Transilvania University of Brasov

FACULTY OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

Study programme:
Cyber Security - Master

DISSERTATION PROJECT

Student: Mihaiu Iulia-Ana-Maria
Scientific coordinator: Conf. dr. ing. Bălan Titus

Brașov, 2022

Department: Electronică și calculatoare
Study programme: Securitate cibernetică (lb. engleză) - Master

MIHAIU Iulia-Ana-Maria

Ransomware in Healthcare: Analysis of Ryuk and Conti

Scientific coordinator:

Conf. dr. ing. BĂLAN Titus

BRAȘOV, 2022

Abstract

What happens if a hospital shuts down and cannot treat a patient? With the rise of ransomware, particularly ransomware-as-a-service, an industry as critical as healthcare is being heavily targeted (hospitals, pharmacies, supply chains, medical clinics), compromising patients' care. The healthcare industry is a suitable target due to the heavy dependency on technical infrastructure (web applications, mobile applications, cloud services) for them to operate when sensitive data, such as personal records, appointments, imaging diagnoses, and others, are often managed without adequate cybersecurity practices or backups.

In addition, hospitals often lack the necessary security protocols to ward off any ransomware attack and deal poorly with the consequences. Recent studies show that educating an organization's employees, the most common attack vector for ransomware, is crucial to delivering a higher degree of protection.

We perform a static and dynamic analysis of ransomware Ryuk and ransomware as a service Conti. Furthermore, we implement Ransomware4Students, an innocuous but illustrative proof of concept, to showcase and educate on how ransomware works. We present a use case of the application of the studied ransomware in healthcare and provide a set of recommendations to mitigate their attacks. By educating end-users and employees, we hope that the healthcare industry can have a higher degree of resiliency when faced with a ransomware threat.

Keywords: ransomware, hybrid analysis, Conti, Ryuk, healthcare

Rezumat

Ce se întâmplă dacă un spital se închide și nu poate trata un pacient? Odată cu dezvoltarea Ransomware-ului, în special a Ransomware-as-a-Service, industriile importante, chiar vitale, precum cea medicală (spitale, farmacii, lanțuri de aprovizionare, clinici medicale) sunt puternic vizate/afectate, punând în pericol viața pacienților. Sectorul medical este o țintă potrivită pentru atacatori datorită dependenței de infrastructura tehnică (aplicații web, aplicații mobile, servicii cloud) necesară funcționării lor atunci când datele sensibile (înregistrările personale, programările, diagnosticele imagistice și altele) sunt adesea gestionate fără practici adecvate de securitate cibernetică sau back-up.

În plus, deseori, spitalele nu dețin protocoalele de securitate necesare evitării oricărui atac ransomware și nu fac față consecințelor. Studii recente arată că educarea angajaților unei organizații - cel mai comun vector de atac pentru ransomware - este crucială pentru a oferi un grad mai ridicat de protecție.

În această lucrare este efectuată analiza statică și dinamică a Ransomware-ului Ryuk și a Ransomware-as-a-Service Conti. Mai departe, este realizată implementarea unui proof of concept, Ransomware4Students, cu scopul ilustrării întru-un mod educativ a metodelor folosite într-un atac de tip ransomware și a modului de funcționare a unei aplicații ransomware. De asemenea, este prezentat un studiu de caz asupra unui atac ransomware din sectorul sănătății, urmat de o listă de recomandări pentru mitigarea acestui tip de atac. Prin educarea utilizatorilor și a angajaților sperăm că sectorul medical poate dezvolta un grad mai mare de rezistență în cazul confruntării cu o amenințare de tipul ransomware.

Contents

Chapter 1

Introduction

Ransomware is predicted to cause $265 billion of damage per year by 2031, the equivalent gross domestic product of a country like Romania [1]. Ransomware is malware becoming more and more present in today's society. Malware, a contraction for "malicious software," is software with the intent to cause prejudice to computers, be it destroying the computer, using the computer for non-authorized purposes (such as using its computing power), and obtaining illegal financial gain [2].

Ransomware as we know it appeared around 1996, when two researchers from Columbia University presented, at the 1996 IEEE Security & Privacy conference, a file-encrypting program [3]. Perceived as a threat to gain much momentum, ransomware is software that encrypts data from a victim and demands a payment to decrypt such data, typically in cryptocurrencies [4] such as Bitcoin [5]. Since its inception, ransomware has given birth to a new industry, where attackers organize to exploit victims [6]. Upon its creation, ransomware attacks were primarily used to target individuals; however, more recently, criminals target the networked data to which computers have access, thus transforming every employee at every organization connected to the internet into a possible attack vector. Criminals attempt to gain access to an employee's machine and then perpetrate the attack at the organizational level. Educating employees to follow best practices does not seem sufficient to avoid ransomware attacks, but it surely helps prevent and recover from them [7, 8].

The fact that there is great economic opportunity for hackers, ease of access to information technology education, and ease of obtaining payments makes ransomware development an appealing activity for cybercriminals, with hundreds of thousands of new variants of different ransomware software being developed per day [9]. According to the US Department of Health and Human Services Office for Civil Rights Breach Portal [10], which displays a list of breaches of unsecured protected health information affecting 500 or more individuals, more than 869 thousand breaches have been reported (in the form of improper disposal of information, loss, theft, unauthorized access, or others). Such breaches can generate leads for the criminals to target (for example, the e-mails of the hospital employees) [10].

Ransomware is sent to victims via e-mails that try to impersonate the victim's contacts. These e-mails may contain a URL to the malicious software or an infected attachment, such as a Word document containing macros or an executable. Once the victim engages with the component containing the payload, the malware is downloaded. It infects the machine, possibly executing a series of steps, including but not limited to searching the hard drive, network files, external drives, and cloud drives for all data that can be encrypted, encrypting the data, and demanding payment for the decryption key.

In the course of a few years ransomware alone was responsible for more than $50 billion in damage across several sectors [8]. The year 2021 set an unprecedented trend in exploiting this malware on companies, and predictions set the trend to keep rising sharply [1]. Figure 1.1 shows the recent trends on the raising ransomware-caused damage.

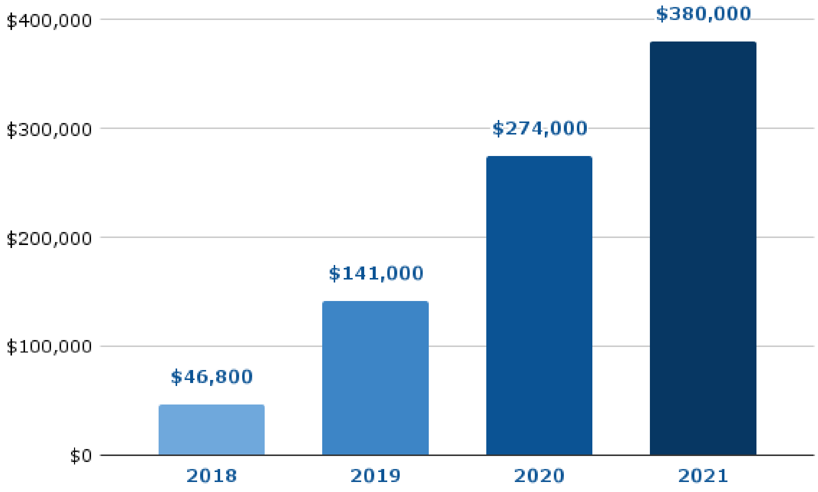


Figure 1.1: Average cost of downtime caused by ransomware from 2018 to 2021 [11]

An industry particularly affected by the rise in ransomware is healthcare. Healthcare is the industry that includes but not limited to hospitals or medical clinics, healthcare industry services, pharmaceutical, and hospices.

Healthcare is a susceptible industry because the operationally of health systems have life or death consequences, and thus any downtime may be critical for a patient's survival. According to a recent report, [12, 13], ransomware attacks on healthcare organizations increased 94 percent in 2021, leading to an average cost of recovery of $1.85 million, and averaging one week to recover. About 65% of healthcare organizations pay the ransom, indicating that there is still a need to educate professionals and provide the tools to mitigate ransom attacks.

The healthcare industry and hospitals in specific have become an attractive target for criminals for several reasons: (1) healthcare generates many data (e.g., electronic medical records, diagnostic imaging records, core directives, medical history, and others) [14] and (2) the security vulnerabilities in IT systems and networks, that are frequently outdated [15], and (3) the lack of training and security-awareness education of the employees, including IT staff, making them susceptible to techniques such as social engineering [16, 9].

In this document, we explore the usage of the emerging Ransomware CONTI, and its predecessor Ryuk. Ryuk is a ransomware family, first discovered in the wild in August 2018 [17]. CONTI [18] was first observed in May 2020 and is an improvement over Ryuk. CONTI has a very significant market share in the ransomware scene (second highest by beginning of 2021 [19]). Both target the healthcare industry, causing millions of sensitive data to be disclosed, billions of dollars of damage, and even deaths [20].

Although some educational resources exist to help mitigate ransomware [21, 22], there are few resources aimed at providing such recommendations to the healthcare industry, focusing on ransomware-as-a-service.

1.1 Research Questions and Contributions

In this thesis, we will explore the impact of several ransomware-as-a-service in the industry of healthcare, namely CONTI and Ryuk, by conducting static analysis of the mentioned ransomware. After that, we take several measures to educate healthcare providers to reduce the impact of the studied ransomware. In particular, we pose the following research questions (RQ):

- RQ1: How do CONTI and Ryuk behave?
- RQ2: How to alleviate the impact of the CONTI and Ryuk ransomware families through education?

We conduct a hybrid analysis (static analysis supported by dynamic analysis) to answer the first research question. Using static analysis techniques, such as analyzing the ransomware binary, we characterize CONTI and Ryuk and create our comparison framework.

To address the second research, we conduct the implementation of a ransomware tool, which we call Ransomware4Students. Ransomware4Students is a proof of concept that showcases how a ransomware attack is conducted, intending to bring awareness-raising to people susceptible to being attack vectors for the ransomware. We deploy our implementation in two virtual machines (VM), one Kali machine simulating the attacker and one Windows 10 VM simulating the victim. We explain the phases of attack preparation, intrusion, and ransomware infection. Following the demo, we propose a list of actionable recommendations for preventing the infection from ransomware, in particular CONTI and Ryuk.

1.2 Thesis Outline

The thesis is organized as follows: in the second chapter, we present the related work, including work done on analyzing Ryuk, Conti, and ransomware in healthcare. After that, we present the necessary background for the reader to comprehend the work on this thesis in Chapter 3. Next, in Chapter 4, we conduct static analysis of Ryuk (Chapter 4.1) and Conti (Chapter 4.2). After that, we discuss the similarities and differences between both. To finalize the chapter, we introduce Ransomware4Students. Chapter 5 presents a case study illustrating how Conti was deployed in the Irish healthcare system. In the sixth chapter, we provide recommendations to mitigate ransomware attacks. Finally, we conclude this thesis.

Chapter 2

Related work

In this section, we discuss the related work. We compare the existing literature, including academic papers, blog posts, theses, books, white papers, and technical reports to the output of this thesis. To do so, we used Google Scholar. Google Scholar is a modern search engine that indexes the major online libraries.

Static Analysis

In this section, we identify related studies that performed a static ransomware analysis. Many papers explain and explore different static analysis techniques, such as directed to the Android operating system [23, 24, 25, 26, 27, 28], using machine learning [29, 30, 31, 32, 33], discussing tradeoffs between static and dynamic analysis [34, 35, 36]. However, few studies depict static analysis of Conti and Ryuk. While some works include descriptions of analysis of the studied ransomware, see for example [37, 18], we focus on cases directed to the healthcare domain. To narrow down our search, in Google Scholar, we used the search keys "conti ransomware healthcare and ryuk ransomware healthcare. The first query returns 154 results and the second 39 results. Due to the very low number of results, our search process finished when we read the papers' title and abstract. If relevant, we include them in this section.

Ryuk Analysis in Healthcare

In [38], the authors explain the case of the Benesov hospital, a Czech hospital that Ryuk hit in 2019. The authors characterize the organizational factors of the cyberattack and put them into perspective against lessons learned. This is a cheerful case where "ransom was not paid, no data were lost, key systems were restored within a week, and the security incident led to a great improvement in the IT security), which might be used by other IT or security specialists." While this article focuses on Ryuk's application in the healthcare scenario and provides measures to mitigate attacks, it does not cover Ryuk from a technical perspective. Another study [39] covers cyberattacks on the Benesov hospital, providing a qualitative analysis and a set of "minimal cyber security standards."

In [37], the authors analyze the source of Ryuk-based infections are based on network traffic logs. However, there is no static analysis of Ryuk, nor recommendations to organization employees.

In [40], the authors analyze data breaches in healthcare and provide recommendations for improving the current security standards to avoid data breaches. Ryuk is the ransomware used in the "UVM Healthcare Breach," where phone and e-mail networks were taken down, and patient records and appointments were eliminated, leading to damage of around $60 million. This study covers a use case but does not analyze Ryuk technically. In [41], the authors analyze portable executables of ransomware samples. Although mentioning Ryuk, the authors do not analyze it. Many other references, e.g., [42, 43, 44, 45, 46, 47] illustrate the impact of ransomware in the healthcare industry, introducing Ryuk shortly. However, these studies do not explain Ryuk's application in a practical context or do any further analysis.

