

# **Behavioural Analysis of Current Evolution Ransomware Attack Exfiltration Methods**

ALESSANDRO RAVIZZOTTI

Submitted in partial fulfilment of the requirements of  
Edinburgh Napier University  
for the degree of  
Master of Science in Advance Security & Digital Forensics

School of Computing

April 2024

### MSc dissertation check list

Student Name: Alessandro Ravizzotti	Matric: 40621497
--	---------------------

Please insert this form, loose-leaf, into each copy of your dissertation submitted for marking.

Milestones	Date of completion	Target deadline
Proposal	n/a	Week 3
Initial report	n/a	Week 7
Full draft of the dissertation	n/a	2 weeks before final deadline

Learning outcome	The markers will assess	Pages <sup>1</sup>	Hours spent
<b>Learning outcome 1</b> Conduct a literature search using an appropriate range of information sources and produce a critical review of the findings.	* Range of materials; list of references * The literature review/exposition/background information chapter	73-79 6-24	200
<b>Learning outcome 2</b> Demonstrate professional competence by sound project management and (a) by applying appropriate theoretical and practical computing concepts and techniques to a non-trivial problem, <u>or</u> (b) by undertaking an approved project of equivalent standard.	* Evidence of project management (Gantt chart, diary, etc.) * Depending on the topic: chapters on design, implementation, methods, experiments, results, etc.	80-81 25-62	200
<b>Learning outcome 3</b> Show a capacity for self-appraisal by analysing the strengths and weakness of the project outcomes with reference to the initial objectives, and to the work of others.	* Chapter on evaluation (assessing your outcomes against the project aims and objectives) * Discussion of your project's output compared to the work of others.	62-65 62-65	100
<b>Learning outcome 4</b> Provide evidence of the meeting learning outcomes 1-3 in the form of a dissertation which complies with the requirements of the School of Computing both in style and content.	* Is the dissertation well-written (academic writing style, grammatical), spell-checked, free of typos, neatly formatted. * Does the dissertation contain all relevant chapters, appendices, title and contents pages, etc. * Style and content of the dissertation.		80
<b>Learning outcome 5</b> Defend the work orally at a viva voce examination.	* Performance * Confirm authorship		1 hour

Have you previously uploaded your dissertation to Turnitin? No

Has your supervisor seen a full draft of the dissertation before submission? Yes

Has your supervisor said that you are ready to submit the dissertation? No

<sup>1</sup> Please note the page numbers where evidence of meeting the learning outcome can be found in your dissertation.

---

## Authorship Declaration

I, Alessandro Ravizzotti, confirm that this dissertation and the work presented in it are my own achievement.

Where I have consulted the published work of others this is always clearly attributed;

Where I have quoted from the work of others the source is always given. With the exception of such quotations this dissertation is entirely my own work;

I have acknowledged all main sources of help;

If my research follows on from previous work or is part of a larger collaborative research project I have made clear exactly what was done by others and what I have contributed myself;

I have read and understand the penalties associated with Academic Misconduct.

I also confirm that I have obtained **informed consent** from all people I have involved in the work in this dissertation following the School's ethical guidelines

Type name: Alessandro Ravizzotti

Date: 16/03/2024

Matriculation no: 40621497

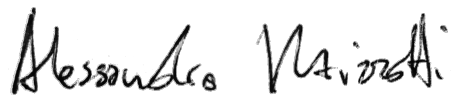
---

## General Data Protection Regulation Declaration

Under the General Data Protection Regulation (GDPR) (EU) 2016/679, the University cannot disclose your grade to an unauthorised person. However, other students benefit from studying dissertations that have their grades attached.

Please sign your name below one of the options below to state your preference.

The University may make this dissertation, with indicative grade, available to others.

A handwritten signature in black ink, reading "Alessandro Miondi". The signature is written in a cursive style with a large initial 'A'.

The University may make this dissertation available to others, but the grade may not be disclosed.

The University may not make this dissertation available to others.

# Abstract

Ransomware attacks have increased dramatically in recent years. With the evolution of ransomware, they started to carry out both encryption and exfiltration of the victim's data, thus creating double extortion. In this way the attacker demands the payment of multiple ransoms from the victim. There is currently a lot of research in the literature regarding crypto-ransomware but that regarding exfiltration-based ransomware is limited.

This work analyzes and evaluates the current ransomware datasets and investigates behavioural analysis of the ransomware exfiltration attack phase. The analysis of malicious datasets is carried out by creating a tool that allows the user to analyze and classify ransomware exfiltration samples. The analysis is carried out using the information saved in the VirusTotal and Malware Bazaar databases and using the AVclass2 tool. The behavioural analysis of exfiltration-based ransomware is carried out by running real ransomware exfiltration samples, and through the simulation of ransomware exfiltration methods including Living off the Land.

During the exfiltration dataset analysis, more than 1800 samples taken from the malicious datasets of other papers and from online databases were analyzed. From the results obtained, more than 18.50% of the samples appear to be incorrectly classified in a specific ransomware family. Furthermore, behavioural analysis was carried out on 35 samples belonging to 12 exfiltration-based ransomware families. Also, ransomware exfiltration methods were simulated for families which use Living off the Land behaviour. Through behavioural analysis, it was possible to validate the classification of the dataset used, identify two cases in which some of the information present in the databases was incorrect, and evaluate the exfiltration methods across a range of different metrics.

# Contents

<b>Abstract</b>	<b>i</b>
<b>List of Figures</b>	<b>iv</b>
<b>List of Tables</b>	<b>v</b>
<b>Acknowledgement</b>	<b>vi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Aims and Objectives . . . . .	3
1.2 Ethical Compliance . . . . .	4
1.3 Structure . . . . .	4
<b>2 Literature Review</b>	<b>6</b>
2.1 Introduction . . . . .	6
2.2 Ransomware Taxonomy . . . . .	6
2.2.1 Ransomware Classification . . . . .	6
2.2.2 Ransomware Attack Chain and Phases . . . . .	8
2.2.3 Ransomware Evolution . . . . .	9
2.3 Data Exfiltration . . . . .	11
2.3.1 Data Exfiltration Techniques . . . . .	11
2.3.2 Target Victim Data . . . . .	12
2.3.3 Ransomware Attack Exfiltration . . . . .	12
2.3.4 Ransomware Data Exfiltration Network Protocols . . . . .	13
2.3.5 Ransowmare Data Exfiltration Methods . . . . .	14
2.4 Experimental Methods . . . . .	16
2.4.1 Ransomware Analysis . . . . .	17

2.4.2	Exfiltration Ransomware Environment . . . . .	20
2.4.3	Target Victim Dataset . . . . .	21
2.4.4	Exfiltration Ransomware samples . . . . .	21
2.5	Conclusion . . . . .	22
<b>3</b>	<b>Methodology and Design</b>	<b>25</b>
3.1	Introduction . . . . .	25
3.2	Research Methodology . . . . .	25
3.3	Experimental Methodology . . . . .	26
3.4	Experimental Design . . . . .	28
3.4.1	Testing Environments . . . . .	29
3.4.2	Data Collection and Analysis Tools . . . . .	33
3.4.3	Target Victim Dataset . . . . .	34
3.4.4	Malicious Dataset . . . . .	34
3.5	Conclusion . . . . .	37
<b>4</b>	<b>Experiments, Results and Evaluation</b>	<b>39</b>
4.1	Introduction . . . . .	39
4.2	Exfiltration Ransomware Analysis . . . . .	39
4.2.1	Script to Gather Sample Metadata . . . . .	40
4.2.2	Results . . . . .	48
4.3	Real Ransomware Exfiltration Experiment . . . . .	50
4.3.1	Dynamic Analysis . . . . .	51
4.3.2	Static Analysis . . . . .	58
4.4	Ransomware Exfiltration Simulation . . . . .	60
4.4.1	Results . . . . .	60
4.5	Experiments Evaluation . . . . .	62
4.5.1	Ransomware Classification . . . . .	62
4.5.2	Real Ransomware Exfiltration . . . . .	64
4.5.3	Ransomware Simulation Exfiltration . . . . .	65
4.6	Conclusion . . . . .	65
<b>5</b>	<b>Conclusion</b>	<b>67</b>
5.1	Aims and Objectives . . . . .	68

5.2	Objective 1: Literature Review . . . . .	69
5.3	Objective 2: Methodology and Design . . . . .	69
5.4	Objective 3: Experiments, Results and Evaluation . . . . .	70
5.5	Limitations and Future Work . . . . .	71
<b>References</b>		<b>73</b>
<b>Appendices</b>		<b>80</b>
	Appendix 1: Project Management . . . . .	80
	Appendix 2: Scripts source code on GitHub . . . . .	82
	Appendix 3: Exfiltration-based Ransomware samples used . . . . .	83



# List of Figures

1.1	Cyber Kill Chain . . . . .	2
2.1	Ransomware Phases . . . . .	8
2.2	Timeline of Evolution of New Ransomware Families (Gómez Hernández et al. 2023) . . . . .	10
3.1	Research Methodology . . . . .	26
3.2	Testing Environment After the Initial Setup . . . . .	30
3.3	Testing Environment for Real Exfiltration Ransomware . . . . .	32
3.4	Testing Environments for Simulation . . . . .	32
4.1	Script Flowchart . . . . .	41
4.2	Output Folder Structure . . . . .	45
4.3	Example of "ransomware-metadata*.csv" . . . . .	45
4.4	Gather Hashes from Malware Bazaar . . . . .	47
4.5	Webpage to Clean and Format the Data for the Python Script . . . .	48
4.6	Execution of Conti Sample . . . . .	52
4.7	Darkside Sample Wrongly Classified as Blackmatter . . . . .	56
4.8	Wireshark: ExMatter Network Traffic . . . . .	58
4.9	Stealbit Decompiled Code from Ghidra . . . . .	59
4.10	Simulation of Exfiltration to an FTP Server . . . . .	62
7.1	Project Management - Gantt Chart . . . . .	80
7.2	Example of diary entry . . . . .	81
7.3	Scripts Source Code on GitHub . . . . .	82

# List of Tables

2.1	Ransomware Families that Perform Data Exfiltration, from 2019 to 2023 . . . . .	13
2.2	Ransomware Exfiltration Methods and Cloud Storage. . . . .	15
2.3	ExMatter's Target File Extension. . . . .	16
3.1	Testing Environment Specification . . . . .	31
3.2	Target Dataset Information . . . . .	34
3.3	Gathered Fields from Databases by the Python Script . . . . .	35
3.4	Real Exfiltration-based Ransomware Samples and Exfiltration Modules . . . . .	36
3.5	Rclone Exfiltration Flags from Different Ransomware Families . . .	37
4.1	Ransomware Exfiltration Dataset . . . . .	49
4.2	Analysis of multiple datasets . . . . .	50
4.3	Results of Dynamic Analysis . . . . .	54
4.4	Samples Classification . . . . .	55
4.5	Relevant Network Activities . . . . .	57
4.6	Simulation of Ransomware Exfiltration . . . . .	61
7.1	Exfiltration-based Ransomware samples used . . . . .	83

# Acknowledgement

I would like to thank Rich Macfarlane for continuously guiding me and motivating me in the development of this project.

Special thanks to the ransomware research group who provided me with so many ideas for this work and helped me achieve my goals.

My final thanks is to my family who has always supported me throughout the whole Master's degree.

# Chapter 1

## Introduction

A type of malware that has become increasingly important in recent years is ransomware extortion attacks. This is due to the continuously growing number of attacks of this type of malware and the ever-increasing economic loss. The first major ransomware attack was caused in 2014 by Cryptowall which infected over 600,000 systems in approximately 40 countries with an estimated total loss of \$1 million (Oz et al. 2022). In 2015 the first Ransomware as a Service (RaaS) were born, paid services which allow non-expert attackers to create a personal version of ransomware, increasing the number of ransomware variants existing for individual families (Gómez Hernández et al. 2023). Subsequently, in May 2017, a new ransomware family called WannaCry infected over 250,000 computers in over 150 countries causing a global loss of approximately \$5 billion, leading ransomware to be considered one of the top malware threats of 2018 (Tang et al. 2020). In recent years, the targets of ransomware attacks have shifted from individual users to companies or organizations which has led to the increase in ransom demand. Furthermore, new families have been born and the number of ransomware attacks has increased dramatically. In 2017, just over 183 million attacks were recorded; this value grew by 165% in 2020 reaching approximately 305 million attacks and, the following year (2021) the number of attacks more than doubled reaching over 620 million (Chainalysis 2024).

Ransomware is a type of malware that steals and/or blocks access to the victim user's data and, asks for a ransom in cryptocurrency in exchange for guarantee-

ing access to the victim user's information and not publishing the stolen data (Hou et al. 2024). Typically three categories of ransomware are defined as follows: Crypto-Ransomware, Locker-Ransomware and Exfiltration-based ransomware (Oz et al. 2022). Exfiltration-based ransomware is the most recently created category of ransomware and its aim is to steal data saved on the victim machine and then ask for a ransom so as not to publish or sell the data online (Almeida & Vasconcelos 2023). Initially, ransomware usually fell into a single category but in recent years, there are more and more ransomware that perform double extortion attacks. Therefore, the ransomware requests multiple ransoms, incentivizes the victim to pay and possibly has other methods of profit such as selling the data online. This form of ransom has become more common recently because it takes less time to complete the malicious activities and it is easier to evade compared to crypto-ransomware.

In the literature, the Cyber Kill Chain (CKC) (Lenaerts-Bergmans 2022) or the MITRE ATT&CK framework are often used to describe the different phases of a ransomware attack (Gómez Hernández et al. 2023). In this way, researchers have the same method to categorize the different behaviours of malware and better understand their phases. Figure 1.1 depicts the phases of the CKC. The phases concerning the ransomware exfiltration and encryption are located in Phase 7: Actions on Objective.

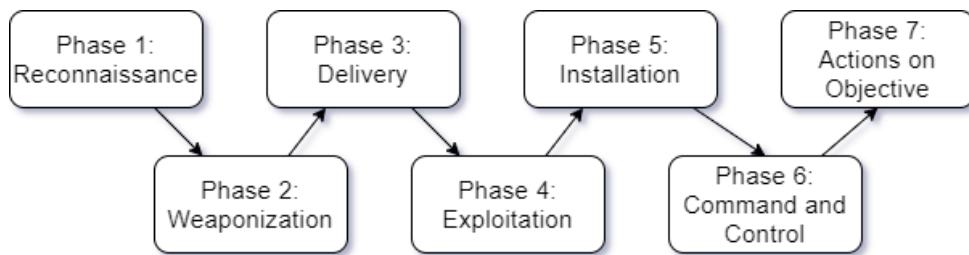


Figure 1.1: Cyber Kill Chain

Currently, research works are mainly focused on the encryption phase regarding Crypto-Ransomware. Several studies perform static or dynamic analysis of ransomware in order to collect as much information as possible regarding the different families. This data is used to analyze the behaviour of the ransomware and

to create datasets to train detection systems, many based on machine learning. Some work has been done around the ransomware exfiltration phase, but there are limited datasets of exfiltration methods and exfiltration samples. Furthermore, a topic that has not been explored much concerns the selection methods of the ransomware datasets used by researchers. For these reasons, this dissertation focuses on the analysis and classification of the ransomware samples which include exfiltration modules. Moreover, this work aims to produce and analyse useful behavioural analyses of a range of Exfiltration-based ransomware families and methods.

## 1.1 Aims and Objectives

This project aims to evaluate the behavioural analysis of the ransomware exfiltration methods, by running real ransomware exfiltration samples, and through the simulation of ransomware exfiltration methods including Living off the Land (LOTL). Furthermore, this research aims to analyse and evaluate the current ransomware datasets in order to analyse and classify ransomware exfiltration samples.

In order to achieve these aims, the following objectives are defined:

1. Carry out a literature review on the topic of ransomware, focusing on the exfiltration phase. Identify the methods, tools and scripts used by ransomware exfiltration to steal data.
2. From the knowledge acquired from the literature review, develop a methodology and an experimental design to analyse ransomware datasets and the behaviour of ransomware exfiltration.
3. Based on the methodology and design, develop the experiment regarding dataset analysis, real ransomware exfiltration and the simulation of ransomware exfiltration through LOTL. Further, analyse and critically evaluate the results and of the experiment, and discuss with reference to other work to this area.

## 1.2 Ethical Compliance

Care must be taken when working with malware and therefore potential complexities regarding this area need to be assessed. Malware is developed to perform malicious activities and spread to other devices. More precisely, the malware considered in this paper, called ransomware, encrypts and exfiltrates a device's data and usually tries to infect other devices within the network. For this reason, precautions must be taken to ensure that none of the malware can spread beyond the scope during the experiment.

The following precautions have been taken to secure the malware experimentation:

- Ransomware execution only occurs in an isolated and controlled environment.
- Malware is compressed and password-protected until it reaches the isolated environment.
- The controlled exfiltration of data directed towards the outside network is only simulated using legitimate software of which the executed code and destination are known.
- No files are transferred from the isolated environment to the outside after the malware has been unpacked and executed.

## 1.3 Structure

The paper is organized as follows:

**Chapter 2:** Outline a literature review of ransomware and its data exfiltration phase. Information is provided regarding the taxonomy of ransomware, its evolution, focusing on the exfiltration phase and therefore on the category of double extortion ransomware. Furthermore, the differences between an information stealer and data exfiltration ransomware are analyzed. In addition, the methodologies, testing environments, datasets used in other papers and ransomware exfiltration methods are compared and analyzed in order to carry out testing in this area.

**Chapter 3:** Shows the methodology and design used, justifying the choices made. The parts of the experiment design that include the testing environment, methods

for collecting data, the target dataset, and the malicious datasets are described.

**Chapter 4:** Describes the experiments carried out regarding the analysis of datasets, dynamic and static analysis of ransomware exfiltration and simulation of exfiltration methods used by ransomware. Furthermore, the results achieved by the different experiments are shown, evaluated and critically analysed by comparing them with other papers related to the study area.

**Chapter 5:** Concludes the study of this paper. Outline the findings in the literature review, the methodology, the design and the experiments carried out. Furthermore, the objectives achieved, limitations and future work in this area are highlighted.



## Chapter 2

# Literature Review

### 2.1 Introduction

This chapter presents the results of the literature review. This includes ransomware classification, the phases that form a ransomware attack and related literature. Then the topic of data exfiltration is explored, starting from a general point of view including general exfiltration research and then focusing on ransomware exfiltration. Finally, research regarding the experimental methods used by other papers regarding ransomware and data exfiltration is examined, towards the methodology for this work.

### 2.2 Ransomware Taxonomy

Ransomware is a type of malware that steals data and/or blocks access to the victim user's file or device and, usually asks for a ransom in cryptocurrency in exchange for giving the victim access to their files or device, again and not publishing the stolen data online (Oz et al. 2022). If the ransom is not paid, the victim will not be able to access their files and their contents could be published online or sold on the dark web.

#### 2.2.1 Ransomware Classification

Typically ransomware is classified into three categories: Crypto-Ransomware, Locker-Ransomware and Exfiltration-based Ransomware, sometimes Leakware or Doxware names are also used (Oz et al. 2022). Crypto-Ransomware is the most

widespread category and of which there are more studies in the literature. This type of ransomware encrypts, partially or totally, the files of the victim machine, making them inaccessible to the user unless they know the encryption keys (Gómez Hernández et al. 2023). Locker-Ransomware blocks the victim user from accessing their infected machine. This type does not usually perform any encryption, which is why it is sometimes possible to recover files from the victim machine and, after restoring the system, continue to use the no longer infected machine (Razaulla et al. 2023). Exfiltration-based ransomware is the most recently created category of ransomware. It aims to exfiltrate data saved on the victim machine and then ask for a ransom in exchange for not publishing or selling the stolen data online (Almeida & Vasconcelos 2023). Exfiltration-based ransomware attacks have recently become more common compared to crypto-ransomware. This is because it is easier to evade the detection system and it takes less time to complete the malicious activities. This type of ransomware is often used against organisations, medical institutions and governments as they hold private and sensitive data and possible leaks can cause serious economic and image damage to the affected company. Initially, ransomware usually fell into a single category but in recent years, there are more and more ransomware that perform more than one malicious behaviour. This is done to request multiple ransoms and so perform a double extortion, to incentivize the victim to pay and possibly to have other methods of profit by selling the user's data online (Gómez Hernández et al. 2023). Lately, it is increasingly common to find ransomware families that are simultaneously Crypto-Ransomware and Exfiltration-based Ransomware which therefore steal and encrypt the files of the infected machine, thus carrying out a double extortion.

