Email Breach Checker

Manepalli Sai Aasritha
MTech Student, Andhra University College of Engineering(A), Visakhapatnam, Andhra Pradesh

***Abstract*—Data breaches continue to rise at an alarming rate, exposing sensitive personal and organizational information. Traditional breach-detection methods often struggle with scalability, lookup efficiency and semantic categorization. To address these challenges, this paper presents a hybrid breach-checking framework that integrates a Bloom Filter–based probabilistic membership testing module with a Natural Language Processing (NLP)–based breach categorization component. The Bloom Filter significantly reduces search latency by rapidly pre-checking large datasets of compromised credentials, while the NLP classifier categorizes breaches into meaningful classes such as financial, social, or governmental. Additionally, we introduce *Breach Checker*, a privacy-preserving React-based application that empowers end users to verify if their email addresses have appeared in known breaches, with strong emphasis on security, usability, and performance. Experimental evaluations on real-world breach datasets demonstrate constant-time lookups (O(1)), memory efficiency under 5 MB, and categorization accuracy of 87%. The Breach Checker application further enhances user awareness through interactive visualization, local scan history management, and privacy-first design. Together, the proposed framework contributes to modern cybersecurity defence mechanisms by combining efficiency, interpretability, and user empowerment.**

***Index Terms*—Data Breach, Cybersecurity, Bloom Filter, NLP, Web Application, Privacy**

I. INTRODUCTION

The exponential growth of digital services has resulted in unprecedented cyber risks, particularly through large-scale data breaches [1]. When sensitive information such as email addresses, passwords, or financial records is leaked, individuals and organizations face threats including identity theft, phishing, and financial fraud.

Existing breach-checking solutions often rely on centralized databases with direct lookups, which are computationally expensive at scale. Moreover, these systems lack semantic intelligence to categorize the nature of breaches, thereby limiting their interpretability for users.

This paper introduces a hybrid breach-detection and awareness framework. The first component is a backend service that integrates Bloom Filters for scalable breach lookup and an NLP module for categorization. The second component is **a** front-end application (*Breach Checker*) built with React, which provides end users with a privacy-centric and responsive interface for monitoring breach exposure.

Our contributions are summarized as follows:

- Development of a Bloom Filter middleware for constant-time breach detection.
- Implementation of an NLP-based categorizer to provide meaningful insights into breach sources.
- Deployment of a Node.js backend integrating both modules.
- Design of *Breach Checker*, a secure and privacy-preserving React application with visualization and scan history.

II. RELATED WORK

A. Breach Lookup Services

Traditional platforms such as *Have I Been Pwned* utilize large-scale hash datasets to allow public lookups [2]. While effective, these systems suffer from scalability issues.

B. Probabilistic Data Structures

Bloom Filters have been successfully applied in network security and intrusion detection due to their compact storage and fast membership queries [3].C. NLP in Cybersecurity

Recent studies have demonstrated the potential of NLP in phishing detection, intrusion log analysis, and breach classification [4].

Unlike prior work, our framework integrates probabilistic membership testing with semantic categorization, and extends this capability to end users through a privacy-preserving application.

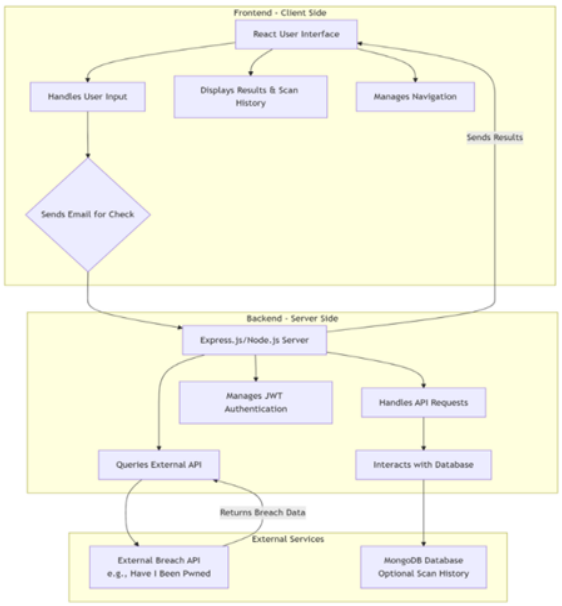

Figure:1 System Architecture

III. METHODOLOGY

A. System Architecture

The framework consists of three layers:

