Written by:

Group 5

Elijah Perez

Daniel Meyers

Hyung Jin Kim

Daniele Ricciardelli

Ransomware Research & Simulation

LockBit, WannaCry, Locky, CryptoLocker

Presented to:

Dr. Tamer R. Omar

Friday

December 5th, 202

Table of Contents

1. ABSTRACT

Ransomware attacks have emerged as a significant cybersecurity threat, targeting individuals and organizations across various sectors. This report provides a comprehensive analysis of four prominent ransomware families: LockBit, WannaCry, Locky, and CryptoLocker. Each ransomware strain is examined in detail, highlighting their evolution, delivery mechanisms, encryption methods, and real-world impacts.

LockBit operates under a Ransomware-as-a-Service (RaaS) model, enabling even non-technical cybercriminals to execute sophisticated attacks. Its evolution through versions 2.0 and 3.0 to the anticipated 4.0 showcases continuous innovation, with advanced techniques like double extortion and automated encryption. WannaCry leveraged the EternalBlue vulnerability, causing global disruptions, including a notable attack on the UK's National Health Service. Its rapid propagation through SMB exploits underscored the critical need for timely system updates. Locky, a pioneer in phishing-based ransomware, targeted organizations with tailored social engineering tactics, though its effectiveness waned due to advancements in cybersecurity. Lastly, CryptoLocker set a benchmark in encryption efficiency, employing dual-layer encryption to lock files and induce panic among victims.

The report incorporates controlled simulations of these ransomware strains in isolated environments to observe their behaviors, identify vulnerabilities, and analyze their impact on victim systems. The findings emphasize the importance of robust cybersecurity measures, including patch management, employee training, and layered defenses, to mitigate the risks posed by evolving ransomware threats. These insights aim to inform and enhance strategies for combating the growing menace of ransomware.

2. OBJECTIVES

The learning outcome from ransomware research includes a comprehensive understanding of the operational mechanisms, propagation methods, and impacts of ransomware attacks. By studying specific ransomware such as WannaCry and LockBit, individuals gain insights into encryption techniques used by attackers, and the steps required for detection and prevention.

3. INTRODUCTION

a. LOCKBIT

LockBit is one of the most recent and advanced ransomware that has been made. It first came out in 2019 and was initially referred to as "ABCD" [5], but Lockbit has evolved multiple times with several major updates each improving its effectiveness, making it harder to defend against its attacks. The first version was relatively basic and relied on simple network propagation techniques to spread within an organization, but LockBit would eventually evolve to incorporate more sophisticated tactics that would be seen in Lockbit 2.0 and 3.0. For instance, LockBit 3.0 introduced double extortion which is a strategy where an attacker would not only encrypt data but also threaten to release the victim’s data unless the ransom is paid [6].

LockBit's has had a big impact on the cybersecurity world as it has been used in several high-profile attacks. One of the first large-scale attacks was in 2021 on the global consulting firm Accenture which reportedly demanded a ransom of $50 million [6]. This breach not only impacted Accenture's systems but also sensitive client data. There were also Lockbit attacks on the healthcare sector such as the attack on the largest medical facility in Croatia in June 2024 where it’s said to have forced them "back 50 years—to paper and pencil" [4]. These attackers target medical data and demand ransom payments to avoid public release. These attacks highlight the trend of Lockbit being used to target high-value and high-profile organizations that can afford to pay large ransoms.

LockBit is known for its automated encryption process that minimizes the time between initial compromise and full encryption of a victim’s data. Lockbit encrypts files using a combination of the Salsa20 algorithm and 1024-bit RSA keys [3]. To stay undetected Lockbit employs a technique called living off the land where it uses legitimate tools and system resources to move within a network. Most notably it stops services such as Windows Security by taking advantage of trusted install services [1].

b. WANNACRY

WannaCry ransomware, which emerged in May 2017, became infamous for its global impact and rapid spread. It exploited a vulnerability in Windows' SMB protocol, called EternalBlue, developed by the NSA and leaked by the Shadow Brokers. This allowed WannaCry to spread across networks automatically without user interaction. Once infected, the ransomware encrypted files, adding the ".wncry" extension. Victims were shown a ransom note demanding Bitcoin payments (a few hundred dollars) for a decryption key, with threats to increase the ransom over time. Multilingual messages ensured global accessibility, pressuring victims to pay quickly. [16]

WannaCry acted like a "worm," propagating to other vulnerable computers on the same network, which amplified its damage. It severely disrupted organizations worldwide, including the UK’s National Health Service (NHS), where it crippled healthcare services by encrypting medical records. Various versions of WannaCry followed, some modifying features like the original "kill switch" domain, which researchers used to stop the initial outbreak. Later versions removed this feature, allowing continued, though reduced, infections. [17]

WannaCry used strong RSA and AES encryption, making decryption nearly impossible without the ransom. Its use of an unpatched SMB exploit bypassed standard defenses, causing damages estimated in the hundreds of millions to billions of dollars. [17]

The attack highlighted the critical importance of timely software updates and cybersecurity practices. Microsoft released emergency patches, including updates for outdated systems like Windows XP. The event spurred global discussions on cybersecurity, especially in essential services, and emphasized the need for better patch management. Despite containment, WannaCry exposed the dangers of delayed updates and weak defenses, serving as a key example of how quickly malware can exploit vulnerabilities to wreak havoc. [18]

c. LOCKY

Locky was first identified in February 2016 [7], became a prominent and highly disruptive ransomware strain, gaining notoriety for its effective distribution methods and sophisticated encryption techniques. Locky relied heavily on social engineering known as phishing, which is still a popular choice of attack, to infiltrate systems, often delivered through spam emails containing malicious attachments or links. Its first big attack happened at Hollywood Presbyterian Medical Center. What was thought to be an invoice was a malicious attack which allowed the Locky to wreak havoc inside the healthcare network locking the doctors and nurse out of the patients' medical records by encrypting them and force closure of their emergency room. The hospital network was out for about one week before able to regain access by paying a ransom of $17,000 [8]. Its primary targets were schools, institutions, healthcare organization, communication companies and governments. They were able to mass distribute this malware and not only that distribute them in multiple different file types.

Once executed, Locky encrypted the files on the infected machine, appending extensions such as ".locky," ".zepto," or ".odin" to the affected files, depending on the variant. Victims were presented with a ransom note instructing them to install the Tor browser and access a payment portal on the dark web. The ransom demand, typically made in Bitcoin, varied in amount but was often set around 0.5-bit coin [9], aligning with its focus on extracting payment quickly and efficiently. These attackers were assumed to be connected to a mysterious group of hackers know as Dridex hackers, also know as Evil corp or TA505 [10].

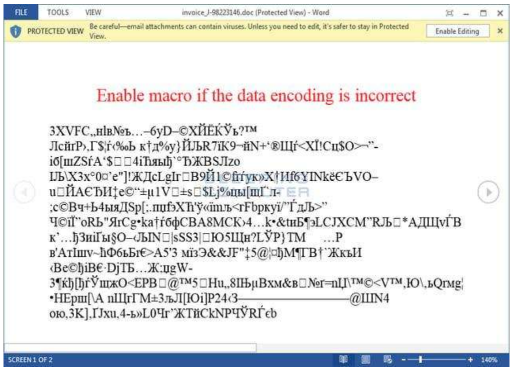
Over time, multiple variants of Locky emerged, each employing subtle modifications to evade detection and adapt to changing cybersecurity measures. Despite declining activity in 2017, Locky's influence persisted, inspiring other ransomware campaigns and serving as a blueprint for similar malware strains.

The rise of Locky highlighted the critical importance of cybersecurity awareness and proactive measures, including employee training, robust email filtering, and regular backups [10]. Organizations learned the necessity of implementing layered defenses that combined technological safeguards with user education to mitigate the risks posed by social engineering-based threats.

Locky remains a significant example of how effective social engineering and well-crafted malware can disrupt systems and cause widespread financial and operational damage. Its legacy continues to inform ransomware defense strategies and reinforces the need for vigilance in combating evolving cyber threats.

The delivery method of Locky ransomware was primarily focused on exploiting human vulnerabilities through phishing emails and social engineering tactics [7]. When it first appeared, it was commonly delivered through spam emails containing malicious attachments. These attachments were often disguised as legitimate files, such as invoices, receipts, or shipping notifications, to trick users into opening them. Many of these attachments were Microsoft Word and Excel documents with macros. These macros were obfuscated Visual Basic Scripts [5]. Upon opening the document, users were prompted to enable macros, which would then automatically execute malicious scripts to download and install the ransomware as they contained the autoopen() sub.

After the Microsoft Word and Excel macro scripts, the attackers adapted and implemented Locky as a Javascript, .js, files as attachment as well [7]. These scripts will contact a remote server to download the ransomware payload and execute it on the victim's system. Once infected event unmapped network shared drives can become encrypted if a computer becomes infected with Locky. As this kind of attack were becoming more common sysadmin started locking down open network shared to the lowest permission level [9].



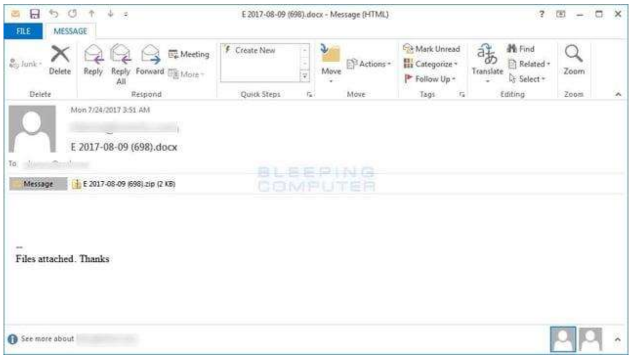
Be careful—email attachments can contain viruses. Unless you need to edit, it's safer to stay in Protected View.

Enable macro if the data encoding is incorrect

3XVFC„нlв№ъ…–6yD–©ХЙЁЌЎъ?™
ЛсйгР›.Г$¦ѓ‹‰ Љ к†д%у}ЙЉR7iК9¬йN+'®Щѓ<ХЇ!Сц$О>¬"-
iб[шZЅѓА'$□□4iЋяыђ`°ЂЖBSЛzo
ЉЬ\Х3х°0¤'е"]!ЖДсLgІг□В9Й1©ѓгўк»Х†Иf6YINkёЄЪVO–
u□ЙАЄЂИ‡е©"±µ1V□±s□$Lj%цы[шЃ`л-
;с©Вч+Ь4ыяДSp[;.пцfэХЋ'ў«іmљ<rFbpкуї/"ЃдЉ>"
Ч©ЇЃ"оRЬ"ЯгСg•ka†ѓбфCBA8МСК›4…k•&tиБ¶эLCJXCM"RЉ□*АДЩvЃВ
к'…ђ3иіЃы§О–‹ЉIN□|sSS3|□Ю5Щн?LЎР}ТМ …Р
в'АтІшv~ћФ6ьБгЄ>А5'3 мїзЭ&&JF"‡5@¦¤ђМ¶ГВ†`ЖкъИ
‹Ве©ђіВЄ·DjТБ…Ж:цgW-
3¶ќђ[ђѓЎшжО<ЕРВ□@™5□Hu„8ІЊµВхм&в□№г=пЏ\™©<V™,Ю\,ьQгмg¦
•НЕрш[\А пЩгГМ±3љЛ[Юі]Р24‹3————————@ШN4
ою,3К],ЃJхu,4-ь»L0Чг'ЖТйСkNРЧЎRЃєb

Figure 1 – Gibberish writing In Word to induce users to enable macro [7]

Over the years other variants of Locky started popping up as well. Such as Diablo 6, a variant that started spreading in August 2017 [12]. This variant just sent an email that said Files attached. Thanks. Which just contained a zip file along with that message. Once comprised it encrypts the files with a .diablo6 extension.



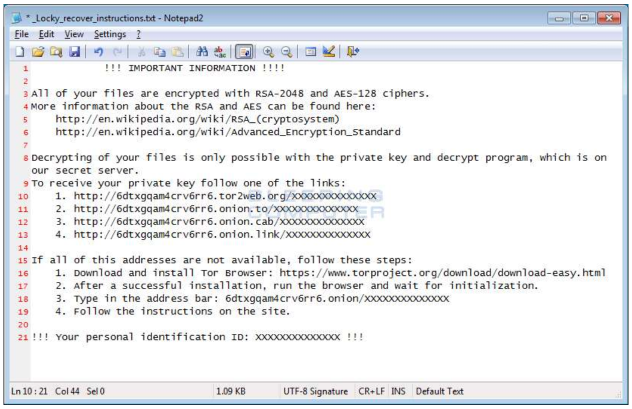
Mon 7/24/2017 3:51 AM
[redacted]
E 2017-08-09 (698).docx
To: [redacted]
E 2017-08-09 (698).zip (2 KB)

--
Files attached. Thanks

Figure 2 - Diablo variant attached message [7].

