

Webrat, disguised as exploits, is spreading via GitHub repositories

SL securelist.com/webrat-distributed-via-github/118555

Maxim Starodubov

December 23, 2025



In early 2025, security researchers [uncovered](#) a new malware family named Webrat. Initially, the Trojan targeted regular users by disguising itself as cheats for popular games like Rust, Counter-Strike, and Roblox, or as cracked software. In September, the attackers decided to widen their net: alongside gamers and users of pirated software, they are now targeting inexperienced professionals and students in the information security field.

Distribution and the malicious sample

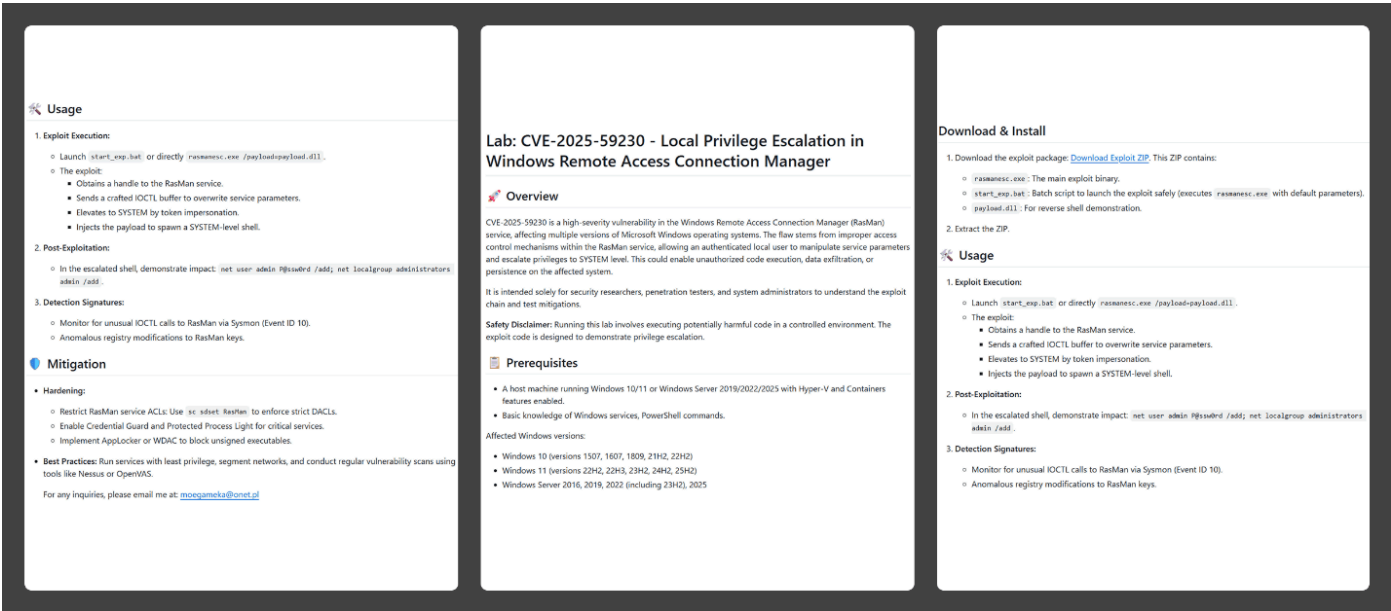
In October, we uncovered a campaign that had been distributing Webrat via GitHub repositories since at least September. To lure in victims, the attackers leveraged vulnerabilities frequently mentioned in security advisories and industry news. Specifically, they disguised their malware as exploits for the following vulnerabilities with high CVSSv3 scores:

CVE	CVSSv3
CVE-2025-59295	8.8
CVE-2025-10294	9.8
CVE-2025-59230	7.8

This is not the first time threat actors have tried to lure security researchers with exploits. Last year, they similarly [took advantage of the high-profile RegreSSHion vulnerability](#), which lacked a working PoC at the time.

In the Webrat campaign, the attackers bait their traps with both vulnerabilities lacking a working exploit and those which already have one. To build trust, they carefully prepared the repositories, incorporating detailed vulnerability information into the descriptions. The information is presented in the form of structured sections, which include:

- Overview with general information about the vulnerability and its potential consequences
- Specifications of systems susceptible to the exploit
- Guide for downloading and installing the exploit
- Guide for using the exploit
- Steps to mitigate the risks associated with the vulnerability



Contents of the repository

In all the repositories we investigated, the descriptions share a similar structure, characteristic of AI-generated vulnerability reports, and offer nearly identical risk mitigation advice, with only minor variations in wording. This strongly suggests that the text was machine-generated.

