International Journal of Recent Advances in Multidisciplinary Topics
Volume 4, Issue 6, June 2023
https://www.ijramt.com | ISSN (Online): 2582-7839

Data Leakage Detection Using Cloud Computing

Zakiya Banu[1*], Chandrika Prasad[2]
[1]Student, Department of Computer Science and Engineering, Ramaiah Institute of Technology, Bangalore, India
[2]Assistant Professor, Department of Computer Science and Engineering, Ramaiah Institute of Technology, Bangalore, India

***Abstract*: Data leakage refers to the unintentional or accidental disclosure of confidential or sensitive information to unauthorized individuals or entities. This poses a significant challenge for businesses, as the frequency and costs associated with data leakage incidents continue to rise. The issue of data leakage is compounded by the lack of regulation and monitoring for transmitted data, including emails, instant messaging, file transfers, and web forms. This makes it difficult to track and control the flow of data to its intended recipients. In our research, we focus on a specific problem: a data distributor entrusts sensitive data to a group of agents who are deemed trustworthy. However, some of the data is leaked and found in unauthorized locations, such as the internet or unauthorized devices. The distributor must assess the probability that the leaked data originated from one or more of the entrusted agents, rather than being obtained independently through other means. To address this challenge, we propose data allocation strategies among the agents that aim to improve the likelihood of identifying data leakages. Importantly, our methods do not rely on modifying the released data through techniques like watermarks. In certain cases, we also explore the possibility of introducing realistic but fabricated data records to further enhance our ability to detect leaks and identify the responsible party.**

***Keywords*: Data leak, leaks, guilty agents.**

1. Introduction

Data leakage, also known as data breach or data loss, is a critical concern for individuals, industries, and institutions alike. It involves unauthorized access to important data, which can range from customer information and business plans to financial records and patient data, depending on the nature of the organization or individual involved. While sharing data with employees or customers is often necessary, it increases the risk of sensitive information falling into the wrong hands. Data leakage can occur due to various reasons, including accidents, mistakes, or deliberate actions by both internal and external parties. External threats, such as hackers and cybercriminals, pose a significant risk to the security of sensitive data.

The consequences of a data leak can be severe, causing significant harm and damage to the affected organization or individual. To combat data leakage, organizations employ various measures, including data leak detection systems. These systems typically utilize techniques like set intersection, comparing sets of n-grams derived from the content and sensitive data to identify potential leaks. However, it is important to note that no system is fool proof, and constant vigilance is required to stay ahead of evolving threats.

One effective security measure is the implementation of Forms Authentication, which enhances website security and prevents unauthorized access to sensitive data. This technique ensures that only authenticated users can access the website, reducing the risk of data leakage. By promptly notifying administrators about potential data leaks, they can take immediate action to mitigate the damage and protect sensitive information. In summary, data leakage is a significant and complex issue affecting individuals, industries, and institutions.

A. Scope

The main scope of this project is to provide complete information about the data/content that is accessed by the users within the website. Authentication technique is used to provide security to the website in order to prevent the leakage of the data and identify the agent that leaked the data. The files are watermarked by the agent's username when downloaded.

2. Literature Survey

Data leakage refers to the unintended or accidental disclosure of confidential or sensitive information to unauthorized individuals or entities. This sensitive data can include intellectual property, financial details, patient records, personal credit card information, and other pertinent information depending on the nature of the business and industry. Moreover, in numerous instances, sensitive data is shared among various parties such as remote employees using personal devices, business partners, and customers. This amplifies the potential for unauthorized access to confidential information. Whether resulting from deliberate actions or inadvertent errors, originating from insiders or external actors, the exposure of sensitive information can inflict significant harm on an organization.

A. Literature Review

In reference [1], the author utilizes a sequence alignment method to detect complex data-leakage patterns. This algorithm is employed to identify significant and lengthy data patterns. Additionally, they incorporate a sampling algorithm to assess the similarity between independently tested sequences. The combination of these techniques achieves high accuracy in detecting transformed leakage.

*Corresponding author: zakiyarasheed111@gmail.com

Two algorithms for searching and detecting transformed leakage information are implemented by the author in [2]. This framework demonstrates superior accuracy compared to existing inspection systems and exhibits strong scalability when parallelized on graphics processing units (GPUs), making it suitable for large organizations.

The authors propose a privacy-preserving data-leak detection system called fuzzy fingerprinting in [3]. This system utilizes special digests to minimize the exposure of sensitive data during detection. The authors have conducted tests to verify the accuracy, privacy, and efficiency of their proposed solutions.

