Scientific Research Publishing

Journal of Information Security, 2025, 16(2), 283-312
https://www.scirp.org/journal/jis
ISSN Online: 2153-1242
ISSN Print: 2153-1234

On Threat Models for Information-Stealing Malware (ISM) Targeting Password Managers

Vedika Sunil Bang, Vijay Madisetti*

School of Cybersecurity and Privacy, Georgia Institute of Technology, Atlanta, USA
Email: vedikabang@gatech.edu, *vkm@gatech.edu

How to cite this paper: Bang, V.S. and Madisetti, V. (2025) On Threat Models for Information-Stealing Malware (ISM) Targeting Password Managers. *Journal of Information Security*, **16**, 283-312.
https://doi.org/10.4236/jis.2025.162015

Received: January 15, 2025
Accepted: April 14, 2025
Published: April 17, 2025

Abstract

Information-stealing malware (ISM) is redefining the cybersecurity threat landscape, particularly through its integration into the malware-as-a-service (MaaS) ecosystem. Traditional threat models, while effective against generic vulnerabilities, struggle to keep pace with the sophisticated and evolving tactics of ISMs. These advanced threats exploit software vulnerabilities, bypass conventional defenses, and thrive on usability-security trade-offs, leaving critical systems exposed. This research delves into the intricate attack vectors of ISMs, uncovering gaps in existing frameworks. By leveraging the MITRE ATT&CK framework and focusing on Tactics, Techniques, and Procedures (TTPs), we introduce a refined threat model designed to outmaneuver these challenges. The proposed approach offers precise, actionable strategies to combat ISM threats, setting a new standard for resilience in a world of ever-advancing cyber adversaries.

Keywords

Malware (ISM), Malware-As-A-Service (MaaS), Password Manager Vulnerabilities, MITRE ATT&CK Framework, Ttp-Based Threat Modeling, Adaptive Cybersecurity, Real-Time Threat Defenses, Phishing-Resistant Mechanisms, Usability-Security Trade-Offs, Malware Evasion Techniques, Cryptographic Weaknesses, Next-Generation Malware Defenses

1. Introduction

Authentication confirms the legitimacy of a user and traditionally involves something you know (password), something you have (smartphone, security token), or something you are (fingerprint, facial recognition) [1]. Passwords have long been the primary method, even though there is a shift towards using multiple factors

for better security. Recent data from 2024 shows that 3 in 10 users have experienced a data breach due to poor password practices [2]. A Georgia Tech study found that 75% of popular websites do not meet password security standards, allowing users to create weak passwords; and 12% do not specify a minimum password length [3]. These weak protocols highlight the risks of password-based authentication and the need for stronger password policies and password management tools.

The psychology behind the use of password managers reveals a mixture of user behaviors, emotions, and decision-making processes. The average user manages 25 online accounts, and the cognitive burden of managing various passwords can be eased with password managers [4]. These applications generate and store passwords securely and simplify the login process across platforms. Password management tool is only as secure as the password, Master Password, you set to protect it [3]. Although password managers are not foolproof, they significantly mitigate risks associated with weak credentials and password reuse. The growing reliance on password manager is evident in the market's expansion from $2.5 billion in 2023 to an estimated $3.06 billion in 2024, a 22.4% yearly increase. It is projected to reach $6.8 billion by 2028, continuing at a 22.1% annual rate [5]. This growth attracts cybercriminals, targeting vaults and master keys to access the valuable data inside.

Attackers stole LastPass's company password vault by exploiting a vulnerability in a third-party media software on an engineer's home computer. They installed a keylogger to capture the master password, accessing sensitive data [6]. This shows how any vulnerability can be leveraged to infiltrate password managers, exploiting both user and system weaknesses. Similarly, Norton LifeLock faced a credential stuffing attack where attackers used compromised credentials to access customer accounts [7]. This tactic exploits password reuse, bypassing the need to leak new passwords. In 2023, Bitwarden users encountered malicious Google Ads redirecting them to phishing sites [8]. These ads, appearing at the top of the search results, led users to convincing phishing sites that mimicked the Bitwarden login page [9]. This exploits user trust in search engine results and interactions with trusted platforms. These incidents show how attackers exploit trust in known platforms, predictable behavior, and single points of failure to access critical data. Information thieves target all types of credentials, and password managers are a common gateway.

Passwords remain essential despite the move towards passwordless authentication [10]. Bill Gates predicted a passwordless future in 2004 [11], yet we are still defending against password-based authentication attacks today. A passwordless world is appealing, but it's not our reality after two decades. Major data breaches often involve phishing or brute-forcing passwords [12].

The remainder of this paper is organized into four main sections. Section 2 explores security flaws in open-source password managers and how ISMs exploit them. Section 3 discusses the Malware-as-a-Service (MaaS) platform and its im-

pact on organizations. Section 4 introduces the proposed TTP-based threat model, supported by a case study on Raccoon 2.0, and provides a list of recommended actions users can take to protect password managers against ISMs. Finally, Section 6 offers concluding remarks.

2. Password Managers

This section examines the architectural vulnerabilities and security trade-offs of leading third-party password managers targeted by ISMs. We focus primarily on two open-source password managers, Bitwarden and KeePass, while briefly touching on the architecture of others towards the end. These password managers, due to their open-source nature, have the most amount of public documentation available which makes it ideal for our analysis—highlighting weaknesses in design, storage, and data transmission. A common theme amongst password managers that makes it easy for ISMs to steal data is that they tend to have a wide cross-platform footprint. A compromise in any one of the access points (such as web vaults, browser extensions, desktop applications, or mobile applications) results in a complete loss for users.

2.1. Bitwarden

Bitwarden an open-source cross-platform password manager with a web vault, browser extensions (Chrome, Firefox, Edge, Safari, Opera, Brave), desktop apps (Windows, macOS, Linux), and mobile apps (iOS, Android). It also offers a command-line interface (CLI), a self-hosted option, and Bitwarden Send, a secure method to share information [13].

2.1.1. Architecture

Sensitive data, including passwords, is encrypted client-side before transmission over HTTPS. Only encrypted data is stored, ensuring a zero-knowledge architecture. Bitwarden login process involves several key steps to ensure the security and privacy of users' data. A high-level overview of the Bitwarden architecture is shown in Figure 1.

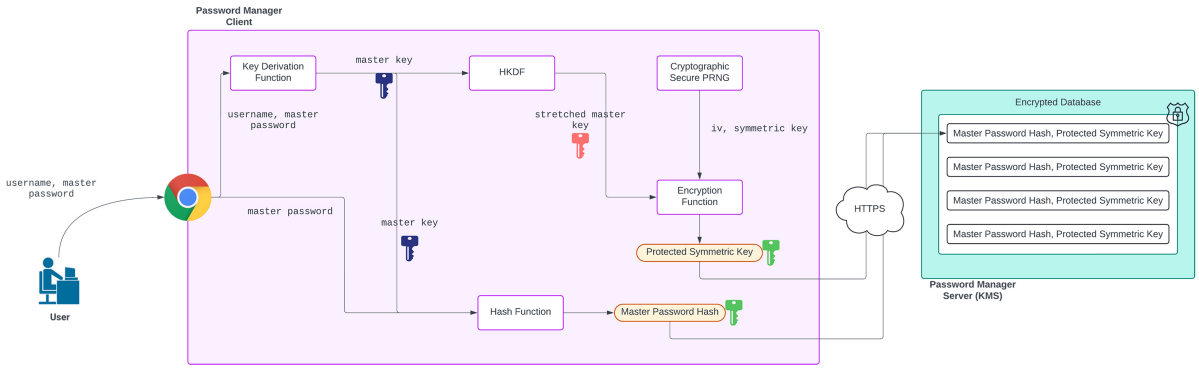

Figure 1. Bitwarden® Architecture [14].

1) Creation of a new master account:

a) To create a Bitwarden account, provide the user provides their email and a master password. The password, salted with your email, undergoes key derivation via PBKDF2 (600,000 iterations) or Argon2id, producing a 256-bit master key.

b) A hash of the master key is sent to the Bitwarden server for account creation and authentication. The 256-bit master key is stretched to 512 bits using HKDF. Simultaneously, a CSPRNG generates a 256-bit symmetric key and an initialization vector (IV).

c) The symmetric key is encrypted with the stretched master key and the IV, forming a Protected Symmetric Key, which is securely stored. Items added to your vault are encrypted with the symmetric key using AES-256 in GCM mode, ensuring only encrypted data is sent to Bitwarden servers.

2) Adding a new password:

New passwords are encrypted with AES-256 using a symmetric key and a random IV before storage. The encrypted data is sent to the Bitwarden server, synced across devices, and decrypted locally, ensuring the server never accesses plaintext passwords [14].

2.1.2. Known Attack Vectors

Breaking into the encrypted vault is computationally infeasible, largely due to the high number of iterations used in the encryption process. However, if attackers manage to obtain password hashes, they may attempt to crack them offline. This is why Bitwarden uses hashing algorithms with a high iteration count, making such attacks extremely difficult and time-consuming. Even so, attackers might target the master password key through phishing or keyloggers, or attempt to exploit other aspects of the application if direct access proves too challenging.

1) Adversary-in-the-Middle Attack (AitM) with Phishing

Adversary-in-the-Middle (AitM) phishing attacks increase the risk of credential theft by intercepting login credentials, multi-factor authentication (MFA) codes, and session cookies. This allows attackers to bypass MFA and access accounts, presenting severe authentication risks.

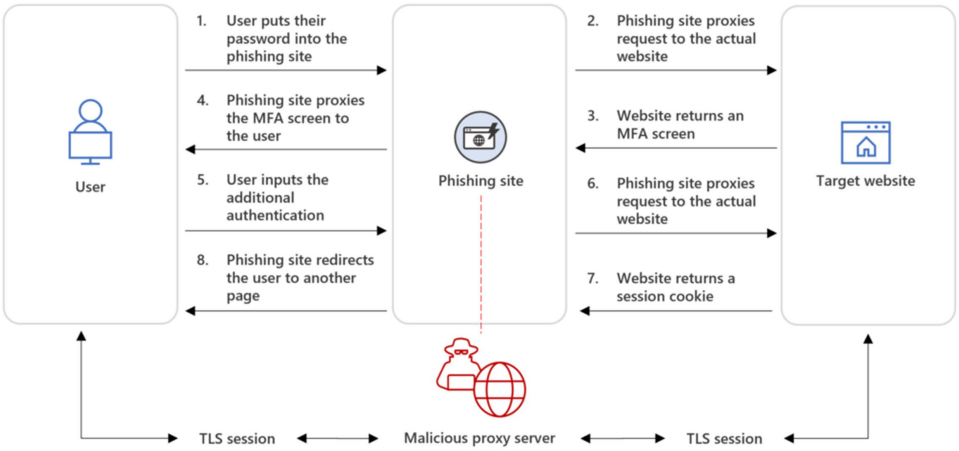

Figure 2. The diagram demonstrates how an attacker intercepts and manipulates the communication between a user and a legitimate service [16].

