infinitum IT
Power of integrated Security

Eva0x00

Ransomware

www.infinitumit.com.tr

infinitumitlabs

CONTENTS

Eva0x00 Ransomware and What You Need to Know

What is Eva0x00 Ransomware?

Eva0x00 Ransomware is a data encryptor known for its low virus detection rate. By default, it encrypts file types such as **".pdf," ".png," ".jpg," ".jpeg," ".mp3," ".mp4," ".txt," and ".exe,"** but it also allows users to optionally add different file types.

Eva0x00 Ransomware encrypts files from within, which means that even if the file extension is restored to normal, the content remains encrypted, rendering the victim's data inaccessible.

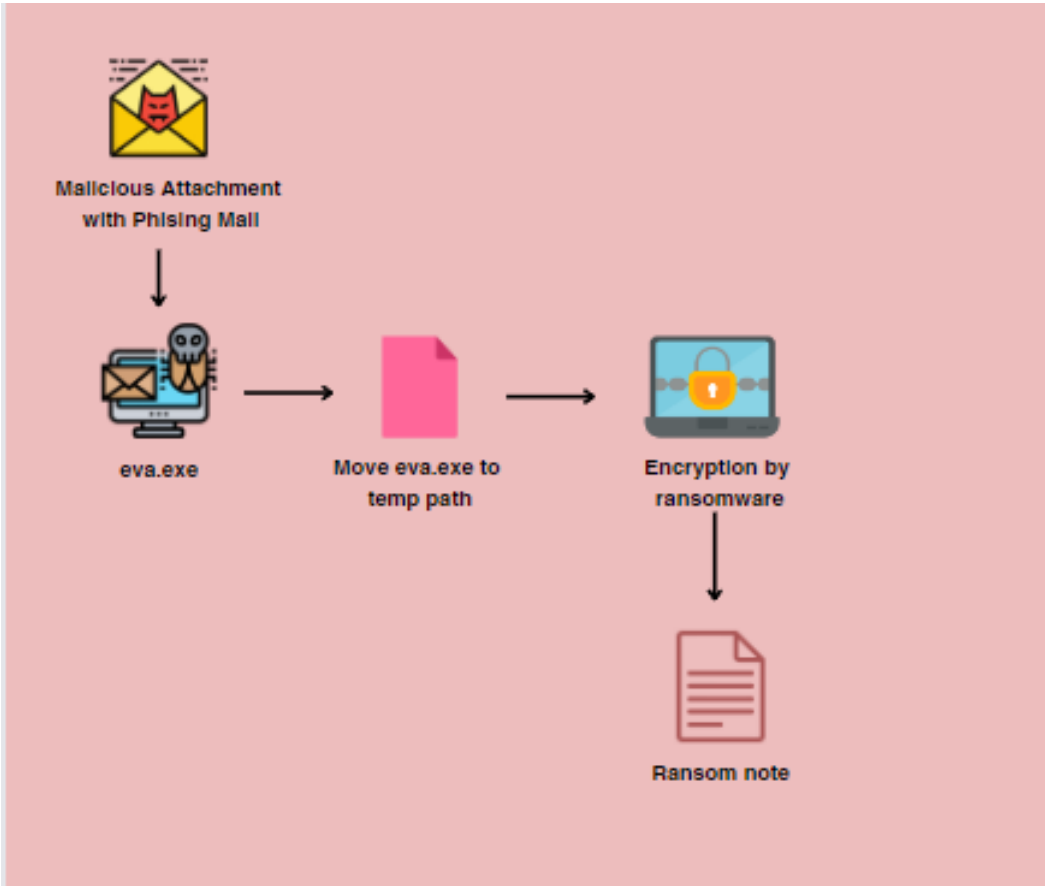
The ransomware encrypts data with the default format **".encrypt_by_eva0x00"** but users have the option to change this extension according to their preference. Since the malware encrypts data in parallel, shortly after the malware is executed, all data for the specified file types is encrypted.

After encrypting all data, **Eva0x00 Ransomware** creates a readable file on the victim's machine. This file can be customized by the ransomware user to serve their purpose.

It's important to note that **Eva0x00 Ransomware** does not run as an autorun process, is not present in the Registry Editor (Regedit) section, and does not restart itself. Therefore, if the infected system is shut down and restarted after the malware has executed, the encryption process does not continue.

Eva0x00 Ransomware has a notably **low detection rate by antivirus software**, making it challenging to detect and mitigate.

