

CASB-Driven Browser Extension for Data Leakage Prevention

Mukesh Kumar¹, Vidhi Patel², Diya Patel³, Hansel Baraiya⁴, Mukesh Patidar⁵, Vanshi Sharma⁶, Tanvi Patel⁷

^{1,2,3,4,5,6,7}Dept. Computer Engineering & Technology (Cyber Security), PIET, Parul University, Vadodara, India

¹mukesh.manit86@gmail.com, ²2203031260172@paruluniversity.ac.in, ³2203031260171@paruluniversity.ac.in, ⁵mukesh.patida34885@paruluniversity.ac.in ⁷tanvi.patel27691@paruluniversity.ac.in

Keywords: Password Manager, Cloud Access Security Broker, Browser Extension, Data Loss Prevention, Compliance, and Privacy.

Abstract

Cloud services have become very popular for conducting business and storing data, but security problems such as data leakage, unauthorized access, and insufficient endpoint security still prevail. CASBs have emerged to help enforce policies, prevent data leaks, and further compliance. Herein, the design and implementation of a CASB-enabled browser plugin with a variety of security features are presented, including an encrypted password manager with a master PIN for protection, a secure cloud-related URL bookmark manager, automatic clearing of browser history, cache, and cookies, and file upload detection and blocking. Each feature has a toggle switch so that the user can turn controls on/off as needed. Along with data security, the model proposal strengthens the end users' privacy while accessing cloud platforms. Efficiency and accuracy, along with other performance indicators such as false positive rates, are discussed, and comparisons are drawn with existing tools. The findings of this research indicate that deploying CASB principles at the browser level can go a long way to mitigate the security risks and enhance usability and compliance.

1 Introduction

This cloud computing services in a very short period redefined the fronts of data management, storage as well as retrieval in organizations. Cloud computing infrastructures such as AWS, Microsoft Azure, and Google Cloud offer access to very scalable, extensively flexible and cost-effective infrastructure without which an organization does not involve itself in operating and maintaining on-premise systems. Although this shift has resulted in increased efficiency and eased innovation, emerging and evolving security is equally encountered by the organizations. Some of the risks include leakage, regulatory infractions, insider threats and browser vulnerabilities. The larger worry is that most cloud interactions occur through web browsers. Though CASBs at the enterprise level should provide visibility into users' interactions with cloud service providers, enforcement of policies, they often overlook endpoint risks that can reside directly within browsers. For example, employees may unknowingly upload private documents, such as PDFs with personal information or CSVs with financial data, to unauthorized cloud services, which would bypass corporate data loss prevention controls. In addition, browsers cache, store cookies, and retain session history-all of which might be leveraged by bad actors to steal credentials, hijack sessions, or get unauthorized access. This paper proposes the development of a CASB-enabled browser plugin as a portable but powerful extension at the access point to overcome these limitations. Unlike the traditional CASBs, which operate at the network or service layer, this plugin targets the end-user environment directly, where most of the data exchanges with the cloud occur.

The plugin houses security-related modules such as sensitive file upload monitoring, automatic browsing data clearing, secure bookmark management, and encrypted password storage. Combining these features enhances data security and such improves Privacy management and compliance awareness to the end users.

1.1.1 Objectives

This research aims to Design a CASB supported browser extension along with the cloud security thoughts to data security and legal compliance. Use a number of security modules, incorporate the provision of data hygiene such as deletion of cache, cookies, and history among sensitive files monitoring and secure storage applications embedded with encrypted passwords and bookmarks. Dissect the planning of the offered plugin according to the major metrics of usability, false positive rates and detection accuracy. Simulate differentiation using the current security tools by means of increasing efficiency, compliance support, and ease of use.

1.1.2 Literature Survey

1.1.2 Title: The rapid growth of cloud computing has inspired researchers to investigate a variety of frameworks and mechanisms related to cloud service security. Particularly, Cloud Access Security Brokers have been studied extensively as intermediaries that enforce security policies between cloud providers and users. In relation to this, browser-level security enhancements in the form of plugins and extensions have gained interest as valuable tools for improving end-user security. Although CASBs have been recognized as being

helpful to prevent data loss and ensure compliance, their implementation at the browser endpoint remains largely unexplored, based on the literature review. Most research overlooks this big need for cloud compliance, data hygiene, and password management at the browser and instead focuses on solutions at the enterprise level or in isolated features related to encryption and phishing protection. Below table summarizes key related works, their main contributions, and their limitations.