Toolkits like Evilginx [15] facilitate AitM attacks against softwares employing with traditional MFA or weak authentication. These attacks establish an HTTPS connection between the user, attacker, and target site. Despite HTTPS, attackers intercept credentials, MFA codes, and session cookies. Even out-of-band methods like push notifications are vulnerable; once approved, the session cookie is sent to the attacker, who then retains access to the target site while severing the user's connection [16]. **Figure 2** illustrates the AitM attack flow.

2) Malvertising & Typosquatting

Malvertising uses ad vulnerabilities to install malware, while typosquatting redirects users via typographical errors. Malvertising can infect correctly spelled domains [8] and extends beyond a single platform or service, affecting a wide range of websites like Google, Bing, and DuckDuckGo.

The attack flow is complex yet sophisticated. In September 2023, Proofpoint identified a new malware called ZenRAT, which was being distributed via fake installation packages of the password manager Bitwarden. The malware specifically targets Windows users, while redirecting individuals on other operating systems to a benign webpage [17].

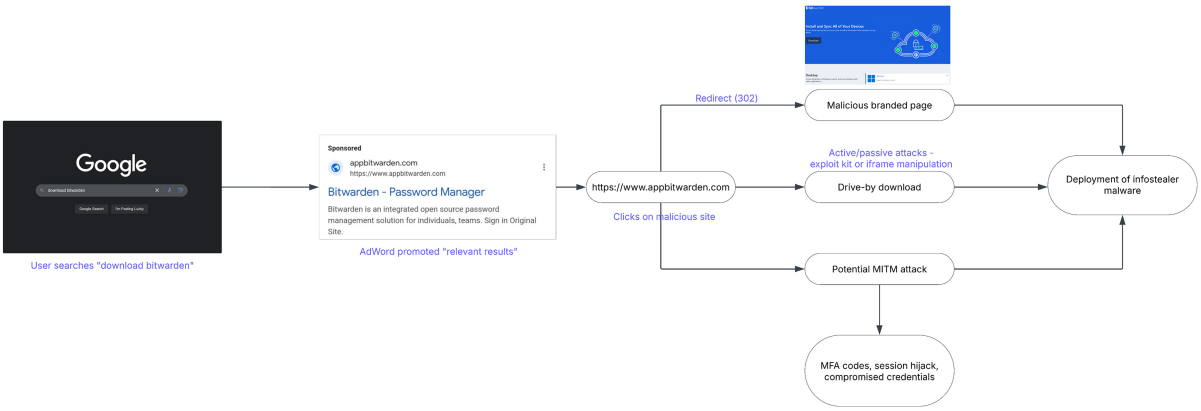

Figure 3. The diagram illustrates the sequential stages from initial phishing to credential theft, highlighting key attack vectors.

As shown in **Figure 3**, the initial stages of the attack remain consistent across various incidents. However, the distribution of the malicious payload—facilitated by malware-as-a-service platforms can vary significantly based on the tactics used by the attackers.

3) Memory Retention Vulnerability

Bitwarden's master password remains in memory while the vault is unlocked, posing security risks. Attackers can access the password during active sessions, compromising data security. Clearing the password upon vault locking or logout may leave residuals due to memory management and runtime environment limitations [18].

Bitwarden has implemented several fixes. In July 2023, a partial fix reloaded application processes to purge memory addresses. Recently, crashing the renderer

process in Electron applications was added to ensure the master password is not retained. The Firefox addon for Linux does not retain the master password in memory, either in unlocked or locked states. The Bitwarden Chrome extension with respect to memory retention issue was fixed by crashing the renderer process to clear memory [18].

4) Clickjacking and Auto-fill Attacks

In March 2023, Flashpoint published research on iframe vulnerabilities in password managers like Bitwarden. They found that insecure auto-fill and default URI matching can lead to credential theft [19]. An attacker can embed a non-sandboxed external iframe on an uncompromised site to steal credentials if "Auto-fill on page load" is enabled. Additionally, attackers can use subdomains sharing the base domain with a provider login form. This iframe issue has been known for over a decade [20].

Flashpoint notes some password managers detect cross-origin iframes, but the evaluated number is unspecified. No comprehensive vulnerability analysis exists. Despite iframe checks, managers remain susceptible to common exploits. Iframe checks are a temporary fix for a pervasive vulnerability in most password managers [21].

It is very unlikely that website is wholly self-contained. Web security and trust issues exist between publishers and third-party content providers. The Same Origin Policy does not fully address this. Publishers must either sandbox third-party content, limit functionality, or risk security by direct embedding. Browser vendors must choose to protect users or hold publishers responsible [22] [23].

5) Self-Hosted Environment Risks: Aggravating User Paranoia

When users encounter suspicious links online or similar-looking page, Bitwarden community often collaborates to address potential threats. Users report these links to Bitwarden support, which notifies search engines. However, about half of the time, these links are self-hosted Bitwarden versions mistakenly flagged as malicious due to user confusion over [24] Bitwarden self-hosting options, which allow hosting of the entire Bitwarden service [25].

The Bitwarden documentation advises against using their name for self-hosted versions to avoid confusion. Unfortunately, this guide is overlooked, leading to difficulties in distinguishing legitimate Bitwarden instances from malicious ones. A case in point is when users searching for "Bitwarden manager" encountered a domain, bitwarden [dot] pw, which initially raised alarms but was later claimed to be a legitimate self-hosted version, even registered through Namecheap, the same registrar Bitwarden uses [26]. Based on our observations from community forums and user feedback, it is evident that this ongoing tension can lead to user unease, as they struggle with the constant concern over the security of their sensitive information [24].

6) Cryptographic Failure Risks

In 1978, Robert Morris and Ken Thompson improved password security by introducing cryptographically secure one-way hash functions. This made stolen

password files much harder to crack, especially if the passwords were strong. However, as computational power doubles roughly every two years, as predicted by Moore's Law, cryptographic methods must constantly evolve to keep up with growing attack capabilities [27].

After the 2023 LastPass hack [6] [28], it was found that Bitwarden had 100,000 client-side iterations and 100,000 server-side iterations by default.If these were compromised, It could facilitate offline attacks by reducing the time needed to crack passwords. This issue was BWN-01-009 in Bitwarden's 2018 Security Assessment and remained unresolved until 2023 [29] [30]. Bitwarden has always supported iteration customization and offers Argon2 [31]. Bitwarden increased the PBKDF2 iterations following OWASP's announcement of new minimum iteration standards in 2023. However, it remains unclear whether this update applies universally across all platforms for both existing and new users. Noted and community platforms [32] and Bitwarden white paper [33], Adjusting PBKDF2 iterations necessitates re-encrypting vault data, which carries the risk of vault corruption.

2.2. KeePass

KeePass is a desktop password manager primarily for Windows, with cross-platform versions like KeePassXC for Windows, macOS, and Linux. Mobile apps like KeePassDroid (Android), Strongbox, and KeePassium (iOS) support syncing and managing KeePass databases. Browser integration is achievable through plugins and extensions such as Kee for Firefox and Chrome. KeePass also offers a portable USB version and various plugins to extend its functionality [34].

It regularly updates to counter threats, to ensure reliable password management. The core component is its vault database, accessible through the desktop or command-line interface, with a workflow as depicted in **Figure 4**. The KeePass plug-in framework allows users to customize functionality, such as file format import/export or autofill features [35]. A significant aspect of KeePass is its use of the KDBX vault database format, which is well documented and open source. KeePass encrypts the entire database, including passwords, usernames, and notes, using AES-256 encryption. The security of KeePass is heavily based on the strength of the cryptographic formats used and the strength of the passwords chosen by users [34].

2.2.1. Architecture

KeePass database requires a master password, a key file, or both. KeePass 2.x adds a third option: authentication via the current Windows user. KeePass encrypts the database using AES, Twofish, or ChaCha20 ciphers. AES is the default in both editions. Twofish is specific to KeePass 1.x, while ChaCha20 is in KeePass 2.35 and above. A plugin provides Twofish encryption for KeePass 2.x [34].

In KeePass 1.x (KDB), data integrity is verified via a SHA-256 hash of the plaintext, whereas in KeePass 2.x (KDBX), data authenticity is ensured using an HMAC-SHA-256 hash of the ciphertext (Encrypt-then-MAC). Passwords are protected in

memory while running. On Windows Vista and later, passwords are encrypted using Windows Data Protection API, with keys stored securely in non-swappable memory. In earlier Windows versions, KeePass uses the ARC4 cipher with a temporary random session key [35]. KeePass also supports OTP as an additional layer of security which can be used along with the master key for multi-factor authentication.

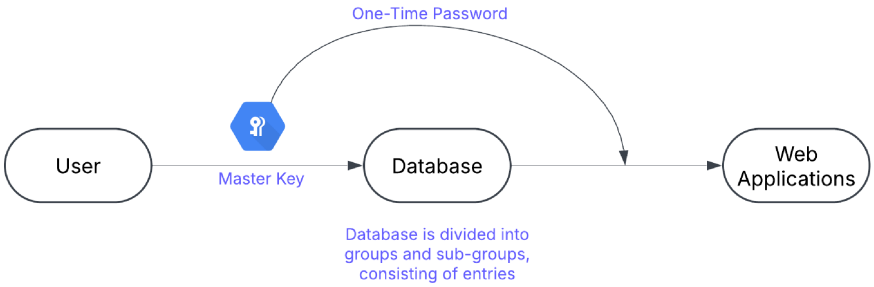

Figure 4. Simplified KeePass WorkFlow [36].

