Hexalocker-v2-being-proliferated-by-Skuld-Stealer

Published: 2025-01-09 · Archived: 2026-04-05 18:58:08 UTC

CRIL analyzes the return of Hexalocker Ransomware in a new version that leverages the Skuld Stealer and other advanced capabilities.

Key Takeaways

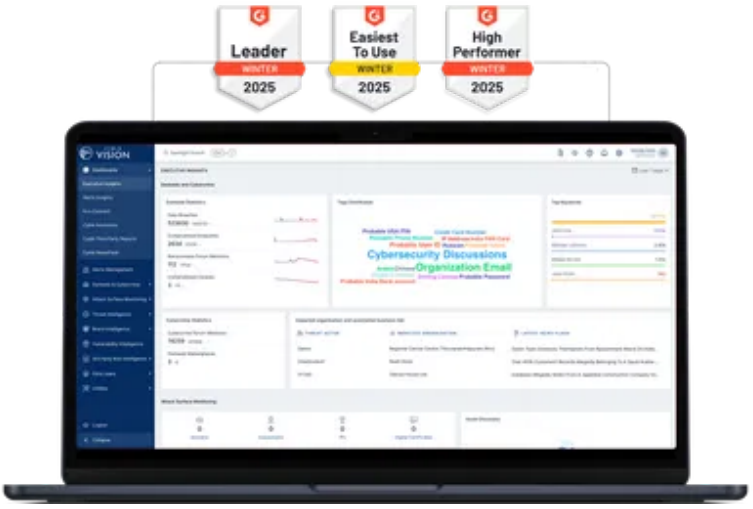
- HexaLocker was first discovered in mid-2024, with version 2 introducing significant updates and enhanced functionalities.
- HexaLocker V2 includes a persistence mechanism that modifies registry keys to ensure continued execution after the affected system reboots.
- The updated version downloads Skuld Stealer, which extracts sensitive information from the victim's system before encryption.
- Unlike its predecessor, HexaLocker V2 exfiltrates victim files before encrypting them, following the double extortion method of data theft and file encryption.
- HexaLocker V2 utilizes a combination of advanced encryption algorithms, including AES-GCM for string encryption, Argon2 for key derivation, and ChaCha20 for file encryption.
- HexaLocker V2 replaces the TOXID communication method with a unique hash, enabling victims to communicate with the Threat Actors' (TA's) site.

Executive Summary

On August 9th, the HexaLocker ransomware group announced a new Windows-based ransomware on their Telegram channel. The post highlighted that the ransomware was developed in the Go programming language and claimed that their team included members from notable groups like LAPSUS$ and others. Following this announcement, researchers from **Synacktiv** analyzed this ransomware variant and published their findings shortly after.

On October 21st, cybersecurity researcher PJ04857920 shared a post on X, revealing that the admin behind HexaLocker had decided to shut down the operation and put the ransomware's source code and web panel up for sale based on information from the HexaLocker group's Telegram channel.

Later, on December 12th, they provided another update on X, stating that the HexaLocker ransomware had been revived, with signs of ongoing development and activity. The Telegram post also mentioned that the upgraded version of HexaLocker would feature enhanced encryption algorithms, stronger encryption passwords, and new persistence mechanisms.



Cyble Research and Intelligence Labs (CRIL) came across a new version of the HexaLocker ransomware. Upon execution, it copies itself to the %appdata% directory, creates a run entry for persistence, encrypts files, and appends the “HexaLockerv2” extension to them.

Prior to encryption, the ransomware also steals the victim’s files and exfiltrates them to a remote server. Notably, in this new version, the ransomware downloads an open-source stealer named Skuld to collect sensitive information from the victim’s machine before encryption. The figure below shows the Hexalocker Ransomware Site used for Victim’s communication.

