attacker's designated wallet address.

In the next-to-last stage, pertaining to the specific scenario of a crypto-ransomware attack, it is noteworthy that subsequent to the payment, the ransomware may selectively undertake a course of action, which could involve automatic decryption of the files or provision of a decryption binary to the victim, or it may alternatively fail to render the encrypted data intelligible.

In conclusion, the final step in a cyber-attack involves converting the digital currency reward into national currency. This process poses significant risks as it can compromise the anonymity of the attacker and link them to the crime. To mitigate this risk, attackers may use the shuffling technique to cover their tracks. By obfuscating redemption payments, the origin and destination of funds are concealed, preserving the anonymity of the attacker. Despite its effectiveness, the exchange of digital currencies remains a precarious step, requiring careful consideration and strategic planning to avoid legal and criminal consequences.

### 2.2. Analyzing Ransomware Attacks on Virtual Machines: Understanding the Effects

Ransomware attacks are executed in five stages, which are as follows:

- during the deployment phase, ransomware undergoes a process whereby its various components are installed in order to effectively infect, encrypt, or crash the targeted

virtual machine. This is achieved through the utilization of drive-by downloads, phishing e-mails, and the exploitation of system vulnerabilities that are readily accessible to the malware.

- within the installation phase of a ransomware attack, a payload is delivered to the targeted system, thereby triggering the start of the infection process. The code responsible for the attack is carefully crafted to avoid detection and subsequently establish communication with command and control (C&C) servers. This is achieved through the manipulation of keys within the operating system registry, which are specifically configured to activate starting code. As the malware propagates itself throughout the network, it utilizes standard processes such as "explorer.exe" or "svchost.exe".
- after the previous steps have been successfully completed, the code starts reaching the command and control server for instructions, which include commands about the types of files to be encrypted, how long to wait before starting the process, and similar commands based on the specifics of the attack. The code will also extract system information such as the IP address, domain name, operating system, and anti-malware products to the C&C channels which can be unencrypted HTTP, encrypted HTTP or anonymous via TOR.
- the fourth stage, which involves destruction, will mark the beginning of an encryption procedure for all targeted files, which involves all types of documents .JPEG, GIF, .CAD, or others without being limited to these circumstances only.
- in the ultimate stage, the coercive strategy of blackmail is employed, whereby a pop-up interface materializes subsequent to the complete encryption of files. This interface contains explicit instructions regarding the remittance of ransom and the consequences that will ensue if the payment is not made within the designated timeframe.

The use of programming languages like C and C++ in the development of ransomware attacks provides attackers with a high degree of flexibility and control. These languages are well-suited to the development of malware due to their low-level access to computer hardware and operating system resources. The development of ransomware attacks is typically characterized by the use of programming languages such as C or C++, and the evolution of these attacks results in increasingly complex and dangerous forms of malware. Understanding the programming languages and evolutionary patterns of ransomware attacks is crucial for developing effective mitigation strategies and protecting against these threats.

The latest generation of ransomware has made it increasingly difficult to gain access to files without complying with the attacker's demands. This is achieved through the use of a combination of symmetric and asymmetric encryption techniques in the encryption process. Symmetric encryption, where the same key is used for both encryption and decryption, is employed to encrypt the bulk of the data quickly. Asymmetric encryption, on the other hand, where a public key is used for encryption and a private key is used for decryption, is used to encrypt the symmetric key used for encryption.

In the following section, we present a breakdown of a particular case of ransomware:

- when the link that contains the malicious code is accessed for the first time, the ransomware will be downloaded on the targeted system.
- after the ransomware is accessed and executed, the execution stage is initiated.
- as per reference [4], the ransomware gathers hashed details about the targeted virtual machine, which may include but are not restricted to CPU, hostname, and RAM. This data is used to recognize the device architecture and generate a type key that is stored in the registry at the following paths: *HKCU\Software\<uniquecomputer id>\<random id>* and *HKCU\Software\[random]*.