2.2.2. Known Attack Vectors

The plugin framework is user-friendly, but poses a security risk—malicious plugins. An attacker could insert a malicious plugin into the KeePass "Plugins" folder. To mitigate this, KeePasss suggests to implement a file system ACL for the KeePass directory, including the "Plugins" folder, and require administrator privileges for write access s [37].

1) KeePass-password-dumper

Another vector of attack involves KeePass's password input system. KeePass 2.X uses a custom text box, called SecureTextBoxEx, to enter passwords. This text box is used not only for the master password but also for other passwords within KeePass. The vulnerability occurs because, when you type in the text box, intermediate strings are created in memory for every character typed. This is a problem because, due to the way .NET manages memory, these strings can't easily be removed. For instance, if you type Password, leftover strings like *a, **s, ***s, remain in memory, allowing attackers to search these patterns and potentially reveal the password.

The effectiveness of this attack depends on the typing patterns and the number of passwords entered in a session. Even with multiple passwords or typos, .NET's memory allocation process can make these leftover strings appear in order, allowing an attacker to recover multiple passwords. Fortunately, the KeePass developers took this into account and implemented fixes in June 2023 [38].

2) KeePass Export Trigger Exploit

A fix for the CVE-2023-24055 vulnerability in KeePass addressed a specific issue where attackers with write access to the KeePass configuration file could exploit it to extract passwords in plain text by adding an export trigger. The vulnerability was mitigated by a change implemented in KeePass version 2.53.1, which removed the "Export-No Key Repeat" application policy flag, a configuration set-

ting that previously allowed password exports without re-prompting for the master key under specific circumstances [39].

3) Typosquatting with Punycode in Malvertisements

This attack vector affects any software, with KeePass as a notable example. It illustrates a sophisticated evolution in phishing techniques, particularly the exploitation of Punycode—a method used to create visually similar but malicious URLs that can deceive users into thinking they are accessing legitimate sites. This technique leverages the encoding of non-ASCII characters to craft deceptive web addresses, making it a powerful way of phishing attack across various platforms.

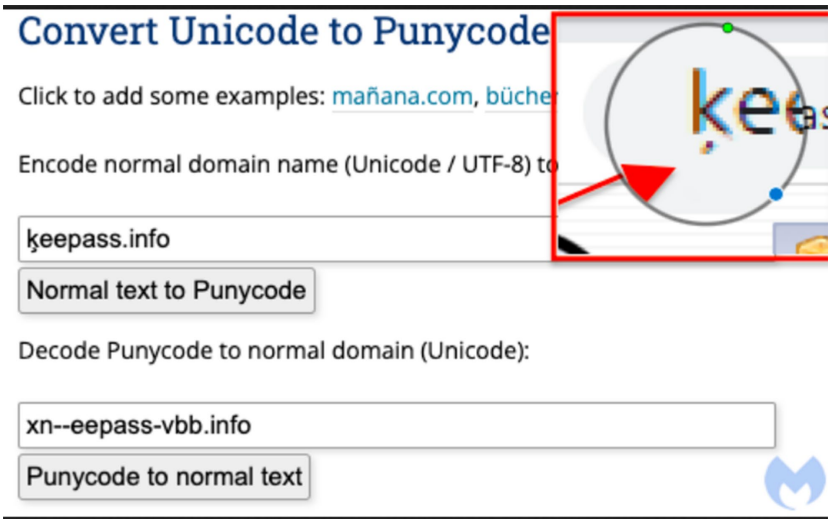

Figure 5. Demonstration of punycode conversion [40].

Figure 5 shows how a legitimate domain like "keepass.info" can be encoded into Punycode, resulting in a visually similar but malicious domain name [40].

4) Automatic Update Vulnerability

KeePass's earlier use of HTTP for retrieving version information files posed a potential man-in-the-middle attack risk [41], though it did not permit automatic updates or unauthorized changes. KeePass has since transitioned to using HTTPS and signed version files, which enhances the accuracy and security of update notifications. However, attackers have continued to exploit this previous shortcoming by using it to deceive users into thinking they need to install a fake update, thereby facilitating phishing attacks [34].

2.3. Other Password Managers

This section explores popular vendor-provided password managers and proprietary third-party solutions, emphasizing their features and the generalized attack vectors they face in the context of ISMs. While our analysis is not as comprehensive as it is for Bitwarden and KeePass due to their proprietary nature, we nevertheless believe that they share many of the same attack vectors that affect the for-

mer.

2.3.1. Vendor-Provided Password Managers

1) **Apple iCloud Keychain**

Apple iCloud Keychain is integrated into the Apple ecosystem, supporting macOS and iOS platforms. It features end-to-end encryption, biometric authentication through Face ID or Touch ID, and cloud-based device synchronization [42]. Its reliance on the Apple ID for security, however, presents potential vulnerabilities if the Apple ID is compromised or exploited via phishing attacks [43].

2) **Google Password Manager**

Embedded within Google accounts and accessible through Chrome and Android, Google Password Manager offers password auditing, alerts for reused or compromised credentials, and synchronization across devices. While convenient, its browser-based implementation introduces risks, particularly when adversaries exploit session cookies or compromise Google accounts [44].

3) **Microsoft Wallet**

Microsoft Wallet integrates with Windows and Microsoft Edge, and lets users take advantage of biometric authentication and account synchronization when permitted by GPO policies. While offering a user-friendly interface, its security depends heavily on the robustness of the Microsoft account, which, if breached, may allow adversaries to access stored credentials [45].

2.3.2. Proprietary Third-Party Password Managers

1) **1Password**

1Password is known for its robust security architecture, 1Password offers unique features such as Travel Mode [46], which removes sensitive data from devices during travel, and cross-platform support. However, as a proprietary tool, its closed-source nature may limit transparency, raising concerns for some users [47].

2) **Dashlane**

Dashlane offers many features over the conventional password manager, including Dark Web monitoring and password health assessments. Despite its strengths, browser extensions and syncing mechanisms may become targets for attackers seeking to compromise user data [48] [49].

3) **LastPass**

LastPass, a widely used password manager, has faced significant scrutiny following recent security breaches. These incidents highlight the vulnerabilities inherent in proprietary infrastructure, including server-side breaches and the exposure of encrypted vaults [6].

2.3.3. Generalized Attack Vectors

1) **Phishing and Social Engineering Attacks**

Adversaries exploit trust in password managers by creating convincing phish-

ing campaigns to steal master passwords or login credentials. Social engineering attacks often involve spoofed recovery emails or fraudulent websites mimicking legitimate services [50].

2) **Master Password and Account Recovery Exploits**

Attackers target weak master passwords and exploit poorly secured account recovery processes. These vulnerabilities undermine the primary layer of security for both vendor-provided and proprietary managers. An example of this is the LastPass breach mentioned in Section 1, where attackers stole encrypted password vaults. The security of these vaults depended on the strength of users' master passwords, with weak ones being vulnerable to brute-force attacks.

3) **Insider Threats and Infrastructure Breaches**

Proprietary managers relying on centralized server infrastructure are vulnerable to insider threats and breaches, as seen in the LastPass attacks where encrypted vaults were exfiltrated [6].

4) **Platform-Specific Bugs**

Vendor-provided password managers may inherit platform-specific vulnerabilities, such as insecure clipboard handling [51] on Windows or memory retention issues on macOS and iOS [52]. These bugs offer adversaries additional vectors for exploitation.

3. Malware-As-A-Service

This section explores ISM trojans, focusing on their ecosystem and impact on password manager security. It also details the cycle and dynamics of ISM trojans, including distribution and infection methods. Next, it analyzes the threats these trojans pose to password managers and their adversarial capabilities, crucial for developing a robust threat model to strengthen password manager defenses.

Malware as a Service (MaaS) is not new, but its scale and sophistication have surged, making it a significant concern in 2024 [53]. MaaS operates on a business model like Software as a Service (SaaS), offering pre-packaged malware on the dark web for purchase or rent by cybercriminals. Common MaaS tools from July 2023 to July 2024 included malware loaders (77%), Cryptominers (52%), Botnets (39%), Information-stealing malware (36%) and Proxy botnets (15%) [54].

New ISMs as Malware-as-a-Service (MaaS) enter the cybercrime market frequently and are popular among threat actors. Subscribers, threat actors, get a prebuilt stealer, its admin panel, and direct support from MaaS providers.

3.1. Ecosystem of ISM as a Service

This model, as shown in **Figure 6** supports two main roles: vendors and buyers. Vendors develop, market, and sell the ISM information stealer and its administration panel. Buyers, also cybercriminals, subscribe for temporary access and handle its distribution.

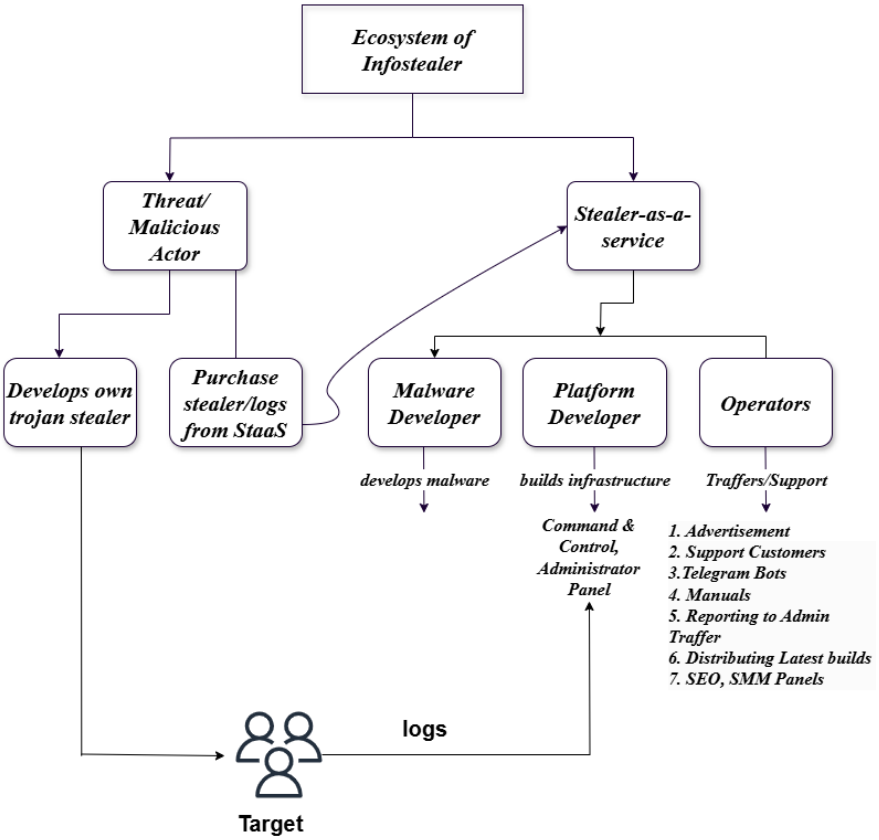

Figure 6. Workflow of an information stealer malware (ISM) ecosystem [55].

