IBIMA Publishing
Communications of International Proceedings
https://ibimapublishing.com/p-articles/46SP/2025/4648425/
Vol. 2025 (37), Article ID 4648425

Infostealers – Research Objectives*

Michał GLET

Military University of Technology, Faculty of Cybernetics, Poland

Correspondence should be addressed to: Michał GLET, michal.glet@wat.edu.pl

* Presented at the 46th IBIMA International Conference, 26-27 November 2025, Ronda, Spain

Abstract

The growing prevalence of infostealer malware poses a significant threat to both individual users and organizations, leading to the theft of login credentials, cryptocurrency wallets, multi-factor authentication tokens, browsing history, and other sensitive data. Despite the widespread use of the Windows Data Protection API (DPAPI) to secure data at rest, recent attacks demonstrate that this mechanism can be circumvented when malware operates within the same user context. The existing literature lacks a comprehensive analysis of how modern infostealer families exploit DPAPI and of the effectiveness of detection strategies against such threats. This study addresses this gap by systematically analyzing the behavior of selected infostealer malware—Vidar Stealer, Raccoon Stealer, RedLine Stealer, Aurora Stealer, and Lumma Stealer. The research focuses on their methods of abusing DPAPI to extract sensitive information from web browsers and credential stores. The findings reveal common techniques employed by these malware families and highlight the limitations of current data protection approaches. Based on the results, the study proposes research objectives to develop detection mechanisms capable of identifying both known and novel infostealer variants, and to evaluate alternative, more secure methods for protecting sensitive data at rest on Windows systems.

Keywords: infostealer, Windows Data Protection API, DPAPI, CryptProtectData, CryptUnprotectData

Introduction

Infostealer malware has evolved over the past few years into a key component of modern cybercrime. Instead of encrypting or destroying user data, infostealers mainly focus on silently stealing user credentials, cryptocurrency wallets, multi-factor authentication tokens, web browsing history, and other sensitive data. Recent threat intelligence reports [1][2] show a steady rise in infections. Families such as RedLine, Vidar, Raccoon, Aurora and Lumma are offered by cybercriminals through a Malware-as-a-Service (MaaS) model [3]. This is another factor which stimulates growth. In many cases, infostealers target data at rest, which on the Windows platform is often protected with the Windows Data Protection API (DPAPI). The studies showed that most modern web browsers behave this way. The goal of this work is to focus specifically on the intersection between Windows DPAPI and infostealer malware. First, the paper introduces the fundamentals of DPAPI and illustrates its widespread use to protect sensitive data in web browser software. Next, it presents a summary of the analysis of selected infostealers, highlighting how each family leverages DPAPI to recover credentials and other secrets. Finally, the paper formulates research objectives in three areas: understanding in detail how infostealers abuse DPAPI in practice, designing detection mechanisms that can identify such abuse even for previously unseen variants, and exploring more secure and generic approaches to storing data at rest on Windows.

Cite this Article as: Michał GLET, Vol. 2025 (37) "Infostealers – Research Objectives " Communications of International Proceedings, Vol. 2025 (37), Article ID 4648425, https://doi.org/10.5171/2025.4648425

Basics Of The DPAPI

The Windows operating system provides developers with the Data Protection API (DPAPI) [4]. This is a set of structures and functions that allow for data encryption/decryption based on the login context in which the process runs. This API hides the problem of key management. The keys are automatically generated and managed by the operating system in the context of the logged-in user. As a result, only processes running in the same login context can decrypt secured data. What is more, the encryption process is not only tied to the login context. For non-roaming users, it is also tied to the specific computer. This means that data can be decrypted only by a process running in the same login context and on the same machine. If needed, encryption can be made without a login context. In this case, all processes running on the same computer can do the decryption.

