



What is a ransomware and how does it work?

Luca Lamsouber

2023 Laurea



Laurea University of Applied Sciences

What is a ransomware and how does it work?

Luca Lamsouber
Business Information Technology
Thesis
November, 2023

Luca Lamsouber

What is a ransomware and how does it work?

Year

2023

Pages

38

The purpose of this bachelor's thesis was to create an educational guide about ransomware for organizations as well as other individuals, which raises awareness in defending against ransomware attacks. In this thesis a well-known ransomware called "Black Basta" was investigated and analyzed. One objective was to compile a comprehensive theoretical framework about malware, focusing on ransomware, which provides the necessary foundation for later understanding how ransomware attacks work. Another goal was to examine what happens when a ransomware gets detonated.

The theoretical part of this thesis answers to questions such as what malware are, how do they differ from other malware, how do ransomware work. This part also goes through the process of a ransomware attack and how attackers succeed in a ransomware attack. The empirical part of this thesis was done with qualitative methods. It further inspects the topic on a practical level to truly gain an understanding about ransomware.

The outcome of this thesis was successful since an educational guide that answers to each research questions with theoretical or practical examples was compiled. As ransomware attacks evolve rapidly, it is essential to be up to date with this subject and the tactics, techniques, and procedures that the cyber attackers use. This thesis can be used for this purpose of teaching organizational staff or other individuals wanting to be aware on how to be prepared for a ransomware attack.

Keywords: ransomware, cybersecurity, malware, malware analysis

Tämän opinnäytetyön tarkoituksena oli luoda yrityksille sekä yksityisille henkilöille koulutusopas kiristysohjelmista, joka auttaa suojautumaan kiristyshyökkäyksiä vastaan. Tässä opinnäytetyössä tutkittiin ja analysoitiin hyvin tunnettua kiristysohjelmaa nimeltä ”Black Basta”. Yhdenä tavoitteena oli koota kattava tietoperusta haittaohjelmista, keskittyen kiristysohjelmiin, joka tarjoaa tarvittavan pohjan myöhemmin ymmärtääkseen, miten kiristyshyökkäykset toimivat. Toinen tavoite oli tutkia, mitä tapahtuu, kun kiristysohjelma laukaistaan.

Tämän opinnäytetyön teoriaosuus vastaa kysymyksiin, mitä haittaohjelmat ovat, miten ne eroavat muista haittaohjelmista, miten kiristysohjelmat toimivat. Tämä osio myös käy läpi kiristyshyökkäyksen prosessit ja miten hyökkääjä onnistuu kiristyshyökkäyksessä. Opinnäytetyön tutkimusmenetelmät olivat kokonaisuudessaan kvalitatiivisia. Tutkimusosassa tarkastellaan syvemmin aihetta käytännön tasolla saadakseen entistä paremman ymmärryksen kiristysohjelmista.

Opinnäytetyön lopputulos oli onnistunut, sillä kattava opetusopas, joka vastaa jokaiseen tutkimuskysymyksen teoreettisiin tai käytännön esimerkein onnistuttiin kokoamaan. Koska kiristyshyökkäykset kehittyvät nopeasti, on tärkeää olla ajan tasalla tästä aiheesta ja kyberhyökkääjien käyttämistä taktiikoista, tekniikoista ja menetelmistä. Tämän opinnäytetyön avulla voidaan opettaa yritysten henkilökuntaa tai muita henkilöitä, jotka haluavat olla tietoisia siitä, kuinka välttyä kiristyshyökkäyksiin lankeamiselta.

Table of Contents

1	Introduction	5
2	Background of the thesis	6
3	Malware.....	6
3.1	History of Malware	7
3.2	Types of Malware.....	8
3.2.1	Viruses and worms.....	9
3.2.2	Trojan	9
3.2.3	Ransomware	9
3.3	Statistics about malware	10
4	Implementation of a ransomware attack	12
4.1	Ransomware's functionality	12
4.2	Cyber Kill Chain	13
4.3	Stages of a ransomware attack.....	15
5	Testing and analyzing a ransomware in a safe environment	16
5.1	Testing environment	17
5.2	Malware demo	20
6	Summary & conclusion.....	34
	References.....	35
	Figures	38

1 Introduction

Ransomware isn't a new topic in the cyber industry, but it has become even more relevant than ever. Organizations fall into ransomware attacks continuously and this is difficult to mitigate as the threat actors keep constantly evolving their tactics, techniques, and procedures.

This thesis is an educational guide to gain knowledge about malware, understanding how threat actors' processes work in ransomware attacks, and how a ransomware functions in practice. With this, the reader should be able to avoid falling into a ransomware attack even in the earlier stages.

Chapters 3 & 4 are theoretical, and they will give the reader an understanding on what is a malware, what types of malware are there, history of malware, malware statistics, how ransomware operate, what cyber kill chain is, what stages are in a ransomware attack, and what happens in each stage of a ransomware attack. Chapter 5 is about testing and analyzing a ran-

somware sample and trying to understand the behavior of it. This will demonstrate how ransomware work and show in practice what happens when a ransomware gets detonated. This chapter also covers malware analysis on a basic level with practical examples of static and dynamic analysis methods.

2 Background of the thesis

Ransomware attacks have become very popular recently and this is why I wanted to further educate myself on this topic and to compile an educational guide about ransomware. This work is for the new people in the cyber industry, for them who wants to be more familiar with this topic, and for organizations as this gives all around information about malware, and more specifically about ransomware.

This thesis teaches what kinds of malware are there, statistics and history of malware, how do malware operate, how do threat actors succeed in ransomware attack, ransomware attacks processes, how to setup an environment where you can analyze malware, and a demo on what a famous ransomware looks like. It is good to know how threat actors do their attack and how their processes work to create even better defensive methods. Ransomware can do a lot of damage and it could cost organizations millions if they get attacked by a ransomware, so it is important to be aware of this topic.

Research methods used in this thesis are a mixture of theoretical and empirical research methods. The theoretical method gives the reader a clear understanding about malware, their operations, and processes. In the empirical research part, a ransomware was demonstrated, observed, tested, and analyzed.

3 Malware

The term malware comes from words “malicious” and “software”. It is a software which intentions are malicious. It is not a typical software that you would purchase or install knowingly, but instead, it is installed onto your systems to gain access and to perform malicious acts. Some symptoms of an infected system can be slow performance, unexpected pop-up windows, starting unknown processes, loss of bandwidth, etc. (Gibson, D. 2017)

Malware can be delivered through various methods such as via phishing emails, downloading something from a compromised website, plugging in an infected USB drive, clicking on a pop-up window and more. (Theprisi, T. 2023)

Some people have the misconception that virus and malware is the same thing which is incorrect. Virus is only a type of malware, and malware has lots of different forms such as worms, adware, logic bombs, Trojans, rootkits, to name a few. (Gibson, D. 2017)

Malware is continuously evolving and according to AV-TEST Institute, they register over 450 000 new malware and potentially unwanted applications (PUA) every day. These new malware are usually just slightly modified versions of existing malware. (AV-TEST. 2023)

3.1 History of Malware

The history of malware dates to the late 1940s, when a Hungarian mathematician John Von Neumann created a theory about self-replicating computer programs. Some of his work including this concept we now know as malware, was compiled in 1966 to a paper called “Theory of self-Reproducing Automata”. (Livingston, Z. 2022)

While the theory of a malicious program relates to the 1940s, malware’s first actual proof of concept was introduced in the year 1971, when a computer programmer Bob Thomas created the first computer virus called “Creeper”, named after a character from Scooby-Doo. Though the program is usually credited as the first “virus”, it behaves like a computer worm with the difference of spreading without human interaction. Bob Thomas wanted to know if an executing program can have the possibility to move between computers without interrupting ongoing operations of a program, and this led to the creation of Creeper. The creeper isn’t technically classified as a malware because its intentions were not malicious and the only thing it did was to display a message “I’m the creeper, catch me if you can”. This program was released to the ARPANET, the precursor of today’s internet, and it would move around between computers displaying the aforementioned message. The program was coded to remove previous copies of itself from the host before moving to another computer, and in later versions (created by Ray Tomlinson) it would copy itself to other computer to make it spread like a virus. The program became an annoyance to some, and later Tomlinson created a different program called “Reaper” which basically cleaned up the mess and became the first antivirus software. (Bagde, A. 2021)

The second most significant event about malware happened in 1982 when a high school student made a practical joke by writing a piece of code. Little did he know that piece of code turned out to be the first known malware on Apple computers. This was a boot sector virus called “Elk Cloner” and it would only target Apple 2 computers. At that time when floppy disks were common, they were passed around to share software’s. Richard Skrenta, who is the creator of Elk Cloner started this by distributing altered floppy disks to his friends as a prank. This boot sector virus made a copy of the virus in the computer’s memory and then stayed there to find a new clean floppy disk to spread the virus. The Elk Cloner was not as harmful as nowadays malware and everything it would do was to display a poem every 50th

boot. Despite of its harmlessness, the program was the first one to spread in the wild, so also untargeted people were affected by Elk Cloner. (Saengphaibul, V; Kelly R. 2022)

One remarkable event in malware history happened in 1986 when first PC virus was developed. This virus was created by two brothers from Pakistan with their intention of proving that PCs are not secure. This was also a boot sector virus, meaning that it spread by sharing floppy disks. The virus listed information about the creators including phone number so that people can contact them about the virus. This virus had a global impact since it spread from Pakistan to USA. The virus became well-known and after this point in time, new malware started to appear more regularly with little modifications. Information security became important after this event. (Milosevic, N. 2013)

In 1989 appeared the first ransomware which was far more crucial than the above-mentioned viruses. The ransomware was a trojan, meaning it disguises its true malicious intent by acting as a normal software that don't make anyone suspicious. This ransomware's name was "AIDS Trojan" created by Joseph Popp. The ransomware was in a floppy disk and the disks were distributed to researchers worldwide by physical mail. The disks had questions related to human AIDS which was relevant at that time, and it would act as a normal software. The true intention became clear when the disk was booted the 90th time displaying a ransom note and encrypting/hiding files from the user. The demands were to send a specific amount of money to a PO Box located in Panama. The encryption key to recover all the files was "Dr. Joseph Lewis Andrew Popp Jr.". (Saengphaibul, V. 2022)

The history of malware has lots of significant events, but these are some of the most important ones. Malware started in a form of a simple virus, without doing any harm and now they have become more complicated and malicious with the intent of doing harm.

3.2 Types of Malware

There are different unique malware types where each works differently. Common types of malware are viruses, worms, trojans, and ransomware. In the wild, there are plenty of other malware such as polymorphic malware, fileless malware, keyloggers, bots, botnets, and so forth. Adversaries sometimes need to get through different defense mechanisms like anti-virus and anti-malware solutions, so they need to use different kind of evasion and obfuscation techniques to get through them. To avoid signature based anti-virus solutions, adversary could use a polymorphic malware which modifies its code to get through signature-based detections. The anti-virus solution might detect the malware's hash value and say that it is harmful, but as the malware keeps changing the code, the hash value also changes, and the anti-virus solution doesn't anymore understand that it is the same harmful malware. It is good to understand what types of malware are out there and how do they differ with each other to know how to defend against the threat actors. (Tunggal, A. 2023)

3.2.1 Viruses and worms

Viruses and worms are one of the most common malware types that exists. They have their resemblance, but they are little bit different from each other. Virus is a piece of code that spreads from computer to computer. It must be executed in order to perform its malicious act. The execution can happen by opening an attachment you got to your email or simply by inserting a USB drive. Virus attaches itself to a program and can spread in the system when it is executed. Viruses need some kind of human interaction for example clicking a link, but worms in other hand don't.

Worms are basically the same thing as virus, but they can spread in a network without the need of a user action. A worm uses transport protocols to move around across a network. Some of the most common signs of a device that has been infected with a virus or worm can be slow computer performance, loss of network bandwidth, system crashes modified or missing files and more. They both are alike with the only key difference being viruses need some kind of interaction and worms can self-reproduce by themselves. (Malwarebytes)

3.2.2 Trojan

Trojan is a type of malware that got its name from the trojan horse in Greek mythology. This computer trojan has the same functionality, it looks something good but ends up being a malicious trap. Trojan can for example be a normal looking software that you download and install, but when you run it, it performs other actions that the victim is not aware of such as stealing information, damage computer or files, making a backdoor so that attackers can get access to the machine and more. (Aycock, J. 2006) (McAfee)

One common type of trojan is a trojan-downloader which lays low on the system and waits for an internet connection, so that it can later connect to remote servers to download malicious programs to the infected system. (F-Secure)

3.2.3 Ransomware

One type of malware that has expanded incredibly in the past years is ransomware. Ransomware is a malware that gets control on victim's data and usually encrypts them so that the victim can no longer access the data. The attacker then demands a payment from the victim if they want to retrieve their data back. To get the data back you need the decryption key to get all the encrypted files back. If the victim chooses to pay the ransom for the attacker in hoping to receive the decryption key, there is no guarantee the attacker gives the key or even has it.

Ransomware has evolved significantly and has different forms or types. The most common type is crypto ransomware which encrypts all the data in a system and to access them, you

need a decryption key. One very similar type is a locker which completely locks the user out of their system, and then usually a screen is displayed informing about the ransom. Other ones can be scareware which purpose is to scare without doing any damage. Scareware usually floods the users screen with pop up alerts. Doxware/ leakware is another type of ransomware which claims to have personal information and they will share that information on public if a ransom is not paid. (Crowdstrike, 2023)

In research done by cybersecurity venture, it was said that the annual ransomware damage cost for victims will be 265 billion dollars by 2031 and that a new ransomware-attack for organizations will occur every 2 second. The estimations in 2019 was that these attacks occurred every 14 seconds and in 2021 every 11 seconds. The growth of ransomware-attacks has been very noticeable in the past years. (Morgan, S, 2023)

3.3 Statistics about malware

AV-TEST Institute is information technology related research institute from Germany. They have one of the biggest malware sample databases in the world. They have analyzed the data about malware and with this we can easily see what types of malware are common and which operating systems have the most malware.