Author & Year	Title	Findings	Limitations
Sharma & Gupta (2021)	"A review of cloud security frameworks"	Highlighted CASB as a key enabler for cloud compliance.	Did not address browser-side risks.
Zhang et al. (2020)	"Data loss prevention in cloud applications"	Proposed DLP algorithms for enterprise use.	Focused only on enterprise, not end-user browser-level controls.
Alharthi et al. (2021)	"Survey on CASB adoption in enterprises"	Showed increase in adoption in banking/health care.	Did not explore password/data hygiene features.
Patel & Mehra (2019)	"Browser extension security enhancements."	Demonstrated plugins can mitigate phishing.	Did not integrate CASB principles.
Li & Chen (2020)	"Secure Storage of Credentials in Browser Extension"	Suggest AES-Based encryption for passwords.	Lacked integration with cloud access p.

2 Proposed Model

A browser extension with CASB enabled is recommended in order to enhance cloud security at the browser level. This is where most of the cloud interactions are done. This model embeds lightweight, user-friendly in-browser security modules that can be enabled or disabled depending upon specific needs, where traditional CASB enterprise solutions are more focused on the enforcement of policies between the organization and cloud provider. This gives a balance between usability and compliance, as flexibility is gained without shaking security. The extension consists of four key security modules.

1. Sensitive File Upload Monitoring

- The plugin constantly monitors the browser's file upload activity.
- It provides high risk file formats that generally contain private or sensitive information, such as CSV, PDF, and DOCX. According to the policy, the upload will either be blocked or notified when it is detected in order to eliminate any accidental or unauthorized data leakages.

2. Auto-Clear Browser Data

This automatically deletes cookies, cache and surfing history when browser closes or when a user logs out thus minimizing digital footprints. This aspect will assist in eliminating session hijacking, theft of credentials and unauthorized access to stored browsing artifacts.

3. Secure Bookmark Manager

An encrypted bookmark manager could be used for safe storage of the URLs related to the cloud. Such a system will ensure encryption of local storage, which is in contrast to

traditional browser bookmarks stored in plaintext. This reduces the risks such as phishing, tampering, or unauthorized disclosure.

4. Built-in Password Manager

- The extension includes an easy password jump that securely stores user credentials.
- Passwords are encrypted with AES-256 and can only be accessed with a master PIN, so although the local storage is compromised, the credentials will not be able to be read.

One of the advantages of this design is the toggle mechanism. Because each of the four modules comes with an on/off switch, it allows users to turn on only those features that they need at any particular time. Because of its adaptability, the plugin can put up a variety of use cases without charging the user with strict enforcement.

A. Flow Diagram



The user accesses the cloud platforms through the browser. The activity is monitored in real time by the Plugin Interceptor. The CASB Policy Engine evaluates whether the action is in compliance with established security policies. A decision is executed that allows the activity, blocks it, or securely logs the pertinent data. In this workflow, gaps in traditional CASB solutions are closed by the assurance that every cloud-related action is subject to policy enforcement earlier to its execution.

1.1.3 System Architecture

The extension is proposed on the basis of modular architecture as it consists of the following items.

- **Frontend:** Easy to use and simple interface featuring four modules which can be turned on/off by simple switches. Design as ease of use comes at first to ensure simplicity in the management of security settings by the user.
- **Backend:** A layer of service personnel is the enforcement layer of the CASB model. The back-end staff manages the program and keep track of the activities occurring on the browser they are able to include uploaded files, abandoned sessions, and even a request to save a password.
- **Encryption Layer:** The sensitive data such as bookmarks or passwords required to keep safe are encrypted using AES-256 so that the information can stay confidential even when an attacker has used local storage.
- **Database:** The storage system of browser stores the data in encrypted form. The encryption layer also minimizes the risk of compromise since it is not exposed to raw data.

2. Methodology

The browser extension created by CASB was created according to a balanced point of view on security, usability, and compliance. The delicate data security process entails the determination of a relevant framework, tool, engagement of modules in parts and encryption methods. The methodology is divided into two significant sections: instruments and frameworks used and how the whole development process will take place in general systematic manner.