- the program has a built-in mechanism to detect the presence of any previous ransomware that may have infected the system. It performs a thorough scan of the system's files and folders to search for any indicators of ransomware activity, such as encrypted files, ransom notes, or suspicious processes running in the background.
- the ransomware is designed to inject itself into the "explorer.exe" extension, which is a legitimate executable file used by Windows to manage the desktop, taskbar, and file explorer. To achieve this, the ransomware creates a new instance of "explorer.exe" and modifies its code to include the malicious payload.
- the ransomware establishes a connection with a command and control server to remain undetected by the target's internal security protocols. This connection allows the ransomware to receive commands from the attacker and send back information about the infected system and makes it more difficult for the target's security team to detect and respond to the ransomware attack in a timely manner.
- reference [5] suggests that ransomware can behave differently depending on the specifics of the attack. Upon execution, the ransomware may start running and performing its malicious activities in one or more of the following ways: % Localappdata %, % ProgramData%, % UserProfile%, % Temp%.
- alternatively, it is also possible to create the extension "svchost.exe" for the injection of the ransomware. This extension is typically located in the "C:\Windows\System32" directory and is a legitimate process used by Windows to host multiple services.
- once the ransomware is injected into the target system, the attackers may use various services to carry out their malicious activities. For instance, the "vssadmin.exe" service can be used to delete or encrypt system copies. In some cases, the "wmic.exe" service may also be employed to delete all files on the infected system.
- the attackers may attempt to prevent backup applications from altering the startup options of the infected system. This is typically achieved by disabling startup recovery using the "bcdedit.exe" extension.
- after disabling backup options and other security measures, the attackers may use various Windows services to encrypt the victim's files. The attackers may use "syskey.exe" to encrypt the victim's files using a randomly generated key and, once the files are encrypted, the attackers may use another Windows tool called "cipher.exe" to manage the encrypted data.
- once the attackers have successfully encrypted the victim's files using tools like "syskey.exe" and "cipher.exe", they notify the command and control (C&C) server to provide the encryption keys along with the target ID and password.
- in order to keep the communication between the attackers' command and control (C&C) server and the infected system secure and private, data traffic is generally executed using an encrypted protocol such as HTTPs or TOR where the attackers can hide the data traffic between the server and the infected system from detection by security tools and network administrators.
- after the encryption process is completed, the files are transformed into a format that is not readily accessible to the victim. Once the encryption process is finished, a notification message is displayed on the user's device. This message typically contains a ransom demand from the attacker, which outlines the conditions for the release of the encrypted files.
- Ultimately, as part of the encryption process, the ransomware will often change the file extensions of the encrypted files to specific ones like ".crypt", ".cryptolocker", ".cryptowall", and ".encrypted" and to make it easier for the attacker to identify them for future reference.

To summarize, ransomware is a type of malicious software that can infect a computer system and encrypt files using complex mathematical algorithms. The ransomware may migrate and hide in the "explorer.exe" extension, changing it to a new, infected extension that it copies to paths like "% Appdata%" and "% Programdata%", and modify the registry value in "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\Currentversion\Run". The ransomware can also disable useful services using extensions like "vssadmin.exe" and "bcdedit.exe" to prevent the system from reacting and delete sensitive data to force the target to pay the ransom. After encryption, a public key is applied using one of the asymmetric encryption algorithms. If the system is rebooted, the ransomware will run continuously and keep the connection between the command and control server at all times.

## 3. Case study - WannaCry

This scientific research describes a method for carrying out a cyber-attack and assesses the risks that cyber systems face through the SEAP platform. The SEAP platform is used to publish procedures related to awarding public procurement contracts/framework agreements or advertising notices. As a result, it represents a potential target for cyber-attack and, in this case study also highlights the vulnerabilities of cyber systems that can be exploited by attackers.

To access the procedure, users can navigate to the related initiation notice and select the "View procedure" button. This will redirect the user to a list of initiation notices of the corresponding type, from which the user can choose the desired initiation notice. Once registered in the procedure, the economic operator can proceed to submit their offer by accessing the "My offer" section of the procedure viewing screen. The application process involves uploading the necessary files in the "My Offer" section, specifically in the Qualification Documents subsection.

The contracting authority will download and review the documents that have been submitted during the award procedure/advertisement in order to assess the potential bidders and ultimately determine the winning bid.

Our study aims to examine the process of transmitting the offer, which is submitted in a PDF format. We will focus on a particular type of worm that is injected into the offer, and which is designed to execute a cyber-attack of the ransomware variety. The worm's primary objective is to encrypt the contracting authority's data, rendering it inaccessible and held for ransom. Through our investigation, we hope to gain a better understanding of the potential vulnerabilities within the offer submission process and to develop effective countermeasures against such attacks in the future.