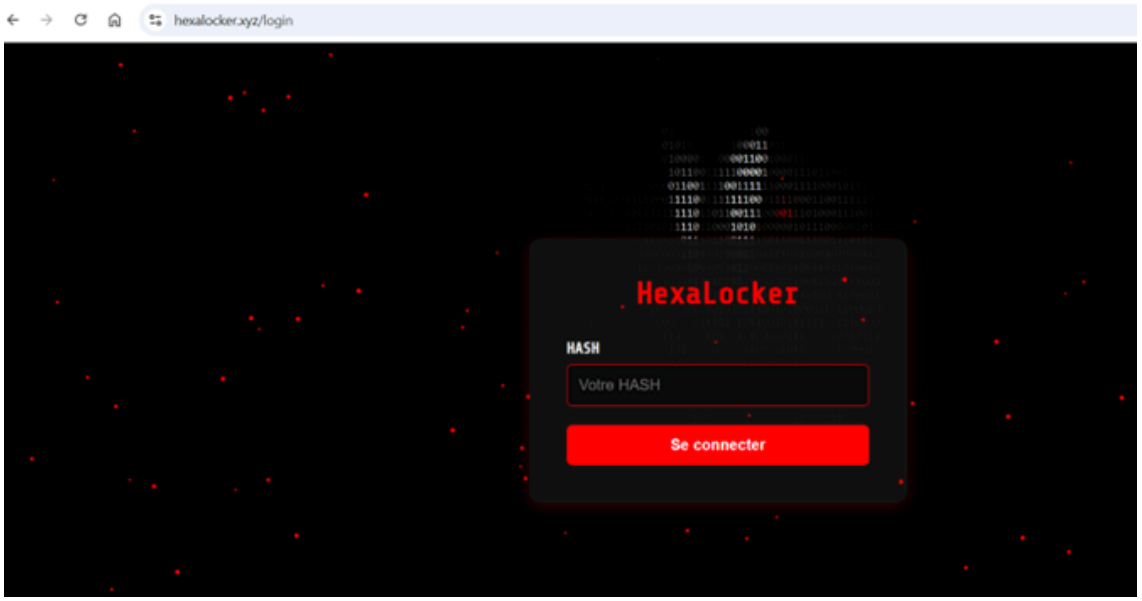


Figure 1 – Ransomware login page

Technical Details

Persistence

Upon execution, the HexaLocker ransomware creates a self-copy named “*myapp.exe*” in the “*%appdata%\MyApp*” directory and establishes persistence by adding an AutoRun entry at

"*HKCU\Software\Microsoft\Windows\CurrentVersion\Run*" with the value "*MyAppAutostart*" ensuring the ransomware binary executes upon system reboot.