In the figure below (Figure 1), the red bars display the amount of malware and blue ones are potentially unwanted applications (PUA). The first chart shows the total amount of malware that AV-TEST Institute have in their collections. In 2023, the number of malware is little over one billion and number of PUAs is 200 million. 10 Years ago, there was only 100 000 malware, so there has been a clear rise on them.

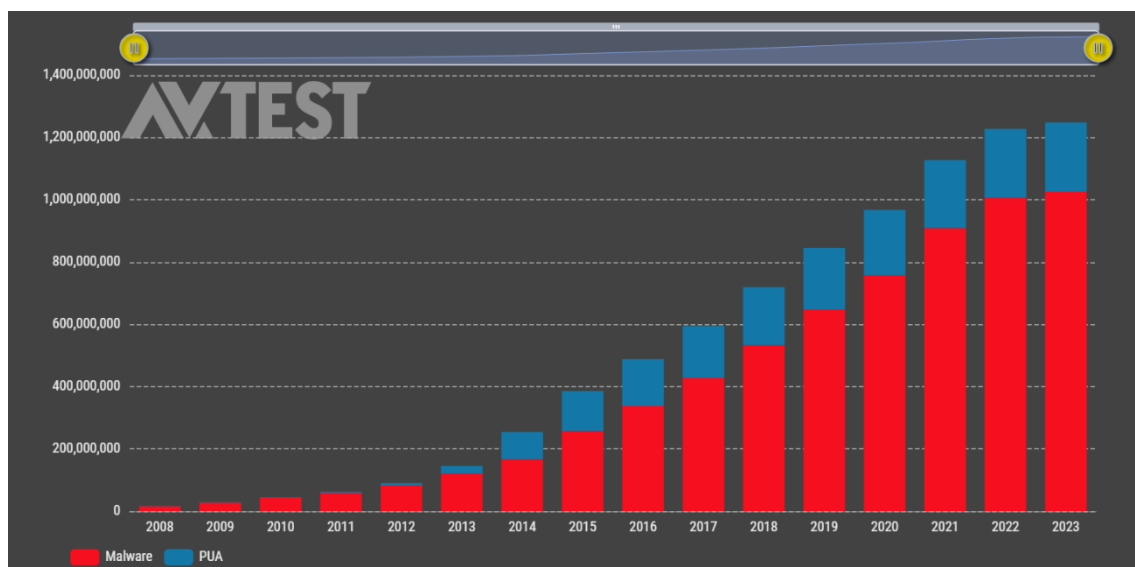


Figure 1: Total amount of malware and PUA (AV-Test 2023)

If we compare the total amount of malware by operating systems, Windows is the most affected. Almost 800 million of these malware have been detected in Windows machine, which is 80% of all the malware. Android has the second largest amount of malware detections which is only 30 million. Linux has little over 4 million and MacOS almost one million. With this information we can see that Windows operating system is the most targeted one. AV-TEST Institute has analyzed which malware are the most common in Windows operating system and in the chart below we can see newly discovered malware from the past 14 days. The total amount of malware in Windows machines has been over 3 million and one quarter of them has been trojan horses. The second most common malware in this operating system has been Worms with over 600 000 samples. Downloaders have been lately the third most common malware in Windows operating system with over 450 000 samples according to AV-TEST Institutes research. According to the analysis there has been over 300 000 samples of backdoors and the last category in the figure below (Figure 2) is “other” which has 500 000 malware samples. (AV-Test. 2023)

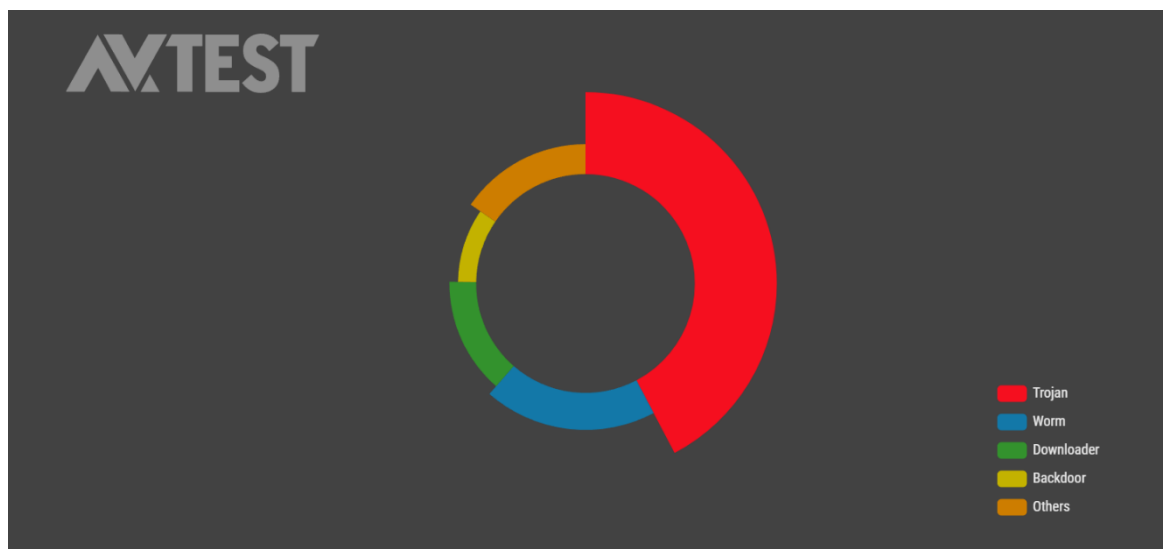


Figure 2: Windows malware categories (AV-Test 2023)

Although Windows operating systems are the most likely to be a target of a malware, it is still important to make other operating systems as secure as possible. Over half of the malware sample on Linux operating systems have been detected in the past two years. Five years ago, Linux was barely targeted with these malware but these days it is not so uncommon anymore.

4 Implementation of a ransomware attack

Ransomware have been one of the most growing types of malware recently and many organizations have been targeted by ransomware attacks. To better defend for these attacks, it is good to know how threat actors think when they plan and execute ransomware attacks.

When threat actors do their malicious intents for example by delivering a ransomware and getting the user to execute the malware, there are lots of steps and preparing that the threat actors do. It is not an easy process, and this chapter explains how ransomware operate and what cyber kill chain (CKC) is and how do malicious actors implement that framework in their attacks.

4.1 Ransomware's functionality

Over the years, techniques used for ransomware attacks have become far more complex and now there are multiple methods on how to get the malware from attacker to the target without anyone noticing a thing. Ransomware can be delivered in many ways. Usually, attackers deliver their ransomware via email with an attachment or malicious link, and this is called phishing. There are other delivery methods such as delivering malware through drive-by downloads, removable media like USB drive, exploit kits, malvertisement which is a malicious advertisement and more.

After the ransomware is delivered, it then stays on the system and tries to locate all the user data on the system while evading important directories so that the system's stability remains. When the ransomware has located the important user data, it then encrypts the data usually with symmetric or asymmetric encryption. Symmetric encryption uses the same key for the data encryption and decryption, which means that if the password for encryption is "Test", then the password for decryption also is "Test". Asymmetric encryption uses a public key and a private key which means when the data is encrypted with the public key, you then need the private key to decrypt the data and vice versa. (Gibson, D. 2017) After the ransomware has encrypted the data on the system, it sometimes tries to exploit vulnerabilities to spread to other systems but normally at this point the ransomware displays a ransom note on the screen with instructions on how the user can retrieve their data back and how much money or cryptocurrency they need to pay. (Miller, L. 2020)

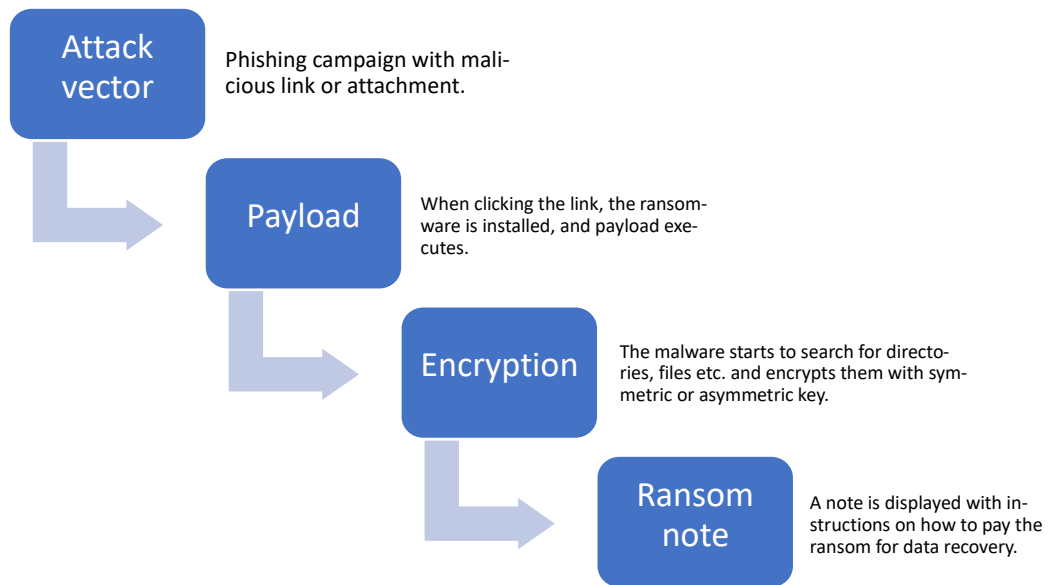


Figure 3: How ransomware functions (adapted from Miller. L 2020)

It is important to know that if you fall into these traps and your files get encrypted, you shouldn't pay for the ransom because there is no guarantee that the malicious actor will give you the necessary key for the decryption of the data. This only tells the bad actor that their ransomware works. To recover files by yourself you can try to use free ransomware decryption tools such as No more ransom decryptor. In some cases, you can back up your data and wait for a decryptor to be published to retrieve your data eventually. Ransomware are as annoying as any other malware, and everyone should think twice before opening any attachments or clicking links to avoid these situations.

4.2 Cyber Kill Chain

Advanced Persistent Threats (ATP) are deeply planned, complex intrusion campaigns that modern day's adversaries are doing. To analyze and understand how these attacks work and how to defend against them, we need to use attack frameworks specifically designed for defending and understanding what stages sophisticated attacker will go through. There are different well known attack frameworks to choose from and, in some cases, organizations might create their own attack framework, but this section will describe what cyber kill chain is and how it can be used in defending against cyber intrusions. (Chapple & Seidl, 2020)

Cyber kill chain is one of the most well-known attack frameworks used by cybersecurity specialists. This framework was originally created by Lockheed Martin in 2011. The model's purpose is to identify what stages adversaries take to achieve their goal. If any of these chained links (stages/steps) break, adversaries plan for intrusion will fail. Cyber kill chain consists of 7 stages as shown in Figure 4: Cyber kill chain (adapted from Martin. L 2015).

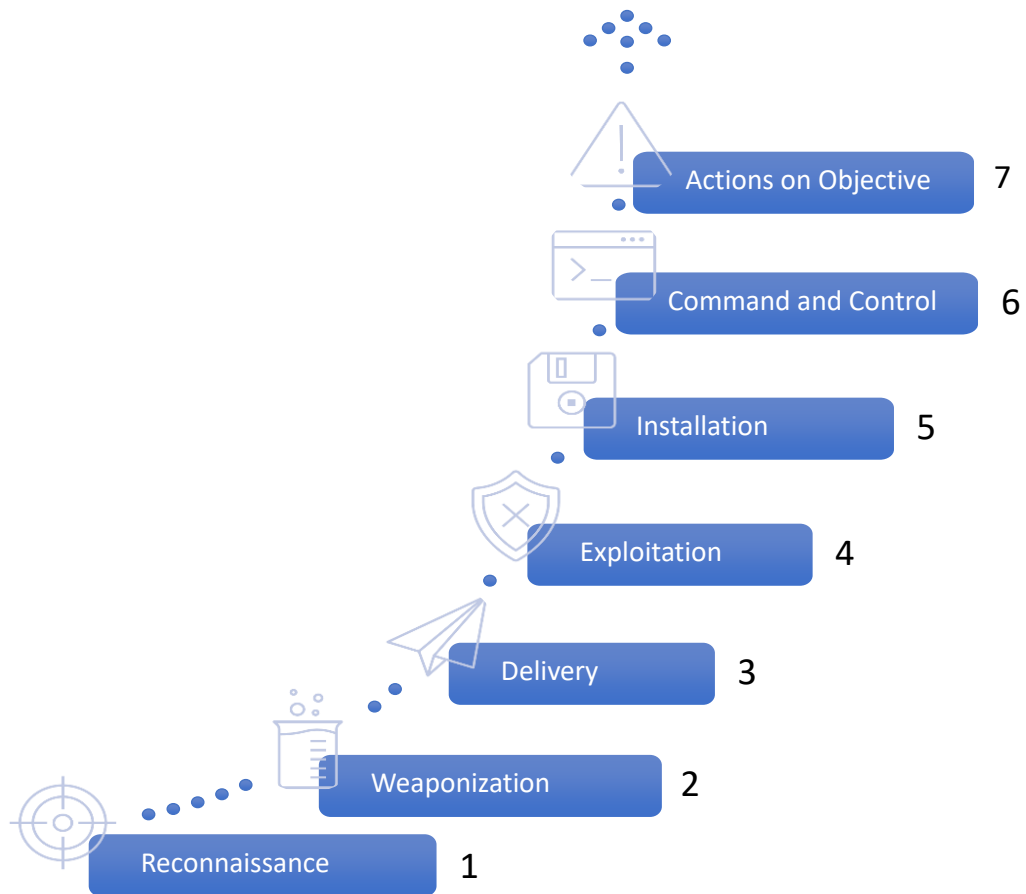


Figure 4: Cyber kill chain (adapted from Martin. L 2015)