Two main functions provided by the DPAPI are *CryptProtectData* and *CryptUnprotectData*. *CryptProtectData* performs encryption in the user/machine context, and *CryptUnprotectData* performs decryption.

```
DPAPI_IMP BOOL CryptProtectData(
  [in]           DATA_BLOB                 *pDataIn,
  [in, optional] LPCWSTR                   szDataDescr,
  [in, optional] DATA_BLOB                 *pOptionalEntropy,
  [in]           PVOID                     pvReserved,
  [in, optional] CRYPTPROTECT_PROMPTSTRUCT *pPromptStruct,
  [in]           DWORD                     dwFlags,
  [out]          DATA_BLOB                 *pDataOut
);
```

Picture 1. CryptProtectData – C++ declaration of the function

```
DPAPI_IMP BOOL CryptUnprotectData(
  [in]           DATA_BLOB                 *pDataIn,
  [out, optional] LPWSTR                   *ppszDataDescr,
  [in, optional]  DATA_BLOB                *pOptionalEntropy,
                  PVOID                    pvReserved,
  [in, optional]  CRYPTPROTECT_PROMPTSTRUCT *pPromptStruct,
  [in]           DWORD                     dwFlags,
  [out]          DATA_BLOB                 *pDataOut
);
```

Picture 2. CryptUnpotectData – C++ declaration of the function

As depicted in the pictures 1 and 2, those functions don't take cryptographic keys as input arguments. The only required data is the input data (in bytes) and the DWORD value that specifies the behaviour (e.g., whether to run in user/machine context). This makes using DPAPI very simple and straightforward. All underlying cryptographic operations, such as key generation or data processing, are handled by Windows.

CryptProtectData, as stated in the official documentation [5], derives a cryptographic key, encrypts data and calculates a Message Authentication Code (MAC) to ensure data integrity. Microsoft officially is not providing any information on what exact algorithms are being used. However, there are some public posts [6] indicating that, since Windows 7, DPAPI has used AES-256 in CBC mode for encryption, HMAC-SHA2-512 for integrity checks, and PBKDF2 for key generation.

Sample DPAPI Use-Cases

The Windows DPAPI is often used by software vendors to secure data at rest (on disk). One of the most common example are web browsers. They are storing highly sensitive user data, including authentication tokens, passwords, credit card details, and cookies. As of now, the majority of them, including Google Chrome, Microsoft Edge (classic and Chromium-based), Brave, Opera, and Internet Explorer, use DPAPI.

Web Brower	DPAPI use-cases
Google Chrome (version < 80)	Used directly to secure sensitive user data, like passwords. [7]
Google Chrome (version >= 80)	Used indirectly to secure sensitive user data, like passwords. Data are secured using a randomly generated key. This key is secured using DPAPI. [8]
Microsoft Edge (Chromium-based)	Used indirectly to secure sensitive user data, like passwords. Data are secured using a randomly generated key. This key is secured using DPAPI. [9]
Microsoft Edge (classic)	Used directly to secure sensitive user data, like passwords.
Brave	Used indirectly to secure sensitive user data, like passwords. Data are secured using a randomly generated key. This key is secured using DPAPI. [10]
Internet Explorer	Used directly to secure sensitive user data, like passwords.
Opera	Used indirectly to secure sensitive user data, like passwords. Data are secured using a randomly generated key. This key is secured using DPAPI. [11]

As we can see, nowadays, DPAPI is most often used indirectly to secure users' data. This means that infostealers, in addition to calling *CryptUnprotectData*, must take further steps to steal secrets. However, the only real layer of security is provided by DPAPI, which is insufficient when malware runs in the same user context as a web browser. Switching from direct to indirect use made recovery a little harder, but it was still doable [12].

Infostealers

The practical part of my research focused on selected infostealer malware used by cybercriminals over the past few years. The samples were obtained from the MalwareBazaar website. To assess the use of the Windows Data Protection API, static and dynamic analysis of the samples were conducted in IDA (Interactive Disassembler) within a controlled test environment. Some of my findings are briefly summarized below.

Infostealer	Example of the DPAPI usage
Vidar Stealer	• Steal passwords for Outlook accounts. • Steal sensitive data from web browsers.
Raccoon Stealer	• Steal sensitive data from web browsers. • Steal credentials from Credential Manager.
RedLine Stealer	• Steal sensitive data from web browsers.
Aurora Stealer	• Steal sensitive data from web browsers.
Lumma Stealer	• Steal sensitive data from web browsers.

As we can see, every analyzed infostealer was using DPAPI calls to steal sensitive data from web browsers.

Mitre Att&Ck T1555

MITRE ATT&CK is a publicly available database of tactics and techniques used by cybercriminals. The descriptions are based on real-life attacks. It is very often used to model, detect and analyze cyber threats. It contains technique T1555 [13] called “Credentials from Password Stores” that is a part of a tactic called “Credential Access”. T1555 describes retrieving credentials data from different credentials stores. For example, T1555.003 concerns obtaining credentials from web browsers, and T1555.004 concerns obtaining credentials from Windows Credential Manager. Proposed detection strategies could be incorporated into the mechanism for detecting infostealer activity.