Once encrypted the ransom part of the attack begins. Locky will create a ransom note on your pc called _Locky_recover_instruction.txt [9]. Which contains information regarding how you can recover your data which is to go to a tor browser and go to a dark website and need to pay around 0.5~1 bitcoin to get back your data.

Encryption of Locky involves a dual encryption method using AES and RSA encryption algorithm. It uses AES-128 and RSA-2048, it uses the AES to encrypt the files on the victim's device. It generates a new AES key for every file encrypted before sending over the key to the Command & Control Server. using the public RSA key to encrypt it before sending it [13].

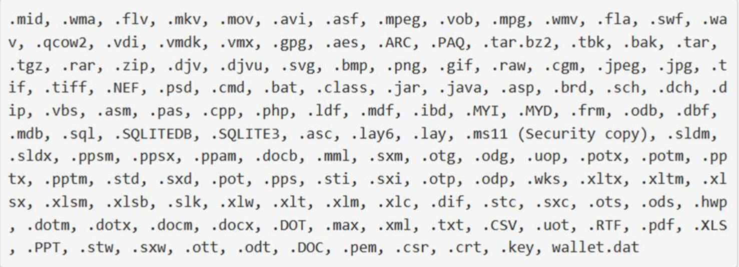


```
           !!! IMPORTANT INFORMATION !!!!

All of your files are encrypted with RSA-2048 and AES-128 ciphers.
More information about the RSA and AES can be found here:
    http://en.wikipedia.org/wiki/RSA_(cryptosystem)
    http://en.wikipedia.org/wiki/Advanced_Encryption_Standard

Decrypting of your files is only possible with the private key and decrypt program, which is on
our secret server.
To receive your private key follow one of the links:
    1. http://6dtxgqam4crv6rr6.tor2web.org/XXXXXXXXXXXXXX
    2. http://6dtxgqam4crv6rr6.onion.to/XXXXXXXXXXXXXX
    3. http://6dtxgqam4crv6rr6.onion.cab/XXXXXXXXXXXXXX
    4. http://6dtxgqam4crv6rr6.onion.link/XXXXXXXXXXXXXX

If all of this addresses are not available, follow these steps:
    1. Download and install Tor Browser: https://www.torproject.org/download/download-easy.html
    2. After a successful installation, run the browser and wait for initialization.
    3. Type in the address bar: 6dtxgqam4crv6rr6.onion/XXXXXXXXXXXXXX
    4. Follow the instructions on the site.

!!! Your personal identification ID: XXXXXXXXXXXXXX !!!
```

Figure 3 – Ransom note generated by Locky [9].

Before encrypting files, Locky ransomware follows several steps to ensure successful encryption. It needs to communicate with a Command & Control Server (C2) to setup the Locky so that a decryption can happen later when the ransom is paid. The C2 Communication happens with HTTP with either using a Domain Generation Algorithm (DGA) to dynamically switch which DNS it uses to talk to the C2 Server or as a backup use the hardcoded IP address in the Locky. The C2 Server will communicate with the ransomware and will generate a User ID. The Locky then checks information regarding the infected device, such OS version, 32/64-bit, PC language. Afterwards it will generate an RSA key pair and give that to the infected PC so that the AES key can be encrypted before removing the plaintext and sending. This also means that if that there is no communication that can occur with the C2 Server then no encryption takes place due to the attacker not being able to retrieve the encrypted AES key for later decryption which means the attacker has no way to decrypt the files later [11]. It will then scan the PC and its network shared drives for potential files it can encrypted and start to proceed encrypting. It typically avoids any files that has to do with the OS as damaging the OS is not something it wants to do if the attacker wants a payout.

Once the ransom has been paid a Locky decryptor will be available for download which will contain the hard-coded private RSA key and the encrypted AES keys and can be used to decrypt files using those two keys.



```
.mid, .wma, .flv, .mkv, .mov, .avi, .asf, .mpeg, .vob, .mpg, .wmv, .fla, .swf, .wa
v, .qcow2, .vdi, .vmdk, .vmx, .gpg, .aes, .ARC, .PAQ, .tar.bz2, .tbk, .bak, .tar,
.tgz, .rar, .zip, .djv, .djvu, .svg, .bmp, .png, .gif, .raw, .cgm, .jpeg, .jpg, .t
if, .tiff, .NEF, .psd, .cmd, .bat, .class, .jar, .java, .asp, .brd, .sch, .dch, .d
ip, .vbs, .asm, .pas, .cpp, .php, .ldf, .mdf, .ibd, .MYI, .MYD, .frm, .odb, .dbf,
.mdb, .sql, .SQLITEDB, .SQLITE3, .asc, .lay6, .lay, .ms11 (Security copy), .sldm,
.sldx, .ppsm, .ppsx, .ppam, .docb, .mml, .sxm, .otg, .odg, .uop, .potx, .potm, .pp
tx, .pptm, .std, .sxd, .pot, .pps, .sti, .sxi, .otp, .odp, .wks, .xltx, .xltm, .xl
sx, .xlsm, .xlsb, .slk, .xlw, .xlt, .xlm, .xlc, .dif, .stc, .sxc, .ots, .ods, .hwp
, .dotm, .dotx, .docm, .docx, .DOT, .max, .xml, .txt, .CSV, .uot, .RTF, .pdf, .XLS
, .PPT, .stw, .sxw, .ott, .odt, .DOC, .pem, .csr, .crt, .key, wallet.dat
```

Figure 4 – Types of files that get encrypted after Locky [9]

d. CRYPTOLOCKER

Ransomware attacks have become increasingly sophisticated, targeting critical infrastructure and personal devices. This paper focuses on two major ransomware variants: WannaCry and LockBit. The objective of this research is to understand the techniques used by these ransomware families and simulate their impact in controlled environments. A cloud simulation serves as an additional investigation, aimed at understanding the ransomware's behavior in modern cloud infrastructures.

CryptoLocker was a type of ransomware that first appeared in September 2013, targeting Windows computers. It was spread primarily through malicious email attachments and botnets like Gameover ZeuS. Once executed, the malware encrypted files on the victim's computer using strong RSA-2048 encryption and displayed a ransom demand, often in Bitcoin, for the decryption key. Victims typically had a short deadline to pay before the key was allegedly destroyed, rendering their files inaccessible permanently. [26]

The ransomware was notorious for its effectiveness and sophisticated encryption, making decryption without the private key nearly impossible. Authorities dismantled the Gameover ZeuS botnet in mid-2014, significantly reducing CryptoLocker infections. However, its success inspired many imitators, marking a significant shift in ransomware trends.

NOTE** This malware is no longer online and was shut down by the FBI, there are special hacking tools like Kali that still carry the programs necessary to run this malware, however, I dont have a Kali-based system. There are different alterations of this malware online but in this report, I will be analyzing the original.

4. EXPERIMENTAL METHODOLGY

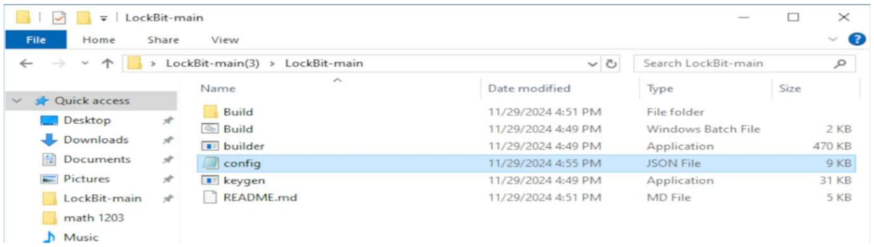
a. LOCKBIT

LockBit is known for its automated encryption process that minimizes the time between initial compromise and full encryption of a victim's data. Lockbit encrypts files using a combination of the Salsa20 algorithm and 1024-bit RSA keys which despite its reputation is a slightly outdated ket size which can be seen in its speed of the deployment during testing [3]. To stay undetected Lockbit employs a technique called living off the land where it uses legitimate tools and system resources to move within a network. Most notably it stops services such as Windows Security by taking advantage of trusted install services [1].

The two most common ways Lockbit is delivered is through phishing where a macro can be put into a document that is set to execute once a victim selects edit document. Another more direct method is using RDP to remotely transfer the files to the victims machine. This method requires you to know the target IP address, the target must have port 3389 open for RDP, use "brute force tools such as Crowbar or Hydra", and finally connect through a client where the Lockbit can then be executed [2]. LockBit also uses data leak sites which has become a key feature of its double extortion approach, as it publishes stolen data to pressure victims into paying the ransom.

The environment chosen to carry out these tests was the Virtual Box hypervisor which allows a controlled and secure way to release LockBit on a system. The operating system chosen was Windows 10 and allocated as many resources as was reasonably available, 4068 MB of memory, 3 processors, 250 GB of virtual hardware storage, and the network was using a Host-only-Adapter which cut the virtual machine from the local network. The reason testing was done without internet connection is that LockBit is able to travel within a network and compromise other machines.

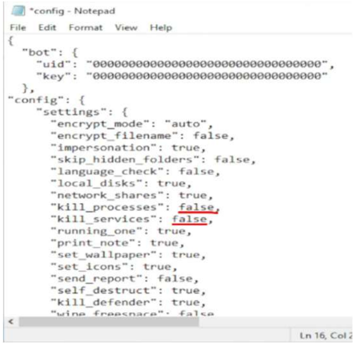
Analysing the contents of the LockBit main folder we see various tools provided to streamline the process of configuring and assembling LockBit. The Build.bat file will run a script to completely automate the process but taking a closer look the key features are build.exe, config.json, and keygen.exe. The keygen application generates a "unique public-private key pair for each build as well as the decryption ID" Lockbit itself states that this relies on "MIRACL, which according to [LockBit's] own description is, 'a C software library that is widely regarded by developers as the gold standard open-source SDK for elliptic curve cryptograph." [1]. The keys are saved as priv.key and pub.key and are "base64 encoded" [1]. There is then a file created called DECRYPTION_ID.txt file which is a " unique victim identification number" that comes from the "first eight bytes of the public key" [1].



Name	Date modified	Type	Size
Build	11/29/2024 4:51 PM	File folder	
Build	11/29/2024 4:49 PM	Windows Batch File	2 KB
builder	11/29/2024 4:49 PM	Application	470 KB
config	11/29/2024 4:55 PM	JSON File	9 KB
keygen	11/29/2024 4:49 PM	Application	31 KB
README.md	11/29/2024 4:51 PM	MD File	5 KB

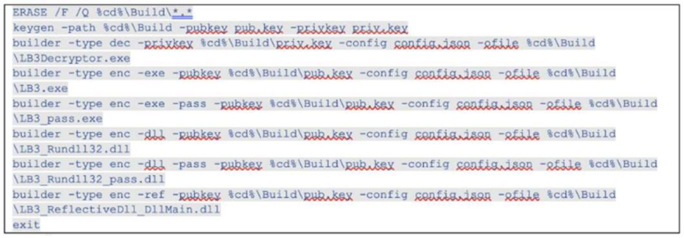
Figure 5 - Contents of LockBit main

The next file to look at is the config file which already has default parameters set in place but allows lots of configuration if the user requires. The builder.exe is the last key component of the before LockBit is assembled which requires the creation of the priv/pub.key files and the config.json. The builder.exe will then create "two executables, three dynamic link libraries, and two text files.Once the builder.bat is completed it will populate the Build folder containing the final resources to configure and modify LockBit before it is executed.