A further classification of ransomware concerns families. Each family has defined phases, code signatures and uses the same commands and tools, with some variation from one version to another. The introduction of families has been very important since the birth of RaaS as multiple variants of the same ransomware are created which will all have components in common.

### 2.2.2 Ransomware Attack Chain and Phases

The attack phases are common to all ransomware families. In the Cyber Kill Chain, shown in Figure 1.1, the first phase corresponds to the Reconnaissance phase in which the attacker identifies the target and searches the network for the presence of weaknesses or vulnerabilities (Gómez Hernández et al. 2023). In the second phase, an attack vector is created which allows the vulnerabilities found to be exploited to gain access to the target machine. In the Delivery phase, ransomware is sent to the target and in the next two phases, which are called Exploitation and Installation, the malicious code is executed and installed on the system. In the sixth phase, the Command and Control server is contacted for exchanges of information and possible lateral movements. Finally, in phase 7: Actions on Objective, the exfiltration and encryption phase of the ransomware is activated (Lenaerts-Bergmans 2022).

In order to better understand the phases of the attack vector relating only to ransomware, Figure 2.1 below shows the ransomware phases which are usually carried out by the latest generations of ransomware which have both an exfiltration and an encryption phase. Created by the author, it is based on the most relevant phases of the MITRE ATT&CK framework, for a Exfiltration-based ransomware (TrendMicro 2023*b,a*).

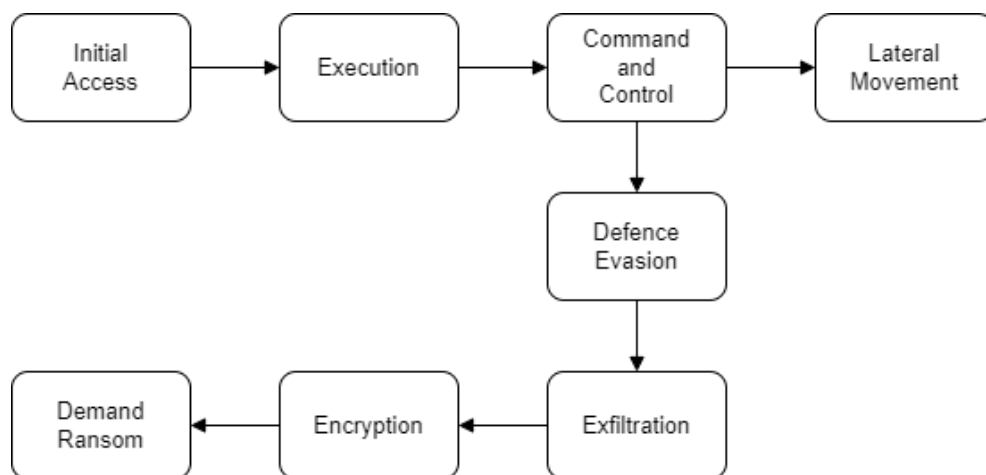


Figure 2.1: Ransomware Phases

During the first two phases, Initial Access and Execution, the ransomware arrives on the target machine and begins to carry out various actions including Credential Access, Privilege Escalation, Persistence and Network Discovery. Subsequently, the C&C server is contacted to exchange cryptographic keys and further information. At this point, there may be Lateral Movements which allow the ransomware to spread to other devices within the same network, increasing the impact of the attack (TrendMicro 2021*a*). Defence Evasion is another ransomware preparation phase during which techniques are used to prevent the ransomware from being detected by the security systems of the infected machine. Subsequently, the Exfiltration phase occurs, the phase on which this work focuses, in which various information is stolen from the victim computer and sent to the C&C server or other external cloud services (TrendMicro 2022*a*). In the next phase, Payload Execution, the ransomware payload is executed and consequently files on the victim machine are encrypted. Finally, the ransomware displays ransom-demand messages via the desktop wallpaper and the creation of multiple text files. This last phase also includes paying the ransom and subsequently deciphering the encrypted files.

### 2.2.3 Ransomware Evolution

The number of ransomware attacks has grown enormously since the birth of this type of malware in the 1980s and with the development of new ransomware families, there have been several evolutions in the methods and technologies used by attackers, shown in Figure 2.2. Those have resulted in greater gains against them and in ever-increasing economic loss for the people and companies affected by the attacks (Razaulla et al. 2023). Initially, the main problem with ransomware was the method of paying the ransom anonymously, which was practically impossible with the technologies of the time. This factor changed with the advent of cryptocurrencies which allowed attackers to obtain pseudo-anonymous ransom payments and consequently, in the following years, the number of ransomware attacks increased dramatically. In fact, approximately 60,000 new ransomware families were detected in 2011 (Oz et al. 2022).

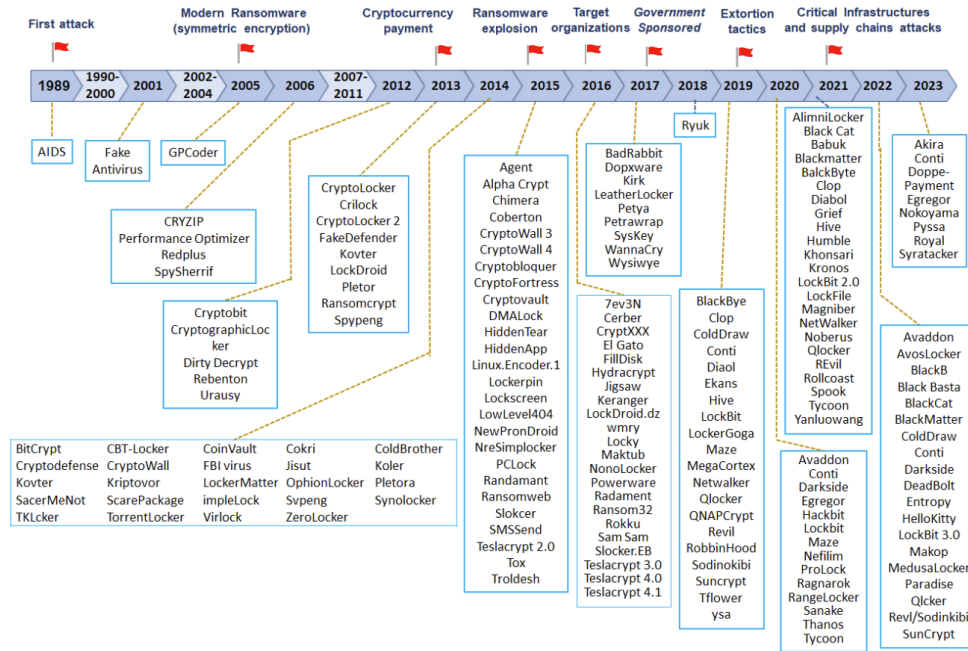


Figure 2.2: Timeline of Evolution of New Ransomware Families (Gómez Hernández et al. 2023)

The impact and growth of ransomware further increased in 2015 when a new business model was born called Ransomware as a Service (RaaS), a paid service in which expert groups of attackers, called *operators*, take care of the creation of the malware, its infrastructure and sell these services to people who are not necessarily expert called *affiliates* (Gómez Hernández et al. 2023). This has allowed the creation of multiple versions of ransomware, as well as a visible increase in attacks carried out using this type of malware. Another evolution of ransomware occurred with the union of information stealers with ransomware, creating the category of Exfiltration-based Ransomware (Gómez Hernández et al. 2023). This step had a major impact as it was found that victims of these types of ransomware attacks are more inclined to pay ransom than Crypto-Ransomware. This is likely because both the affected individuals and companies find it more important to not have their data made public than to regain access to their own data. Furthermore, ransomware attacks have begun to carry out not only the encryption of data, but also their exfiltration, thus leading to the birth of double extortion. It allows the attacker to demand the payment of multiple ransoms. One ransom will be used to unlock the encrypted data on the victim device, while the second ransom

will be used to ensure that the stolen data is not being published online (Meurs et al. 2024). A further change in ransomware that has taken place in recent years concerns the targets of the attacks. Initially, the main targets were individual users but currently ransomware attacks are mainly focused on companies, government bodies and medical institutions (Oz et al. 2022). This change in targeting has further increased for hospitals with the advent of COVID-19 (Gómez Hernández et al. 2023). In this way, the attacker can demand a higher ransom as the impact of the success of the attack will be greater. For the affected company, the monetary loss is not only equivalent to the amount of the ransom but also the downtime period of its services and damage to its image.

## 2.3 Data Exfiltration

Nowadays, data is increasingly important both for individual users and for government bodies, hospitals, banks, etc. For this reason, data is continuously more the target of attacks and exfiltration. Data exfiltration, also known as data extrusion or simply data stealing, is the intentional unauthorized transfer of data from a device, also called victim device, to another storage device. Exfiltration can be conducted either automatically, using scripts, or manually (IBM 2023). It is very important not to confuse data exfiltration with data leakage or data breach. Data leakage is usually caused by a technical security error which therefore accidentally exposes private data to the public, while a data breach corresponds to unauthorized access to private data. The substantial difference is therefore that data exfiltration is both the unauthorized access and transfer of private data from one device to another (IBM 2023).

### 2.3.1 Data Exfiltration Techniques

To carry out data exfiltration, there are multiple techniques used by attackers (Aboze 2024). In this dissertation, the focus is only on techniques via the network and not physical attacks. Below are listed some of the most used techniques to steal data using the network:

- **Malware/Ransomware.** Which will be discussed further down.
- **Fileless Malware and Living off the Land (LOTL) Attacks.** This

attack uses legitimate programs or scripts already present on the victim machine to perform data transfer. This type of attack is very difficult to identify as it is seen as legitimate traffic on the network.

- **Remote Access Tools (RATs)**. Software that allows an attacker to control a victim device remotely.
- **Phishing**. A social engineering technique, one of the most common methods for carrying out data stealing.
- **SQL Injection (SQLI)**. Attack that allows you to view, modify and delete data in a database.
- **Cross-Site Scripting (XSS)**. Attack that allows you to insert codes into a web page.
- **Man in the Middle Attacks (MITM)**. This attack is carried out by intercepting the communication between two parties. In this way, the attacker could be able to steal login credentials or other relevant information.

### 2.3.2 Target Victim Data

The data that is usually targeted changes based on the technique used, whether the target is a single user or whether the victim under attack is a company. The target data is usually different if we are talking about an info-stealer or an Exfiltration-based Ransomware. Infostealers are usually Trojans or keyloggers that have the sole purpose of stealing data, such as user credentials and login information, to carry out other attacks, stealing wallets and selling the information found online (TrendMicro 2020a). Exfiltration-based Ransomware has instead the objective of getting a ransom paid and carrying out double extortion. Ransomware data exfiltration usually targets sensitive data and not all files present on the victim machine in order to reduce the chances of being detected and further incentivize the payment of the ransom (Hou et al. 2024). Typically the target files are the same files targeted during the encryption of the crypto-ransomware.

### 2.3.3 Ransomware Attack Exfiltration

As mentioned previously, data exfiltration is one of the latest evolutions that ransomware had in recent years. As shown in Figure 2.1, since 2019 there have

been a lot of families that perform data stealing either as single extortion or double, thus also carrying out the encryption of the files or blocking access to the data or device (Agcaoili et al. 2021).

Ransomware Families				
AgeLocker	Cl0p	Hive	NetWalker	Royal
Akira	Conti	LockBit	Play	Ryuk
AlumniLocker	DarkSide	Maze	ProLock	Sekhmet
Avaddon	DoppelPaymer	MedusaLocker	REvil	Snatch
Black Basta	Egregor	Mespinoza	RagnarLocker	SunCrypt
BlackByte	Ekans	MountLocker	Ragnarok	Thanos
BlackCat	Everest	Nefilim	RansomExx	Xinof

Table 2.1: Ransomware Families that Perform Data Exfiltration, from 2019 to 2023

### 2.3.4 Ransomware Data Exfiltration Network Protocols

While multiple families of ransomware perform data exfiltration, there are a limited number of network protocols that are used to steal files (Sabir et al. 2021):

- **HTTP/HTTPS.** Usually, a POST request is made which contains the contents of the files or other information to be exfiltrated in the *Data* field.
- **FTP/SFTP.** The data is stolen through the use of a regular FTP server.
- **DNS Tunnel.** The data to be exfiltrated is first encoded and divided into blocks of 512 bytes so as it does not exceed the limits of the UDP datagram size. After that, each encoded block is joined to the URL to be sent to the DNS in the *Query String* field.
- **ICMP.** The content of the data to be exfiltrated is divided into chunks and inserted into the *Data* field of the ICMP echo packets.

While the (S)FTP protocol can be suspicious and may be blocked by some Intrusion Detection and Prevention Systems (IDPS), the HTTP(S), DNS and ICMP protocols are more difficult to be identified as suspicious packets by the IDPS as they are highly used protocols by every device connected to the network and not specifically designated for file transfer.



### 2.3.5 Ransomware Data Exfiltration Methods

Ransomware, during the exfiltration phase, uses small scripts written specifically for this phase or more recently external methods are used. In the past, ransomware has carried out data exfiltration activities mainly through the use of Living off the Land Methods, legitimate tools used by common users for file transfer. Recently ransomware that uses specific software for data exfiltration has become increasingly frequent. An important classification regarding ransomware data exfiltration concerns where the stolen data is transferred. In fact, based on the ransomware family and the methods used, the data can be sent to the C&C server, which can be identified either by a specific IP or by a URL, or to a cloud storage service such as Google Drive, Mega or Pcloud. The cloud storage service used most by ransomware attacks is *MEGA*.

#### Living off the Land Methods

Several legitimate software are used during the exfiltration phase by some ransomware families. These tools can be divided into three categories based on the actions they perform: compression, transfer and data storage. The two most used tools for compression are *WinRar* and *7zip* (TrendMicro 2021*b*, 2024*a,c*). Since they are very famous programs and widely used by common users, it is very likely that they are already installed on the victim device and their execution would not be suspicious. The tools for transferring data from the victim machine to external devices are many. *Rclone* is the most used LOTL method by ransomware, probably due to its versatility and ease of use. It is a command-line program that allows the user to manage files on cloud storage, allowing access to over 70 different cloud storage services and data transfer using standard protocols for third-party devices (Breitinger et al. 2022, TrendMicro 2022*a*, 2021*a*, 2023*d*). The second most legitimate tool found in ransomware analysis was *WinSCP*, an open-source (S)FTP client and file manager. Other methods used for data exfiltrate are *FileZilla*, an FTP application, and *MegaSync*, an application that allows the user to send files to the *MEGA* cloud storage service (TrendMicro 2023*a*, 2024*a*).

From the analysis of technical articles, some ransomware families have been taken

into observation: Akira, Black Basta, BlackByte, BlackCat, Cl0p, Conti, Darkside, Hive, Lockbit, Play and Royal. From the technical articles it emerged that six out of eleven of the ransomware families use the *Rclone* tool for data exfiltration (TrendMicro 2023a, 2022a, 2023b, 2021a,b, 2023d) and five out of eleven send the stolen data to the *MEGA* cloud service. Other legitimate methods that are used are *WinSCP* and *FileZilla* (TrendMicro 2024c). To compress the files to be sent, the *7-Zip* and *WinRAR* tools are often used. The families that stand out for exfiltration are BlackCat, Cl0p and LockBit which use specialized exfiltration software (TrendMicro 2022b, 2023c, 2024b). Additional information regarding ransomware exfiltration methods can be found in Table 2.2.

Ransomware	Exfiltration Tool	Compression	Cloud
Akira	Rclone, WinSCP FileZilla		
Black Basta	Rclone		
BlackByte	ExByte	WinRAR	MEGA
BlackCat (ov)	Rclone, WinSCP MEGASync	7-Zip	MEGA
BlackCat (lv)	ExMatter Module		
Cl0p	Dewmode		
Conti	Rclone, WinSCP		MEGA
Darkside	Rclone	7-Zip	MEGA
Hive	MEGAsync, AnonFiles SendSpace, uFile	7-Zip	MEGA
Lockbit 1.0	FreeFileSync		MEGA
Lockbit 2.0	StealBit		
Play	WinSCP	WinRAR	C&C w/ PHP
Royal	Rclone		

Table 2.2: Ransomware Exfiltration Methods and Cloud Storage.

In Table 2.2 if the cell is left empty it means that no information has been found about it. In the case of compression, it may be carried out directly by the exfiltration tool used. *ov* and *lv* mean older versions and latest versions respectively

## Ransomware Exfiltration Software

Ransomware family specific software modules for ransomware exfiltration are *ExByte*, *ExMatter* and *StealBit*. *ExByte* is the exfiltration tool used by the RaaS *BlackByte*. It is written in Golang and uploads files to the *MEGA* cloud storage (Symantec 2022). *ExMatter* is the exfiltration tool used by *BlackCat*. It is written in .NET, uploads files to a remote SFTP server and after execution runs a PowerShell script to delete its traces. The files that are exfiltrated must not belong to specific folders and need to have a specific extension, as shown in Table 2.3. This method for the selection of the files and exclusion of the folder is the same one used by crypto-ransomware. Moreover, the files must have a size between 4MB and 67MB. Furthermore, file exfiltration is prioritized by controlling the LastWrite-Time (Symantec 2021, Mayer 2022).

File Extensions					
.aspx	.doc	.jpg	.png	.rtf	.ts
.bmp	.docx	.js	.ppt	.sda	.txt
.config	.dwg	.json	.pptx	.sdm	.xls
.cs	.ipt	.msg	.pst	.sdw	.xlsx
.csv	.jpeg	.pdf	.rdp	.sql	.zip

Table 2.3: ExMatter’s Target File Extension.

*StealBit* is the exfiltration tool used by the RaaS *Lockbit*. There are different versions of this software and it is a very versatile program as there are multiple options to perform different actions such as changing the file and folder targets. File transfer is very fast and takes place via HTTP. This software, at the beginning of its execution, checks whether the computer is located or has the languages set in the former Soviet Union and in this case, no malicious behaviour is performed (Aleksandar & Kotaro 2021).

## 2.4 Experimental Methods

Several literature searches were carried out to better understand the current knowledge regarding ransomware and exfiltration. The papers through which this re-

search began are the survey papers of (Oz et al. 2022, Gómez Hernández et al. 2023, Benmalek 2024, Sabir et al. 2021). This type of article, in addition to explaining the fundamental and historical notions of the topic, aggregates many studies by other researchers present in the literature, comparing them with each other and highlighting critical points. Furthermore, the survey papers expose what future research is possible and therefore also the present gaps in the literature. Based on these papers, more in-depth research was carried out regarding ransomware, exfiltration and defence methods.

### 2.4.1 Ransomware Analysis

In order to carry out malware analysis, two different paths can be followed: static and dynamic analysis (Oz et al. 2022). *Static analysis* consists of analyzing the sample without executing it. This way you can get different information using a disassembler or other tools. Some papers such as Vasconcelos & Almeida (2023) used IDA Pro to perform static analysis and dynamic analysis. The main advantage is that it allows the researcher to safely analyze the sample. On the other hand, all the information that is dynamically generated by the malware cannot be seen and some data may not be readable due to obfuscation and encryption. Regarding ransomware searches, static analysis is usually performed if the paper focuses on a single ransomware or a small dataset such as (Eliando & Warsito 2023). *Dynamic Analysis* consists of executing the malware and analyzing its behaviour. In this case, it is necessary to have a safe environment such as a sandbox or a testbed to be able to analyze the sample. In this case, it is possible to see the actual behaviour of the malware with some exceptions. If the malware uses anti-debugging techniques or tries to figure out whether it is running on a real machine or in a sandbox, its behaviour may be skewed and not match normal behaviour. Regarding ransomware searches, dynamic analysis is used either to understand the behaviour of a family of ransomware or usually to create datasets that will be used by other researchers (Hussain et al. 2023) or be given as input to machine learning (ML) algorithms for the creation of an Intrusion Detection Prevention System (IDPS) (Mehrban & Geransayeh 2024). Both static and dynamic analysis have advantages and disadvantages, which is why it is sometimes advisable to carry out

both analyses to be able to get the most information from the samples (Oz et al. 2022, Hussain et al. 2023).

Since the encryption phase is the most frequently found in ransomware, consequently research in the literature also focuses mainly on this topic. Several papers have studied the possibility of finding the encryption keys in memory immediately after the execution of the payload (de Loaysa Babiano et al. 2023). This method, however, has become inefficient and in many cases unsuccessful as currently different keys are used to block access to different files. Furthermore, the methods for encrypting data are very fast and it means that the encryption keys do not remain in memory (Gómez Hernández et al. 2023). For this reason, late studies such as that of Hou et al. (2024) focus on identifying the ransomware attack before it can carry out malicious actions or ensuring that the identification is carried out very quickly during the offensive phases of the malware, so such as to lose the least amount of data possible.