We conclude that there is minimal literature, to the best of our knowledge, that technically analyses Ryuk in the context of healthcare, delivering material that attempts to mitigate the attacks. Moreover, most work done is from 2021, revealing that studies on this ransomware are very recent.

Conti Analysis in Healthcare

In [48], the authors explore the deployment of Conti on Ireland's Health Services Executive, bringing estimated damage of $100 million. The authors explore Conti from a technical perspective and explore the use case. However, no recommendations are given to mitigate future attacks. Two more studies refer this incident [49, 50], but they are less detailed or focus on other aspects [51, 52, 53]. In [54], the authors analyze Conti from a technical perspective and present the impact of negotiation in reducing the amount of ransom to pay. However, there is no mention of the healthcare industry.

In [18], the Cybersecurity and Infrastructure Security Agency (CISA) and the Federal Bureau of Investigation (FBI) provide a list of recommendations to mitigate the impact of the CONTI ransomware, including but not limited to requiring multifactor authentication (MFA), implementing network segmentation, and keeping operating systems and software up to date. In [55], the authors collect and analyze data on ransomware data (e.g., payment, target). The authors show that the ransomware as a Service (RaaS) is a highly lucrative market, being Conti the second-highest revenue generator across the examined ransomware. No technical analysis, use case, or recommendations are provided.

We hypothesize that there is very little related work because Conti and Ryuk are very recent programs (especially the primer). There are lots of different variations being developed constantly, making versioning and tracking difficult. The following table summarizes the related work and how this thesis compares. In

Paper	Ryuk	Ryuk-T	Conti	Conti-T	Use Case	Actions
[38]	✓	✓	✗	✗	✓	✓
[37]	✓	✓	✗	✗	✗	✗
[40]	✓	✗	✗	✗	✓	✓
[41]	✓	✗	✗	✗	✗	✗
[48]	✗	✗	✓	✓	✓	✗
[54]	✗	✗	✓	✓	✗	✓
[18]	✗	✗	✓	✓	✗	✓
[55]	✗	✗	✓	✗	✗	✗
This thesis	✓	✓	✓	✓	✓	✓

Table 2.1: Comparison of the identified related work. The checkmark (✓) means that criteria is satisfied and the cross (✗) means otherwise.

the columns, we have comparison criteria: does the work informally describe Ryuk (Ryuk); does the work describe Ryuk technically (Ryuk-T); does the work informally describe Conti (Conti); does the work describe Conti technically (Conti-T); does the work relates a situation/use case where Ryuk or Conti was utilized (use case), and finally, does the work provide a list of actions to mitigate an attack (actions)?

From the analysis of Table 2.1, we conclude that our work fills a gap in the existing literature.

Chapter 3

Background

In this section, we introduce the background.

3.1 Ransomware

Ransomware is ”characterized by widespread targeting, fixed ransom demands, and technically-adept operators” [55]. Typically, ransomware operators take advantage of existing versions, tweaking it and causing the formation of a ransomware family.

1. Preparation: first, the criminal might do the groundwork on how to target a victim. This includes obtaining contacts, studying interaction patterns, obtaining intelligence about security practices on the target organization, and so on.

2. Infection: the attack includes social engineering tactics to increase engagement. The internet is the preferred means to initiate an infection. E-mails are often the initial attack vector. The victim typically is redirected to an infected website or downloads an infected attachment containing the ransomware payload.

3. Encryption. After the infection, the ransomware establishes a connection to a server operated by criminals, the so-called command and control (C&C) server, and retrieves the necessary encryption keys to perform the data encryption. Typically, data is encrypted using RSA or AES, which we explain later in this section. The wide availability of cryptography libraries in different programming languages lowers the technical knowledge barrier for criminals.

4. Extortion and Payment. After the encryption, ransomware operators display a ransom notice that stipulates the value to be paid. In exchange, criminals promise to decrypt the victims’ data, but that is not always the

case. Finally, the victim may pay the ransom, typically in cryptocurrencies, to ease laundering.

The typical mitigation measures are having backups and a contingency plan. The philosophy of having backups is that if an attack is successfully executed, the data lost/damage extension is limited to the date of the latest backup. Having backups is, in fact, a crucial good practice that all enterprises should adopt. However, it turns out that many organizations do not have a backup plan in place and are vulnerable to ransomware, having two options: to lose the data or to pay the ransom and have the possibility to recover it (since there is no guarantee that the data will be accessible again).

Ryuk is "one of the first ransomware families to include the ability to identify and encrypt network drives and resources" [17], and to disable shadow copies. In practice, it forces users to use backups as the only way to mitigate the attack or to pay a ransom.

First, Ryuk attempts to encrypt all files on the disk, including backup files. After that, it attempts to end all the security services linked with the device, including the built-in Windows Automatic Startup Repair, and changes the boot status policy to ignore all failures. Lastly, a ransom (typically on cryptocurrencies) is required for a decryption key, typically shown as RyukReadMe.html, to recover the files. After the victim's assets are encrypted, a ransom notice is displayed. Optionally (typically happening in RaaS), there is a chat support negotiation with the ransomware operator. When a ransom price has been accorded, the victim obtains Bitcoin or other cryptocurrency and is sent to the criminal. The criminal attempts to launder the received money and, finally, cashes out.

Cryptocurrencies remain the payment method of choice for criminal ransomware actors. While many cryptocurrencies exist, each typically supported by a blockchain, Bitcoin is the preferred one due to its large market capitalization [5]. Bitcoin has gathered a reputation as a proper medium of exchange and store of value. Those characteristics allied to the fact that identities are pseudonymous make them attractive to criminals. However, Bitcoin is transparent, meaning that all transactions on the network can be verified and analyzed. If a criminal links their Bitcoin account with, for example, a cryptocurrency exchange, their true identity might be revealed [55].

3.2 Ransomware-as-a-Service

Ransomware as a service (RaaS) is ransomware sold by creators to criminals and makes revenue via a commission from each attack (or a flat monthly fee). RaaS became widespread in 2019, capturing a large share of the ransomware market [6]. They often provide a payment portal, where victims contact the ransomware operators to obtain payment information and support getting their data back. Darkside is one of the most notable RaaS families whose affiliates were responsible for the Colonial Pipeline attack in 2021 [56]. Since ransomware operations do not need the technical expertise to develop sophisticated software, RaaS has a very low barrier to entry, yielding a lucrative scheme.

Conti is a famous RaaS that has been observed since 2020. Conti is believed to be distributed by a Russia-based group,

3.3 Malware Code Analysis

There are two main techniques for malware analysis: static analysis and dynamic analysis [35]. Static analysis is the process of analyzing the code of a program without running the executable. Static analysis can cover the entire code and possibly capture the program's behavior. However, it might be challenging to capture the whole logic if the studied program is extensive and has numerous execution paths. Dynamic analysis requires running malware in a controlled environment to analyze its behavior. Dynamic analysis is a necessary complement to static analysis because, for example, it can capture the most common execution paths in malware. This thesis uses a hybrid approach, mostly focused on static analysis.

3.4 Applications of Cryptography to Ransomware

Ransomware relies on cryptography. In this section, we explain how is cryptography (encryption and also hashing) important to analyse ransomware. We assume the reader understands how public-private key cryptography and hashing works.

Encryption is used to obfuscate information between parties [57]. To that end, ciphers are used. A cipher is an algorithm that is used to encrypt information. A message (or, more generally, information) is encrypted using a key as the parameter of the cipher. The same key can be used to decrypt the message (in symmetric key cryptography) or a different one (in asymmetric key cryptography). Ransomware criminals use encryption algorithms to obfuscate victims' data, leaving victims unable to read or use their data. The end goal is to provide the decryption key in exchange for the ransom. Examples of symmetric cryptography algorithms include the data. Encryption Standard (DES), the Advanced Encryption Standard (AES). Typical examples of asymmetric cryptography include elliptic curve algorithms, Diffie. Hellman, and RSA [57].

Hash functions, such as SHA-256 or MD5 [57] allow the creation of a fixed-length digest from an input (for example, a ransomware executable). As hash functions are one-direction and tamper-resistant, they allow us to take the "signature" of a ransomware executable and thus identify each variant.

Chapter 4

Ransomware Analysis

In this section, a malware analysis is conducted on the Ryuk Ransomware and the Ransomware-as-a-Service Conti. In the analysis of the malware samples I used both a manual approach and an automatic one. I started with the Intezer Analyzer platform [58] for automatic analysis of files and also the automatic reports from Joe Sandbox[59] for getting an overview of the malware samples capabilities. Furthermore, I manually analyzed the malware samples on a Windows virtual machine on which I installed the Flare-VM[60] security distribution provided by Mandiant for reverse engineering, penetration testing and malware analysis.

4.1 Ryuk analysis

Part of the analysis were the samples shown in table 4.1, which were further placed in two different clusters: before 2021 (referred as Ryuk-Group1) and after 2021 (referred as Ryuk-Group2).

Date	Group	SHA256
August, 2018	Ryuk-Group1	98ece6bcafa296326654db862140520 afc19cfa0b4a76a5950deedb2618097ab
April, 2021	Ryuk-Group2	8f368b029a3a5517cb133529274834585 d087a2d3a5875d03ea38e5774019c8a
August, 2018	Ryuk-Group1	8d3f68b16f0710f858d8c1d2c699260e6 f43161a5510abb0e7ba567bd72c965b
February, 2021	Ryuk-Group2	2820bf6bb2a2d0670c6233301b414ae2 93cf85536127ac63b2ebd07d8616ae89

Table 4.1: Ryuk samples analyzed

4.1.1 Initial access

The most common infection vector for a Ryuk campaign is the using of targeted phishing attacks, sometimes alongside fake call center campaigns, also known as BazaCalls. The attackers are sending tailored emails to employees of the targeted company. Often, the email has an infected attachment, waiting to be opened by the user. When the attachment is open (typically a document with an embedded script in the macro section) it prompts the user to enable the macro for seeing the whole content of it. If the macros are enabled, the malicious script is run.

In the Ryuk campaigns there are two common infection chains that are using malware droppers to get a foothold on the targeted machine and further deploying the actual ransomware.

The first infection chain implies the download of TrickBot (or Emotet, which then distributes TrickBot), a well-known credential-harvesting banking Trojan targeting Windows machines, now used as a malware loader. TrickBot is used by the attackers to move laterally within the system, steal sensitive information and distribute legitimate tools for post-exploitation if the infected system seems to be an industry target. Once the attacker compromised the system and exfiltrated data, the ransomware can be deployed.

The second infection chain implies the retrieving and running of the BazarLoader payload. Once the payload is run, a connection is created to a Command-and-Control (C&C) server. BazarBackdoor is sent through the network traffic from the server and it is installed on the targeted system. Having a foothold, post-exploitation tools can be dropped on the compromised system, followed by the deployment of the actual ransomware.

Both of these infection chains have CobaltStrike as a common post-exploitation tool. CobaltStrike is a paid penetration testing tool, initially built for ethical hackers, but nowadays used by many cybercriminal groups to launch complex attacks. The tool is distributed on the compromised system as an agent, called beacon, which runs in memory and can be executed remote from the C&C server. The beacon has a multitude of capabilities including, but not limited to, privilege escalation, lateral movement, Mimikatz and file transfer.

The threat actors also uses tools like Bloodhound and AdFind - to enumerate the local domains, network shares and Active Directory(AD), LaZagne - to steal passwords stored locally and Kerberos attacks in order to harvest hashes from the AD.

Spreading mechanism

- Manually

  The samples from Ryuk-Group1 do not show any functionality of moving laterally through the network without human interaction, thus they require a **dropper** and then manually operation from the attackers [61].

A dropper is a program that helps with the dissemination and installation of the malware. It usually has embedded several binaries of the same ransomware so it can be used on systems with different configurations. Once run, the dropper can automatically check for the architecture of the system (checking if a x86 or a x64 binary is needed) and based on the version of the operating system, it drops the ransomware either in C:\Documents and Setting\Default folder, for old systems (e.g. Windows XP, Windows 2000), or in the C:\Users\Public folder for the new systems.

- Automatically

  The Ryuk-Group2 showed a **"worm-like"** functionality, currently working only on Windows systems [61]. A computer worm is a malware application that can disseminate copies of itself on multiple systems and propagate across a network through Internet or Local Area Network (LAN) connection without human interactions. Using the worm-like ability, the samples from Ryuk-Group2 can copy themselves in the current directory. The copies follow a naming convention (e.g. for the replication copy concatenating the checksum of the current username with "r.exe" or using the default name "rep.exe") and each of them have a different functionality that is executed through a command line. One of the copies is responsible for sending Wake-On-Land packets, trying to turn on the hosts in the compromised network, while the other is used for self-replication on the targeted systems[62, 63].

4.1.2 Ransomware binops

Imports

advapi32.dll, shel32.dll, kernel32.dll, ole32.dll, mpr.dll, msvcrt.dll, rpcrt4.dll, gdi32.dll sspicli.dll, shlwapi.dll, win32u.dll, ntdll.dll, Iphlpapi.dll, cryptbase.dll, bcrypt-primitives.dll.

Mutexes

In some of the samples from Ryuk-Group2 there are used mutexes[63]. When the copy responsible for self-replication is executed, it first tries to create a mutex with the username of the targeted machine and if the mutex already exists (meaning the machine was infected before), the process will stop, thus the system will not be reinfected.