In [4], the author introduces the Aquifer security system, which imposes host export limitations on data involved in user interface (UI) workflows. The system recognizes that data sharing is often part of a larger workflow involving multiple applications. By allowing applications to retain control over their data throughout the workflow, the system enhances security and ensures data privacy.

In their publication [5], the authors present Attire, an application for computers and smartphones that utilizes an avatar to convey real-time data exposure. This lightweight and unobtrusive approach updates the avatar's clothing to provide the user with visual feedback on data exposure.

In reference [6], the authors present the Data-Driven Semi-Global Alignment (DDSGA) method. DDSGA enhances scoring systems by employing unique alignment parameters for each user, thus improving security effectiveness. It allows for slight modifications in the low-level representation of command functionality to accommodate transformations in user command sequences. Furthermore, DDSGA adapts to changes in user behavior by updating the user's signature based on their current behavior. To optimize runtime overhead, DDSGA reduces alignment overhead, parallelizes detection, and incorporates modifications.

In [7], the author proposes a novel method to extract richer semantics from user determinants. The technique leverages the observation that, in text-based applications, user determinants are displayed as text on the screen and users make modifications if needed. A prototype called Gyrus is developed to enforce proper application behavior based on user determinants. Gyrus effectively prevents destructive activities, such as social network impersonation attacks and online financial services fraud, by analyzing and validating network traffic. Evaluation results demonstrate Gyrus's success in thwarting modern malware and its resilience against future attacks. Performance analysis confirms Gyrus as a viable solution for standalone PCs with continuous user interaction, filling an important gap in security measures that consider user attention when assessing network traffic legitimacy.

In [8], the authors implement a domain-specific concurrency model that supports a wide range of Intrusion Detection System (IDS) analyses, regardless of the specific detection technique used. The implemented technique divides the stream of network events into subsets processed by the IDS, ensuring that each subset contains events relevant to a specific detection case. This partitioning method is based on the concept of detecting common applicability. The designed model can support both simple, per-flow detection techniques and more complex, high-level detectors.

According to the findings of author [9], the detection of essential data becomes challenging due to transformations and unpredictable leakage patterns caused by insertions and deletions in the content. Existing automata-based string matching algorithms are not suitable for detecting transformed data leakage due to their complexity and lack of required regular expressions. The authors propose two novel algorithms that achieve high detection precision in identifying changed leaks compared to state-of-the-art inspection techniques. They parallelize their design on graphics processing units and demonstrate the scalability of their data leakage detection system when analyzing large datasets.

In paper [10], the authors highlight the limitations of apparent distance metrics used to compute behavioral similarity between network hosts. These metrics fail to capture the semantic significance of network protocols and neglect the long-term temporal structure of the objects being analyzed. To address these issues, the authors introduce a new behavioral distance metric for network hosts that incorporates semantic and temporal attributes. They compare its performance with a metric that disregards such information, emphasizing the importance of considering semantics and temporal aspects in analyzing network data.

Shoulin Yin et al. [11] introduce the concept of searchable asymmetric encryption, which enhances information protection and prevents the leakage of user search criteria (search patterns) in encrypted data. This technique contributes to security and search operations on encrypted data, ensuring the privacy of user search queries.

3. Proposed System

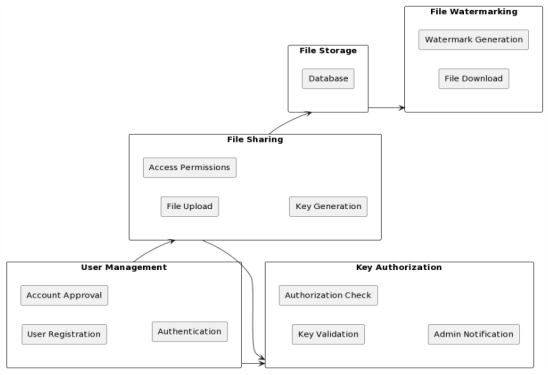


Fig. 1. System architecture

Data leakage detection involves assessing the likelihood of a user being responsible for a leak. This assessment is based on the overlap between their data and the leaked data, as well as the data of other users. The probability of objects being guessed by other means is also considered. The module's main goal is to provide comprehensive information on user-accessed data within a website. Forms Authentication is used to enhance website security and prevent data leakage.

Authentication technique is used to provide security to the website in order to prevent the leakage of the data and identify

the agent that leaked the data. The files are watermarked by the agent's username when downloaded.

The diagram illustrates the high-level modules and their relationships in a system related to user management, file sharing, key authorization, file storage, and file watermarking.