From the research of Vasconcelos & Almeida (2023) the current gaps and difficulties regarding reverse engineering techniques used to analyze malware are discussed. In the study, a static and dynamic analysis is carried out, using the IDA Pro program and the LLaMa-12B language model, on 4 ransomware families that perform data exfiltration or encryption. This work managed to understand of how data exfiltration occurs by the AlphV and Black Basta ransomware. Below is the pseudocode found that can be used to carry out simulations.

```
1 // AlphV Ransomware Exfiltration Pseudocode
2 function exfiltrate_alphv() {
3     files = collect_sensitive_files();
4     compressed = compress(files);
5     encrypted = asym_encrypt(compressed, PUB_KEY);
6     stealth_upload("[URL]", encrypted);
7 }
```

```
1 // Black Basta Exfiltration Pseudocode
2 function exfiltrate_basta() {
3     docs = scan_filter_docs();
4     for doc in docs {
5         enc_doc = encrypt(doc, SYM_KEY);
6         packet = create_packet(enc_doc);
```

```
7     send_data("[URL]", packet);  
8 }  
9 }
```

There are not many papers in the literature that focus on network traffic generated by ransomware. Mehrban & Geransayeh (2024) used machine learning algorithms to identify and detect ransomware in computer networks. Features such as IPs, protocols used, number of packets and bytes exchanged per conversation were extrapolated from the network traffic. The research results showed a high accuracy rate in identifying ransomware and benign software. A negative aspect of this work is that it focuses on general ransomware, in fact, no distinction is made between exfiltration ransomware and other types of ransomware. Another paper that analyzes network traffic, but focusing on exfiltration ransomware, to detect ransomware activities is Almeida & Vasconcelos (2023). In this research, the primary objective is to understand the effectiveness of the Bidirectional Encoder Representations from Transformers (BERT) model to identify and analyze network traffic patterns regarding data exfiltration carried out by ransomware families. Furthermore, the paper highlights how the creation of network traffic datasets, generated in a simulated network environment, is fundamental to being able to understand and analyze the behaviour of ransomware. Almeida & Vasconcelos (2023), in its tests, monitored the 10 ransomware families for a period of 12 hours using a series of virtual machines with different characteristics, having a real-world network scenario, analyzing network traffic and investigating further methods for detecting ransomware. The features that were extracted consider the frequency of communication, the amount of stolen data and the percentage of network bandwidth used by the ransomware. From the results of the paper, it emerged that data exfiltration is selective and mainly focused on folders containing sensitive data. This behaviour, as highlighted by Almeida & Vasconcelos (2023), indicates that ransomware has specific exfiltration techniques and does not proceed to steal all the data present on the victim machine. In this way, not only less network traffic is generated, which would make exfiltration-based ransomware easier to detect, but the content of the stolen data is more relevant and consequently, the victim is more willing to pay the ransom. One of the most recent papers that deals with the topic of ransomware and exfiltration is the paper by Hou et al. (2024). The

study aimed to improve ransomware detection thorough multiple analysis. The study found that 90% data exfiltration is carried out using TCP, UDP and HTTP packets. In almost 22% of cases the leakware uses system APIs to transfer data and in 18% of exfiltrations cloud file-sharing tools are used and therefore the data is not sent directly to the C&C server. Regarding the results relating to file access, this study, like Almeida & Vasconcelos (2023), also found that in many cases the samples only access sensitive files.

### **2.4.2 Exfiltration Ransomware Environment**

When behavioural analysis of ransomware must be carried out in order to analyze their data exfiltration phase, different experimental environments can be used. First of all, it is essential to run real ransomware or other types of malware, only in isolated and controlled environments so as not to allow the malware to spread to machines other than the target machines. Furthermore, in the case of exfiltration-based ransomware it is important that the ransomware has no real connection to its C&C server so that information cannot be given to attackers Ozturk et al. (2024). There are three environments for analyzing the behaviour of ransomware. The use of a physical testbed allows the environment to be more similar to the real one but involves difficulties and requires longer testing times. The use of virtual environments by creating virtual machines. This option is very versatile and facilitates several steps in the testing phase, however making the environment less real than a physical computer. Finally, sandboxes such as AnyRun and Cuckoo exist online which allow you to run malware in complete safety and without having to worry about the experimental environment (Mehra & Pandey 2015, Kok et al. 2022, Agrawal et al. 2022). The disadvantage in this case is that you do not have full access to the machine as with the previous two options. Several papers in the literature that have researched the behavioural analysis of ransomware have selected the virtual environment (Hou et al. 2024, Ozturk et al. 2024, Hussain et al. 2023) as an experimental environment. Furthermore, among the options on the market, a fairly recurring one concerns the use of the VirtualBox software for the creation and management of virtual environments (Hussain et al. 2023).

### 2.4.3 Target Victim Dataset

There is not much documentation in the methodologies of papers regarding target victim datasets. The importance of the target dataset is fundamental in research involving behavioural analysis. In fact, without a target dataset that populates the victim machine, it would not be possible to replicate a real environment. Defining the characteristics and methods used to create the target dataset is a central step in a paper so that other researchers can replicate the work. For this reason Hussain et al. (2023) addressed the problem and declared the used of the NapierOne dataset in their methodology. In order to create a realistic environment in which ransomware can operate, the NapierOne dataset (Davies et al. 2022) is used. This dataset was designated for these reasons and also allows easy access to real-world files with different extensions. Other research such as Berrueta et al. (2020) addresses the problem of the absence of public datasets by creating methods and datasets.

### 2.4.4 Exfiltration Ransomware samples

This section compares the malicious datasets used by the papers in the literature and the methods that researchers have used to collect them. In the context of exfiltration ransomware datasets, one problem concerns the lack of comprehensive malicious datasets (McIntosh et al. 2023). The exfiltration ransomware families most used as samples during searches are BlackBasta, BlackCat, Conti, Darkside and Lockbit. The most used database from which the samples were downloaded is VirusTotal. Ozturk et al. (2024), Vasconcelos & Almeida (2023), Liu & Chen (2023b) and Ozturk et al. (2024) all used a number of families no greater than 10, downloading samples from the VirusTotal database. Hussain et al. (2023) used 933 different ransomware samples, belonging to 14 families downloaded from the MalwareBazaar database. None of the most recent ransomware families such as Akira, BlackCat and LockBit, were found in the dataset. Mehrban & Geransayeh (2024) used a greater number of families, equal to 54 ransomware families for a total of 396 samples. Their dataset did not have just exfiltration ransomware. In this work the samples were downloaded from Ransomware Tracker and the families found do not correspond to the latest ransomware families. Almeida &



Vasconcelos (2023) used 271 samples, belonging to the top 10 ransomware groups of 2023, downloaded from various databases such as VirusTotal, VirusShare and other public datasets. The paper that used the largest dataset is Hou et al. (2024). In fact it is composed of 7796 active ransomware samples from the last 3 years, belonging to 95 different families which target Microsoft Windows. Furthermore, unlike the other papers, the samples were collected from 7 databases: VirusTotal, VX Vault, InTheWild, Malware Bazaar, tutorialjinni, vx-underground, and theZoo. Hou et al. (2024) further differs from the other papers in that in addition to having said where the samples were taken from and in what quantity, it described the process used for the samples section. According to the researchers, each sample has been tested on the testbed. Each sample had to perform encryption, locking, leave notes or change the desktop background actions to be able to be selected for the experiment. Furthermore, each sample was analyzed using AVClass2 (Sebastián & Caballero 2020), to assign tags and categories, and had to have at least two security vendors listed by VirusTotal to be included in the dataset. Unfortunately, By performing multiple tests on the dataset developed in the Hou et al. (2024) research, this last point was not found, as many of the samples are not present on VirusTotal.

## 2.5 Conclusion

This chapter shows the results of the literature review, providing the author with an understanding of the topic of ransomware, its classification, its phases and the evolutions that over time have led ransomware to become the danger it is today. Furthermore, the topic of exfiltration was explored in depth, analyzing the techniques, protocols and methods used. Finally, the more technical aspect regarding the methods used to analyse ransomware exfiltration in other papers is presented.

The continuous increase in ransomware attacks during the last few years is most likely due to the strong profit for the attackers. With the advent of RaaS and the shift of targets from the individual user to companies or public bodies such as governments and hospitals, the impact that this type of malware has on society is increasingly greater. The introduction of data exfiltration in ransomware, in ad-

dition to encryption and locker actions, has led to the creation of double extortion (MS-ISAC 2022). It allows the attacker to request multiple ransoms and to incentivize the payment of the ransom, since if it is not paid, not only the user no longer has access to their files but they are typically sold or published online and therefore anyone with an internet connection can access them (Anand & Shanker 2023).

Considering Exfiltration-Ransomware, two main data stealing techniques are usually selected, performing a LOTL attack, through the use of legitimate tools, or carrying out exfiltration using dedicated software. The most used legitimate method for data exfiltration is *Rclone* as it is very versatile, followed by *FileZilla* and *WinSCP* (Symantec 2024, Demboski 2023, Oren Biderman 2024). While for exfiltration using dedicated software, the most famous are ExMatter and StealBit (Mayer 2022). Furthermore, it is important where the files are sent, in most cases the destination is the C&C server but there are various ransomware that instead use cloud storage services such as *MEGA*.

The analyses carried out by other researchers in the literature regarding exfiltration-ransomware mainly focus on the creation of datasets and the use of machine learning to be able to identify and consequently block the execution of ransomware. Furthermore, the experimental methods used by multiple papers in the literature were analysed. The virtual environment, through the use of virtual machines, is the most used experimental environment in the literature. It is very versatile and allows the researchers to speed up the experiments. This is because by creating snapshots it is possible to restore the system to a previous safe state (Hou et al. 2024, Ozturk et al. 2024, Hussain et al. 2023). The NapierOne dataset created by (Davies et al. 2022) is the latest up to date dataset for this type of work. It allows the researcher to populate a machine with a set of files gripped by file type. Furthermore, the file types present in the dataset are relevant for carrying out behavioural analysis of ransomware Hussain et al. (2023). There are limited datasets about ransomware exfiltration. This problem causes difficulties in carrying out research in this area. Many of the papers in the literature review create their own dataset by downloading samples from databases such as VirusTotal (Almeida &

Vasconcelos 2023, Ozturk et al. 2024, Vasconcelos & Almeida 2023). Hou et al. (2024) in addition to downloading the samples, it uses methods to classify them into their families. These methods include validating the information with Virus-Total and using tools such as AVclass2 which analyzes and assigns families and tags to the samples (Sebastián & Caballero 2020).

## Chapter 3

# Methodology and Design

### 3.1 Introduction

This chapter presents the research methodology, based on the information collected from the papers analyzed in the literature review, and defines the experimental methodology and the design of the experiments. It covers the testing environments, the data collection and analysis tools, the target dataset and the malicious exfiltration dataset.

### 3.2 Research Methodology

The research methodology, shown in Figure 3.1, is based on four central points. The starting point is the analysis of the papers in the literature to understand the current focuses, limitations, and problems and identify potential missing elements. Following the literature review, there is a part of developing theories and hypotheses, based on previous studies, which led to the implementation of the experimentation phase. Finally, the data observation and analysis phase took place. This process, although initially linear, became cyclical as it moved from one phase to another based on the discoveries made. This leads to better development of the paper, as it is possible to introduce new variables within the experiments as each phase proceeds, allowing the research to be more complete and not limited to the first literature review.

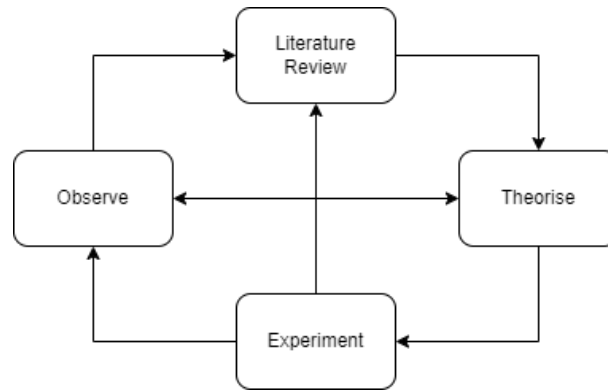


Figure 3.1: Research Methodology

### 3.3 Experimental Methodology

From the literature review it was understood that the absence of datasets regarding exfiltration-based ransomware is a central problem (McIntosh et al. 2023). Furthermore, there are gaps regarding methods for selecting samples to create a dataset to use in experimentation. In fact, among the papers analyzed only (Hou et al. 2024) explained the method they used. Another topic that does not seem to be explored in depth is the behavioural analysis of exfiltration-based ransomware. For these reasons, this dissertation aims to carry out the analysis of exfiltration samples and the behavioural analysis of ransomware exfiltration attacks. We want to build a tool/script that allows researchers to take and analyze information about samples from online databases. The development of the script has the purpose of analysing and consequently validating samples, to possibly create datasets. Furthermore, we want to analyze the behaviour of ransomware exfiltration attacks by executing samples, modules and simulations. The analysis is done by monitoring activities, processes, network traffic and file accesses.

The steps for analyzing the datasets and subsequently selecting the samples to create the malicious dataset are as follows:

1. Get the list of hashes from datasets or databases
2. Format the input file
3. Run the python script
4. Analyze the results

In the experiments, 12 between real exfiltration-based ransomware families and modules have been selected for analysis. There are two to four samples for each family adding up to a total of 35 samples. In addition, there are four simulation methods. For all samples, modules and methods used, more than a single experiment is carried out in order to validate that external variables have not led to non-normal behaviour. The tests include a dynamic and static analysis. The experimental methodology used in this project is divided into two parts. The first corresponds to the initial setup of the test environment and is the same for all the experiments carried out. It is composed of the following steps:

1. Creation of virtual machines, one with Windows and the other with REMnux respectively.
2. Having the machines connected to the internet, install on both machines the tools necessary for the operation of the subsequent phases such as analysis tools, tools for transferring information from one machine to another and tools for decompressing files.
3. Download the target dataset to populate the Windows machine.
4. Import a text file containing all the links from which to download the samples on both machines.
5. Move the REMnux machine within the internal network.
6. Take a snapshot of the ready-to-use and secure stage of both machines.

Since multiple experiments are carried out, the second part of the experimental methodology will be repeated several times and varies depending on the experiment. For real ransomware exfiltration samples, it consist of the following steps:

1. Download the ransomware sample to the victim machine.
2. Move the Windows machine within the internal network.
3. Disable Windows Security on the victim machine.
4. Take a snapshot of the Windows machine with the malicious sample on it.
5. Activate the tools on both machines to capture the behaviour of the sample.
6. Record the screen from the PC where the virtual machines are on.
7. Unzip the sample.
8. Run the sample as administrator.

9. Wait for the end of the sample execution up to a maximum of 20 minutes of execution.
10. Analyze the results.

For the static analysis of the samples the experimental methodology concerns only the REMnux machine and it is performed only for the samples which did not work properly in the dynamic analysis. The steps are the following:

1. Connect the REMnux machine to the internet to download the samples.
2. Download a batch of samples to the machine.
3. Move the REMnux machine inside the internal network.
4. Take a snapshot.
5. Unzip the sample.
6. Analyze the sample using Ghidra.

For the simulation of the exfiltration-ransomware samples the experimental methodology is composed of the following steps:

1. Connect the Windows machine to both networks (internal and external).
2. Connect the REMnux machine to the internal network only.
3. Set up the connection to the cloud storage and the FTP server on the Windows machine.
4. Take a snapshot of both machines.
5. Activate the tools on the Windows machine to capture the behaviour of the exfiltration method.
6. Run the sample as administrator.
7. Wait for the end of the sample execution up to a maximum of 20 minutes of execution.
8. Analyze the results.

### 3.4 Experimental Design

This section describes the testing environments, highlighting the differences based on which experiments must be carried out. Furthermore, the tools used to carry out the static and dynamic analysis of ransomware data exfiltration are defined.

Finally, the choices regarding the target dataset and the malicious datasets are explained.

### 3.4.1 Testing Environments

The testing environment is very important with regards to behavioural malware analysis as we want to recreate an environment that is as similar to the real one as possible but at the same time is reproducible so that others can repeat the experiments. Furthermore, it is very important to have a closed and secure environment so that malware does not spread within the network or to other devices (Ozturk et al. 2024). Since both real ransomware samples and Living off the Land simulation methods are performed in the experimentation, it is necessary to have two testing environments. They need to have the highest number of similar characteristics but with substantial differences regarding the network. Thus to be able to conduct different types of experiments and analyze the results of the two environments in the same way.

The experiments can be performed in 3 different environments which all have strengths and weaknesses. Using a real PC testbed the results would be more similar to the real world as there is no virtualization but on the other hand, it is more difficult to capture the information and return to a safe stage to test further samples. Using instead a Virtual Machine (VM) it is much easier to carry out the multiple phases of the experiments as well as restore the machine to a safe state. On the other hand, the environment is less realistic and in some cases, the ransomware samples could seek to detect that they are inside a VM and consequently not execute their malicious behaviours. The final environment in which ransomware behaviour can be analyzed is via online sandboxes such as *AnyRun* and *Cukoo* (Mehra & Pandey 2015, Kok et al. 2022, Agrawal et al. 2022). They allow the researcher to run ransomware in an online sandbox making the experiments very safe as these types of environments are created specifically for malware analysis. On the other hand, they do not allow complete interaction with the testing environment compared to PCs and VMs. Furthermore, by using online sandboxes the environment for running ransomware samples and simulations



should be different, not allowing a homogeneous data collection. In this work, it was decided to use virtual machines as the testing environments to facilitate the experimental method (Hou et al. 2024, Ozturk et al. 2024, Hussain et al. 2023). To create the virtual environment the program *Oracle VM VirtualBox* has been selected as in Hussain et al. (2023) paper.

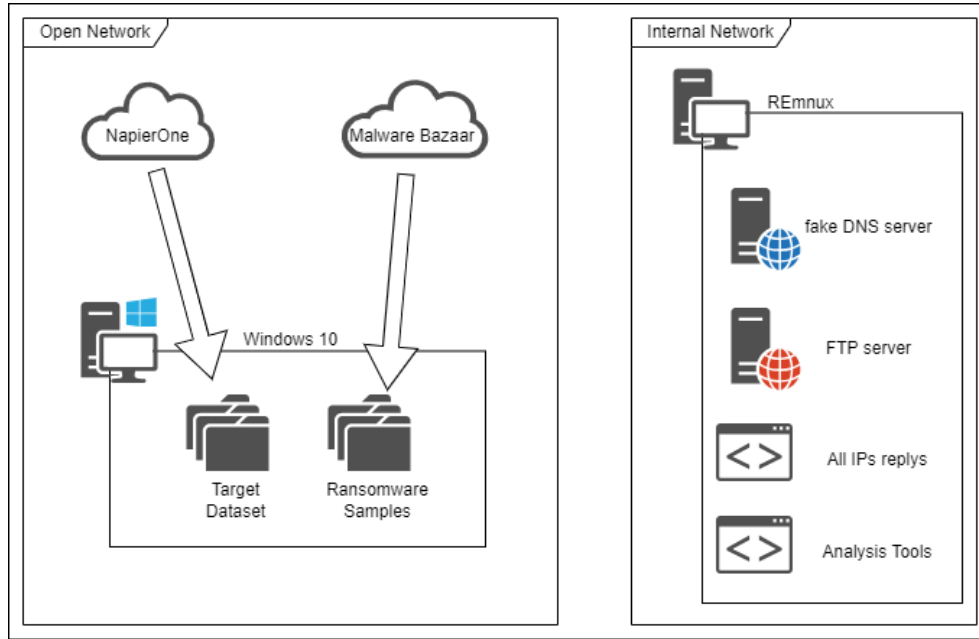


Figure 3.2: Testing Environment After the Initial Setup

As shown in Figure 3.2, the victim machine has Microsoft Windows 10 as its operating system (OS). This was decided not only because the majority of research currently present in the literature analyzes the ransomers specific to this OS but also because it is the operating system most targeted by the ransomware (Hussain et al. 2023). From now on the Windows machine will also be named as the victim machine or the victim device. For the second virtual machine, which is used to intercept traffic and act as a C&C server, the Ubuntu-based Linux distro called *REmnux* has been selected. The reason behind this choice is due to the presence of dedicated malware analysis tools already installed in the operating system such as *Wireshark*, *fakedns*, *INetSim* and *accept-all-ips* (REmnux 2022). By doing so, all the needed tools are in the environment, so it is possible to intercept and respond to network traffic generated by ransomware or other LOTL methods, after the proper network configuration. The related tools allow the user to respond to

DNS queries with the specified IP address, accept connections to all IPs and emulate common network services so that malware can interact with the machine. Furthermore, in order to be able to decompress the ransomware samples, some tools such as *WinRAR* and *7-zip* were installed on the *Windows* machine. Table 3.1 shows some hardware and software specifications of the experiments.