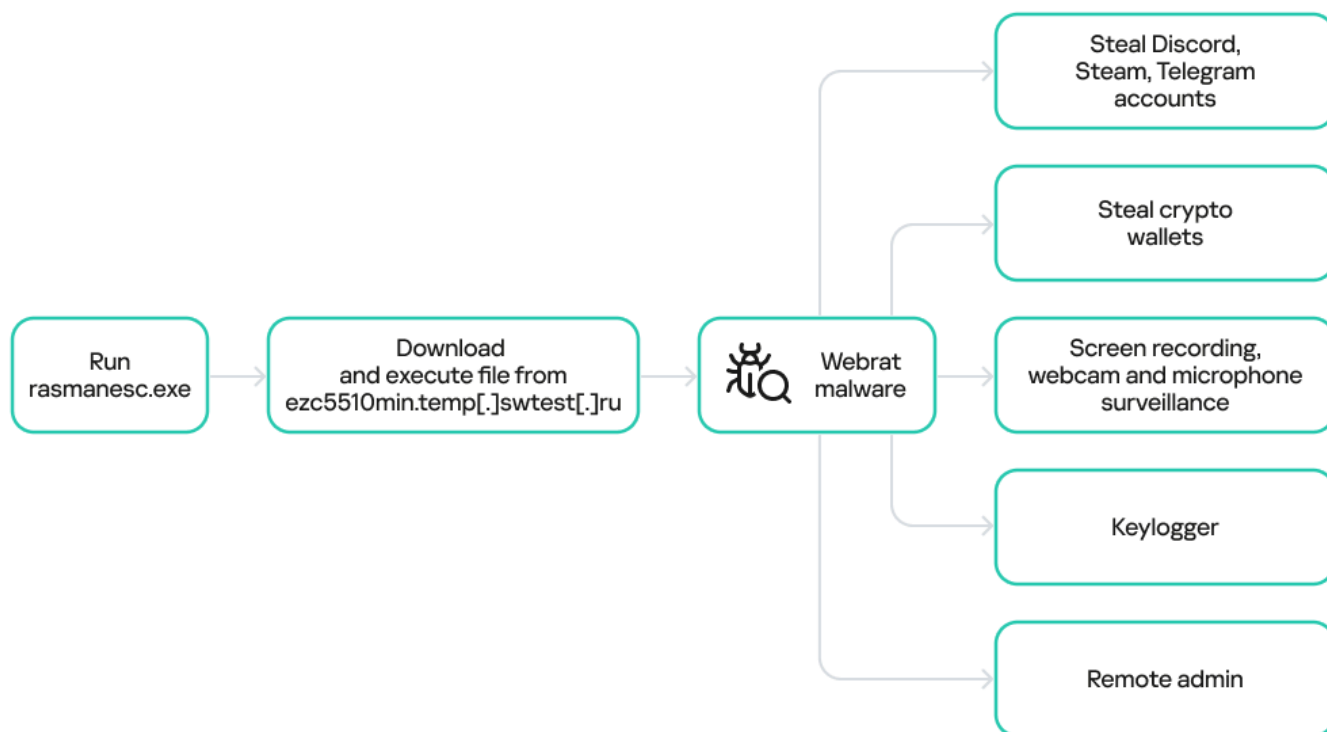
The Download Exploit ZIP link in the Download & Install section leads to a password-protected archive hosted in the same repository. The password is hidden within the name of a file inside the archive.

Name	Size	Packed Size	Modified	Created	Accessed	Attributes	Encrypted	Comment	CRC	Method	Characterist...	Host OS
pass - 8511	0	12	2025-10-20...			AI	+		00000000	ZipCrypto S...	NTFS : Encr...	FAT
payload.dll	55 015	25 030	2025-10-21...			AI	+		D38531FF	ZipCrypto ...	NTFS : Encr...	FAT
rasmanesc.exe	8 591 242	8 398 746	2025-10-20...			AI	+		A6DA783E	ZipCrypto ...	NTFS : Encr...	FAT
start_exp.bat	51	63	2025-10-21...			AI	+		06702282	ZipCrypto S...	NTFS : Encr...	FAT

0 / 4 object(s) selected

The archive downloaded from the repository includes four files:

1. 1 pass – 8511: an empty file, whose name contains the password for the archive.
2. 2 payload.dll: a decoy, which is a corrupted PE file. It contains no useful information and performs no actions, serving only to divert attention from the primary malicious file.
3. 3 rasmanesc.exe (note: file names may vary): the primary malicious file (MD5 61b1fc6ab327e6d3ff5fd3e82b430315), which performs the following actions:
 - Escalate its privileges to the administrator level (T1134.002).
 - Disable Windows Defender (T1562.001) to avoid detection.
 - Fetch from a hardcoded URL (ezc5510min.temp[.]swtest[.]ru in our example) a sample of the Webrat family and execute it (T1608.001).
4. 4 start_exp.bat: a file containing a single command: start rasmanesc.exe, which further increases the likelihood of the user executing the primary malicious file.