Conclusion And Research Objectives

Having done previous research [14][15][16][17][18] and a PhD on ransomware and its detection, the author shifted the focus to infostealers. In this paper, he described current findings and the paths he plans to research in the near future. Infostealers are gaining increasing traction, often targeting entire corporations. They have become a highly profitable way for cybercriminals to make money. That is why it is very important to research them thoroughly and find an effective way to detect and defend against them.

Current research objectives focus on infostealers in three different areas:

1. To systematically characterize the behaviour of infostealer malware families on Windows, with particular emphasis on their interaction with password stores and the Windows Data Protection API (DPAPI).
2. To propose, design, and implement behavioural detection mechanisms capable of identifying both known and unknown infostealer variants that abuse DPAPI and related credential-storage mechanisms.
3. To investigate and evaluate alternative, more secure and generic approaches for protecting sensitive data at rest on Windows, and to compare them against DPAPI in terms of security guarantees, deployability, and resilience to infostealer attacks.

All results obtained in the research will be compared and referenced against the latest state-of-the-art achievements and scientific publications in the field.

References

- IBM X-Force Threat Intelligence Index, https://www.ibm.com/think/x-force/2024-x-force-threat-intelligence-index, accessed 10.2025
- Check Point – "Infostealers – How to Prevent and Mitigate?", https://www.checkpoint.com/cyber-hub/threat-prevention/what-is-malware/infostealers/, accessed 10.2025
- Check Point – "Malware-as-a-Service (MaaS): Cybercrime's Subscription Model", https://www.checkpoint.com/cyber-hub/threat-prevention/what-is-malware/malware-as-a-service-maas/, accessed 10.2025
- Microsoft – "CNG DPAPI", https://learn.microsoft.com/en-us/windows/win32/seccng/cng-dpapi, accessed 10.2025
- CryptProtectData function, https://learn.microsoft.com/en-us/windows/win32/api/dpapi/nf-dpapi-cryptprotectdata, accessed 10.2025
- DPAPI Secrets. Security analysis and data recovery in DPAPI (Part 1), https://www.passcape.com/index.php?cmd=details&id=20&section=blog, accessed 10.2025
- Source code, https://chromium.googlesource.com/chromium/src/%2B/112.0.5615.165/components/os_crypt/os_crypt_win.cc, accessed 11.2025
- Source code, https://github.com/agentzex/chrome_v80_password_grabber, accessed 11.2025
- Microsoft Edge password manager security, https://learn.microsoaft.com/en-us/deployedge/microsoft-edge-security-password-manager-security, accessed 11.2025
- Sensitive data storage, https://support.brave.app/hc/en-us/articles/29808985123085-Sensitive-data-storage, accessed 11.2025
- Chromium kernel browser cookies and password extraction, https://cn-sec.com/archives/1740857.html, accessed 11.2025
- Web Browser Stored Credentials, https://pentestlab.blog/2024/08/20/web-browser-stored-credentials/, accessed 11.2025
- Credentials from Password Stores, https://attack.mitre.org/techniques/T1555/, accessed 11.2025
- Bajera, J., Glet, M.. Ransomware Attack on the QNAP Device – The Case Study, 41th IBIMA Conference, Seville, Spain, 06.2023
- Glet, M., Kaczyński, K. (2022). POSTER: Ransomware Detection Mechanism – Current State of the Project. In: Zhou, J., *et al.* Applied Cryptography and Network Security Workshops. ACNS 2022. Lecture Notes in Computer Science, vol 13285. Springer, Cham. https://doi.org/10.1007/978-3-031-16815-4_36

- Glet M., Kaczyński K., POSTER: Ransomware Detection Mechanism – Project Status at the Beginning of 2023, Lecture Notes in Computer Science, Springer Nature Switzerland, 2023, pp. 659–663, doi: 10.1007/978-3-031-41181-6_35
- Glet M., Kaczyński K., Ransomware and Honeypots, Proceedings of the 38th International Business Information Management Association Conference (IBIMA), 2021.
- Bajera, J., Glet, M.. Detection of cryptographic functions within binary executable ransomware files, 43rd IBIMA Conference, Seville, Spain, 06.2024