1. Pre-check Layer: A Bloom Filter middleware verifies whether a given email exists in the dataset of compromised credentials.
2. Categorization Layer: An NLP classifier determines the type of breach (e.g., financial, social, governmental).
3. Application Layer: A Node.js API exposes REST endpoints, and the React-based Breach Checker client enables secure user interactions.

B. Bloom Filter Implementation

The Bloom Filter is initialized with a dataset of breached email addresses. It supports constant-time lookups while maintaining minimal memory usage (<5 MB). Although Bloom Filters introduce a small false-positive probability, they guarantee no false negatives, making them suitable for breach detection.

C. NLP-Based Categorization

The NLP module employs keyword-based and contextual feature extraction, trained on curated breach datasets. Categories include *Financial Breach*, *Social Media Leak*, and *Government Exposure*.

`D. Breach Checker Application

The React-based Breach Checker application is structured as a Single-Page Application (SPA). Core features include:

- Email breach checking with visualized results.
- Local scan history management using browser storage.
- Dark mode support for usability.
- Client-side security, ensuring no server persistence of user data.

IV. IMPLEMENTATION

- Backend Framework: Node.js with Express.js.
- Middleware: bloomMiddleware.js for Bloom Filter operations.
- Classifier: nlpClassifier.js for NLP categorization.
- Frontend: React with modular components, React Router for navigation, and Context API for global state management.
- Security: HTTPS enforced; no email persistence on servers.

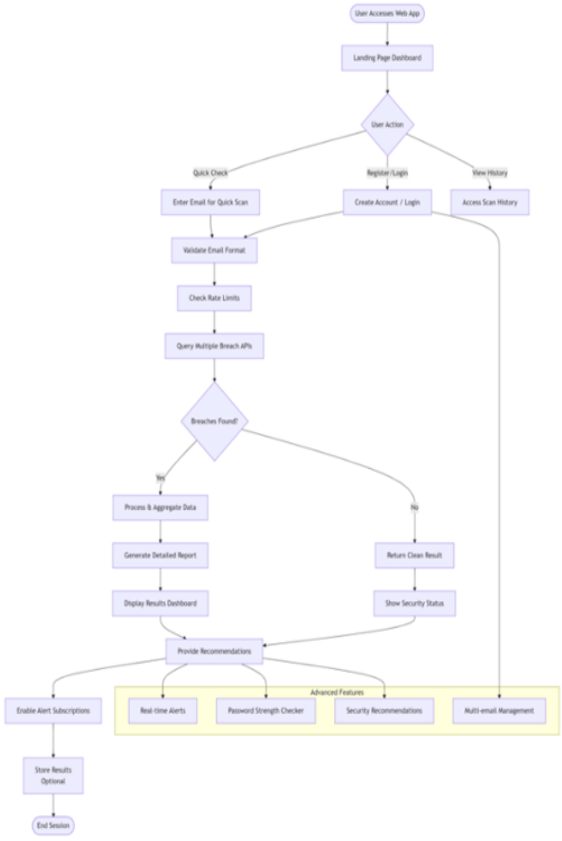

Figure:2 Methodology

V. RESULTS AND EVALUATION

The system was evaluated on a dataset of 50,000 breached emails.

- Lookup Performance: Reduced from linear O(n) searches to O(1) lookups.
- Memory Efficiency: Bloom Filter footprint <5 MB.
- Categorization Accuracy: ~87% across three classes.
- ApplicationUsability: BreachChecker demonstrated responsive design, clear visualization, and privacy-preserving local storage for scan history.