Anti-reverse engineering techniques

Return Pointer Abuse

In [63] it was identified a part of code that takes advantage of the return pointer in order to make the analysis of the code more complex. A ret instruction is a combination of a pop instruction and a jmp instruction[64]. Putting the ret instruction in the middle of a function leads to either the disassembler not being able to show the target code to which the program jumped to, or the disassembler terminating the function.

Anti-debugging techniques

In order to avoid the code being debugged, Ryuk uses Windows API functions calls (Native API - T1106) and also manually checking system structures (System checks - T1497.001) [63].

- Windows API calls

  IsDebuggerPresent API function - checks the value of the flag BeingDebugged from the Process Environment Block (PEB) structure; if the value is zero, it means no debugger is attached to the process.

  NtQueryInformationProcess API function - retrieves information about the specified process [65]; The function has a parameter called ProcessInformationClass which can retrieve the value of the queried flags. Ryuk's code is checking for the following flags:

  - ProcessDebugFlags - if it returns the value 0, meaning the program is being debugged, the program will immediately crash

  - ProcessDebugPort - it first checks if it has any value, if yes, then it substitutes the ProcessInformation with the value zero, and on a following check, thanks to that value it will crash the program

  - ProcessDebugObjectHandle - for each debug process there is created a debug object, if such object exists for the program, then the program will crash

  - ProcessBasicInformation - it has as attributes a UniqueProcessId and an InheritedFromUniqueProcessId, given these, the name of the parent processes can be recursevely check against a list of popular debugger tools' names; if such name is found, the program will crash.

- Manually checking system structures

  Checking the BeingDebugged Flag

  The PEB structure contains all the user-mode parameters associated with a process, including the debugger status[64]. The malware tries to confuse the analyist by spliting the execution of the manual check in multiple functions and then jumping from one to another[63]. First, it copies the contents from the PEB referenced location (fs:[30h]) into the EAX data register, then it puts the value of the BeingDebugged (EAX+2) into a variable, and at last it checks if the value of the variable is zero. Depending on the value, the malware does a conditional jump which determines the code path.

Anti-Virtual Machine Techniques

Time based evasion

Ryuk uses the NtDelayExecution Windows native API to delay the execution of the malware.

Process white/blacklist

The samples have a predefined list of 41 processes and 64 services [63, 62]. The malware start a new thread for enumerating the running processes and services and it checks them against the before mentioned list. It then uses the function ShellExecuteA function to start several instances of conhost.exe, taskkill.exe and net1.exe , attempting to stop the services and kill or put to sleep for some time the running processes that are mostly belonging to antivirus products (e.g. stop "Sophos Safestore Service" /y, stop "McAfeeEngineService" /y), databases (e.g. /IM "mysqld.exe" /F), backup products (e.g. stop "Veeam Backup Catalog Data Service" /y, "zoolz.exe") or file editing tools (e.g. /IM "powerpnt.exe" /F, /IM "wordpad.exe" /F)[58].

Deletion of back-ups and shadow copies

Upon execution, Ryuk creates a "windows.bat" file in "C:\Users\Public". The batch script file contains commands for deleting shadow copies using the vssadmin tool (e.g. "vssadmin Delete Shadows /all /quiet"[58]) and also using the del command to delete multiple backup related files (e.g "del /s /f /q c:\*.VHD c:\*.bac c:\*.bak c:\*.wbcat c:\*.bkf c:\Backup*.* c:\backup*.* c:\*.set c:\*.win c:\*.dsk"[58]). After the script execution, the batch file deletes itself too, in order to not leave any trace.[66, 67, 62]

4.1.3 Persistence mechanism and privilege escalation

Registry Keys

Ryuk is achiving persistence by adding the actual malware executable to the startup folder. The malware either runs an instruction that overrides the registry key (Ryuk-Group1) or uses dynamically loaded registry functions, part of the Windows API (e.g. RegOpenKeyEx, RegSetValueEx, RegGetValue), in order to query and edit the registry values (Ryuk-Group2). The registry key on which the malware seem to focus is "HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Run\svchost", and by overriding the value of it, Ryuk is ensuring that the malware can automatically run after every reboot.

Access Token Manipulation

The malware is using the AdjustTokenPrivileges function to modify the privileges of the SeDebugPrivilege token. The before mentioned token allows the debugging and adjusting the memory of any process or thread regardless of the security descriptors, allowing the ransomware to gain full access to the system processes. This step paves the way for carrying out a defense evasion through Process Injection - T1055.

Change Drive Permissions

The malware is looking for all the mounted local drives using the GetLogicalDrives function. To grant full access to all of the identified drives, Ryuk is using the build-in Windows tool icacls with the needed flags[63].

Process Injection

First step in performing a code injection is to enumerate all the processes and store information about them in a list. The information for each process is stored as a structure containing: process type (0 - system process, 1 - custom process and 2 - inherited process), process ID and process name[66].

The next step is iterating over the list multiple times until the code is injected in the proper places. The injected code holds the core functionality used for the file encryption[67]. For each process it will use a basic direct injection technique[64]. First, it calls the OpenProcess() to get the handle on the target process and it will allocate a memory buffer in the process' address space via VirtualAllocEx(). The evil code will then be written in the allocated space through the WriteProcessMemory function and in order to ensure that the thread will run in the injected process, the malware creates a separate thread in the process using CreateRemoteThread().

4.1.4 Encryption scheme and key management

Pre-encryption

Importing libraries

The malware uses an API list of imports names that are obfuscated, in order to make the static analysis of the binary harder. The functions needed for the encryption process are decrypted using a predefined key[67] and then they are dynamically loaded using GetProcAdress() and LoadLibraryA(). Among the dynamically loaded libraries used in the file encryption process, we observed Windows CryptoAPI.

File discovery

The malware has two hard-coded lists of paths and filenames. Ryuk uses functions from kernell32.dll (e.g. FindFirstFileW(), FindNextFileW()) to enumerate the directories and files that are on the system and checks the details of the files against the lists before mentioned. Any file that corresponds to any entry of the lists will not be encrypted.

For the samples of Ryuk-Group2, the malware has one more verification. For files with "index." in the filename it will overwrite the contents with the html version of the ransom note, and for the ".php" files it will generate php code which will load the ransom note at every attempt of the user to access a website[63].

After the enumeration of the files, in order to speed up the encryption process, Ryuk will create a new thread for each to-be-encrypted file[66].

Network shares discovery

The ransomware has a similar approach regarding network shares discovery and encryption[66, 67, 63]. Ryuk uses functions from the mpr.dll (e.g. WNEtEnumResourceA(), WNetOpenEnumW()) to enumerate all the network share and stores them in a list. And for each of the entry in the list, it uses the steps that are explained in the below scheme.

Encryption scheme

Based on the analysis from [67], the encryption scheme of Ryuk has a similiar approach for all the variants. Ryuk uses a combination of symetric encrytpion (AES-256) [68] and asymmetric encryption (RSA-2056 or RSA-4096) [68, 63, 67], having a three-tier or three-layered RSA approach[67]. The encryption steps are explained below[69, 68].

First layer

1. The attacker holds a globally RSA pair of keys ($RSA_{Apub}; RSA_{Apriv}$).
2. The attacker embeds only the RSA_{Apub} key in the malware binary (usually found in the "UNIQUE_ID_DO_NOT_REMOVE").

Second layer

3. The ransomware generates for each victim an RSA key pair ($RSA_{Vpub}; RSA_{Vpriv}$).
4. RSA_{Vpriv} key is encrypted with the RSA_{Apub} key and stored locally on the victim's system (typically in a file named "PUBLIC" in the Windows directory).
5. The ransomware generates an AES_{file_i} for each to-be-encrypted $file_i$.

Third layer

6. The AES_{file_i} key is encrypted using the RSA_{Vpub} key.
7. A suffix ("HERMES" for Ryuk-Group1 or "RYUKTM" for Ryuk-Group2) is attached at the end of the file, together with the encrypted key from step 6.
8. After the encryption process of the file finished, Ryuk destroys the AES_{file_i} key.

Post-encryption

Scheduled tasks

In the samples from Ryuk-Group2, we can observe a new ability. Ryuk can schedule non-interactive tasks on the host machine and also on the remote systems from the network using the Windows tool schtasks.exe.

- On host

Ryuk schedules a task which daily prints 50 copies of the contents of the ransom note and the password for the contact form the victim has to fill up in order to get their files back. In figure 4.1 there's an example from [63] of the creation of such task.

```
"SCHTASKS /CREATE /NP /SC DAILY /TN "PrintvE" /TR "C:\Windows\
System32\cmd.exe /c for /l %x in (1,1,50) do start wordpad.exe
/p C:\users\Public\YTKkI.dll" /ST 10:25 /SD 05/18/2021 /ED
05/25/2021"
```

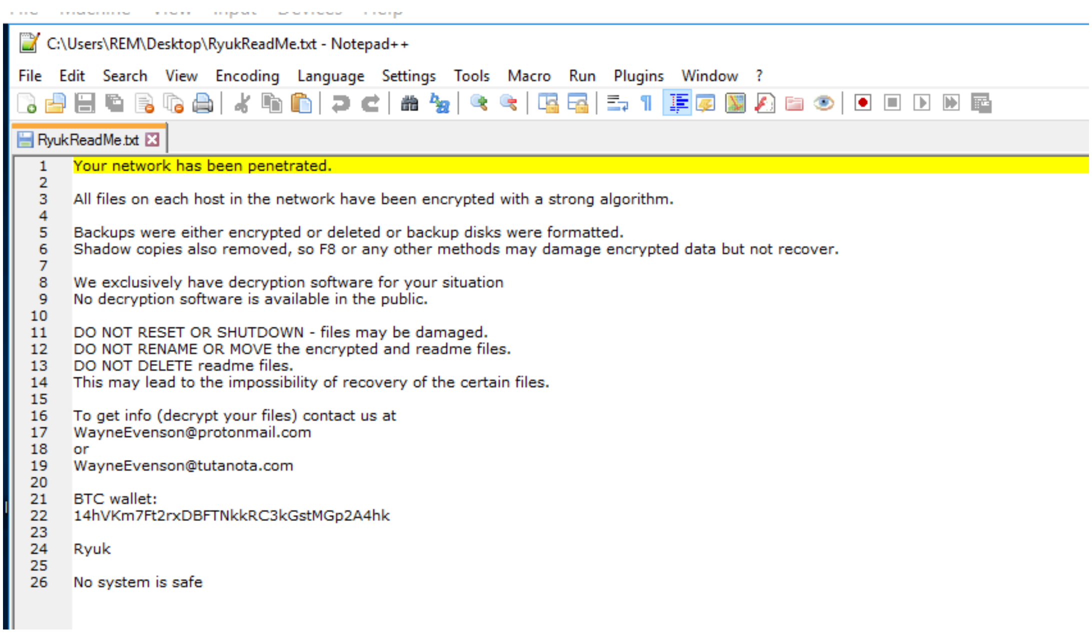
Figure 4.1: Command issued for a scheduled task

- On remote

Ryuk, thanks to the "worm-like" functionality, manages to first move laterally through the network to the hosts on which a RDP (Remote Desktop Protocol) connection can be created and copy itself on the machines using the SMB (Server Message Block) protocol. The scheduled tasks are created then for each host in order to start the ransomware infection at the time specified by the attackers.

4.1.5 Ransom Note

In the samples from the Ryuk-Group1, the ransom note is usually written to a "RyukReadMe.txt" file. The note contains the rules to be followed for getting back the files and information about the payment (unique contact e-mail addresses, BTC wallet address). An example of such ransom note can be seen in 4.2



```
Your network has been penetrated.

All files on each host in the network have been encrypted with a strong algorithm.

Backups were either encrypted or deleted or backup disks were formatted.
Shadow copies also removed, so F8 or any other methods may damage encrypted data but not recover.

We exclusively have decryption software for your situation
No decryption software is available in the public.

DO NOT RESET OR SHUTDOWN - files may be damaged.
DO NOT RENAME OR MOVE the encrypted and readme files.
DO NOT DELETE readme files.
This may lead to the impossibility of recovery of the certain files.

To get info (decrypt your files) contact us at
WayneEvenson@protonmail.com
or
WayneEvenson@tutanota.com

BTC wallet:
14hVKm7Ft2rxDBFTNkkRC3kGstMGp2A4hk

Ryuk

No system is safe
```

Figure 4.2: Ransom note from sample [1]

In the samples from Ryuk-Group2, the attackers removed the BTC wallet addresses[70] and even the e-mail addresses. Instead they have an html note with a contact button, guiding you to access a TOR link in order to fill the contact form[63]. The form asks you for your email, the organization, a password to identify the pawned system and there's also some space in which you can write a message to the attackers.

4.2 CONTI analysis

Part of the analysis were the samples from the below table 4.2. Additionally, I analysed the source code that was leaked this year by a Conti affiliate, that can be found here [71] (referred through the analysis as the ContiLeaks sample).