A. Tools and Frameworks

To implement the suggested model, the following mixes of browser APIs, programming languages, and libraries of security were used:

- **Browser API:** The extension was designed keeping in mind the state of art standards of browser extension. The extension front-end interface is developed with three basic web technologies, namely, JavaScript, HTML, and CSS. They are what make Toggle switches, event handlers, and popup user interfaces, which are used to control the security modules of the plugin, possible.
- **Flask-for-exhibits:** This interface is developed based on a simple web framework of Python, which shows the interaction between the browser extension and the cloud systems in controlled test setups.
- **AES-256 Encryption (done via Crypto JS library):** The encryption of passwords and safe bookmarks is carried out with the help of the AES-256 standard. The cryptographic capabilities of the Crypto JS library will allow high data confidentiality.

B. Development Steps

The CASB browser extension is developed in a systematized sequence, which includes the following steps:

1. Capture Browser Events

- The extension hears major events in the browser regarding closing of tabs and file upload. They are usually trigger points to fire on the security modules of the plug-in.

2. Apply CASB Security Policies

On detection, the engine of this extension, which is a policy, evaluates an action performed with respect to fixed security rules. This includes blocking unauthorized uploads of files, clearing of the cookies, browser history as well as clearing the cache automatically to prevent session hijacking.

3. Encrypt sensitive user data before storage.

- Secrecy Store sensitive user data in encrypted format.

- AES-256 encrypts passwords and bookmarks, as well as other information, and then stores the encrypted data in local storage by the browser. This is to make sure that although the local storage is violated, the sensitive data will not be read with the master PIN.

4. Popup UI Display Controls

The user will be provided with a popup.html interface whereby they are provided with toggle switches to enable or shut off modules.

- This is an intuitive way of accessing the security preferences in real time.

The suitability of the proposed CASB was proven by a number of tests through various security modules. The key performance indicators that were selected to be measured during Evaluation were aimed at capturing the elements of accuracy and usability of security:

1. **Accuracy (%)** - This ratio is calculated based on sensitive files or events that are properly identified by the extension.

A higher accuracy is a mark of strong detection capability with effective enforcement of CASB policies.

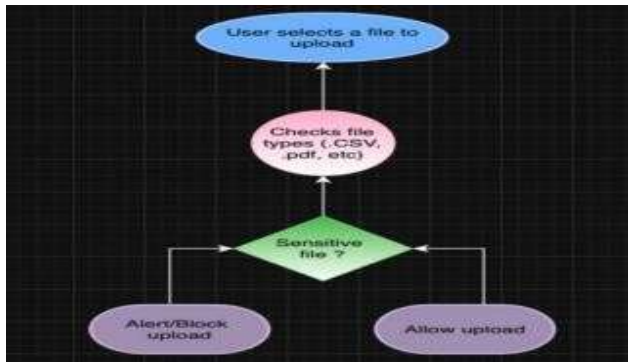
2. **False Positive Rate (FPR %)** - It indicates the ratio of the nonsensitive files or safe events which are wrongly labeled as risky. The lesser the FPR, the better, because too many false alarms will eventually irritate users and weaken their trust in the system.

3. **Execution Time (Latency in MS)** - How much latency the extension introduces during uploads monitoring or processing. It should have very low latency to not interfere with a good browsing experience.

VI. FEATURES

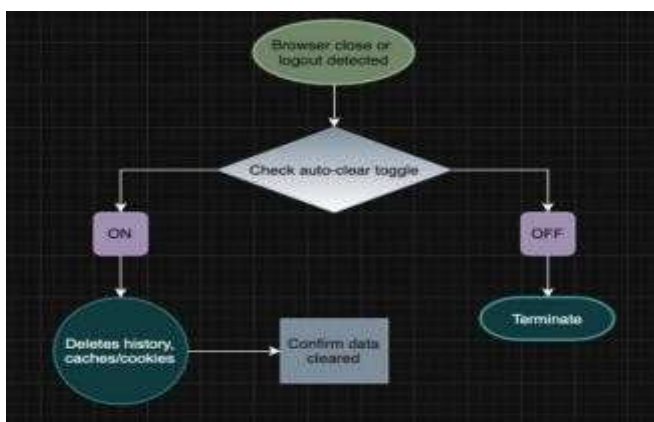
A. Feature 1: Sensitive file upload monitor (DLP)

This feature acts as a DLP mechanism through monitoring file uploads. When the user selects a file that is to be uploaded, the system determines the type of the file. The system identifies whether there is sensitive information in the file. If the file is sensitive, the system will either alert the user or prevent the upload to stop data leaks. Otherwise, if the file is not sensitive, the upload can proceed. This ensures that personal data will not get intentionally or unintentionally uploaded to unapproved websites.