3.2. Distribution Channels

YouTube Channels: An attacker uploads a YouTube video showing how to install cracked software, including a link in the description to a distribution site. This link often leads to services like MediaFire, transfer.sh, or Google Drive. To bypass antivirus software, the file is a password-protected archive with the password in the description. When the victim extracts the file, the disguised ISM is executed. The video may be posted from either a fake account created specifically for the campaign or a compromised account obtained from previous ISM campaigns or purchased from cybercrime marketplaces. This method is favored for its simplicity and low technical demands [56].

SEO Poisoning: is another prevalent technique in which attackers manipulate search engine algorithms to promote their malicious websites. These sites typically host fake or cracked software. Unsuspecting users click on the top search results and download password-protected archives, initiating the same infection process as the 911 chain. This method requires advanced SEO skills and the resources to host malicious websites [57].

Malvertising: Malvertising involves using Google ads [9] to spread fake versions of legitimate websites, such as video conferencing software or VPN clients, to deliver malware. Innovation in infection chains has created new roles in the cybercrime ecosystem. Telegram channels and cybercrime forums show that crim-

inals sell copies of legitimate software websites and search engine optimization guides to rank these malicious sites higher [58]. This trend indicates the growing professionalism and expansion of the cybercrime industry.

Figure 7. Redline stealer logs distributed via telegram.

3.3. Processing and Sale of Stolen Data

The data exploitation process starts with collecting valuable information, such as log-in credentials, email addresses, and cryptocurrency details, from dark web sources such as command and control (C2) systems, Telegram channels, and Discord servers. This collection forms the raw material for exploitation. The collected data are then aggregated into organized sets, merging disparate pieces into a unified format for easier processing. Advanced tools and scripts automate this, handling vast data efficiently. The data is then sorted and categorized by type, such as personal credentials, financial information, or specific email addresses. Sorting is crucial for targeting the data's use or sale, meeting buyers' needs in the underground market. Sophisticated operations can tag or annotate data to add contextual value, increasing its appeal. Data cleaning processes meticulously remove duplicates, ensuring that each entry is unique and improving data quality and value [59].

Once sorted, refined data is distributed through dark web forums, encrypted

Telegram channels, and various marketplaces such as XSS [60], BHF [61], Exploit.in, Lolz.guru [62], Cracked. These platforms allow cybercriminals to buy, sell, or trade data, each with their own rules and security measures [63]. **Figure 7** shows a Telegram screenshot of a channel advertising stolen logs collected using the Redline malware., and **Figure 8** shows the lifecycle of logs in information stealer operations.

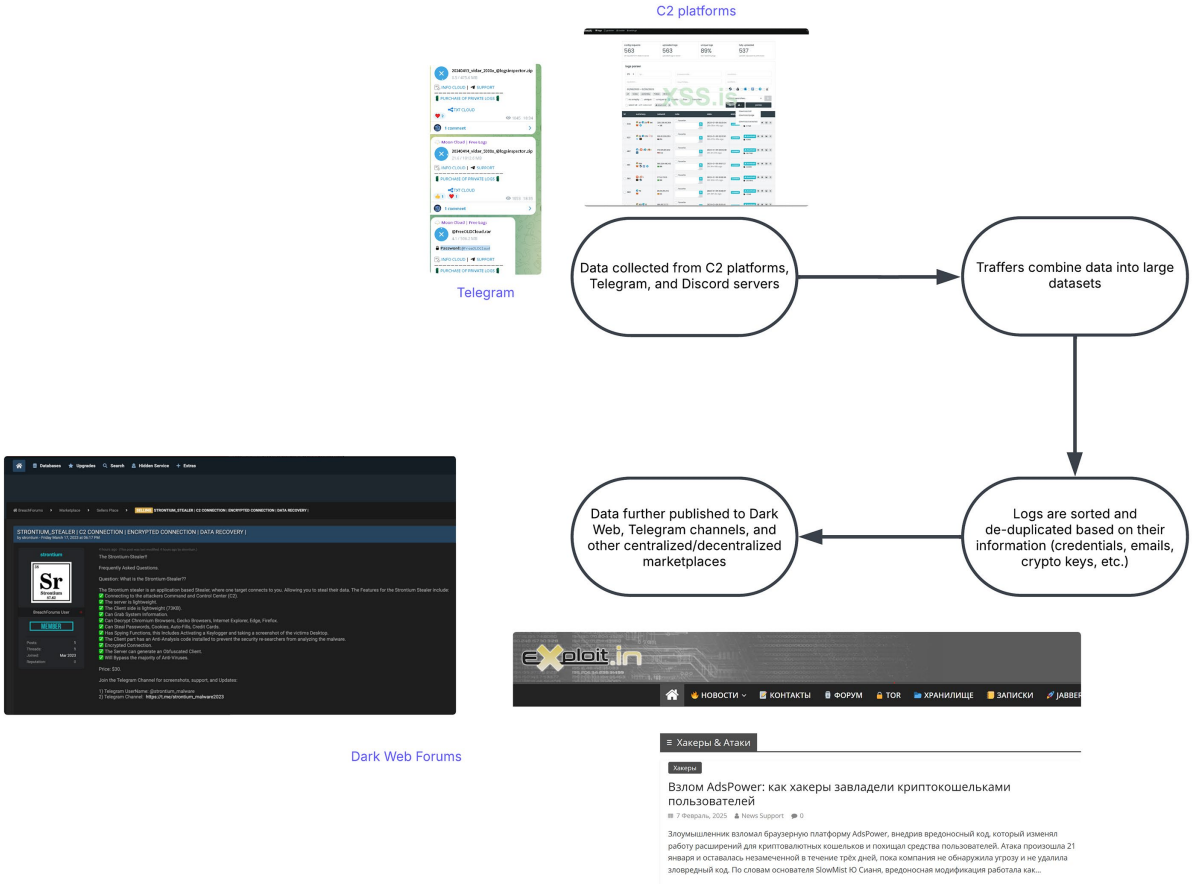

Figure 8. Lifecycle of Logs in Information Stealer Operations.

3.4. Impact of ISM on Password Managers

In 2023, ISM attacks increased by 30% [64], particularly targeting password managers such as Bitwarden® and KeePass. This surge shows cybercriminals' growing interest in these tools. ISMs can exfiltrate thousands of credentials in minutes. For example, a Redline Stealer campaign harvested more than 100,000 credentials in one month, proving their efficiency. Password managers are prime targets due to their valuable data. [54] reports that 60% of ISM campaigns in 2023 had modules to extract data from popular password managers. Table 1 shows the capabilities of various malware that steal information.

In 2023, ISM attacks increased by 30% [64], particularly targeting password managers such as Bitwarden® and KeePass. This surge shows cybercriminals' growing interest in these tools. ISMs can exfiltrate thousands of credentials in minutes.

For example, a Redline Stealer campaign harvested more than 100,000 credentials in one month, proving their efficiency. Password managers are prime targets due to their valuable data. [54] reports that 60% of ISM campaigns in 2023 had modules to extract data from popular password managers. **Table 1** shows the capabilities of various malware that steal information.

Table 1. Comparison of various Information Stealing Malware (ISM) based on their credential theft capabilities [65].

Malware / Capabilities	Browserdata	Crypto Wallets	Password Manager	Email	Steam	Discord	Telegram	Host information	FTP	Others	Screenshots
Vidar	✓	✓	✗	✓	✗	✗	✓	✓	✓	✓	✓
Redline	✓	✓	✗	✗	✓	✓	✓	✓	✓	✓	✓
Arkei	✓	✓	✓	✗	✗	✗	✗	✓	✗	✗	✓
Luca	✓	✓	✓	✗	✓	✓	✓	✓	✗	✗	✓
Raccoon	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✗

4. TTP-Based ISM Threat Model

Defending against ISM is challenging, requires adaptive strategies. ISM is a primary threat to password managers. This analysis identifies and examines the tactics, techniques, and procedures (TTPs) used to compromise them. Although focused on TTPs, the application of the STRIDE framework can extend this model to cover a broader range of threats.

4.1. The Modeling Paradox

According to OWASP guidelines [66], "A threat model is a structured representation of information that affects the security of an application". It views the application and its environment through the lens of security [66]. We need to define the "lens of security". We explore if the lens of security focus on a high-level overview, or does this approach risk missing critical details when complex threats are automated through playbooks. Furthermore, we discuss if it is appropriate to treat spoofing, data tampering, and privilege escalation as standalone threat events, or should they be integrated into broader, more comprehensive attack scenarios within the mode.

In this section, we propose a hybrid threat model that focuses on the tactics, techniques, and procedures (TTPs) used by advanced persistent threats (APTs) to spread information-stealing malware (ISMs) targeting password managers.

Existing Threat Models: The STRIDE framework [67] is a widely used tool for identifying and categorizing system threats. It includes both intentional (malicious) and unintentional (accidental) actions compromising security. The six STRIDE categories—Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege—offer a fundamental approach to identifying vulnerabilities. However, these categories provide only a high-level overview of broader and more complex behaviors.

While STRIDE ensures a broad consideration of threats, it raises a crucial question: Should defenders consider "everything", or focus on an accurate understanding of the threat landscape? Given the rise in ransomware attacks [68], it's impractical to debate if "Adversary-in-the-Middle" falls under tampering, spoofing, or information disclosure. Effective security management requires a threat modeling approach aligned with the organization's development cycles. If it takes 2 months to develop and prioritize risks, the model may be outdated by implementation, especially in fast-paced agile environments.

This highlights a critical limitation of STRIDE: It can promote a checklist mentality that may not be practical in urgent scenarios where precision and speed are essential [69]. Instead of trying to account for every possible threat, the goal should be to have a precise understanding of the most relevant and impactful threats, enabling defenders to respond effectively in real-time rather than being overwhelmed by an exhaustive list of potential vulnerabilities, which are often marked as "ignored" to maintain SDLC requirements [70].

