REPUBLIC OF ALBANIA
NATIONAL CYBER SECURITY AUTHORITY
CYBER SECURITY ANALYSIS DIRECTORATE

HIDDEN IN THE STARS

A malware using steganography

CVE-2017-11882

Version: 1.0
Date: 03/03/2025

TLP:CLEAR

CONTENT

LIST OF FIGURES

This report has limitations and should be interpreted with caution!

Some of these limitations include:

First phase:
Sources of information: The report is based on information available at the time of its preparation. However, some aspects may differ from actual developments.

Second phase:
Analysis details: Due to resource limitations, some aspects of the malicious file may not have been analyzed in depth. Any additional unknown information may reflect changes in the report.

Third phase:
Information Security: To protect sources and confidential information, some details may be redacted or not included in the report. This decision was made to maintain the integrity and security of the data used.

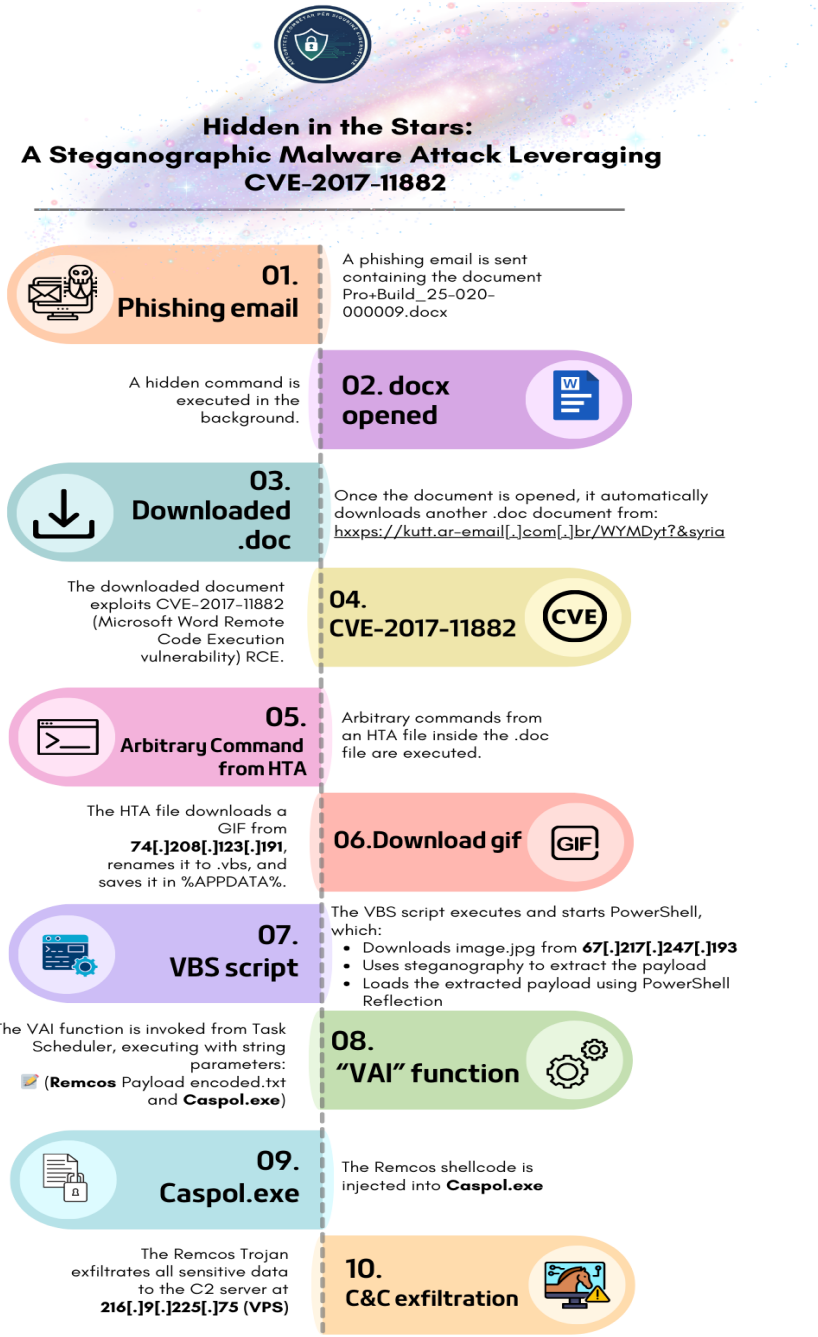
NCSA reserves the right to change, update, or amend any part of this report without prior notice.

This report is not a final document.

The findings of the report are based on the information available at the time of the investigation and analysis. There is no guarantee regarding possible changes or updates to the information reported during the subsequent period. The authors of the report do not assume responsibility for the misuse or consequences of any decision-making based on this report.