Cyber kill chain starts with a reconnaissance phase where the attacker plans, selects their target, and does their research on what kinds of vulnerabilities they can exploit. The more information the attacker gathers in this phase, the more complex and successful the attack is. (Martin, L. 2015) (Yadav, T & Mallari, A. 2015)

Weaponization is the phase where attacker utilizes the information gathered in reconnaissance phase and designs the right attack vector that can be used as an exploit. This phase is important for the defenders to understand what payload the attackers used and how it was made. (Yadav, T & Mallari, A. 2015)

The third stage in cyber kill chain is delivery in which the attacker launches their operation for example by sending an email containing the payload, with a storage device, or from a malicious website depending on the scenario. (Yadav, T & Mallari, A. 2015)

After the attacker has successfully delivered the payload, then comes the exploitation stage where the payload is executed, and the attacker gains a foothold to the target's environment. This can happen in various ways such as plugging in a USB drive, opening attachment

received via email, drive by download or by clicking malicious links. (Yadav, T & Mallari, A. 2015)

Fifth stage is installation, and here usually attacker creates a backdoor to communicate with external parties and to gain persistence on the system. After this starts a stage called command and control. This is where the attacker establishes a two-way communicating channel with a remote server to exfiltrate sensitive data, getting further control, to download more tools for expanding the attack surface, and to stay undetected by mimicking ordinary traffic. (Yadav, T & Mallari, A. 2015)

The seventh and final stage is actions on objectives, and this is when the goal of the attack is succeeded. Here the attacker usually has full control and can do whatever they want meaning they can encrypt all the data, collect credentials, move laterally, and escalate privileges, do damage on the system or network, exfiltrate confidential information, and more. (Martin, L. 2015) (Yadav, T & Mallari, A. 2015)

4.3 Stages of a ransomware attack

This section goes quickly through the cyber kill chain specifically in ransomware attacks. A ransomware attack can have 4 to 8 steps depending on how you look at the situation. If the payment steps are included in the lifecycle, then the cyber kill chain will be longer.

Exabeam has analyzed the behavior of 86 strains of ransomware and in their ransomware threat report, they included the most common steps involved in typical ransomware attack. The Exabeam's ransomware lifecycle has 6 steps, and these steps are almost always involved in a common ransomware attack to achieve the goal to deliver the ransomware to the target computer and then to find files, encrypt them and demand ransom. The ransomware lifecycle steps are shown in figure 5. (Exabeam, 2017))



Figure 5: Ransomware kill chain (adapted from Exabeam 2017)

The first step usually starts with social engineering which can be in a form of a phishing attack. That is usually the easiest way to get the target to make an error which is by accidentally opening some attachment in an email. The attachment can be a word document with macros. If you are not familiar with macros, they are used to automate tasks with commands. When macros are enabled in word, they can do something malicious. The phishing word document can include a text "An error has occurred. Could not load this word document because

of disabled content. Try to enable content to display the file". Some people might enable macros if they are for example in a hurry or tired and everyone really could accidentally fall into a phishing scam. When the macros are enabled, it might create a trojan-dropper and here starts the second steps called infection where the dropper downloads an executable from a remote server for the installation of the ransomware. Then starts the third step called staging where the ransomware starts to gain persistence in the infected system by making itself to run even if the system is booted or in recovery mode, modifying registry keys, looking for important information from configuration settings, modifying shortcuts and such things. In this step the ransomware also tries to find shadow copies and then deletes them so that the infected machine can't recover the system to the recent saved snapshot. (Graziano, D. 2015)

At this point the ransomware has gained its persistence in the system even when shutdown so now the ransomware starts the scanning phase with local scanning which completes in seconds then continuing with network scanning and cloud scanning. When all the files have been located, the encryption step starts, and the ransomware begins to encrypt all the detected files. Newer versions of ransomware use a new technique called hybrid encryption where the ransomware uses both symmetric and asymmetric encryption. This technique gets rid of the downsides of both symmetric and asymmetric encryption algorithms. In the last step the ransomware displays a note to the target system by changing the desktop background to the ransom note with instructions. Sometimes there is a timer in the note and when the timer ends the attacker then won't give the decryption keys, the price of the payment goes up or they leak all the information online. (Graziano, D. 2015)

5 Testing and analyzing a ransomware in a safe environment

Malware analysis is the process of testing, studying, and understanding the behavior of a malware. Malware analysts use different tools and techniques to understand how a malware work, what type it is, how to detect it, and how to eliminate it. This can be done with two different techniques which are static analysis and dynamic analysis. The difference between these two are that in static analysis, the malware is not executed and in dynamic analysis it is executed in a safe environment. Both techniques are good in their own way and to fully understand the malware, these both are used to gather as much information as possible (Monappa, K. 2018)

Static analysis provides information about a malware without running it. There are different ways to do this, and each way provides small bits of information about the malware. You can scan the malware sample in multiple antivirus programs to find out if it has already been identified. You can try to fingerprint the malware with hashing. This means that the hash is run through a hashing program which gives the malware unique hash value. This hash can

then be shared with other malware researchers to find further information about it, or you can search the hash in VirusTotal which is a service that has multiple antivirus scanning engines. Sometimes hex editor or other types of tools such as 'file' utility or CFF Explorer are used to identify the file type. Hex editor can examine each byte of the malware sample to find the file signature. With the file signature you can easily identify what type of a file the malware is, and which operating system is the malware trying to target. Other ways to perform static analysis can be by finding strings (this can give you clues on how the malware functions), Inspecting PE headers (contains metadata about the malware), using tools like import hash or fuzzy hashing to classify in which malware family the malware belongs to. (Monnappa, K. 2018) (Sikorski, M & Honig, A. 2012)

In dynamic analysis it is important to note that you are running an actual malware in the analysis process. This means that you must have a proper, secure environment for the testing. If the testing environment is not secure, then the malware could potentially spread on the network and infect your system and others as well. Dynamic analysis also known as behavioral analysis is the process of executing a malware sample and examining how the malware behaves. This can be different types of monitoring such as process monitoring, file system monitoring, registry monitoring, or network monitoring. Common tools used for these can be Process Monitor, Process Explorer, Regshot, and Wireshark. Process monitor is a Windows monitoring tool used for monitoring file system, registry, process, and thread activity. Process Explorer is basically a more powerful task manager. Regshot is a registry comparison tool, used for taking a snapshot before and after executing a malware and comparing the differences. Wireshark is one of the most known tools for network packet analyzing. With this you can capture network traffic when running the malware to understand the communication used by the malware. (Monnappa, K. 2018) (Sikorski, M & Honig, A. 2012)

5.1 Testing environment

Safe environment is the key for malware testing. When an environment doesn't have secure in mind it can have severe consequences. Malware analysis environments vary a lot, and they can be very complex depending on the needs of testing. This part covers how to set up a simple Malware analysis environment where a malicious software can be safely detonated. Some aspects are not covered such as doing network analysis, since it needs a bit complex lab where there are communicating virtual machines to log network traffic safely. Malware can be analyzed using physical machines on air-gapped networks, but this part covers the easier and more common way which is by using virtual machines.

Virtual machines (VMs) are basically a virtualized computer, in another words a computer inside a computer. On a virtual machine, a guest operating system (OS) is installed within the

host OS. Here the OS running in the VM is isolated from the host OS, so that the executed malware cannot do any damage to the host OS.

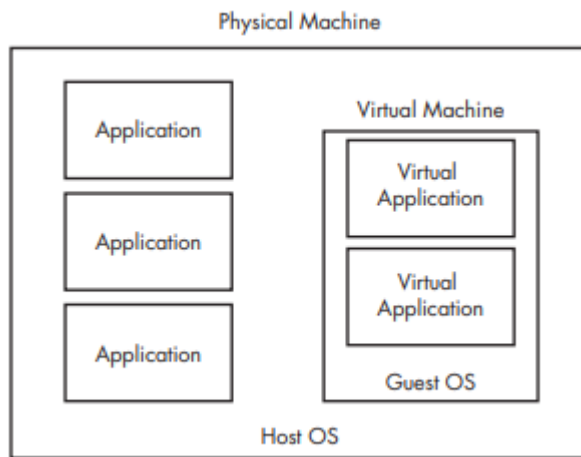


Figure 6: Illustration of a virtual machine (Sikorski, M & Honig, A. 2012)

If the malware does damage to the VM, then you can just reinstall the OS in it or use snapshots to return to a clean state. Snapshots are a way to return to a clean state just like a backup. When you take a snapshot of a clean state of an environment, you can execute a malware knowing that you can return to the earlier snapshot if things don't go as planned. (Sikorski, M & Honig, A. 2012)

To use a virtual machine, a hypervisor must be first installed. Hypervisor is a virtualization software that can create and run VMs. Typical free hypervisors are Oracle VM VirtualBox and VMware Workstation Player. VirtualBox will be used in this testing environment. VirtualBox can be download from their website by simply choosing the suitable installation for your host OS. (VirtualBox, 2023)

The virtual machine used in this case will be REMnux. It is a open source distribution used in reverse-engineering and malware analysis. It provides a collection of pre-installed tools, so you don't have to install them yourself. Easiest way to install REMnux is to download it in OVA (Open Virtual Appliance) format and import it into your hypervisor. VirtualBox has its own specific OVA file for this which can be seen on picture x. When using other hypervisors such as VMware, the general OVA must be chosen. (REMnux, 2023)

Step 1: Download the Virtual Appliance File

The REMnux virtual appliance is approximately 5 GB. It comes as an industry-standard OVA file, which you can import into your virtualization software. It's based on Ubuntu 20.04 (Focal).

Decide which OVA file to download. Unless you're using Oracle VM VirtualBox, get the general OVA file. If you're using VirtualBox, get the VirtualBox version. Download your preferred OVA file:

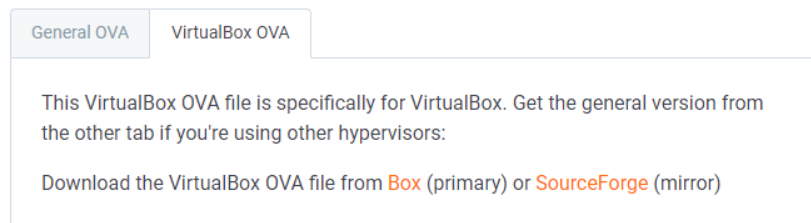


Figure 7: Downloading the virtual appliance file (REMnux 2023)

After the download, it is recommended to verify that the file's hash value matches the expected value. The right value can be found in the step 2 of the REMnux installation manual when downloading the file. When the hash value corresponds to the correct one, then the OVA file can be imported to VirtualBox. To do this, open VirtualBox, click on file in the upper left corner and select 'Import Appliance...', then the Import Virtual Appliance wizard pops up and the REMnux OVA file should be selected to be imported. On the next page, resource adjustments can be made but usually the default settings are enough. These can be later customized if any changes are needed to be made afterwards. After this the importing starts and when finished, it is ready to be used in the home screen. (REMnux. 2023) Few things should be remembered when analyzing malware in the environment. Keeping virtualization software up to date is good practice, so that it limits the chances of the malware to exploit vulnerabilities in the virtualization software or escape from the virtual environment. Not connecting any removable devices such as USB drives or a charger connecting to a phone. Using a different host OS than the one in the malware testing VM, meaning that when analyzing a Windows malware, an OS such as Linux or macOS should be used (even if the malware escapes the VM it can't infect the host machine). Taking snapshots regularly and not storing any sensitive information in the virtual machine. Last key thing to remember while analyzing a malware is to use 'not attached' network adapter setting, host-only network or simulated services in the environment, this way the malware is contained within the VM and isn't connected to the internet. (Monappa, K. 2018)

5.2 Malware demo

Here I will be using a Windows 10 computer with VirtualBox version 7.0.10 installed and a REMnux VM. First thing to do in this demo is to take a clean snapshot of the machine. This is done by clicking on the three dots in the VM on the home screen, then choosing the snapshots setting. On the snapshots screen, clicking to the 'Take' button takes a snapshot of the current status of the VM. A name can be chosen for the snapshot and a description can be set.