The proposed TTP-based model offers actionable intelligence for organizations and users to remain vigilant. TTP hunting began with MITRE ATT&CK [65]. In 2022, Trellix launched the Adaptive Defense Model, integrating MITRE ATT&CK and the Unified Kill Chain, providing a comprehensive view of dynamic cyber threats [71]. The proposed TTP-based threat model targets ISMs for Password Managers but can be adapted for other threats. Hence, it balances the need for a high-level overview while focusing on critical details in complex threat scenarios.

4.2. Implementation of TTP-Based ISM Threat Model

We enumerate key assets as shown in **Figure 9** within password managers by mapping them to STRIDE threat categories, providing a high-level risk overview. Detailing every threat and mitigation strategy exceed this paper's scope; thus, threats will be attributed to relevant Tactics, Techniques, and Procedures (TTPs) of ISM. The following table outlines critical assets of open source password managers, focusing on Bitwarden and KeePass, although similar assets and threats exist in proprietary managers.

Figure 9 illustrates how a threat can have varied characteristics. It demonstrates the importance of categorizing each edge case within an overview to ensure that no threat is overlooked. By implementing a TTP-based threat model, user can effectively map each scenario to the relevant TTPs and develop mitigation strategies. This approach highlights the need for flexibility, as a one-size-fits-all solution is not sufficient; the model must be adaptable to address the unique aspects of each threat.

When securing sensitive data in password managers, it is important to set **trust boundaries**. These boundaries are distinct zones within the system, each with varying levels of security. Each boundary manages data flow according to its level of trustworthiness. The User Boundary includes the user's device and browser, where personal information is entered and stored. The Application Boundary encompasses the application that processes and encrypts passwords. The System

Boundary covers the OS and hardware resources that interact with the application. The External Boundary pertains to interactions with cloud services or third-party tools. The following figure demonstrates a TTP-based threat model for ISMs.

Assets	STRIDE Threat	Importance	Specific Artifacts
Master Password	S,T,I,D,E	Key to the vault : unlocks all stored credentials and sensitive data within the password manager	Bitwarden User Vault, KeePass Database (KDBX) File (C:\Users\Username\Documents\KeePass\)
Encrypted Password Database	T,I,D,E	Stores all user credentials and sensitive data in an encrypted format.	Bitwarden JSON Export Files, KDBX File (KeePass Database) (e.g.C:\Users\Username\KeePass\Database.kdbx)
User Authentication Data	S,T,I,E	Safety measures - 2FA, PassKeys.	Bitwarden Account Details (email, 2FA), KeePass Key File(C:\Users\Username\KeePass\KeyFiles\KeyFile.key))
Browser Integration Components	T,I,S	Responsible for autofill and other interactions between the password manager and web browsers.	Bitwarden Browser Extension Data, KeePassXC-Browser Extension Data
Backup Files	I,T,E	Stores encrypted copies of the password database for Emergency.	Bitwarden allows exporting of vault data as JSON or CSV files. These are stored in local directories specified by the user (~/Documents/Bitwarden_Backups/).
Session Tokens	S,I,E	Authenticates ongoing user sessions.	Stored in browser cookies or local storage, particularly for syncing across devices. (%APPDATA%\Bitwarden)

Figure 9. Asset Identification and STRIDE Categorization for Password Managers targeted by ISMs.

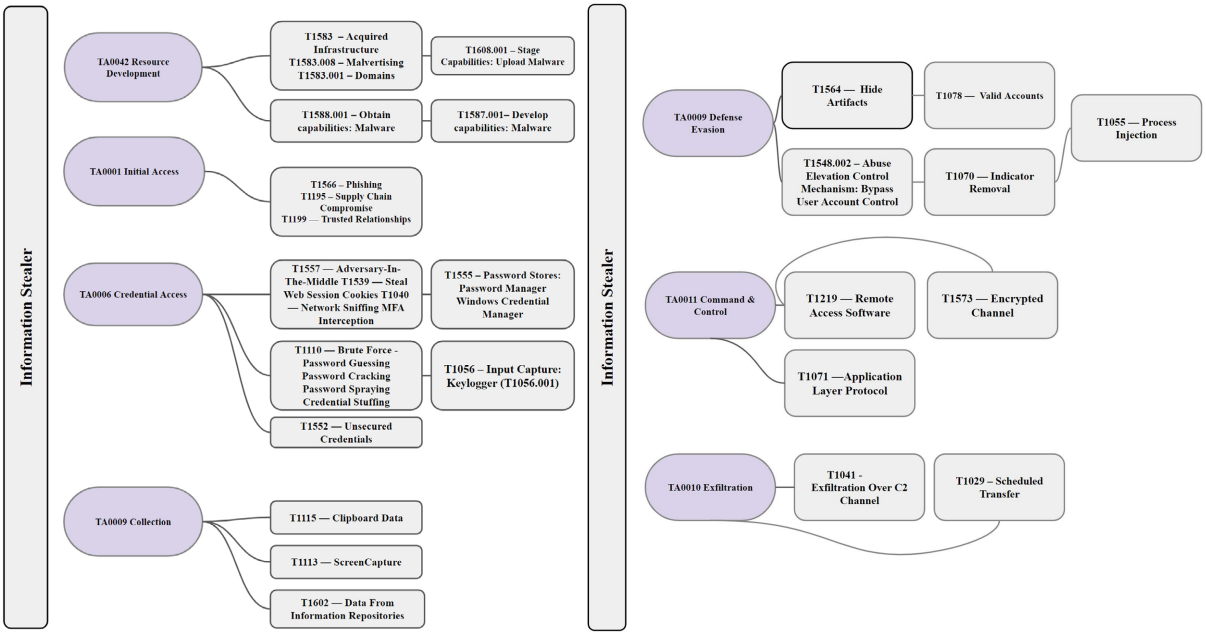

Figure 10. Proposed TTP-based threat model for password managers targeted by ISMs.

There are common usability and security trade-offs that password managers face and have adequate research [3] [23] [72]-[74] on it; differences lie in the way

each tool handles these challenges.

Our threat model employs MITRE ATT&CK and STRIDE to assist both enterprises and individuals in combating information-stealing ISMs. While enterprises can deploy sophisticated defenses, individuals often face greater challenges, where striking the right balance between security and usability becomes critical. Based on our analysis, we propose the following recommendations to mitigate identified threats. Proper implementation of these measures can significantly reduce vulnerabilities, but poor execution—such as with passkey implementations—may introduce new weaknesses [75].

1) **Auto-fill** vulnerabilities in password managers arise from the trade-offs between web technologies and security. Attackers can exploit auto-fill features through phishing sites and deceptive iframes, tricking users into revealing credentials. Auto-fill improves usability but also introduces risks if mismanaged. To mitigate these risks, password managers have enhanced URL matching and verification algorithms, ensuring auto-fill only activates on trusted domains with valid SSL/TLS certificates. Conditional auto-fill, such as requiring user confirmation for sensitive logins and configurable security levels, balances convenience and security. For example, auto-fill is off by default in Bitwarden [19].
2) **Clipboard Data Leakage** depends largely on the operating system's clipboard handling. For example, Windows stores and syncs clipboard history, which can potentially expose sensitive data from password managers even after the clipboard is cleared [76]. KeePass addresses this by automatically clearing the clipboard after use [35]. To mitigate this risk, users can disable clipboard sync or manually clear their clipboard history. Alternatively, employing auto-fill can reduce the need for copying sensitive data, as no encrypted information is converted into plain text for a form. However, this introduces a trade-off, as auto-fill features come with their own set of vulnerabilities [77].
3) **Pin Security** can be confusing due to various rules. Bitwarden recommends enabling the "Lock with Master Password on restart" option. This locks the app, purges sensitive data on restart or inactivity, and securely manages the encryption key, reducing unauthorized access risks. These risks increase if users aren't familiar with PIN-based unlocking security settings [78].
4) **OWASP recommends** at least 600,000 PBKDF2 iterations with SHA-256 to maintain robust security, considering current and anticipated hardware advancements. As hardware capabilities improve, brute-force attacks become increasingly feasible, making it imperative to periodically review and update key derivation function (KDF) settings [79]. However, changes to KDF configurations can result in data loss if not carefully managed, emphasizing the importance of creating backups before rotating encryption keys or modifying iterations [80].

Password security encompasses several critical factors beyond iteration

counts. For example, while employing 1,000,000 iterations may appear secure, it can introduce performance trade-offs that affect user experience. Additionally, reused or compromised passwords remain vulnerable to credential stuffing and dictionary attacks, regardless of the iteration count. The use of passphrases with higher entropy is recommended to enhance security and mitigate these risks [74].

5) Emergency Kit is crucial for all users upon signing up for a password manager. Note, it doesn't protect against ISM and must be secured. This [81] [82] is useful for protection.
   - **Master Password**: The key to user's vault. Users must ensure that it's strong and secure [83].
   - **Email Address**: Important for account recovery and communication. Essential for identity verification or password reset.
   - **2FA Recovery Code**: Allows vault access if user's 2FA method is unavailable.

Where to Store It: Emergency kit must be stored in a physically secure place. However, it is not a complete backup. Here are some limitations:
- **Vault Backup:** The kit does not include a full vault backup, which is necessary for recovering from data loss or changes.
- **Recovery Codes:** The kit does not store recovery codes for services like Google, Etsy/personal accounts, VPNs, or phone providers. Manage these separately.
- **Authenticator App:** The kit does not contain TOTP secrets, which are needed if user's phone is lost or becomes inoperable.

6) **Support from OSINT Tools:** The growing open-source community offers great support. The Bitwarden Community Forum and KeePass on SourceForge are active and helpful, especially during unprecedented events. Using OSINT tools such as *dnstwist* [84], *urlscan.io* [85], and *VirusTotal* [86] can help prevent phishing attacks or defend against active attack through automation or manual URL analysis.

7) **Phishing-Resistant or Passwordless Authentication:** This approach is a promising idea, though it is slow to gain widespread adoption. Utilizing physical security keys, such as *YubiKeys* [87], or implementing protocols like *FIDO2* can enhance security, but they come with costs to consider.