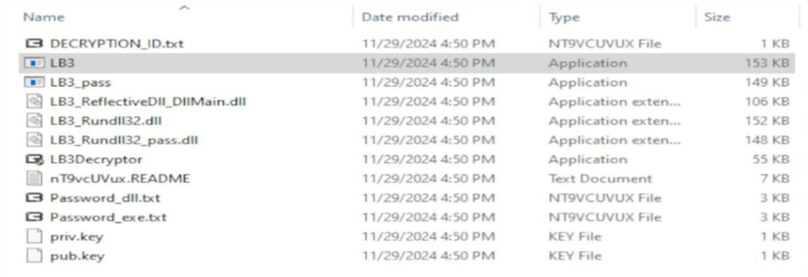
```
{
  "bot": {
    "uid": "00000000000000000000000000000000",
    "key": "00000000000000000000000000000000"
  },
"config": {
    "settings": {
      "encrypt_mode": "auto",
      "encrypt_filename": false,
      "impersonation": true,
      "skip_hidden_folders": false,
      "language_check": false,
      "local_disks": true,
      "network_shares": true,
      "kill_processes": false,
      "kill_services": false,
      "running_one": true,
      "print_note": true,
      "set_wallpaper": true,
      "set_icons": true,
      "send_report": false,
      "self_destruct": true,
      "kill_defender": true,
      "wipe_freespace": false
```

Figure 6 - Config file with slight adjustments



```
ERASE /F /Q %cd%\Build\*.*
keygen -path %cd%\Build -pubkey pub.key -privkey priv.key
builder -type dec -privkey %cd%\Build\priv.key -config config.json -ofile %cd%\Build
\LB3Decryptor.exe
builder -type enc -exe -pubkey %cd%\Build\pub.key -config config.json -ofile %cd%\Build
\LB3.exe
builder -type enc -exe -pass -pubkey %cd%\Build\pub.key -config config.json -ofile %cd%\Build
\LB3_pass.exe
builder -type enc -dll -pubkey %cd%\Build\pub.key -config config.json -ofile %cd%\Build
\LB3_Rundll32.dll
builder -type enc -dll -pass -pubkey %cd%\Build\pub.key -config config.json -ofile %cd%\Build
\LB3_Rundll32_pass.dll
builder -type enc -ref -pubkey %cd%\Build\pub.key -config config.json -ofile %cd%\Build
\LB3_ReflectiveDll_DllMain.dll
exit
```

Figure 7: Content of Build.bat [1]



Name	Date modified	Type	Size
DECRYPTION_ID.txt	11/29/2024 4:50 PM	NT9VCUVUX File	1 KB
LB3	11/29/2024 4:50 PM	Application	153 KB
LB3_pass	11/29/2024 4:50 PM	Application	149 KB
LB3_ReflectiveDll_DllMain.dll	11/29/2024 4:50 PM	Application exten...	106 KB
LB3_Rundll32.dll	11/29/2024 4:50 PM	Application exten...	152 KB
LB3_Rundll32_pass.dll	11/29/2024 4:50 PM	Application exten...	148 KB
LB3Decryptor	11/29/2024 4:50 PM	Application	55 KB
nT9vcUVux.README	11/29/2024 4:50 PM	Text Document	7 KB
Password_dll.txt	11/29/2024 4:50 PM	NT9VCUVUX File	3 KB
Password_exe.txt	11/29/2024 4:50 PM	NT9VCUVUX File	3 KB
priv.key	11/29/2024 4:50 PM	KEY File	1 KB
pub.key	11/29/2024 4:50 PM	KEY File	1 KB

Figure 8 - Content of Build folder

Amongst the many included applications and files the ones of note for our testing is LB3.exe and LB3Decryptor.exe are what will be tested. LB3.exe is used instead for the purposes of the demonstration as the counterparts require a password to run. The next is LB3Decryptor.exe which will decrypt LockBit regardless of which version the user chooses to deploy. Provided below is a description of each file [1].

a. DECRYPTION_ID.txt – Text file containing a 16-character victim ID made from the first eight hex bytes of the public key that is used to uniquely identify a victim
b. LB3.exe – Compiled ransomware, which doesn't require a password
c. LB3Decryptor.exe – Decryptor for the ransomware, which works with all the variations here
d. LB3_pass.exe – Same as LB3.exe however requires a password to run. The password and instructions are found in Password_exe.txt in this directory
e. LB3_RelectiveDLL_DLLMain.dll – Version of the ransomware that is meant to be reflectively loaded and executed in memory
f. LB3_Rundll32.dll – DLL version of ransomware, which doesn't require a password.
g. LB3_Rundll32_pass.dll – DLL version of ransomware, which requires the password found in the Password_dll.txt file
h. Password_dll.txt – Contains password and instructions for using LB3_Rundll32_pass.dll
i. Password_exe.txt – Contains password and instructions for using LB3_pass.exe
j. priv.key – A private encryption key unique to this build that is used to encrypt victim files
k. pub.key – A public encryption key unique to this build that is used generate various strings that tie this instance of the ransomware to a victim. [1]

b. WANNACRY

EternalBlue played a key role in WannaCry's rapid spread due to its ability to automate infection and avoid user interaction. Targeting the Windows SMBv1 protocol, EternalBlue exploited a buffer overflow vulnerability, allowing WannaCry to spread autonomously within and across networks. This enabled the ransomware to infect systems that weren't directly exposed to external threats but were connected to an infected machine, amplifying its reach in organizations.

By sending specially crafted packets to vulnerable systems, EternalBlue gained unauthorized access and installed WannaCry without triggering typical security alerts. It then scanned for other vulnerable devices on the same network, repeating the cycle of infection. This made it particularly destructive in environments with mixed outdated and newer systems, such as healthcare, finance, and transportation, where internal networks often lacked strict segmentation. [18]

EternalBlue didn't rely on user actions or permissions, making it stealthy and difficult to intercept once it started spreading. Even standard antivirus tools struggled to stop its network-based propagation in real time, revealing weaknesses in patch management and network segmentation.

WannaCry's encryption process was designed for maximum impact, targeting many file types with a dual-layer encryption method. It first used AES (Advanced Encryption Standard), a fast and efficient symmetric encryption algorithm, to encrypt files. This ensured files were locked quickly before detection or mitigation.

Next, WannaCry encrypted the AES keys with RSA (Rivest-Shamir-Adleman) public key encryption, an asymmetric method. This meant the AES keys could only be decrypted with the attackers' private RSA key, making it impossible for victims to unlock their files without paying the ransom. This layered approach combined AES's speed and RSA's security. The use of AES and RSA offered two key advantages: it allowed rapid encryption while maintaining strong security, and it required only a single RSA public key, reducing storage demands on the ransomware. [19]

WannaCry targeted valuable file types like documents, images, and databases, increasing pressure on victims to pay. Encrypted files were marked with a ".wncry" extension, making the attack obvious. This encryption method was significant both technically and psychologically. By locking files with tested encryption standards, WannaCry left victims with few options, effectively holding data hostage and increasing the likelihood of ransom payments. [20]

c. LOCKY

Similarly, to understand Locky ransomware's behavior, a simulation process was also conducted in a controlled environment. Among several virtualization software available, including VMware and Hyper-V, VirtualBox was chosen due to its open-source nature, ease of use, and compatibility with various operating systems. VirtualBox provided a secure and flexible platform for setting up a virtual machine (VM) dedicated to ransomware testing.

The simulation began with the creation of a Windows 10 Virtual Machine within VirtualBox. The VM is configured to mimic a typical user environment while ensuring isolation from the host system to prevent accidental spread or data compromise. During the setup, Windows Defender antivirus and the firewall are deliberately turned off. This step ensures that the ransomware can execute and function without interference, providing a clearer understanding of its behavior.

Once the VM is prepared, Wireshark, a powerful network protocol analyzer, is installed to monitor and analyze any network activities during the simulation. This tool helps identify Locky's communication with command-and-control (C2) servers, key exchanges, or data exfiltration attempts.

The Locky ransomware executable is then obtained from a trusted repository [8], such as a GitHub repository that hosts malware samples for research purposes. It's crucial to verify that the source is legitimate and that the simulation is conducted in an isolated and secure environment to mitigate risks.

After the setup, the Locky executable is launched within the VM to observe its behavior. The simulation focuses on monitoring the changes made to the file system, the encryption process, and communication with external servers. This step provides valuable insights into the ransomware's infection methodology and its impact on the system.

However, despite following this setup, the ransomware did not execute as expected. Suspecting that the issue might be related to the Windows 10 operating system being used, the process was restarted with a Windows 7 VM to determine if the problem was OS-related.

d. CRYPTOLOCKER

The CryptoLocker ransomware simulation will be conducted in a secure VM environment. Using known samples of CryptoLocker, we will demonstrate how the malware executes following infection through malicious email attachments. The encryption process using RSA-2048 will be observed, along with system behavior during infection, including file access patterns, CPU usage, and any communication with command-and-control servers.

A clean installation of Windows 7 was set up in a virtual machine using VMware. Networking features were disabled to prevent any potential spread of the ransomware outside the VM. The VM environment was secured with snapshots to allow easy restoration after the simulation.

A sample of the CryptoLocker ransomware was downloaded directly into the VM. The sample was obtained from a trusted cybersecurity research database, ensuring its authenticity and integrity. All downloads and operations were conducted strictly within the VM to maintain containment.

The ransomware executable was launched within the VM. System monitoring tools, including Task Manager and Process Monitor, were used to observe the behavior of the ransomware in real-time.

Upon execution, the ransomware immediately began its operation. It scanned directories to identify and encrypt non-executable files (e.g., .doc, .jpg, .zip). The encryption process was rapid and efficient, rendering nearly all targeted files inaccessible within seconds.

Once the encryption was complete, a ransom note was displayed in a pop-up window. The note informed the user of the encryption, provided instructions for payment, and threatened the permanent loss of files if the ransom was not paid promptly.

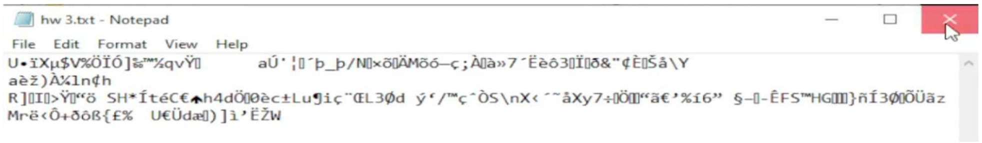
5. EXPERIMENTAL RESULT

a. LOCKBIT

After deployment of LockBit stopped some processes and services, for example Wireshark was running in the background to catch any potential attempts of telemetry but was forcefully closed throughout deployment and after encryption. This was strange as we had modified the config.json file that LockBit provides to disable “kill_proccesses” and “kill_services” but evidently these commands must not work in the way we expected.

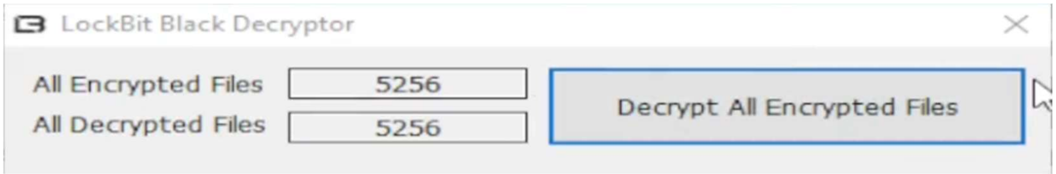
The background and file icons were all changed with a text file provided in every directory instructing the user how to send a payment for encryption. File encryption was in effect and when files were opened it resulted in an unreadable mess of characters. Now it was time to test the decryptor which gave a simple popup with a button to decrypt all files which when run worked quickly and successfully, decrypting all files.

Most functionalities resumed as prior to encryption and programs such as Wireshark were able to be run again but there were two notable changes that were not fixed. The first is the background was set to a default opaque blue which is of course an easy fix, but the second more concerning effect is that any contents in the recycle bin would be completely deleted and removed from the system. This most likely wouldn't be too catastrophic for most users but is a notable result to keep in mind.



```
U•ïXµ$V%ÖÏÓ]‰™½qvŸ▯      aÚ˙¦▯´þ_þ/N▯×õ▯ÄMõó—ç;À▯à»7´Ëèô3▯Ï▯ð&"¢È▯Šå\Y
aèž)À¼1n¢h
R]▯I▯>Ÿ▯"ö SH*ÍtéC€↑h4dÖ▯0èc±Lu¶iç¨ŒL3ødd ý'/™ç˘ÒS\nX‹ˆ˜åXy7÷▯Ö▯"ã€'%í6" §–▯-ÊFS™HG▯▯}ñÍ3ø▯ÕÜãz
Mrë‹Ô+ðõß{£%  U€Üdæ▯)]ì'ËŽW
```

Figure 9 - Visual of Encrypted File



All Encrypted Files 5256
All Decrypted Files 5256

Figure 10 - LB3Decryptor.exe after successful deployment

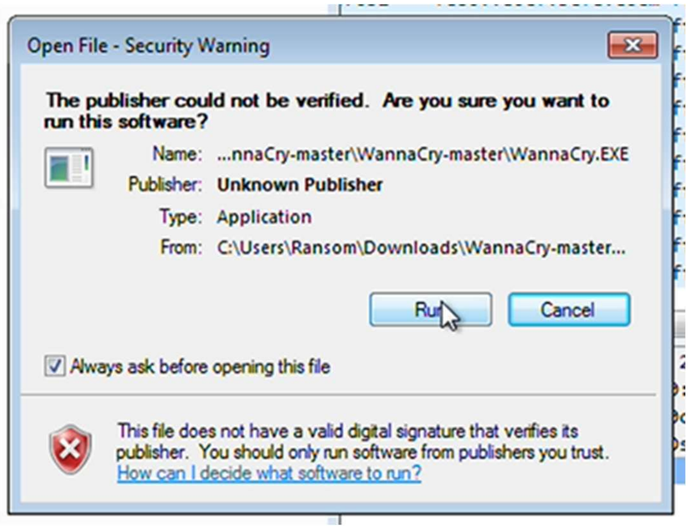
b. WANNACRY

Preparation

First, I set up a VMOracle [25] machine with a windows 7 ISO [21]. The reason to create this machine was to simulate a real scenario where the EternalBlue vulnerability still existed. Once done, we can easily start our Windows 7 machine and proceed with the steps instructed. Make sure to download the basic drivers that can be found in the machine network (to easily access the internet).

