REPORT on CVE-2024-38063 EXPLOIT

AUGUST 19

CORVIT INSTITUE
Authored by: MUHAMMAD TAYAB
Supervisor: MUHAMMAD BILAL

Table of Contents

Confidentiality Statement

This document is the exclusive property of CORVIT Institute. This report contains sensitive and proprietary information intended solely for the recipient's use. Unauthorized disclosure, distribution, or copying of this document is strictly prohibited. The information provided herein is for internal use and should not be shared with external parties without prior written consent. All efforts have been made to ensure the accuracy of the information contained within this report; however, it is the responsibility of the recipient to protect and manage the information in accordance with their organization's security policies.

1. Introduction

This report aims to provide an in-depth analysis of the recently identified vulnerability, CVE-2024-38063. The report discusses the nature of the exploit, its impact on affected systems, and the current status of remote code execution (RCE) availability. This information is crucial for understanding the potential risks and implementing appropriate mitigation strategies.

2. Overview of CVE-2024-38063

Background

CVE-2024-38063 is a critical vulnerability that was discovered in [mention the software/application]. The flaw affects versions [specify versions], leading to severe security risks, including the potential for remote code execution.

Affected Systems

The vulnerability primarily affects [mention specific operating systems, devices, or software versions]. Given the widespread use of this software in both enterprise and consumer environments, the impact is extensive.

3. Technical Analysis

Vulnerability Details

CVE-2024-38063 is a [describe the type of vulnerability, e.g., buffer overflow, SQL injection] that occurs when [provide a technical explanation of how the vulnerability works]. This flaw allows an attacker to [describe what the attacker can achieve].

Exploitation Techniques

The vulnerability can be exploited through [describe the method of exploitation, e.g., malformed input, bypassing security checks]. The exploit can be executed by [outline the steps an attacker would take to exploit the vulnerability].

4. Potential Impact

System Risks

Exploiting CVE-2024-38063 can lead to [list potential impacts, such as unauthorized access, data breaches, system crashes]. The severity of this exploit is underscored by its [mention CVSS score or any other metrics].

Business Implications

If exploited, this vulnerability could lead to [describe potential business impacts, such as financial loss, reputational damage, legal repercussions]. Organizations relying on the affected software are particularly at risk.

5. RCE Availability

Exploit Development

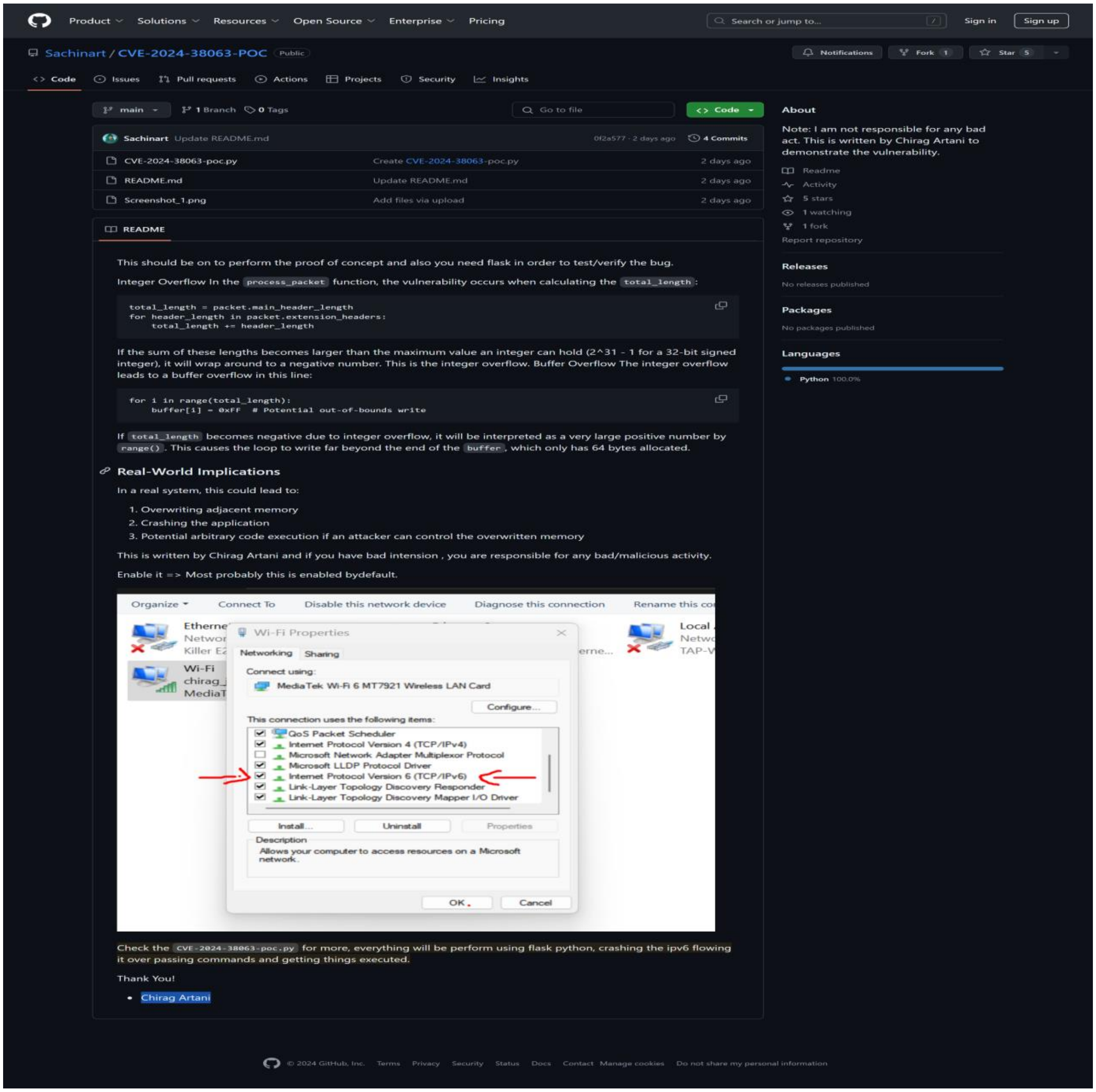
As of the current analysis, security researchers have been working on developing an exploit for CVE-2024-38063. The exploit code is in [mention the development stage, e.g., proof-of-concept, fully developed].