Infection Chain



Eva0x00 Ransomware Overview

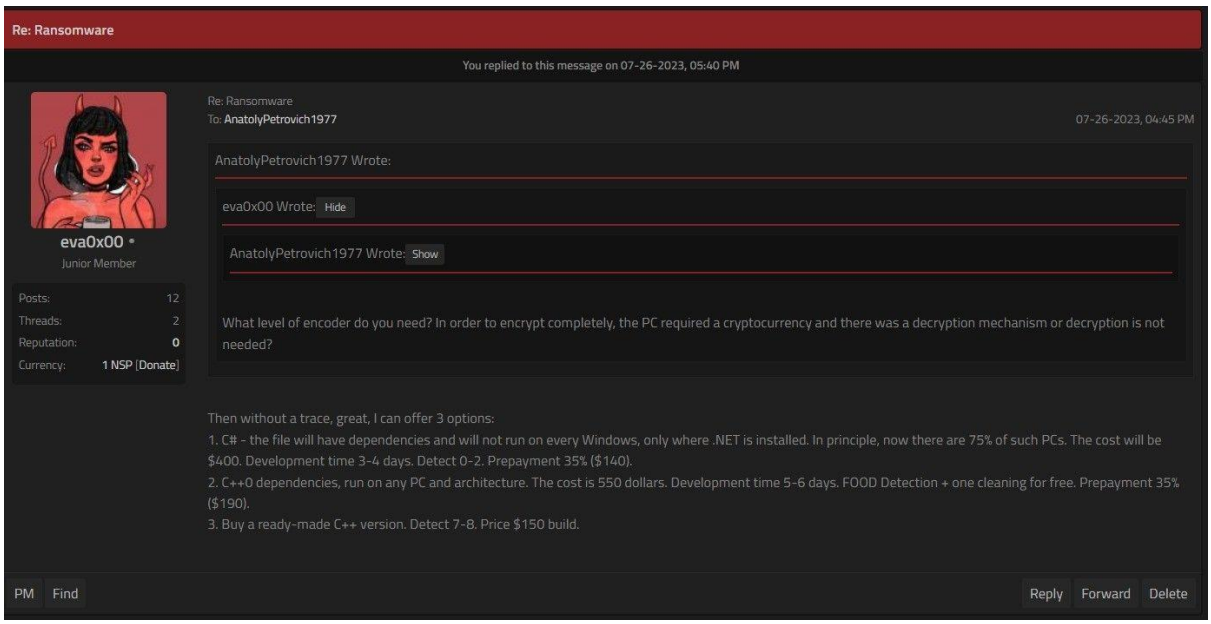


Figure 1- Dark Web forum info

Eva0x00 is a user on a dark web forum. He is selling custom made ransomware virus to the forum users. The ransomware does not require an internet connection. It is an offline ransomware and does not need any network connectivity. It's able to bypass many antivirus software used by computer users.

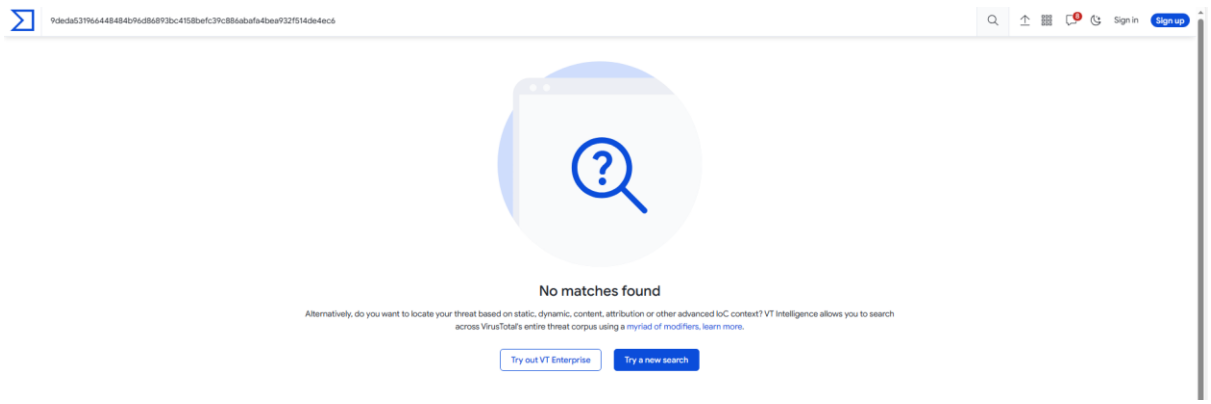


Figure 2- Virus Total info

The hash information of the Eva0x00 Ransomware **does not appear in the VirusTotal results**. This situation indicates that the software hasn't been scanned by anyone. For viruses, this is often observed in newly released products.

Scan result:	This file was detected by [4 / 40] engine(s)
File name:	rramsom.exe
File size:	216064 bytes
Analysis date:	2023-09-14 | 20:51:20
CRC32:	a59afdb1
MD5:	45a357276700238ab1ea952f8ae8ab1b
SHA-1:	c5afe7a1585f00829d2793b649447b98aa857844
SHA-2:	9deda531966448484b96d86893bc4158befc39c886abafa4bea932f514de4ec6
SSDEEP:	3072:bhtYy2U9Hj+lieT131k7M1R7l1v7X3uzrVi1ELE:lrCTT13ee51v7X+s1E

Figure 3 - Scan Result

Eva0x00 Ransomware has **4/40 detection rate** on Kleenscan. It bypasses mostly used antivirus programs.