Technical Information

A phishing campaign targeting infrastructures in Albania has been identified, sending an email containing a malicious file named **Pro+Build_25-020-000009.**



Analysis of Pro+Build_25-020-000009 file

Pro+Build_25-020-000009 is a docx file and at first glance appears to be a regular, legitimate Word file. If we access the file, a URL will appear that attempts to download a file.

Figure 1: External URL

```
THIS IS WORK IN PROGRESS - Check updates regularly!
Please report any issue at https://github.com/decalage2/oletools/issues

-------------------------------------------------------------------------------
File: '../Pro+Build_25-020-000009.docx'
Found relationship 'attachedTemplate' with external link https://kutt.ar-email.com.br/WYMDyt
extract file embedded in OLE object from stream '\x01Ole10Native':
Parsing OLE Package
Filename = "FEB2025.xlsx"
Source path = "C:\Users\91974\OneDrive\Desktop\WordFile\2025\2025New\FEB2025.xlsx"
Temp path = "C:\Users\91974\AppData\Local\Temp\FEB2025.xlsx"
saving to file ../Pro_Build_25-020-000009.docx_FEB2025.xlsx
```

Figure 2: External URL analysis

If we try to access the URL in a browser, automatically will be downloaded a file named: **nicepersongivenmebestoptionsforlongtime___nicepersongivenmebestoptionsforlongtime___nicepersongivenmebestoptionsforlongtime.doc**.

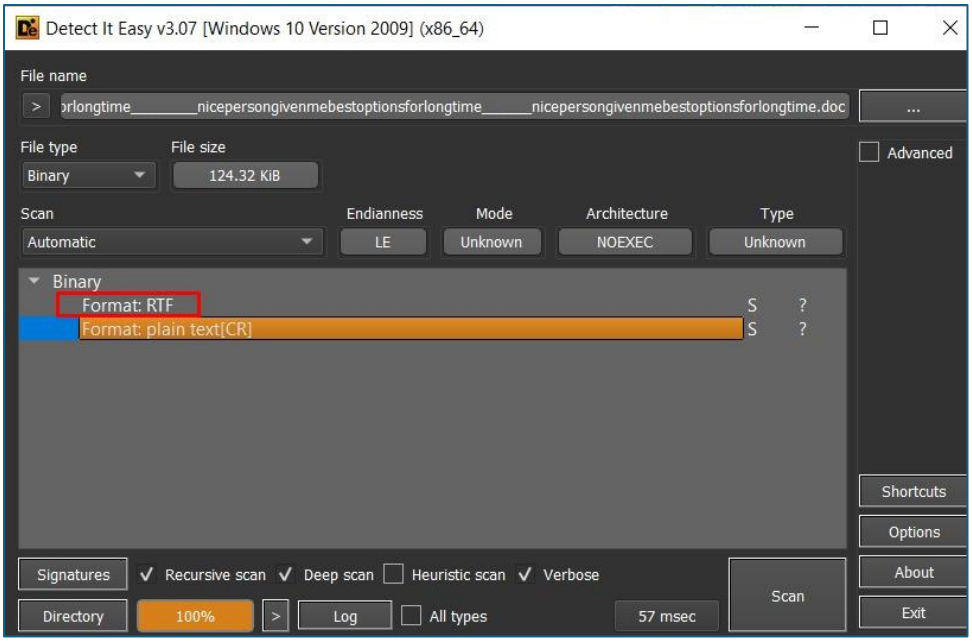


Figure 3: RTF file

Typically, files with the **.doc** extension are used in campaigns that exploit vulnerabilities in **Microsoft Office (Word)** that execute arbitrary commands on the infected computer using **HTA** files as payloads. To confirm this fact, we continue the dynamic analysis and identify a Powershell command that is executed after the .doc file is downloaded.