1) *User Management*
   - User Registration: Handles the process of registering new users.
   - Account Approval: Manages the approval of user accounts.
   - Authentication: Deals with user authentication and verifying user identity.
2) *File Sharing*
   - File Upload: Allows users to upload files to the system.
   - Access Permissions: Manages the permissions and access rights for shared files.
   - Key Generation: Generates secret keys required for accessing shared files.
3) *Key Authorization*
   - Key Validation: Validates the secret keys provided by users.
   - Authorization Check: Ensures that users have the necessary authorization to access shared files.
   - Admin Notification: Notifies administrators about any unauthorized access attempts.
4) *File Storage*
   - Database: Represents the storage mechanism for files and related data.
5) *File Watermarking*
   - Watermark Generation: Adds watermarks or unique identifiers to files to track their usage.
   - File Download: Enables users to download files from the system.

4. Implementation

Data leakage detection involves assessing the likelihood of a user being responsible for a leak. This assessment is based on the overlap between their data and the leaked data, as well as the data of other users. The probability of objects being guessed by other means is also considered. The module's main goal is to provide comprehensive information on user-accessed data within a website. Forms Authentication is used to enhance website security and prevent data leakage.

A. Problem Setup and Notation

The distributor possesses a collection of valuable data objects represented as set T = {t1, t2, ..., tm}. The distributor intends to share these objects with a group of agents, denoted as U1, U2, ..., Un. However, it is crucial that the objects are not disclosed to any unauthorized third parties. The objects in set T can vary in type and size, such as tuples in a relation or entire databases.

Each agent Ui receives a subset of objects, denoted as Ri, which is determined either through a sample request or an explicit request. The two types of requests are defined as follows:

1) *Sample request Ri = SAMPLE(T, mi)*
   - This type of request allows Ui to obtain any subset of mi records from the set T.
   - Ui may receive a random sample or a specific sample based on predetermined criteria.
2) *Explicit request Ri = EXPLICIT(T, condi)*
   - With this type of request, Ui is provided with all objects from set T that satisfy the specified condition condi.
   - Ui receives the complete set of objects that meet the given condition.

In summary, the distributor facilitates the sharing of data objects from set T with agents Ui, either through sample requests or explicit requests, ensuring that the objects are only disclosed to authorized recipients while maintaining confidentiality.

B. System Features

The system encompasses several key features to facilitate effective data leakage detection using cloud computing. The User Management feature ensures secure user registration and authentication, with admin approval for account creation. The File Sharing capability enables agents to upload files to the cloud storage, define access permissions, and generate unique keys for secure file access. The Key Authorization module validates key requests, authorizing agents to access specific files while promptly notifying the admin of any unauthorized attempts. To enhance data traceability, the File Watermarking functionality adds unique watermarks, such as the agent's name or identifier, to downloaded files, aiding in the identification of leaked content. Together, these features provide a robust foundation for data protection and monitoring within the system.

C. Information About the implementation of Modules.

The User Management Module is responsible for handling the registration and authentication process of agents. It allows agents to create accounts by providing necessary details, and their accounts are approved by the admin. This module ensures that only authorized agents can access the system and share files. It includes functionalities such as registration form validation, account approval workflow, and login/logout features.

The File Sharing Module facilitates secure sharing of files among agents. It enables agents to upload files to the cloud storage and specify access permissions. Each shared file is assigned a unique key that is required to access and download the file. The module also tracks the usage of keys to identify any unauthorized attempts to guess the key. This ensures that only authorized agents can access the shared files, enhancing the security of sensitive data.

The Key Authorization Module handles the process of authorizing key requests from agents. It validates the requests to check if agents are authorized to access specific files. If the request is legitimate, the module provides the requested key to the agent, allowing them to access the corresponding file. In case of unauthorized or suspicious activity, such as attempts to

guess the key, the module notifies the admin and takes appropriate action, such as blocking the leaker's account. This module acts as a crucial gatekeeper to prevent unauthorized access and potential data leaks.

The File Watermarking Module adds watermarks to files downloaded by agents. It embeds the agent's name or a unique identifier as a watermark within the file. This watermark serves as a means to identify the source of leaked files, should any unauthorized sharing occur. By prominently displaying the agent's information, the module discourages agents from sharing files without permission and enhances accountability.

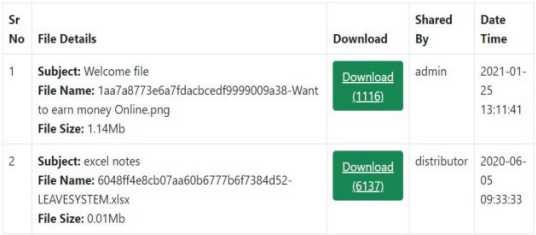
The Admin Notification Module is responsible for sending notifications and alerts to the admin. It keeps the admin informed about key requests, suspicious activities, and potential data leaks. These notifications enable the admin to stay vigilant and take necessary actions promptly. Depending on the severity of the situation, the admin can choose to block accounts, investigate reported incidents, or implement additional security measures. This module ensures that the admin has full visibility and control over the system, contributing to effective data leakage detection and prevention.