4.3. Operationalizing a TTP-Based Threat Model: Raccoon v2

In this section, we present an operationalized TTP based threat model tailored to detect and respond to the activities of Raccoon v2 [88]. As Malware-as-a-Service (MaaS) is expanding, the Raccoon ISM shares malware updates and new features on underground forums [89]. The proposed threat model in **Figure 10** underscores the need to understand tactics and techniques of new information-stealing ISMs with different strategies. Many ISMs tend to have similar Tactics and varying Techniques. Raccoon was specifically chosen for this analysis due to its widespread

distribution and effectiveness in targeting password managers, making it a relevant and significant threat. Although there are other ISMs in circulation, expanding the scope of our analysis to include all of them would go beyond the focus of this paper. Future work could explore additional ISMs to provide a broader understanding of the evolving threat landscape, but for now, we focus on Raccoon v2 as a representative example of modern ISM tactics.

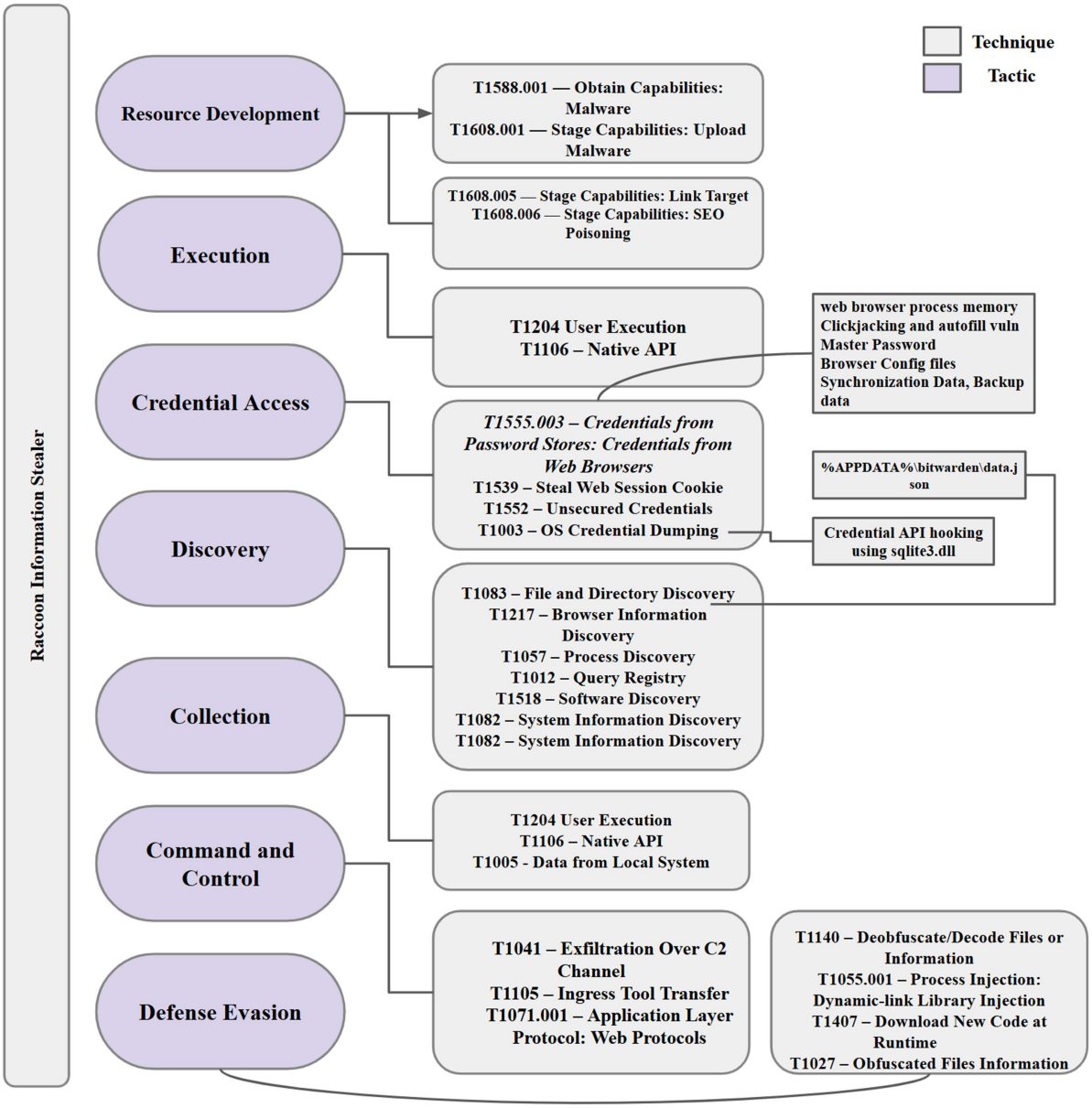

Figure 11. TTP-based threat model for Raccoon.

Figure 11 derived from our proposed model, illustrates potential attack scenarios organizations may face when targeted by Raccoon Stealer. These scenarios highlight the critical need for organizations to implement both automation and human oversight. While automation can efficiently handle routine detection and

response tasks, the evolving and adaptive nature of attack techniques requires human expertise to interpret anomalies, and respond to sophisticated threats. This balanced approach ensures comprehensive coverage against both known IoCs and emerging TTPs.

Defenders can create plugins or add rules to Endpoint Detection and Response (EDR) systems or Security Information and Event Management (SIEM) solutions to improve detection and response against advanced threats. These rules provide actionable defenses that can be directly incorporated into visibility tools for real-time monitoring and mitigation. Using the proposed playbook, defenders can streamline these efforts, mapping TTP-based detections and IoCs to operational workflows for consistent and effective defense across all phases of an attack.

4.3.1. Playbook for Raccoon Stealer Targeting Password Managers

IoCs and TTPs are key to every security team but serve distinct purposes. IoCs, such as file hashes or IPs, identify known threats but are short-lived due to adversary adaptation. TTPs, like T1555.003 (Credentials from Password Stores), provide consistent behavioral insights into how threats like Raccoon Stealer operate. This playbook prioritizes TTPs for proactive detection, supported by IoCs for immediate action.

Defenders can create playbooks based on our proposed TTP-based threat model, categorizing strategies into three key phases-Before the Attack, During the Attack, and After the Attack—to enable a structured and comprehensive defense against threats like Raccoon Stealer.

Before the Attack: Prevention and Preparation

Harden defenses to prevent adversaries from achieving **Initial Access (TA0001)** and preparing for **Credential Access (TA0006)**.

- **TTP Alignment:**
  - T1204 (User Execution): Raccoon Stealer often relies on phishing or malicious downloads to execute its payloads.
  - T1566.001 (Spear Phishing via Email): A common vector for delivering malicious links or attachments.
- **Preventive Measures:**

a) **Phishing Awareness Training:** Train users to recognize phishing attempts targeting password managers, ensuring suspicious emails or downloads are reported promptly.

b) **Application Control:** Enforce execution restrictions to block unauthorized binaries in directories like %TEMP% and %AppData%.

c) **IoC Integration:** Proactively block known Raccoon Stealer C2 servers, such as 136.244.65.99 [90] or 140.82.52.55, at the network perimeter.

d) **Credential Vault Hardening:** Restrict file permissions for sensitive files (e.g., vault.json or browser credential stores).

During the Attack: Detection and Containment Objective: Detect Raccoon Stealer activity in real-time, focusing on behaviors defined in **Credential Access (TA0006)** and related TTPs.

- **TTP Alignment:**
  - T1555.003 (Credentials from Password Stores): Raccoon Stealer hooks browser APIs and accesses files like Cookies to steal saved credentials.
  - T1071.001 (Application Layer Protocols): Data exfiltration occurs over HTTP/HTTPS to evade detection.
- **Detection and Containment:**
  a) **Vault Access Monitoring:** Ensure tools like EDR and SIEM are configured to monitor unauthorized access to password vaults or browser credential stores.
  b) **Malicious Execution Detection:** Leverage endpoint monitoring tools to detect execution of unauthorized binaries or known malicious hashes.
  c) **Outbound Traffic Analysis:** Monitor abnormal data exfiltration attempts to known C2 servers using network monitoring systems.
  d) **Automated Response:**
    - Isolate compromised endpoints via SOAR tools.
    - Block malicious IPs dynamically and reset credentials for affected accounts.

After the Attack: Investigation and Resilience Objective: Perform post-attack analysis to improve detection and recovery while enhancing future defenses.

- **TTP Alignment:**
  - T1005 (Data from Local System): Analyze logs to identify which data was accessed or exfiltrated.
  - TA0001 (Initial Access): Trace the origin of the breach, such as phishing emails or malicious downloads.
- **Post-Attack Actions:**
  a) **Forensic Analysis:** Review system and network logs to determine the attack's scope and collect evidence for identifying new IoCs.
  b) **Root Cause Analysis:** Identify the entry point and compromised credentials, then close identified gaps.
  c) **IoC and Rule Updates:** Update detection rules in SIEM or EDR systems based on newly identified IoCs and adversarial behaviors.
  d) **Reporting and Sharing:** Document incident findings and share them with threat intelligence platforms like MISP to improve industry-wide defenses.
  e) **Hardening and Simulation:** Use updated detection rules to simulate similar attack scenarios and validate improved readiness.

4.3.2. Operationalizing the Playbook with Splunk

By leveraging the TTP-based threat model and the provided IoCs, organizations can configure Splunk rules to detect and respond to malicious activities targeting password managers. While writing all possible rules is beyond the scope of this section, the following five rules serve as a foundation for defenders to adapt and expand upon based on their specific environments and threat intelligence feeds.

Splunk Rule 1: Monitor Unauthorized Access to Password Manager Vaults

This rule detect attempts to access sensitive files, such as password manager vaults

or browser-stored credentials, aligning with T1555.003 (Credentials from Password Stores)

```
Index = edr_logs
file_path IN ("C:\\Users\\*\\AppData
        \\Roaming\\bitwarden\\*",
                "C:\\Users\\*\\AppData
        \\Local\\Google\\Chrome
        \\User Data\\Default\\Cookies",
                "C:\\Users\\*\\AppData
        \\Roaming\\Mozilla\\Firefox
        \\Profiles\\*.sqlite")
| stats count by host, user, file_path
| where count > 5
```

Action: Alert the SOC when access patterns exceed a defined threshold, indicating potential credential dumping activity.