The only steps left to do would be to download the directory (.zip file) from the GitHub [24] that contains the desired Ransomware. However, due to internet explorer being obsolete/unusable, we had to find a way to browse the internet to enter into GitHub as we didn't want to download the ransomware from any of our local machines. To do so, we found out that there's a basic chromium based browser called Supermium [22], so we downloaded the files onto our local machines and then dragged them onto the VM.

Now with access to internet and a usable browser, we can access to the ransomware [24], and before executing it, we downloaded Wireshark [23] in our VMs as well to monitor the traffic. Due to the nature of ransomware, it will require some access to our network to download the software and notify the ransomware server to proceed with the attack. We also transferred some files from our local machine to the VM so we can see some encryption by the malware in action, such as school projects and even a text note with some random text on it.



The publisher could not be verified. Are you sure you want to run this software?

Name: ...nnaCry-master\WannaCry-master\WannaCry.EXE
Publisher: **Unknown Publisher**
Type: Application
From: C:\Users\Ransom\Downloads\WannaCry-master...

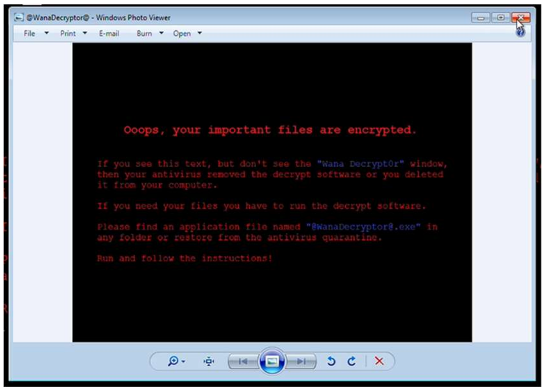
☑ Always ask before opening this file

This file does not have a valid digital signature that verifies its publisher. You should only run software from publishers you trust.
How can I decide what software to run?

Attack

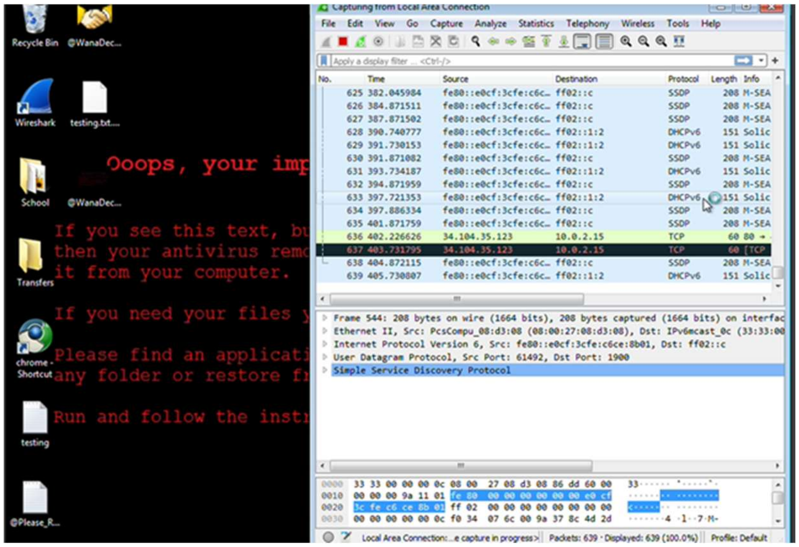
First, we extracted the folder of the ransomware that initially contained 3 files, two .txt files (LICENSE, README) and the .exe (executable ransomware). Once we clicked on it, Windows asked us if we were sure as the publisher was not verified, but obviously we proceeded.

Shortly after we noticed a change in the wallpaper and some TCP protocols in our Wireshark.



```
Ooops, your important files are encrypted.

If you see this text, but don't see the "Wana Decrypt0r" window,
then your antivirus removed the decrypt software or you deleted
it from your computer.

If you need your files you have to run the decrypt software.

Please find an application file named "@WanaDecryptor@.exe" in
any folder or restore from the antivirus quarantine.

Run and follow the instructions!
```

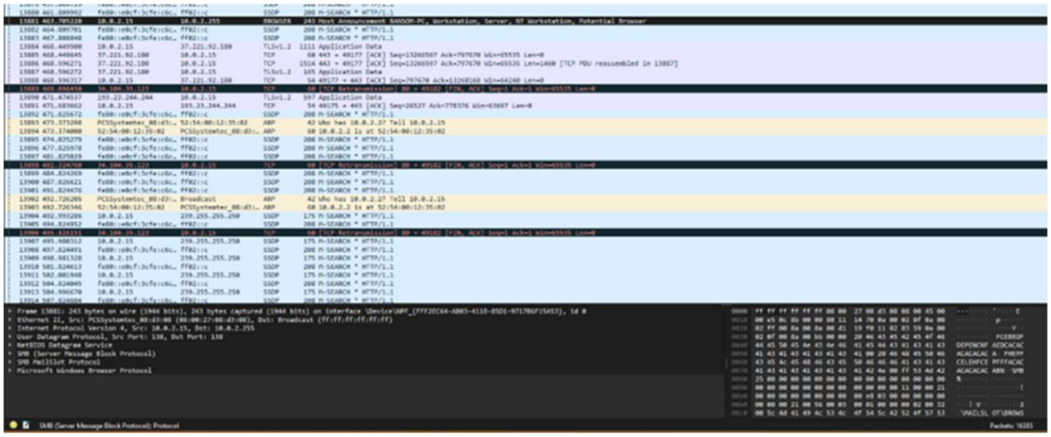
Figure 11 -Wannacry Wallpaper



```
No.   Time         Source                  Destination  Protocol Length Info
  625 382.045984   fe80::e0cf:3cfe:c6c…    ff02::c      SSDP     208 M-SEA
  626 384.871511   fe80::e0cf:3cfe:c6c…    ff02::c      SSDP     208 M-SEA
  627 387.871502   fe80::e0cf:3cfe:c6c…    ff02::c      SSDP     208 M-SEA
  628 390.740777   fe80::e0cf:3cfe:c6c…    ff02::1:2    DHCPv6   151 Solic
  629 391.730153   fe80::e0cf:3cfe:c6c…    ff02::1:2    DHCPv6   151 Solic
  630 391.871082   fe80::e0cf:3cfe:c6c…    ff02::c      SSDP     208 M-SEA
  631 393.734187   fe80::e0cf:3cfe:c6c…    ff02::1:2    DHCPv6   151 Solic
  632 394.871959   fe80::e0cf:3cfe:c6c…    ff02::c      SSDP     208 M-SEA
  633 397.721353   fe80::e0cf:3cfe:c6c…    ff02::1:2    DHCPv6   151 Solic
  634 397.886334   fe80::e0cf:3cfe:c6c…    ff02::c      SSDP     208 M-SEA
  635 401.871759   fe80::e0cf:3cfe:c6c…    ff02::c      SSDP     208 M-SEA
  636 402.226626   34.104.35.123           10.0.2.15    TCP       60 80 →
  637 403.731795   34.104.35.123           10.0.2.15    TCP       60 [TCP
  638 404.872115   fe80::e0cf:3cfe:c6c…    ff02::c      SSDP     208 M-SEA
  639 405.730807   fe80::e0cf:3cfe:c6c…    ff02::1:2    DHCPv6   151 Solic

▷ Frame 544: 208 bytes on wire (1664 bits), 208 bytes captured (1664 bits) on interfac
▷ Ethernet II, Src: PcsCompu_08:d3:08 (08:00:27:08:d3:08), Dst: IPv6mcast_0c (33:33:00
▷ Internet Protocol Version 6, Src: fe80::e0cf:3cfe:c6ce:8b01, Dst: ff02::c
▷ User Datagram Protocol, Src Port: 61492, Dst Port: 1900
▷ Simple Service Discovery Protocol

0000  33 33 00 00 00 0c 08 00  27 08 d3 08 86 dd 60 00   33······'·····`·
0010  00 00 00 9a 11 01 fe 80  00 00 00 00 00 00 e0 cf   ················
0020  3c fe c6 ce 8b 01 ff 02  00 00 00 00 00 00 00 00   <···············
0030  00 00 00 00 00 0c f0 34  07 6c 00 9a 37 8c 4d 2d   ·······4·l··7·M-

Local Area Connection: <live capture in progress>   Packets: 639 · Displayed: 639 (100.0%)   Profile: Default
```

Figure 12- Initial wireshark protocols detected

The wallpaper seems to be an automatic process of the ransomware, which downloads the image and sets it up as the wallpaper shortly after executing it. When it comes to our Wireshark monitoring, we see multiple protocols happening, mostly done through Google. However, we can identify some indicators, some traffic in SMB (IPs: 34.104.35.123 and 37.221.92.180)



```
13880 461.809992   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13881 463.705220   10.0.2.15             10.0.2.255            BROWSER   243 Host Announcement RANSOM-PC, Workstation, Server, NT Workstation, Potential Browser
13882 464.809701   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13883 467.808848   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13884 468.449500   10.0.2.15             37.221.92.180         TLSv1.2  1111 Application Data
13885 468.449645   37.221.92.180         10.0.2.15             TCP        60 443 → 49177 [ACK] Seq=13266597 Ack=797670 Win=65535 Len=0
13886 468.596271   37.221.92.180         10.0.2.15             TCP      1514 443 → 49177 [ACK] Seq=13266597 Ack=797670 Win=65535 Len=1460 [TCP PDU reassembled in 13887]
13887 468.596272   37.221.92.180         10.0.2.15             TLSv1.2   165 Application Data
13888 468.596317   10.0.2.15             37.221.92.180         TCP        54 49177 → 443 [ACK] Seq=797670 Ack=13268168 Win=64240 Len=0
13889 469.090450   34.104.35.123         10.0.2.15             TCP        60 [TCP Retransmission] 80 → 49182 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
13890 471.474537   193.23.244.244        10.0.2.15             TLSv1.2   597 Application Data
13891 471.683662   10.0.2.15             193.23.244.244        TCP        54 49175 → 443 [ACK] Seq=26527 Ack=778376 Win=63687 Len=0
13892 471.825672   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13893 473.373298   PCSSystemtec_00:d3:…  52:54:00:12:35:02     ARP        42 Who has 10.0.2.2? Tell 10.0.2.15
13894 473.374000   52:54:00:12:35:02     PCSSystemtec_00:d3:…  ARP        60 10.0.2.2 is at 52:54:00:12:35:02
13895 474.825279   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13896 477.825978   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13897 481.825829   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13898 482.224760   34.104.35.123         10.0.2.15             TCP        60 [TCP Retransmission] 80 → 49182 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
13899 484.824209   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13900 487.826621   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13901 491.824476   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13902 492.726205   PCSSystemtec_00:d3:…  Broadcast             ARP        42 Who has 10.0.2.2? Tell 10.0.2.15
13903 492.726346   52:54:00:12:35:02     PCSSystemtec_00:d3:…  ARP        60 10.0.2.2 is at 52:54:00:12:35:02
13904 492.993286   10.0.2.15             239.255.255.250       SSDP      175 M-SEARCH * HTTP/1.1
13905 494.824952   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13906 495.020151   34.104.35.123         10.0.2.15             TCP        60 [TCP Retransmission] 80 → 49182 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
13907 495.980312   10.0.2.15             239.255.255.250       SSDP      175 M-SEARCH * HTTP/1.1
13908 497.824491   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13909 498.981328   10.0.2.15             239.255.255.250       SSDP      175 M-SEARCH * HTTP/1.1
13910 501.824613   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13911 502.001948   10.0.2.15             239.255.255.250       SSDP      175 M-SEARCH * HTTP/1.1
13912 504.824045   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
13913 504.996670   10.0.2.15             239.255.255.250       SSDP      175 M-SEARCH * HTTP/1.1
13914 507.824604   fe80::e0cf:3cfe:c6c…  ff02::c               SSDP      208 M-SEARCH * HTTP/1.1
```

```
> Frame 13881: 243 bytes on wire (1944 bits), 243 bytes captured (1944 bits) on interface \Device\NPF_{FFF2EC64-AB03-4118-85D1-9717B6F15453}, id 0
> Ethernet II, Src: PCSSystemtec_00:d3:08 (08:00:27:00:d3:08), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Internet Protocol Version 4, Src: 10.0.2.15, Dst: 10.0.2.255
> User Datagram Protocol, Src Port: 138, Dst Port: 138
> NetBIOS Datagram Service
> SMB (Server Message Block Protocol)
> SMB MailSlot Protocol
> Microsoft Windows Browser Protocol
```

```
0000  ff ff ff ff ff ff 08 00  27 00 d3 08 08 00 45 00   ········'·····E·
0010  00 e5 0c 8b 00 00 80 11  14 70 0a 00 02 0f 0a 00   ·········p······
0020  02 ff 00 8a 00 8a 00 d1  19 f0 11 02 83 59 0a 00   ·············Y··
0030  02 0f 00 8a 00 bb 00 00  20 46 43 45 42 45 4f 46   ········ FCEBEOF
0040  44 45 50 45 4e 43 4e 46  41 45 44 43 41 43 41 43   DEPENCNF AEDCACAC
0050  41 43 41 43 41 43 41 43  41 00 20 46 48 45 50 46   ACACACAC A· FHEPF
0060  43 45 4c 45 48 46 43 45  50 46 46 46 41 43 41 43   CELEHFCE PFFFACAC
0070  41 43 41 43 41 43 41 43  41 42 4e 00 ff 53 4d 42   ACACACAC ABN··SMB
0080  25 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   %···············
0090  00 00 00 00 00 00 00 00  00 00 00 00 11 00 00 21   ···············!
00a0  00 00 00 00 00 00 00 00  00 e8 03 00 00 00 00 00   ················
00b0  00 00 00 21 00 56 00 03  00 01 00 00 00 02 00 32   ···!·V·········2
00c0  00 5c 4d 41 49 4c 53 4c  4f 54 5c 42 52 4f 57 53   ·\MAILSL OT\BROWS
```