The execution flow and capabilities of rasmanesc.exe

Webrat is a backdoor that allows the attackers to control the infected system. Furthermore, it can steal data from cryptocurrency wallets, Telegram, Discord and Steam accounts, while also performing spyware functions such as screen recording, surveillance via a webcam and microphone, and keylogging. The version of Webrat discovered in this campaign is no different from those documented previously.

Campaign objectives

Previously, Webrat spread alongside game cheats, software cracks, and patches for legitimate applications. In this campaign, however, the Trojan disguises itself as exploits and PoCs. This suggests that the threat actor is attempting to infect information security specialists and other users interested in this topic. It bears mentioning that any competent security professional analyzes exploits and other malware within a controlled, isolated environment, which has no access to sensitive data, physical webcams, or microphones. Furthermore, an experienced researcher would easily recognize Webrat, as it's well-documented and the current version is no different from previous ones. Therefore, we believe the bait is aimed at students and inexperienced security professionals.

Conclusion

The threat actor behind Webrat is now disguising the backdoor not only as game cheats and cracked software, but also as exploits and PoCs. This indicates they are targeting researchers who frequently rely on open sources to find and analyze code related to new vulnerabilities.

However, Webrat itself has not changed significantly from past campaigns. These attacks clearly target users who would run the “exploit” directly on their machines — bypassing basic safety protocols. This serves as a reminder that cybersecurity professionals, especially inexperienced researchers and students, must remain vigilant when handling exploits and any potentially malicious files. To prevent potential damage to work and personal devices containing sensitive information, we recommend analyzing these exploits and files within isolated environments like virtual machines or sandboxes.

We also recommend exercising general caution when working with code from open sources, always using reliable security solutions, and never adding software to exclusions without a justified reason.

Kaspersky solutions effectively detect this threat with the following verdicts:

- HEUR:Trojan.Python.Agent.gen
- HEUR:Trojan-PSW.Win64.Agent.gen
- HEUR:Trojan-Banker.Win32.Agent.gen
- HEUR:Trojan-PSW.Win32.Coins.gen
- HEUR:Trojan-Downloader.Win32.Agent.gen
- PDM:Trojan.Win32.Generic

Indicators of compromise

Malicious GitHub repositories

<https://github.com/RedFoxNxploits/CVE-2025-10294-Poc>

<https://github.com/FixingPhantom/CVE-2025-10294>

<https://github.com/h4xnz/CVE-2025-10294-POC>

<https://github.com/usjnx72726w/CVE-2025-59295/tree/main>

<https://github.com/stalker110119/CVE-2025-59230/tree/main>

<https://github.com/moegameka/CVE-2025-59230>

<https://github.com/DebugFrag/CVE-2025-12596-Exploit>

<https://github.com/themaxlpalfaboy/CVE-2025-54897-LAB>

<https://github.com/DExplo1ted/CVE-2025-54106-POC>

<https://github.com/h4xnz/CVE-2025-55234-POC>

<https://github.com/Hazellooks/CVE-2025-11499-Exploit>

<https://github.com/usjnx72726w/CVE-2025-11499-LAB>

<https://github.com/modhopmarrow1973/CVE-2025-11833-LAB>

<https://github.com/rootreapers/CVE-2025-11499>

<https://github.com/lagerhaker539/CVE-2025-12595-POC>

Webrat C2

[http://ezc5510min\[.\]temp\[.\]swtest\[.\]ru](http://ezc5510min[.]temp[.]swtest[.]ru)

[http://shopsleta\[.\]ru](http://shopsleta[.]ru)

MD5

[28a741e9fcd57bd607255d3a4690c82f](#)

[a13c3d863e8e2bd7596bac5d41581f6a](#)

[61b1fc6ab327e6d3ff5fd3e82b430315](#)

From cheats to exploits: Webrat spreading via GitHub

Your email address will not be published. Required fields are marked *

Name *

Email *

This site uses Akismet to reduce spam. [Learn how your comment data is processed.](#)