The contracting authority is likely to face significant challenges in protecting against a ransomware attack similar to WannaCry. Such attacks have been observed to spread rapidly through the use of a worm component, making it difficult to contain and mitigate the damage caused. Furthermore, the encryption component of WannaCry employs public key cryptography, which is a robust and widely-used encryption method that presents significant challenges for decryption without the proper key, so it is crucial for the contracting authority to develop and implement comprehensive cybersecurity measures to minimize the risk of such attacks and ensure the integrity of their systems and data.

Moving forward, we will analyze two distinct executable components: the worm and encryption components. Table 2, 3 and 4 [6] provides a detailed breakdown of these components, including their respective functions and properties.

**Table 2.** Executable components: the worm and encryption components

| | **Worm component** |
|---|---|
| MD5 | db349b97c37d22f5ea1d1841e3c89eb4 |
| SHA 1 | e889544aff85ffaf8b0d0da705105dee7c97fe26 |

| | **Worm component** |
|---|---|
| SHA 256 | 24d004a104d4d54034dbcffc2a4b19a11f39008a575aa614ea04703480b1022c |
| File type | PE32 executable (GUI) Intel 80386, for MS Windows |
| | **Encryption component** |
| MD5 | 84c82835a5d21bbcf75a61706d8ab549 |
| SHA 1 | 5ff465afaabcbf0150d1a3ab2c2e74f3a4426467 |
| SHA 256 | ed01ebfbc9eb5bbea545af4d01bf5f1071661840480439c6e5babe8e080e41aa |
| File type | PE32 executable (GUI) Intel 80386, for MS Windows |

The Pestudio analysis conducted on WannaCry's executable components yielded the following results regarding its worm and encryption functionalities:

**Table 3.** The Dynamic Link Libraries (DLLs) that comprise the structure of a worm

| **Library** | **Imports** | **Description** |
|---|---|---|
| ws2_32.dll | 13 | Windows Socket 2.0 32-bit DLL |
| iphlpapi.dll | 2 | IP Helper API |
| wininet.dll | 3 | Internet Extensions for Win32 |
| kernel32.dll | 32 | Windows NT Base API Client DLL |
| advapi32.dll | 11 | Advance Windows 32 Base API |
| msvcp60.dll | 2 | Windows NT C++ Runtime Library DLL |
| msvcrt.dll | 28 | Windows NT CRT DLL |

**Table 4.** The Dynamic Link Libraries (DLLs) that constitute the encryption component

| **Library** | **Imports** | **Description** |
|---|---|---|
| kernel32.dll | 54 | Windows NT Base API Client DLL |
| advapi32.dll | 10 | Advance Windows 32 Base API |
| user32.dll | 1 | Multi-User Windows User API Client DLL |
| msvcrt.dll | 49 | Windows NT CRT DLL |

During its execution, the worm framework DLLs call upon the iphlpapi.dll extension to obtain the network configuration settings of the infected host. On the other hand, the encryption component heavily relies on the kernel32.dll and msvcrt.dll libraries, which are among the most frequently invoked DLLs. This suggests that these two malicious libraries are responsible for the primary encryption functionality implemented in the component. To verify this assertion, a closer examination of the imported functions of these libraries will be carried out.

**Table 5.** The functions that facilitate the encryption format [6]

| **Function** | **Location** |
|---|---|
| GetCurrentThread | 0x53a |
| GetStartupInfoA | 0xa97a |
| StrartServiceCtrDispatcherA | 0xa6f6 |
| RegisterServiceCtrDispatcherA | 0xa6d8 |
| CreateServiceA | 0xa688 |
| StartServiceA | 0xa662 |
| CryptGenRandom | 0xa650 |
| CryptAcquireContextA | 0xa638 |
| OpenServiceA | 0xa714 |
| GetAdaptersInfo | 0xa792 |
| InternetOpenUrlA | 0xa7c8 |
| OpenMutexA | 0xda84 |
| GetComputerNameW | 0xd8b2 |
| CreateServiceA | 0xdc2a |
| OpenServiceA | 0xdc62 |
| StartServiceA | 0xdc52 |
| CryptReleaseContext | 0xdc14 |
| RegCreateKeyW | 0xdc04 |

| Function | Location |
|---|---|
| fopen | 0xdcd4 |
| fread | 0xdccc |
| fwrite | 0xdcc2 |
| fclose | 0xdcb8 |
| CreateFileA | 0xd922 |
| ReadFile | 0xd964 |

Table 5 displays the list of the most suspicious functions identified among them. Overall, the analysis indicates that WannaCry predominantly employs Microsoft's Crypto, File Management, and C Runtime APIs for its operations. The crypto API library is specifically used to generate and manage both symmetric and asymmetric cryptographic keys, which play a vital role in ensuring secure transmission and storage of sensitive information.