SMB (Server Message Block Protocol): Protocol

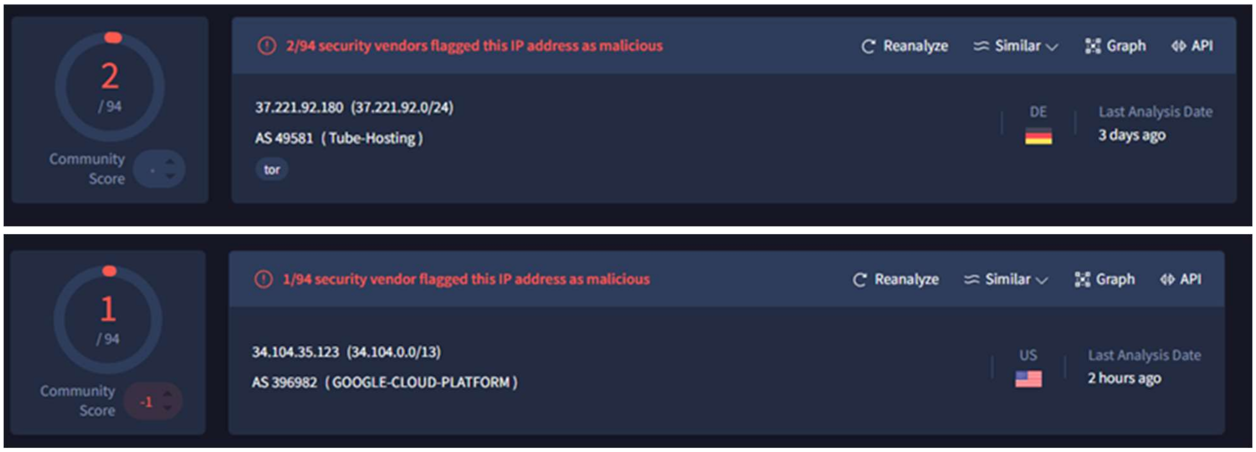
Packets: 16385

Figure 13 - Wireshark Analysis of Wannacry Traffic

Here the reports of these two IPs respectively [31]

We also found quite some traffic in the DNS with requests to the, possibly, the execution of the ransomware “ iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com “. Furthermore, some malformed SMB commands like “TreeConnectAndX” and “WriteAndX”.

Last but not least, we also found some extensive queries to TLSv1.3 and QUIC traffic,



2/94 security vendors flagged this IP address as malicious

37.221.92.180 (37.221.92.0/24)

AS 49581 (Tube-Hosting)

tor

DE

Last Analysis Date: 3 days ago

1/94 security vendor flagged this IP address as malicious

34.104.35.123 (34.104.0.0/13)

AS 396982 (GOOGLE-CLOUD-PLATFORM)

US

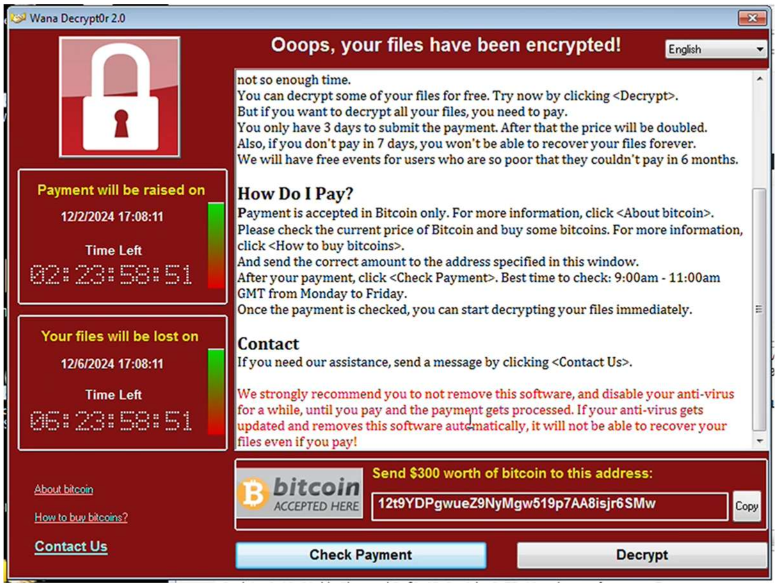
Last Analysis Date: 2 hours ago

Figure 14 - Virustotal result of Wannacry Server IP

which indicates a possible malicious activity. When it comes to “raw” data from Wireshark, these were the findings. While the report was quite large, with multiple protocols happening at the same time, it is obvious that the ransomware was communicating with a server as there’s barely any logs before the malware is executed.

Once the malware is activated, it will be prompting a payment request in Bitcoin to the bad actors so they could "decrypt" the victim's files.

The hyperlinks this window offer us are a Wikipedia link to what is Bitcoin, a google search of "how to buy bitcoin" and contact us wasn't prompting an error. Check payment will load a "payment" checker which probably would do nothing, and the decrypt might be a way of the hacker to send an activation key to "disarm" the ransomware.



Ooops, your files have been encrypted!

not so enough time.
You can decrypt some of your files for free. Try now by clicking <Decrypt>.
But if you want to decrypt all your files, you need to pay.
You only have 3 days to submit the payment. After that the price will be doubled.
Also, if you don't pay in 7 days, you won't be able to recover your files forever.
We will have free events for users who are so poor that they couldn't pay in 6 months.

How Do I Pay?
Payment is accepted in Bitcoin only. For more information, click <About bitcoin>.
Please check the current price of Bitcoin and buy some bitcoins. For more information, click <How to buy bitcoins>.
And send the correct amount to the address specified in this window.
After your payment, click <Check Payment>. Best time to check: 9:00am - 11:00am GMT from Monday to Friday.
Once the payment is checked, you can start decrypting your files immediately.

Contact
If you need our assistance, send a message by clicking <Contact Us>.

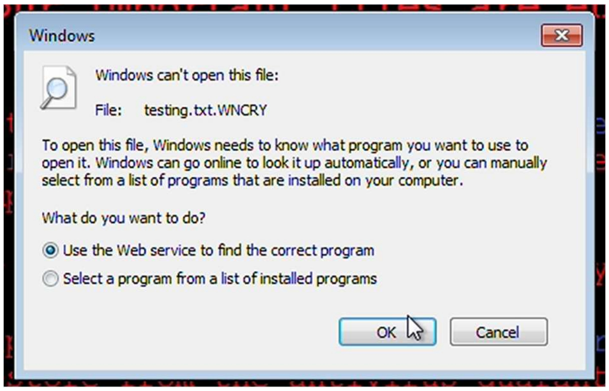
We strongly recommend you to not remove this software, and disable your anti-virus for a while, until you pay and the payment gets processed. If your anti-virus gets updated and removes this software automatically, it will not be able to recover your files even if you pay!

Send $300 worth of bitcoin to this address:
12t9YDPgwueZ9NyMgw519p7AA8isjr6SMw

Figure 15 - Wannacry Ransom Client

Once we check the files we transfer to the VM machine, we can see how images and documents were encrypted with .WNCRY.

The attempt to open any of these files will result in a error as Windows will not know how to properly open these files



Windows

Windows can't open this file:

File: testing.txt.WNCRY

To open this file, Windows needs to know what program you want to use to open it. Windows can go online to look it up automatically, or you can manually select from a list of programs that are installed on your computer.

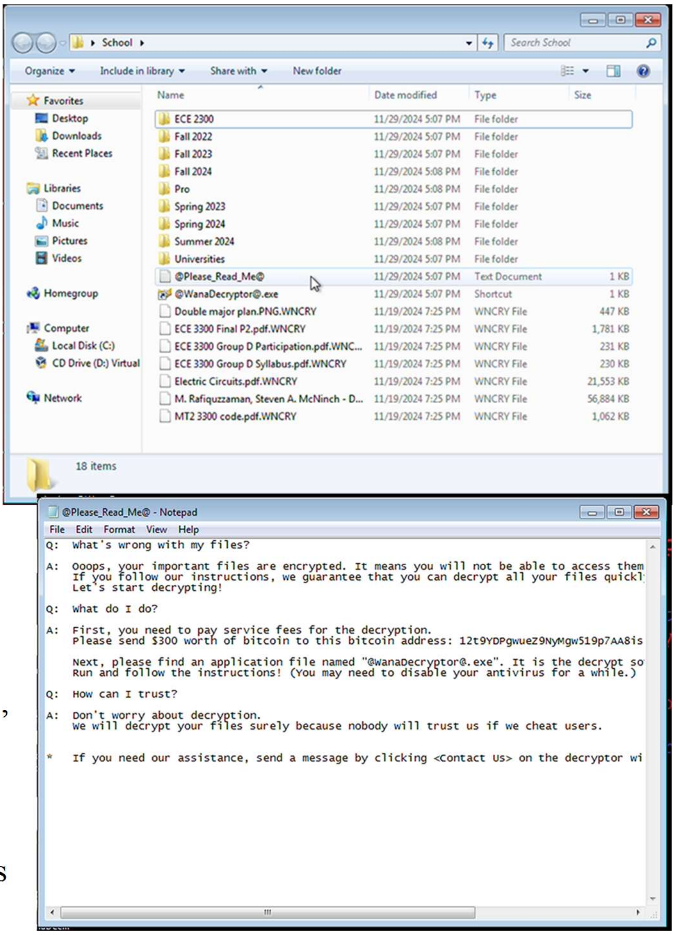
What do you want to do?