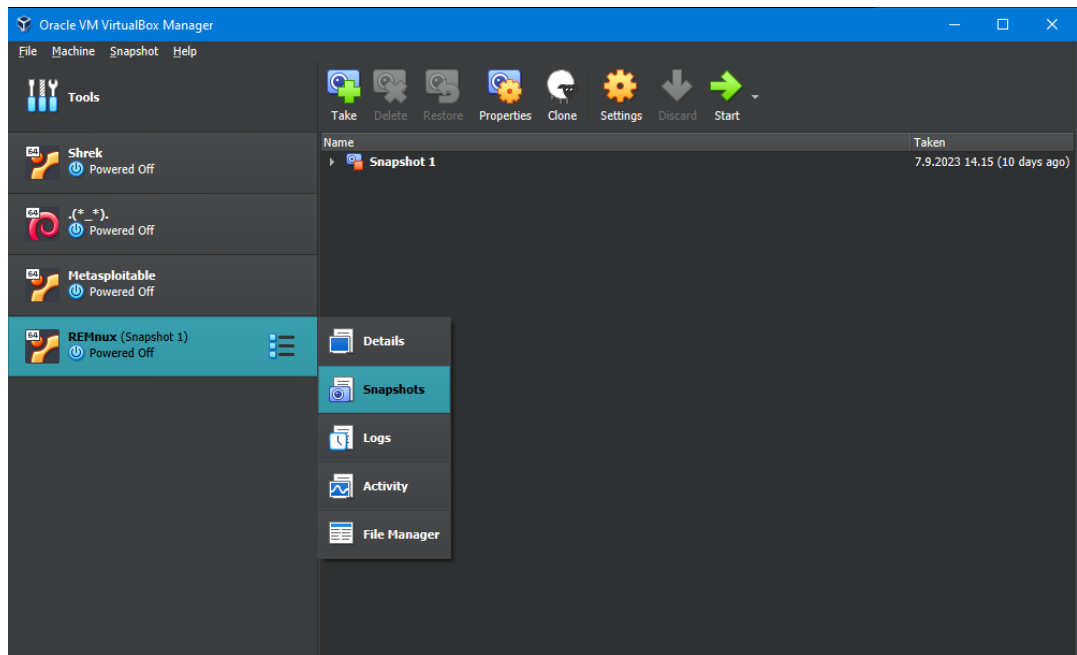


Figure 8: Taking a snapshot of a clean virtual machine in VirtualBox

On the default settings page, clicking on the network lets you modify the network settings. At first the network should be on network address translation displayed as 'NAT', or similar where the internet connection is working properly for downloading the malware safely.

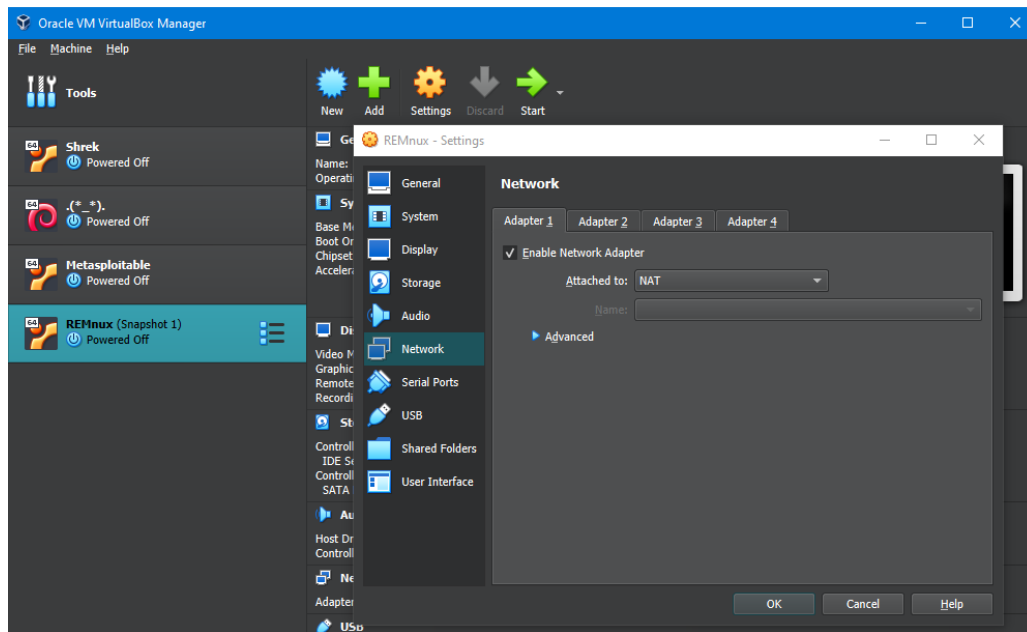


Figure 9: Network settings page in VirtualBox

Malware samples can be downloaded from different sources such as MalwareBazaar, theZoo, VirusShare, etc. Here we are going to download two similar Linux samples from MalwareBazaar which is a project of Abuse.ch where IT-security researchers can share malware samples for free and further analyze them. (Abuse.ch, 2020) These two samples look to be almost identical to each other. They both have a 'BlackBasta' signature which is a well-known ransomware group that offers ransomware as a service. Black Basta group became active in April 2022, and they have an approach where they usually choose their targets and not spam phishing attacks and hope for the best. They usually use a double-extortion tactic where they not only ask for a ransom payment for recovering files but also ransom payment for not leaking any data publicly. (Trend Micro Research, 2022) Here the file type is 'elf' (Executable and Linkable Format) indicating it to be a Linux based sample. The SHA 256 is different on both, so we know that they aren't identical. Ransomware have encryptors which encrypts all the data and decryptors which is used to recover the data, so here the other file is probably encryptor and other decryptor. MalwareBazaar has 36 Black Basta malware samples when searching by signature and 37 when searching for Black Basta with tags. 5 of the Black Basta samples are Linux based and the rest are Windows based. To download these as zip file, simply click on the 'download sample'. The zip files are password protected automatically to avoid accidental detonations.

Field	Value
SHA256 hash:	96339a7e87f942f4d70418a56ea38270bc42db937fa8517a8659301ac78
SHA3-384 hash:	e53bbc664cb90c029c50c8819e8c0142f6bd79f98b3d618f942f4d70418a56ea38270bc42db937fa8517a8659301ac78
SHA1 hash:	8ccac360e2ca37b2fa9f5fa81b22114fb8936120
MD5 hash:	7688c1b7a1124c1cd9413f4b535b2f44
humanhash:	nine-yellow-winter-tennis
File name:	96339a7e87f942f4d70418a56ea38270bc42db937fa8517a8659301ac78
Download:	download sample
Signature @	BlackBasta Alert
File size:	209170 bytes
First seen:	2022-06-09 12:35:15 UTC
Last seen:	2023-07-25 23:35:03 UTC
File type:	elf
MIME type:	application/x-executable
ssdeep @	6144:OUjqtclKpiqKLiCZM5cUq29shXs6u7ulx97Z52Gd:ftq4KoVkcM9oV
TLSH @	T12C144B47F2D61CFFC6CADE304797A1266D62B82193211D3F2584C6301A9BF681F1EB66
telhash @	t115c0120498680b4c4d635620dd9d1b5150436d28b5ae3b016ff8d995411c64f424ae5f

Figure 10: Downloading first malware sample from MalwareBazaar (MalwareBazaar 2023)

Field	Value
SHA256 hash:	0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef
SHA3-384 hash:	39e90510e908e2e6ab8fc3186d5992ba7c2e2d4c386ee86a2f8d115668f4f81f1b48039e88c92b379241ab41d7f92153
SHA1 hash:	b363e038a6d6326e07a02e7ff99d82852f8ec2d2
MD5 hash:	32f17040ddaf3477008d844c8eb98410
humanhash:	nebraska-four-berlin-equal
File name:	0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef
Download:	download sample
Signature @	BlackBasta Alert
File size:	222120 bytes
First seen:	2022-11-15 11:02:41 UTC
Last seen:	2022-11-15 13:01:45 UTC
File type:	elf
MIME type:	application/x-executable
ssdeep @	6144:qHxwGbi2dn97rh3akMS2vEUrhsQpN1W4XaOZ/6gpZF7:YG+y97KvDW2N
TLSH @	T18B245C4BF7961CFFC5CADE708687A1256D22B83192211D3F2544CB30199AF6C2F1EB66
telhash @	t13cc0120588691b4849635630d99d175151436c66b5ae3b113fbc9d5411c64b0246e5f

Figure 11: Downloading second malware sample from MalwareBazaar (MalwareBazaar 2023)

Now that the samples have been downloaded, the network settings should be changed to 'not attached' and turn off the 'cable connected' option. After this another snapshot is recommended to take to avoid any accidents.

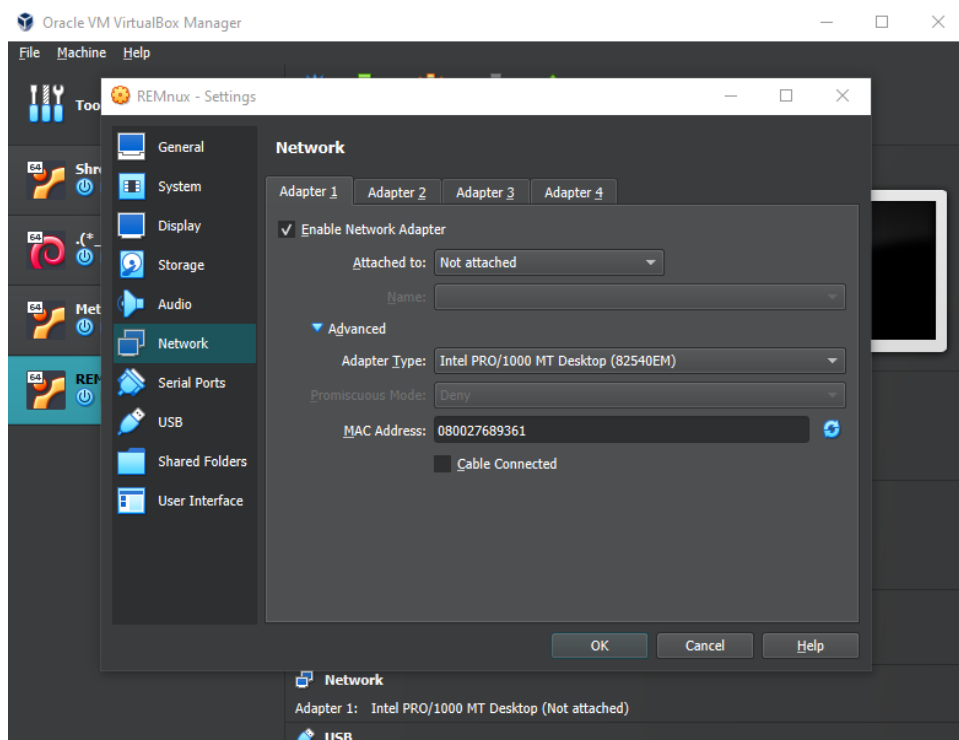


Figure 12: Customizing network settings of REMnux

To get information about the file straight away, we can do a quick static analysis with VirusTotal. VirusTotal is an online service that uses multiple antivirus (AV) scanners and block-lists to analyze malicious files, URLs, hashes, IP addresses and other types of malicious threats and shows which vendor's AVs find the samples malicious by flagging the sample. (VirusTotal, 2023) If we copy and paste the hashes of the files downloaded to VirusTotal, it gives you some indicators whether the files are malicious or not. I checked all the Linux based Black Basta files that were in Malware bazaar and two of those had been flagged by 9 vendors in VirusTotal, one file was flagged by 34 vendors and two which I'm going to analyze in this thesis were flagged by 39 vendors. The file sizes of the two files that were flagged by 9 vendors are both around 95 KB and the files that were flagged by 39 vendors are little bit over 200 KB. From this I would assume that the files flagged by 39 vendors have both the encryptor and the correct decryptor because they have similarities. Although the samples (Figure 13 & Figure 14) analyzed by VirusTotal might look identical and share many similarities, they should be different since the hash values differ from one another, the file sizes are different with a 12KB difference, lot of the same vendors have flagged both of the files but not all of them, and the file names named by vendors differ often indicating that they are from the same families but still differing from each other.

URL, IP address, domain, or file hash

39 / 61

39 security vendors and no sandboxes flagged this file as malicious

96339a7e87fceb6ced247feb9b4cb7c05b83ca315976a952215bad726b8e5be

Size: 204.27 KB | Last Analysis Date: 6 days ago

Community Score

DETECTION DETAILS RELATIONS BEHAVIOR COMMUNITY 13

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label: ransomware.blackbasta.kbbwbf

Threat categories: ransomware trojan

Family labels: blackbasta kbbwbf yxcrt

Security vendors' analysis

Vendor	Detection	Engine	Signature
AhnLab-V3	Ransomware/Linux.BlackBasta.209170	ALYac	Trojan.Ransom.Linux.Gen
Antiy-AVL	Trojan(Ransom)Win32.BlackBasta.gen	Arcabit	Trojan.Ransom.BlackBasta.A
Avast	ELF-Filecoder-DZ [Trj]	AVG	ELF-Filecoder-DZ [Trj]
Avira (no cloud)	LINUX/Filecoder.kbbwbf	BitDefender	Trojan.Ransom.BlackBasta.A
Cynet	Malicious (score: 99)	Cyren	E64/DCBlackbat_JICC

Figure 13: VirusTotal's analysis of first malware sample (VirusTotal 2023)

URL, IP address, domain, or file hash

39 / 62

39 security vendors and no sandboxes flagged this file as malicious

0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef

Size: 216.91 KB | Last Analysis Date: 24 days ago

Community Score

DETECTION DETAILS RELATIONS BEHAVIOR COMMUNITY 12

Join the VT Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label: ransomware.blackbasta.lyxcfj

Threat categories: ransomware trojan

Family labels: blackbasta yxcfj jgroj

Security vendors' analysis

Vendor	Detection	Engine	Signature
AhnLab-V3	Ransomware/Linux.BlackBasta.222120	ALYac	Trojan.Ransom.Linux.Gen
Antiy-AVL	Trojan(Ransom)Win32.BlackBasta.gen	Arcabit	Trojan.Linux.Ransom.Q
Avast	ELF-Filecoder-DZ [Trj]	AVG	ELF-Filecoder-DZ [Trj]
Avira (no cloud)	LINUX/Filecoder.jgroj	BitDefender	Trojan.Linux.Ransom.Q
Cynet	Malicious (score: 99)	Cyren	E64/DCBlackbat.KHEC

Figure 14: VirusTotal's analysis of second malware sample (VirusTotal 2023)

Before detonating the ransomware, some test files should be created to see if they get encrypted or not. I have three .jpg files and two .txt files in my Documents directory.