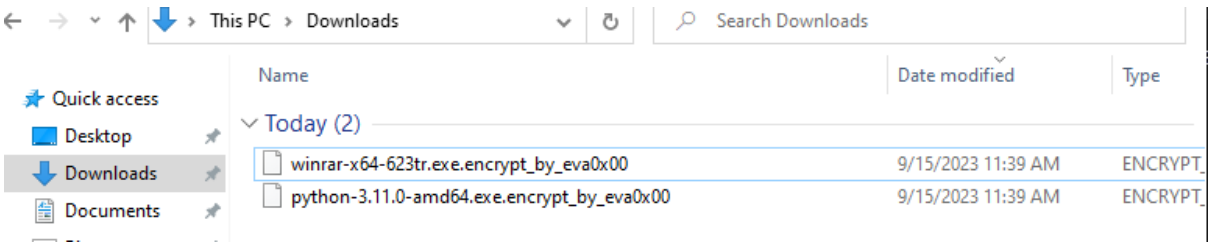


Figure 4- Encrypted file extension

Eva0x00 Ransomware encrypts the victim data with the extension of **"encrypt_by_eva0x00".** This extension can be changed by the user.

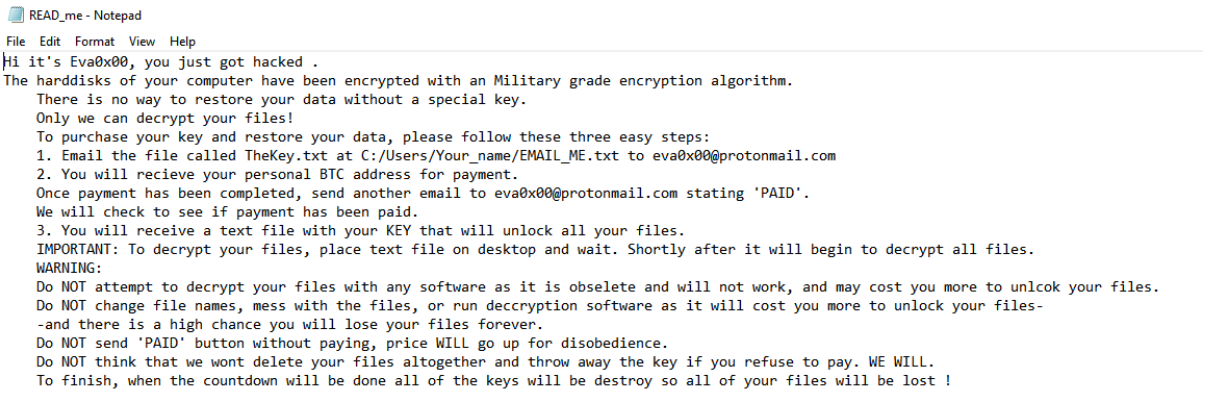


Figure 5- Readme.txt file

Once the ransomware is being executed, it encrypts all data under **in 3 minutes** and creates the **"READ_me.txt"** file. The file's content stipulates the necessity of a Bitcoin payment for decryption.

Static Analysis

eva.exe Analysis

File Name	eva.exe
MD5	45a357276700238ab1ea952f8ae8ab1b
SHA256	9deda531966448484b96d86893bc4158befc39c886abafa4bea932f514de4ec6
File Type	PE/32