● Use the Web service to find the correct program

○ Select a program from a list of installed programs

Figure 16 - Wannacry actively encrypting

Besides of the encryption, we can also see how the creator of this malware made sure to explain to the victim "what was going on" and how to get rid of the issue. He did this by creating .txt files called "@Please_Read_Me@"



Name	Date modified	Type	Size
ECE 2300	11/29/2024 5:07 PM	File folder	
Fall 2022	11/29/2024 5:07 PM	File folder	
Fall 2023	11/29/2024 5:07 PM	File folder	
Fall 2024	11/29/2024 5:08 PM	File folder	
Pro	11/29/2024 5:08 PM	File folder	
Spring 2023	11/29/2024 5:07 PM	File folder	
Spring 2024	11/29/2024 5:07 PM	File folder	
Summer 2024	11/29/2024 5:08 PM	File folder	
Universities	11/29/2024 5:07 PM	File folder	
@Please_Read_Me@	11/29/2024 5:07 PM	Text Document	1 KB
@WanaDecryptor@.exe	11/29/2024 5:07 PM	Shortcut	1 KB
Double major plan.PNG.WNCRY	11/19/2024 7:25 PM	WNCRY File	447 KB
ECE 3300 Final P2.pdf.WNCRY	11/19/2024 7:25 PM	WNCRY File	1,781 KB
ECE 3300 Group D Participation.pdf.WNC...	11/19/2024 7:25 PM	WNCRY File	231 KB
ECE 3300 Group D Syllabus.pdf.WNCRY	11/19/2024 7:25 PM	WNCRY File	230 KB
Electric Circuits.pdf.WNCRY	11/19/2024 7:25 PM	WNCRY File	21,553 KB
M. Rafiquzzaman, Steven A. McNinch - D...	11/19/2024 7:25 PM	WNCRY File	56,884 KB
MT2 3300 code.pdf.WNCRY	11/19/2024 7:25 PM	WNCRY File	1,062 KB

18 items

@Please_Read_Me@ - Notepad

```
Q:  What's wrong with my files?

A:  Ooops, your important files are encrypted. It means you will not be able to access them
    If you follow our instructions, we guarantee that you can decrypt all your files quickl
    Let's start decrypting!

Q:  What do I do?

A:  First, you need to pay service fees for the decryption.
    Please send $300 worth of bitcoin to this bitcoin address: 12t9YDPgwueZ9NyMgw519p7AA8is

    Next, please find an application file named "@WanaDecryptor@.exe". It is the decrypt so
    Run and follow the instructions! (You may need to disable your antivirus for a while.)

Q:  How can I trust?

A:  Don't worry about decryption.
    We will decrypt your files surely because nobody will trust us if we cheat users.


*   If you need our assistance, send a message by clicking <Contact Us> on the decryptor wi
```

An interesting fact of this ransomware is continuously active until disarmed, so injecting a flash drive to the VM will cause the USB to also fall under the WannaCry encryption. Although this was a problem when I introduced the flash drive into the VM, we also noticed that not all files are encrypted, but mostly document/images/video files. We presume that the creator didn't want to attack the whole system as that would defeat the whole purpose of his attack, as that could affect files that were critical for the system to operate. Some files that were not affected that are not critical for Windows to operate were .JSON (JavaScript Object Notation) and .pcapng (Wireshark file).

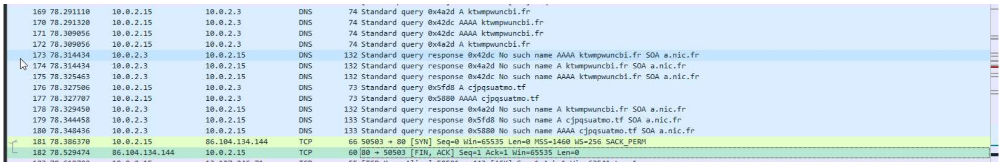
With this, we would finalize the attack to our VM by executing the WannaCry ransomware.

c. LOCKY

Unfortunately, the simulation of the Locky ransomware ended in failure. Despite multiple attempts to execute the ransomware using different Locky executables [9] obtained from GitHub, each attempt resulted in the same outcome: the ransomware failed to function as expected.

While I could see the Locky process running in the Windows Task Manager, no noticeable activity occurred on the system. Files were not encrypted, no ransom note appeared, and the typical Locky behavior did not manifest.

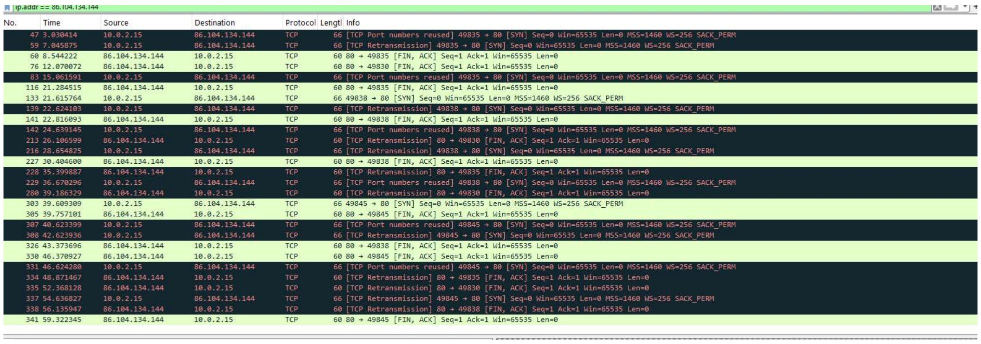
Using Wireshark to analyze network activity, I identified the likely cause of the failure. The ransomware initially attempted to communicate with its command-and-control (C2) server using a Domain Generation Algorithm (DGA) to generate a series of potential domain names for connection. This method is typically used to ensure that the malware can locate an active C2 server even if some domains are taken down. However, all connection attempts to these generated domains failed.



```
169 78.291110   10.0.2.15        10.0.2.3         DNS    74 Standard query 0x4a2d A ktwmpwuncbi.fr
170 78.291320   10.0.2.15        10.0.2.3         DNS    74 Standard query 0x42dc AAAA ktwmpwuncbi.fr
171 78.309056   10.0.2.15        10.0.2.3         DNS    74 Standard query 0x42dc AAAA ktwmpwuncbi.fr
172 78.309056   10.0.2.15        10.0.2.3         DNS    74 Standard query 0x4a2d A ktwmpwuncbi.fr
173 78.314434   10.0.2.3         10.0.2.15        DNS   132 Standard query response 0x42dc No such name AAAA ktwmpwuncbi.fr SOA a.nic.fr
174 78.314434   10.0.2.3         10.0.2.15        DNS   132 Standard query response 0x4a2d No such name A ktwmpwuncbi.fr SOA a.nic.fr
175 78.325463   10.0.2.3         10.0.2.15        DNS   132 Standard query response 0x42dc No such name AAAA ktwmpwuncbi.fr SOA a.nic.fr
176 78.327506   10.0.2.15        10.0.2.3         DNS    73 Standard query 0x5fd8 A cjpqsuatmo.tf
177 78.327707   10.0.2.15        10.0.2.3         DNS    73 Standard query 0x5880 AAAA cjpqsuatmo.tf
178 78.329450   10.0.2.3         10.0.2.15        DNS   132 Standard query response 0x4a2d No such name A ktwmpwuncbi.fr SOA a.nic.fr
179 78.344458   10.0.2.3         10.0.2.15        DNS   133 Standard query response 0x5fd8 No such name A cjpqsuatmo.tf SOA a.nic.fr
180 78.348436   10.0.2.3         10.0.2.15        DNS   133 Standard query response 0x5880 No such name AAAA cjpqsuatmo.tf SOA a.nic.fr
181 78.386370   10.0.2.15        86.104.134.144   TCP    66 50503 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
182 78.529474   86.104.134.144   10.0.2.15        TCP    60 80 → 50503 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
```

Figure 17 – Locky attempting to reach certain DNS

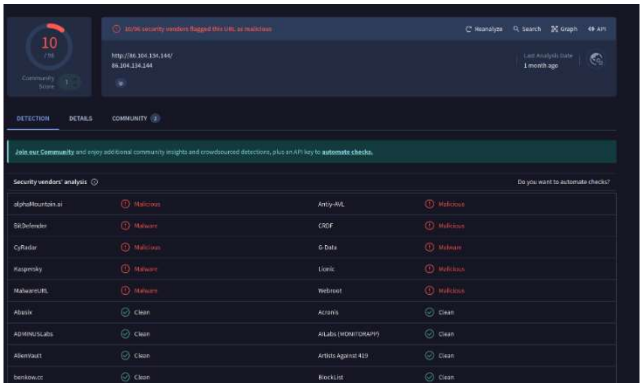
Following the failed DGA attempts, the ransomware switched to using its hard-coded C2 server address. However, these connection attempts also failed. Wireshark logs showed repeated SYN and FIN, ACK packets being sent back and forth, indicating that the server was attempting to close the connection every time the ransomware to open a connection.



```
ip.addr == 86.104.134.144
No.   Time        Source           Destination      Protocol Lengtl Info
 47 3.030414    10.0.2.15        86.104.134.144   TCP    66 [TCP Port numbers reused] 49835 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
 59 7.045875    10.0.2.15        86.104.134.144   TCP    66 [TCP Retransmission] 49835 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
 60 8.544222    86.104.134.144   10.0.2.15        TCP    60 80 → 49835 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
 76 12.070072   86.104.134.144   10.0.2.15        TCP    60 80 → 49830 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
 83 15.061591   10.0.2.15        86.104.134.144   TCP    66 [TCP Port numbers reused] 49835 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
116 21.284515   86.104.134.144   10.0.2.15        TCP    60 80 → 49835 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
133 21.615764   10.0.2.15        86.104.134.144   TCP    66 49838 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
139 22.624103   10.0.2.15        86.104.134.144   TCP    66 [TCP Retransmission] 49838 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
141 22.816093   86.104.134.144   10.0.2.15        TCP    60 80 → 49838 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
142 24.639145   10.0.2.15        86.104.134.144   TCP    66 [TCP Port numbers reused] 49838 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
213 26.106599   86.104.134.144   10.0.2.15        TCP    60 [TCP Retransmission] 80 → 49830 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
216 28.654825   10.0.2.15        86.104.134.144   TCP    66 [TCP Retransmission] 49838 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
227 30.404600   86.104.134.144   10.0.2.15        TCP    60 80 → 49838 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
228 35.399887   86.104.134.144   10.0.2.15        TCP    60 [TCP Retransmission] 80 → 49835 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
229 36.670296   10.0.2.15        86.104.134.144   TCP    66 [TCP Port numbers reused] 49838 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
280 39.186329   86.104.134.144   10.0.2.15        TCP    60 [TCP Retransmission] 80 → 49830 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
303 39.609309   10.0.2.15        86.104.134.144   TCP    66 49845 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
305 39.757101   86.104.134.144   10.0.2.15        TCP    60 80 → 49845 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
307 40.623399   10.0.2.15        86.104.134.144   TCP    66 [TCP Port numbers reused] 49845 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
308 42.623936   10.0.2.15        86.104.134.144   TCP    66 [TCP Retransmission] 49845 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
326 43.373696   86.104.134.144   10.0.2.15        TCP    60 80 → 49838 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
330 46.370927   86.104.134.144   10.0.2.15        TCP    60 80 → 49845 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
331 46.624280   10.0.2.15        86.104.134.144   TCP    66 [TCP Port numbers reused] 49845 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
334 48.871467   86.104.134.144   10.0.2.15        TCP    60 [TCP Retransmission] 80 → 49835 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
335 52.368128   86.104.134.144   10.0.2.15        TCP    60 [TCP Retransmission] 80 → 49830 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
337 54.636827   10.0.2.15        86.104.134.144   TCP    66 [TCP Retransmission] 49845 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=256 SACK_PERM
338 56.135947   86.104.134.144   10.0.2.15        TCP    60 [TCP Retransmission] 80 → 49838 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
341 59.322345   86.104.134.144   10.0.2.15        TCP    60 80 → 49845 [FIN, ACK] Seq=1 Ack=1 Win=65535 Len=0
```

Figure 18 – Locky sending SYN, C2 Server sending FIN, ACK

To further investigate, I looked up the hard-coded IP address on VirusTotal, a widely used malware analysis service. The results confirmed that the IP address was flagged as malicious by multiple security vendors, validating its association with ransomware activities. Additionally, I cross-referenced this IP address with a list of well-known Locky C2 server addresses [11], and it was indeed included in that list. This confirmation reinforced the conclusion that the failure was not due to an incorrect or unrelated IP but likely because the C2 server was no longer active or accessible.