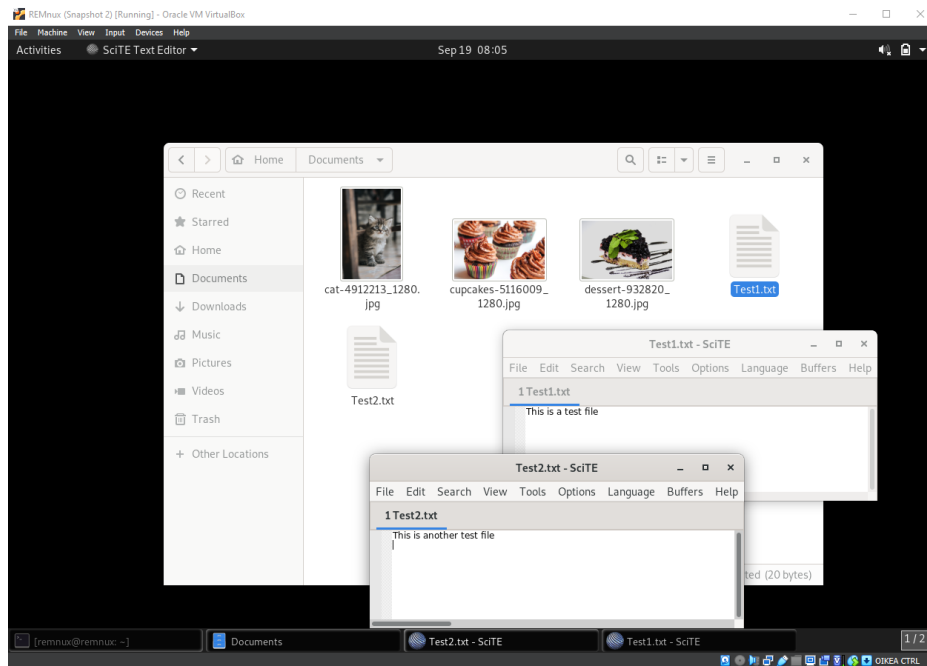


Figure 15: Test files in Documents folder

In my downloads directory are the ransomware extracted as shown in the picture (x). Here the files don't have execution right automatically and to detonate it must be given.

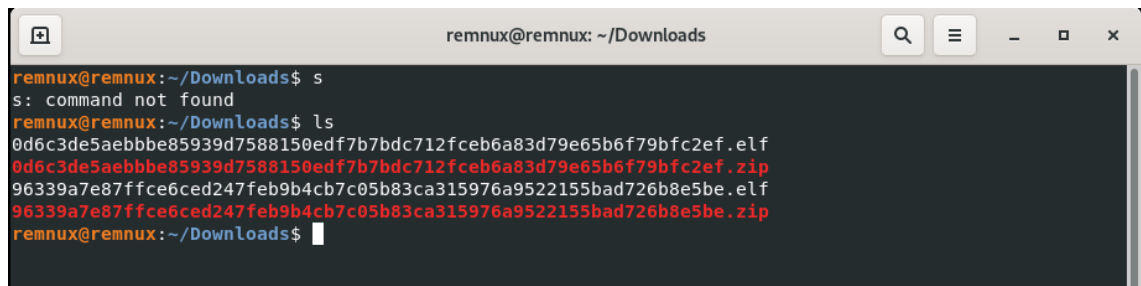
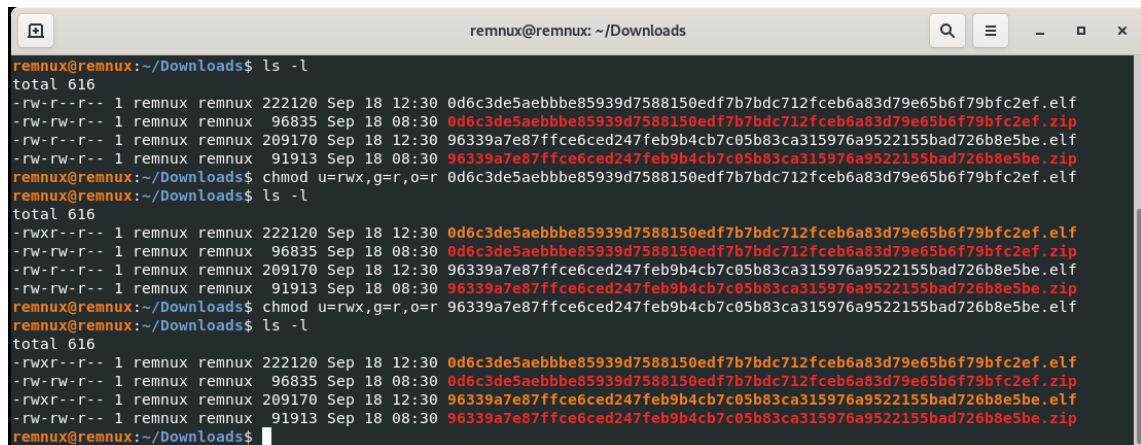


Figure 16: Ransomware shown in Downloads folder

Here I used the `chmod` command to give the necessary rights and I only gave the user the execution right with the following command: `chmod u=rwx,g=r,o=r <file>`. Here the letters `u`, `g`, and `o` stand for user, group, and other, and `r`, `w`, and `x` stand for read, write, and execute. In the picture `x` you can see that after the files have the execution right, it changes colour.



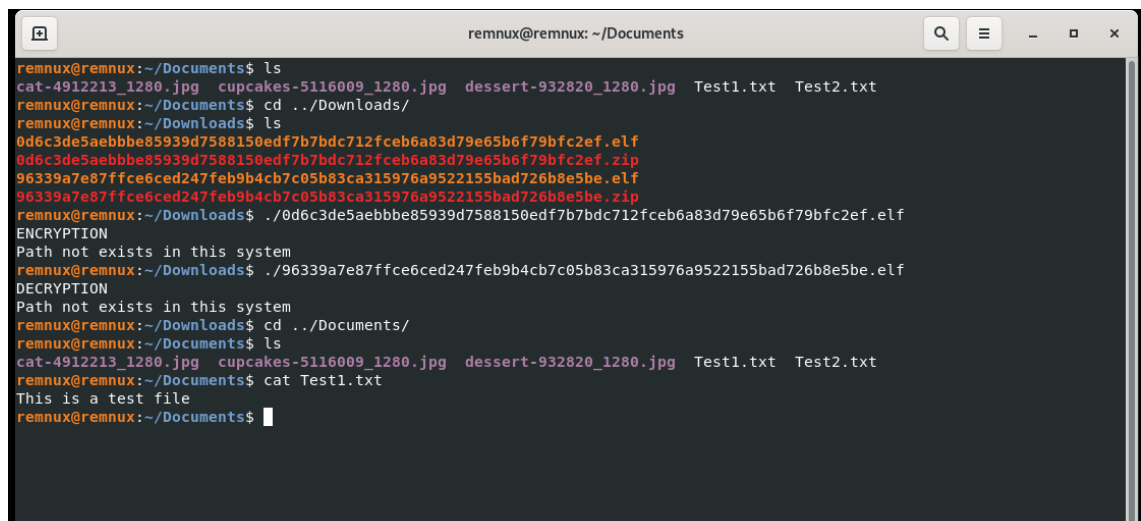
```

remnux@remnux: ~/Downloads
remnux@remnux:~/Downloads$ ls -l
total 616
-rw-r--r-- 1 remnux remnux 222120 Sep 18 12:30 0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
-rw-rw-r-- 1 remnux remnux 96835 Sep 18 08:30 0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.zip
-rw-r--r-- 1 remnux remnux 209170 Sep 18 12:30 96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.elf
-rw-rw-r-- 1 remnux remnux 91913 Sep 18 08:30 96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.zip
remnux@remnux:~/Downloads$ chmod u=rwx,g=r,o=r 0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
remnux@remnux:~/Downloads$ ls -l
total 616
-rwxr--r-- 1 remnux remnux 222120 Sep 18 12:30 0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
-rw-rw-r-- 1 remnux remnux 96835 Sep 18 08:30 0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.zip
-rw-r--r-- 1 remnux remnux 209170 Sep 18 12:30 96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.elf
-rw-rw-r-- 1 remnux remnux 91913 Sep 18 08:30 96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.zip
remnux@remnux:~/Downloads$ chmod u=rwx,g=r,o=r 96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.elf
remnux@remnux:~/Downloads$ ls -l
total 616
-rwxr--r-- 1 remnux remnux 222120 Sep 18 12:30 0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
-rw-rw-r-- 1 remnux remnux 96835 Sep 18 08:30 0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.zip
-rwxr--r-- 1 remnux remnux 209170 Sep 18 12:30 96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.elf
-rw-rw-r-- 1 remnux remnux 91913 Sep 18 08:30 96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.zip
remnux@remnux:~/Downloads$

```

Figure 17: Giving executable rights to the ransomware files

To execute a file in the command line, simply put “./” in front of the file name you want to execute. I executed the files but for some reason they didn’t work as they should have. A message “ENCRYPTION/DECRYPTION Path not exists in this system” was printed. Now at least I can assume that the other file is encryptor and the other is the decryptor.



```

remnux@remnux: ~/Documents
remnux@remnux:~/Documents$ ls
cat-4912213_1280.jpg  cupcakes-5116009_1280.jpg  dessert-932820_1280.jpg  Test1.txt  Test2.txt
remnux@remnux:~/Documents$ cd ../Downloads/
remnux@remnux:~/Downloads$ ls
0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.zip
96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.elf
96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.zip
remnux@remnux:~/Downloads$ ./0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
ENCRYPTION
Path not exists in this system
remnux@remnux:~/Downloads$ ./96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.elf
DECRYPTION
Path not exists in this system
remnux@remnux:~/Downloads$ cd ../Documents/
remnux@remnux:~/Documents$ ls
cat-4912213_1280.jpg  cupcakes-5116009_1280.jpg  dessert-932820_1280.jpg  Test1.txt  Test2.txt
remnux@remnux:~/Documents$ cat Test1.txt
This is a test file
remnux@remnux:~/Documents$

```

Figure 18: Execution of ransomware

To find out why the ransomware are not working properly I’m going to do some malware analyzing. For this I will use a tool named ‘Detect It Easy’ that is pre-installed in REMnux. Detect It Easy is an easy-to-use tool for doing static and dynamic analysis. It has a simple interface where the file only has to be selected and lots of information comes up.

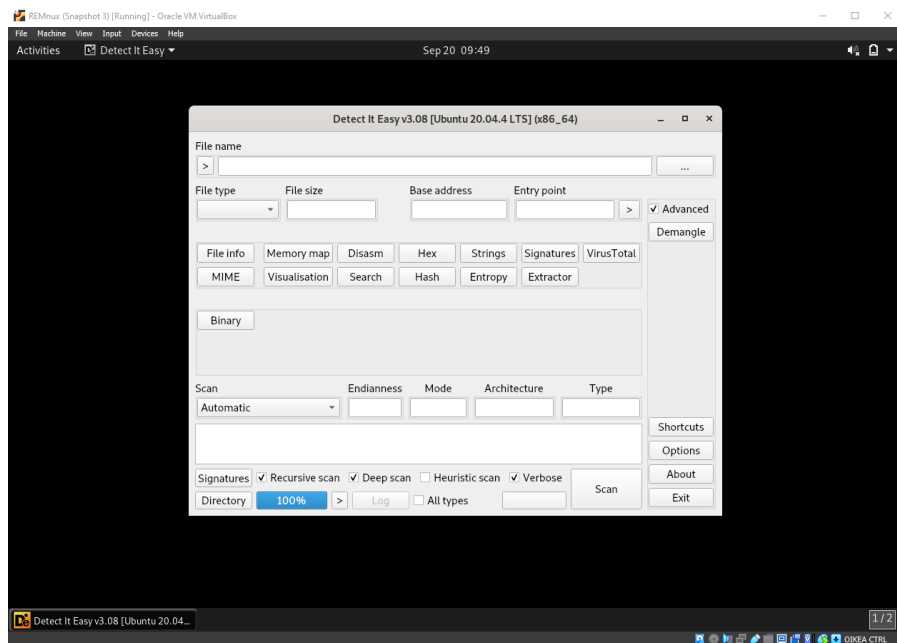


Figure 19: Displaying the default screen of Detect It Easy

When either of the files are selected the tools tells what kind of a file it is, operating system information and the executables are written in C/C++.