Virtual Environment	Oracle VM VirtualBox 7.0	
Victim Machine	Windows 10	
	CPU	6 core
	RAM	8 GB
	Storage	60 GB
	Internal Network IP	172.31.100.1
Server Machine	REMnux	
	CPU	1 core
	RAM	4 GB
	Storage	60 GB
	Internal Network IP	172.31.100.2
Hosting Machine	Windows 11	
	CPU	i7-10750H
	RAM	16 GB
	Storage	500GB SSD

Table 3.1: Testing Environment Specification

### Testing Environments for Real Exfiltration Ransomware

During the execution of the real exfiltration ransomware, shown in Figure 3.3, the Windows machine is located inside the internal network to be able to communicate with the *REMnux* machine but not with external devices.

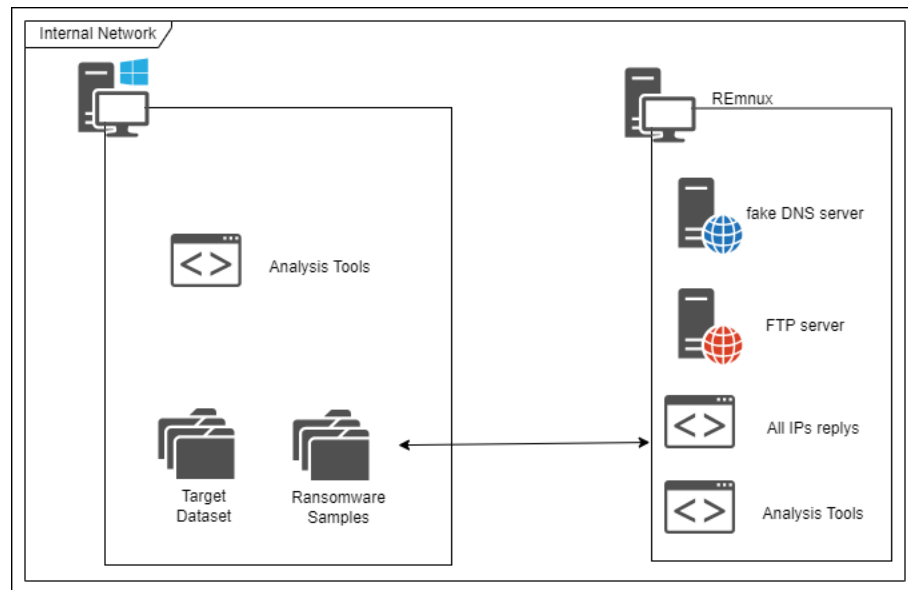


Figure 3.3: Testing Environment for Real Exfiltration Ransomware

### Testing Environments for Simulation

The testing environment for simulating ransomware exfiltration using LOTL methods is shown in Figure 3.4. The Windows machine is connected to both the internal and external network so it can communicate with both the cloud storage service and the FTP server on the REMnux machine. As analyzed in the literature, multiple ransomware makes use of the *MEGA* cloud service, for this reason, it was also chosen in this research.

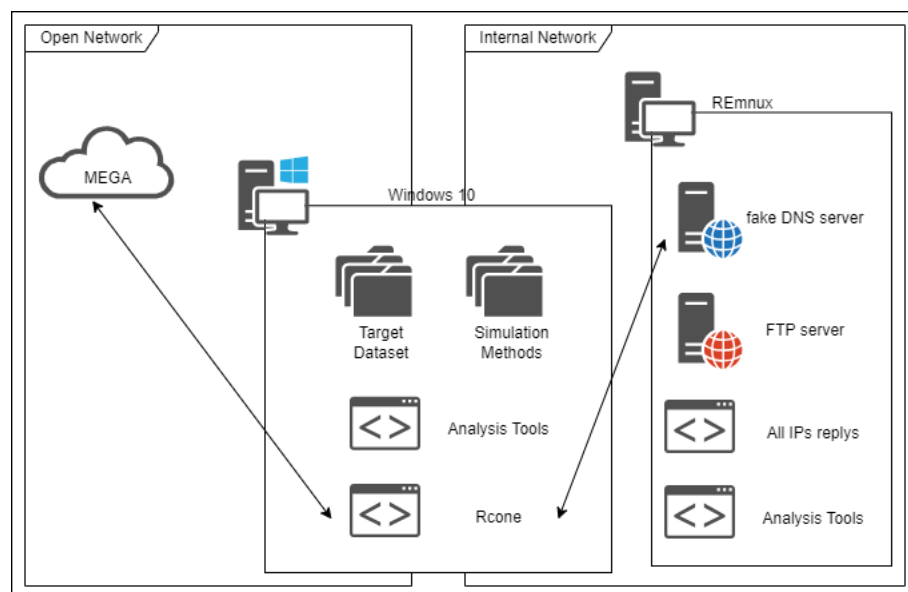


Figure 3.4: Testing Environments for Simulation

### 3.4.2 Data Collection and Analysis Tools

The data collected can be classified into two categories based on the type of analysis carried out, static or dynamic. The static analysis is carried out using *Ghidra*, a software reverse engineering (SRE) tool developed by the NSA's Research Directorate in support of the Cybersecurity mission (NSA 2019). It allows the user to decompile the executable file and read the *Assembly* and *C* language codes. *Ghidra* runs on the *REMnux* machine because since the exfiltration ransomware to be analyzed was written to run on *Windows*, it is safer to analyze it on a different operating system. This analysis is carried out to better understand the behaviour of the ransomware samples which do not seem to execute properly. Thus identify and understand errors raised during the execution of the exfiltration-based ransomware. The dynamic analysis of ransomware and simulated methods is carried out using 3 tools: *Wireshark*, *Procmon* and *Procexp*. *Wireshark* is used to capture and analyze network traffic generated by the victim machine Lamping & Warnicke (2004). This tool runs on the *REMnux* machine and its output is a *.pcap* file that would be saved on the Desktop. *Procmon* is a widely used tool for monitoring and analyzing file system, registry, process and thread activity in the context of malware behavioural analysis (Russinovich 2023b). This tool allows the user to monitor in real-time the activities carried out by a program, in the case of our expresses, ransomware. Furthermore, it allows the researcher to analyse the activities both in terms of network traffic and individual accesses to files, dlls etc. *Procexp* is another fundamental tool for the dynamic analysis of ransomware as it allows the user to monitor in real-time the processes, sub-processes, resources used and other relevant information. This is done to verify the correct functioning of the ransomware and identify the creation of sub-processes and their termination (Russinovich 2023a). Both *Procmon* and *Procexp* run on the *Windows* machine. All the tools mentioned in this section are used to collect information regarding the ransomware code, network traffic and process activity. The data found are subsequently analyzed and compared with each other.

### 3.4.3 Target Victim Dataset

To populate the victim machine it was decided to use the *NapierOne* dataset (Davies et al. 2022) as done by the paper such as Hussain et al. (2023). *NapierOne* is a dataset designed to perform ransomware detection and easily allows the user to use real-world files belonging to different file types. This makes it easier to replicate the testing phase of this work and also allows researchers to have common datasets and methods to carry out their tests, creating a standard for research in this area. For these reasons, this dataset was selected as the target victim dataset in this work. For the experiment, a collection of 1.70GB was used. It is composed of 958 selected files grouped by 21 different extensions. The files have different sizes that can range from 10KB up to 150MB. The selection of file extensions was not random as only the extensions that are usually included or excluded during the exfiltration phase were selected. Table 3.2 shows which extensions have been selected, in what quantity and the minimum file size. Some folders were created in the path C:\Users\Alice\ and the files were distributed among those folders, as is done by Hussain et al. (2023).

Extension	Quantity	Minimum Size (KB)
bmp, docx, dwg, exe, jpg, mp3, mp4, png, ppt, pptx, rar, svg, xls, zip	50	350
pdf	45	350
xlsx	27	350
doc	50	70
html, json	50	50
txt	8	50
xml	28	10

Table 3.2: Target Dataset Information

### 3.4.4 Malicious Dataset

#### Ransomware Exfiltration Dataset Analysis

The selection of the ransomware dataset is very important as a fundamental part of the experiment is based on it. Both the quantity and the characteristics of the

samples are important to be able to collect relevant data on which to carry out an analysis (McIntosh et al. 2023). The fundamental consideration for choosing the samples was that they could be run on a machine with *Windows* and the presence of file exfiltration. Unlike other papers, the selected samples were chosen regardless of their extension as it was considered penalizing to place this limitation due to the restriction of possible ransomware.

The selection of samples began from the datasets used by other papers in order to be able to delve deeper into the research and be able to make comparisons between the results achieved by the current research and others. The datasets used in the other papers are mostly made up of only hashes, and no further data is given to classify and give information regarding the samples Hou et al. (2024). It was decided to create a *Python* script, to have a better understanding of the ransomware selected by the other studies and to understand which exfiltration-based ransomware to consider in this work. The script will be able to gather information from online databases about the samples by just having their hash.

The developed script takes as input a text file with a list of sha256 hashes inside and searches for information on two main malware databases *VirusTotal* and *Malware Bazaar*. The script collects data shown in Table 3.3 from the individual databases and combines them together into a single .csv file. Furthermore, the *AVclass2* tool was also used to collect further information such as malware families and different tags and so to have a better analysis of the samples (Sebastián & Caballero 2020, Hou et al. 2024).

Gathered Fields		
SHA-256	File Type	AVclass Family
File Names	First Submission Time	AVclass Tags
Signature	First Submission Date	Databases
Tags	Threath Names	

Table 3.3: Gathered Fields from Databases by the Python Script

After an initial analysis of the datasets used by the papers present in the literature,

it was decided to build a series of semi-automatic scripts to collect lists of hashes from some databases, to analyse with the Python script. Additional malware databases were considered, such as those used by Hou et al. (2024). After in-depth analysis, it was decided to use *VirusTotal*, *Malware Bazaar* and *Triage* to collect the data. This is because they showed the most relevant results and had the greatest amount of information, making it more difficult to have misclassified samples in the dataset. To do this, a methodology was created composed of the following steps:

1. Search the databases for samples of multiple ransomware families based on tags and signatures via the browser interface.
2. Modify the web requests in such a way as to have all the hashes in a single web page.
3. Use a javascript script to quickly collect all the hashes found on the page.
4. Clean and format the results using the web page created to present the hashes in the right format for the Python script.

Based on all the information collected, the ransomware or dedicated exfiltration methods to be executed were selected as shown in the following Table 3.4.

<b>Ransomware / Exfiltration Modules</b>		
Akira	BlackMatter	Lockbit
BlackBasta	Cl0p	Play
BlackByte	ExByte	Royal
BlackCat	ExMatter	StealBit

Table 3.4: Real Exfiltration-based Ransomware Samples and Exfiltration Modules

All samples were downloaded individually to the victim machine for dynamic analysis and to the *REMnux* machine for static analysis via *Ghidra*.

### Simulation Exfiltration Methods

To simulate ransomware exfiltration, different methods belonging to the Akira (Demboski 2023), BlackCat (Oren Biderman 2024) and RagnaLocker (Symantec 2024) ransomware were used. In all three cases, the data is exfiltrated using the Rclone tool and sent to the MEGA cloud storage. Furthermore, an exfiltration

simulation has been added by sending data to an FTP server. This further simulation was included in the tests as some articles have reported that ransomware like Akira and Maze used the FTP protocol to exfiltrate data (TrendMicro 2023a, Kennely et al. 2024). The code that has been found for the simulations is shown below. All variables are written in uppercase and are defined for the different families in Table 3.5.

```
rclone.exe copy C:\Users\Alice DESTINATION: --auto-confirm
--ignore-existing -P --max-age AGE --multi-thread-streams N
--transfers N FILTERS
```

Family	Destination	Age	Thread	Filters
Akira	MEGA	1y	25	exclude
Akira	FTP server	1y	25	exclude
BlackCat	MEGA	2y	12	filter-from
RagnarLocker	MEGA	2095d	6	exclude

Table 3.5: Rclone Exfiltration Flags from Different Ransomware Families

Although the actual script uses the `--max-age` flag, it was decided not to use this parameter since all the files belonging to the target dataset have a creation date before the Ages set by the script (Age column of the Table 3.5) and that would have resulted in the exfiltration of any of the data belonging to the target dataset.

### 3.5 Conclusion

This chapter presented the research and experimental methodology and the design of the experiment. The "circular" research methodology allows us, during the dissertation, to go back and forth from the literature to the experiments. Thus to make changes to the experiments based on the analysis and observation found. It is composed of four phases: Literature Review, Theorise, Experiment and Observe. Starting from gaps found in the literature review the experimental methodology was defined. It defines the gaps that this dissertation aims to fill. Furthermore, the experiments and the fundamental steps to be able to reproduce them correctly are defined. In the design of the experiments, fundamental parts such as the testing environments, target victim dataset and malicious dataset are described



and analyzed. The testing environment comprises two virtual machines, with Windows and REMnux as operating systems, which respectively represent the victim machine and the server to intercept the traffic. In order to collect the data to be analyzed, various tools are used including Ghidra for static analysis, Wireshark, Procmon and Procexp for behaviour analysis which includes network traffic and process activities. To recreate, on the Windows machine, an environment closer to the real machine, it was decided to use the NapierOne dataset to populate the machine. A collection of 1.70GB was used, composed of 958 files. All the files were distributed inside folders in the path `C:\Users\Alice\`. Additionally, to analyse and select the proper samples to make a malicious dataset, it has been decided to develop a script to gather information about the sample. The selection was based on datasets used by other researchers and on information collected from the VirusTotal, Malware Bazaar and Triage databases. Moreover, the information collected during the literature review was used to select the samples to simulate and it includes the execution of tools including Rclone.

## Chapter 4

# Experiments, Results and Evaluation

### 4.1 Introduction

This chapter presents the performed experiments and achieved results. It covers the ransomware exfiltration analysis, describing the scripts that have been developed to analyze the ransomware samples. Moreover, the dynamic and static analysis of the selected ransomware exfiltration samples has been described. Further, the data relating to the ransomware exfiltration simulation is shown. In each of the sections, the results obtained from the experimentation phase are also shown. Finally, the evaluation of the results is presented.

### 4.2 Exfiltration Ransomware Analysis

Given that the selection of which ransomware samples to use during the experimental phase is very important in order to have truthful and classifiable results, it was decided to start from the analysis of the datasets used in the papers presented in the literature review. In many cases, only generic information about the datasets is present such as the quantity of samples, the different families, etc. and not the actual hashes. For this reason, Hou et al. (2024) were contacted to ask for the ransomware dataset and exfiltration-based ransomware dataset used in their work. It was decided to contact the researchers of this paper not only because their dataset is the largest that can currently be found in the literature, but also

because part of their research focuses on the exfiltration phase of ransomware. Since the dataset provided to us by Hou et al. (2024) is composed solely of hashes, without further information, a Python script was implemented that could collect information regarding hashes quickly through the use of online databases including VirusTotal and Malware Bazaar.

#### 4.2.1 Script to Gather Sample Metadata

The script, called "ransomware-metadata.py", takes as input a text file which contains a list of hashes, one for each line, and generates as output a folder with multiple files inside it. The script, developed with Python 3.12.0, has a couple of requirements in addition to the internet connection. The first requirement concerns the possession of an API key, necessary to make requests on VirusTotal, and the second requires the installation of AVclass2, necessary to have a better analysis of the samples (Sebastián & Caballero 2020).

##### Script Analysis

The script, after reading the input file, begins the data-gathering phase on VirusTotal and subsequently on MalwareBazaar. The order of this sequence is critical as the VirusTotal API used has a daily limit of 500 requests. If the maximum number of requests is reached, a limit is also set on the number of requests made to Malware Bazaar in such a way as to have the same number of hashes analyzed in the results. During the data-gathering phase, immediately after making the request to the database, the response taken from the server is given as input to the AVclass2 program. Subsequently, the relevant information, shown in Table 3.3, is selected, analysed and saved. The flow chart of the script is shown in Figure 4.1

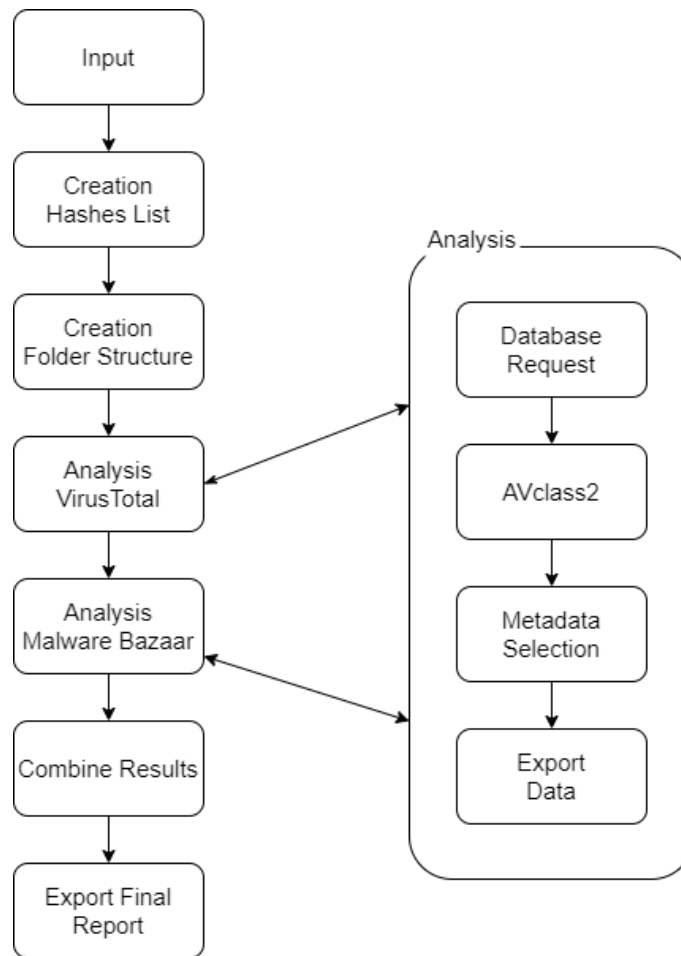


Figure 4.1: Script Flowchart

One of the most important pieces of information that is collected from the databases are the "Threat Names" which correspond to the classifications of the sample from a series of antivirus or third-party companies. After several analyses, it was observed that in some cases the "Threat Names" that classify ransomware are generic or completely different from each other. For this reason, it was decided to select only some of the "Threat Names" found in the databases. The "Threat Names" chosen have been defined by TrendMicro, Microsoft, Kaspersky, BitDefender and ReversingLabs (Listing 4.2, line 29).