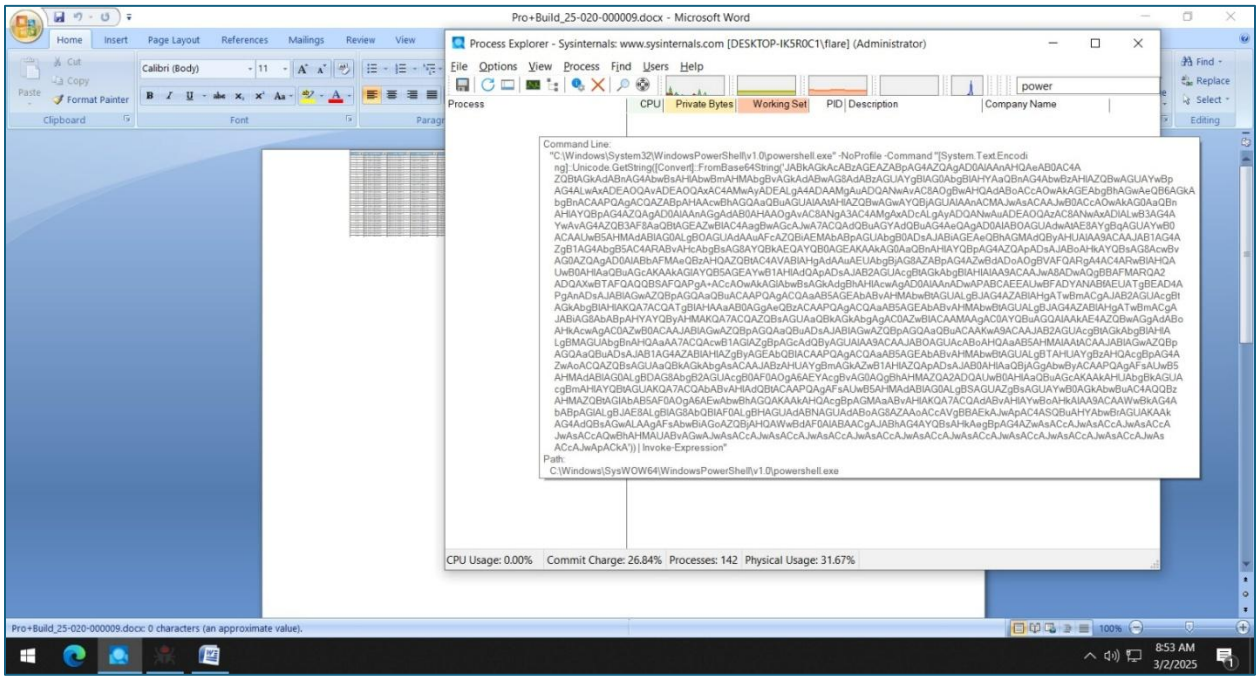


Figure 4: Arbitrary powershell command

If we analyze the code that is executed, we see that we are dealing with a **base64** encoded command, which during execution is decoded and executed with **Invoke-Expression**.
To understand the behavior of the file, we take the full command and modify it by adding variables to see the result.

Figure 5: Powershell command decode

The download of a file with the **.vbs** extension is recorded, where it is stored in the **%APPDATA%** location. If we access the file in this directory, a file with the name: **"nicepersongivenmebestoptionsforlongtim.hta"** appears, which contains the payload that is exploited by the Microsoft Office Word vulnerability **CVE-2017-11882** from the **.doc** file itself downloaded in the first stage.

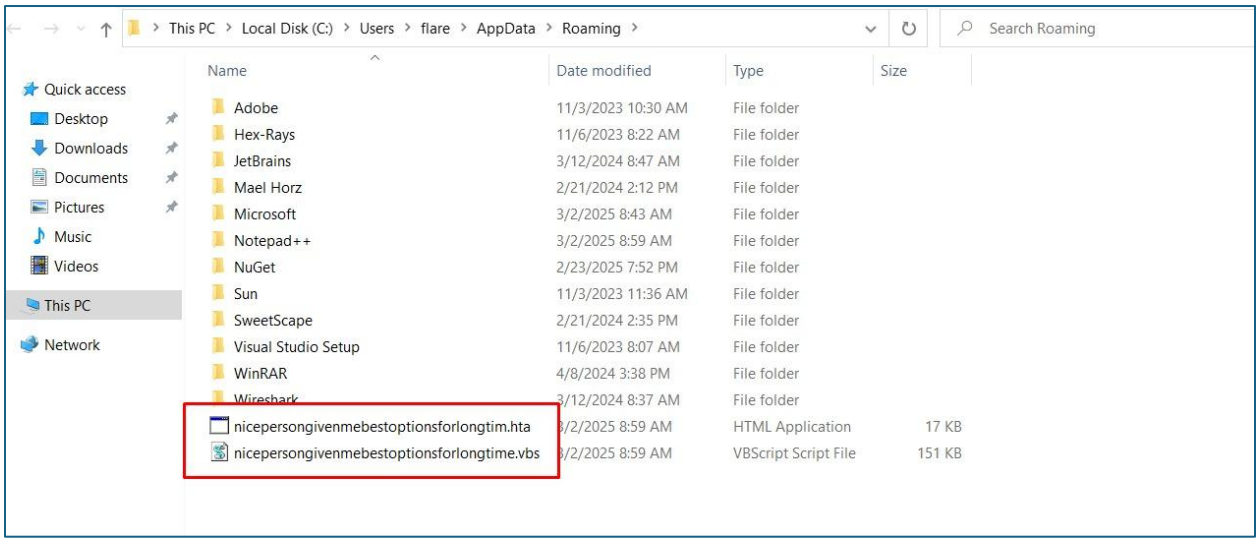


Figura 6: HTA payload and .vbs file

The **.vbs** file has a very large number of lines of code and variables that do not seem to have any malicious intent, but during execution at runtime a Powershell command encoded in **base64** is executed. We modify the code and manage to display what is executed and follow it with the debug process.