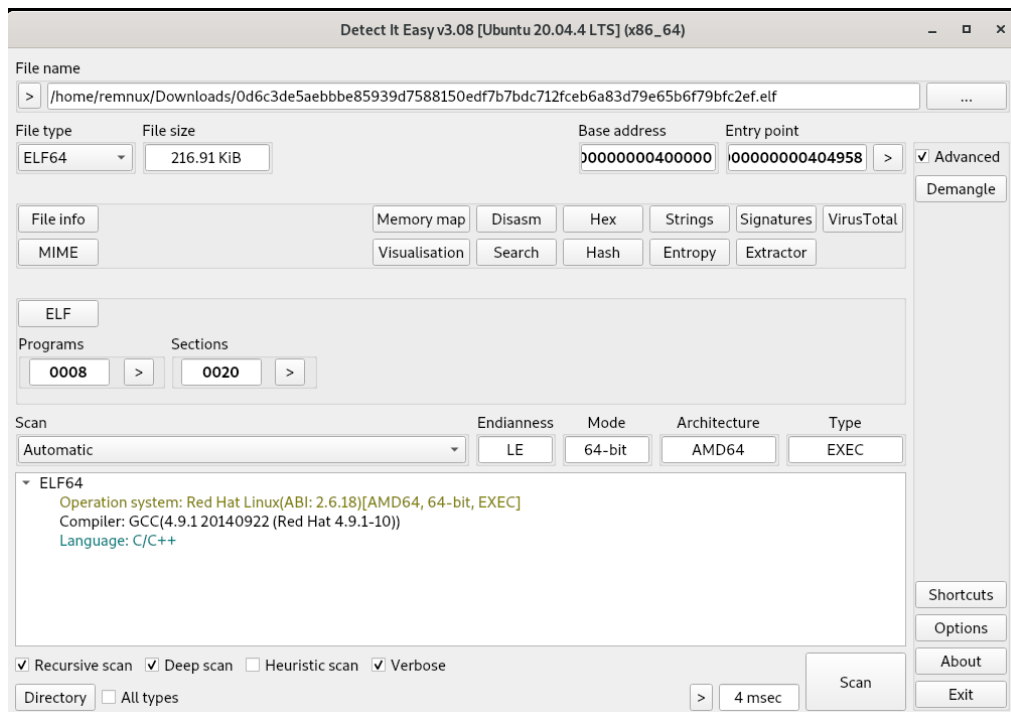


Figure 20: Detect It Easy revealing information about the ransomware

It is possible to analyze the hex values of the files with this tool. This gives some basic information that can be retrieved in static analysis.

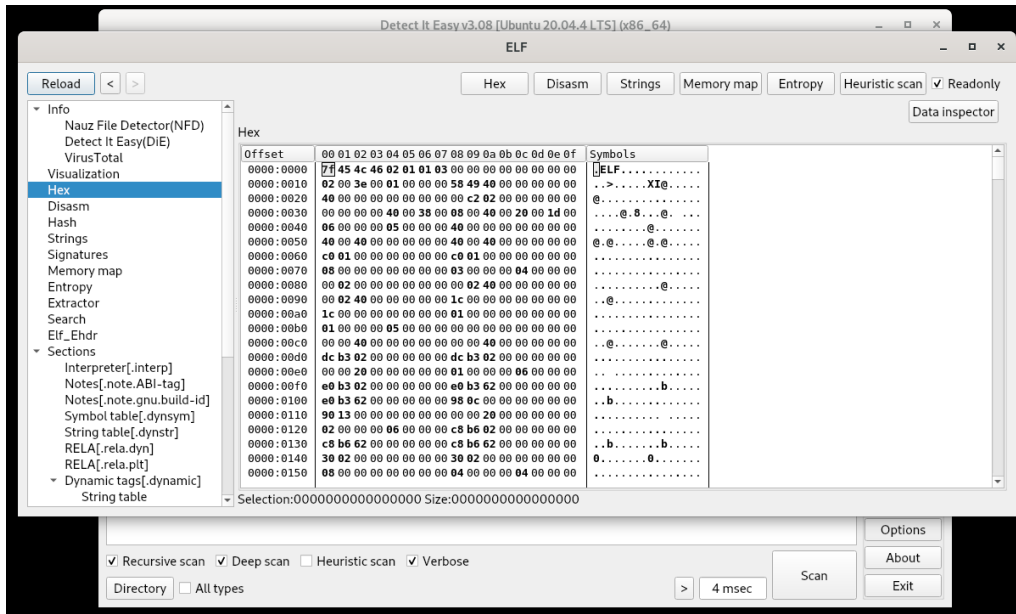


Figure 21: Detect It Easy displaying hex values

To find and understand where the issues are I can try to find clues from the strings section of the tool. With this I can maybe get an understanding how the program functions.

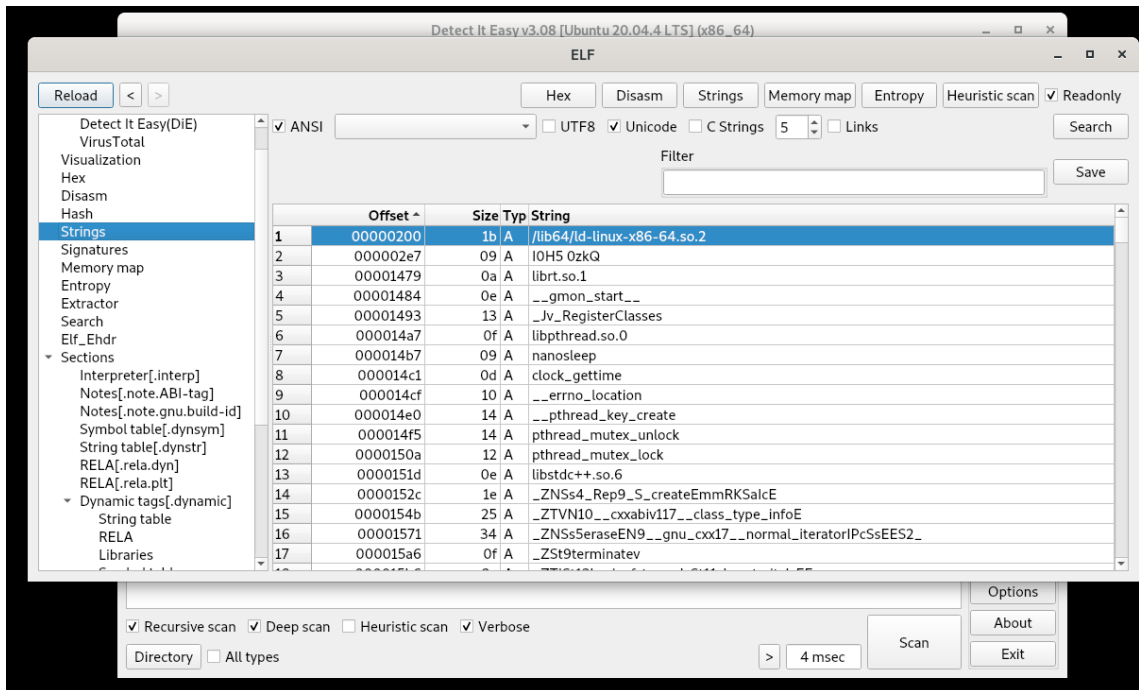


Figure 22: Analyzing strings with Detect It Easy

This section gives around 1400 different strings that are associated with each file. When looking quickly through all of the strings it seems that some of the executable is not working properly, maybe missing something and stopping before it gets to encrypt all the files. Lots of

error indicating strings can be found. The strings section has a note which seems to be the ransom note that should be added after the encryption. Even if the executable is malfunctioning, the tool lets you see the ransom note. The ransom note says that my data are stolen and encrypted, and it will be published if a ransom payment isn't made. It gives a link to a TOR site and give a specific login id.

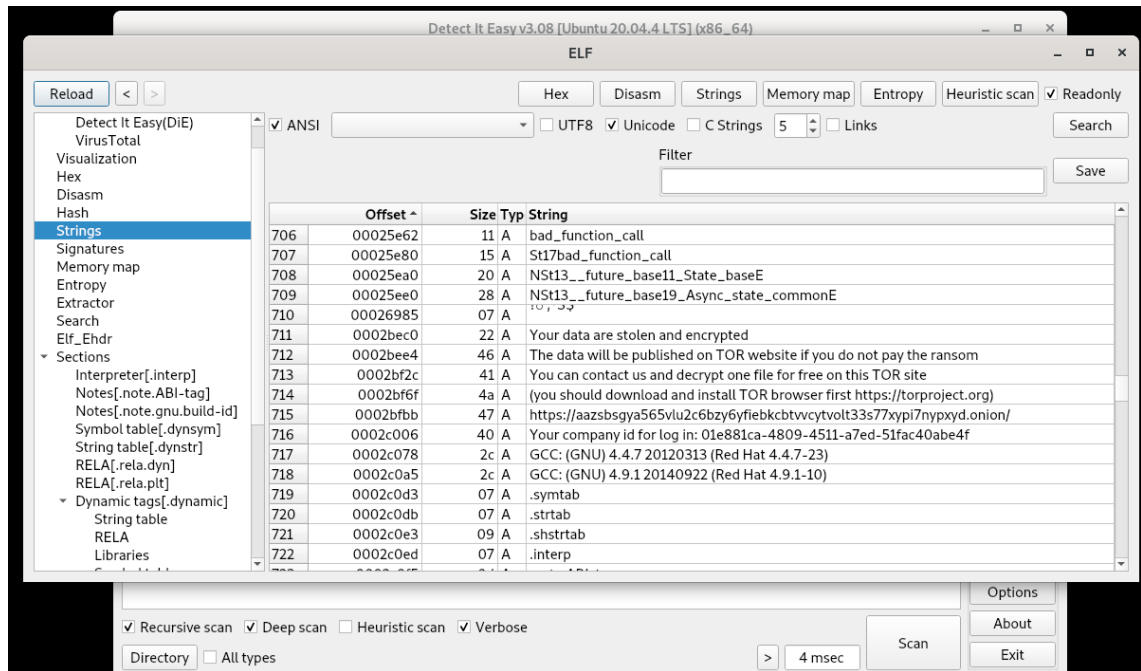


Figure 23: Strings reveal a ransom note in Detect It Easy

After looking for the strings section quite some time I found an indicator where the file might malfunction. In row 658 is listed a path /vmfs/volumes followed by “force path”, and 5 rows after that is listed an output “Path not exists in this system”. I noticed that this kind of path doesn't exist in my virtual machine. I tried to find this path, but nothing appeared. Both of the ransomware files seem to have this same problem.

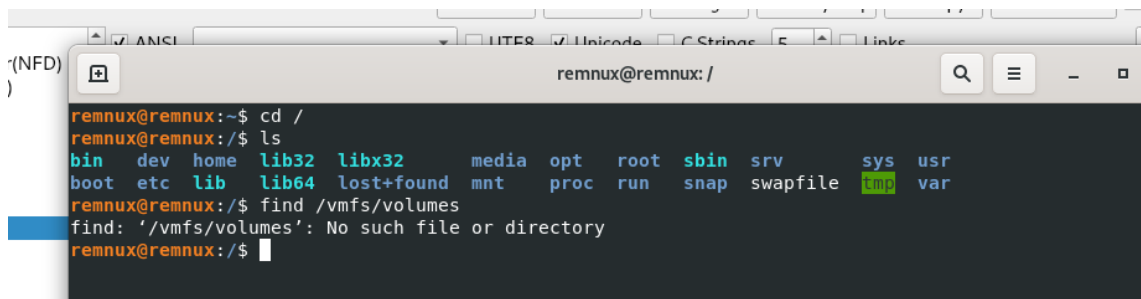
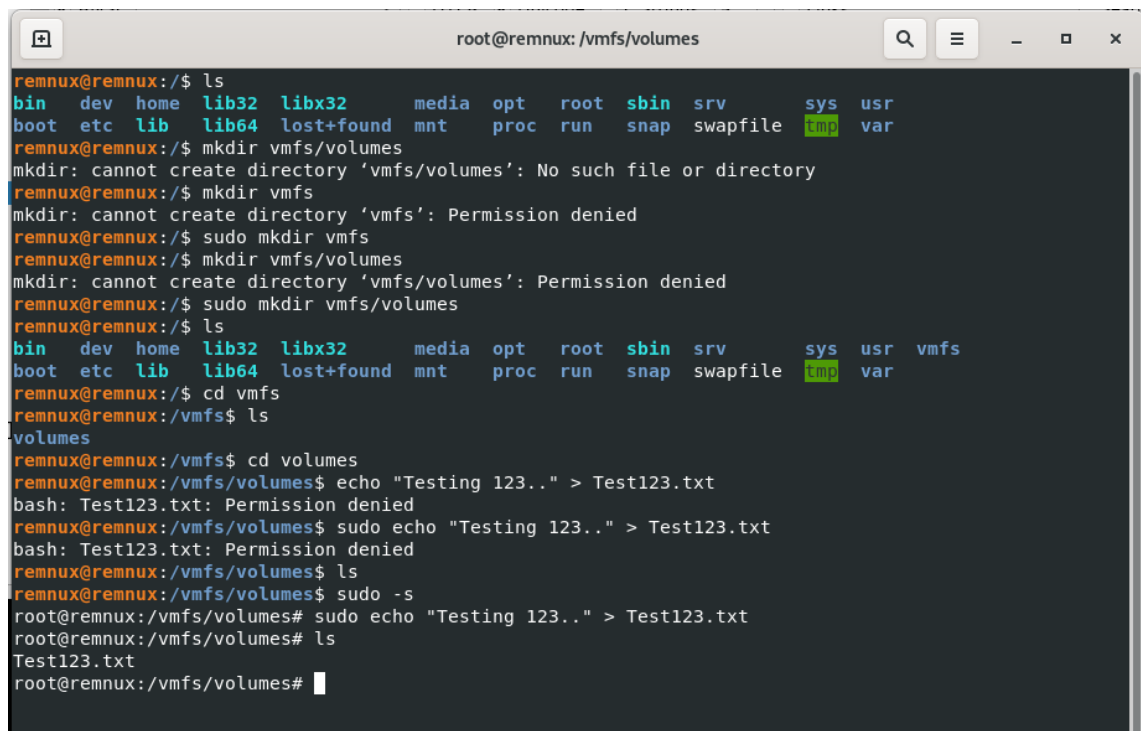