Date	Variant	SHA256
June, 2020	Conti-V2	1ef1ff8b1e81815d13bdd293554ddf8 b3e57490dd3ef4add7c2837ddc67f9c24
October, 2019	Conti-V1	94bdec109405050d31c2748fe3db32a3 57f554a441e0eae0af015e8b6461553e
September, 2020	Conti-V3	5d4b4d5adb2cd3fefd95b15725a77c4bf4 8e14e89a23b94733a9ec7b86e09ea2
November, 2020	Conti-V3	d3c75c5bc4ae087d547bd722bd8447 8ee6baf8c3355b930f26cc19777cd39d4c
February, 2021	Conti-V3	3cd91569a21a59a71582397b762633 808e0b413c78210ee48cefcffede66f356
December, 2021	Conti-V3	41324493142b10db127217274e2 1df37f6ccd13f01a8d29d2b23b7b1463423a7

Table 4.2: Conti samples analyzed

4.2.1 Initial access

The initial infection is usually starting with a spear phishing campaign. The attacker sends a tailored email, sometimes even from a legitimate account, asking for sensitive information about the user and/or attaching a malicious file to the email. The file is typically a Microsoft Office document with an embedded script in the macro section. When opening the file the user is prompted to enable the macros in the document in order to see its contents. On the enabling of the macros, the embedded script is executed, resulting in the downloading of malware loaders or droppers (e.g. TrickBot, BazarLoader and the newest addition Bumblebee) that will be further used by the cyber-criminals to do some initial reconnaissance on the target system, look for sensitive data and exfiltrate

[1] SHA256:98ece6bcafa296326654db862140520afc19cfa0b4a76a5950deedb2618097ab

it, followed by the download of post-exploitation tools and finally, the deployment of the ransomware.

Some other infection vectors used in the Conti infection chains are the use of stolen or weak RDP credentials and the exploitation of the common vulnerabilities found in the external assets. The ransomware group is known for exploiting vulnerabilities as follows:

- PrintNightmare (CVE-2021-34527) - remote code execution vulnerability affecting the Windows Print Spooler service;

- EternalBlue (CVE-2017-0144) - remote code execution vulnerability in the Windows Server Message BlockV1 (SMB) protocol, allowing access to network shares on remote systems;

- Netlogon or Zerologon (CVE-2020-1472) - elevation of privilege vulnerability in the Microsoft Netlogon Remote Protocol, allowing an unauthenticated attacker to impersonate a domain-joined computer, further being even able to obtain admin privileges;

- ProxyShell (CVE-2021-34473, CVE-2021-34523, and CVE-2021-31207) - pre-authenticated remote code execution vulnerability in the Microsoft Exchange Servers;

In the infection chain there are used post-exploitation tools like CoblatStrike beacon - to gather more information about the domain admin accounts, Kerberos attacks - to get admin hashes and trying to brute force them, RouterScan - to gather credentials of the routers in the network and also get information about the possible vulnerabilities they have. Furthermore, the threat actors will try to maintain an open C&C communication channel through the installation and persistence (using the registry run keys) of Remote Desktop software such as AnyDesk or Atera.

For the exfiltration of data, the RClone tool was frequently used alongside a MEGA account, to upload the sensitive data collected there before the publishing of it on the Conti site.

4.2.2 Ransomware binops

Spreading mechanism

Manually

Conti does not show any functionality of moving laterally through the network without human interaction, therefore the usage of a **dropper** or a loader is needed before the execution. After the ransomware is dropped on the targeted system, it is manually executed in memory via a command-line interface. This feature allows the attacker to have a better control on how to scan for data and how to choose the files to be encrypted. The ransomware can skip the encryption of the local files and target the network shares of the systems to which it can connect on the SMB port through TCP. The targeted IPs are typically the ones found in the ARP cache of the infected host machine.

The command line options for the arguments are explored in section 4.2.4.

Imports

kernel32.dll, ntdll.dll, kernelBase.dll, user32.dll, ole32.dll, gdi32.dll, Shell32.dll, Ws2_32.dll, Shlwapi.dll, Advapi32.dll, Iphlpapi.dll, Rstrtmgr.dll, Netapi32.dll, sechost.dll, cryptbase.dll, bcryptprimitives.dll.

Mutexes

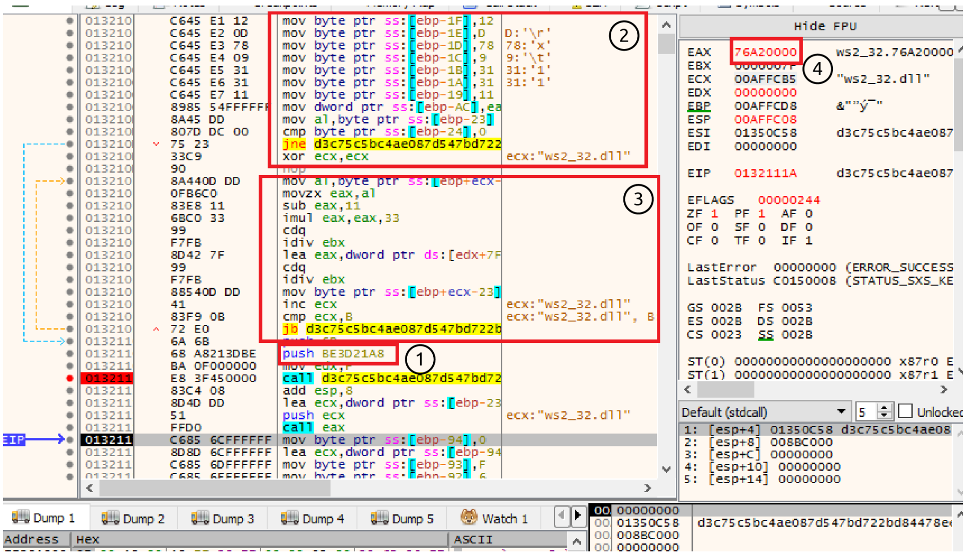
In order to avoid two instances of the same malware to run at the same time and slow encryption times, starting with Conti-V2, a mutex is created on the infected system. As from Conti-V3, the creation of the mutex is optionally and can be specified via command line as a flag. [72]

Anti-reverse engineering techniques

Obfuscation and Dynamic library linking

Conti uses an obfuscation technique together with dynamically linking the necessary libraries. During the start of the execution, the malware retrieves the entries from a list of obfuscated imports and decodes the modules' names. The functions from the modules are retrieved through API calls to GetProcessAdress() and LoadLibraryA() when they are first needed in the attack flow [72], and they are stored into a global variable for later calls.

In the samples from Conti-V1 and Conti-V2, the names of the Windows APIs are encoded using a XOR instruction with a byte value, that differs from one sample to another. Starting with the Conti-V3, the ransomware is using Murmur2A as the API hashing function, with a seed that has a different value for each sample. In 4.3 we can see an example of the following: ① the Murmur function seed, ② part of the encoded string, ③ the decoding loop and ④ the cache address of the imported module.



```
013210  C645 E1 12        mov byte ptr ss:[ebp-1F],12
013210  C645 E2 0D        mov byte ptr ss:[ebp-1E],D       D:'\r'
013210  C645 E3 78        mov byte ptr ss:[ebp-1D],78      78:'x'
013210  C645 E4 09        mov byte ptr ss:[ebp-1C],9       9:'\t'
013210  C645 E5 31        mov byte ptr ss:[ebp-1B],31      31:'1'
013210  C645 E6 31        mov byte ptr ss:[ebp-1A],31      31:'1'
013210  C645 E7 11        mov byte ptr ss:[ebp-19],11
013210  8985 54FFFFFF     mov dword ptr ss:[ebp-AC],ea
013210  8A45 DD           mov al,byte ptr ss:[ebp-23]
013210  807D DC 00        cmp byte ptr ss:[ebp-24],0
013210  75 23             jne d3c75c5bc4ae087d547bd722
013210  33C9              xor ecx,ecx                      ecx:"ws2_32.dll"
013210  90                nop
013210  8A440D DD         mov al,byte ptr ss:[ebp+ecx-
013210  0FB6C0            movzx eax,al
013210  83E8 11           sub eax,11
013210  6BC0 33           imul eax,eax,33
013210  99                cdq
013210  F7FB              idiv ebx
013210  8D42 7F           lea eax,dword ptr ds:[edx+7F
013210  99                cdq
013210  F7FB              idiv ebx
013210  88540D DD         mov byte ptr ss:[ebp+ecx-23]
013210  41                inc ecx                          ecx:"ws2_32.dll"
013210  83F9 0B           cmp ecx,B                        ecx:"ws2_32.dll", B
013210  72 E0             jb d3c75c5bc4ae087d547bd722b
013211  6A 6B             push 6B
013211  68 A8213DBE       push BE3D21A8
013211  BA 0F000000       mov edx,F
013211  E8 3F450000       call d3c75c5bc4ae087d547bd72
013211  83C4 08           add esp,8
013211  8D4D DD           lea ecx,dword ptr ss:[ebp-23
013211  51                push ecx                         ecx:"ws2_32.dll"
013211  FFD0              call eax
013211  C685 6CFFFFFF     mov byte ptr ss:[ebp-94],0
013211  8D8D 6CFFFFFF     lea ecx,dword ptr ss:[ebp-94
013211  C685 6DFFFFFF     mov byte ptr ss:[ebp-93],F
```

```
Hide FPU

EAX   76A20000     ws2_32.76A20000
EBX   0000007F
ECX   00AFFCB5     "ws2_32.dll"
EDX   00000000
EBP   00AFFCD8     &""ý¯"
ESP   00AFFC08
ESI   01350C58     d3c75c5bc4ae087
EDI   00000000

EIP   0132111A     d3c75c5bc4ae087

EFLAGS   00000244
ZF 1   PF 1   AF 0
OF 0   SF 0   DF 0
CF 0   TF 0   IF 1

LastError   00000000 (ERROR_SUCCESS
LastStatus  C0150008 (STATUS_SXS_KE

GS 002B   FS 0053
ES 002B   DS 002B
CS 0023   SS 002B

ST(0) 00000000000000000000 x87r0 E
ST(1) 00000000000000000000 x87r1 E
```

```
Default (stdcall)   5   Unlocked
1: [esp+4]  01350C58 d3c75c5bc4ae08
2: [esp+8]  008BC000
3: [esp+C]  00000000
4: [esp+10]  00000000
5: [esp+14]  00000000
```

```
00000000
01350C58   d3c75c5bc4ae087d547bd722bd84478e
008BC000
00000000
```

Figure 4.3: Obfuscation technique found in sample[2] using x32dbg

API-unhooking

The ransomware also uses an anti-hooking technique in order to evade the Endpoint Detection and Response(EDR) or Antivirus(AV) tools. The EDRs and AVs tools have a built in functionality of hooking functions (that are known as being the most abused) and modify their definitions, using a jmp instruction to redirect the call into the inspection module to evaluate the behaviour. One of the ways Conti is able to achieve the API-unhooking is by comparing the function definition from the imported linked library on the disk with the definition of the function from the running memory[73], if the malware finds a difference (e.g. having at the beginning of the memory code allocated for the function an opcode of a jmp instruction), it will immediately override the function definition and remove the API hook[74].

Anti-debugging techniques

In order to avoid the code being debugged, Conti uses Windows API functions calls (Native API - T1106), such as IsDebuggerPresent API and NtQueryInformationProcess API, and it also manually checks system structures (System checks - T1497.001), such as the BeingDebugged Flag. Both of these techniques are explained in section 4.1.2.

Anti-Virtual Machine techniques

Time based evasion

Conti uses the NtDelayExecution Windows native API to delay the execution of the malware.

Process white/blacklist

The samples from Conti-V1 and Conti-V2 are using a predefined list of to-be-stopped services, most of them belonging to antivirus solutions, backup products and databases. The malware is spawning multiple instances of net.exe, attempting to stop the services from the before mentioned list and kill the processes that contain the ”sql” string in their name[72].

Following Conti-V3, the functionality of stopping services from a predefined list was removed[72]. Instead, the attackers choose to deploy multiple batch files, after they established an RDP connection on the first compromised host in the network[75].

Deletion of back-ups and shadow copies

In Conti-V1 and Conti-V2, the malware proceeds with deleting the shadow copies using the vssadmin tool. The list of commands is hard-coded and through this approach only a fixed amount of drives can have the shadow copies deleted.

In the following samples, starting with Conti-V3, the malware is using a dynamically approach, using methods of a class contained in the root/cimv2 namespace[72]. The root/cimv2 namespace is the default namespace of the Windows Management Instrumentation (WMI) repository and ”it contains classes for computer hardware and configuration”[76]. The malware is querying the Win32_ShadowCopy class for the IDs of the existent shadow copies and pro-

[2] SHA256:d3c75c5bc4ae087d547bd722bd84478ee6baf8c3355b930f26cc19777cd39d4c

ceeds with deleting the shadow copies found by spawning multiple instances of WMIC.exe, issuing delete commands with the specified shadow copy's ID[77].

4.2.3 Persistence mechanism and privilege escalation

No persistence or lateral movement mechanisms were encountered in the malware binary. Yet, the attackers are using in the infection chains external tools like CobaltStrike and AnyDesk to achieve persistence and move laterally. Multiple techniques used for the deployment of such tools were observed as follows[75]:

- Creation of a temporary service for the execution of the CobaltStrike Beacon DLL;

- Abusing of the registry AutoRun keys for the execution of RDP tools (e.g. AnyDesk, Atera);

- Creation of a new admin account;

4.2.4 Encryption scheme and key management

Even from the first samples, Conti has an approach that allows for encryption parallelisation, shortening the encryption time considerably.

To achieve parallelisation, the malware uses Input/Output Completion Ports (IOCPs)[72]. IOCPs are a threading model in which a queue is used for helping the threads to process multiple asynchronous I/O request in a quick and well-organized way. When operating with this model, processes can handle many concurrent asynchronous I/O requests without the need of creating new threads.