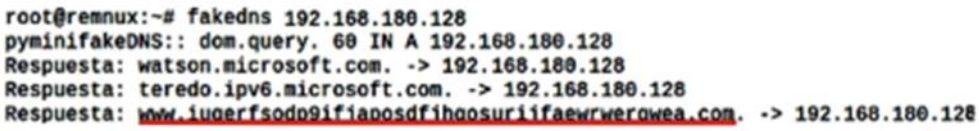


**Fig. 1.** The process of capturing malicious DNS requests through FakeDNS [6]

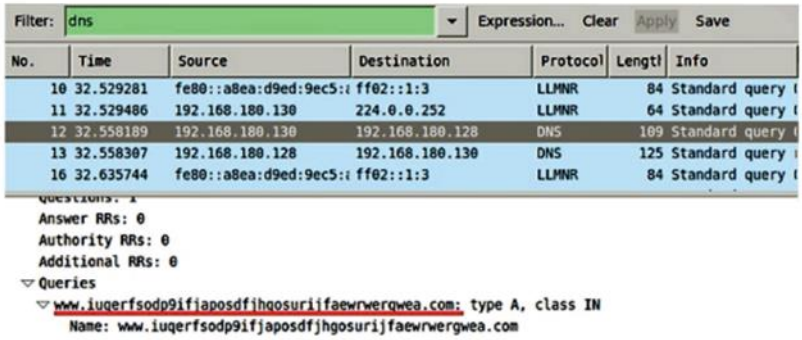


**Fig. 2.** The malicious DNS request was captured using Wireshark [6]

Throughout the live analysis, it was observed that upon initialization, the worm component makes an attempt to establish a connection with a specific domain. This connection is initiated using the InternetOpenUrl function, which is commonly used for accessing resources over the internet. The domain in question is identified as “www.iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com” [6], and is suspected to be associated with the spread of the malware.

The examination carried out the worm component via runtime execution revealed that upon initialization, the component attempts to establish communication with a specific domain utilizing the InternetOpenUrl function. The mentioned domain serves as a kill-switch domain, meaning that if the domain is active, the worm component terminates its execution. Conversely, if the worm component fails to establish a connection with this domain, it continues to run and installs itself as the "Microsoft Security Center Service (2.0)" process mssecsvs2.0 on the compromised system [6]. Therefore, the existence of this kill-switch domain can be employed as a detection technique when developing a defense mechanism against WannaCry.

The malicious DNS request on port 80 was captured by the FakeDNS utility of REMnux, as illustrated in Fig. 1. In addition, Fig. 2 depicts the query field of DNS packets sent from the infected machine (IP 192.168.180.130) to the DNS server on REMnux (IP 192.168.180.128), as observed through Wireshark [6].

Upon failure to connect to the kill-switch domain, the WannaCry worm component initiates the creation of an mssecsvs2.0 process with DisplayName of "Microsoft Security Center (2.0) Service".

This event is visible in the Process Hacker tool, which displays the process with a PID of 4016 (Fig. 3). Furthermore, the malware extracts the hardcoded R resource binary, which represents the WannaCry encryption component binary, and copies it to the "C:\Windows\taskche.exe" directory path. The worm subsequently launches the executable with the command line parameters "C:\Windows\taskche.exe/i". Additionally, it attempts to move the file "C:\Windows\taskche.exe" from "C:\Windows\qeriuwjhrf" to replace the original file, if it exists. This is executed to enable multiple infections and to circumvent any issues related to tasksche.exe process creation [6].