Figure 24: Checking if this virtual machine has /vmfs/volumes path

My guess is that this executable must scan this directory but since I don't have it in my system, it doesn't run correctly and skips rest of the instructions it should do. Vmfs stands for Virtual Machine File System, and it stores data about virtual machines. If a host has virtual machines, then this executable would probably encrypt data about them too. Next thing I will do, is to test, if creating a directory named /vmfs/volumes lets the executable run correctly. I added a vmfs directory to the root and volumes directory to vmfs. I wanted to create a test file to the vmfs/volumes directory to see what will happen to that file if the ransomware executable works.



```

root@remnux: /vmfs/volumes
remnux@remnux:/$ ls
bin  dev  home  lib32  libx32  media  opt  root  sbin  srv  sys  usr
boot  etc  lib  lib64  lost+found  mnt  proc  run  snap  swapfile  tmp  var
remnux@remnux:/$ mkdir vmfs/volumes
mkdir: cannot create directory 'vmfs/volumes': No such file or directory
remnux@remnux:/$ mkdir vmfs
mkdir: cannot create directory 'vmfs': Permission denied
remnux@remnux:/$ sudo mkdir vmfs
remnux@remnux:/$ mkdir vmfs/volumes
mkdir: cannot create directory 'vmfs/volumes': Permission denied
remnux@remnux:/$ sudo mkdir vmfs/volumes
remnux@remnux:/$ ls
bin  dev  home  lib32  libx32  media  opt  root  sbin  srv  sys  usr  vmfs
boot  etc  lib  lib64  lost+found  mnt  proc  run  snap  swapfile  tmp  var
remnux@remnux:/$ cd vmfs
remnux@remnux:/vmfs$ ls
volumes
remnux@remnux:/vmfs$ cd volumes
remnux@remnux:/vmfs/volumes$ echo "Testing 123.." > Test123.txt
bash: Test123.txt: Permission denied
remnux@remnux:/vmfs/volumes$ sudo echo "Testing 123.." > Test123.txt
bash: Test123.txt: Permission denied
remnux@remnux:/vmfs/volumes$ ls
remnux@remnux:/vmfs/volumes$ sudo -s
root@remnux:/vmfs/volumes# sudo echo "Testing 123.." > Test123.txt
root@remnux:/vmfs/volumes# ls
Test123.txt
root@remnux:/vmfs/volumes#

```

Figure 25: Creating /vmfs/volumes path and adding a test file in volumes folder

After running the ransomware encryptor again an output "Text file busy" was outputted and this happened because I still had it open on Detect It Easy tool. I closed the tool that was dynamically analyzing the files and tested to run the ransomware again. This time it displayed the encryption and how fast it did the encryption.

```

root@remnux: /home/remnux/Downloads
root@remnux: /home/remnux/Downloads# ls
0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.zip
96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.elf
96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.zip
root@remnux: /home/remnux/Downloads# ./0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
bash: ./0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf: Text file busy
root@remnux: /home/remnux/Downloads# ls
0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.zip
96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.elf
96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.zip
root@remnux: /home/remnux/Downloads# cd ../Do
Documents/ Downloads/
root@remnux: /home/remnux/Downloads# cd ../Do
Documents/ Downloads/
root@remnux: /home/remnux/Downloads# cd ../Documents/
root@remnux: /home/remnux/Documents# ls
cat-4912213_1280.jpg  cupcakes-5116009_1280.jpg  dessert-932820_1280.jpg  Test1.txt  Test2.txt
root@remnux: /home/remnux/Documents# cat Test1.txt
This is a test file
root@remnux: /home/remnux/Documents# cd ../Downloads/
root@remnux: /home/remnux/Downloads# ./0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
ENCRYPTION
Done time: 0.0020 seconds, encrypted: 0.0000 gbroot@remnux: /home/remnux/Downloads#

```

Figure 26: Executing the encryptor

In my documents folder nothing was encrypted. I opened the test pictures and they worked normally. I also opened the .txt files created in documents folder, and they weren't either encrypted. My desktop also didn't change, and I didn't seem to have any ransom note that was seen in the dynamic analysis.

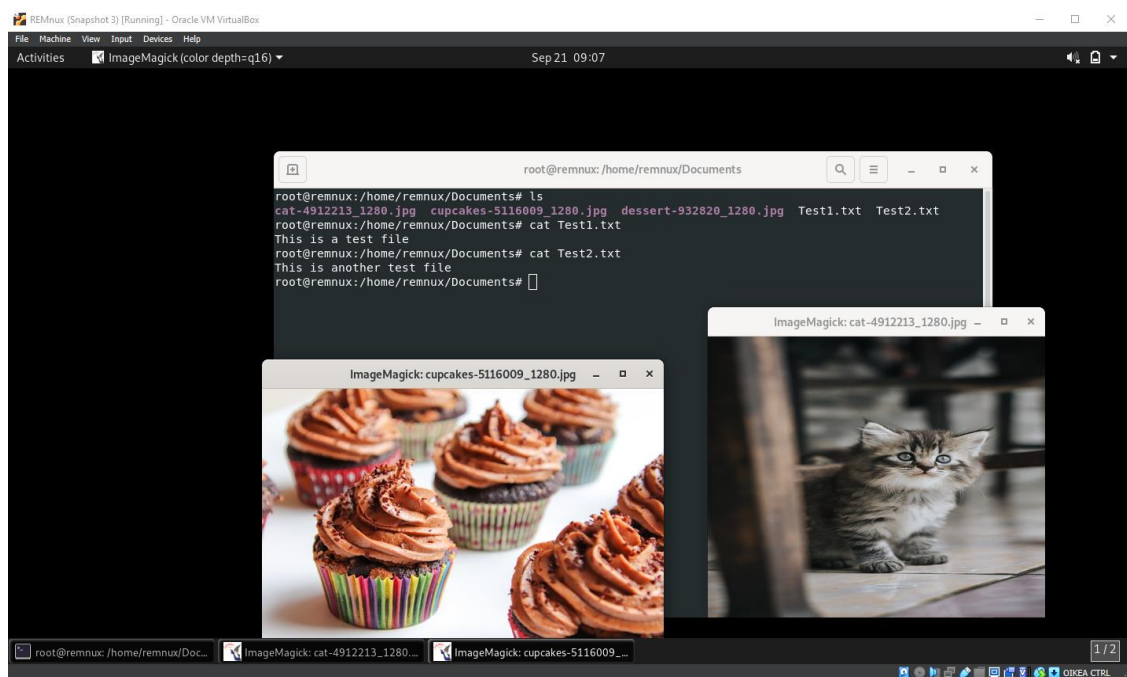
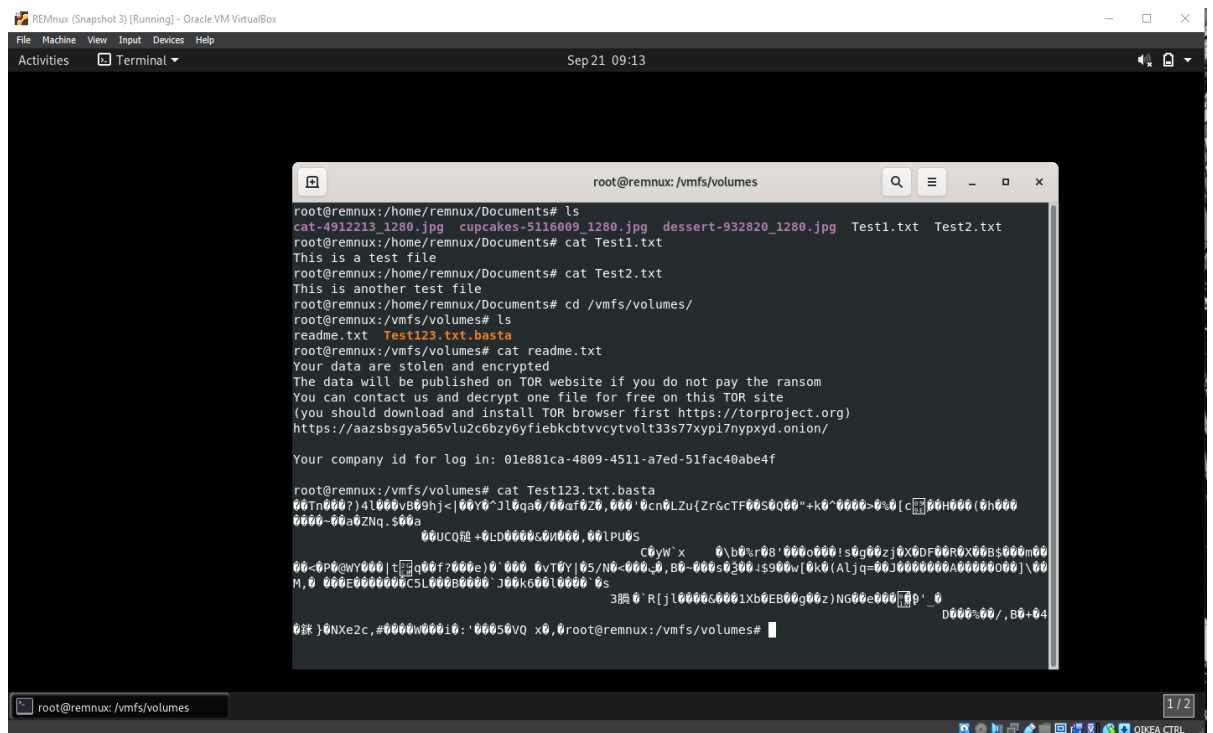


Figure 27: Displaying test files in Documents folder

I then checked the `vmfs/volumes` directory and there seemed to be some changes. A `readme.txt` was added there which is the ransom note and the `.txt` file created for testing was changed to `.basta` file with execution rights. This means that the encryption might have only happened in the `vmfs/volumes` directory and that's why the encryptor didn't work in the first place. There weren't any other signs of targeted folders, so this might indicate that the target is VMware ESXi hosts. VMware ESXi server are virtual machines, and their data is usually stored in the `vmfs/volumes` directory. This ransomware is a simple variant of other Black Basta ransomware since this specific one only encrypts a small part and usually Black Basta ransomware are far more severe.



```

root@remnux: /vmfs/volumes
root@remnux:/home/remnux/Documents# ls
cat-4912213_1280.jpg  cupcakes-5116009_1280.jpg  dessert-932820_1280.jpg  Test1.txt  Test2.txt
root@remnux:/home/remnux/Documents# cat Test1.txt
This is a test file
root@remnux:/home/remnux/Documents# cat Test2.txt
This is another test file
root@remnux:/home/remnux/Documents# cd /vmfs/volumes/
root@remnux:/vmfs/volumes# ls
readme.txt  Test123.txt.basta
root@remnux:/vmfs/volumes# cat readme.txt
Your data are stolen and encrypted
The data will be published on TOR website if you do not pay the ransom
You can contact us and decrypt one file for free on this TOR site
(you should download and install TOR browser first https://torproject.org)
https://aazsbsgya565vlu2c6bzygyfielkcbtvvctvolt33s77xypi7nypxyd.onion/

Your company id for log in: 01e881ca-4809-4511-a7ed-51fac40abe4f

root@remnux:/vmfs/volumes# cat Test123.txt.basta
00Tn000?)4l000vB09hj<|00Y0^j10qa0/00af0Z0,000'0cn0LZu{ZrscTF0S000"+k0^0000>0%0[c00H000(0h000
0000-00a0Znq.$00a
00UC000+0L00000S0i000,00lPU0S
C0yW*x 0\b0:r08'000o000!s0q00zj0X0DF0R0X0B$000m00
00-0P0@wY000)t0q00f?000e)0'000 0VT0Y|05/N0-000_0,B0-000s0300:$900w[0k00(ALjq=00J0000000A00000000)\00
M,0 000E0000000CSL000B0000'J00kc00l0000'0s
300 0 R[jl0000c0001Xb0EB00j00z)NG00e000'00' 0
D000:00/,B0+04
000}0NXe2c,#0000v000i0:000S0VQ x0,0root@remnux:/vmfs/volumes#

```

Figure 28: Displaying encrypted test file in `/vmfs/volumes`

To recover the encrypted files, I have to run the decryptor that I downloaded. The decryptor should be the correct one for this specific encryptor since it looked almost exactly the same in the Detect It Easy tool. I run the decryptor file, but this also gave me some problems. The test file that was encrypted didn't change and the bites and hash value also seem to be the exact same. I then tested if I need root privileges for the decryptor to work correctly and it removed the `.basta` extension. The file was still an executable and wasn't fully decrypted. The bits had changed but it seemed to be halfway through the decryption.


```

root@remnux:/vmfs/volumes# ls
readme.txt  Test123.txt.basta
root@remnux:/vmfs/volumes# cd /home/remnux/Downloads/
root@remnux:/home/remnux/Downloads# ls
0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.zip
96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.elf
96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.zip
root@remnux:/home/remnux/Downloads# ./96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.elf
DECRYPTION
Done time: 0.2870 seconds, encrypted: 0.0000 gbroot@remnux:/home/remnux/Downloads# cat /vmfs/volumes
^C
root@remnux:/home/remnux/Downloads# cd /vmfs/volumes/
root@remnux:/vmfs/volumes# ls
readme.txt  Test123.txt
root@remnux:/vmfs/volumes# cat Test123.txt
0P030*py0}
root@remnux:/vmfs/volumes#