Pre-encryption

The malware first checks for the number of processors threads the system has (n_{cpu}) and creates $2 \times n_{cpu}$ threads and IOCPs for handling the operations with the files soon-to-be-encrypted. In the latest versions of Conti we can see the use of C++ tail queues, together with a threadpool implementation done by the attackers to increase the efficiency of the encryption process.

Windows Command Shell

Starting with Conti-V3, the ransomware can run with command lines arguments, such as -h [path_to_hosts_file] - allows the malware to first scan a list of host for possible to-be-encrypted network shares, -p [system_path] - allows the malware to first scan the specified location for to-be-encrypted files. Moreover, Conti has multiple modes of encryption that can be given through the command line argument -m. These modes include: -m all - the encryption of the whole system (which is the one enabled by default), -m local - only the local drives, -m net - only the network shares and -m backup - which is not implemented in this sample.

File Discovery

The malware is using the GetLogicalDriveStringsW() function in order to enumerate all the local drives on the infected host and stores their paths in a DrivesList.

For each local drive found it will recursively search for files and store their path in a DirectoryList. In the search process, the malware checks each directory and file name against a corresponding hard-coded blacklist.

In order to reduce the time of the encryption process the attackers implemented a different encryption mode that have the following options: FULL_ENCRYPTION, PARTLY_ENCRYPTION (20% or 50%) and only HEADER_ENCRYPTION. Based on the file size or the file type the ransomware will encrypt the file using one of the before mentioned modes. In the latest analysed sample [71] the encryption is done as follows:

FULL_ENCRYPTION

- files that have a database extension in the filename
- files with the size $\leq 1Mb$

PARTLY_ENCRYPTION

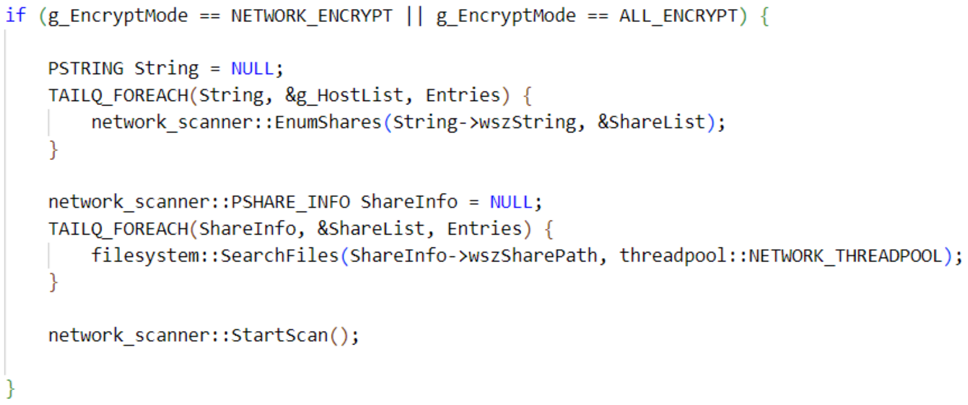
- 20% - files that have a virtual machine extension in the filename
- 50% - files with the size $\geq 5Mb$

HEADER_ENCRYPTION

- files with the size $\geq 1Mb$ and $\leq 5Mb$

Network shares discovery

After the discovery and iteration of the local drives and files, the ransomware will first check if there is a list of hosts provided through the command line argument and if there is, it will start iterating through the list checking which of these hosts have open SMB network shares using a call to the NetShareEnum API [72]. The function will return a ShareList that will be then iterated and for each share found the SearchFile will be called in order to populate the Network Threadpool with the paths to the shares to-be-encrypted. The last step is retrieving through a GetIpNetTable() call the ARP cache from the local system and checks every entry of the ARP table against the following masks: 172.*, 192.168.*, 10.*, 169.*. If such entry exist it will be added to a SubnetList and it will be scanned for SMB connection[78].



```
if (g_EncryptMode == NETWORK_ENCRYPT || g_EncryptMode == ALL_ENCRYPT) {

    PSTRING String = NULL;
    TAILQ_FOREACH(String, &g_HostList, Entries) {
        network_scanner::EnumShares(String->wszString, &ShareList);
    }

    network_scanner::PSHARE_INFO ShareInfo = NULL;
    TAILQ_FOREACH(ShareInfo, &ShareList, Entries) {
        filesystem::SearchFiles(ShareInfo->wszSharePath, threadpool::NETWORK_THREADPOOL);
    }

    network_scanner::StartScan();

}
```

Figure 4.4: Network shares enumeration function from ContiLeaks sample

Importing libraries

Among the dynamically loaded libraries used in the file encryption process, we observed Windows CryptoAPI.

Encryption scheme

Conti uses a combination of asymmetric encryption (RSA-4096) and symmetric encryption (AES-256, ChaCha8 or ChaCha20)[78, 72, 77].The ransomware has a two-layer RSA encryption scheme approach[68]. The steps of the encryption scheme are explained below:

First layer

1. The attacker creates an RSA pair of keys per victim ($RSA_{Vpub}; RSA_{Vpriv}$).
2. The attacker embeds only the RSA_{Vpub} key in the malware binary (usually found in the .data section of the portable executable (PE)).

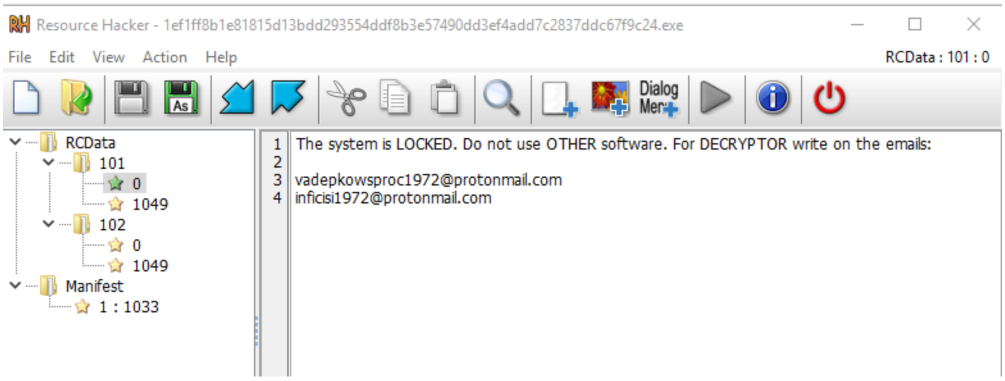
Second layer

3. The ransomware generates an AES_{file_i} key or a $ChaCha_{file_i}$ key and the ChaCha Initialization Vector (for the samples starting with Conti-V3) for each to-be-encrypted $file_i$ and stores them in a file info structure.
4. $file_i$ is encrypted using the AES_{file_i}/ $ChaCha_{file_i}$ key.
5. The AES_{file_i}/ $ChaCha_{file_i}$ key is encrypted using the RSA_{Vpub} key.
6. The encrypted key from step 5 is attached at the end of the encrypted $file_i$.
7. The key from the file info structure of each $file_i$ is deleted.

In the earlier versions of the Conti ransomware, the .CONTI extension was appended to the encrypted files, yet the latest version generates a random 5-character string and appends it to the encrypted files.

4.2.5 Ransom Note

After the encryption of the files has taken place, the ransomware leaves a "R3ADM3.txt" in each folder that was encrypted, notifying the victim of the attack and providing some contact details for negotiations. In the first samples of Conti, the attackers were providing as least as possible information. In the figure 4.5 we can see an extracted ransom note using the Resource Hacker tool.


Resource Hacker - 1ef1ff8b1e81815d13bdd293554ddf8b3e57490dd3ef4add7c2837ddc67f9c24.exe
File
Edit
View
Action
Help
RCData : 101 : 0
Dialog Mer
RCData
101
0
1049
102
0
1049
Manifest
1 : 1033

```
The system is LOCKED. Do not use OTHER software. For DECRYPTOR write on the emails:

vadepkowsproc1972@protonmail.com
inficisi1972@protonmail.com
```

Figure 4.5: Ransom note from sample [3]

In the later samples, the text of the ransom note is moved from the .rsrs section to .data section and it contains more contact information, this time including an .onion address to access through TOR, an email for contact and an ID that you can introduce on their portal for them to identify you.

4.3 Discussion

In this section we will be answering RQ1 by comparing the results from the analysis performed on Ryuk Ransomware (section 4.1) and Conti Ransomware-as-a-Service (section 4.2).

Initial access and spreading mechanism

Both ransomware families need a loader in order to be deployed on the target system. In the most recent samples of Ryuk we observed the "worm-like" ability which helps with the self-propagation and self-replication of Ryuk sample across the networks in an automatic way, without human operation. Conti

[3] SHA256:1ef1ff8b1e81815d13bdd293554ddf8b3e57490dd3ef4add7c2837ddc67f9c24

on the other side has a command line interface designed to be manually operated by an attacker. The command line interface allows for a better control of the encryption process flow, the attacker being able to skip the files that do not present any interest, shortening the time of the encryption process.

Ryuk uses scheduled tasks in order to start the ransomware infection on the discovered hosts from the network (on which it replicated) at a specific time designated by the attackers.

The approach of Ryuk might compromise more systems, but the information on them might not present enough interest. Using scheduled tasks to compromise the rest of the hosts from the network is a good method for keeping the noise on the security events management platforms to a minimum, delaying the discovery of the security incident or complicating the work of the incident response team. Conti can choose which machines from the network would be a better target for getting a bigger revenue.

Mutexes

Ryuk uses a mutex with the name of the username of the system to make sure the ransomware sample executes only once on that machine. If the mutex already exist it will kill the process.

The mutex seems to be used to prevent an overinfection of the host machines in the network, which would give an overhead to the encryption time.

Conti makes use of a mutex with a hard-coded unique identifier for each sample that can be found in the binary.

File and network shares discovery

Conti is searching recursively for the files and check the filenames against multiple hard-coded lists, including but not limited to an extension blacklist, virtual machine and database extension lists, to apply a different mode of encryption.

In table 4.3 we can see the file extensions that each of the ransomware families are skipping from the encryption process.

	Ryuk	Conti
.dll	✓	✓
.lnk	✓	✓
.hrmlog	✓	
.ini	✓	
.exe	✓	✓
.sys		✓
.msi		✓

Table 4.3: File extensions blacklist for the encryption process

Ryuk is making use of two hard-coded lists of paths and filenames and has an extra check for the ”index.” or ”.php” files, which instead of being encrypted, they are overwritten with the html version of the ransom note, respectively with php code that renders the html version of the ransom note. This approach is a quite unique one among the ransomware families, as it seems that the new functionality helps with public websites defacement.

Concerning the network shares discovery, both of the ransomware families are checking the entries from the ARP table, retrieved from the local system, against a hard-coded list of network masks. The entries that fit the criteria are saved in a list by Conti and checked for an open SMB connection. The extra feature that Ryuk has is the Wake-On-Land command, being able to remotely power-up the systems in the network, followed by the use of RPC to copy itself on the identified network shares. After the copy arrives on the system, a command is issued for scheduling a task implying the start of the infection at a designated time.

By scanning only the ARP table entries, the ransomware is focusing mostly on the system that the host usually connects to, avoiding in this way the generation of a big volume of security alerts on the SEM.

Encryption

Both Ryuk and Conti make use of a combination of symmetric encryption algorithms and asymmetric encryption algorithms. For the encryption of the files, Ryuk uses AES-256. The first samples of Conti were using AES-256 for encrypting the file, but they gradually changed to ChaCha algorithms, the latest version analysed having an implementation of ChaCha20. This change, from AES-256 to ChaCha20, seems to be due to the fact that ChaCha20 is significantly faster than AES and it does not require special hardware specification to run seamlessly[79].

Ryuk makes use of two RSA public keys (RSA_Apub and RSA_Vpub) that can be found soon after the infection on the compromised system. Conti has only one RSA public key (RSA_Vpub) embedded in the binary, as observed from the ContiLeaks source code, it can be found in the threadpool implementation file. Embedding the public key for encryption in the binary of the ransomware is a fail-safe method, meaning in case the connection to the C&C server is not possible, the encryption of the system can still happen. Also, having a unique encryption key per victim means that the threat actor is using a tailored binary for each target.

Ryuk has a multi-threading approach for the encryption of the files, using a thread for each file to be encrypted. The encryption of the files is done in blocks of 1Mb. Conti, on the other side, is using encryption parallelisation ($2 \times n_{cpu}$ threads running at the same time) through IOCPs and C++ tail queues used in the implementation of the threadpools. Using threadpools allows for a more efficient way to deal with multiple threads, allowing the application to reuse an already existent thread, fact that improves the speed of the encryption process.

Furthermore, Conti has multiple modes of encryption. The files are encrypted totally, partially (in jumps) or only their header based on the size or the type of the file. This approach was taken to speed up the encryption process.

To not encrypt the same file twice, Ryuk checks if the file contains either the string ”HERMES” or the string ”RYUKTM”. Concerning Conti, the malware looks for the extension of the file in order to not try to encrypt it again. In the first samples of Conti ransomware, the extension of the encrypted files was ”.CONTI”, but by being too easily to detect by the AVs or EDRs, the thread actors changed the approach, now generating a random extension for each Conti sample.

MITRE ATT&CK matrix

The MITRE ATT&CK (Adversary Tactics, Techniques and Common Knowledge) framework is a guideline used by the cybersecurity specialists. The matrix contains all the documented steps of cyberattacks based on observation from multiple sources.