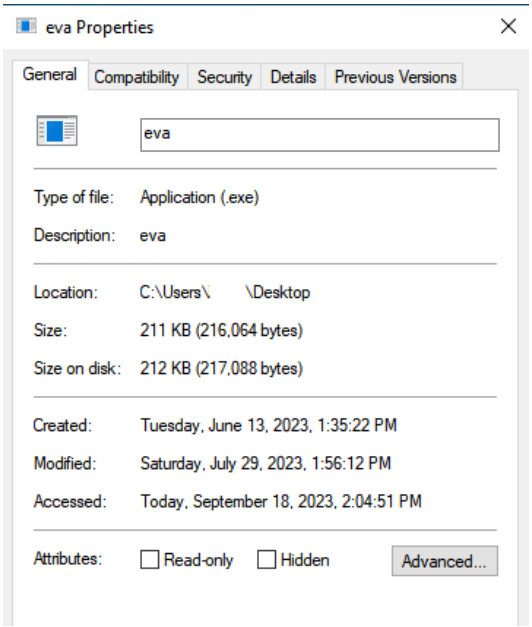


Figure 1- Information about malicious file

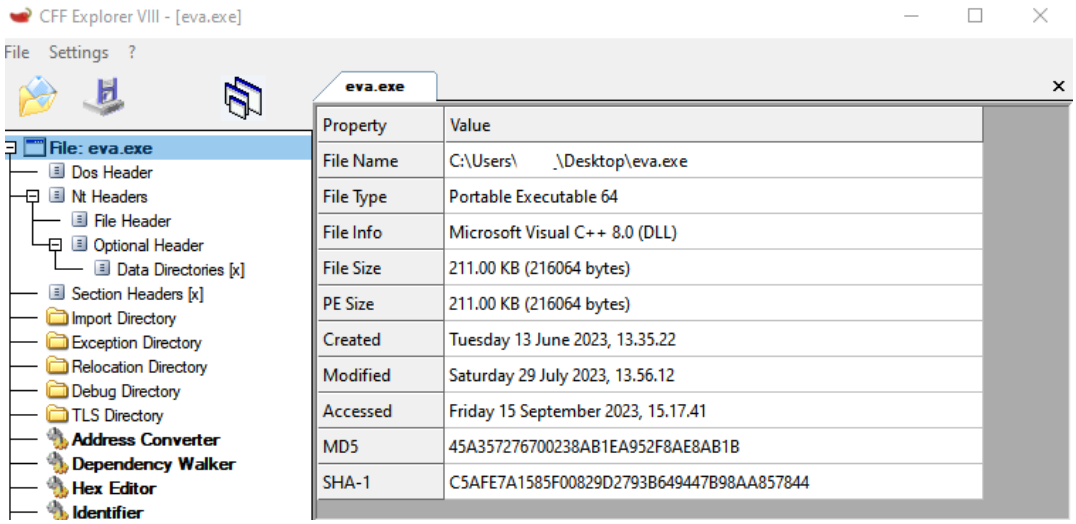


Figure 2- General Information

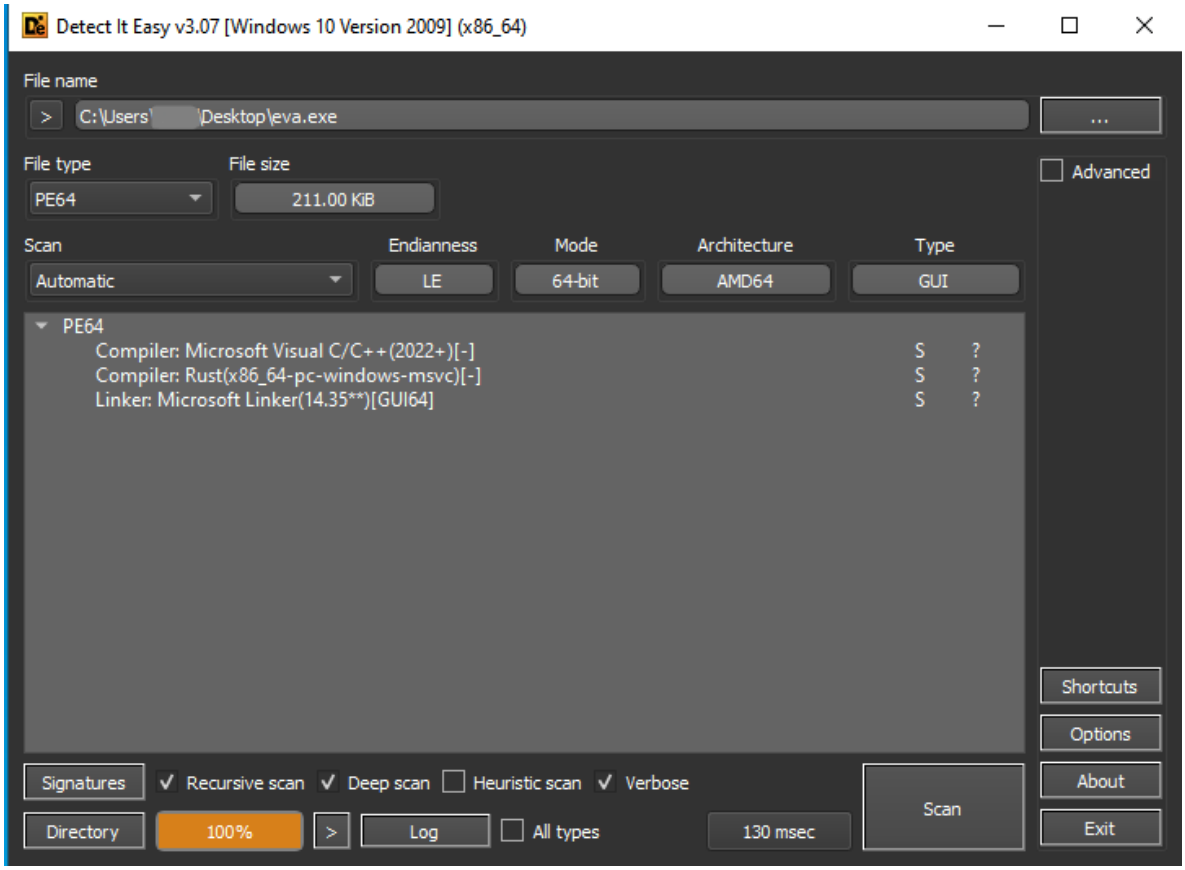


Figure 3- General Information

The ransomware was compiled using Microsoft Visual **C/C++** compiler, specifically a version from 2022 or later. The ransomware was also compiled using the **Rust programming language**, targeting the x86_64 architecture on the Windows platform with the MSVC (Microsoft Visual C++) toolchain.

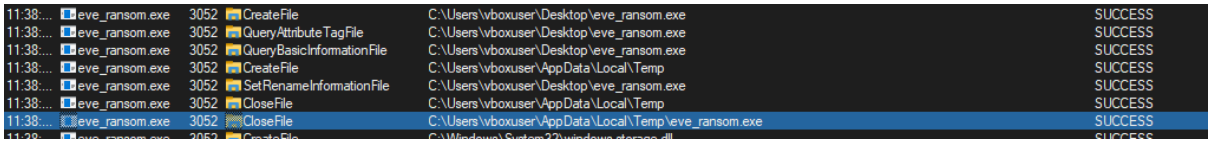


Figure 4- Move ransomware to Temp path

After the software is executed, it moves itself from the directory it was run in, to the **$HOME\AppData\Local\Temp** directory.