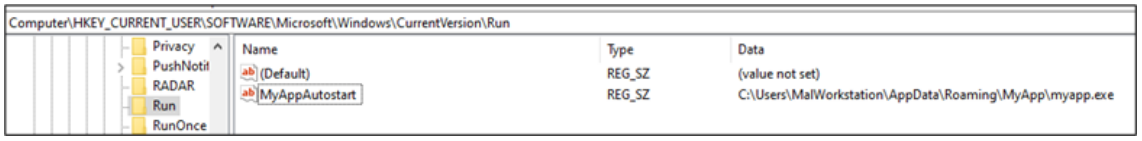


Figure 2 – AutoRun entry

Obfuscation

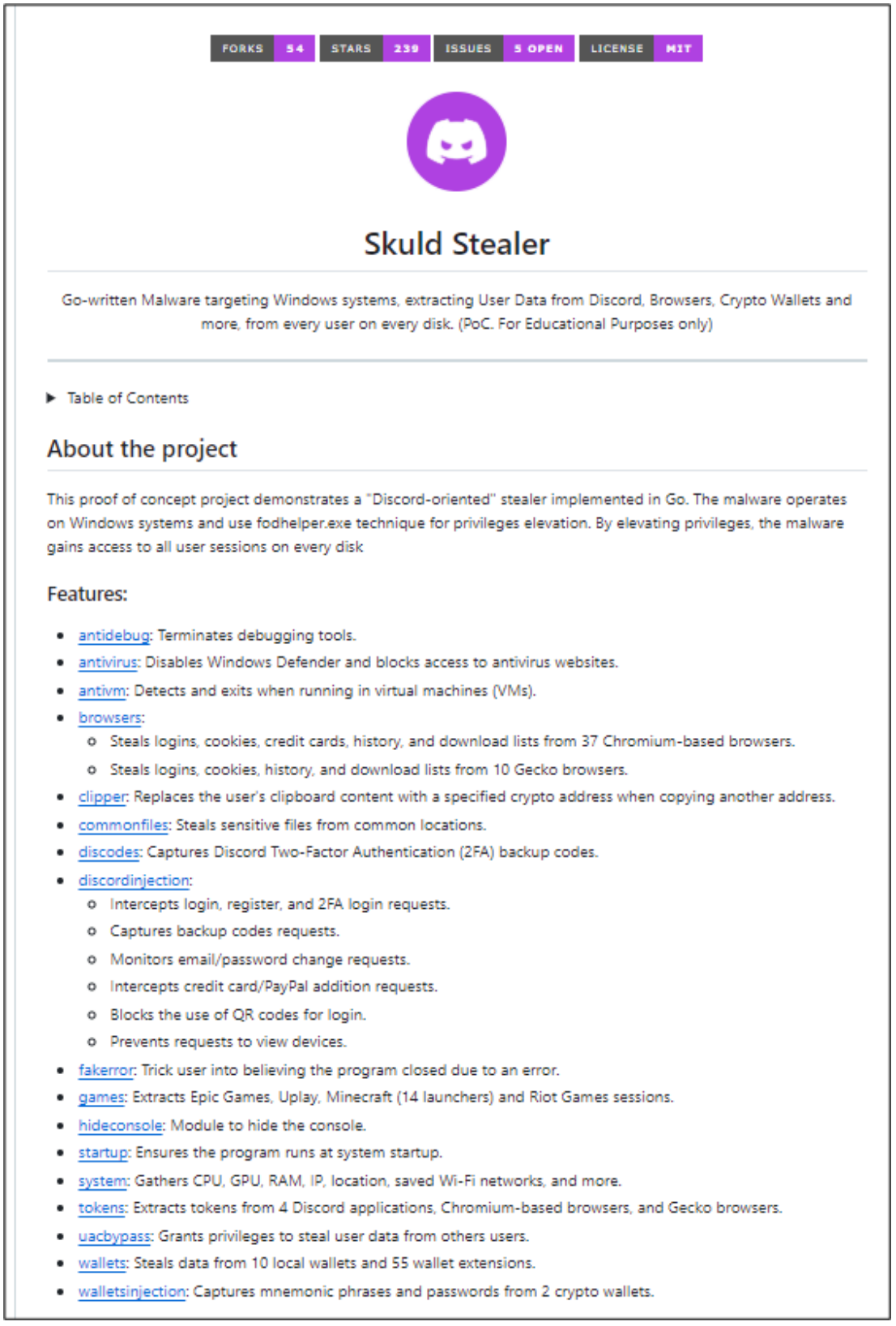
All string references, including the Stealer URL, file paths, folder names, environment variable names, WMIC commands, and ransom notes, are generated during runtime through multiple layers of AES-GCM decryption. This approach effectively obfuscates the strings, making them harder to detect by security solutions. In contrast, all strings in the previous version were statically visible.

Figure 3 – String Decryption

Stealer

Prior to initiating the encryption process, the ransomware downloads a stealer binary, a Go-compiled program, from the URL hxxps[:]//hexalocker.xyz/SGDYSRE67T43TVD6E5RD[.]exe and executes it from the current directory. This stealer functionality was absent in the previous version of HexaLocker.

The downloaded stealer, identified as Skuld, is an open-source tool designed to target Windows systems and steal user data from various applications such as Discord, browsers, crypto wallets, and more.



FORKS 54 STARS 239 ISSUES 5 OPEN LICENSE MIT

Skuld Stealer

Go-written Malware targeting Windows systems, extracting User Data from Discord, Browsers, Crypto Wallets and more, from every user on every disk. (PoC. For Educational Purposes only)

▶ Table of Contents

About the project

This proof of concept project demonstrates a "Discord-oriented" stealer implemented in Go. The malware operates on Windows systems and use fodhelper.exe technique for privileges elevation. By elevating privileges, the malware gains access to all user sessions on every disk

Features:

- antidebug: Terminates debugging tools.
- antivirus: Disables Windows Defender and blocks access to antivirus websites.
- antivm: Detects and exits when running in virtual machines (VMs).
- browsers:
  - Steals logins, cookies, credit cards, history, and download lists from 37 Chromium-based browsers.
  - Steals logins, cookies, history, and download lists from 10 Gecko browsers.
- clipper: Replaces the user's clipboard content with a specified crypto address when copying another address.
- commonfiles: Steals sensitive files from common locations.
- discodes: Captures Discord Two-Factor Authentication (2FA) backup codes.
- discordinjection:
  - Intercepts login, register, and 2FA login requests.
  - Captures backup codes requests.
  - Monitors email/password change requests.
  - Intercepts credit card/PayPal addition requests.
  - Blocks the use of QR codes for login.
  - Prevents requests to view devices.
- fakerror: Trick user into believing the program closed due to an error.
- games: Extracts Epic Games, Uplay, Minecraft (14 launchers) and Riot Games sessions.
- hideconsole: Module to hide the console.
- startup: Ensures the program runs at system startup.
- system: Gathers CPU, GPU, RAM, IP, location, saved Wi-Fi networks, and more.
- tokens: Extracts tokens from 4 Discord applications, Chromium-based browsers, and Gecko browsers.
- uacbypass: Grants privileges to steal user data from others users.
- wallets: Steals data from 10 local wallets and 55 wallet extensions.
- walletsinjection: Captures mnemonic phrases and passwords from 2 crypto wallets.