The tables below contain all the techniques and sub-techniques, as defined by MITRE ATT&CK framework, that were observed through the analysis of the ransomware samples' behavior. The techniques are grouped based on the phase of the attack in which they are used, as follows: Execution (Table 4.4), Persistence (Table 4.5), Privilege escalation (Table 4.6), Defense evasion (Table 4.7), Credential access (Table 4.8), Discovery (Table 4.9) and Impact (Table 4.10).

	Ryuk	Conti
Native API	✓	✓
Command and scripting interpreter: Windows command shell	✓	✓
Command and scripting interpreter: Powershell		✓
Windows Management Instrumentation (WMI)		✓
User execution		✓
Scheduled task	✓	✓
System checks	✓	✓
Shared modules		✓

Table 4.4: Execution MITRE ATT&CK techniques used

	Ryuk	Conti
Registry Run Keys	✓	
Startup Item	✓	✓
External Remote Services		✓
Scheduled Task	✓	✓

Table 4.5: Persistence MITRE ATT&CK techniques used

	Ryuk	Conti
Valid accounts: Domain accounts	✓	✓
Dynamic-link Library Injection	✓	✓
Scheduled Task	✓	
New Service		✓

Table 4.6: Privilege escalation MITRE ATT&CK techniques used

	Ryuk	Conti
Obfuscated Files or Information	✓	✓
Deobfuscate/Decode Files or Information		✓
Impair Defenses: Disable or Modify tools	✓	✓
Access Token Manipulation	✓	✓
Windows File and Directory Permissions Modification	✓	
Masquerading	✓	
Thread Execution Hijacking	✓	
Indicator Removal on Host: Clear Windows Event Logs		✓

Table 4.7: Defense evasion MITRE ATT&CK techniques used

	Ryuk	Conti
Brute Force		✓
Steal/Forge Kerberos Tickets: Kerberoasting		✓
OS Credential Dumping	✓	✓
Credentials from Password Store	✓	✓
Steal Web Session Cookie		✓

Table 4.8: Credential access MITRE ATT&CK techniques used

	Ryuk	Conti
File and Directory Discovery	✓	✓
System Network Connections Discovery	✓	✓
Process Discovery	✓	✓
Network Share Discovery	✓	✓
System Information Discovery	✓	✓
Security Software Discovery		✓
Remote System Discovery		✓
System Language Discovery	✓	

Table 4.9: Discovery MITRE ATT&CK techniques used

	Ryuk	Conti
Data Encrypted for Impact	✓	✓
Service Stop	✓	✓
Inhibit System Recovery: vssamin	✓	✓
Inhibit System Recovery: wmic		✓

Table 4.10: Impact MITRE ATT&CK techniques used

4.4 Ransomware4Students

Ransomware4Students[4] showcases a dummy implementation of a ransomware and shows a demo of an interaction between an attacker and a victim. The goal of the demo is to illustrate two items 1) how simple yet effective a ransomware as a service implementation could be, and 2) what is a possible software development process for ransomware creation[5].

The victim can be the localhost or a remote server. The attacker runs RAASNet.py, and has access to the attacker's dashboard, where several actions can be made: start the server (and receive incoming requests from victims), generate payload (customizable), compile payload (that will be sent to the victim), and a set of commands send to the victims. Those commands are to encrypt files and decrypt files, where the latter happens upon payment.

RAASNet can generate payloads on-the-go, supporting multiple operation systems, being quite customizable. A range of around 40 parameters can be customizable, from which we emphasize the host, port, encoding, message, and remove_payload. The host parameter refers to the machine running RAASNet, along with its port, so the victim can connect to the attacker. The encoding specifies what is the encoding used on the payload that will be sent to the victim. This implies that the executable will have obfuscated code and a decoder. This hampers the work of malware analysis, because the source code, if caught, is obfuscated. The message specifies the message to be shown to the victim, when the encryption process starts. Finally, remove_payload contains the self destruct directive. This helps the source code not to be taken by malware analyists.

Upon customization and generation, the script payload.py is the ransomware that will be sent to the victim, via an infected e-mail, or similar means. We simulate the victim downloading an infected e-mail's attachment, containing the payload. The payload is run and does four things: 1) it generates a symmetric key, 2) it encrypts the target directories and files, 3) it sends the generated key to the attacker, and 4) self destroys.

When the ransom is paid, the attacker sends over the channel a decryption script, decryptor.py, also encoded, which the victim can run to have its files decrypted. There can be a variety of ways to automate the payment process, such as checking if a certain transfer of cryptocurrencies was made to a payment account.

We try to use similar techniques to adapt the actual base implementation to have features closer to the behaviour of a real malware. We implemented a layered approach to encryption, to circumvent the cases when the connection to the C&C server is not possible. We create a RSA key pair. The private key stays with the attacker, while the public key is hard-coded and sent along with

[4] available at Github on https://github.com/maramih/Ransomware4Students. This repository is an extension to the leonv024/RAASNet repository, https://github.com/leonv024/RAASNet.

[5] Some networking tools are also available, e.g., test_socket.py, that confirms if a connection to a test port on the localhost, 127.0.0.1, on port 8989 is successful.

the payload binary. The victim, when executing the payload, has their files encrypted with AES. The AES key is encrypted with RSA and it is appended to the file.

The attacker's incentive is not just to exfiltrate and encrypt the data, getting the revenue depends on detaining the decryption key. It is a method to convince the people that they can really retrieve their data back, being encouraged to pay the ransom asked.

This proof of concept is a starting point for students that want to pursue a career in malware analysis. By having a configurable tool that won't harm your computer, but helping you learn the flow of an attack.

Chapter 5

Case Study

In this chapter, we explore a use case that can illustrate to the reader the destructive power of Conti in an attack known as the largest to date against a healthcare service [80]. We base our case study in several sources [48, 49, 81, 82, 83, 84].

5.1 Attack Description

In May 2021, Ireland's HSE (Health Service Executive), responsible for the provision of Ireland's public health services in hospitals and communities across the country, was attacked by the cyber criminal group known as Wizard Spider. Known for attacking countries such as the United States, France, Germany, Canada, the U.K., Italy, Australia, Spain, and the Netherlands, Wizard Spider is a highly sophisticated, financially motivated group. The attack took several months to be considered settled and had deep repercussions at HSE. Its systems had 80% of its data encrypted due to a Conti ransomware attack, leading to predicted damages in the tens or even hundreds of millions of euros due to services being shut down and patient lawsuits. This caused three major problems: 1) a severe disruption of healthcare services throughout Ireland, preventing the ideal service provided to patients, 2) disruption of communication channels, both at HSE's national center and within its operational services, including email and networked phone lines, and 3) roughly 700 Gb of personal data (regarding Irish people who received COVID-19 vaccines) were stolen by the attackers, putting people's privacy at risk. Figure 5.1 shows the timeline of the attack. In March 2021, the first victim was infected. On May 14th, the attacker executed Conti on HSE, encrypting most of its files and leading to a general shutdown. On the same day, the relevant stakeholders were briefed about the attack. The following day, an Incident Response team was created to respond to the incident. The recovery began on May 21st when the decryption key was provided. Only in June, about half of the services were brought back online. Finally, all the data was decrypted by September, and all the systems were operational.

The impact of this attack was very significant. Several hospitals could not access electronic systems and records and had to rely on paper records, causing inefficiencies and disruption of services, such as routine appointments and oncology services.

Next, we detail each ransomware life cycle phase for our use case.

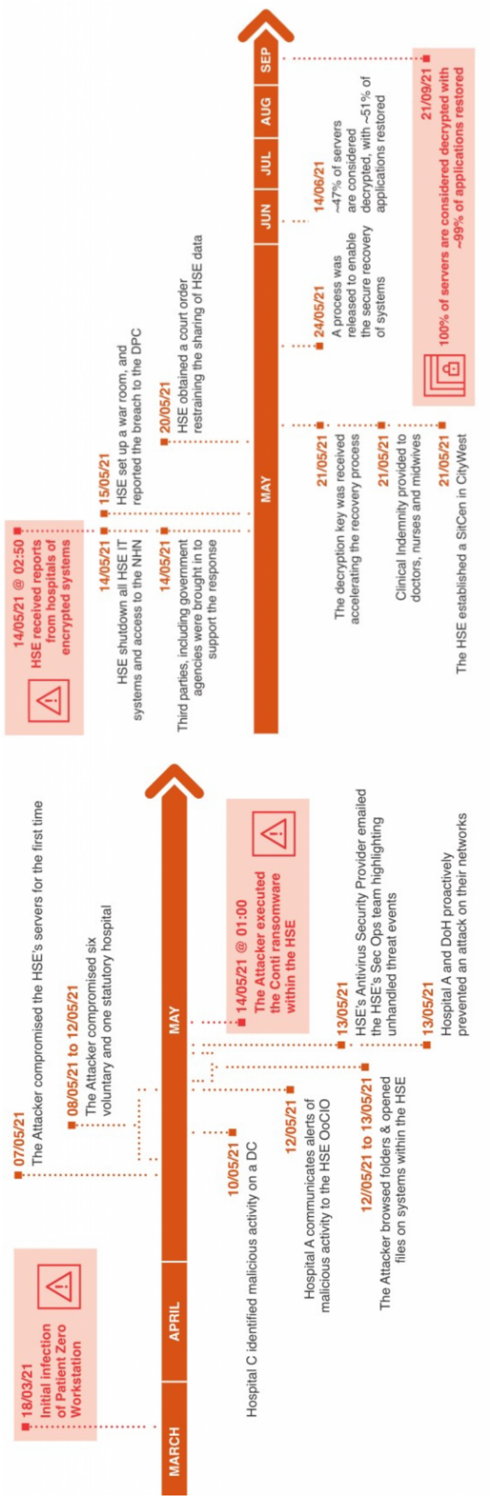

MARCH
APRIL
MAY
18/03/21 Initial infection of Patient Zero Workstation
07/05/21 The Attacker compromised the HSE's servers for the first time
08/05/21 to 12/05/21 The Attacker compromised six voluntary and one statutory hospital
10/05/21 Hospital C identified malicious activity on a DC
12/05/21 Hospital A communicates alerts of malicious activity to the HSE OoCIO
12/05/21 to 13/05/21 The Attacker browsed folders & opened files on systems within the HSE
13/05/21 HSE's Antivirus Security Provider emailed the HSE's Sec Ops team highlighting unhandled threat events
13/05/21 Hospital A and DoH proactively prevented an attack on their networks
14/05/21 @ 01:00 The Attacker executed the Conti ransomware within the HSE
14/05/21 @ 02:50 HSE received reports from hospitals of encrypted systems
14/05/21 HSE shutdown all HSE IT systems and access to the NHN
14/05/21 Third parties, including government agencies were brought in to support the response
15/05/21 HSE set up a war room, and reported the breach to the DPC
20/05/21 HSE obtained a court order restraining the sharing of HSE data
MAY
JUN
JUL
AUG
SEP
21/05/21 The decryption key was received accelerating the recovery process
21/05/21 Clinical Indemnity provided to doctors, nurses and midwives
21/05/21 The HSE established a SitCen in CityWest
24/05/21 A process was released to enable the secure recovery of systems
14/06/21 ~47% of servers are considered decrypted, with ~51% of applications restored
21/09/21 100% of servers are considered decrypted with ~99% of applications restored

Figure 5.1: Timeline of the attack (from preparation to resolution). Source [83]

5.2 Preparation and Infection

Wizard Spider is the group that perpetrated the attack, allegedly conducting a series of attacks on the U.S. Department of Health. One can conclude that this group has domain expertise in healthcare.

The attack started with the group sending a malicious email to a workstation connected to the HSE network on March 16th, 2021. Two days later, a single user opened a malicious file attached to a phishing email, causing the remote access tool known as Cobalt Strike Beacon to be installed on the machine, used further for moving laterally in the environment and for contacting the command-and-control server(C2). The next step was using a reflective DLL loader from Cobalt Strike, known as a beacon, to deliver the ransomware into memory. CobaltStrike allowed artifacts not to be left behind, preventing analysis by the security analysts from HSE. Finally, it deployed the Conti ransomware payload in the network. This proved to be an effective attack since anti-viruses from the network only started emitting warnings on March 31st.

5.3 Encryption

The ransomware is executed manually in memory, via a command-line interface, across all active endpoints, after as many files as possible have been exfiltrated.

Upon execution, it makes many misleading WinAPI calls with invalid arguments to intentionally throw exceptions. These are then handled by the malware and act as an anti-emulation/sandbox evasion technique. After infection, the ransomware can immediately begin to encrypt the victim's files (Conti uses a unique AES-256 encryption key per file, which is then encrypted with an RSA-4096 encryption key). In the earlier versions of the Conti ransomware, the CONTI extension was appended to the encrypted files, yet version v3 -32 bits used in this attack generates a random 5-character string (*.FEEDC) and appends it to the encrypted files.

5.4 Extorsion

Several media outlets reported that the attackers were initially asking for a ransom of 16.5 million euros to provide a decryption key and not to leak sensitive data. Other outlets reported that a ransom demand of three bitcoin, or 124 thousand euros, at the time, was demanded. The Irish prime minister stated that no ransom would be paid. The Conti operators did something rare in this area: they provided the decryption key. According to [83], the attackers said, on the negotiation platform: "We are providing the decryption tool for your network for free. But you should understand that we will sell or publish many private data if you do not connect us and try to resolve the situation". However, the private medical data of 520 patients were published online. More than 40 hospitals were affected.