Dynamic Analysis

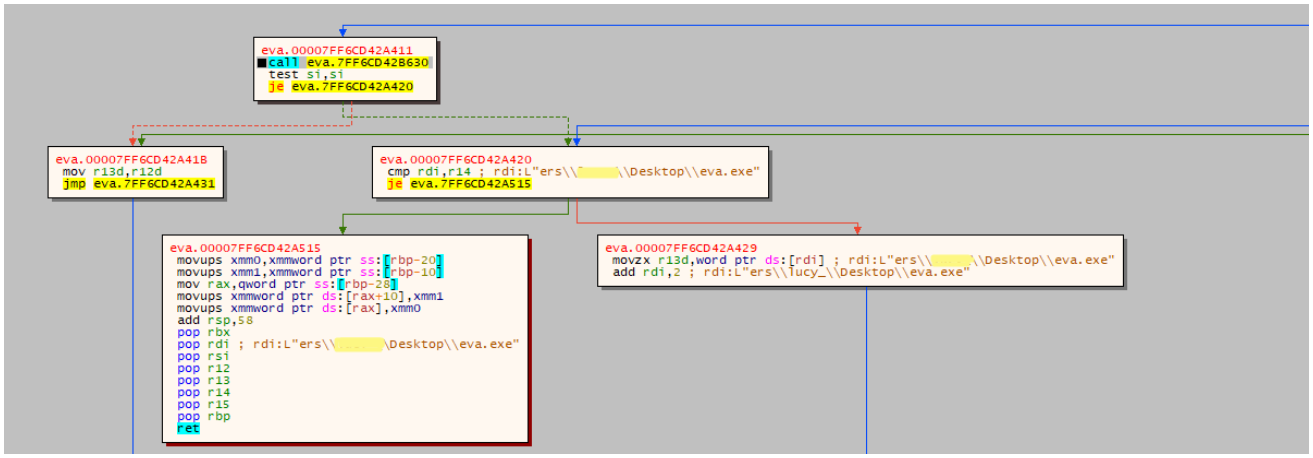


Figure 1- Gets location information

The malicious file gets the information in which location it is running.

Figure 2- Gets temp path information

When the ransomware runs, it uses the **GetTempPathw API** to move itself to the temp directory from its current location to get the path to the temp directory.

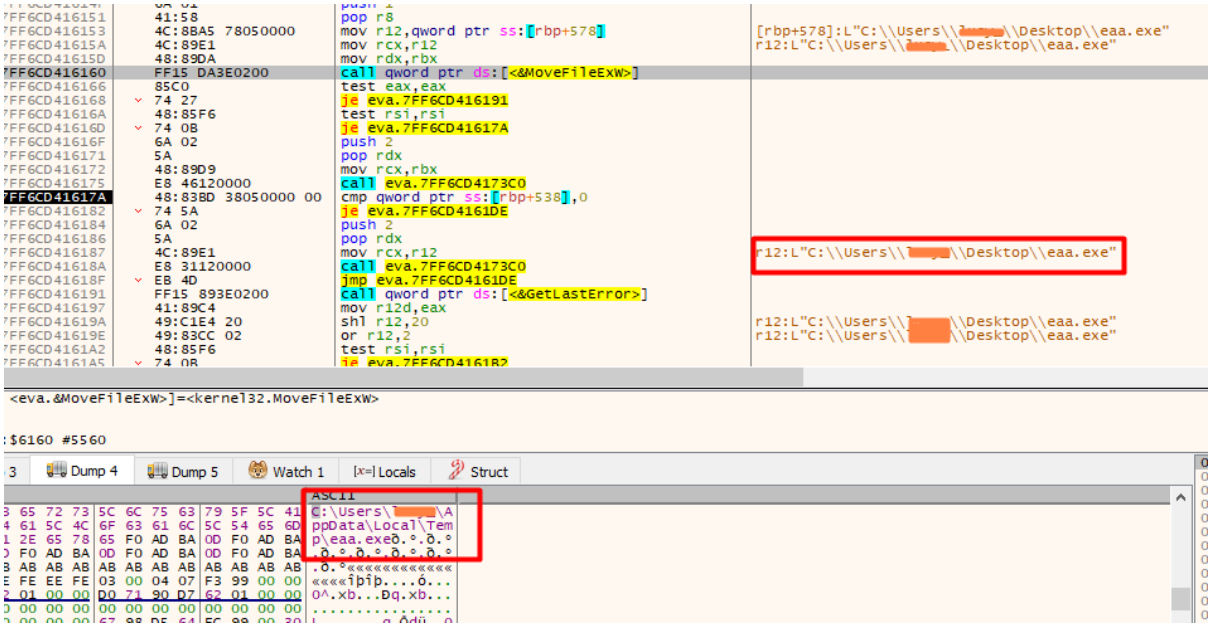


Figure 3- Move ransomware to Temp path

After the malware detects its location and the location of the temp directory, it moves the malicious file to the temp directory using the **MoveFileEx API**.
C:\Users\Admin\AppData\Local\Temp\eva.exe