```

Figure 29: Decrypting the test file and displaying it

I once again tried to run the decryptor but now it didn't change the file in any way even with the root privileges. I then tested if the encryptor changes the file to the previously encrypted version or does it become even more encrypted file.

```

root@remnux:/vmfs/volumes# /home/remnux/Downloads/./0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
ENCRYPTION
Done time: 0.0100 seconds, encrypted: 0.0000 gbroot@remnux:/vmfs/volumes# ls
readme.txt  Test123.txt.basta
root@remnux:/vmfs/volumes# cat Test123.txt.basta
}00)3'0$^00L00vB09hj<|00Y0^jL0qa0/00wf0Z0,000'0cn0LZu{Zr&cTF00S0Q00"+k0^0000>0%0[c00]00H000(0h000
0000-00a0ZNq.$00a
00UCQ0  +0LD0000c0N000,00LPu0S
C0yW`x  0\b0%r08'000o000!s0g00zj0X0DF00R0X00B$000m00
00-0P0@wY000|t[0q00f?000e)0`000 0vT0Y|05/N0<000_0,B0-000s0300-!$900w[0k0(Aljq=00J0000000A00000000]\00
M,0 000E00000000C5L000B0000`J00k600L0000`0s
3膈0`R[jl0000c0001xb0EB00g00z)NG00e000[0]9' _0
D000%00/,B0+04
0鏗}0NXe2c,#0000W000i0:'00050VQ x0,0root@remnux:/vmfs/volumes#

```

Figure 30: Executing the encryptor for the second time

The encryption became different from the first version. This didn't make much sense. After this I decrypted the file again with the root privileges and this time the file turned back to normal. It still was an executable for some reason, but it was recovered. The decryption phase might be made purposely a bit complicated, so that if a victim of this attack finds a correct decryptor for this ransomware they maybe would give up after testing this only one time.

```

root@remnux:/vmfs/volumes
root@remnux:/vmfs/volumes# /home/remnux/Downloads/./0d6c3de5aebbbe85939d7588150edf7b7bdc712fceb6a83d79e65b6f79bfc2ef.elf
ENCRYPTION
Done time: 0.0100 seconds, encrypted: 0.0000 gbroot@remnux:/vmfs/volumes# ls
readme.txt  Test123.txt.basta
root@remnux:/vmfs/volumes# cat Test123.txt.basta
}00}3'0$sk^00L00vB09hj<|00Y0^Jl0qa0/00af0Z0,000'0cn0LZu{Zr&cTF00S0Q00"+k0^0000>0%0[c00]00H000(0h000
0000-00a0ZNq.$00a
00UCQ00+0LD0000s0N000,00LPU0S
C0yW`x 0\b0%r08'000o000!s0g00zj0X0DF00R0X00B$000m00
00<0P0@WY000|t00q00f?000e)0`000 0vT0Y|05/N0<000_0,B0~000s0200!$900w[0k0(ALjq=00J0000000A00000000)\00
M,0 000E0000000C5L000B0000`J00k600L0000`0s
300 0`R[jl0000s0001Xb0EB00g00z)NG00e000[00]0' _0 D000%00/,B0+04
00}0NXe2c,#0000w000i0:'00050VQ x0,0root@remnux:/vmfs/volumes# ^C
root@remnux:/vmfs/volumes# /home/remnux/Downloads/./96339a7e87ffce6ced247feb9b4cb7c05b83ca315976a9522155bad726b8e5be.elf
DECRYPTION
Done time: 0.3050 seconds, encrypted: 0.0000 gbroot@remnux:/vmfs/volumes# ls
readme.txt  Test123.txt
root@remnux:/vmfs/volumes# cat Test123.txt
Testing 123..
root@remnux:/vmfs/volumes# █

```

Figure 31: Displaying the decrypted file

To decrypt the file, you need to run the encryptor, decryptor, encryptor again and last time the decryptor. This pattern recovers the files. It doesn't matter how many times in a row you would run the same encryptor or decryptor, it doesn't change anything the other time you run it. So, if you would run the first encryptor three times, it would only change it once, and if you then would run the decryptor three times, it would change it once, and this repeats until it gets recovered eventually.

This specific ransomware sample was interesting since it really was able to encrypt any data created to the vmfs/volumes folder. I did some more testing and if a file was created to the /vmfs folder it wasn't encrypted when the ransomware was detonated. If a folder was added to /vmfs folder for example "/vmfs/Test", its contents would not be encrypted either. The target of this ransomware was very specific, and the decryption method was interesting as well since only running the decryption once was not enough. Maybe a more professional can analyze this sample with the right tools and tell why the decryption worked this way, but my hypothesis is that it is only for making it slightly harder for the victim to decrypt the files and maybe give up before understanding how the decryption works.

6 Summary & conclusion

Ransomware keep constantly evolving far more complex and organizations must take the needed steps to mitigate the risks of ransomware attacks. Cyber criminals won't be applying the same tactics as they have by now but use new trends as they come. Organizations should

be up to date as these new trends appear and should update on a regular base their risk management plans, vulnerability management plans, and incident response plans. Back-ups won't be the thing that can be relied solely in the future as cyber criminals are using double extortion or similar techniques. Security awareness training should be more relevant to organizations since phishing is the most common method how these attacks start, and everyone can be a target. Doing security awareness trainings by reading a plain text can be a bit boring especially to non-technical people and a good way is to turn these trainings into an interactive game which is far more interesting to many people. Least privilege is also an important security principle to implement which means that everyone should only have the minimum privileges they need. This reduces attack surface and if an attacker gets access to an account, they usually still need to get access to a target with more privileges. Other similar method for preventing these attacks is to use a zero-trust model where no one should be trusted even anyone from inside. This way everyone should verify their access multiple times and if multifactor authentication is added to this, unauthorized access would be minimal. There are lots of different ways to protect from these attacks and even good security practices such as thinking before clicking, using strong passwords, keeping software's and operating system up to date, not sharing personal information on social media, and regular backups will give a solid base for defending against these attacks.

The thesis gives a fine theoretical background about malware, specifically ransomware, and their histories, ransomware attack processes, and practical example on how a ransomware functions. After reading this the reader should know the basics of malware. Even if the malware analysis part was included in this thesis, it wasn't the focus of it, but more on learning the cybercriminals processes and not to fall into ransomware attacks. This work can be used as a guide for individual cybersecurity enthusiasts or for organizations. It can be used to raise employee awareness or to teach cybersecurity students about ransomware.

Ransomware can be very damaging and costly for organizations. It is way better to invest in defending against ransomware attacks than to fall into one and needing to pay for the ransom. Many organizations don't do enough to protect from these attacks and there are continuously some organizations who becomes a victim. Even big named organizations sometimes become the victims, and this should be changed. Ransomware attacks have become popular, and this topic must be taken seriously to protect from these attacks.

References

Printed sources

Chapple, M; Seidl, D. July 2020 CompTIA Cybersecurity Analyst (CySA+) Study Guide Exam CS0-002 Second Edition Accessed 5.4.2023.

Gibson, D. 12 October 2017. CompTIA Security+: Get Certified Get Ahead SY0-501 Study Guide. Accessed 27.3.2023.

Monnappa, K. June 29, 2018. Learning Malware Analysis. Accessed 30.8.2023.

Sikorski, M & Honig, A. February 2012. PRACTICAL MALWARE ANALYSIS The Hands-On Guide to Dissecting Malicious Software Accessed 1.9.2023.

Electronic

Abuse.ch. March 17, 2020. Accessed 8.11.2023 <https://abuse.ch/blog/introducing-malware-bazaar/>

AV-TEST Institute. 2023. Malware. 27.3.2023 <https://www.av-test.org/en/statistics/malware/>

AV-Test Institute. 2023. Statistics. Accessed 5.4.2023 <https://portal.av-atlas.org/malware/statistics>

Aycock, J. 2006. Computer Viruses and Malware. E-book. Accessed 5.4.2023.

Bagde, Ayush. March 9, 2021. History of Malware TryHackMe Writeup. Accessed 27.3.2023 <https://override.medium.com/history-of-malware-tryhackme-writeup-4fcb5aef68bb>

CrowdStrike. January 30, 2023. 5 TYPES OF RANSOMWARE. Accessed 30.3.2023 <https://www.crowdstrike.com/cybersecurity-101/ransomware/types-of-ransomware/>

Exabeam, 2017. THREAT RESEARCH REPORT THE ANATOMY OF A RANSOMWARE ATTACK. Accessed 29.4.2023 https://www.exabeam.com/wp-content/uploads/2017/07/Exabeam_Ransomware_Threat_Report_Final.pdf

F-Secure. Trojan-Downloader. 27.3.2023 <https://www.f-secure.com/v-descs/trojan-downloader.shtml>

Graziano, D. October 29, 2015. MICROSOFT WORD 'MACRO MALWARE' VIRUSES ARE BACK IN A BIG WAY Accessed 29.4.2023 <https://www.mailguard.com.au/blog/microsoft-word-macro-malware-viruses-are-back-in-a-big-way>

Kelly, R. January 14, 2022. What Was Elk Cloner, the First Computer Virus to Spread 'In the Wild'? Accessed 27.3.2023 <https://www.digit.fyi/elk-cloner/>

Livingston, Z. November 2, 2022. The History of Computer Viruses & Malware. Accessed 27.3.2023 <https://www.esecurityplanet.com/threats/computer-viruses-and-malware-history/>

McAfee. Understanding Trojan Viruses and How to Get Rid of Them. Accessed 27.3.2023 <https://www.mcafee.com/learn/understanding-trojan-viruses-and-how-to-get-rid-of-them/>

Malwarebytes. Computer Virus. Accessed 27.3.2023 <https://www.malwarebytes.com/computer-worm>

Malwarebytes. Computer Worm. Accessed 27.3.2023 <https://www.malwarebytes.com/computer-virus>

Martin, L. 2015 GAINING THE ADVANTAGE Applying Cyber Kill Chain® Methodology to Network Defense. Accessed 15.4.2023 https://www.lockheedmartin.com/content/dam/lockheed-martin/rms/documents/cyber/Gaining_the_Advantage_Cyber_Kill_Chain.pdf

Miller, L. January 15, 2020. Ransomware Defense for Dummies, Cisco 2nd Special Edition. E-book. Accessed 5.4.2023.

Milosevic, N. February 2012. History of malware. Accessed 27.3.2023 https://www.researchgate.net/publication/235666537_History_of_malware

Morgan, S. July 7, 2023. Global Ransomware Damage Costs Predicted To Exceed \$265 Billion By 2031 Accessed 30.3.2023 <https://cybersecurityventures.com/global-ransomware-damage-costs-predicted-to-reach-250-billion-usd-by-2031/>

REMnux February 2023. INSTALL THE DISTRO - Get the Virtual Appliance. Accessed 5.9.2023 <https://docs.remnux.org/install-distro/get-virtual-appliance>

Saengphaibul, V. March 15, 2022. A Brief History of The Evolution of Malware. Accessed 27.3.2023 <https://www.fortinet.com/blog/threat-research/evolution-of-malware>

Thepsiri, T. February 1, 2023. What Is Malware? How Can It Affect My Business? Accessed 27.3.2023 <https://www.kelsercorp.com/blog/what-is-malware-how-does-it-spread>

Trend Micro Research. September 1, 2022. Accessed 8.11.2023 <https://www.trendmicro.com/vinfo/us/security/news/ransomware-spotlight/ransomware-spotlight-blackbasta>

Tunggal, A. April 6, 2023. 22 Types of Malware and How to Recognize Them in 2023. Accessed 3.10.2023 <https://www.upguard.com/blog/types-of-malware>

VirtualBox 2023. Download VirtualBox. Accessed 10.9.2023 <https://www.virtualbox.org/wiki/Downloads>

VirusTotal 2023. How it works. Accessed 8.11.2023 <https://support.virustotal.com/hc/en-us/articles/115002126889-How-it-works>

Yadav, T & Mallari, A. August 2015. Technical Aspects of Cyber Kill Chain. Accessed 15.4.2023
https://www.researchgate.net/publication/281148852_Technical_Aspects_of_Cyber_Kill_Chain

Figures

Figure 1: Total amount of malware and PUA (AV-Test 2023).....	10
Figure 2: Windows malware categories (AV-Test 2023)	11
Figure 3: How ransomware functions (adapted from Miller. L 2020).....	13
Figure 4: Cyber kill chain (adapted from Martin. L 2015).....	14
Figure 5: Ransomware kill chain (adapted from Exabeam 2017)	15
Figure 6: Illustration of a virtual machine (Sikorski, M & Honig, A. 2012).....	18
Figure 7: Downloading the virtual appliance file (REMnux 2023)	19
Figure 8: Taking a snapshot of a clean virtual machine in VirtualBox	20
Figure 9: Network settings page in VirtualBox	21
Figure 10: Downloading first malware sample from MalwareBazaar (MalwareBazaar 2023)...	22
Figure 11: Downloading second malware sample from MalwareBazaar (MalwareBazaar 2023)	22
Figure 12: Customizing network settings of REMnux	23
Figure 13: VirusTotal's analysis of first malware sample (VirusTotal 2023)	24
Figure 14: VirusTotal's analysis of second malware sample (VirusTotal 2023)	24
Figure 15: Test files in Documents folder	25
Figure 16: Ransomware shown in Downloads folder	25
Figure 17: Giving executable rights to the ransomware files	26
Figure 18: Execution of ransomware.....	26
Figure 19: Displaying the default screen of Detect It Easy	27
Figure 20: Detect It Easy revealing information about the ransomware.....	27
Figure 21: Detect It Easy displaying hex values	28
Figure 22: Analyzing strings with Detect It Easy	28
Figure 23: Strings reveal a ransom note in Detect It Easy	29
Figure 24: Checking if this virtual machine has /vmfs/volumes path.....	29
Figure 25: Creating /vmfs/volumes path and adding a test file in volumes folder	30
Figure 26: Executing the encryptor	31
Figure 27: Displaying test files in Documents folder	31
Figure 28: Displaying encrypted test file in /vmfs/volumes.....	32
Figure 29: Decrypting the test file and displaying it	33
Figure 30: Executing the encryptor for the second time.....	33
Figure 31: Displaying the decrypted file	34