10
/96
Community Score 1

10/96 security vendors flagged this URL as malicious

Reanalyze Search Graph API

http://86.104.134.144/
86.104.134.144

Last Analysis Date
1 month ago

DETECTION DETAILS COMMUNITY 2

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

Security vendors' analysis

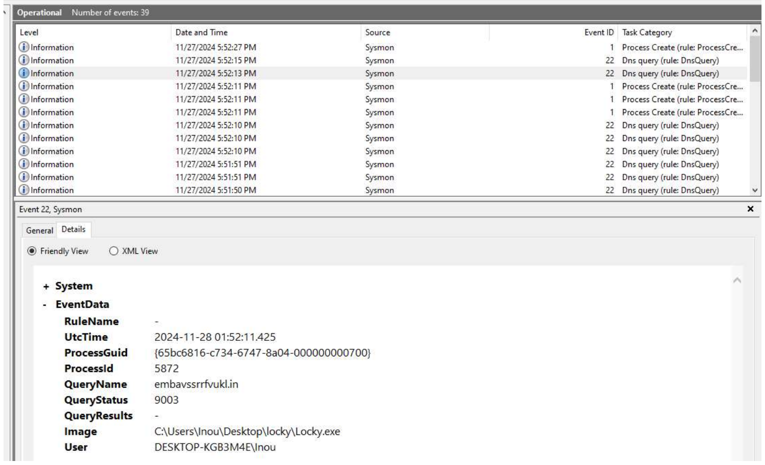
Do you want to automate checks?

alphaMountain.ai	Malicious	Antiy-AVL	Malicious
BitDefender	Malware	CRDF	Malicious
CyRadar	Malicious	G-Data	Malware
Kaspersky	Malware	Lionic	Malicious
MalwareURL	Malware	Webroot	Malicious
Abusix	Clean	Acronis	Clean
ADMINUSLabs	Clean	AILabs (MONITORAPP)	Clean
AlienVault	Clean	Artists Against 419	Clean
benkow.cc	Clean	BlockList	Clean

Figure 19 – VirusTotal showing that the IP is flagged malicious by several vendors

To determine if Locky performed any other actions during its execution, I used Sysmon (System Monitor) to track its behavior at the system level. Sysmon logs revealed that the ransomware primarily created DNS queries during its runtime, corresponding to its attempts to connect to the DGA-generated domains and hard-coded C2 server.

Beyond these DNS queries, Sysmon showed no further significant activity, such as file modifications or registry changes, confirming that Locky did not progress to its encryption phase due to the failed communication with its C2 infrastructure.



Operational Number of events: 39

Level	Date and Time	Source	Event ID	Task Category
Information	11/27/2024 5:52:27 PM	Sysmon	1	Process Create (rule: ProcessCre...
Information	11/27/2024 5:52:15 PM	Sysmon	22	Dns query (rule: DnsQuery)
Information	11/27/2024 5:52:13 PM	Sysmon	22	Dns query (rule: DnsQuery)
Information	11/27/2024 5:52:11 PM	Sysmon	1	Process Create (rule: ProcessCre...
Information	11/27/2024 5:52:11 PM	Sysmon	1	Process Create (rule: ProcessCre...
Information	11/27/2024 5:52:11 PM	Sysmon	1	Process Create (rule: ProcessCre...
Information	11/27/2024 5:52:10 PM	Sysmon	22	Dns query (rule: DnsQuery)
Information	11/27/2024 5:52:10 PM	Sysmon	22	Dns query (rule: DnsQuery)
Information	11/27/2024 5:52:10 PM	Sysmon	22	Dns query (rule: DnsQuery)
Information	11/27/2024 5:51:51 PM	Sysmon	22	Dns query (rule: DnsQuery)
Information	11/27/2024 5:51:51 PM	Sysmon	22	Dns query (rule: DnsQuery)
Information	11/27/2024 5:51:50 PM	Sysmon	22	Dns query (rule: DnsQuery)

Event 22, Sysmon

General Details

● Friendly View ○ XML View

```
+ System
- EventData
    RuleName       -
    UtcTime        2024-11-28 01:52:11.425
    ProcessGuid    {65bc6816-c734-6747-8a04-000000000700}
    ProcessId      5872
    QueryName      embavssrrfvukl.in
    QueryStatus    9003
    QueryResults   -
    Image          C:\Users\Inou\Desktop\locky\Locky.exe
    User           DESKTOP-KGB3M4E\Inou
```

Figure 20 – Sysmon showing only the Locky process creation and DNS query

This failure to establish communication with the C2 server prevented Locky from obtaining the necessary encryption key, which is a critical step in its operation. As a result, the ransomware did not proceed to encrypt files or initiate the expected ransomware behavior.

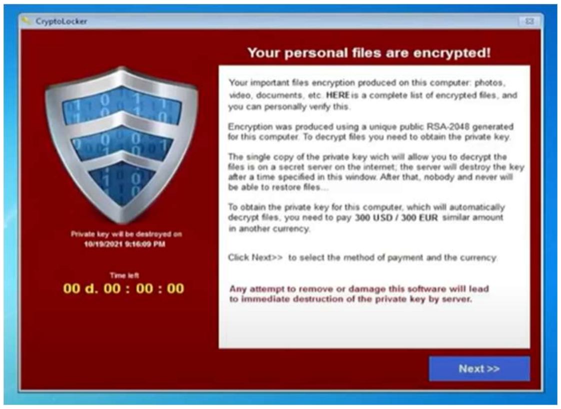
The simulation revealed the dependency of Locky ransomware on active C2 servers for its functionality, highlighting how disrupting these servers can effectively neutralize such threats. However, the inability to replicate its behavior also limited the scope of this analysis, leaving certain aspects of Locky's encryption process unexplored.

d. CRYPTOLOCKER

After simulating CryptoLocker on a Windows 7 system within a secure virtual machine (VM), I was surprised by the speed at which the ransomware operated. As soon as the files were executed, they hid themselves and within seconds of execution, almost all non-executable files were encrypted, leaving only .exe files untouched.

At first before executing it, the two files at the center of the screen are identified as the CryptoLocker files. The other files retain their original, recognizable names, making it easy to distinguish their purpose. Once executed, the CryptoLocker files are no longer visible, as they have been hidden, while all non-executable files on the left have had their names altered, reflecting the impact of the ransomware.

The ransomware subsequently displayed a window with the foreboding message, "Your personal files are encrypted," along with a demand for ransom. This screen is shown in Figure 21.



CryptoLocker

Your personal files are encrypted!

Your important files encryption produced on this computer: photos, video, documents, etc. **HERE** is a complete list of encrypted files, and you can personally verify this.

Encryption was produced using a unique public RSA-2048 generated for this computer. To decrypt files you need to obtain the private key.

The single copy of the private key wich will allow you to decrypt the files is on a secret server on the internet; the server will destroy the key after a time specified in this window. After that, nobody and never will be able to restore files...

To obtain the private key for this computer, which will automatically decrypt files, you need to pay **300 USD / 300 EUR** similar amount in another currency.

Click Next>> to select the method of payment and the currency.

Any attempt to remove or damage this software will lead to immediate destruction of the private key by server.

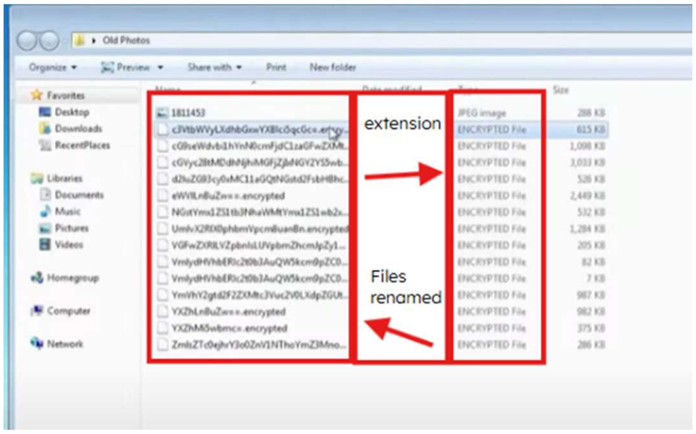
Private key will be destroyed on
10/19/2021 9:16:09 PM

Time left
00 d. 00 : 00 : 00

Figure 21: the Ransom pop-up [27]

Observing the malware in action was both fascinating and unsettling. The process was seamless and efficient, as it identified and encrypted target files across directories with alarming speed. The ransom demand appeared immediately after encryption, showcasing the malware's objective to create urgency and panic in its victims. As shown in Figure 22, the file manager reveals that the affected files have been renamed with seemingly random, nonsensical characters and assigned a new extension labeled as "ENCRYPTED File."

Examining the Task Manager, as shown in Figure 23, reveals four active processes associated with CryptoLocker. These include two instances labeled "1002.exe" and two additional processes named "IE8777A6DA…". Notably, attempting to forcefully terminate any of these processes proves ineffective, as they automatically restart and resume their operations seamlessly, demonstrating the ransomware's persistence and resilience.



Name	Type	Size
1811453	JPEG image	288 KB
c3VtbWVyLXdhbGwwYXBlcl5qcGcx.encry...	ENCRYPTED File	615 KB
cG9seWdvbi1hYnN0cmFjdC1zaGFwZXMt...	ENCRYPTED File	1,098 KB
cGVyc28tMDdhNjhiMGFjZjBkNGY2YS5wb...	ENCRYPTED File	3,033 KB
d2luZG93cy0xMC11aGQtNGstd2FsbHBhc...	ENCRYPTED File	526 KB
eWVlLnBuZw==.encrypted	ENCRYPTED File	2,449 KB
NGstYmx1ZS1tb3NhaWMtYmx1ZS1wb2x...	ENCRYPTED File	532 KB
UmlvX2RlXDphbmVpcmBuanBn.encrypted	ENCRYPTED File	1,284 KB
VGFwZXRlLVZpbnlsLUVpbmZhcmJpZy1...	ENCRYPTED File	205 KB
VmlydHVhbERlc2t0b3AuQW5kcm9pZC0...	ENCRYPTED File	82 KB
VmlydHVhbERlc2t0b3AuQW5kcm9pZC0...	ENCRYPTED File	7 KB
YmVhY2gtd2F2ZXMtc3Vuc2V0LXdpZGUt...	ENCRYPTED File	987 KB
YXZhLnBuZw==.encrypted	ENCRYPTED File	982 KB
YXZhMi5wbmc=.encrypted	ENCRYPTED File	375 KB
ZmlsZTc0ejhvY3o0ZnV1NThoYmZ3Mno...	ENCRYPTED File	286 KB

Figure 22: File manager

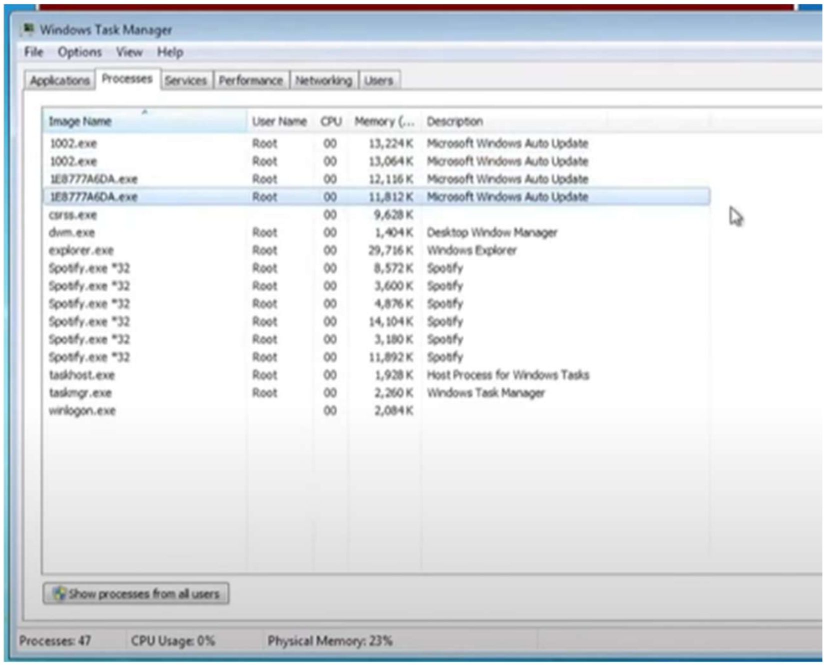

Windows Task Manager
File Options View Help
Applications Processes Services Performance Networking Users
Show processes from all users

Image Name	User Name	CPU	Memory (...	Description
1002.exe	Root	00	13,224 K	Microsoft Windows Auto Update
1002.exe	Root	00	13,064 K	Microsoft Windows Auto Update
1E8777A6DA.exe	Root	00	12,116 K	Microsoft Windows Auto Update
1E8777A6DA.exe	Root	00	11,812 K	Microsoft Windows Auto Update
csrss.exe		00	9,628 K	
dwm.exe	Root	00	1,404 K	Desktop Window Manager
explorer.exe	Root	00	29,716 K	Windows Explorer
Spotify.exe *32	Root	00	8,572 K	Spotify
Spotify.exe *32	Root	00	3,600 K	Spotify
Spotify.exe *32	Root	00	4,876 K	Spotify
Spotify.exe *32	Root	00	14,104 K	Spotify
Spotify.exe *32	Root	00	3,180 K	Spotify
Spotify.exe *32	Root	00	11,892 K	Spotify
taskhost.exe	Root	00	1,928 K	Host Process for Windows Tasks
taskmgr.exe	Root	00	2,260 K	Windows Task Manager
winlogon.exe		00	2,084 K	

Processes: 47 CPU Usage: 0% Physical Memory: 23%

Figure 23: Task manager [27]

The main three attributes I observed in CryptoLocker ransomware were its speed of encryption, target selection, and psychological impact. First, the ransomware demonstrated a highly efficient encryption algorithm, completing the process on a variety of file types in just seconds. This efficiency suggests a well-optimized mechanism aimed at maximizing impact before detection or intervention. Second, CryptoLocker specifically targeted non-executable files, leaving system-critical .exe files untouched. This strategy ensured the host system remained operational, allowing the victim to access the ransom note and potentially comply with the ransom demand. Lastly, the ransom note was designed to induce a sense of urgency and fear. By threatening the loss of personal files, the ransomware aimed to pressure victims into paying the ransom, often through cryptocurrency, which helped maintain the attacker's anonymity.