Splunk Rule 2: Detect Execution of Known Malicious Hashes This rule identifies the execution of binaries associated with Raccoon Stealer, leveraging IoCs like file hashes [90] to detect malicious payloads (T1204-User Execution).

```
index = edr_logs
sha256 IN ("0123b26df3c79bac0a3fda79072e36c15-
        9cfd1824ae3fd4b7f9dea9bda9c7909",
        "022432f770bf0e7c5260100fcde2ec7c4-
        9f68716751fd7d8b9e113bf06167e03",
        "048c0113233ddc1250c269c74c9c9b8e9-
        ad3e4dae3533ff0412d02b06bdf4059")
| stats count by host, process_name, sha256
```

Action: Automatically isolate the host using an integrated SOAR tool to prevent further execution.

Splunk Rule 3: Monitor Outbound Connections to Known C2 Servers This rule detects network traffic directed to known Raccoon Stealer command-and-control (C2) servers [90] (T1071.001-Application Layer Protocols).

```
index = network_logs
dest_ip IN ("136.244.65.99",
 → "138.197.179.146", "140.82.52.55",
                "142.132.180.233",
 → "142.132.229.12", "185.225.19.190")
| stats sum(bytes_out) as total_bytes by
,→ src_ip, dest_ip
| where total_bytes > 5000000
```

Action: Block the destination IP dynamically and trigger an alert for further investigation on EDR.

Splunk Rule 4: Identify Anomalous Vault Export Attempts This rule correlates login events followed by vault export requests to detect suspicious behavior

(T1555.003)

```
index = api_logs
(action = "vault_export" OR action = "login")
| transaction user maxspan = 10 m
| where action = "vault_export" AND
  → previous_action = "login"
| stats count by user, ip, timestamp
```

Action: Alert the SOC teams of unusual export attempts and lock the affected account.

Splunk Rule 5: Detect Abnormal File Execution from Temporary Directories This rule identifies Identify unauthorized binary execution from directories like %TEMP% and %AppData%, associated with T1204.

```
Index = edr_logs
file_path IN
  → ("C:\\Users\\*\\AppData\\Local\\Temp
\\*.exe",
  → "C:\\Users\\*\\AppData\\Roaming\\*.dll")
  | stats count by host, file_path,
    → process_name
```

Action: Alert the SOC and terminate the process and quarantine the binary for forensic analysis.

5. Limitations and Future Work

While the proposed TTP-based threat model provides a comprehensive framework for mitigating threats posed by Raccoon Stealer and other information-stealing malware (ISMs), it has not yet been empirically validated. The rules and playbook recommendations are derived from a theoretical analysis of Raccoon Stealer's TTPs, IoCs, and behavioral patterns based on publicly available threat intelligence [90].

This limitation highlights the need for further validation through empirical testing. For example:

- Deploying the proposed threat model in a controlled environment with simulated Raccoon Stealer attacks to evaluate its detection capabilities.
- Comparing the proposed model against existing solutions to assess its relative performance in reducing dwell time, improving detection rates, and mitigating data exfiltration.

6. Conclusions

Information-stealing malware, particularly in the context of malware-as-a-service (MaaS), poses a unique challenge that traditional threat models often fail to address. Existing models tend to be overly general, overlooking the advanced tactics ISMs use to exploit software vulnerabilities and avoid detection. Our approach bridges this gap by focusing on the tactics, techniques, and procedures (TTP) of

the top ISMs, as mapped using the MITRE ATT&CK framework. This targeted approach allows defenses that not only address traditional limitations but also adapt to evolving threats in real-time.

Our proposed threat model represents a significant step forward in addressing the increasing risks posed by Information Stealing Malware (ISM). TTP-based threat modeling provides actionable insights that align with the dynamic and evolving tactics of ISMs, empowering organizations and individuals to prevent and respond effectively to cyber threats. By transforming abstract security events into practical, real-world solutions, they enable timely interventions and foster robust defenses.

Through a detailed analysis of attack vectors, usability-security trade-offs, and software vulnerabilities, this model attempts to set a new standard for ISM defense. It shows the power of innovative security frameworks to translate cutting-edge research into meaningful, practical impact-fortifying password managers and protecting systems against the growing sophistication of cyberattacks.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] National Institute of Standards and Technology (NIST) (2024) Protecting Controlled Unclassified Information in Nonfederal Systems and Organizations. National Institute of Standards and Technology. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-171r2.pdf

[2] Exploding Topics (2024) 80 + Password Statistics for 2023: Data on Usage, Attacks & Attitudes. https://explodingtopics.com/blog/password-stats

[3] Roomi, S.A. and Li, F. (2023) A Large-Scale Measurement of Website Login Policies. In: 32*nd USENIX Security Symposium* (*USENIX Security* 23), USENIX Association, 2061-2078. https://www.usenix.org/conference/usenixsecurity23/presentation/al-roomi

[4] Fagan, M., Albayram, Y., Khan, M.M.H. and Buck, R. (2017) An Investigation into Users' Considerations towards Using Password Managers. *Human-Centric Computing and Information Sciences*, **7**, Article No. 12. https://doi.org/10.1186/s13673-017-0093-6

[5] Research and Markets (2024) Password Manager Global Market Report 2025. https://www.researchandmarkets.com/reports/5951840

[6] Goodin, D. (2023) LastPass Says Employee's Home Computer was Hacked and Corporate Vault Taken. https://arstechnica.com/information-technology/2023/02/lastpass-hackers-infected-employees-home-computer-and-stole-corporate-vault/

[7] Greig, J. (2023) Norton LifeLock Says 925,000 Accounts Targeted by Credential-Stuffing Attacks. https://therecord.media/norton-lifelock-says-925000-accounts-targeted-by-credential-stuffing-attacks

[8] Abrams, L. (2023) Bitwarden Password Vaults Targeted in Google ads Phishing Attack. https://www.bleepingcomputer.com/news/security/bitwarden-password-vaults-targeted-in-google-ads-phishing-attack/

[9] Ilascu, I. (2024) Fake Bitwarden Sites Push New Zenrat Password-Stealing Malware. https://www.bleepingcomputer.com/news/security/fake-bitwarden-sites-push-new-zenrat-password-stealing-malware/

[10] Okta, Inc. (2023) Passwordless Authentication for Customer Identity. https://www.okta.com/customer-identity/passwordless/

[11] Kotadia, M. (2004) Gates Predicts Death of the Password. https://www.cnet.com/news/privacy/gates-predicts-death-of-the-password/

[12] Information Is Beautiful (2024) World's Biggest Data Breaches & Hacks. https://informationisbeautiful.net/visualizations/worlds-biggest-data-breaches-hacks

[13] Bitwarden (2024) Password Manager Overview. https://bitwarden.com/help/password-manager-overview/

[14] Orenstein, G. (2023) Understanding Bitwarden Architecture. https://bitwarden.com/blog/understanding-bitwarden-architecture/

[15] Kgretzky/Evilginx2 (2024) Advanced Phishing with Two-Factor Authentication Bypass. https://github.com/kgretzky/evilginx2

[16] Microsoft Threat Intelligence (2023) Dev-1101 Enables High-Volume AiTM Campaigns with Open-Source Phishing Ki. https://www.microsoft.com/en-us/security/blog/2023/03/13/dev-1101-enables-high-volume-aitm-campaigns-with-open-source-phishing-kit/

[17] Proofpoint (2023) ZenRAT: Malware Brings More Chaos than Calm. https://www.proofpoint.com/us/blog/threat-insight/zenrat-malware-brings-more-chaos-calm

[18] Github Issue #3166: Vulnerability: Sensitive Information Is Not Purged from Process Memory on app Lock or Logout. https://github.com/bitwarden/clients/issues/3166

[19] Bitwarden Community (2024) Autofill: Should We Turn It Off? https://community.bitwarden.com/t/autofill-should-we-turn-it-off/52331/4

[20] Acar, G. (2017) No Boundaries for User Identities: Web Trackers Exploit Browser Login Managers. https://freedom-to-tinker.com/2017/12/27/no-boundaries-for-user-identities-web-trackers-exploit-browser-login-managers/

[21] Flashpoint (2023) Bitwarden Password Pilfering: Understanding the Risks. https://flashpoint.io/blog/bitwarden-password-pilfering/

[22] web.dev, (2024) Third-Party Content and Privacy. https://web.dev/learn/privacy/third-parties

[23] Silver, D., Jana, S., Boneh, D., Chen, E. and Jackson, C. (2014) Password Managers: Attacks and Defenses. In: 23*rd USENIX Security Symposium* (*USENIX Security* 14). USENIX Association, 449-464. https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/silver

[24] Bitwarden Community (2024) Phishing Website: Bitwardenlogin.com. https://community.bitwarden.com/t/phishing-website-bitwardenlogin-com/49704

[25] Bitwarden (2023) Bitwarden Trademark Guidelines: Standards for Use. https://github.com/bitwarden/server/blob/main/TRADEMARK_GUIDELINES.md#standards-for-use

[26] Bitwarden Community (2023) Phishing Website: Bitwardenlogin.com. https://community.bitwarden.com/t/phishing-website-bitwardenlogin-com/49704/22?page=2

[27] Morris, R. and Thompson, K. (1978) Password Security: A Case History. *Bell Laboratories*, 594-597.