Below there are two functions of the script to gather data from the databases. Listing 4.1 shows the function used to execute the python tool AVclass2, simply named AVclass in the script. The json response (Listing 4.2, lines 13-15) of the database is given as input and information regarding the family and tags of the

sample is taken.

```

1 def AVclass(result, folderName, h):
2     path = f"{folderName}\\VirusTotal\\{h}.json"
3     exportJson(result, path, False)
4     avclass = subprocess.run(["avclass", "-f", path, "-t"], stdout=subprocess.
        PIPE, stderr=subprocess.PIPE)
5     avclass_results = avclass.stdout.decode().split("\t")
6     output = {}
7     tags = []
8     if len(avclass_results) > 1:
9         for i in avclass_results[2].split(","):
10             label = i.split("|")[0]
11             if "FAM" in label:
12                 output["family"] = label[4:]
13             else:
14                 tags.append(label)
15         tags.sort()
16         output["tags"] = tags
17         if "family" not in output:
18             output["family"] = ""
19         return output
20     return None

```

Listing 4.1: Function to get AVclass data

The function shown in Listing 4.2 concerns the collection of information from the VirusTotal database. It takes as input the list of hashes created in the main and for each hash it searches VirusTotal for its information. The output of the function is an array of objects in which each object contains all the information of a single sample.

```

1 def virusTotal(hash_list, folderName):
2     api_url = "https://www.virustotal.com/api/v3/files/"
3     headers = {"accept": "application/json", "x-apikey": api_key_virustotal}
4     metadata_list = []
5     counter = 0
6     num_lines = len(hash_list)
7
8     print("Starting VirusTotal Scan...")
9
10    for h in hash_list:
11        counter += 1
12        response = requests.get(api_url + h, headers=headers)
13        result = response.json()
14
15        avclass = AVclass(result, folderName, h)
16

```

```

17     hash_metadata = {}
18     hash_metadata['index'] = counter
19     hash_metadata['sha256'] = h
20
21     if response.status_code == 200:
22         print(f"{counter}/{num_lines} - {h}")
23
24         hash_metadata['names'] = getNames(result['data']['attributes']['names'], h)
25         hash_metadata['file_type'] = result['data']['attributes']['type_description']
26         hash_metadata['fs_date'] = datetime.fromtimestamp(result['data']['attributes']['first_submission_date']).strftime("%d/%m/%Y")
27         hash_metadata['fs_time'] = datetime.fromtimestamp(result['data']['attributes']['first_submission_date']).strftime("%H:%M:%S")
28
29         hash_metadata['threat_names'] = getThreatNamesVT(result['data']['attributes']['last_analysis_results']['TrendMicro']['result'], result['data']['attributes']['last_analysis_results']['Microsoft']['result'], result['data']['attributes']['last_analysis_results']['Kaspersky']['result'], result['data']['attributes']['last_analysis_results']['BitDefender']['result'])
30         hash_metadata['error'] = "None"
31
32         if avclass != None:
33             hash_metadata['avclass_FAM'] = avclass["family"]
34             hash_metadata['avclass_TAGS'] = avclass["tags"]
35         else:
36             if response.status_code == 429:
37                 print(f"Exiting VirusTotal scan because error: {result['error']['message']}")
38                 print()
39                 return metadata_list, counter
40             else:
41                 print(f"{counter}/{num_lines} - {h} - Error: {result['error']['message']}")
42                 hash_metadata['error'] = result['error']['message']
43
44         metadata_list.append(hash_metadata)
45     return metadata_list, counter

```

Listing 4.2: Function that Gather Data from VirusTotal

## Output Folder and Files

All data gathered from the databases and information collected from the analysis are saved inside a folder which is divided as follows:

- A "JSON" folder containing 3 .json files relating to the fields involved in the total investigation into VirusTotal, Malware Bazaar and their results combined together.
- A "MalwareBazaar" folder and a "VirusTotal" folder which contain a list of .json files, one for each hash analysed, with the respective responses to the requests made to the online databases.
- Three .csv files relating to the analysis carried out by the script, respectively one for VirusTotal, one for Malware Bazaar and one with the combined results of the two.

In the main function, shown in Listing 4.3, it is possible to see how the files and folders are created and the order of execution of the main parts of the script.

```

1  def main():
2      ...
3      Path(folderName).mkdir(parents=True, exist_ok=True)
4      Path(f"{folderName}\\VirusTotal").mkdir(parents=True, exist_ok=True)
5      Path(f"{folderName}\\MalwareBazaar").mkdir(parents=True, exist_ok=True)
6      Path(f"{folderName}\\JSON").mkdir(parents=True, exist_ok=True)
7
8      vt_mdata, counterVT = virusTotal(hash_list...)
9      exportJson(vt_mdata, f"{folderName}\\JSON\\VirusTotal.json", True)
10     of_nameVT = f"{folderName}\\ransomware-metadata-virusTotal-{{nameTime}}.csv"
11     exportMetadata(vt_mdata, of_nameVT, False)
12
13     mb_mdata = malwareBazaar(hash_list...)
14     exportJson(mb_mdata, f"{folderName}\\JSON\\malwareBazaar.json", True)
15     of_nameMB = f"{folderName}\\ransomware-metadata-malwareBazaar-{{nameTime}}.csv"
16     exportMetadata(mb_mdata, of_nameMB, False)
17
18     combined_mdata = combinedDatasets(vt_mdata, mb_mdata, folderName)
19     exportJson(combined_mdata, f"{folderName}\\JSON\\combined.json", True)
20     of_name = f"{folderName}\\ransomware-metadata-{{nameTime}}.csv"
21     exportMetadata(combined_mdata, of_name, True)

```

Listing 4.3: Main Function

Figure 4.2 shows an example of the contents of the output folder. All "ransomware-

metadata\*.csv” files contain within their name the time and date of execution of the script (instead of ”XXX”).

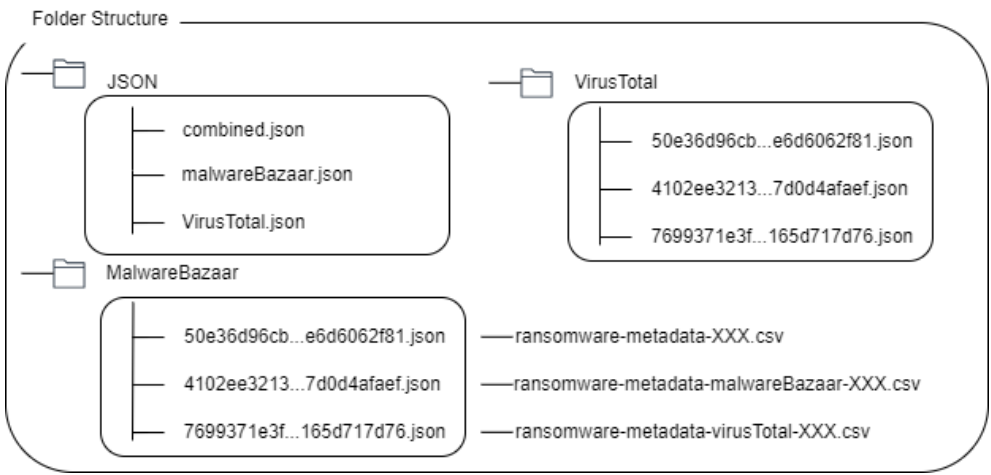


Figure 4.2: Output Folder Structure

An example of a ”ransomware-metadata\*.csv” file that contains the information collected from both VirusTotal and Malware Bazaar is shown in Figure 4.3.

	A	B	C	D	E	F	G
1	Inc	SHA-256	File Name	Signature	Tags	File Type	First Submission
			5110c4c4b4836d926be20f95c973ab29.exe enc.exe 2023-05-08_5110c4c4b4836d926be20f95c973ab29_dark				
2	1	063fcedd3089e3cea8a7e07665ae033ba765t	side	BlackMatter	BlackMatter	Win32 EXE	2
			75256873a03f4a4bc073185f48c1097c.exe 228.vir LB3.exe ConfirmEmail.exe		45,139,105,143 BlackMatter exe		
3	2	068ca3e92c65eb907b5a34be16580e267efb13cPr3UyW.exe.part.exe		BlackMatter	opendir	Win32 EXE	1
n Tim		H	I	J	K	L	
	First Submission Dat	Threat Names	AVclass Fam	AVclass Tags	Databases		
		Ransom.Win32.LOCKBIT.SMYXCIN Ransom.Win32.BlackmatterIMTB HEUR:Trojan-Ransom.Win32.Generic Trojan.Ransom.PIC Win.Ransomware.BlackMatter-9965914-0			BEH:filecrypt CLASS:ransomware FILE:os:windows UNK:blackmatter	VirusTotal MalwareBazaar	
10:16:43	25/04/2023	Win32.Ransomware.BlackMatter	cryptmodng				
		Ransom.Win32.LOCKBIT.SMYXCIN Ransom.Win32.Lockbit.AK!ibit UDS:DangerousObject.Multi.Generic Trojan.Ransom.PIC Win.Ransomware.BlackMatter-9965914-0			BEH:filecrypt CLASS:ransomware FILE:os:windows UNK:blackmatter	VirusTotal MalwareBazaar	
3:29:57	08/01/2023	Win32.Ransomware.Lockbit	cryptmodng				

Figure 4.3: Example of ”ransomware-metadata\*.csv”

The example shows the first two lines of one of the outputs of the script run on a BlackMatter dataset collected by the author. The output is the csv file generated by combining the information found in the VirusTotal and Malware Bazaar databases. As you can see, each row is indexed and information regarding the sources of the collected information is saved in column L. The file names can be very useful because if there are different extensions you can make comparisons



with column F, File Type, and therefore be able to find the correct extension to be able to run the sample. The signature and tags are very important as together with the AVclass columns (column J, K) they usually show the most important information to be able to classify the samples. As you can see in the example, columns D and J have different results for both samples. This means that there are some misclassifications. The columns just described are compared with column I, "Threath Name". This is done to evaluate the classification of the samples by the databases and define whether the samples have been misclassified. A final consideration to make concerns the Submission date and time (column G, H). When the information collected comes from more than one database, only the first date is inserted into the output file. This is done because this data cannot be wrong in databases and therefore the first date reveals when the sample was first found and loaded into a database.

### **Ransomware Metadata Analysis**

The script was run on multiple datasets and in some of the cases, as for the data sent to us by Hou et al. (2024), the analyses showed discrepancies with what was written in the papers. More precisely, it was observed that some of the exfiltration samples do not appear to belong to ransomware exfiltration and others are not actually present in the databases from which the paper claimed to have carried out some validation. For this reason, it was decided to carry out a further analysis to decide which samples to use, that are relevant to this study. A semi-automated script was created to collect hashes from Malware Bazaar and Tria.ge to be given as input to the python script.

### **Script to Gather Sample Hashes**

The Malware Bazaar and Triage databases can be searched based on tags, signatures and ransomware families. Using a browser, the ransomware exfiltration families shown in Table 3.4 were searched for on those databases. To simplify the searches, the requests to the server were modified so that all the hashes could be shown on a single page and using the browser console the script, shown in Listing 4.4 below, was executed to select the hashes from the HTML code, as shown in Figure 4.4.

```

1 function onlyUnique(value, index, array) {
2   return array.indexOf(value) === index;
3 }
4
5 hashes = []
6 if (window.location.href.includes("tria.ge")){//TRIA GE
7   x = document.getElementsByClassName("clipboard")
8   for (i=0; i<x.length; i++){
9     hashes.push(x[i].childNodes[1].dataset.clipboard)
10  }
11 }else{//MALWAREBAZAAR
12   x = document.getElementsByClassName("shortify")
13   for (i=0; i<x.length; i++){
14     hashes.push(x[i].innerHTML)
15   }
16 }
17 var uniqueHashes = hashes.filter(onlyUnique);
18 console.log(uniqueHashes)

```

Listing 4.4: Script to Gather Data from Databases

The screenshot shows the MalwareBazaar website interface. On the left, there is a search bar and a table of malware entries. The table has columns for Date (UTC), SHA256 hash, Type, Signature, Tags, Reporter, and DL. The right side of the image shows a browser console with the output of the script from Listing 4.4, which is a JSON array of unique hashes.

Date (UTC)	SHA256 hash	Type	Signature	Tags	Reporter	DL
2022-03-03 01:53	6b795d9faa48ce3ae31f0...	exe	Steadbit	exe, ransom, stealbit	r3dbU7z	
2022-03-03 00:27	ce6b07c00894fd72d7733...	exe	Steadbit	exe, stealbit	r3dbU7z	
2021-12-24 11:08	968875370dbc26a64398...	exe	Steadbit	exe, stealbit	Arkbird_SOLG	
2021-12-24 11:08	61ac7ac9087914562f58...	exe	Steadbit	exe, stealbit	Arkbird_SOLG	
2021-12-24 11:08	8b5f88aeaa4d50c90e0c...	exe	Steadbit	exe, stealbit	Arkbird_SOLG	
2021-08-19 03:50	3407f26b3d69f1dfce767...	exe	LockBit	exe, lockbit, ransomware	JAMESWT_MHT	
2021-08-19 03:50	4db7eed852946803c16...	exe	LockBit	exe, lockbit, ransomware	JAMESWT_MHT	
2021-08-19 03:49	bd14872dd9fdeadd89fc07...	exe	LockBit	exe, lockbit, ransomware	JAMESWT_MHT	
2021-08-18 15:34	07a3dcdb8d9b062fb4806...	exe	LockBit	exe, lockbit, ransomware	JAMESWT_MHT	

Showing 1 to 9 of 9 entries

Previous 1 Next

© abuse.ch 2024

```

>> function onlyUnique(value, index, array) {
  return array.indexOf(value) === index;
}

hashes = []

if (window.location.href.includes("tria.ge")){//TRIA GE
  x = document.getElementsByClassName("clipboard")
  for (i=0; i<x.length; i++){
    hashes.push(x[i].childNodes[1].dataset.clipboard)
  }
}else{//MALWAREBAZAAR
  x = document.getElementsByClassName("shortify")
  for (i=0; i<x.length; i++){
    hashes.push(x[i].innerHTML)
  }
}

var uniqueHashes = hashes.filter(onlyUnique);
console.log(uniqueHashes)

```

Figure 4.4: Gather Hashes from Malware Bazaar

As you can see in Figure 4.4 the list of hashes found is shown in the browser console. Finally, the output text is copied and inserted into the created web page shown in Figure 4.5. This is done, to clean the output and format it for the python script. The web page shown in Figure 4.5 was created so that no data was sent to third-party websites.

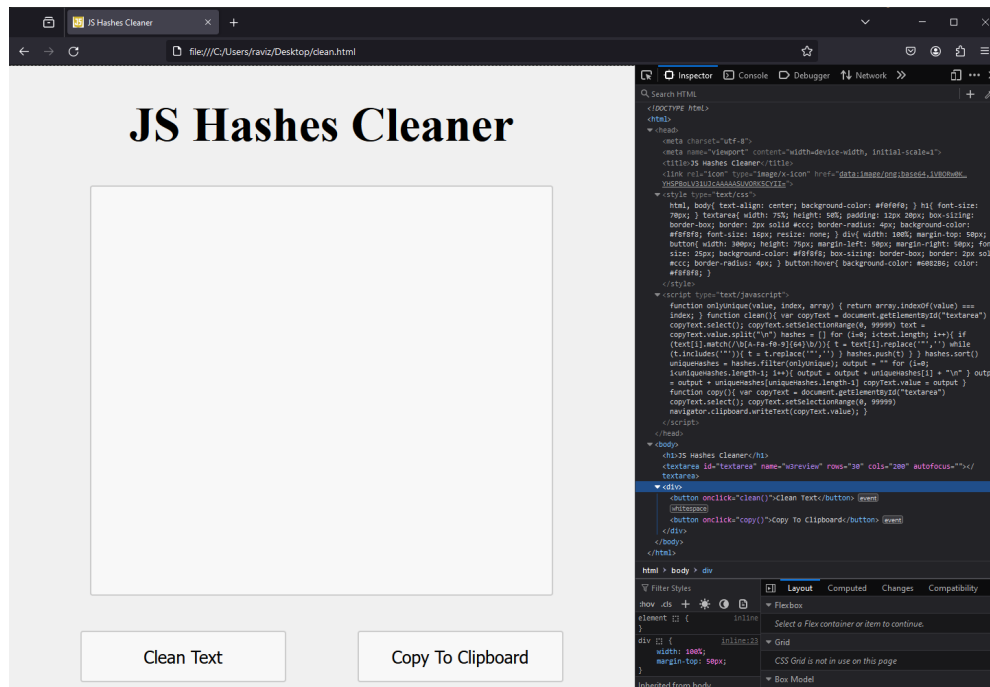


Figure 4.5: Webpage to Clean and Format the Data for the Python Script

## 4.2.2 Results

### Ransomware Exfiltration Dataset Selection

After carrying out multiple dataset analyses explained in the previous section, the ransomware exfiltration samples were selected. To select the samples, more than 1800 samples belonging to more than 20 exfiltration-based families were investigated with the python script. The samples chosen may belong to a ransomware family that performs exfiltration or to exfiltration tools and methods used by ransomware. The 35 samples, grouped into 12 families, are shown in Table 4.1 with some of the information gathered through the implemented scripts. For reasons of space and formatting, the entire hash of the samples is never inserted in the tables in the main chapters. For this reason, the appendix contains the Table 7.1 in which the family and hash of each sample are shown.

Family	Hash	File Type	Submission Date	AVclass Family
Akira	4102ee3213...faef	EXE	01/04/2024	neshta
Akira	50e36d96cb...2f81	Python	28/08/2023	agentb
BlackBasta	5d2204f3a2...b0aa	EXE	24/04/2022	delshad
BlackBasta	ae7c868713...1b6e	EXE	28/04/2022	delshad
BlackBasta	2558d08175...008a	EXE	04/02/2023	redcap
BlackByte	1df11bc19a...74ad	EXE	30/09/2021	
BlackByte	884e96a75d...7534	JavaScript	20/08/2021	sagent
BlackByte	c22a6401a4...98da	Powershell	10/11/2021	
BlackCat	3c300726a6...82d5	EXE	10/03/2019	
BlackCat	7d8671c91a...0ac9	Powershell	09/09/2021	
BlackMatter	a82aec54ca...8478	EXE	23/09/2022	darkside
BlackMatter	c6e2ef30a8...ce99	EXE	02/08/2021	darkside
BlackMatter	99d3003cc5...3d4b	EXE	26/12/2022	zeroaccess
BlackMatter	f3474589ca...a4c1	EXE	27/12/2022	cryptmodng
Cl0p	220c50ccf6...bb0c	EXE	30/01/2021	cl0p
Cl0p	3320f11728...1207	EXE	11/02/2019	cl0p
Cl0p	15f9ed36d9...2649	EXE	04/03/2021	cl0p
Conti	1ce8a939b3...94e9	EXE	28/04/2021	conti
Conti	4bfd58d4e4...2618	EXE	06/06/2021	conti
Conti	837c0f1e97...2d5c	EXE	07/08/2023	conti
ExByte	0097b8722c...3142	EXE	23/10/2022	mansabo
ExByte	3fb160e177...6b70	EXE	23/10/2022	mansabo
ExMatter	886cb22ffe...aadf	EXE	03/02/2022	
ExMatter	4a0e10e1e9...7d30	EXE	07/10/2021	redcap
ExMatter	b6bc126526...7bd7	EXE	17/10/2021	
ExMatter	8eded48c16...17ef	EXE	07/10/2021	cerbu
Hive	0077e7d6e9...2dcb	EXE	01/07/2022	hive
Hive	19c54b520e...5718	EXE	11/02/2022	hive
Hive	16baebd1ad...4b8c	EXE	23/01/2022	sabsik
Ryuk	e6762cb7d0...c32c	EXE	09/07/2019	ryuk
Ryuk	a1ce524372...7fc2	EXE	21/08/2019	ryuk
Stealbit	6b795d9faa...786a	EXE	03/03/2022	stealbit
Stealbit	61ac7ac908...aaee	EXE	28/09/2021	
Stealbit	07a3dcb8d9...a9ae	EXE	17/08/2021	lockbit
Stealbit	968875370d...7bec	EXE	08/11/2021	

Table 4.1: Ransomware Exfiltration Dataset

## Datasets Analysis

Using the javascript script, almost 1650 hashes were collected from the Malware Bazaar and Triage databases. Subsequently, this data was given as input to the python script and it generated various outputs as expected. Furthermore, other samples were analyzed by finding their hashes on online technical journals such as (Aleksandar & Kotaro 2021). The total number of samples analyzed is more than

1800. Analyzing the output of the python script, it was observed that more than 18.50% of the samples are classified incorrectly, as shown in Table 4.2.