Figure 4 – Skuld Stealer's features

In this case, the TA has utilized only the browser module from the many available in the open-source Skuld Stealer. The image below shows function names corresponding only to the browser module from the Skuld project.

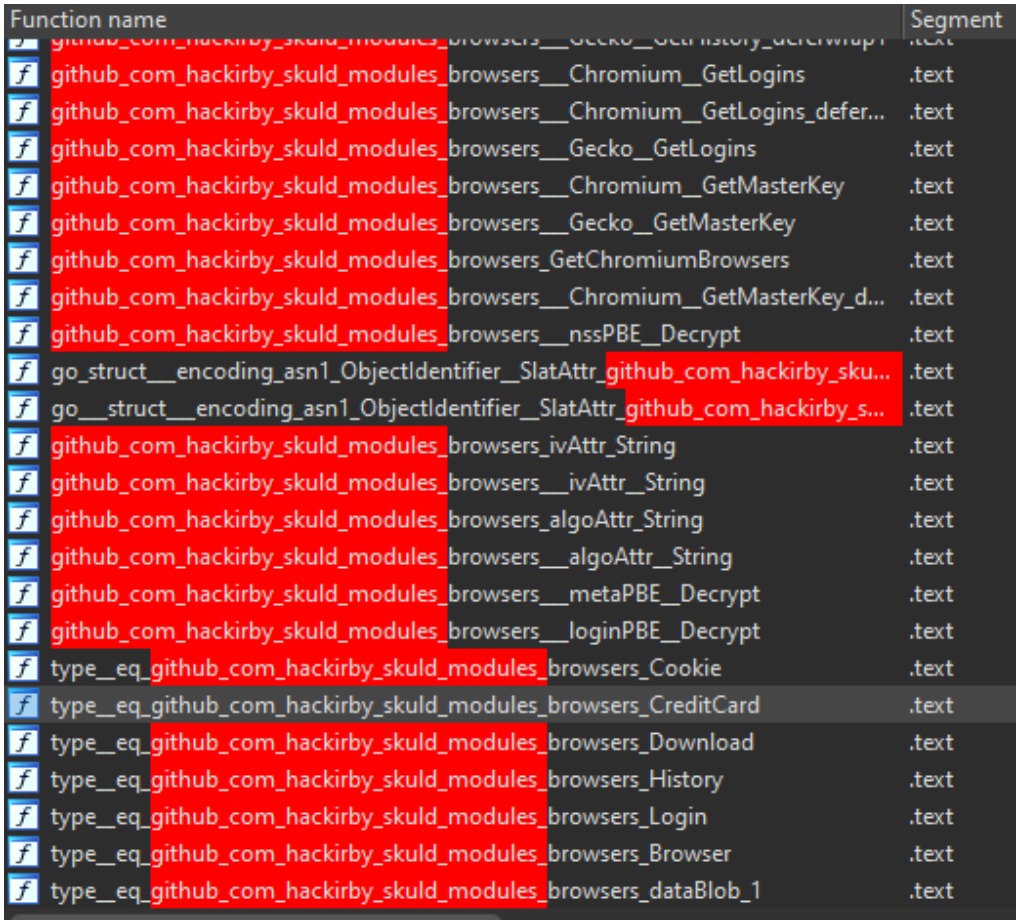


Figure 5 – Browser modules

The stealer collects various sensitive data stored by Chromium and Gecko-based browsers, such as cookies, saved credit card information, downloads, browsing history, and login credentials. Skuld Stealer targets the following web browsers in this campaign.