These modules provide a solid foundation for implementing data leakage detection using cloud computing. These modules provide a basic framework for implementing data leakage detection using cloud computing with features like secure file sharing, key authorization, watermarking, and admin monitoring.

5. Results

Sr No	Request By	File	Confirm	Date_time
1	user1	**Subject:** excel notes **File Name:** 6048ff4e8cb07aa60b6777b6f7384d52-LEAVESYSTEM.xlsx **File Size:** 0.01Mb	Confirmed	2020-06-05 09:29:57

Fig. 2. Key request list



Sr No	File Details	Download	Shared By	Date Time
1	**Subject:** Welcome file **File Name:** 1aa7a8773e6a7fdacbcedf9999009a38-Want to earn money Online.png **File Size:** 1.14Mb	Download (1116)	admin	2021-01-25 13:11:41
2	**Subject:** excel notes **File Name:** 6048ff4e8cb07aa60b6777b6f7384d52-LEAVESYSTEM.xlsx **File Size:** 0.01Mb	Download (6137)	distributor	2020-06-05 09:33:33

Fig. 3. File shared details

6. Conclusion

In the proposed system we have implemented a secure file sharing mechanism. When a user wants to share a file with others, a unique secret key is generated. To access or download the file, other users need to request this secret key from the file sender. Once the sender authorizes the request, the user can use the provided secret key to download the file.

However, if someone tries to download the file without obtaining the secret key through unauthorized means, such as guessing the key, it will be identified as a potential data leaker. In such cases, the system will raise an alert and notify the sender and the administrator about the unauthorized access attempt. This helps in detecting and preventing data leakage by identifying and blocking potential leakers from accessing sensitive files without proper authorization.

References

[1] Shu, Xiaokui, et al., "Privacy-Preserving Detection of Sensitive Data Exposure." *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 5, May 2015, pp. 1092–103

[2] Madhavi R. Suryawanshi et al., "Survey on Privacy-Preserving Detection of Sensitive Data Exposure." *International Journal of Science and Research*, vol. 4, no. 12, Dec. 2015, pp. 632–34.

[3] Suraj S. Morkhade et al., "A Survey on Data Mining Based Intrusion Detection Systems." International Journal of Application or Innovation in Engineering & Management, vol. 2, no. 3, pp. 338-343, March 2013.

[4] Magdy, A., M. Mahros, and E. Hemayed, "Firewall-based Solution for Preventing Privilege Escalation Attacks in Android", International Journal of Computer Networks and Communications Security, vol. 2, no. 9, pp. 318–327, 2014.

[5] Blomberg, Jeanette, and Helena Karasti. "Reflections on 25 Years of Ethnography in CSCW." *Computer Supported Cooperative Work (CSCW)*, vol. 22, no. 4–6, Springer Science and Business Media LLC, Jan. 2013, pp. 373–423.

[6] "Detection of Masquerade Attacks by Using DDSGA: Data-Driven Semi-Global Alignment Approach with CIDS Framework." *International Journal of Science and Research*, vol. 6, no. 1, pp. 1164–67, Jan. 2017.

[7] McParland, Chuck, et al. "Monitoring Security of Networked Control Systems: It's the Physics." *IEEE Security & Privacy*, vol. 12, no. 6, pp. 32–39, Nov. 2014,

[8] Lin, Yi-Shan, et al. "A Capability-Based Hybrid CPU/GPU Pattern Matching Algorithm for Deep Packet Inspection." *International Journal of Computer and Communication Engineering*, vol. 5, no. 5, pp. 321–330, 2016.

[9] Yaji, Sharath, and B. Neelima. "Parallel Computing for Preserving Privacy Using K-anonymisation Algorithms from Big Data." *International Journal of Big Data Intelligence*, vol. 5, no. 3, p. 191, 2018.

[10] Yue, Han, et al. "Graph–Graph Similarity Network." *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–11, 2022.

[11] Coull, Scott E., et al. "Access Controls for Oblivious and Anonymous Systems." *ACM Transactions on Information and System Security*, vol. 14, no. 1, pp. 1–28, May 2011.

[12] Yin, Shoulin, et al. "Distributed Searchable Asymmetric Encryption." *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 4, no. 3, p. 684, Dec. 2016.