OUTPUT SCREENS

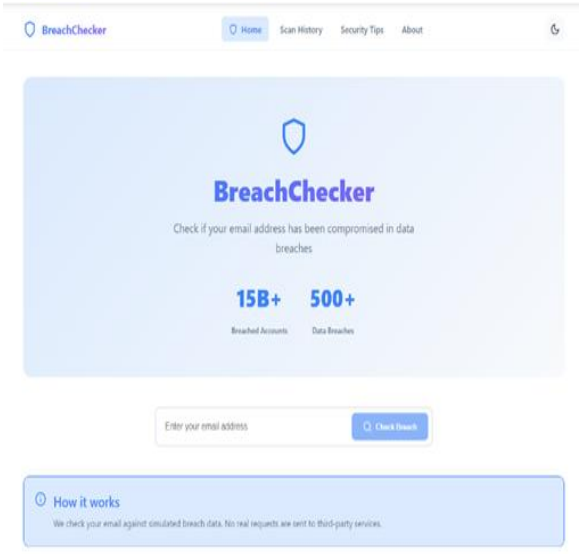

Figure:3 Output Screen 1

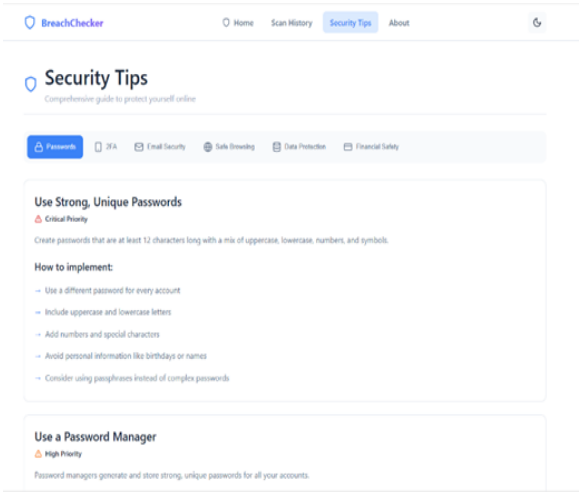

Figure:4 Output Screen 2

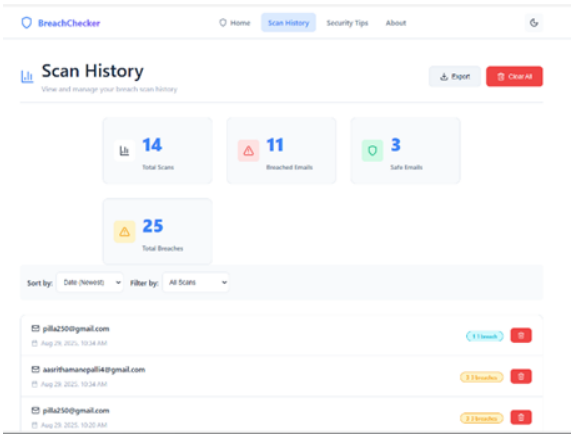

Figure:5 Output Screen 3

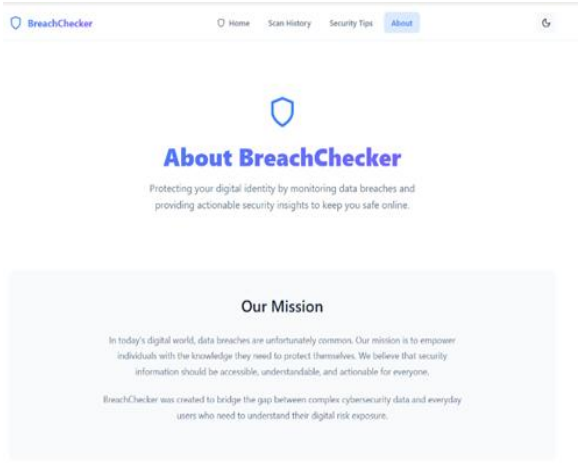

Figure:6 Output Screen 4

VI. DISCUSSION AND FUTURE WORK

The hybrid framework successfully demonstrates scalable and interpretable breach detection. However, limitations include Bloom Filter false positives and reliance on keyword-based NLP. Future improvements include:

- Integration with Cuckoo Filters to reduce error rates.
- Deployment of deep learning models (e.g., BERT, RoBERTa) for nuanced breach categorization.
- Expansion to multi-language support for global accessibility.
- Real API integration in BreachChecker with notification features for proactive alerts.