Gecko-based browsers

Firefox	SeaMonkey
Waterfox	K-Meleon
Thunderbird	IceDragon
Cyberfox	BlackHaw
Pale Moon	mercury

Chromium browsers

Chrome SxS	ChromePlus	7Star
Chrome	Chedot	Vivaldi
Kometa	Elements Browser	Epic Privacy Browser

Uran	Fenrir Inc	Citrio
Coowon	liebao	QIP Surf
Orbitum	Dragon	360Browser
Maxthon3	K-Melon	CocCoc
BraveSoftware	Amigo	Torch
Sputnik	Edge	DCBrowser
YandexBrowser	UR Browser	Slimjet
Opera		

The stolen data is compressed into a ZIP archive named ‘BrowsersData-*.zip’ and stored in the AppData\Local\Temp directory before being exfiltrated to the remote server “*hxxps://hexalocker[.]xyz/upload.php”*. The image below shows the console output of the stealer upon completing each stage.

Figure 6 – Stealer Console Output

Exfiltration

Upon executing the stealer payload, the ransomware exfiltrates the victims’ files by scanning all folders starting from “*C:*” to find files with extensions matching those listed in the table below. The identified files are compiled into a single ZIP archive named “*data_*.zip*”, stored in the “*%localappdata%\DataHexaLocker*” directory, and subsequently transmitted to the attacker’s remote server via “*hxxps[:]//hexalocker.xyz/receive.php*”.

Category	File Types
Documents	.pdf, .doc, .docx, .rtf, .txt, .wps, .xls, .xlsx, .csv, .ppt, .pot, .xps, .xsd, .xml
Images	.jpg, .jpeg, .png, .bmp, .gif, .tif, .tiff, .ico, .jpe, .dib, .raw, .psd, .exr, .bay
Audio	.mp3, .wav, .wma, .m4a, .m4p, .flac, .aac, .amr, .ogg, .adp
Video	.mp4, .mkv, .avi, .mov, .wmv, .flv, .3gp, .m4v, .amv, .swf
Compressed Files	.zip, .rar, .7z, .tar, .gz, .bz2, .cab, .iso, .lzh, .ace, .arj
Code & Scripts	.php, .asp, .htm, .html, .js, .jsp, .css, .py, .java, .c, .cpp, .asm, .vbs, .cmd, .bat
Executable Files	.exe, .msi, .dll, .apk, .lnk

Database Files	.db, .dbf, .mdb, .sql, .odc, .odm, .pst, .mdf, .myi, .tab
3D/Design Files	.3ds, .dae, .stl, .max, .dwg, .dxf, .obj, .r3d, .kmz, .opt
Web/Markup Files	.html, .htm, .xml, .xsl, .rss, .cfm, .xsf
System/Backup Files	.bak, .cer, .crt, .pfx, .p12, .p7b, .log, .cfg, .ini, .lnk
Others	.sum, .sln, .dif, .dmg, .p7c, .opt, .sie, .key, .vob

Encryption

The ransomware generates a key and the salt needed for encryption and sends them to a remote server at "*hxxps[:]//hexalocker.xyz/index[.]php*," along with host-specific details such as the IP address, computer name, and ID. This information is used to identify the victims and facilitate the recovery of the encrypted files.

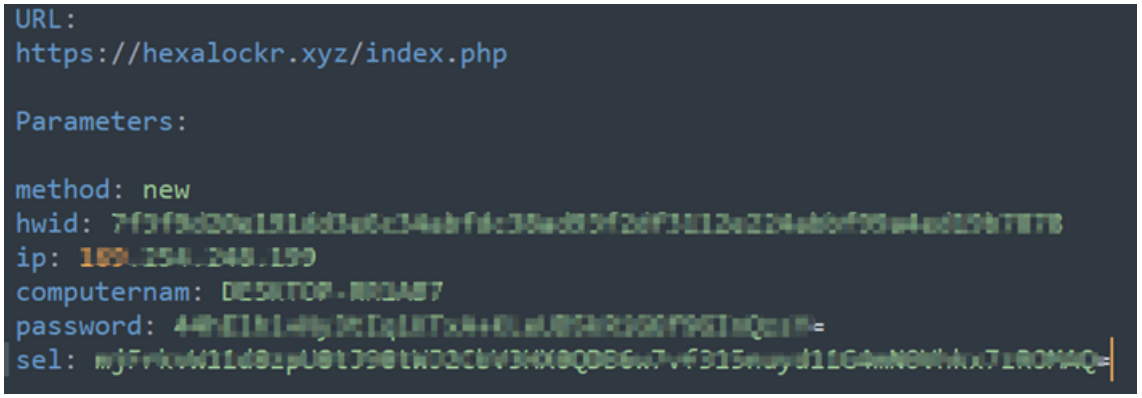


Figure 7 – Victim's Details

Once the gathered information is transmitted to the attacker, HexaLocker proceeds to scan the "C:\Users<username>" directory on the victim's machine. It searches for files that match a specific set of extensions, as listed in the table below.