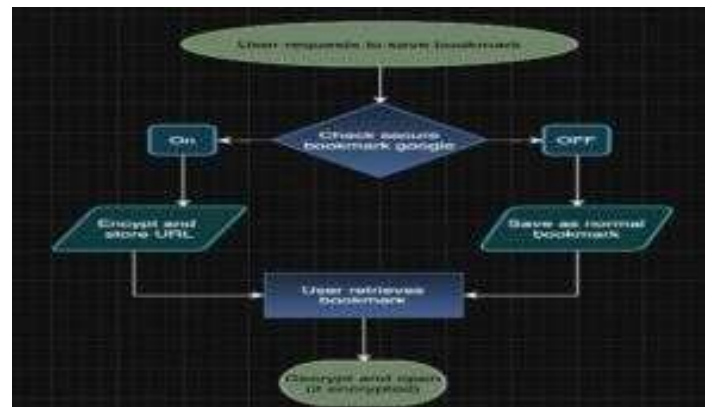
B. Feature 2: Auto clear browser data

This feature will automatically clear all of your browsing data, including cookies, caches, and history, once you either log out or close the browser. This feature can be enabled or disabled using a simple toggle. If the auto-clear toggle is turned on, the system deletes your data and verifies the clearance thereof. In case it is turned off, the process stops right there, and your data stays on the device. This will provide a useful method for enhancing your privacy by making sure that no browsing data is stored beyond your session.



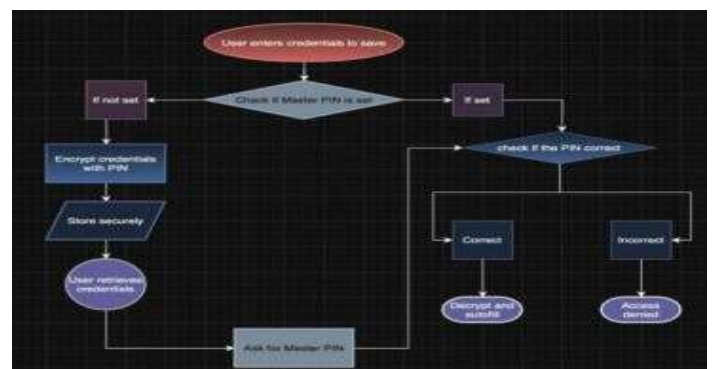
C. Feature 3: Secure bookmark manager

This feature provides a secure way to manage browser bookmarks. When the user requests to save a bookmark, the system determines if the secure bookmark feature is turned on. If it is, then the URL is encrypted and retained in a secure manner. If the secure bookmark feature is off, then the URL is saved as a regular bookmark, without encryption. Later, when the user retrieves and opens the bookmark, the system will automatically decrypt the URL, if it was saved securely, thereby protecting potentially private information in bookmarks.



D. Feature 4: Password manager with encryption

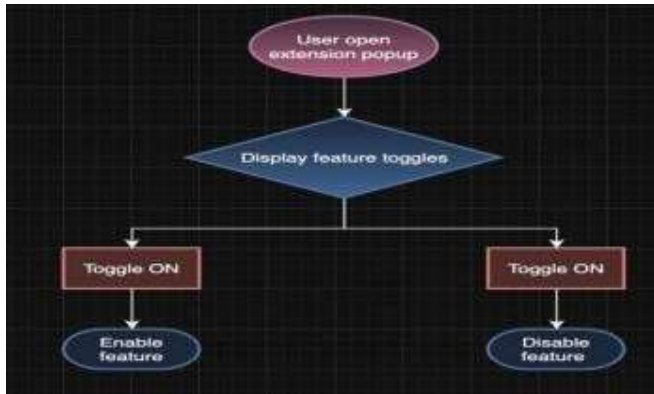
This feature securely stores and manages user credentials. After a user selects to save login credentials, the system will check if a Master PIN has been set. If not, it uses the PIN for saving the credentials securely and encrypting them. If there is already a Master PIN set, then users are asked to enter their saved credentials. A confirmation is then made by the system whether the correct PIN has been entered. If so, auto-entering and decrypting take place. If incorrect, access will be denied. Since this ensures that all saved credentials are encrypted and accessible only with the correct Master PIN, the security greatly increases.