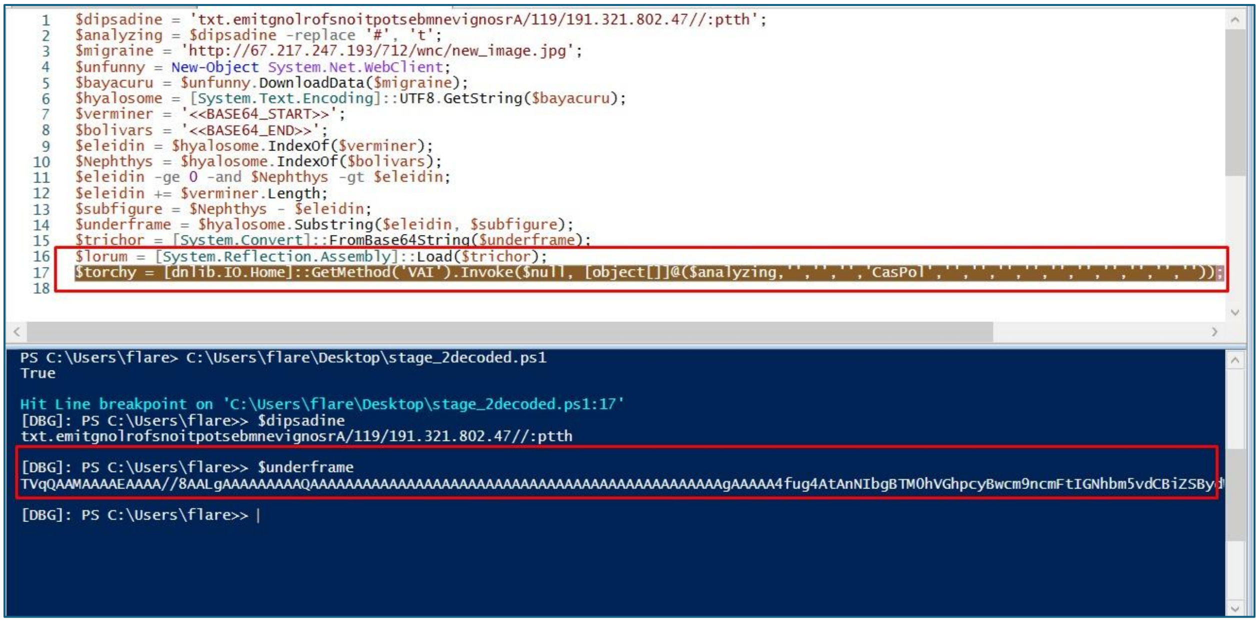


Figure 7: Powershell command decode

We see a large number of variables. The variable **$dipsadine** seems to store a URL where the string of characters is reversed. The variable **$migraine** contains an image.