Category	**Extensions**
Text Documents	.txt, .doc, .odt, .rtf, .wps, .dot
Databases	.sql, .mdb, .dbf, .pdb, .mdf, .mdw, .myi
Spreadsheets	.xls, .ods, .csv, .xla, .xlw, .xlm, .xlt, .slk
Presentations	.ppt, .odp, .pps, .pot
Programming Files	.cpp, .css, .php, .asp, .ini, .inc, .obj, .bat, .cmd, .vbs, .jsp, .asm, .cfm
Archives	.zip, .rar, .tar, .iso, .bz2, .cab, .lzh, .ace, .arj

Images	.jpg, .png, .bmp, .gif, .tif, .ico, .psd, .raw, .svg, .jpe, .dib, .iff, .dcm, .bay, .dcr, .nef, .orf, .r3d
Audio	.mp3, .mka, .m4a, .wav, .wma, .flv, .pls, .adp
Video	.mp4, .mkv, .avi, .mov, .wmv, .3gp, .m4v, .amv, .m4p, .vob, .mpv, .3g2, .f4v, .m1v
Web Files	.htm, .html, .xml, .css, .js, .jsp, .rss
Executables	.exe, .jar, .msi, .dll
Scripts	.php, .asp, .vbs, .cmd, .bat
Backup/Logs	.bak, .log
3D/CAD	.3ds, .dae, .dwg, .max, .geo
Compressed	.zip, .rar, .tar, .bz2, .gz
Configuration	.ini, .cfg, .xml
Emails	.msg, .oft, .pst, .dbx
Fonts	.ttf, .otf, .woff
Certificates	.crt, .cer, .pfx, .p12, .p7b, .p7c
Others	.lnk, .dat, .sum, .opt, .dic, .tbi, .xps, .key, .tab, .stm, .ai3, .ai4, .ai5, .ai6, .ai7, .ai8, .opt

The ransomware reads the content of the original file and uses the ChaCha20 algorithm to encrypt the data. Once the encryption is complete, it creates a new file with the “.HexaLockerV2” extension and writes the encrypted content to this newly created file. The ransomware then proceeds to delete the original file using the os.Remove function, leaving only the encrypted file behind. The figure below shows the chacha20 encryption algorithm used by the ransomware binary.

```
loc_686106:
mov     [rsp+1F0h+var_C0], rax
lea     rax, unk_6AEF80
mov     rbx, [rsp+1F0h+var_1B8]
mov     rcx, rbx
nop     dword ptr [rax]
call    runtime_makeslice
mov     [rsp+1F0h+var_18], rax
mov     rbx, rax
mov     rcx, [rsp+1F0h+var_1B8]
mov     rdi, rcx
mov     rsi, [rsp+1F0h+var_B8]
mov     r8, rcx
mov     r9, [rsp+1F0h+var_1B0]
mov     rax, [rsp+1F0h+var_C0]
call    golang_org_x_crypto_chacha20___Cipher__XORKeyStream
xor     eax, eax
mov     rbx, [rsp+1F0h+arg_0]
mov     rcx, [rsp+1F0h+arg_8]
lea     rdi, aHexalockerv2 ; ".HexaLockerV2"
mov     esi, 0Dh
call    runtime_concatstring2
mov     [rsp+1F0h+var_B0], rax
mov     [rsp+1F0h+var_1A8], rbx
mov     rdi, [rsp+1F0h+var_1B8]
lea     rdx, [rdi+18h]
cmp     rdx, 18h
ja      short loc_6861A3
```

Figure 8 – Chacha20 Algorithm

The figure below illustrates the files encrypted by the HexaLocker Ransomware, which have the “.HexaLockerV2” extension.