5.5 Analysis and Response

This attack puts into perspective the setup that leads to the attack. Firstly, HSE's technology has been growing steadily and becoming more overly complex, increasing the attack surface. A lack of effective patching and security updates (including in antiviruses) across the I.T. infrastructure exacerbated this problem. This hampered the organization's ability to attract a qualified workforce. Let such a role had been taken before the attack, the risks that led to the attack might have been effectively debated and mitigated. The postmortem response was likely to o slow and ineffective due to the lack of a clear contingency plan. Such a fact was not accompanied by investing in cyber security: the HSE did not have an executive level responsible for cybersecurity. This left the role of providing oversight of cybersecurity in the void. The organization responsible for the Irish's state national security, the National Cyber Security Centre had few resources and key roles to fill, such as the Director role (supposedly due to the lack of competitive compensation). The lack of qualified human resources led to a heavy reliance on a few individuals, hampering response capacity in the event of an attack. Issues like this were known but not solved. It was also known that the antivirus tools that were installed in several systems were over-relied on detection and prevent cyber attacks.

This attack led HSE to be more proactive about its cyber security. HSE, which had no tools to investigate the attack alone, started working with the National Cyber Security Centre, the Garda Síochána, Irish Defence Forces, and various domestic and international partners, including Europol and Interpol. PwC, a software consultancy firm, proposed HSE actions to mitigate future attacks, including but not limited to a 24-hour monitoring system for I.T. systems in the HSE, developing, documenting, and exercising a plan for coordinating cyber security incidents, and multi-factor authentication for accessing services (some action points to mitigate ransomware infection are presented in the next chapter).

Strategic actions, including reinforcing the governance of I.T. and cybersecurity, leadership and transformation of I.T. and cybersecurity, development of clinical and services continuity, and crisis management that supports outages.

Chapter 6

Recommendations to mitigate ransomware infections

This section presents a list of measures to mitigate Ryuk and Conti infection to technical employees, answering the second research question. Our suggestions come from several sources [18, 22, 85, 86, 87, 88] and our experience with analysing the ransomware studied in this thesis.

6.1 Recommendations to a non-technical audience

Employees that perform their work on computers typically have access to organization networks, assets, and contacts, being one of the largest attack surfaces for ransomware. Thus, in this section, we tailor our recommendations to this audience. The non-technical audience includes administrative employees, business people, marketing, sales, and others.

Recommendation 1 - Promote Education

Educating employees is the first line of defense against ransomware.

Education is the first line of defense. All employees should have mandatory security training, including one specific to ransomware. Some of the essential items an employee should have present are 1) phishing education (never click on unknown links, for example via email); 2) not opening suspicious email attachments; 3) do not navigate in websites flagged as insecure by antiviruses and antimalwares. The training for the employees and testing of their reactions to cyber incidents should be kept updated to maintain the organization secure. The workforce should also be able to identify ransomware attacks and their attempts and know how to report them.

Recommendation 2 - Do regular backups of important data

Backups should be immutable, encrypted, air-gapped, and follow a 3-2-1 backup strategy (keep three copies of your data, store two copies on different storage media, and ensure one copy is kept off-site).

In most ransomware attacks, the threat actor will often delete or encrypt all data available within the infected machine and the networks it connects. It is essential to keep updated backups stored in a different machine and network. Otherwise, the backups might be targeted.

Recommendation 3 - Do regular software updates

Updating software regularly reduces the chances of criminals exploiting software vulnerabilities.

Ransomware groups target unpatched software and operating systems vulnerabilities to gain access to whole networks. In order to close the window of time, an attacker has to target you or your customers. It is recommended to patch the applications and systems on a timely basis, ideally as soon as updates are pushed out.

Recommendation 4 - Monitor your system regularly

Monitoring your system and network is a safeguard and early warning system for potential problems.

Actively monitoring computer and network metrics is a good practice. Suppose the employee notices changes in the baseline metrics (for example, inexplicable changes in CPU load or intensive disk input/output activity), it might indicate a ransomware attack. System metrics can be sent as logs to a detection and response solution that will identify and notify the end-user about an attack or its possibility.

Recommendation 5 - Have a contigency plan in place

Ensure there is an Incident Response & Disaster Recovery Plan agreed upon by relevant stakeholders and actionable in case of ransomware infection.

Developing an Incident Response & Disaster Recovery Plan is critical to handling cyberattacks, as it can reduce downtime and damage the target systems. Furthermore, a structured approach might help recover from financial and reputation loss because relevant stakeholders have agreed on what to do in case that happens. Strategies for that plan include 1) defining the strategy for risk analysis by identifying the risks and vulnerabilities to the confidentiality, integrity, and availability of all organizational data, 2) risk management by implementing security measures sufficient to reduce identified threats, and 3) business associate agreements, including defining processes, and responsibilities to prevent, manage and disclose cyberattacks. Leveraging existing recommendations such as HIPAA Security Rule can help covered entities and business associates prevent malware infections, including ransomware [89].

Recommendation 6 - Follow your organization's security policy

All organizations should have a security policy that diminishes the likelihood of a ransomware attack.

Security policies might advise using a virtual private network when using a public Wi-Fi network or even avoiding using it when a computer is more vulnerable to attacks.

Recommendation 7 - Use an antivirus and antimalware

Use antivirus and antimalware software and keep them updated.

Using an effective security solution – installing a tool or a pack of tools that can provide features like antivirus, antimalware, antispyware, antispam, firewall, and others.

Recommendation 8 - Use multifactor authentication

Using multifactor authentication greatly hampers the ability of an attacker to obtain control over a machine or cloud-based service, therefore hampering their ability to encrypt data and demand a ransom.

6.2 Recommendations to a technical audience

Other recommendations can be given to a technical audience, including but not limited to information technology staff, cybersecurity experts, network administrators, and others:

- Recommendation 1 - Implement application whitelisting. Application whitelisting is effective against ransomware as it prevents unknown applications from executing (until deemed safe).

- Recommendation 2 - Disable "problematic features". Disable features like auto-run, remote desktop protocols connections (RDPs), and macros from Microsoft Office (for Windows operating systems), as attackers can easily exploit them.

- Recommendation 3 - Apply the Principle of Least Privilege. Configure employee accounts with only the access privileges required for their job roles and no more. User permissions on installing and running software should be limited to a set of allowed ones.

- Recommendation 4 - Apply network segmentation and filter traffic. Robust network segmentation should be implemented to segregate networks and be able to promote isolation. This helps isolate infected machines and prevent the spread of ransomware upon infection.

  Furthermore, spam filters should be set to prevent phishing emails from reaching end users.

- Recommendation 5 - Remove unnecessary applications. Remove applications not necessary for day-to-day operations, and carefully watch certain families of software. Ransomware, such as Conti, targets software such as remote management software, which can be used to infect more machines. y. Software restriction policies (SRPs) and other controls can prevent programs from executing from common ransomware locations (e.g., temporary folders).

- Recommendation 6 - Implement detection response tools. Detection response tools allow a high degree of visibility into the security status of the organization's services, helping protect against cyberattacks.

- Recommendation 7 - Have a contingency plan. The contingency plan can be defined similarly to a previous recommendation (to non-technical users). This contingency plan can follow a "Ransomware Response Checklist", in case of infection, such as the one recommended by CISA, FBI, and NSA [18].

- Recommendation 8 - Report incidents - Incidents should be reported not only to the appropriate stakeholders within an organization but also to competent authorities.

- Recommendation 9 - Stay up to date. The ability of the organization to stay updated on recent malware trends is essential to understanding the full extent of ransomware's threat to the organization. Thus, an organization can better prepare to respond to it with a better understanding.

Conclusions

Ransomware is a severe, growing threat plaguing our world. We have assisted to steadily usage of this malware in order for criminals to extort victims. In this paper, we characterize two well-known families of ransomware Ryuk and Conti, and explore the side of the attacker and the side of the victim in Ransomware4Students.

Next, we perform a static analysis supported by dynamic analysis. We conclude that samples vary substantially and are continuously improved. Our findings show intriguing details: the attackers seem to put focus on the encryption phase, namely performance. Optimization of performance includes using faster algorithms such as ChaCha. We also found out that ransomware developers put a lot of effort in attempting for their software to be reverse engineered. By applying diverse obfuscation techniques, it hampers the work of malware analysis and authorities.

Next, we present a case study where attackers used Conti and temporarily created havoc in a sovereign country healthcare system, showing the destructive power the malware carries. We conclude that ransomware is both a high impact and high probable threat, so reducing the attack surface and increasing the difficulty level of the infection is key. To do so, education is paramount. We conclude this thesis with a list of actionable items that promote education and therefore protect an organization from malware attacks.

Future Work

Several venues for future work are possible as the follow-up of this work.

An interesting research direction is to conduct malware analysis on a wider range of samples, in order to derive patterns for obfuscation and incremental improvement of malware. This would allow security researchers to identify patterns on the software development life cycle for ransomware.

Although there are a few well-documented case studies dealing with ransomware, that is not enough to provide a full picture on what is a suitable response to such scenario, across different industries. We hypothesise that the scarcity of case studies is due to the best interest of organizations to keep attacks confidential, especially if they have a dim outcome.

We ought to improve Ransomware4Students to provide better educational resources, this proof of concept being a starting point for students that want to pursue a career in malware analysis.

Bibliography

[1] Braue, David, “Global Ransomware Damage Costs Predicted To Exceed $265 Billion By 2031,” 2022, section: Blogs. [Online]. Available: https://cybersecurityventures.com/global-ransomware-damage-costs-predicted-to-reach-250-billion-usd-by-2031/

[2] K. Rieck, T. Holz, C. Willems, P. Düssel, and P. Laskov, “Learning and classification of malware behavior,” in International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment. Springer, 2008, pp. 108–125.

[3] A. Young and M. Yung, “Cryptovirology: Extortion-based security threats and countermeasures,” in Proceedings 1996 IEEE Symposium on Security and Privacy. IEEE, 1996, pp. 129–140.

[4] M. Paquet-Clouston, B. Haslhofer, and B. Dupont, “Ransomware payments in the bitcoin ecosystem,” Journal of Cybersecurity, vol. 5, no. 1, p. tyz003, 2019.

[5] S. Nakamoto, “Bitcoin whitepaper,” URL: https://bitcoin. org/bitcoin. pdf-(: 17.07. 2019), 2008.

[6] P. H. Meland, Y. F. F. Bayoumy, and G. Sindre, “The ransomware-as-a-service economy within the darknet,” Computers & Security, vol. 92, p. 101762, 2020.

[7] Z. Manjezi and R. A. Botha, “Preventing and mitigating ransomware,” in International Information Security Conference. Springer, 2018, pp. 149–162.

[8] E. Clay, “How to Mitigate the Risk of Ransomware Attacks: The Definitive Guide,” Jul. 2020. [Online]. Available: https://touchstonesecurity.com/mitigate-ransomware-attacks/

[9] N. Spence, M. Niharika Bhardwaj, and D. P. Paul III, “Ransomware in healthcare facilities: a harbinger of the future?” Perspectives in Health Information Management, pp. 1–22, 2018.

[10] U. D. of Health and H. Services, "U.S. Department of Health & Human Services - Office for Civil Rights," 2022. [Online]. Available: https://ocrportal.hhs.gov/ocr/breach/breach_report.jsf

[11] Safety Detectives, "Ransomware facts, trends & statistics for 2022," https://www.safetydetectives.com/blog/ransomware-statistics/, 2022, (Accessed on 06/16/2022).

[12] Sophos, "Ransomware attacks on healthcare organisations increased 94 per cent in 2021: Sophos," Jun. 2022. [Online]. Available: https://www.expresshealthcare.in/healthcare-it/ransomware-attacks-on-healthcare-organisations-increased-94-per-cent-in-2021-sophos/435290/

[13] Sophos, "The State of Ransomware in Healthcare 2022," 2022. [Online]. Available: https://www.sophos.com/en-us/whitepaper/state-of-ransomware-in-healthcare

[14] K. Abouelmehdi, A. Beni-Hessane, and H. Khaloufi, "Big healthcare data: preserving security and privacy," Journal of big data, vol. 5, no. 1, pp. 1–18, 2018.

[15] G. George and S. M. Thampi, "Securing smart healthcare systems from vulnerability exploitation," in International Conference on Smart City and Informatization. Springer, 2019, pp. 295–308.

[16] W. Priestman, T. Anstis, I. G. Sebire, S. Sridharan, and N. J. Sebire, "Phishing in healthcare organisations: Threats, mitigation and approaches," BMJ health & care informatics, vol. 26, no. 1, 2019.

[17] MalwareBytes, "Ryuk - What is Ryuk Ransomware?" 2022. [Online]. Available: https://www.malwarebytes.com/ryuk-ransomware

[18] CISA, "Conti Ransomware | CISA," 2022. [Online]. Available: https://www.cisa.gov/uscert/ncas/alerts/aa21-265a

[19] Bhabesh, Raj, "Detecting Conti ransomware - The successor of infamous Ryuk," Sep. 2021. [Online]. Available: https://www.logpoint.com/en/blog/detecting-conti-ransomware-the-successor-of-infamous-ryuk/

[20] H. Kwon, B. Chung, D. Moon, and I. Kim, "Security technology trends to prevent medical device hacking and ransomware," Electronics and Telecommunications Trends, vol. 36, no. 5, pp. 21–31, 2021.

[21] D. F. Sittig and H. Singh, "A socio-technical approach to preventing, mitigating, and recovering from ransomware attacks," Applied clinical informatics, vol. 7, no. 02, pp. 624–632, 2016.