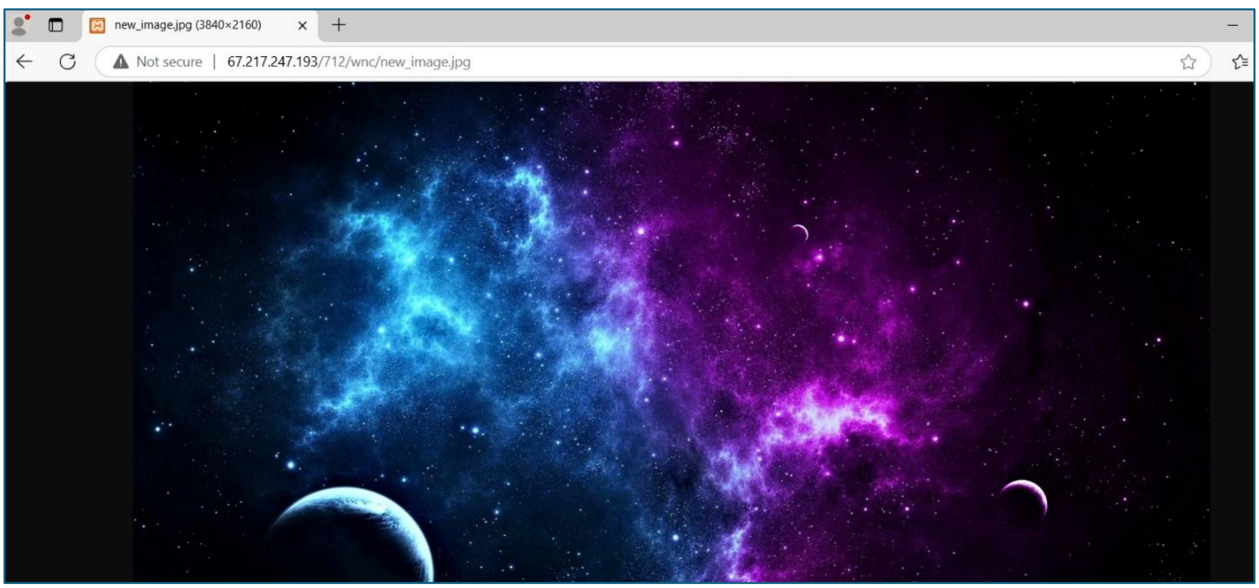


Figure 8: new_image.jpg

The **$unfunny** variable creates a **WebClient objec**t and starts downloading the image with the **$migraine** variable as a parameter. The **$verminer** and **$bolivars** variables contain the beginning and end of a **base64** encoded payload indicating that this is a steganography instance. During execution, it is decoded from **base64** and uses **PowerShell Reflection**, which is used to execute an **.exe** or **.dll** file written in **.NET**. **Home** has a function called **VAI** that takes as a parameter the variable **$analyzing** where it is a text file that is retrieved from the URL specified at the beginning.

To see this executable file we take the result of the variable **$underframe:**

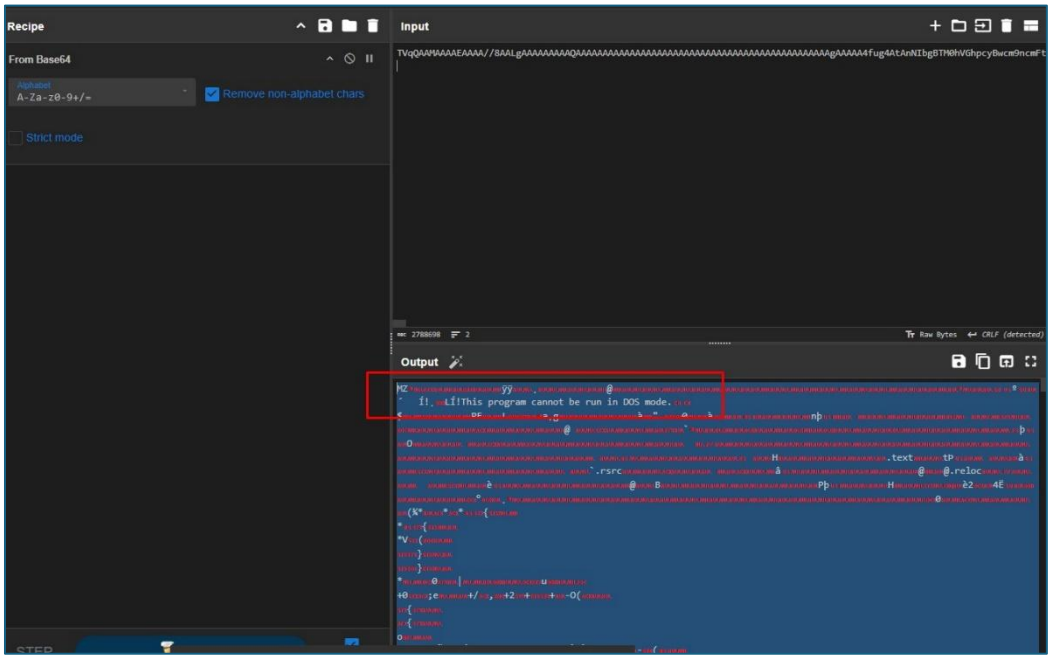


Figure 9: Executable file .NET.

After downloading the file, we perform a more in-depth analysis of it.