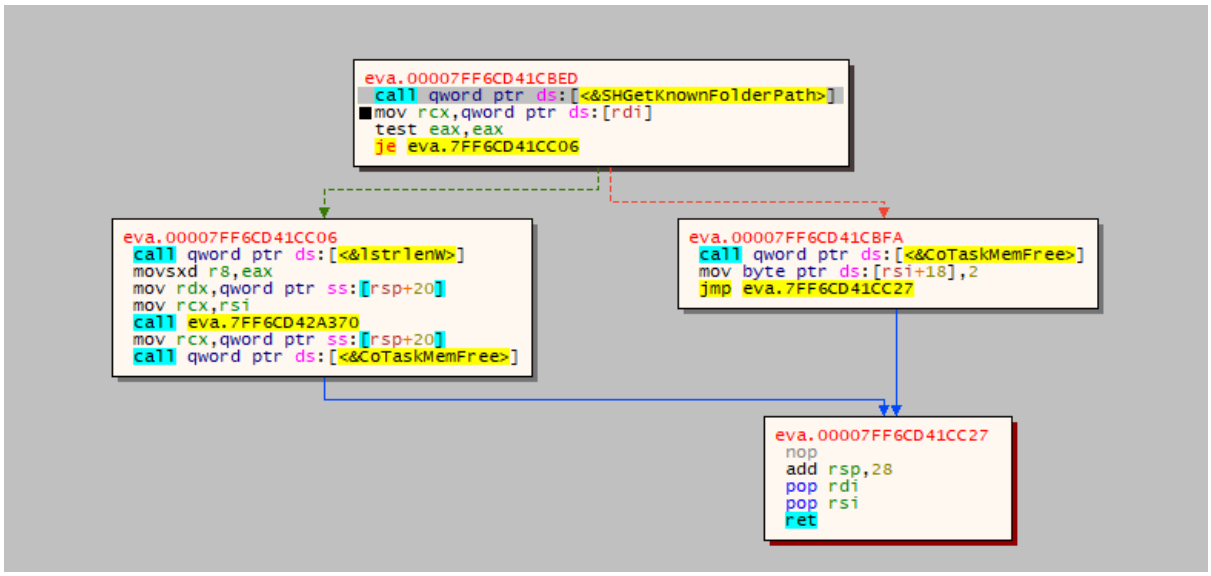


Figure 4- Gets the paths to be encrypted

The program utilizes the **SHGetKnownFolderPath API** to systematically retrieve individual directory paths for encryption. The directories being scanned are specified within a table. The process involves iterating through the provided table, invoking the **SHGetKnownFolderPath API** for each directory entry, and subsequently processing the retrieved paths for encryption purposes.

Figure 5- Gets the downloads path

C:\Users\Admin\Desktop	**C:\Users\Admin\Downloads**
C:\Users\Admin\Documents	**C:\Users\Admin\desktop.ini**
C:\Users\Admin\Videos	**C:\Users\Admin\AppData\Roaming**
C:\Users\Admin\Pictures	**C:\Users\Admin\AppData\Default**
C:\Users\Public	**C:\Users\Admin\AppData\Local**

Table 1 - Paths scanned

The directories being scanned are enumerated exactly as they are listed in the provided table.

Figure 6- Using ZWReadFile

Using the **ZWReadFile API**, the program reads the file slated for encryption in a manner consistent with standard file read operations.

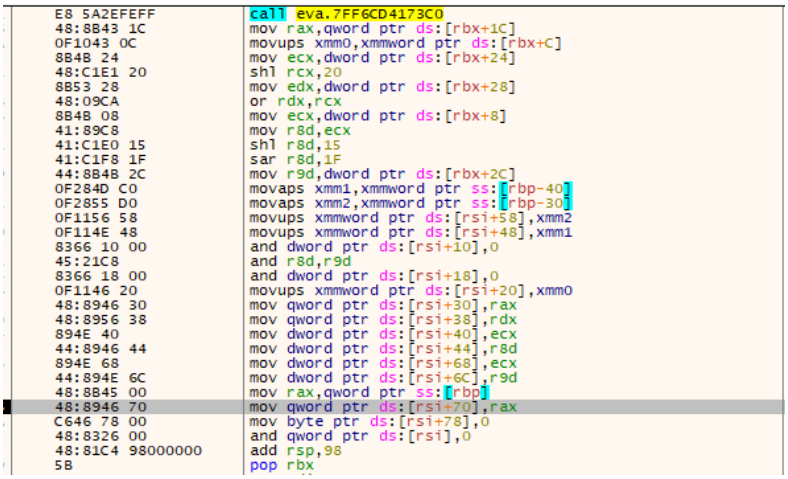


Figure 7- Part of encryption

The eva0x00 ransomware initiates the encryption process on the read file after performing an extended encryption algorithm.