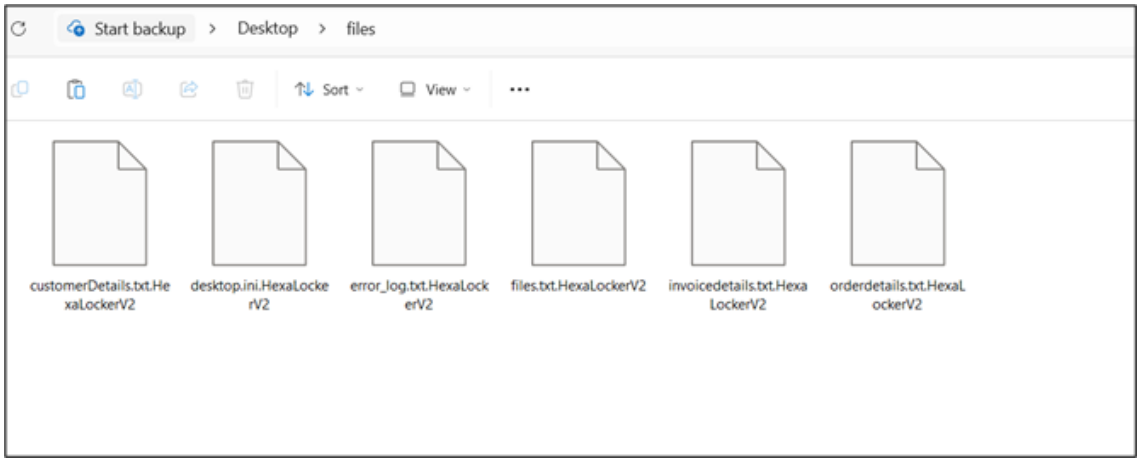


Figure 9 – User files after encryption

Finally, the ransomware displays a ransom note to the victim, instructing them to contact the TA through their communication channels, such as Signal, Telegram, and Web Chat, as shown below.

```
                                                                HexaLocker | Lock. Demand. Dominate. | Since 2024

- Your data has been stolen and encrypted.
- If you do not pay the ransom, your data will be published online.


>>>> Important for employees

If you are an employee who has triggered the ransomware, do not panic. Mistakes happen, and frankly, you're not the first, nor will you be the la

Attempting to hide this from your superiors would be futile—they will find out. Instead, show professionalism by bringing this to their attention
Delays in addressing the issue only prolong the inevitable and give your company less time to act.

Remember, we are the best at what we do. Your company's data is in capable hands : ours. The sooner your management is involved, the faster the s


>>>> Why trust us?

We are not motivated by political reasons, personal vendettas, or any other agenda. We are only interested in money.
We understand this situation is unpleasant, and we sincerely apologize for the inconvenience caused.

If you pay, we will provide the decryption tools and delete your data.
We have no interest in scamming you. If we fail to deliver on our promises, no one will trust us in the future.

Our reputation is critical: we target companies worldwide, and no payer has ever been dissatisfied.
To reassure you, we offer you the ability to decrypt one file for free using your personal hash.

Contact us through our communication channels and wait for our response. We always reply, although there may be occasional delays due to the high

- Signal ID : HexaLocker.01
- Telegram Username : @HexaLockerSupp
- Web Chat (contact here): https://hexalocker.xyz/login


Your personal HASH: [redacted]
```

Figure 10 – Ransom note

The ransom note contains a unique personal hash, which the victim uses to communicate with the TA through a chat window provided by the attacker, as shown below.