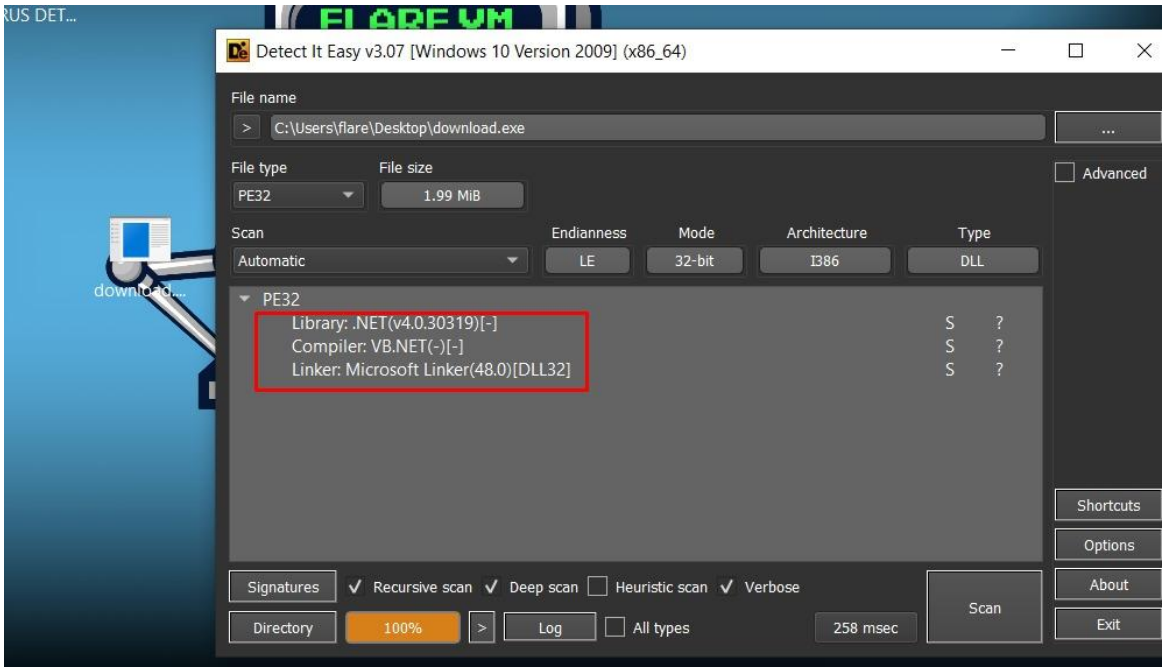


Figure 10:.NET file

If we start the Reverse engineering phase of the file, we will find that it has a name **TaskScheduler.** This file has imported a large number of **Win32API** functions such as: **VirtualAlloc**,

VirtualProtect, WriteProcessMemory_API, etc., which means that it is trying to inject a payload into a legitimate process.

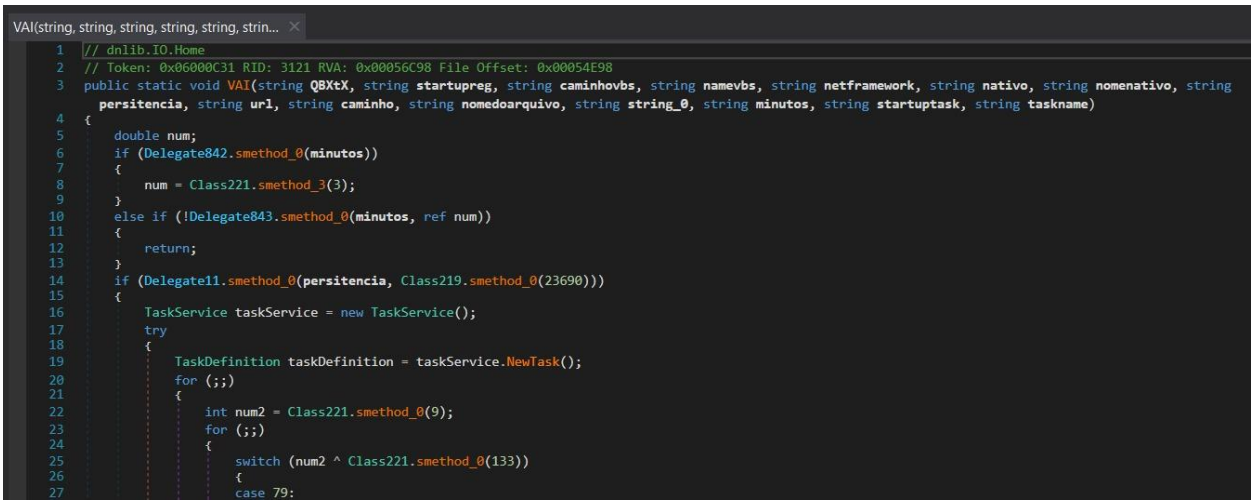


Figure 11: VAI function

VAI function is a function that takes a total of 15 parameters, of which only 2 are the parameters that it uses. One is a payload of type text and the second is the **CasPol** character string.

Figure 12: TXT parameter like payload

This payload is passed as an argument to several functions, transformed and injected into a legitimate process named **CasPol (Code Access Security Policy Tool**). So, the file starts the legitimate **CasPol** process and performs injection into the memory of this process, so to see what happens we follow it with the debug process. .