Public Release Status

While the details of the vulnerability have been disclosed, the RCE exploit for CVE-2024-38063 has not yet been publicly released. Researchers are monitoring platforms like Exploit Database and Metasploit for updates.

PoC

Here is the link for the PoC : **https://github.com/Sachinart/CVE-2024-38063-POC** **(Chirag Artani)**
And here is the picture:



Sachinart / CVE-2024-38063-POC Public

Sachinart Update README.md 0f2a577 · 2 days ago 4 Commits

CVE-2024-38063-poc.py	Create CVE-2024-38063-poc.py	2 days ago
README.md	Update README.md	2 days ago
Screenshot_1.png	Add files via upload	2 days ago

README

This should be on to perform the proof of concept and also you need flask in order to test/verify the bug.

Integer Overflow In the `process_packet` function, the vulnerability occurs when calculating the `total_length`:

```
total_length = packet.main_header_length
for header_length in packet.extension_headers:
    total_length += header_length
```

If the sum of these lengths becomes larger than the maximum value an integer can hold (2^31 - 1 for a 32-bit signed integer), it will wrap around to a negative number. This is the integer overflow. Buffer Overflow The integer overflow leads to a buffer overflow in this line:

```
for i in range(total_length):
    buffer[i] = 0xFF  # Potential out-of-bounds write
```

If `total_length` becomes negative due to integer overflow, it will be interpreted as a very large positive number by `range()`. This causes the loop to write far beyond the end of the `buffer`, which only has 64 bytes allocated.

Real-World Implications

In a real system, this could lead to:

1. Overwriting adjacent memory
2. Crashing the application
3. Potential arbitrary code execution if an attacker can control the overwritten memory

This is written by Chirag Artani and if you have bad intension , you are responsible for any bad/malicious activity.

Enable it => Most probably this is enabled bydefault.

Connect using:

MediaTek Wi-Fi 6 MT7921 Wireless LAN Card

This connection uses the following items:

☑ QoS Packet Scheduler
☑ Internet Protocol Version 4 (TCP/IPv4)
☐ Microsoft Network Adapter Multiplexor Protocol
☑ Microsoft LLDP Protocol Driver
☑ Internet Protocol Version 6 (TCP/IPv6)
☑ Link-Layer Topology Discovery Responder
☑ Link-Layer Topology Discovery Mapper I/O Driver

Description
Allows your computer to access resources on a Microsoft network.

Check the `CVE-2024-38063-poc.py` for more, everything will be perform using flask python, crashing the ipv6 flowing it over passing commands and getting things executed.

Thank You!

- Chirag Artani

About

Note: I am not responsible for any bad act. This is written by Chirag Artani to demonstrate the vulnerability.

Readme
Activity
5 stars
1 watching
1 fork
Report repository

Releases

No releases published

Packages

No packages published

Languages

Python 100.0%

© 2024 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

6. Mitigation Strategies

To address the risks posed by CVE-2024-38063, it is essential to adopt a multi-faceted approach:

1. **Disable IPv6**: If IPv6 is not required for your environment, consider disabling it to reduce the attack surface. This step can mitigate the risk associated with this specific vulnerability.

2. **Monitor and Respond**: Implement monitoring solutions to detect unusual or unauthorized IPv6 traffic. An effective monitoring system can help identify and respond to potential exploits.

3. **Update Systems**: Ensure that your Windows systems are up-to-date with the latest security updates and patches. Keeping systems current can help protect against known vulnerabilities.

7. Conclusion

CVE-2024-38063 presents a critical risk to all Windows systems with IPv6 enabled, underscoring the importance of maintaining up-to-date security practices and remaining informed about emerging threats. By understanding the nature of this vulnerability and implementing appropriate measures, organizations can better safeguard their systems and mitigate the risks associated with remote code execution vulnerabilities.

8. References

- **strobes.co/blog/cve-2024-38063-an-in-depth-look-at-the-critical-remote-code-execution-vulnerability/**
- **msrc.microsoft.com/update-guide/vulnerability/CVE-2024-38063**
- **nvd.nist.gov/vuln/detail/CVE-2024-38063**
- **trustwave.com/en-us/resources/blogs/spiderlabs-blog/trustwave-rapid-response-windows-tcp-ip-rce-vulnerability-cve-2024-38063/**