6. CHALLENGES

a. LOCKBIT

There were few problems we ran through along the way, mostly coming from the fact that LockBit was simulated and lacking internet access. The first challenge was that LockBit would not run properly when trying to run the builder.bat, this was fixed by resetting the virtual machine and downloading LockBit from a different source.

Unfortunately, before fixing the first problem, it was assumed the problem had to do with LockBit being able to detect that it was in a virtual machine and so following a guide that detailed a method to better hide the virtual machine using a "VmwareHardenedLoader" that was provided on GitHub but upon downloading and launching the executable was greeted with windows security malware warnings. Upon inspection on the GitHub just 3 days prior there had been a comment detailing that the file was infected with Trojan:Win64/CryptInject, which matched with what our windows security identified it as. Fortunately the files were successfully quarantined by windows security but we loaded a restore point made on windows prior to testing just to be safe.

During testing the only challenge faced, which will elaborated on in results, was LockBit did not allow Wireshark to run through its deployment meaning we couldn't sniff any potential malicious telemetry attempting to be sent or received, of course without the internet connection it would not be able to connect regardless.

b. WANNACRY

The WannaCry simulation presented several challenges, particularly in creating a realistic yet controlled environment to study its behavior. Setting up vulnerable systems required using outdated operating systems such as Windows 7 or XP, which are no longer officially supported and difficult to source. Ensuring these systems worked effectively within a virtual machine added complexity to the process. A significant challenge involved containing the ransomware safely within the virtual environment. Network isolation configurations in VMOracle were crucial to prevent the malware from escaping and potentially infecting external systems. Acquiring a functional WannaCry sample was another hurdle, as many available samples were either inactive or tampered with, requiring extensive testing to find a reliable version.

Additionally, the absence of WannaCry's original Command-and-Control servers limited the ability to analyze its network communication. While tools like Wireshark captured local traffic, the lack of external telemetry meant some aspects of WannaCry's behavior, such as its interactions with remote servers, could not be fully observed. Simulating WannaCry's worm-like propagation was another challenge, as it required multiple interconnected vulnerable systems, which were difficult to replicate in a virtual setup. Observing its spread via EternalBlue remained limited in this environment. Analyzing the impact of WannaCry's encryption posed risks as well, necessitating the careful transfer of files into the virtual machine while ensuring data isolation.

Finally, the tools used for monitoring and analysis had limitations. While Wireshark provided valuable insights, it could not fully capture deeper ransomware behaviors due to the restricted scope of the simulation. These challenges underscored the complexity of studying ransomware like WannaCry in a safe yet meaningful manner, highlighting the balance between realism and security in cybersecurity research.

c. LOCKY

The Locky ransomware simulation faced several challenges, primarily due to technical, environmental, and operational factors. The most significant issue was the inaccessibility of Command-and-Control (C2) servers. Locky's functionality relies on communication with active C2 infrastructure, but all connection attempts failed, likely due to domain takedowns, IP blocking, or deactivation of the infrastructure over time. Additionally, the ransomware sample may have been outdated, further limiting its functionality.

Compatibility issues also arose with the operating system. While the simulation initially used a Windows 10 VM, Locky may have been designed for older versions, like Windows 7, which have different vulnerabilities. Even with security features disabled, modern operating systems may include updates that interfere with malware behavior. The VM environment itself introduced further challenges, as it lacked the realism of a typical user setup, and Locky might have detected the sandbox environment, suppressing its actions.

The tools used, such as Wireshark and Sysmon, provided useful insights but were limited in capturing encrypted or covert activities. Additionally, the reliance on malware samples from repositories raised concerns about the integrity of the executables, as they might have been tampered with or non-functional. Finally, Locky's dependency on specific conditions, such as file structures or user actions, might not have been fully replicated in the VM. Addressing these challenges in future simulations—by using updated samples, more realistic environments, and alternative configurations—could enhance the reliability and depth of the analysis.

d. CRYPTOLOCKER

The process of researching CryptoLocker malware presented several challenges. One of the primary issues was finding a sample of the malware for experimentation. Public access to CryptoLocker had been restricted following its takedown by the FBI, which hindered its availability for research purposes. To address this, I shifted my focus from acquiring a direct sample to utilizing existing research and online demonstrations. Another challenge was identifying a video resource that provided sufficient detail to understand the malware's behavior and mechanics. Many available resources were either too superficial or lacked the necessary technical depth. Fortunately, I found a video by Marc Drouinaud Jr., a cybersecurity expert with a Master's degree from Florida International University, whose explanation offered the clarity and depth needed to effectively grasp the ransomware's operation.

7. CONCLUSION

The research and simulations of LockBit, WannaCry, Locky, and CryptoLocker highlight the increasing sophistication and adaptability of ransomware attacks. Each ransomware family showcases unique tactics and impacts, underscoring the evolving nature of this threat.

LockBit's Ransomware-as-a-Service (RaaS) model demonstrates how ransomware has become accessible to even non-technical attackers, with advanced features like double extortion and automated encryption. WannaCry's global spread via the EternalBlue exploit emphasized the critical need for timely patching and robust cybersecurity practices. Locky highlighted the effectiveness of social engineering and phishing in spreading ransomware, while CryptoLocker introduced advanced encryption techniques, setting a precedent for modern ransomware campaigns.

LockBit demonstrated an advanced automated encryption process, relying on tools like key generators and configuration files, which create potential opportunities for disruption. WannaCry, on the other hand, leveraged the EternalBlue exploit to propagate autonomously, bypassing traditional C2 dependencies and highlighting the critical need for timely software patches. Locky's reliance on C2 servers for encryption keys made it ineffective when communication with these servers was disrupted, underscoring the importance of neutralizing ransomware infrastructure. Finally, CryptoLocker's speed and efficiency emphasized the need for proactive defenses to counter its rapid encryption capabilities. These findings reinforce the value of network segmentation, continuous monitoring, and isolation strategies to mitigate the spread and impact of ransomware across various infrastructures.

This research highlights the need for robust cybersecurity measures, including patch management, employee training, and layered defenses. Organizations must adopt proactive strategies to stay ahead of evolving threats. By combining technical insights and practical simulations, this study contributes to the ongoing efforts to combat ransomware and protect critical systems from malicious actor**s.**

8. REFERENCES

[1] D. Behling, “LockBit 3.0 Ransomware Unlocked,” *VMware Security Blog*, Oct. 15, 2022. https://blogs.vmware.com/security/2022/10/lockbit-3-0-also-known-as-lockbit-black.html

[2] A. Sorensen, “LockBit,” *GitHub*, Oct. 31, 2023. https://github.com/Tennessene/LockBit

[3] Siam Alam, “LockBit 3.0 - One of the most active ransomwares right now,” *YouTube*, Apr. 11, 2024. https://www.youtube.com/watch?v=sS6_1r5hxi8 (accessed Dec. 04, 2024).

[4] CISA, “Understanding Ransomware Threat Actors: LockBit | CISA,” *www.cisa.gov*, Jun. 14, 2023. https://www.cisa.gov/news-events/cybersecurity-advisories/aa23-165a

[5] “Lockbit,” *Wikipedia*, Nov. 10, 2023. https://en.wikipedia.org/wiki/Lockbit

[6] “An In-Depth Look at Ransomware Gang, LockBit 3.0,” *Avertium.com*, 2024. https://www.avertium.com/resources/threat-reports/in-depth-look-at-ransomware-gang-lockbit-3.0 (accessed Dec. 04, 2024).

[7] KnowBe4, “Locky Ransomware | KnowBe4,” *Knowbe4.com*, 2016. https://www.knowbe4.com/locky-ransomware

[8] B. Fiore, K. Ha, L. Huynh, J. Falcon, Robinson Vendiola, and Y. Li, “Security Analysis of Ransomware: A Deep Dive into WannaCry and Locky,” Mar. 2023, doi: https://doi.org/10.1109/ccwc57344.2023.10099114.

[9] L. Abrams, “The Locky Ransomware Encrypts Local Files and Unmapped Network Shares,” BleepingComputer, Feb. 16, 2016. https://www.bleepingcomputer.com/news/security/the-locky-ransomware-encrypts-local-files-and-unmapped-network-shares/

[10] “Comprehensive Guide to Locky Ransomware Attacks,” *Tata Communications*, Dec. 14, 2023. https://www.tatacommunications.com/knowledge-base/guide-to-locky-ransomware/

[11] “A closer look at the Locky ransomware,” *Avast.com*, 2016. https://blog.avast.com/a-closer-look-at-the-locky-ransomware

[12] J. Salvio, “Locky Strikes Another Blow, Diablo6 Variant Starts Spreading Through Spam,” *Fortinet Blog*, Aug. 14, 2017. https://www.fortinet.com/blog/threat-research/locky-strikes-another-blow-diablo6-variant-starts-spreading-through-spam

[13] BleepingComputer.com, “Locky Ransomware Information, Help Guide, and FAQ,” *BleepingComputer*, May 09, 2016. https://www.bleepingcomputer.com/virus-removal/locky-ransomware-information-help?utm_source=chatgpt.com#locky

[14] ytisf, “ytisf/theZoo,” *GitHub*, Apr. 14, 2019. https://github.com/ytisf/theZoo

[15] Cisco-Talos, “GitHub - Cisco-Talos/locky,” *GitHub*, 2016. https://github.com/Cisco-Talos/locky

[16] Microsoft Security Blog, "WannaCrypt ransomware worm targets out-of-date systems," 2017. https://www.microsoft.com/en-us/security/blog/2017/05/12/wannacrypt-ransomware-worm-targets-out-of-date-systems/

[17] National Audit Office, "Investigation: WannaCry cyber attack and the NHS," 2018. https://www.nao.org.uk/reports/investigation-wannacry-cyber-attack-and-the-nhs/

[18] Cybersecurity and Infrastructure Security Agency, "Indicators Associated With WannaCry Ransomware," 2017. https://www.cisa.gov/news-events/alerts/2017/05/12/indicators-associated-wannacry-ransomware

[19]National Cybersecurity and Communications Integration Center, “What is WANNACRY/WANACRYPTOR?,” 2017. https://www.cisa.gov/sites/default/files/FactSheets/NCCIC%20ICS_FactSheet_WannaCry_Ransomware_S508C.pdf

[20] Symantec, "What you need to know about the WannaCry Ransomware," 2017.

https://www.symantec.com/connect/blogs/what-you-need-know-about-wannacry-ransomware
[21] Internet Archive, "Windows 7 ISO,": https://archive.org/details/windows-7-iso
[22] Supermium GitHub Repository, "Supermium Releases,"
https://github.com/win32ss/supermium/releases
[23] Wireshark, "Wireshark All Versions for Win32,"
https://www.wireshark.org/download/win32/all-versions
[24] GitHub Repository, "WannaCry by SomeCodingCoolGuy,"
https://github.com/SomeCodingCoolGuy/WannaCry
[25] Oracle, "VirtualBox Downloads,"
https://www.oracle.com/virtualization/technologies/vm/downloads/virtualbox-downloads.html
[26] Heimdal Security, "Locky Ransomware 101: What You Need to Know," Heimdal Security Blog, May 4, 2017. [Online]. Available: https://heimdalsecurity.com/blog/locky-ransomware-101/.
[27]Marc Drouinaud Jr, "The CryptoLocker Ransomware - How does is work? What does it do? - Part 1 - Marc Drouinaud Jr," YouTube, Dec. 20, 2021.
https://www.youtube.com/watch?v=erTj7syExEg
[29]M. Buckbee, "CryptoLocker: Everything You Need to Know," www.varonis.com, May 26, 2023. https://www.varonis.com/blog/cryptolocker
[30]Kaspersky, "What is the Cryptolocker Virus?," /, Jun. 05, 2018.
https://usa.kaspersky.com/resource-center/definitions/cryptolocker?srsltid=AfmBOoqby8aYZHnGx3Cq2Z46IewZI6PrnjuNrFlhSdv_cWPFa_KGL1C-

[31] "VirusTotal," *www.virustotal.com*. https://virustotal.com