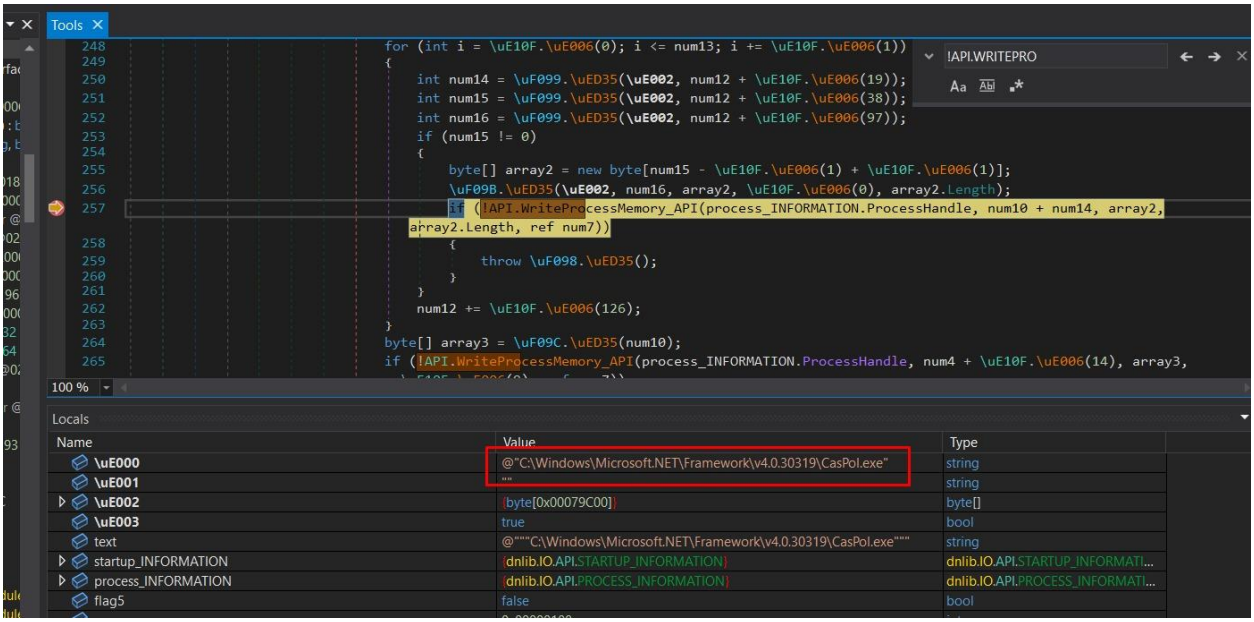


Figure 13: CasPol.exe and byte vector

A vector of bytes is also identified, which indicates that we are dealing with another payload, an executable file that we can distinguish by the hex value **4D 5A.**

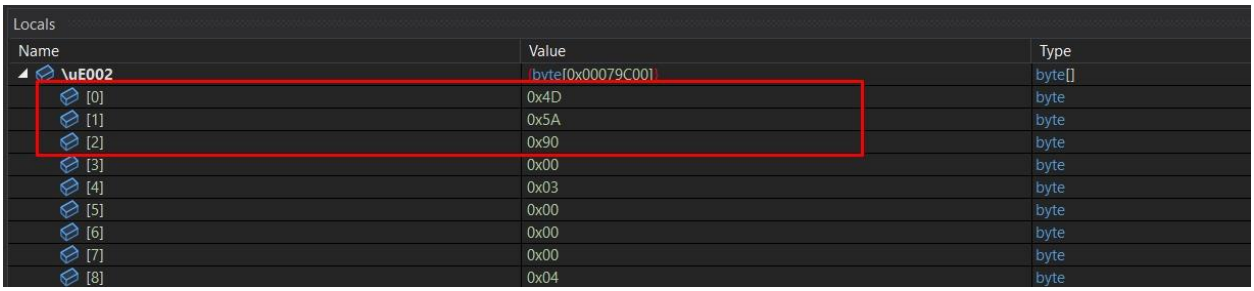
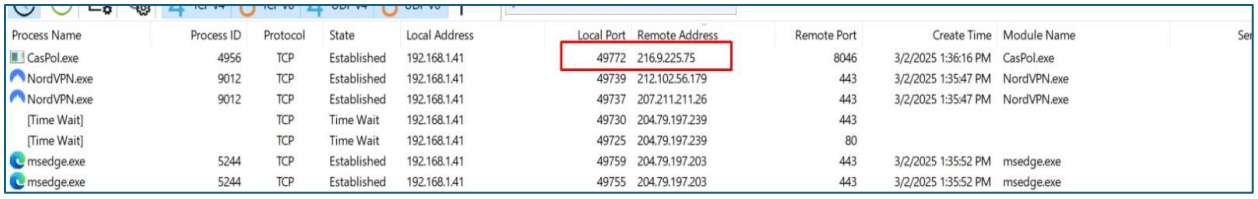


Figure 14: Executable file

After downloading this file, it will be evident that it will automatically receive a logo from which it is understood that we are dealing with **REMCOS RAT**, which is injected as shellcode into the legitimate **CasPol.exe** process.