[28] Palant, W. (2022) LastPass Breach: The Significance of These Password Iterations. https://palant.info/2022/12/28/lastpass-breach-the-significance-of-these-password-iterations/

[29] Dr.-Ing. Mario Heiderich, Cure53 (2023) Pentest-Report Bitwarden Password Manager 11.2018. https://cure53.de/pentest-report_bitwarden.pdf

[30] Palant, W. (2023) A Year after the Disastrous Breach, LastPass Has Not Improved. https://palant.info/2023/09/05/a-year-after-the-disastrous-breach-lastpass-has-not-improved/#the-initial-advisory

[31] (2024) Password Hashing Competition and Our Recommendation for Hashing Passwords: Argon2. https://www.password-hashing.net/

[32] Bitwarden Community (2024) Increasing KDF Iterations. https://community.bitwarden.com/t/increasing-kdf-interations/48059/12

[33] Bitwarden (2024) Account Encryption Key—Rotate Your Encryption Key. https://bitwarden.com/help/account-encryption-key/#rotate-your-encryption-key

[34] KeePass Help Center (2024) KeePass 2.x Setup. https://keepass.info/help/v2/setup.html

[35] (2024) KeePass Links: Resources. https://keepass.info/links.html#res

[36] Kouzari, E. (2010) Software Requirements Specification—KeePass 1.10. https://keepass.info/extensions/v1/docs/SoftwareRequirementsSpecification-KeePass-1.10.pdf

[37] KeePass Help Center (2024) KeePass v1 Plugins: Security. https://keepass.info/help/v1/plugins.html#sec

[38] Vdohney (2024) KeePass-Password-Dumper. https://github.com/vdohney/keepass-password-dumper

[39] SourceForge (2023) Keepass Discussion Thread: Someone Can Read the Passwords Using Export Trigger. https://sourceforge.net/p/keepass/discussion/329220/thread/a146e5cf6b/

[40] Malwarebytes (2023) Clever Malvertising Attack Uses Punycode to Look Like Legitimate Website. https://www.malwarebytes.com/blog/threat-intelligence/2023/10/clever-malvertising-attack-uses-punycode-to-look-like-legitimate-website

[41] Bogner, M. (2016) MitM Attack against KeePass 2's Update Check. https://bogner.sh/2016/03/mitm-attack-against-keepass-2s-update-check/

[42] (2025) Icloud Data Security Overview. https://support.apple.com/en-us/102651

[43] (2025) Make Your Passkeys and Passwords Available on All Your Devices with Iphone and Icloud Keychain. https://support.apple.com/guide/iphone/passwords-devices-iph82d6721b2/ios

[44] Karthick, M.P. (2023) Compromising Google Accounts: Malwares Exploiting Undocumented OAuth2 Functionality for Session Hijacking. https://www.cloudsek.com/blog/compromising-google-accounts-malwares-exploiting-undocumented-oauth2-functionality-for-session-hijacking

[45] Microsoft (2025) Microsoft Wallet. https://www.microsoft.com/en-us/edge/features/wallet?form=MA13FJ

[46] (2025) 1password travel mode. https://support.1password.com/travel-mode/

[47] (2025) Why 1 Password Isn't Open-Source? https://1password.community/discussion/comment/114870/#Comment_114870

[48] (2025) Darklane Security Alerts and Dark Web Monitoring in Dashlane. https://support.dashlane.com/hc/en-us/articles/360000038180-Security-alerts-and-Dark-Web-Monitoring-in-Dashlane#dwmmp

[49] Guenni (2023) Vulnerabilities in Bitwarden Password Manager Browser Extension Can Reveal Passwords. https://borncity.com/win/2023/03/10/vulnerabilities-in-bitwarden-password-manager-browser-extension-can-reveal-passwords/?utm_source=chatgpt.com

[50] Bajwa, H. (2024) Password Managers Hacked: A Comprehensive Overview. https://www.beyondidentity.com/resource/password-managers-hacked-a-comprehensive-overview

[51] Bitwarden Community (2024) Clipboard Security. https://community.bitwarden.com/t/clipboard-security/36507/25

[52] Kim, J., van Schaik, S., Genkin, D. and Yarom, Y. (2023) iLeakage: Browser-Based Timerless Speculative Execution Attacks on Apple Devices. Association for Computing Machinery, 2038-2052. https://doi.org/10.1145/3576915.3616611

[53] Gibraltar Solutions (2024) Malware-As-a-Service: A Growing Threat to Modern Businesses. https://gibraltarsolutions.com/blog/malware-as-a-service/

[54] Darktrace, (2023) End of Year Threat Report 2023. https://cdn.prod.website-files.com/626ff19cdd07d1258d49238d/65d744bd0fc63765c5c1442d_Darktrace%20-%20End%20of%20Year%20Threat%20Report%202023.pdf

[55] SEKOIA (2023) Traffers: A Deep Dive into the Information Stealer Ecosystem. https://blog.sekoia.io/traffers-a-deep-dive-into-the-information-stealer-ecosystem/

[56] The Record (2024) Youtube Infostealer Campaign Uses Cracked and Pirated Video Games. https://therecord.media/youtube-infostealer-campaign-cracked-pirated-video-games

[57] Flashpoint (2023) SEO Poisoning: Threat Actors Using Search Engines. https://flashpoint.io/blog/seo-poisoning-threat-actors-using-search-engines/

[58] Flare (2024) Telegram Fraud: The Hidden Criminal Market. https://flare.io/learn/resources/blog/telegram-fraud/

[59] Sekoia (2023) Overview of the Russian-Speaking Infostealer Ecosystem: The Logs. https://blog.sekoia.io/overview-of-the-russian-speaking-infostealer-ecosystem-the-logs/

[60] Webz.io (2024) XSS: The Top Russian Dark Web Forum. https://webz.io/dwp/xss-the-top-russian-dark-web-forum/

[61] Webz.io (2024) How Dark Web Data Discovers the Hackers behind Hacking Forums. https://webz.io/blog/dark-web-api/how-dark-web-data-discovers-the-hackers-behind-hacking-forums/

[62] Lolz.guru (2024) Lolz.Guru Forum. https://lolz.guru/

[63] Webz.io (2024) Stealer Logs on the Dark Web: What You Need to Know. https://webz.io/dwp/stealer-logs-on-the-dark-web-what-you-need-to-know/

[64] SonicWall (2024) Sonicwall 2024 Mid-Year Cyber Threat Report: Iot Madness, Powershell Problems and More. https://blog.sonicwall.com/en-us/2024/07/sonicwall-2024-mid-year-cyber-threat-report-iot-madness-powershell-problems-and-more/

[65] MITRE Corporation (2019) TTP-Based Hunting. MITRE Corporation. https://www.mitre.org/sites/default/files/2021-11/prs-19-3892-ttp-based-hunting.pdf

[66] Drake, V. (2024) Threat Modeling. https://owasp.org/www-community/Threat_Modeling

[67] (2024) Microsoft Threat Modeling Tool Threats. https://learn.microsoft.com/en-us/azure/security/develop/threat-modeling-tool-threats

[68] AAG IT Services (2024) The Latest Ransomware Statistics. https://aag-it.com/the-latest-ransomware-statistics/

[69] IriusRisk (2024) Threat Modeling Methodology: STRIDE. https://www.iriusrisk.com/resources-blog/threat-modeling-methodology-stride

[70] Horsnell, C. (2024) The Problem with Developers Only Focusing on Functional Requirements. https://medium.com/@chrishorsnell/the-problem-with-developers-only-focusing-on-functional-requirements-bbef26d3a4b1

[71] Trellix (2024) Utilizing Adaptive Defense Model against Information Stealers. https://www.trellix.com/blogs/research/utilizing-adaptive-defense-model-against-information-stealers/

[72] Li, W.Z., He, W., Akhawe, D. and Song, D. (2014) The Emperor's New Password Manager: Security Analysis of Web-Based PASSWORD managers. In: 23*rd USENIX Security Symposium* (*USENIX Security* 14). USENIX Association, 465-479. https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/li_zhiwei

[73] Shahandashti, S.F. and Carr, M. (2020) Revisiting Security Vulnerabilities in Commercial Password Managers. https://arxiv.org/abs/2003.01985

[74] Nisenoff, A., Golla, M., Wei, M., Hainline, J., Szymanek, H., Braun, A., et al. (2023) A Two-Decade Retrospective Analysis of a University's Vulnerability to Attacks Exploiting Reused Passwords. In: 32*nd USENIX Security Symposium* (*USENIX Security* 23), USENIX Association, 5127-5144. https://www.usenix.org/conference/usenixsecurity23/presentation/nisenoff-retrospective

[75] Hansson, D.H. (2024) Passwords Have Problems, but Passkeys Have More. https://world.hey.com/dhh/passwords-have-problems-but-passkeys-have-more-95285df9#

[76] Microsoft (2024) Storing Data to and Reading from the Clipboard. https://learn.microsoft.com/en-us/dotnet/visual-basic/developing-apps/programming/computer-resources/storing-data-to-and-reading-from-the-clipboard?redirectedfrom=MSDN

[77] Luevanos, C., Elizarraras, J., Hirschi, K. and Yeh, J.-h. (2017) Analysis on the Security and Use of Password Managers. 2017 18*th International Conference on Parallel and Distributed Computing, Applications and Technologies* (*PDCAT*), 18-20 December 2017, 17-24. https://doi.org/10.1109/PDCAT.2017.00013

[78] Bitwarden (2024) Unlock with Pin—Understanding Unlock vs. Log in. https://bitwarden.com/help/unlock-with-pin/#understanding-unlock-vs-log-in

[79] Hive Systems (2023) Are Your Passwords in the Green? https://www.hivesystems.com/blog/are-your-passwords-in-the-green

[80] Bitwarden (2024) KDF Algorithms.

https://bitwarden.com/help/kdf-algorithms/

[81] Shainave (2024) Password Emergency Kit. https://github.com/shainave/password-emergency-kit

[82] Bitwarden Community (2024) I Created a Bitwarden Emergency Kit. https://community.bitwarden.com/t/i-created-a-bitwarden-emergency-kit/69523/6

[83] Muth, D. (2024) Diceware Passphrase Generator. https://diceware.dmuth.org/

[84] Ulikowski, M. (2024) Dnstwist—Domain Name Permutation Engine for Detecting Similar Domains. https://github.com/elceef/dnstwist

[85] URLScan.io (2024) About Urlscan.io. https://urlscan.io/about/

[86] VirusTotal (2024) How It Works. https://docs.virustotal.com/docs/how-it-works

[87] Yubico (2024) FIDO2 Authentication Standards. https://www.yubico.com/authentication-standards/fido2/

[88] Abuse.ch (2024) Raccoon Information Stealer Sample. https://bazaar.abuse.ch/sample/022432f770bf0e7c5260100fcde2ec7c49f68716751fd7d8b9e113bf06167e03/

[89] S2W Blog (2024) Raccoon Stealer Is Back with a New Version. https://medium.com/s2wblog/raccoon-stealer-is-back-with-a-new-version-5f436e04b20d

[90] S. Community (2022) Raccoon Stealer Indicators of Compromise (IOCS). https://raw.githubusercontent.com/SEKOIA-IO/Community/refs/heads/main/IOCs/raccoonstealer/raccoon_stealer_iocs_20220628.csv