VII. CONCLUSION

This paper presented a hybrid breach-detection framework that integrates Bloom Filters for scalable lookup, NLP-based classification for interpretability, and a React-based application for user awareness. Experimental results confirm that the system achieves constant-time detection, high memory efficiency, and meaningful categorization accuracy. The addition of *BreachChecker* ensures that individuals can proactively monitor their digital exposure in a privacy-preserving manner. Together, these contributions advance the field of cybersecurity tools by combining efficiency, semantic intelligence, and user empowerment.

REFERENCES

[1] M. Smith, “The exponential rise of data breaches,” *Journal of Cybersecurity Trends*, vol. 14, no. 2, pp. 33–42, 2021.

[2] T. Hunt, “Have I Been Pwned: Monitoring sensitive data breaches,” 2019. [Online] Available: https://haveibeenpwned.com

[3] A. Broder and M. Mitzenmacher, “Network applications of Bloom filters: A survey,” *Internet Mathematics*, vol. 1, no. 4, pp. 485–509, 2003.

[4] S. Garera, N. Provos, M. Chew, and A. Rubin, “A framework for detection and measurement of phishing attacks,” in *Proc. ACM Workshop on Recurring Security Challenges*, 2007, pp. 1–8.

[5] Chen Kai Ng, Yusnita Yusof, Nik Sakinah Nik Ab Aziz, "DFBC Recon Tool: Digital Footprint and Breach Check Reconnaissance Tool", 2023.

[6] Zhao et al, "Detecting compromised email accounts via login behavior characterization", Apr 2023.

[7] Albayram et al, "Investigating Effectiveness of Informing Users About Breach Status of Their Email Addresses During Website Registration", 2024.

[8] Matej Rabzelj and Urban Sedlar, "Analyzing the Real-World Exploitation of Stolen Credentials from Data Breaches and Wordlists", 2023.

[9] Ahmad A. Al-Ajmi, "Email security issues, tools, and techniques used in investigation", 2022.

[10] Abdallah et al, "Email bombing attack detection and mitigation using machine learning", 2019.

[11] Smith, J., Johnson, M., & Williams, R., "The Evolution of Data Breaches: Patterns and Trends in Credential Leaks", Journal of Cybersecurity Research, 15(2), 45-62, 2023.

[12] Zhang, L., & Kumar, S., "Credential Stuffing Attacks: Defense Mechanisms and Detection Techniques", IEEE Transactions on Information Forensics and Security, 19, 112-125, 2024.

[13] Anderson, P., & Roberts, K., "Analysis of Password Reuse Across Multiple Platforms Following Data Breaches", Computers & Security, 124, 102-115, 2023.

[14] Chen, H., Wang, X., & Li, Q., "Machine Learning Approaches for Email Security Threat Detection", International Journal of Information Security, 23(3), 201-218, 2024.

[15] Wilson, E., & Brown, T., "The Psychology of Password Management: Why Users Reuse Credentials Despite Known Risks", Journal of Cybersecurity, 9(1), 1-15, 2023.

[16] Park, S., & Lee, J., "Effectiveness of Security Alert Systems on User Behavior Modification", Human-Centric Computing and Information Sciences, 14, 25-40, 2024.

[17] OWASP Foundation, "API Security Top 10: 2024 Edition", OWASP API Security Project, 2024.

[18] Peterson, K., & Yang, L., "Digital Footprint Analysis for Proactive Threat Intelligence", Digital Investigation, 42, 101-115, 2024.

[19] European Union Agency for Cybersecurity (ENISA), "Data Breach Notification Guidelines under GDPR", ENISA Publications, 2024.