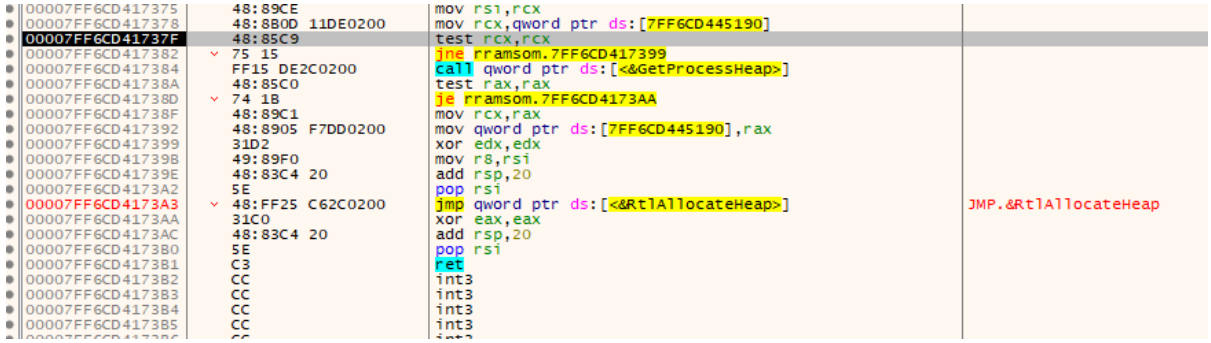


Figure 8- Using RtAllocateHeap

The eva0x00 ransomware allocates memory space equal to the size of the encrypted file using the **RTAllocateHeap API** in order to create the encrypted file.

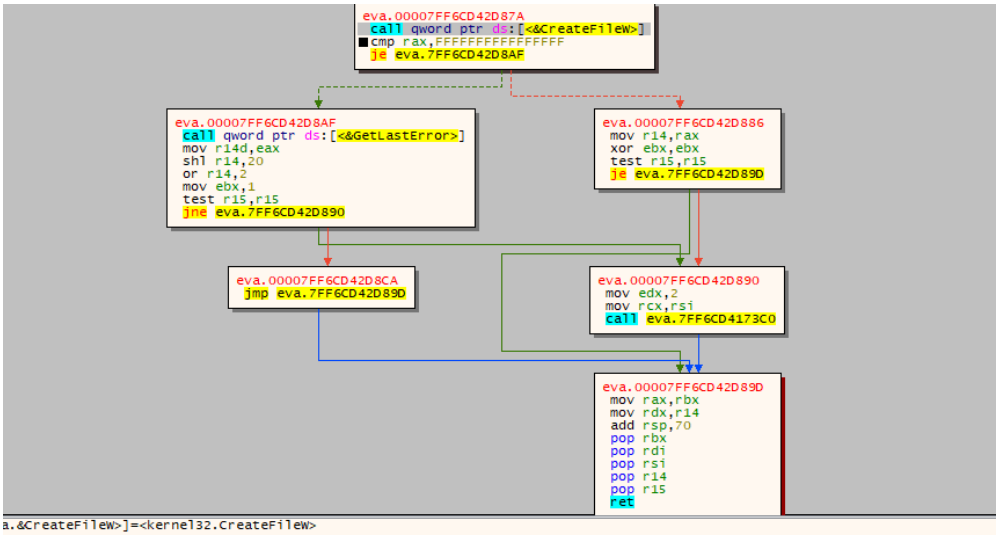


Figure 9- Creating the encrypted file

Using the **CreateFile API**, the eva0x00 ransomware generates the encrypted file in the directory where the original file was located, appending the extension **".encrypt_by_eva0x00"** to it.

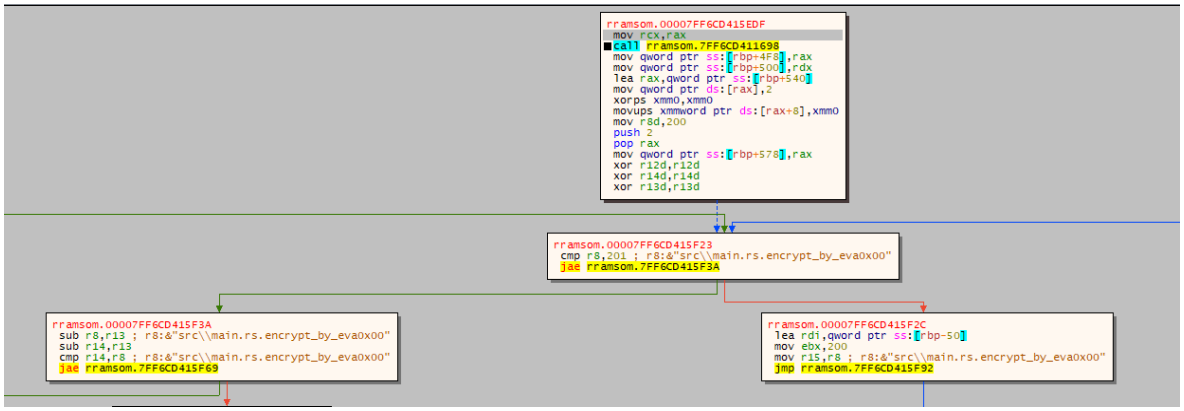


Figure 10- Adding an extension to the encrypted file

In the Rust programming language, the ransomware defines the **".encrypt_by_eva0x00"** extension, typically by using a variable or constant, to represent the file extension.