Family	Searched Samples	Found Samples	Misclassified Samples
akira	3	3	0
blackbasta	62	62	1
blackbyte	6	6	2
blackcat	161	161	2
blackmatter	179	179	3
clop	41	39	21
conti	240	236	35
darkside	117	116	37
dewmode	1	1	0
exbyte	2	2	0
exmatter	8	8	2
hive	260	260	61
play	24	24	10
revil	5	4	3
royal	48	32	11
ryuk	2	2	0
stealbit	462	459	134
maraudermap	206	206	9
maraudermap-exfil	28	3	3
Total	1855	1803	334
Percentage		97.20%	18.52%

Table 4.2: Analysis of multiple datasets

### 4.3 Real Ransomware Exfiltration Experiment

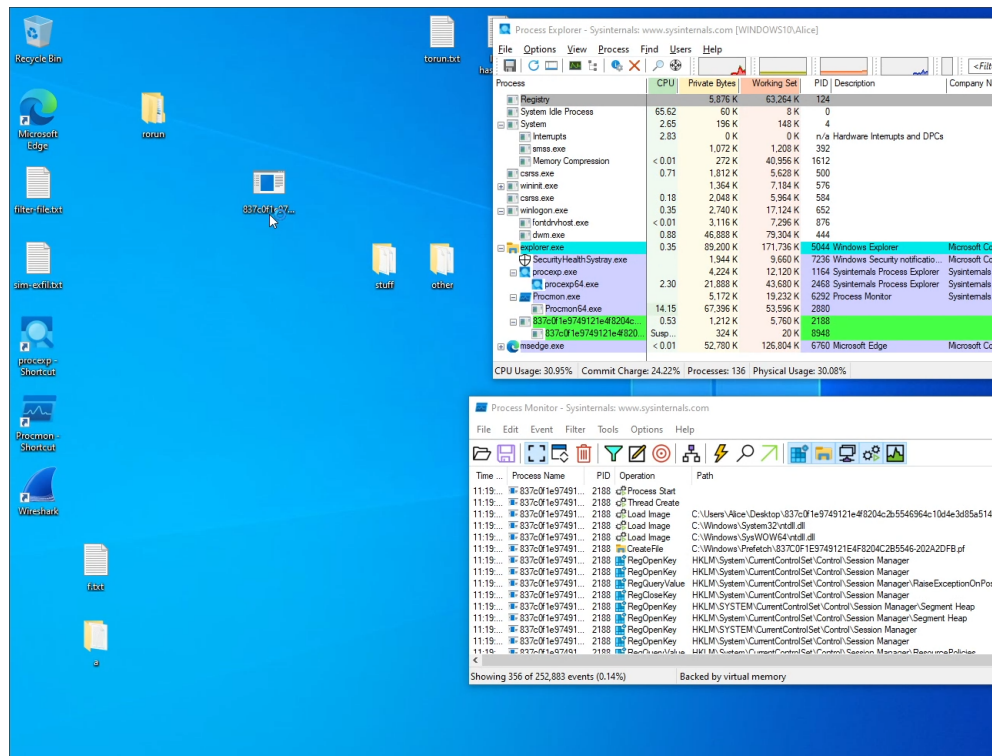
The first step, before executing the ransomware samples, is the environment setup. The programs to analyze the network and the processes are installed, the target dataset has been downloaded and the network topology is implemented as described in the previous chapter. Given that the experiment has to be repeated for the 35 samples, without the execution of one compromising the behaviour of another, after the initial setup a snapshot of both machines was taken on VirtualBox. This passage is done to always have a safe and equal starting state for all samples. After that, the experiments start by downloading one sample at a time, first onto the victim machine to carry out the dynamic analysis and subsequently also onto the Linux machine for the static analysis.

### 4.3.1 Dynamic Analysis

Once the sample has been downloaded onto the Windows machine, before extracting it from the .rar file, the target machine is moved into the internal network. Thus the REMnux machine is the only machine reachable from the victim machine and therefore it is isolated from external devices in order to avoid possible lateral movements by ransomware. On the Windows machine procexp and procmon are activated and on the latter, the filters with the file name to be executed are enabled in such a way as to intercept only its behaviour. Windows Defender and the Firewall are completely deactivated so as not to block any of the sample's activities. Accept-all-ips, fakedns, inetsim and an FTP server are activated on the REMnux machine to allow it to simulate a real network connection. Moreover, Wireshark is activated to capture and analyze network traffic. Furthermore, before running the sample, screen recording is activated on the computer where the virtual machines are hosted via OBS, a screen recording program. After carrying out all these steps, the sample, located on the Desktop, is run as Administrator. After the ransomware is executed, various analyses are carried out including:

- Network traffic on Wireshark, Procmon and fakedns
- Files and Folders on the Windows machine
- Process activity on Procmon

Figure 4.6 shows a screenshot of the first moments of the execution of the Conti sample 837c0f1e9749121e4f8204c2b5546964c10d4e3d85a514458c35cbc021762d5c on the Windows machine. The tools to monitor and analyze activities, Procexp and Procmon have been executed and they are recording the behaviour of the sample.



## Results

The results of the dynamic analysis are summaries in Table 4.3.1. As can be seen, only 18 samples carried out network activities and 10 encrypted files on the target machine. In some of the cases, the executed ransomware shut down the machine, interrupted the running analysis programs or even broke the virtual environment. In 6 cases no action appears to have been performed as the sample activity ended immediately. For these samples, a further static analysis was carried out on the REMnux machine to understand the reasons for their behaviour. It happened several times that the sample run on the first attempt reported system errors due to which it could not work. Several attempts have been made to try to resolve the errors but it has not always been possible. Only a sample of Conti scanned the

internal network via ICMP to find other devices presumably to carry out lateral movements.

Family	Hash	Execution Time	ENC	Network Activity	Connection	Comments
Akira	4102...faef	3min 48sec	Y	N		
Akira	50e3...2f81	1min	N	Y	akira[.]red	Generated an error
BlackBasta	5d22...b0aa	6 sec				Windows shutdown
BlackBasta	ae7c...1b6e	15min 52sec	Y	N		Broke Running Programs
BlackBasta	2558...008a	17min 45sec	Y	N		
BlackByte	1df1...74ad	1 sec	N	N		No activity
BlackByte	884e...7534	4sec				Broke Virtual environment
BlackByte	c22a...98da	2min 25sec	Y	Y	fluentzip[.]org	BlackMatter
BlackCat	3c30...82d5	2sec	N	N		
BlackCat	7d86...0ac9	16sec	N	N		
BlackMatter	a82a...8478	33sec	Y	Y	baroqueetes[.]com rumhsia[.]com	Darkside and Access only user folders
BlackMatter	c6e2...ce99	22sec	Y	Y	paymenthack[.]com mojobiden[.]com	
BlackMatter	99d3...3d4b	1min 1sec	Y	N		Do not seem BlackMatter
BlackMatter	f347...a4c1	34sec	Y	Y	paymenthack[.]com mojobiden[.]com	Activity in Procmon folder
Cl0p	220c...bb0c	1sec	N	N		No activity
Cl0p	3320...1207	terminated	Y	Y	104.86.119[.]114 10.0.2[.]15 104.88.110[.]98 20.189.173[.]11	
Cl0p	15f9...2649	15sec				Error
Conti	1ce8...94e9	16sec	N	N		No activity
Conti	4bfd...2618	1min 26sec	Y	Y		Network Scanning for other hosts
Conti	837c...2d5c	17sec	N	N		No activity
ExByte	0097...3142	11sec	N	Y		Ping activity
ExByte	3fb1...6b70	2sec	N	N		No activity
ExMatter	886c...aadf	terminated	N	Y	164.92.232 [.]192	Access files
ExMatter	4a0e...7d30	terminated	N	Y	157.230.28[.]192	Access files
ExMatter	b6bc...7bd7	3min 2sec	N	Y	159.89.128[.]13	
ExMatter	8ede...17ef	18min 23sec	N	Y	157.230.28[.]192	Access files
Hive	0077...2dcb	5sec	N	Y	185.112.83[.]111	Looking at browser folders



Family	Hash	Execution Time	ENC	Network Activity	Connection	Comments
Hive	19c5...5718	1sec	N	N		Looking at browser folders
Hive	16ba...4b8c	1sec	N	N		No activity
Ryuk	e676...c32c	25sec	N	N		Error .dll not found
Ryuk	a1ce...7fc2	25sec	N	N		Error .dll not found
Stealbit	6b79...786a	3sec	N	Y	5.149.249[.]242	
Stealbit	61ac...aaee	1sec	N	Y	185.182.193[.]120	
Stealbit	07a3...a9ae	1sec	N	Y	93.190.143[.]101	
Stealbit	9688...7bec	1sec	N	Y	185.182.193[.]120	

Table 4.3: Results of Dynamic Analysis

A further interesting observation concerns the ransom notes which were not created by all samples that carried out activities. The ransomware that carried out the encryption phase created the ransom note in all the folders where they encrypted the files, and other samples, such as Akira, did not carry out any encryption of the data but still created a ransom note. In several cases, references to the exfiltrated data were found within the ransom notes even if no network traffic had been monitored. This may be motivated to incentivize payment for the ransomware or the exfiltration methods of the samples have not been activated.

By carefully analyzing the fields of the datasets captured by the python script, it can frequently be observed that the Signatures, Tags, Threat Names, AVclass Family and AVclass Tags have different results for the same hash. This obviously leads to difficulties in being able to use a malware dataset correctly as one cannot be sure of the correct classification of individual samples until they are executed and their behaviour is analysed. Table 4.4 shows the data relating to this analysis regarding the malicious dataset used in the experimentation phase.

Real Family	Expected Family	Hash	Signature	Tag	Threat Name	AVclass Family	AVclass Tag
Akira	Akira	4102...faef			akira	neshta	akira
	Akira	50e3...2f81		akira		agentb	
BlackBasta	BlackBasta	5d22...b0aa	BlackBasta	BlackBasta	BlackBasta DelShad	delshad	blackbasta
BlackBasta	BlackBasta	ae7c...1b6e	BlackBasta	BlackBasta	BlackBasta	delshad	blackbasta
	BlackBasta	2558...008a	BlackBasta	BlackBasta	BlackBasta DelShad	redcap	blackbasta
	BlackByte	1df1...74ad		Blackbyte	Blackbyte		blackbyte wingo
	BlackByte	884e...7534	Blackbyte	Blackbyte	Blackbyte	sagent	blackbyte
<b>BlackMatter</b>	BlackByte	c22a...98da		Blackbyte	BlackMatter		blackmatter pidief
	BlackCat	3c30...82d5	BlackCat	BlackCat	BlackCat		BlackCat
	BlackCat	7d86...0ac9	BlackCat	BlackCat			
<b>DarkSide</b>	BlackMatter	a82a...8478	BlackMatter	BlackMatter	DarkSide	darkside	
BlackMatter	BlackMatter	c6e2...ce99	BlackMatter	BlackMatter	BlackMatter	darkside	blackmatter
	BlackMatter	99d3...3d4b	BlackMatter	BlackMatter Lockbit	BlackMatter Lockbit	zeroaccess	blackmatter
BlackMatter	BlackMatter	f347...a4c1	BlackMatter	BlackMatter	BlackMatter	cryptmodng	blackmatter
	Cl0p	220c...bb0c	Clop	Clop	Clop	clop	hydracrypt
Cl0p	Cl0p	3320...1207	Clop	Clop	Clop	clop	hydracrypt nekark
	Cl0p	15f9...2649	Clop	Clop	Clop	clop	cl0p hydracrypt
	Conti	1ce8...94e9	Conti	Conti	Conti	conti	
Conti	Conti	4bfd...2618	Conti	Conti	Conti	conti	diavolo
	Conti	837c...2d5c	Conti	Conti	Conti	conti	
	ExByte	0097...3142		Exbyte	BlackByte	mansabo	blackbyte
	ExByte	3fb1...6b70		Exbyte	BlackByte	mansabo	blackbyte
ExMatter	ExMatter	886c...aadf	ExMatter	ExMatter	ExMatter		exmatter
ExMatter	ExMatter	4a0e...7d30		ExMatter	ExMatter RedCap	redcap	
ExMatter	ExMatter	b6bc...7bd7		ExMatter	ExMatter Cerbu		exmatter
ExMatter	ExMatter	8ede...17ef		ExMatter	ExMatter Cerbu	cerbu	exmatter
Hive	Hive	0077...2dcb	Hive	Hive	Hive	hive	atraps
	Hive	19c5...5718	Hive	Hive	Hive	hive	bropass
	Hive	16ba...4b8c	Hive	Hive	Hive	sabsik	bropass
	Ryuk	e676...c32c	Ryuk	Ryuk	Ryuk	ryuk	ryspy
	Ryuk	a1ce...7fc2	Ryuk	Ryuk	Ryuk	ryuk	ryspy
Stealbit	Stealbit	6b79...786a	Stealbit	Stealbit		stealbit	
Stealbit	Stealbit	61ac...aaee	Stealbit	Stealbit	Corrempa		corrempa
Stealbit	Stealbit	07a3...a9ae	LockBit	lockbit stealbit	Stealbit Corrempa	lockbit	corrempa
Stealbit	Stealbit	9688...7bec	Stealbit	Stealbit	Corrempa		corrempa

Table 4.4: Samples Classification

As can be seen, of the 35 samples used only 17 seem to truly belong to their classification. The databases misclassified two samples, while for the other 16, it was not possible to declare their family with certainty as they did not show distinctive behaviours during the execution. To establish whether a sample belongs to a specific family, the following factors were taken into consideration:

- Connected ips

- Ransom note
- Encrypted file extension
- Message on desktop wallpaper, as shown in Figure 4.7

If the ransomware did not perform any of the behaviours listed above then it was not classified into any of the families. An important observation to consider is that the python script collects information regarding samples from 3 sources which have different rules on how and who can classify the samples and assign metadata. This is potentially one of the reasons why the data belonging to the different databases are different from each other.

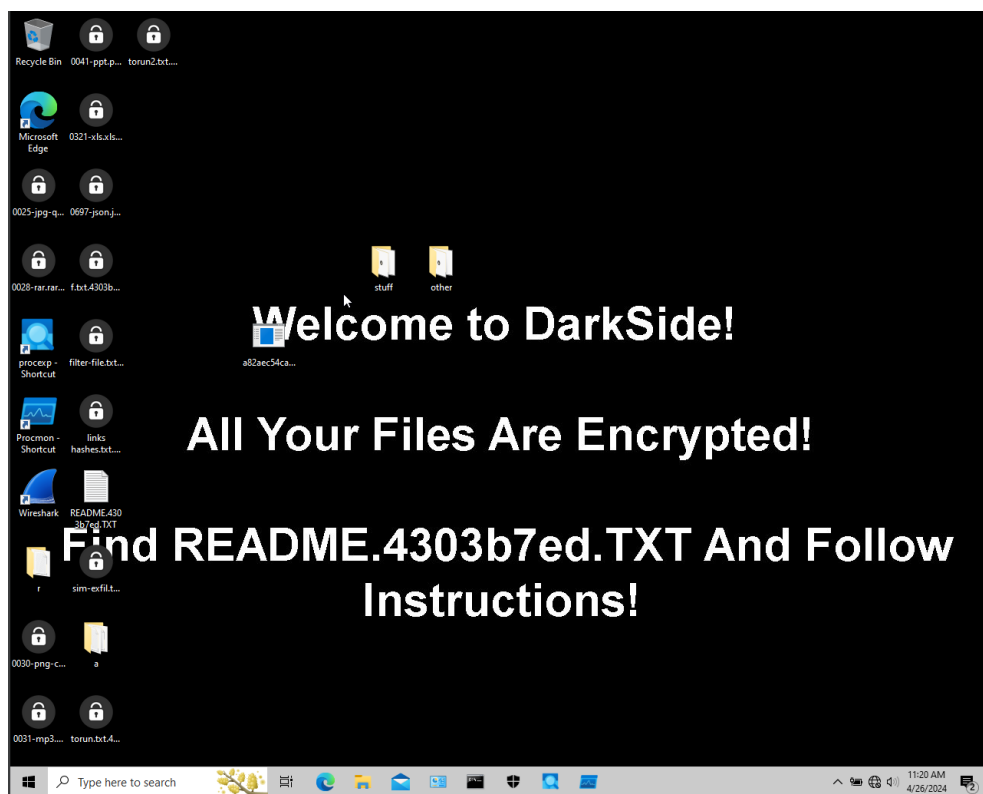


Figure 4.7: Darkside Sample Wrongly Classified as Blackmatter

One observation regarding BlackMatter network traffic is its complete absence in the Procmon program analysis. This should not be due to the creation of subprocesses as their creation has been carefully monitored via Procexp. Analyzing the sample's activities via Procmon, it is believed that tools belonging to the Procmon folder have been modified.

Table 4.5 shows some information about the exfiltration network traffic gathered

with Wireshark. Only BlackMatter and ExMatter are in the Table because all the traffic generated by the other families was not considered related to data exfiltration. As can also be seen in Figure 4.8, all the data exchanged with the C&C server is encrypted and therefore impossible to read but despite this, various analyses can be carried out.

Family	Hash	Packets Gathered	Packets send to C&C	Protocols	Packet Max Size	Other
BlackMatter	a82a...8478	222	123	TLSv1.2 TCP 443	801	
BlackMatter	c6e2...ce99	300	164	TLSv1.2 TCP 80,443	969	2 POST requests
BlackMatter	99d3...3d4b	40	17	TLSv1.2 TCP 443		NO Exfil
BlackMatter	f347...a4c1	274	147	TLSv1.2 TCP 80,443	941	2 POST requests
ExMatter	886c...aadf	5909	2355	TLSv1.2 TCP 22,443	326	
ExMatter	4a0e...7d30	6077	2873	TLSv1.2 TCP 22,443	407	
ExMatter	b6bc...7bd7	5896	2534	TLSv1.2 TCP 22,443	359	
ExMatter	8ede...17ef	6731	2920	TLSv1.2 TCP 22,443	407	

Table 4.5: Relevant Network Activities

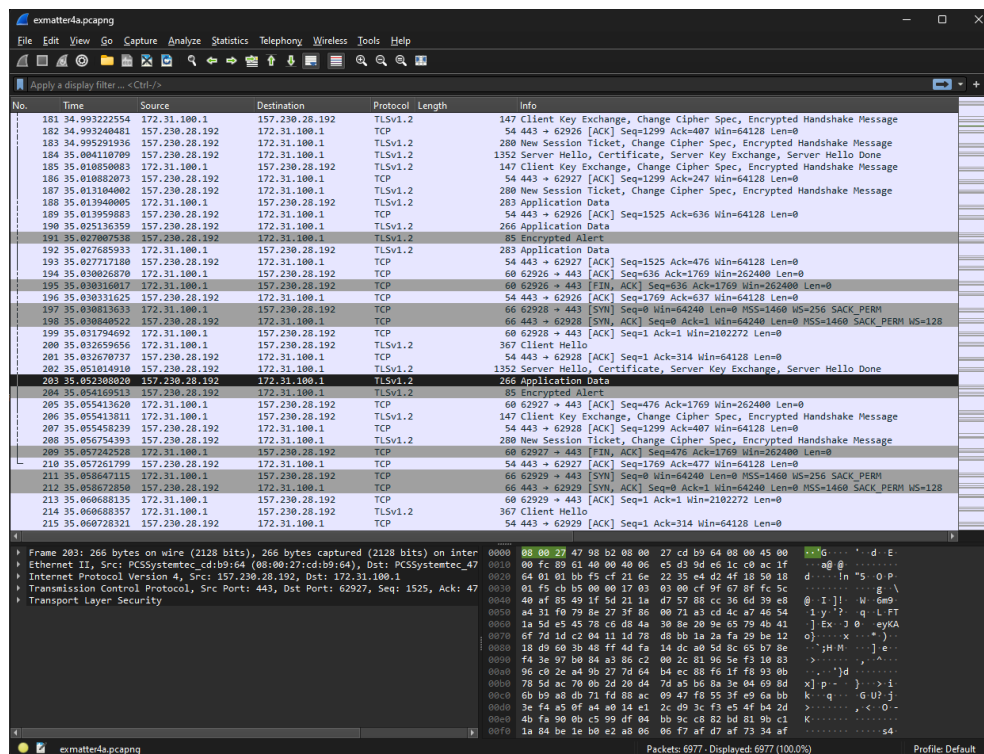


Figure 4.8: Wireshark: ExMatter Network Traffic

Considering the number of packets sent and their size, it can be deduced that only a small part of the data was exfiltrated by BlackMatter and ExMatter. It is not known with certainty which data have been stolen, especially for BlackMatter since by also carrying out the encryption phase, access to a file cannot be traced back to only the exfiltration or encryption phase. On the other hand, ExMatter did not carry out any encryption phase and therefore it can be deduced that all the accesses carried out concerned a potential exfiltration.

### 4.3.2 Static Analysis

The static analysis was conducted on the REMnux machine using the NSA Ghidra tool. The choice to carry out the static analysis on a Linux machine and not a Windows machine is due to being sure that the ransomware cannot be activated inadvertently. Its execution is mainly due to the purpose of being able to learn more information regarding the samples that have behaved anomalously and have not generated any network traffic during the dynamic analysis. After the dynamic analysis of the Stealbit samples, it was noticed that the malware, in addition to not generating any network traffic besides the connection with its C&C server,



ransomware but this step would have required much more time than was available and a very high level of understanding of the assembly language, considering the complexity of the malware.

## 4.4 Ransomware Exfiltration Simulation

In the sample simulation experiment, the environment is first set up, recreating the network topology shown in Figure 3.4. The Windows machine is therefore both connected to the public network and to the private network where the REMnux machine is located. An FTP server is installed on the REMnux machine by using *vsftpd*, while on the Windows machines, Rclone is downloaded and the configuration files are created to set up communication between the Windows machine, the MEGA cloud storage and the FTP server on the REMnux machine. As with the experiments regarding real ransomware samples, before executing the script, Procexp, Procmon and Wireshark are activated on the Windows machine so as to be able to monitor network traffic, process activities, the possible creation of subprocesses and the actual termination of the program.