E. Feature 5: Toggle system

This system allows users to manage an extension's many features through a simple interface. For any extension, it lists feature toggles when the user opens the popup window of the

extension. This allows the user to toggle on/off any feature by changing between the ON/OFF states, thereby activating or deactivating a particular feature. In this way, the user can easily manage the functionalities of an extension according to his preference.



3 Results

The proposed browser extension is an important addition to the security at the browser level since it condenses the principles of CASB into a small and malleable plug-in. This is comparable to a holistic platform, as opposed to the collection of existing tools, such as LastPass as a password management add-on overall or DLP as a standalone tool that would cover a very narrow spectrum of protection of data. This, as the recommendation, consolidates several of its functions-sensitive files tracking, data hygiene automation, safe bookmark storage, and encrypted passwords-in a single module. The practicability of this design was confirmed by the user testing feedback. Stating such, an example of these is the test participants such as the toggle-based controls which, depending on the whims of the user, can be turned on or off and furthermore made to be even more flexible. The very thought that it leaves no information behind when the browser is logged out or upon closing it gave them confidence because it has been directly counteracted against the anxiety of users as to the privacy and misuse of their data. These findings reveal that the browser extension has earned the confidence of the end user besides being competent in implementing security policies. On the other hand, performance testing showed that there was a minor overhead in scanning at file upload. This was slight and could be tolerated in enterprise settings where gains in terms of security are considered initially as compared to inconveniences of a little bit. All this leads in the cumulative direction of placed CASB functions being at the browser end having a perfect balance between a strong policy enforced and user convenience. The future roadmap can then be set up to include latency reduction in the scanning of large files, expansion of browser support, and AI-driven content analysis, which may further make the model accurate and scalable.

Conclusion & Future Work

This research has shown that developing a multi-feature CASB browser extension is a practicable and effective way to enhance endpoint cloud security. With this proposed solution, the risks of leakage, regulatory non-compliance, and user privacy intrusion are reduced by embedding the core features that include tracking sensitive file uploads, automatic data clearing, secure storage of bookmarks and passwords, and user controls through toggling. This extension offers a single source for security within a lightweight browser module and ensures usability will not be sacrificed, which generally characterizes existing single-function tools. Given that this model is representative of the best balance between strong security enforcement and flexibility in operational function, user testing results point toward its being a strong candidate for enterprise-class adoption.

B. Future Work

Future improvement will focus compatibility, scalability, and intelligence. First is the use of AI-powered anomaly detection to detect complex threats that cannot be covered through static policy rules and abnormal browsing behavior. Second is enterprise integrations that will allow central monitoring, analytics, and policy management within CASB dashboards. Third, cross-browser support for Safari and Firefox, together with mobile browser extensions for Android optimization, will be given top priority to avoid latency in the uploading of big files and ensure scalability for high-traffic enterprise environments.

6 References

- [1]. Jones Sharma, R., & Gupta, V. (2021). A Review of Cloud Security Frameworks. *International Journal of Cloud Security*, 12(3), 45-59.
- [2]. Zhang, L., Kumar, S., & Patel, D. (2020). Data Loss Prevention in Cloud Applications. *IEEE Transactions on Cloud Computing*, 8(4), 561-574.
- [3]. Alharthi, M., Hussain, F., & Yang, J. (2021). Survey on CASB Adoption in Enterprises. *Journal of Information Security*, 15(2), 123-137.
- [4]. Patel, M., & Mehta, R. (2019). Browser Extension Security Enhancements. *ACM Digital Library*, 21(5), 8895.
- [5]. Li, H., & Chen, Y. (2020). Secure Storage of Credentials in Browser Extensions. *Journal of Applied Cryptography*, 7(1), 32-44. protect both the password and sensitive data stored within the browser extension.