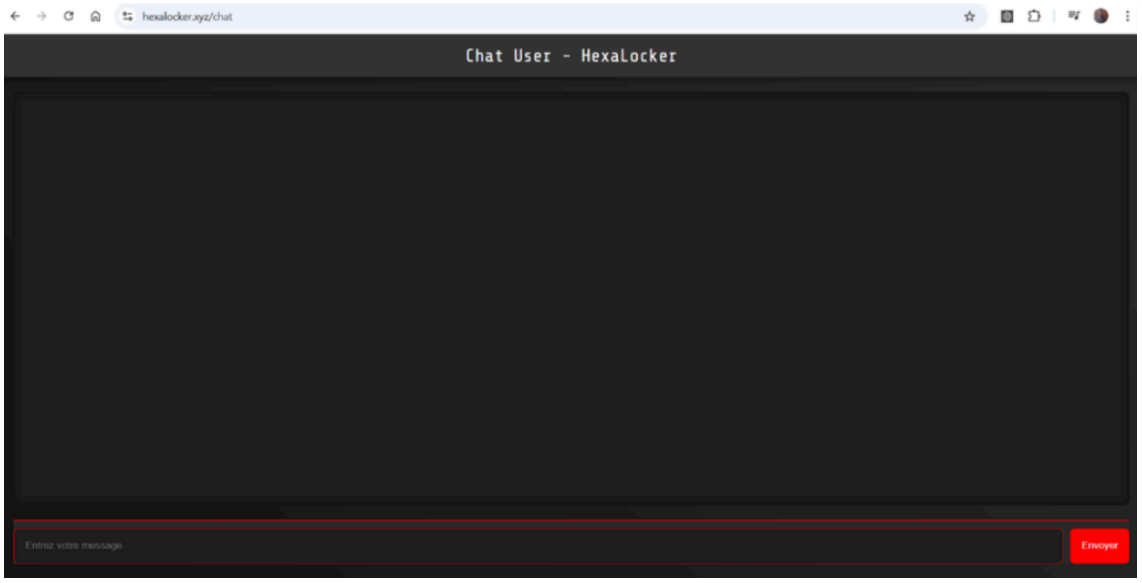


Figure 11 – Web Chat Window

Conclusion

The new version of HexaLocker ransomware represents a significant upgrade, incorporating enhanced encryption logic and a customized stealer component. Developed in Go, this ransomware benefits from Go’s efficiency, making it more challenging to detect by endpoints.

Before initiating the encryption process, the ransomware employs the Skuld stealer to collect sensitive information from the victim’s machine. This strategic combination of the Skuld stealer and the ransomware highlights the continuous evolution and sophistication of the HexaLocker group, posing an ongoing threat to targeted systems.

The Yara rule to detect HexaLocker Version 2 is available for download from the linked Github repository.

Our Recommendations

We have listed some essential cybersecurity best practices that create the first line of control against attackers. We recommend that our readers follow the best practices given below:

Safety Measures to Prevent Ransomware Attacks

- Regularly back up important files to offline or cloud storage, ensuring they are stored securely and not connected to the main network.
- Enable automatic updates for your operating system, applications, and security software to ensure you receive the latest patches and security fixes.
- Implement endpoint protection with reputable anti-virus and anti-malware software to detect and block potential ransomware threats.
- Educate employees or users about phishing attacks and suspicious email links, which are common ransomware delivery methods.
- Restrict user privileges and avoid running unnecessary services to minimize the attack surface, ensuring users only have access to the resources they need.

MITRE ATT&CK® Techniques

Tactic	Technique ID	Procedure
Execution (TA0002)	User Execution (T1204.002)	User executes the ransomware file.
Persistence (TA0003)	Registry Run Keys / Startup Folder (T1547.001)	Adds a Run key entry for execution on reboot.
Defense Evasion (TA0005)	Deobfuscate/Decode Files or Information (T1140)	Ransomware Decrypts strings using the AES algorithm
Discovery (TA0007)	File and Directory Discovery (T1083)	Ransomware enumerates folders for file encryption and file deletion.
Impact (TA0040)	T1486 (Data Encrypted for Impact)	Ransomware encrypts files for extortion.
Credential Access (TA0006)	Credentials from Password Stores: Credentials from Web Browsers (T1555.003)	Retrieves passwords from Login Data
Credential Access (TA0006)	Steal Web Session Cookie (T1539)	Steals browser cookies

Collection (TA0009)	Archive via Utility (T1560.001)	Zip utility is used to compress the data before exfiltration
Exfiltration (TA0010)	Exfiltration Over C2 Channel (T1041)	Exfiltration Over C2 Channel

Indicators of Compromise (IOCs)

Indicators	Indicator Type	Description
8b347bb90c9135c185040ef5fdb87eb5cca821060f716755471a637c350988d8	SHA-256	Stealer
0347aa0b42253ed46fdb4b95e7ffafa40ba5e249dfb5c8c09119f327a1b4795a	SHA-256	HexaLockerV2
28c1ec286b178fe06448b25790ae4a0f60ea1647a4bb53fb2ee7de506333b960	SHA-256	HexaLockerV2
d0d8df16331b16f9437c0b488d5a89a4c2f09a84dec4da4bc13eab15aded2e05	SHA-256	HexaLockerV2
hxxps[:]//hexalocker.xyz/SGDYSRE67T43TVD6E5RD[.]exe	URL	Stealer download url
hxxps[:]//hexalocker[.]xyz/upload[.]php	URL	NA
hxxps[:]//hexalocker[.]xyz/receive[.]php	URL	NA

References

https://www.trellix.com/en-in/blogs/research/skuld-the-infostealer-that-speaks-golang

https://www.synacktiv.com/publications/lapsus-is-dead-long-live-hexalocker.html

---