### 4.4.1 Results

From the analyses carried out, it was observed that the first few steps that the tested methods performed after connecting to the server were to scan the folder and the subfolders to be exfiltrated. In fact, before starting to exfiltrate the data, Procmon observed access to the folders in `C:\Users\Alice` without accessing the files inside them. Only after this phase is completed, the selection of files to exfiltrate begin, based on the set filters. It is important to note that folder scanning is performed more than once during the execution of the tools and during the exfiltration of files. In fact, it is observed that the amount of data to be exfiltrated shown a few seconds after starting the script can increase after a few seconds/minutes. A difference between the methods that carried out the exfiltration to the FTP server and those that instead sent the data to the MEGA cloud storage is the order of access to the files. In all cases, there does not appear to be a specific order for file access but it has been observed that when the FTP protocol is used the files are accessed in random order from different folders on the other hand

once the files in a folder have been accessed, the script remains in that folder until all files have been exfiltrated. The simulation tests were repeated several times, in order to understand whether the different order of access to the files was a coincidence or not. In all the multiple simulations carried out, the results were the same.

Family	Destination	Transfer Data	Filters
Akira	MEGA	2.4GB	–exclude ”*. {MOV, FIT, fit, FIL, fil, mp4, AVI, avi, mov, MOV, iso, exe, dll, psd, PSD, 7z, 7Z, rar, RAR, zip, mox, MOX, wav, WAV, bpm, BPM, mts, MTS, ts, TS, JS, js, ttf, log, map, ai, tmp, TMP, DB, db, mpeg, MPEG, xmp, html, ini, msg, aac, AAC, bak,
Akira	FTP	1.66GB	BAK, DAT, dat, lnk, dwg, indb, indd, svg, idml, ZIP, CAB, EXE, MSI, bin, XML, MMF, DAT, DS.Store, mpp, mp3, m4a, M4A, pkg, gz, ova, iso, mdb, DLL, MP4, mkv, MKV, MP3, WMA, g64x, ufd, vob, VOB, ave, AVE, P01, p01, PO1, po1, dav, DAV, fls, FLS, dist, DIST.c01, C01}
BlackCat	MEGA	546MB	–include *. {jpg, png, doc, docx, pdf, bak, accdb, csv, xlsx, xls, pptx, dwg, xml, html, msg, rtf} –exclude *.*
RagnarLocker	MEGA	1.73GB	–exclude ”*.zip,log,rar,wav,mp4,mpeg”

Table 4.6: Simulation of Ransomware Exfiltration

The network traffic collected by the FTP simulations was composed solely of TCP and FTP-DATA packets, as shown in Figure 4.10. Network traffic is mostly unencrypted, which is why a lot of information can be learned. Respectively, for every FTP packet sent back by the victim machine, at least 10 TCP packets were sent by the server in response. When the FTP protocol is used, analysis has shown that the files to be exfiltrated are divided into 64240b blocks before being sent. The infected machine cyclically requests the list of files present in a specific folder from the FTP server. It is believed that this step is carried out to understand which files have already been sent and which ones still need to be sent to the server.



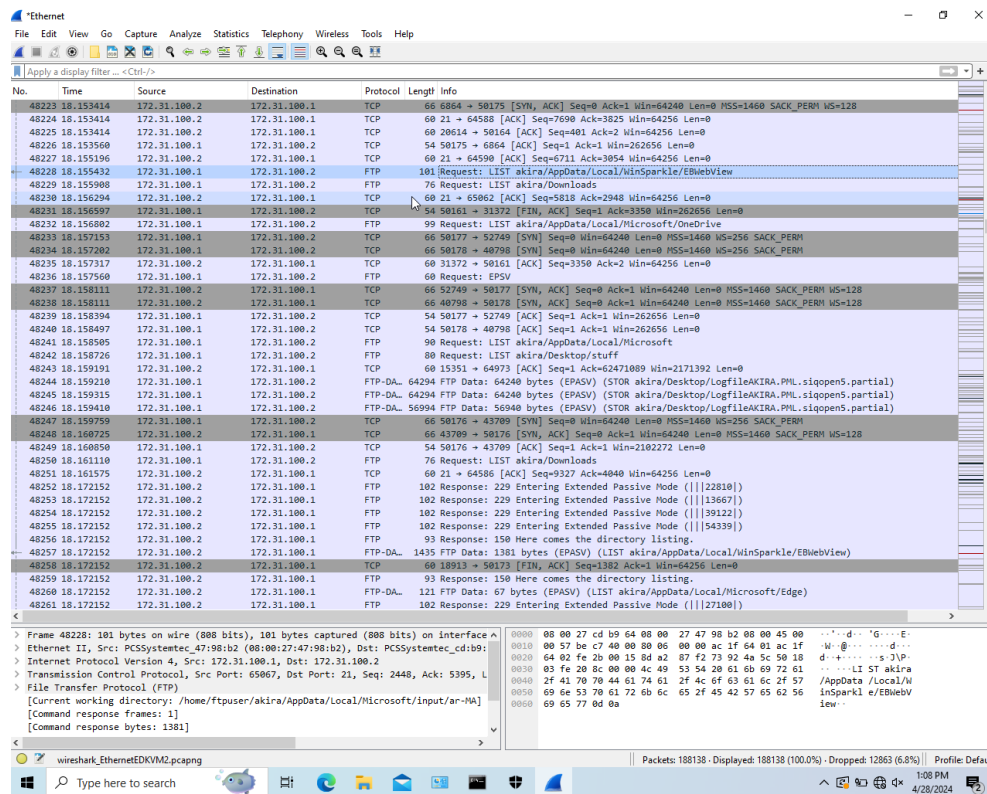


Figure 4.10: Simulation of Exfiltration to an FTP Server

Data exfiltration analyzed from network traffic collected from simulations targeting MEGA cloud storage showed that files are exfiltrated via the HTTP protocol. More precisely, POST requests are made in which the content of the file is encoded in gzip. Unlike simulations with the FTP protocol, the network traffic is all encrypted.

In all the simulations carried out, no traces of the use of compression methods were found. For this reason, it is assumed that if file compression is carried out, this must take place in the staging phase before exfiltration through the use of dedicated methods.

## 4.5 Experiments Evaluation

### 4.5.1 Ransomware Classification

As seen in the literature, the papers usually present their malicious dataset showing the number of samples executed per ransomware family and in some cases

the databases from which the malicious samples were downloaded (Hussain et al. 2023, Almeida & Vasconcelos 2023, Mehrban & Geransayeh 2024). Steps that are almost never described in the papers regarding how the samples were selected and the validations that were carried out to define their belonging to a particular family. Hou et al. (2024) is one of the few that explains the process used to select samples. The python script implemented in the current project takes inspiration from some passages of Hou et al. (2024) to analyze the samples and find their characteristics. From the results obtained from the script and the dynamic analysis, shown in Table 4.4, it can be seen that the results of the script are very promising. In fact, by analyzing the Signature, Tag, Threat Bame, AVclass Family and AVclass Tag columns of the Table 4.4 it is possible to understand, based on what the most recurring information is, which family the ransomware belongs to. The python script, together with the implemented javascript script and web page, constitute a very useful tool for researchers as they allow them to have a fast and reliable analysis of the samples from having only the hashes of some malware. The analysis of the results shows that more than 18.50% samples were misclassified. This result is impressive as it means that if the appropriate validations are not carried out, it is very likely that having a relatively large dataset we will have samples within which do not belong to the hypothesized family. Obviously, this method is not perfect as a manual analysis of the results is still necessary in order to properly classify the samples. Furthermore, the current version 1.0 only searches for information on VirusTotal and MalwareBazaar. We also tried to add the analysis of the Triage database to the python script but unfortunately, we received the API key too late to be able to properly implement the search on this database. These methods can also be used on non-ransomware samples as they scan for information that will still classify other categories of malware. The developed methods lead to a notable improvement in the analysis of samples for the creation of malicious datasets for ransomware experiments because in recent papers the problem of the absence of exfiltration ransomware dataset is expressed and the tool developed in this work can be the solution.

### 4.5.2 Real Ransomware Exfiltration

Most of the samples executed, although they encrypted the files on the target machine, did not performed any data exfiltration. This factor can be given by several reasons. The first reason concerns the checks, carried out by the samples, to understand whether the ransomware is executed on a virtual machine or in a real environment as in the case of Stealbit, as shown in Figure 4.9. The second reason concerns the presence of errors during the execution phase. In fact, some of the samples have raised some errors for reasons related to the absence of .dlls and the wrong location of some files, as can be seen in Table 4.3.1. It is believed that some of these errors may be due to the selected samples which concern only some of the phases of the ransomware kill chain and consequently, in a real case, other files were previously downloaded and some scripts executed in preparation for the samples performed in this research. A further fact that some of the samples did not carry out any exfiltration is believed to be linked to the responses of the server present on the REMnux machine. Although we have tried to recreate an environment more similar to the real one, we do not know what the actual responses that each ransomware expects from its C&C server and this may result in the non-execution of some parts of the code. The last reason why it is believed that some of the samples did not carry out the data exfiltration is due to a possible incorrect classification of the file in the malware database. This factor would mean that some of the samples in question did not steal any data from the victim machine because it was not part of their behaviour. Although it was not possible to analyze the exfiltration of all the samples tested, it was still possible to collect the network traffic for the BlackMatter and ExMatter families. Other papers in the literature review analyzed ransomware data exfiltration (Almeida & Vasconcelos 2023, Hou et al. 2024, Ozturk et al. 2024, Liu & Chen 2023a) but they did include in their dataset those families. The static analysis of the samples was effective in learning information only for StealBit, as for the other samples the Ghidra decompilation time was too high to be able to conduct. Despite this, the information found on Stealbit proved to be important to understand the reason for the lack of exfiltration during the dynamic analysis.

### 4.5.3 Ransomware Simulation Exfiltration

The ransomware exfiltration simulation is based on methods found in the literature using Living off the Land (Demboski 2023, Oren Biderman 2024, Symantec 2024). The experimentation mainly led to identifying the different amount of data exfiltrated based on the method and filters used. As mentioned in the methodology chapter, to ensure that the data taken from NapierOne that populates the victim machine can be exfiltrated, the max-age option has been removed. The simulation can however be defined as unchanged compared to a real environment as the virtual machine was created in the last year and therefore the content of all the files already present in the system is less than 1 year old. Unlike other (Almeida & Vasconcelos 2023) research, data regarding transmission times and speeds were not considered. This choice was made because there would have been a huge difference between the connection between the victim machine and FTP server and the connection with the MEGA cloud storage. This is due to the fact that FTP server is inside the internal network and therefore the network speed is simulated by the virtual environment created by VirtualBox. Furthermore, if not all tests were carried out with a high-performance and stable network, there would be substantial differences not due to the methods used. The simulation of ransomware exfiltration methods allowed us to understand relevant information regarding data exfiltration via Living off the Land. Among the most important data collected are network traffic and file accesses.

## 4.6 Conclusion

This chapter presents the testing phase of this project, showing the results obtained from the test analyses. The testing phase begins with the analysis of the exfiltration-ransomware samples and then the development of the python script. It allows the user to quickly collect important information, used to classify malicious samples. The information is collected from the VirusTotal and Malware Bazaar databases and the AVclass tool is also used to obtain further details regarding the family and the assigned tags to which the analysed samples belong to. This data is fundamental to understanding whether a ransomware sample belongs to a par-

ticular family. The developed script can be useful for any researcher who carries out studies regarding ransomware or other types of malware and who needs to select samples for testing. In fact, since malicious samples are the basis of many searches, it is essential to be able to classify them properly in order to have correct results.

Furthermore, a semi-automatic method has been created that uses a javascript script to be able to collect all the hashes present on Malware Bazaar and Triage and a web page to clean the results obtained. These tools were fundamental for this research as they allowed, after searching for certain families or tags in the databases, to take all the hashes that should belong to a certain category. The output of this process can be given as input to the Python script developed to be able to compare information between different databases.

Through the implemented scripts and an analysis process, 35 malicious samples belonging to 12 ransomware families were selected for the testing phase. The malicious samples were analyzed using dynamic and static analysis and multiple results were achieved. The first concerns the discrepancy between the expected families and the real families regarding some exfiltration-ransowmare samples. Two cases were observed in which the signature and classification carried out by the databases were incorrect. More information about connections to C&C server, encryption and execution time was noted in the chapter. From the static analysis, the presence of CPU-related checks by Stealbit was observed, which is why the dynamic analysis could not give better results regarding the family in question. In the last part of the dynamic analysis, the results of the exfiltration were analysed, which seems to have occurred only in a small part of some samples of the BlackMatter and ExMatter families.

The exfiltration simulation of some ransomware families is described and the network traffic and process activities are analyzed. Finally, the evaluation of the experiments is carried out, highlighting the best results and those where there was not too much success.

## Chapter 5

### Conclusion

This dissertation deals with the topic of ransomware data exfiltration, starting from the analysis of the malicious datasets used in the literature and then continuing with the static and dynamic analysis of the selected samples and finally simulating, through Living off the Land, the exfiltration of data from part of ransomware. The results showed that errors can be found relating to the classification of real ransomware samples in online databases. By developing a script it was possible to carry out an analysis and classification of large datasets in a short time. This script can allow other researchers to more carefully select ransomware samples to run in their experiments. This is considered very important since if the selected samples do not correspond to the families reported in the database metadata it could lead to the failure of an experiment or the improper use of the samples which could not give the expected results. It is believed that the identification of this problem is further relevant for all papers that aim to use machine learning to create defences. In fact, if the starting data is incorrect then the detection system training will probably also be inconsistent.

From the dynamic analysis of exfiltration-ransomware, not much information was found regarding data exfiltration as only two families carried out the exfiltration: BlackMatter and ExMatter. Despite this, it was possible to analyze the network traffic generated and understand some aspects of their exfiltration phase. For other exfiltration-ransomware families, an interaction with their C&C server was observed but did not produce any interesting data. Stealbit was a special case in

that after connecting to its C&C server it immediately stopped execution. After a careful static analysis using Ghidra, functions that took CPU information were identified. This step is usually performed by malware to identify whether it was executed on a real machine or a virtual one. In fact, usually in virtual environments information regarding the computer components is incorrect or dated which can raise the alarm for malware as it is probably a researcher who ran the sample to understand its behaviour.

Finally, simulations of data exfiltration by ransomware were carried out via Living off the Land. For reasons due to the little research on the matter, exfiltration methods were found only via Rclone (which is the most used LOTL tool for data exfiltration) and instead, no scripts regarding WinSCP and FileZilla were found. In the simulation, 4 different methods were performed based on the information found in the literature. From the simulation, information was mainly taken regarding network traffic but further analyses on file access were carried out.

## 5.1 Aims and Objectives

To determine the success of the project, it must be critically evaluated based on the aims and objectives. This project aimed to carry out a behavioural analysis of the ransomware exfiltration phase and to analyse and evaluate the ransomware datasets in order to classify ransomware samples into their families. The behavioural analysis is composed by running real ransomware samples and through the simulation of exfiltration methods through Living off the Land (LOTL).

In order to achieve these aims, the following objectives are defined:

1. Carry out a literature review on the topic of ransomware, focusing on the exfiltration phase. Identify the methods, tools and scripts used by leakware to steal data.
2. From the knowledge acquired from the literature review, develop a methodology and an experimental design to analyse ransomware datasets and the behaviour of ransomware exfiltration.

3. Based on the methodology and design, develop the experiment regarding dataset analysis, real ransomware exfiltration and the simulation of ransomware exfiltration through LOTL. Further, document, analyse and critically evaluate the data found, comparing it with other papers related to this area.

## 5.2 Objective 1: Literature Review

In Chapter 1 the literature review took into account the general aspects of ransomware, dealing with their classification, the Cyber Kill Chain and the phases that define the latest generation crypto-ransomware. Furthermore, some of the most important steps of the evolution of ransomware up to exfiltration-ransomware and double-extortion were observed. Subsequently, we focused increasingly on data exfiltration, starting from the notions also belonging to infostealers up to the exfiltration carried out by ransomware. We analyzed what the targets are, the methods that use LOTL to carry out data exfiltration and ransomware exfiltration software such as ExByte, ExMatter and Stealbit. Finally, the methods used by researchers in the experimentation phase were reviewed, which therefore concern the analysis of ransomware, the target dataset, the testing environment and the exfiltration-ransomware samples.

Chapter 1 has met all the criteria defined in the objectives set in the Introduction.

## 5.3 Objective 2: Methodology and Design

Chapter 3 was documented from the notions learned in the literature review. It contains the methodology and the design of the experiment. The chapter begins by discussing the research methodology composed of 4 fundamental steps that have allowed the advancement of both the understanding of the topic of ransomware exfiltration and the experimental process. Subsequently, the experimental methodology was described in which the phases carried out in the experimental phase were presented in such a way as to both have a well-defined sequence and allow readers to be able to carry out the experiment again. Next, the design of the experiment



is shown. Topics such as testing environments are then covered, which are different for the execution of real exfiltration-ransomware and for the simulation of ransomware exfiltration. The tools and methods used to collect data during the experiment and the selected datasets are described. The datasets are divided into two categories: the target dataset and the malicious dataset. NapierOne is used as the target dataset, while for the malicious dataset, 12 ransomware families / Exfiltration methods were selected for the real testing phase and the methods used by Akira, BlackCat and RagnarLocker for the simulation part.

Chapter 2 delivered the results defined in the objectives.

## **5.4 Objective 3: Experiments, Results and Evaluation**

Chapter 4 describes the testing phase based on the methodology and design defined in Chapter 3. Furthermore, the results of the analyses are shown and their evaluation is carried out. The chapter begins by outlining the various steps taken to carry out the analysis of exfiltration-ransomware by showing the most important parts of the developed script "ransomware-metadata.py". It allows the user to gather information regarding ransomware samples from the VirusTotal and Malware Bazaar databases and the AVclass tool. Through the use of the implemented scripts, various datasets were analyzed to find malicious samples that could be used in the second phase of the experimentation. In the subsequent sections of Chapter 4, the dynamic and static analysis carried out on the malicious dataset is described and the information and results achieved up to this point of the experiment are shown. The results achieved concern the analysis of the exfiltration-ransomware dataset in which discrepancies with the data found in the databases are highlighted, the exfiltration of data by the BlackMatter and ExMatter families and the information collected by Stealbit's static analysis. Subsequently, the last phase of the experimentation concerning the simulation of ransomware exfiltration via Rclone is described. In this case the results mainly focused on the amount of data exfiltrated and the analysis of the network traffic generated by the different methods.

Finally, the experimental evaluation of the three phases of the experimentation is presented.

Chapter 4 delivered the results defined in the objectives even if some experimental steps could have been carried out differently to optimize the results obtained.

## 5.5 Limitations and Future Work

The limitations of this project initially concern the absence of public datasets of exfiltration-ransomware. Furthermore, papers usually do not show the hashes used in their experiments. These reasons led to the need to develop scripts in order to have a dataset with which to carry out the tests. Furthermore, there are multiple databases online which have no further information other than the hash and the sample to download. This has a significant impact on research because these databases cannot be used unless for each hash a check is carried out on other databases, which requires a lot of time since there are no tags or other metadata it is not possible to filter samples for certain groups. A limitation regarding the script is the number of daily requests that can be made with the free VirusTotal API-KEY. This greatly slowed down both the development, to test whether the script worked, and the actual execution in the subsequent phases to obtain information from the online samples. Further limitations concern the simulation of ransomware exfiltration. In fact, apart from Rclone, we don't know the scripts or methods to carry out exfiltration via LOTL.

Possible future work may concern:

- The development of public software to analyze dataset samples, such as this work python script, so that all researchers have a common and correct basis from which to start
- Validation of samples on online databases.
- The creation of ransomware exfiltration datasets in order to make research in this area more effective.
- Double-extortion attack analysis in which the differences between which files

are accessed in the exfiltration and encryption phase can be found.

- The dynamic or static analysis of ransomware exfiltration via LOTL in such a way as to be able to gather the scripts used for exfiltration using tools other than Rclone.

## References

Aboze, B. J. (2024), 'A comprehensive guide to data exfiltration', <https://www.lakera.ai/blog/data-exfiltration#methods-of-data-exfiltration>. Accessed: 18-03-2024.

Agcaoili, J., Ang, M., Earnshaw, E., Gelera, B. & Tamaña, N. (2021), 'Ransomware double extortion and beyond: Revil, clop, and conti', <https://www.trendmicro.com/vinfo/fi/security/news/cybercrime-and-digital-threats/ransomware-double-extortion-and-beyond-revil-clop-and-conti>. Accessed: 05-03-2024.

Agrawal, A. K. et al. (2022), A comparative analysis of open source automated malware tools, *in* '2022 9th International Conference on Computing for Sustainable Global Development (INDIACom)', IEEE, pp. 226–230.