[22] J. Pope, “Ransomware: Minimizing the risks,” Innovations in clinical neuroscience, vol. 13, no. 11-12, p. 37, 2016.

[23] H. Fereidooni, M. Conti, D. Yao, and A. Sperduti, “Anastasia: Android malware detection using static analysis of applications,” in 2016 8th IFIP international conference on new technologies, mobility and security (NTMS). IEEE, 2016, pp. 1–5.

[24] Y. Pan, X. Ge, C. Fang, and Y. Fan, “A systematic literature review of android malware detection using static analysis,” IEEE Access, vol. 8, pp. 116 363–116 379, 2020.

[25] A.-D. Schmidt, R. Bye, H.-G. Schmidt, J. Clausen, O. Kiraz, K. A. Yuksel, S. A. Camtepe, and S. Albayrak, “Static analysis of executables for collaborative malware detection on android,” in 2009 IEEE International Conference on Communications. IEEE, 2009, pp. 1–5.

[26] Y. Feng, S. Anand, I. Dillig, and A. Aiken, “Apposcopy: Semantics-based detection of android malware through static analysis,” in Proceedings of the 22nd ACM SIGSOFT international symposium on foundations of software engineering, 2014, pp. 576–587.

[27] H. Kang, J.-w. Jang, A. Mohaisen, and H. K. Kim, “Detecting and classifying android malware using static analysis along with creator information,” International Journal of Distributed Sensor Networks, vol. 11, no. 6, p. 479174, 2015.

[28] S.-H. Seo, A. Gupta, A. M. Sallam, E. Bertino, and K. Yim, “Detecting mobile malware threats to homeland security through static analysis,” Journal of Network and Computer Applications, vol. 38, pp. 43–53, 2014.

[29] D. Ucci, L. Aniello, and R. Baldoni, “Survey of machine learning techniques for malware analysis,” Computers & Security, vol. 81, pp. 123–147, 2019.

[30] M. Hassen, M. M. Carvalho, and P. K. Chan, “Malware classification using static analysis based features,” in 2017 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE, 2017, pp. 1–7.

[31] V. Syrris and D. Geneiatakis, “On machine learning effectiveness for malware detection in android os using static analysis data,” Journal of Information Security and Applications, vol. 59, p. 102794, 2021.

[32] H. Aghakhani, F. Gritti, F. Mecca, M. Lindorfer, S. Ortolani, D. Balzarotti, G. Vigna, and C. Kruegel, “When malware is packin’heat; limits of machine learning classifiers based on static analysis features,” in Network and Distributed Systems Security (NDSS) Symposium 2020, 2020.

[33] K. Lucas, M. Sharif, L. Bauer, M. K. Reiter, and S. Shintre, "Malware makeover: breaking ml-based static analysis by modifying executable bytes," in Proceedings of the 2021 ACM Asia Conference on Computer and Communications Security, 2021, pp. 744–758.

[34] K. Kendall and C. McMillan, "Practical malware analysis," in Black Hat Conference, USA, 2007, p. 10.

[35] A. Moser, C. Kruegel, and E. Kirda, "Exploring multiple execution paths for malware analysis," in 2007 IEEE Symposium on Security and Privacy (SP'07). IEEE, 2007, pp. 231–245.

[36] E. Gandotra, D. Bansal, and S. Sofat, "Malware analysis and classification: A survey," Journal of Information Security, vol. 2014, 2014.

[37] R. Umar, I. Riadi, and R. S. Kusuma, "Network forensics against ryuk ransomware using trigger, acquire, analysis, report, and action (tara) methods," Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control, vol. 6, no. 2, pp. 133–140, 2021.

[38] O. Filipec and D. Plasil, "The cybersecurity of healthcare the case of the benegov hospital hit by ryuk ransomware, and lessons learned," Obrana a Strategie-Defence & Strategy, pp. 27–51, 2021.

[39] J. Kolouch, T. Zahradnický, and A. Kučínský, Ransomware Attacks on Czech Hospitals at Beginning of Covid-19 Crisis. Cham: Springer International Publishing, 2022, pp. 303–316. [Online]. Available: https://doi.org/10.1007/978-3-030-88907-4_18

[40] J. Reddy, N. Elsayed, Z. ElSayed, and M. Ozer, "Data breaches in healthcare security systems," arXiv preprint arXiv:2111.00582, 2021.

[41] S. Poudyal, K. D. Gupta, and S. Sen, "Pefile analysis: A static approach to ransomware analysis," Int. J. Comput. Sci, vol. 1, pp. 34–39, 2019.

[42] K. C. Roy and Q. Chen, "Deepran: Attention-based bilstm and crf for ransomware early detection and classification," Information Systems Frontiers, vol. 23, no. 2, pp. 299–315, 2021.

[43] C. Mehra, A. Sharma, and A. Sharma, "Elucidating ransomware attacks in cyber-security."

[44] T. Malenfant, "Impact of ransomware attacks on healthcare," Ph.D. dissertation, 2021, copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2022-01-27. [Online]. Available: https://www.proquest.com/dissertations-theses/impact-ransomware-attacks-on-healthcare/docview/2616674142/se-2

[45] K. A. W. Ramsdell and K. E. Esbeck, “Evolution of ransomware,” 2021.

[46] D. J. Middaugh, “Cybersecurity attacks during a pandemic: It is not just it’s job!” Medsurg Nursing, vol. 30, no. 1, pp. 65–66, 2021.

[47] J. Chigada and R. Madzinga, “Cyberattacks and threats during covid-19: A systematic literature review,” South African Journal of Information Management, vol. 23, no. 1, pp. 1–11, 2021.

[48] M. Moran Stritch, M. Winterburn, and F. Houghton, “The conti ransomware attack on healthcare in ireland: Exploring the impacts of a cybersecurity breach from a nursing perspective,” Canadian Journal of Nursing Informatics, vol. 16, no. 3-4, 2021.

[49] H. Harvey, V. Amberger-Murphy, J. Ballot, M. O’Grady, D. O’Hare, G. Lawler, E. Bennette, M. Connolly, C. McNevin, E. Noone et al., “Impact of conti ransomware attack on cancer trials ireland sites.” 2022.

[50] D. Harkin, “Professionalism, pandemics and ransomware: Coping with challenge and uncertainty.”

[51] G. Kim, S. Kim, S. Kang, and J. Kim, “A method for decrypting data infected with hive ransomware,” arXiv preprint arXiv:2202.08477, 2022.

[52] A. S. Lekshmi, “Growing concern on healthcare cyberattacks & need for cybersecurity,” 2022.

[53] S. A. Khanday, H. Fatima, and N. Rakesh, “Deep learning offering resilience from trending cyber-attacks, a review,” in 2021 International Conference on Computational Performance Evaluation (ComPE). IEEE, 2021, pp. 741–749.

[54] T. Cymru, “Analyzing ransomware negotiations with conti: An in-depth analysis.”

[55] K. Oosthoek, J. Cable, and G. Smaragdakis, “A tale of two markets: Investigating the ransomware payments economy,” arXiv preprint arXiv:2205.05028, 2022.

[56] A. Hobbs, The colonial pipeline hack: Exposing vulnerabilities in us cybersecurity. SAGE Publications: SAGE Business Cases Originals, 2021.

[57] A. J. Menezes, P. C. v. Oorschot, and S. A. Vanstone, Handbook of Applied Cryptography. Boca Raton: CRC Press, May 2020.

[58] “Intezer analyzer.” [Online]. Available: https://analyze.intezer.com

[59] “Automated malware analysis - joe sandbox,” https://www.joesandbox.com/#windows, (Accessed on 06/23/2022).

[60] Mandiant, "Flare-vm," https://github.com/mandiant/flare-vm, (Accessed on 06/23/2022).

[61] HHS - United States Department of Health and Human Services, "The evolution of ryuk," 2021. [Online]. Available: https://www.hhs.gov/sites/default/files/ryuk-variants.pdf

[62] ANSII - Agence nationale de la sécurité des systèmes d'information, "Ryuk Ransomware," 2021. [Online]. Available: https://www.cert.ssi.gouv.fr/uploads/CERTFR-2021-CTI-006.pdf

[63] McAfee, "Ryuk Ransomware Now Targeting Webservers," 2021.

[64] A. Honig and M. Sikorski, Practical Malware Analysis: The Hands-On Guide to Dissecting Malicious Software, 2012.

[65] Microsoft, "ZwQueryInformationProcess function." [Online]. Available: https://docs.microsoft.com/en-us/windows/win32/procthread/zwqueryinformationprocess

[66] V. UNTERFINGHER, "Ryuk ransomware: Origins, operation mode, mitigation," 2021. [Online]. Available: https://heimdalsecurity.com/blog/ryuk-ransomware/

[67] B. H. Itay Cohen, "Ryuk ransomware: A targeted campaign breakdown," 2018. [Online]. Available: https://research.checkpoint.com/2018/ryuk-ransomware-targeted-campaign-break/

[68] R. Ploszek, P. Švec, and P. Debnár, "Analysis of encryption schemes in modern ransomware," Rad Hrvatske akademije znanosti i umjetnosti: Matematičke znanosti, no. 546= 25, pp. 1–13, 2021.

[69] V. C. Craciun, A. Mogage, and E. Simion, "Trends in design of ransomware viruses," pp. 259–272, 2019.

[70] "What is Ryuk Ransomware? The Complete Breakdown," Jan. 2019. [Online]. Available: https://www.crowdstrike.com/blog/big-game-hunting-with-ryuk-another-lucrative-targeted-ransomware/

[71] "Conti locker source code," https://github.com/Cracked5pider/conti_locker/, (Accessed on 06/20/2022).

[72] "Conti Unpacked: Understanding Ransomware Development as a Response to Detection – a Detailed Technical Analysis." [Online]. Available: https://assets.sentinelone.com/ransomware-enterprise/conti-ransomware-unpacked

[73] “Bypassing Cylance and other AVs/EDRs by Unhooking Windows APIs.” [Online]. Available: https://www.ired.team/offensive-security/defense-evasion/bypassing-cylance-and-other-avs-edrs-by-unhooking-windows-apis

[74] Minerva Labs, “Conti Ransomware - Built to bypass EDRs, Prevented by Minerva.” [Online]. Available: https://blog.minerva-labs.com/conti-ransomware-built-to-bypass-edrs-prevented-by-minerva

[75] R. Research and I. F. Team, “Conti-nuation: methods and techniques observed in operations post the leaks,” Mar. 2022. [Online]. Available: https://research.nccgroup.com/2022/03/31/conti-nuation-methods-and-techniques-observed-in-operations-post-the-leaks/

[76] “Namespace root/cimv2 - powershell.one.” [Online]. Available: https://powershell.one/wmi/root/cimv2

[77] C. Dong, “Conti Ransomware,” Dec. 2020. [Online]. Available: https://cdong1012.github.io//reverse%20engineering/2020/12/15/ContiRansomware/

[78] “TAU Threat Discovery: Conti Ransomware,” Jul. 2020. [Online]. Available: https://blogs.vmware.com/security/2020/07/tau-threat-discovery-conti-ransomware.html

[79] “XChaCha20 Encryption vs AES-256: What’s the Difference?” [Online]. Available: https://nordpass.com/blog/xchacha20-encryption-vs-aes-256/

[80] BBC, “Cyber attack ’most significant on irish state’ - bbc news,” https://www.bbc.com/news/world-europe-57111615, 2021, (Accessed on 06/23/2022).

[81] PwC, “conti-cyber-attack-on-the-hse-full-report.pdf,” https://www.hse.ie/eng/services/publications/conti-cyber-attack-on-the-hse-full-report.pdf, 2022, (Accessed on 06/23/2022).

[82] M. McGee, “Report dissects conti ransomware attack on ireland’s hse,” https://www.govinfosecurity.com/report-dissects-conti-ransomware-attack-on-irelands-hse-a-18102, 2022, (Accessed on 06/23/2022).

[83] S. Gatlan, “Hhs: Conti ransomware encrypted 80% of ireland’s hse it systems,” https://www.bleepingcomputer.com/news/security/hhs-conti-ransomware-encrypted-80-percent-of-irelands-hse-it-systems/, 2022, (Accessed on 06/23/2022).

[84] HHS CYBERSECURITY PROGRAM, "202202031300_lessons learned from the hse attack_tlpwhite_r," https://www.hhs.gov/sites/default/files/lessons-learned-hse-attack.pdf, 2022, (Accessed on 06/23/2022).

[85] R. Richardson and M. M. North, "Ransomware: Evolution, mitigation and prevention," International Management Review, vol. 13, no. 1, p. 10, 2017.

[86] X. Luo and Q. Liao, "Awareness education as the key to ransomware prevention," Information Systems Security, vol. 16, no. 4, pp. 195–202, 2007.

[87] M. Humayun, N. Jhanjhi, A. Alsayat, and V. Ponnusamy, "Internet of things and ransomware: Evolution, mitigation and prevention," Egyptian Informatics Journal, vol. 22, no. 1, pp. 105–117, 2021.

[88] S. Thakur, S. Chaudhari, and B. Joshi, Ransomware: Threats, Identification and Prevention. John Wiley & Sons, Ltd, 2022, p. 361–387. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119795667.ch16

[89] HSS Gov, "Summary of the hipaa security rule | hhs.gov," https://www.hhs.gov/hipaa/for-professionals/security/laws-regulations/index.html, 2022, (Accessed on 06/16/2022).