Figure 15: Remcos RAT



Process Name	Process ID	Protocol	State	Local Address	Local Port	Remote Address	Remote Port	Create Time	Module Name
CasPol.exe	4956	TCP	Established	192.168.1.41	49772	216.9.225.75	8046	3/2/2025 1:36:16 PM	CasPol.exe
NordVPN.exe	9012	TCP	Established	192.168.1.41	49739	212.102.56.179	443	3/2/2025 1:35:47 PM	NordVPN.exe
NordVPN.exe	9012	TCP	Established	192.168.1.41	49737	207.211.211.26	443	3/2/2025 1:35:47 PM	NordVPN.exe
[Time Wait]		TCP	Time Wait	192.168.1.41	49730	204.79.197.239	443		
[Time Wait]		TCP	Time Wait	192.168.1.41	49725	204.79.197.239	80		
msedge.exe	5244	TCP	Established	192.168.1.41	49759	204.79.197.203	443	3/2/2025 1:35:52 PM	msedge.exe
msedge.exe	5244	TCP	Established	192.168.1.41	49755	204.79.197.203	443	3/2/2025 1:35:52 PM	msedge.exe

Figure 16: Command and Control Server

MITRE ATT&CK

Nr.	Tactic	Technique
1	**Initial Access (TA0001)**	T1566 :Phishing
		T1566.001 : Spear phishing Attacment
2	**Execution (TA0002)**	T1053.005 : Scheduled Task
		T1204.002: Malicious File
3	**Persistence (TA0003)**	T1547.001 : Registry Run Keys / Startup Folder
		T1053.005 : Scheduled Task
4	**Privilege Escalation (TA0004)**	T1140 : Deobfuscation
		T1055.012 : Process Hollowing
		T1053.005 : Scheduled Task
5	**Defense Evasion (TA0005)**	T1564.001 : Hidden Files and Directories
		TA1562.001 : Disable or Modify Tools
		T1055.012 : Process Hollowing
		T1564.003 : Hidden Window
6	**Credential Access (TA0006)**	T1555.003 : Credentials from WebBrowser
		TA1552.001 : Credentials in files
		TA1552.002 : Credentials in registry
7	**Discovery (TA0007)**	T1087.001 : Local Account

Indicators of Compromise (IoCs)

D6D66AB4FA69971164809C21EAE630861605332F38B99BC885402ED2CC92539B	**Pro+Build_25-020-000009.docx**
816EF91298A44C3231B5E17FBD85D6F1CE565819A921331AF56D32FB535F5F52	**nicepersongivenmebestoptionsforlongtim.hta**
E6E5BC0F0C1757DB14691C14FE6E179E06C1F	**nicepersongivenmebestoptionsforlongtime.vbs**

4733D2A91E00C1C0AA4A88C5A59	
AD4E305243EEA4EB48308CE4BCCD6B999ED93F9810A82236987187FDA9E12DE1	**Microsoft.Win32.TaskScheduler**
DF758036E3B14633B29733C7F62A9D52660BFAF6124CB2BB3427C3B81714082B	**NICEPERSONGIVENMEBESTOPTIONSFORLONGTIME_____NICEPERSONGIVENMEBESTOPTIONSFORLONGTIME_____NICEPERSONGIVENMEBESTOPTIONSFORLONGTIME.DOC**
76B1F681DD3B617B88568D2D0A0AAC9B589C89B569FB25AC5BE0DF0839E96E8D	**New_image.jpg**
74[.]208[.]123[.]191	**Dropper**
67[.]217[.]247[.]193	**Dropper**
216[.]9[.]225[.]75	**C2**
hxxps://kutt.ar-email[.]com[.]br/WYMDyt?&syria	**Dropper**

Recommendations

National Cyber Security Authority recommends:

- Immediate blocking of the Indicators of Compromise, mentioned above, on your protective devices..
- Continuous analysis of logs coming from SIEM (Security Information and Event Management).
- Training non-technical staff about "Phishing" attacks and ways to avoid infection from them.
- Installing network perimeter devices that perform deep traffic analysis based not only on access list rules but also on its behavior (NextGen Firewalls).
- The identified systems should be segmented into different VLANs, applying an “Access control list for the entire network perimeter”, web services should be separated from their database, Active Directory should be in a separate VLANs.
- Application and use of the LAPS technique for Microsoft systems, for managing Local Administrator passwords..
- Apply traffic filters in the case of remote access to hosts (employees/third parties/customers).

- Implement solutions that filter, monitor, and block malicious traffic between Web applications and the internet, Web Application Firewall (WAF).
- Conduct traffic analysis at the behavior level for end devices, applying EDR, XDR solutions. This brings the analysis of malicious files not only at the signature level but also at the behavior level.
- Design the "Identity Access Management" user access management solution to control user identity and privileges in real time according to the "Zero-Trust" principle.

Authors:
Adriano Lleshi
Bledar Kacadej
Ervis Qose
Eriola Sadiku
Ergis Gaxho
Gentian Teliti
Kristian Josifi
Redon Hoxha
Sara Qoshi
Vilma Tema