Aleksandar, M. & Kotaro, O. (2021), 'Threat analysis report: Inside the lockbit arsenal - the stealbit exfiltration tool', <https://www.cybereason.com/blog/research/threat-analysis-report-inside-the-lockbit-arsenal-the-stealbit-exfiltration-t>. Accessed: 06-04-2024.

Almeida, G. & Vasconcelos, F. (2023), 'Analyzing data theft ransomware traffic patterns using bert'.

Anand, C. & Shanker, R. (2023), Advancing crypto ransomware with multi level extortion: A peril to critical infrastructure, *in* '2023 2nd International Conference for Innovation in Technology (INOCON)', IEEE, pp. 1–5.

- Benmalek, M. (2024), ‘Ransomware on cyber-physical systems: Taxonomies, case studies, security gaps, and open challenges’, *Internet of Things and Cyber-Physical Systems* .
- Berrueta, E., Morato, D., Magaña, E. & Izal, M. (2020), ‘Open repository for the evaluation of ransomware detection tools’, *IEEE Access* **8**, 65658–65669.
- Breitinger, F., Zhang, X. & Quick, D. (2022), ‘A forensic analysis of rclone and rclone’s prospects for digital forensic investigations of cloud storage’, *Forensic Science International: Digital Investigation* **43**, 301443.
- Chainalysis (2024), ‘Ransomware payments exceed \$1 billion in 2023, hitting record high after 2022 decline’, <https://www.chainalysis.com/blog/ransomware-2024/>. Accessed: 15-03-2024.
- Chung, M.-H., Yang, Y., Wang, L., Cento, G., Jerath, K., Raman, A., Lie, D. & Chignell, M. H. (2023), ‘Implementing data exfiltration defense in situ: A survey of countermeasures and human involvement’, *ACM Computing Surveys* **55**(14s), 1–37.
- Davies, S. R., Macfarlane, R. & Buchanan, W. J. (2022), ‘Napierone: A modern mixed file data set alternative to govdocs1’, *Forensic Science International: Digital Investigation* **40**, 301330.
- de Loaysa Babiano, L. F., Macfarlane, R. & Davies, S. R. (2023), ‘Evaluation of live forensic techniques, towards salsa20-based cryptographic ransomware mitigation’, *Forensic Science International: Digital Investigation* **46**, 301572.
- Demboski, M. (2023), ‘Akira, again: The ransomware that keeps on taking’, <https://news.sophos.com/en-us/2023/12/21/akira-again-the-ransomware-that-keeps-on-taking/>. Accessed: 12-04-2024.
- Eliando, E. & Warsito, A. B. (2023), ‘Lockbit black ransomware on reverse shell: Analysis of infection’, *CogITo Smart Journal* **9**(2), 228–240.

- Gómez Hernández, J. A., García Teodoro, P., Magán Carrión, R. & Rodríguez Gómez, R. (2023), 'Crypto-ransomware: A revision of the state of the art, advances and challenges', *Electronics* **12**(21), 4494.
- Hou, Y., Guo, L., Zhou, C., Xu, Y., Yin, Z., Li, S., Sun, C. & Jiang, Y. (2024), An empirical study of data disruption by ransomware attacks, pp. 1–12.
- Hunters, S. T. (2024), 'Ransomware: Attacks continue to rise as operators adapt to disruption', <https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/ransomware-attacks-exploits>. Accessed: 22-03-2024.
- Hussain, S., Musa, M., Neeshat, T., Batool, R., Ahmed, O., Zaffar, F., Gehani, A., Poggio, A. & Yadav, M. K. (2023), Towards reproducible ransomware analysis, in 'Proceedings of the 16th Cyber Security Experimentation and Test Workshop', pp. 1–9.
- IBM (2023), 'What is data exfiltration?', <https://www.ibm.com/topics/data-exfiltration>. Accessed: 20-03-2024.
- Kennely, J., Goody, K. & Shilko, J. (2024), 'Navigating the maze: Tactics, techniques and procedures associated with maze ransomware incidents', <https://www.mandiant.com/resources/blog/tactics-techniques-procedures-associated-with-maze-ransomware-incidents>. Accessed: 15-04-2024.
- Kok, S., Abdullah, A. & Jhanjhi, N. (2022), 'Early detection of crypto-ransomware using pre-encryption detection algorithm', *Journal of King Saud University-Computer and Information Sciences* **34**(5), 1984–1999.
- Lamping, U. & Warnicke, E. (2004), 'Wireshark user's guide', *Interface* **4**(6), 1.
- Lenaerts-Bergmans, B. (2022), 'What is the cyber kill chain? process & model', <https://www.crowdstrike.com/cybersecurity-101/cyber-kill-chain/>. Accessed: 28-03-2024.
- Liu, S. & Chen, X. (2023a), 'Applying moving target defense against data theft ransomware on windows os'.

- Liu, S. & Chen, X. (2023b), ‘Mitigating data exfiltration ransomware through advanced decoy file strategies’.
- Mayer, D. (2022), ‘Exmatter: Clues to the future of data extortion’, <https://stairwell.com/resources/exmatter-clues-to-the-future-of-data-extortion/>. Accessed: 06-04-2024.
- McIntosh, T., Kayes, A., Chen, Y.-P. P., Ng, A. & Watters, P. (2023), ‘Applying staged event-driven access control to combat ransomware’, *Computers & Security* **128**, 103160.
- Mehra, M. & Pandey, D. (2015), Event triggered malware: A new challenge to sandboxing, in ‘2015 Annual IEEE India Conference (INDICON)’, IEEE, pp. 1–6.
- Mehrban, A. & Geransayeh, S. K. (2024), ‘Ransomware threat mitigation through network traffic analysis and machine learning techniques’, *arXiv preprint arXiv:2401.15285*.
- Meurs, T., Cartwright, E., Cartwright, A., Junger, M. & Abhishta, A. (2024), ‘Deception in double extortion ransomware attacks: An analysis of profitability and credibility’, *Computers & Security* **138**, 103670.
- MS-ISAC (2022), ‘Ransomware: The data exfiltration and double extortion trends’, <https://www.cisecurity.org/insights/blog/ransomware-the-data-exfiltration-and-double-extortion-trends>. Accessed: 27-03-2024.
- Mundt, M. & Baier, H. (2023), ‘Threat-based simulation of data exfiltration toward mitigating multiple ransomware extortions’, *Digital Threats: Research and Practice* **4**(4), 1–23.
- NSA (2019), ‘Ghidra documentation’, <https://ghidra-sre.org/>. Accessed: 13-04-2024.
- Oren Biderman, A. S. (2024), ‘The anatomy of a blackcat (alphv) attack’, <https://www.sygnia.co/blog/blackcat-ransomware/>. Accessed: 12-04-2024.

- Oz, H., Aris, A., Levi, A. & Uluagac, A. S. (2022), ‘A survey on ransomware: Evolution, taxonomy, and defense solutions’, *ACM Computing Surveys (CSUR)* **54**(11s), 1–37.
- Ozturk, M., Demir, A., Arslan, Z. & Caliskan, O. (2024), ‘Dynamic behavioural analysis of privacy-breaching and data theft ransomware’.
- Razaulla, S., Fachkha, C., Markarian, C., Gawanmeh, A., Mansoor, W., Fung, B. C. & Assi, C. (2023), ‘The age of ransomware: A survey on the evolution, taxonomy, and research directions’, *IEEE Access* .
- REMnux (2022), ‘Remnux documentation’, <https://docs.remnux.org/discover-the-tools/explore+network+interactions/services>. Accessed: 10-04-2024.
- Russinovich, M. (2023a), ‘Process explorer v17.05’, <https://learn.microsoft.com/en-us/sysinternals/downloads/process-explorer>. Accessed: 13-04-2024.
- Russinovich, M. (2023b), ‘Process monitor v3.96’, <https://learn.microsoft.com/en-us/sysinternals/downloads/procmon>. Accessed: 13-04-2024.
- Sabir, B., Ullah, F., Babar, M. A. & Gaire, R. (2021), ‘Machine learning for detecting data exfiltration: A review’, *ACM Computing Surveys (CSUR)* **54**(3), 1–47.
- Sebastián, S. & Caballero, J. (2020), Avclass2: Massive malware tag extraction from av labels, in ‘Proceedings of the 36th Annual Computer Security Applications Conference’, pp. 42–53.
- Symantec (2021), ‘Blackmatter: New data exfiltration tool used in attacks’, <https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/blackmatter-data-exfiltration>. Accessed: 06-04-2024.
- Symantec (2022), ‘Exbyte: Blackbyte ransomware attackers deploy new exfiltration tool’, <https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/blackbyte-exbyte-ransomware>. Accessed: 06-04-2024.



Symantec (2024), 'Data exfiltration: Increasing number of tools leveraged by ransomware attackers', <https://symantec-enterprise-blogs.security.com/blogs/threat-intelligence/ransomware-data-exfiltration>. Accessed: 12-04-2024.

Tang, F., Ma, B., Li, J., Zhang, F., Su, J. & Ma, J. (2020), 'Ransomspector: An introspection-based approach to detect crypto ransomware', *Computers & Security* **97**, 101997.

TrendMicro (2020a), 'Info stealer', <https://www.trendmicro.com/vinfo/us/security/definition/info-stealer>. Accessed: 26-02-2024.

TrendMicro (2020b), 'Ransomware double extortion and beyond: Revil, clop, and conti', <https://www.trendmicro.com/vinfo/us/security/definition/info-stealer>. Accessed: 02-03-2024.

TrendMicro (2021a), 'Ransomware spotlight: Conti - security news', <https://www.trendmicro.com/vinfo/gb/security/news/ransomware-spotlight/ransomware-spotlight-conti>. Accessed: 02-04-2024.

TrendMicro (2021b), 'What we know about the darkside ransomware and the us pipeline attack', [https://www.trendmicro.com/en\\_us/research/21/e/what-we-know-about-darkside-ransomware-and-the-us-pipeline-attack.html](https://www.trendmicro.com/en_us/research/21/e/what-we-know-about-darkside-ransomware-and-the-us-pipeline-attack.html). Accessed: 02-04-2024.

TrendMicro (2022a), 'Ransomware spotlight: Black basta - security news', <https://www.trendmicro.com/vinfo/gb/security/news/ransomware-spotlight/ransomware-spotlight-blackbasta>. Accessed: 02-04-2024.

TrendMicro (2022b), 'Ransomware spotlight: Blackbyte - security news', <https://www.trendmicro.com/vinfo/gb/security/news/ransomware-spotlight/ransomware-spotlight-blackbyte>. Accessed: 02-04-2024.

TrendMicro (2023a), 'Ransomware spotlight: Akira - security news', <https://www.trendmicro.com/vinfo/gb/security/news/ransomware-spotlight/ransomware-spotlight-akira>. Accessed: 02-04-2024.

TrendMicro (2023b), ‘Ransomware spotlight: Blackcat - security news’, <https://www.trendmicro.com/vinfo/gb/security/news/ransomware-spotlight/ransomware-spotlight-blackcat>. Accessed: 02-04-2024.

TrendMicro (2023c), ‘Ransomware spotlight: Clop - security news’, <https://www.trendmicro.com/vinfo/gb/security/news/ransomware-spotlight/ransomware-spotlight-clop>. Accessed: 02-04-2024.

TrendMicro (2023d), ‘Ransomware spotlight: Royal - security news’, <https://www.trendmicro.com/vinfo/gb/security/news/ransomware-spotlight/ransomware-spotlight-royal>. Accessed: 02-04-2024.

TrendMicro (2024a), ‘Ransomware spotlight: Hive - security news’, <https://www.trendmicro.com/vinfo/gb/security/news/ransomware-spotlight/ransomware-spotlight-hive>. Accessed: 02-04-2024.

TrendMicro (2024b), ‘Ransomware spotlight: Lockbit - security news’, <https://www.trendmicro.com/vinfo/gb/security/news/ransomware-spotlight/ransomware-spotlight-lockbit>. Accessed: 02-04-2024.

TrendMicro (2024c), ‘Ransomware spotlight: Play - security news’, <https://www.trendmicro.com/vinfo/gb/security/news/ransomware-spotlight/ransomware-spotlight-play>. Accessed: 02-04-2024.

Ugarte-Pedrero, X., Graziano, M. & Balzarotti, D. (2019), ‘A close look at a daily dataset of malware samples’, *ACM Transactions on Privacy and Security (TOPS)* **22**(1), 1–30.

Vasconcelos, F. E. & Almeida, G. S. (2023), ‘Llama assisted reverse engineering of modern ransomware: A comparative analysis with early crypto-ransomware’.

# Appendices

## Appendix 1: Project Management

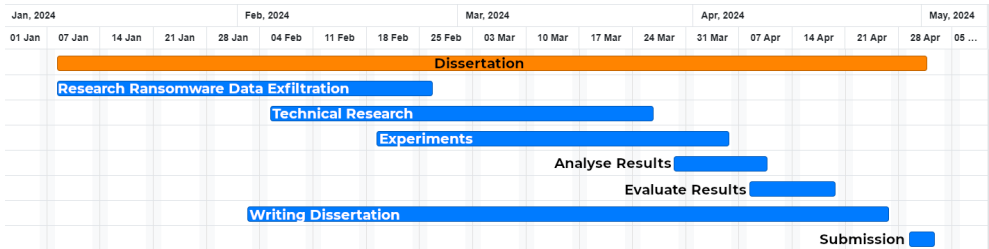


Figure 7.1: Project Management - Gantt Chart

Monday	04/03/2024
09:00 - 12:00 Reading ransomware exfil 14:00 - 18:00 Looking for datasets	
Tuesday	05/03/2024
09:00 - 12:00 exfil. Ransomware analysis 14:00 - 18:00 look for other papers in LT 21:00 - 24:00 Looking for exfil simulation	
Wednesday	06/03/2024
09:00 - 12:00 write lit. rev 14:00 - 16:00 write lit. rev	
Thursday	07/03/2024
09:00 - 10:00 Looking for samples 16:00 - 18:00 Writing scrip for samples analysis	
Friday	08/03/2024
14:00 - 18:00 Solving script problem	
Saturday	09/03/2024
10:00 - 13:00 Looking for improvements for the script	
Sunday	10/03/2024
Day off	
Monday	11/03/2024
09:00 - 12:00 Analysis of data gathered from the script 14:00 - 18:00 Dealing with bugs in the python script	

Figure 7.2: Example of diary entry

## Appendix 2: Scripts source code on GitHub

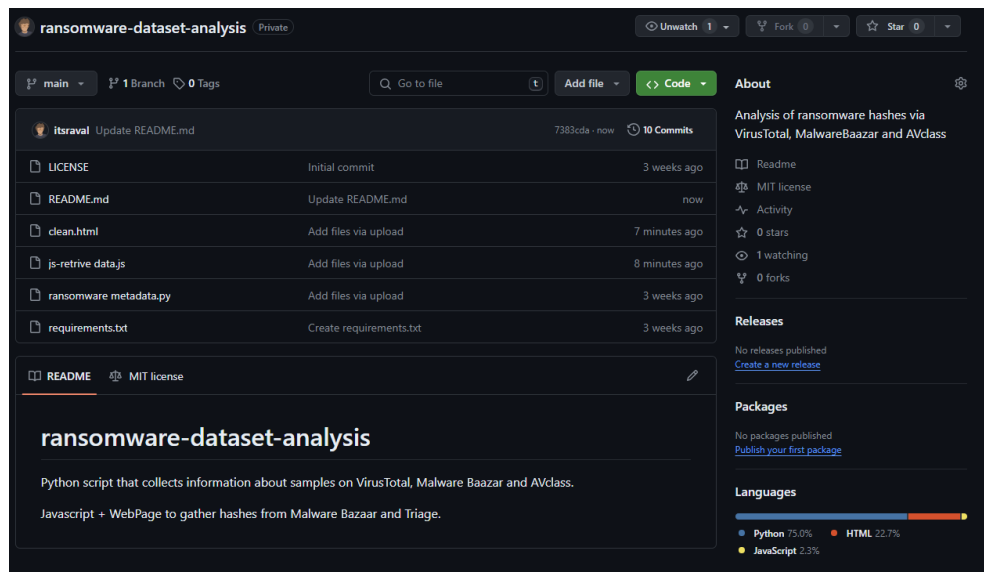


Figure 7.3: Scripts Source Code on GitHub

GitHub: <https://github.com/itsraval/ransomware-dataset-analysis>

## Appendix 3: Exfiltration-based Ransomware samples used

Index	SHA-256
Akira	4102ee321317c71841798078072a6c13b6a4890bbec7aa9c31f6ac7d0d4afaef
Akira	50e36d96cb593c39afa2fc11ac25c976f0ff1586159d2eb2626902e6d6062f81
BlackBasta	5d2204f3a20e163120f52a2e3595db19890050b2faa96c6cba6b094b0a52b0aa
BlackBasta	ae7c868713e1d02b4db60128c651eb1e3f6a33c02544cc4cb57c3aa6c6581b6e
BlackBasta	2558d0817586306d0ddf7beadd371785cd0a0b7ed860ac62760dbbc92866008a
BlackByte	1df11bc19aa52b623bdf15380e3fded56d8eb6fb7b53a2240779864b1a6474ad
BlackByte	884e96a75dc568075e845ccac2d4b4ccecc68017e6ef258c7c03da8c88a597534
BlackByte	c22a6401a415fe642f3d96f38a887dd8ad23dd83a9255ee89d9adf4650ab98da
BlackMatter	a82aec54cad176b368967fa8e41e41a8129ffafe6ab627312e111e63605b8478
BlackMatter	c6e2ef30a86baa670590bd21acf5b91822117e0cbe6060060bc5fe0182dace99
BlackMatter	99d3003cc577c3635bcb883634037469b35c3b1a31109dc145600bb1e683d4b
BlackMatter	f3474589cafa855a73d0830883b9909095f82c28aa468e999940faf85beca4c1
BlackCat	3c300726acdd8a39230f0775ea726c2d42838ac7ff53bfdd7c58d28df4182d5
BlackCat	7d8671c91a02bfbff8b89a76501b9be017a66a8bba624ed4fe2c7f81b9380ac9
Cl0p	220c50ccf6a9d9727e9f442df42469f027d9f7a2ea833319971746280023bb0c
Cl0p	3320f11728458d01eef62e10e48897ec1c2277c1fe1aa2d471a16b4dccfc1207
Cl0p	15f9ed36d9efc6e570b4f506791ce2c6a849853e2f6d587f30fb12d39dba2649
Conti	1ce8a939b3e7d84c59c12dc9e1091532f4336dac533847b6533b01d9dcf494e9
Conti	4bfd58d4e4a6fe5e91b408bc190a24d352124902085f9c2da948ad7d79b72618
Conti	837c0f1e9749121e4f8204c2b5546964c10d4e3d85a514458c35cbc021762d5c
ExByte	0097b8722c8c0840e8c1a4dd579438344b3e6b4d630d17b0bbe9c55159f43142
ExByte	3fb160e1770fafeedff2d77841bf02108c25cca4cb6d77e3fbf759077f356b70
ExMatter	886cb22ffe43a3838ef152ef57bbfa66f52b71c534bfe3d8af3d29ea973daadf
ExMatter	4a0e10e1e9fea0906379f99fa350b91c2af37f0fd2cc55491643cc71a9887d30
ExMatter	b6bc126526e27c98a94aab16989864161db1b3a75f18bd5c72bacbdfccad7bd7
ExMatter	8eded48c166f50be5ac33be4b010b09f911ffc155a3ab76821e4febd369d17ef
HIVE	0077e7d6e90ad972b64e90c343c617482f39505deff44ebff99ff49041252dcb
HIVE	19c54b520e8f6def1aab0c48fdbac5962a0ec57442a3a015a0f860a76115718
HIVE	16baebd1adfc1bae6e35773b383875ac831a011fefed63a0506b875596274b8c
RYUK	e6762cb7d09cd90d5469e3c3bfc3b47979cd67aa06c06e893015a87b0348c32c
RYUK	a1ce52437252001b56c9ccd2d2da46240dc38db8074a5ed39a396e8c8e387fc2
Stealbit	6b795d9faa48ce3ae31f0bde3dcb61a6d738e8cc0e29b5949d93a5c8ee74786a
Stealbit	61ac7ac908791456f2f5827dfd85be27b02027383f76dfd31aba7eff89c1aaee
Stealbit	07a3dcb8d9b062fb480692fa33d12da05c21f544492cbaf9207956ac647ba9ae
Stealbit	968875370dbc26a6439860f854c91f9ee675e588f8dbf78e6cb7e20b6d957bec

Table 7.1: Exfiltration-based Ransomware samples used