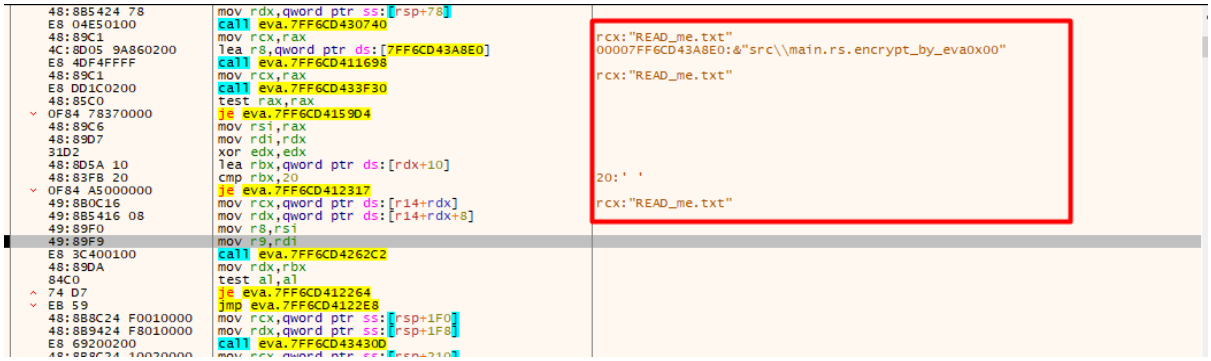


Figure 11- Create READ_me.txt

After creating the encrypted file, the eva0x00 ransomware proceeds to append the **"READ_me.txt"** file to the same directory where the encrypted file is located.

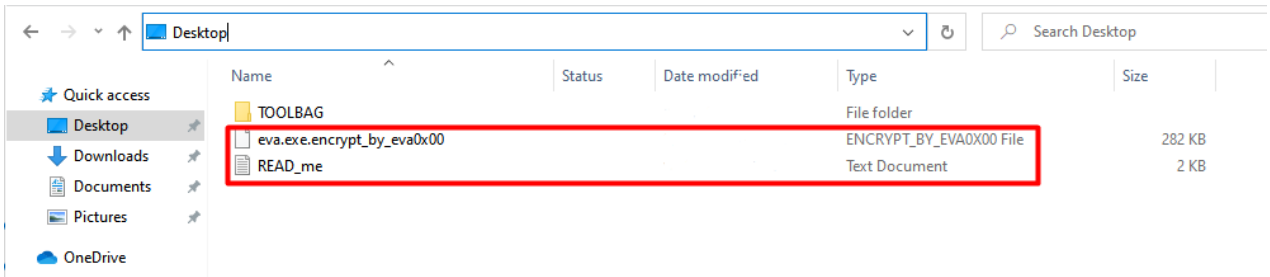


Figure 12- Encrypted files

The program systematically traverses all directories it searches, targeting files with specific extensions **"encrypt_by_eva0x00".**, including **".pdf," ".png," ".jpg," ".jpeg," ".mp3," ".mp4," ".txt," and ".exe."** It encrypts these files and alters their file extensions accordingly. Upon completion of the file traversal process, it generates a **"READ_me.txt"** file. Following the successful encryption of all scanned files, the program proceeds to terminate itself.

IOCs

HASHs:

IOC Type	IOC
MD5	45a357276700238ab1ea952f8ae8ab1b
SHA1	c5afe7a1585f00829d2793b649447b98aa857844
SHA256	9deda531966448484b96d86893bc4158befc39c886abafa4bea932f514de4ec6

YARA RULE

```
import "hash"
rule Eva0x00
{
meta:
        author = "Kerime Gencay"
        description = "Eva0x00 Ransomware Rule"
        file_name = "eva.exe"
        hash = "45a357276700238ab1ea952f8ae8ab1b"
strings:

        $text1 = "TheKey.txtC:\\Users\\abist\\.cargo\\registry\\src\\index.crates.io-
6f17d22bba15001f\\aes-0.7.5\\src\\soft\\fixslice64.rs" //rust library
        $text2 = "library\\std\\src\\sys\\windows\\path.rs" //rust library
        $text3 = "library\\std\\src\\sys_common\\thread_info.rs" //rust library
        $text4 = "src\\main.rs.encrypt_by_eva0x00" // encrypted file extension

        $opc1 = {FF 15 DE 2C 02 00 48 85 C0 74 1B 48 89 C1 48 89 05 F7 DD 02 00 31 D2
49 89 F0 48 83 C4 20 5E 48 FF 25 C6 2C 02 00 31 C0 48 83 C4 20 5E}
        $opc2 = {72 22 48 8B 0E 48 01 D9 48 8D 54 24 2C 49 89 F8 E8 B9 C5 00 00 48 01
FB 48 ?? ?? 10 48 ?? ?? 30 5B 5F 5E}


condition:

        uint16(0) == 0x5A4D and (any of ($text*,$opc*))

}
```

MITRE ATT&CK TABLE

Discovery	Impact	Defense Evasion	Persistence	Reconnaissance
T1082 Information Discovery	T1486 Data Encrypted for Impact	T1036 Masquerading	T1047 Create or Modify Systems	T1566 Phishing
T1543 File and Directory Discovery		T1564.001 Hidden Files and Directories		

MITIGATIONS

● Ransomware is often spread through phishing emails. Be careful when receiving a suspicious email and avoid opening attachments or clicking on links.
● Strong passwords provide better protection against ransomware. Change your passwords frequently and make them as complex as possible.
● Paying the ransom only encourages ransomware attacks. Before paying the ransom, try other methods to recover your files.
● It is recommended that organizations use security products to secure potential entry points such as endpoints, email, webs, and networks. Thus, they can protect themselves against ransomware attacks.