**Fig. 3.** Microsoft Security Center (2.0) Service [6]

At the end of its execution, WannaCry creates a persistent mechanism in the Windows registry to ensure that the malware runs automatically after every system restart. This involves generating a unique string, such as "midtxzggq900", using the computer name and storing it as a new entry in the registry. After this step, the malware copies itself to a randomly named folder within the Common Appdata directory on the infected computer. Finally, WannaCry attempts to establish memory persistence by adding itself to the AutoRun feature, thus ensuring that it executes every time the system boots up [6].

```
Created         C:\ProgramData\midtxzggq900\b.wnry
Modifed 15F936  C:\ProgramData\midtxzggq900\b.wnry
Created         C:\ProgramData\midtxzggq900\c.wnry
Modifed 30C     C:\ProgramData\midtxzggq900\c.wnry
Created         C:\ProgramData\midtxzggq900\msg
Created         C:\ProgramData\midtxzggq900\msg\m_bulgarian.wnry
Modifed         C:\ProgramData\midtxzggq900\msg
Modifed BB07    C:\ProgramData\midtxzggq900\msg\m_bulgarian.wnry
Created         C:\ProgramData\midtxzggq900\msg\m_chinese (simplified).wnry
Modifed D457    C:\ProgramData\midtxzggq900\msg\m_chinese (simplified).wnry
Created         C:\ProgramData\midtxzggq900\msg\m_chinese (traditional).wnry
Modifed 135F2   C:\ProgramData\midtxzggq900\msg\m_chinese (traditional).wnry
```

**Fig. 4.** WannaCry is known to launch files in the working directory [6]


**Fig. 5.** Ransom [6]

The comprehensive dynamic analysis of the WannaCry ransomware was performed in a specially designed virtual testbed [6].

## 4. Conclusion

The study revealed that WannaCry is composed of two distinct components that work together to enable the worm-like self-propagation mechanism and the combined encryption process. The

worm component is responsible for spreading the ransomware to other vulnerable systems on the network, while the encryption component is responsible for encrypting the files on the infected machine.

The evaluation implemented on the WannaCry ransomware provided an in-depth understanding of the technical aspects of the malware, exposing its intricate structure and intricate functionality. This analysis demonstrated the ability of the malware to propagate quickly and effectively, infecting multiple systems and causing significant damage. Moreover, it highlighted the importance of developing and implementing robust defense mechanisms to prevent the spread of similar threats in the future. Overall, the results of this analysis provided valuable insights into the workings of WannaCry, emphasizing the need for continued research and development in the field of cybersecurity to mitigate the impact of such malicious attacks.

The SEAP platform was chosen as the focus of this study due to its potential vulnerability to a WannaCry attack. The analysis aimed to investigate the specific aspects of the attack, including the process of infection, its mechanism for persistence, encryption, and prevention of recovery. Additionally, the study examined the methods by which the ransomware propagated and communicated with its command and control servers. The findings of the analysis were essential in identifying the characteristics and behaviors of the WannaCry ransomware, which have implications for the development of effective defense mechanisms and incident response strategies.

## References

[1]. www.e-licitatie.ro.

[2]. A. Palisse et al., “Ransomware and the Legacy Crypto API”, The 11th International Conference on Risks and Security of Internet and Systems. 5th-7th September 2016 (Roscoff, France: Springer).

[3]. G. Hull, H. John, and B. Arief, “Ransomware deployment methods and analysis: views from a predictive model and human responses”, Crime Science, vol. 8, no. 1, 2019; K. Savage, P. Coogan, and H. Lau, The evolution of ransomware. Symantec, 2015.

[4]. BleepingComputer (2016). Locky Ransomware Information, Help Guide, and FAQ, https://www.bleepingcomputer.com/virus-removal/lockyransomware-information-help.

[5]. L. Abrams (2014). CryptoDefense and How_Decrypt Ransomware Information Guide and FAQ, https://www.bleepingcomputer.com/virusremoval/cryptodefense-ransomware -information; Webroot (2017). MSP Guide: Stopping Crypto Ransomware Infections in SMBs, 16 Easy Actions for MSPs, White Paper.

[6]. WannaCry Ransomware: Analysis of Infection, Persistence, Recovery Prevention and Propagation Mechanisms Maxat Akbanov, Vassilios G. Vassilakis, and Michael D. Logothetis.

[7]. D. O’Brien, “Ransomware 2017”, Internet Security Threat Report, Symantec, July 2017 [Online]. Available: https://www.symantec.com/content/dam/symantec/docs/security-center/white-papers/istr-ransomware-2017-en.pdf.

[8]. K. Savage, P. Coogan, and H. Lau, “The evolution of ransomware”, Security Response, Symantec, June 2015 [Online]. Available: http://www.symantec.com/content/en/us/ enterprise/media/securityresponse/whitepapers/the-evolution-of-ransomware.pdf.

[9]. Pestudio, Malware Assessment Tool [Online]. Available: https://www.winitor.com.

[10]. K. Cabaj and W. Mazurczyk, “Using software-defined networking for ransomware mitigation: The case of CryptoWall”, IEEE Network, vol. 30, no. 6, pp. 14–20, 2016.