International Journal of Engineering and Techniques - Volume 12 Issue 2, March-April - 2026

RESEARCH ARTICLE OPEN ACCESS

A Data Privacy Enforcement Tool for Personally Identifiable Information

Dr. R. Kavitha[1], U. Hari Pratha[2], R. Dhanya[3], M. Arthy[4], S. Lathika[5]

[1] *Professor, Department of Computer Science and Engineering, Parisutham Institute of Technology and Science, Thanjavur, Tamil Nadu – 613006, India*
Emai: pits.hod.cse@gmail.com

[2]*UG Student, Department of Computer Science and Engineering, Parisutham Institute of Technology and Science, Thanjavur, Tamil Nadu – 613006, India*
Email: haripratha56@gmail.com

[3] *UG Student, Department of Computer Science and Engineering, Parisutham Institute of Technology and Science, Thanjavur, Tamil Nadu 613006, India*
Email: dhanyaravi0021@gmail.com

[4]*UG Student, Department of Computer Science and Engineering, Parisutham Institute of Technology and Science, Thanjavur, Tamil Nadu – 613006, India*
Email: gmathiyazhagan49@gmail.com

[5]*UG Student, Department of Computer Science and Engineering, Parisutham Institute of Technology and Science, Thanjavur, Tamil Nadu – 613006, India*
Email: lathikaeniya2005@gmail.com

Abstract:

The rapid growth of digital systems has led organizations to store large volumes of sensitive information, including Personally Identifiable Information. Improper handling of such data can result in privacy breaches and significant security risks. This paper presents a Data Privacy Enforcement Tool that focuses on protecting sensitive data using data masking techniques. The proposed system identifies PII fields within datasets and applies masking methods such as partial masking, character replacement and format-preserving masking to conceal confidential information while retaining its usability.

The system is implemented using a Django-based backend and a React-based web interface, allowing users to upload datasets, configure masking rules and export the protected data in formats such as CSV, JSON and Excel. By restricting the solution to data masking, the tool ensures simplicity, efficiency and ease of integration into existing data workflows.

Experimental results demonstrate that the proposed approach effectively protects sensitive information while maintaining the structural integrity and usability of the data for analysis and testing purposes.

***Keywords*— Data Privacy, Data Masking, Personally Identifiable Information, Data Security**

I. INTRODUCTION:

The rapid expansion of digital infrastructures and data-driven applications has resulted in organizations collecting and storing massive volumes of data in centralized database systems. A significant portion of this data includes Personally Identifiable Information such as names, email addresses, phone numbers, identification numbers and financial records. Improper exposure or misuse of such sensitive information can lead to serious consequences, including privacy violations, identity theft, financial fraud and regulatory non-compliance. Therefore, protecting sensitive information has become a critical concern for organizations managing large-scale data repositories.

Traditional security mechanisms such as encryption, authentication and role-based access control are widely used to safeguard data during storage and transmission. However,

these methods do not sufficiently address privacy risks when real datasets are accessed in environments such as development, testing, analytics, or data sharing. In such cases, sensitive information may still be exposed to internal users or external systems.

To overcome this limitation, data masking has emerged as an effective privacy-preserving technique. Data masking transforms sensitive data into a protected format by applying methods such as partial masking, character replacement and format-preserving masking, ensuring that the original data is concealed while maintaining its structure and usability. This enables organizations to safely use datasets without revealing confidential information.

The proposed system focuses on implementing data masking techniques with automated detection of sensitive fields. It utilizes a Django-based backend for processing and a React-based web interface for user interaction, allowing users to configure masking rules, securely transform data, and export protected datasets for various use cases.

II. RELATED WORK:

Benet Manzanares and Sanchez [1] proposed an approach to improve text data masking using pre-trained large language models. Their method focuses on identifying sensitive information such as names, locations and personal identifiers within unstructured textual data by leveraging contextual understanding. Unlike traditional rule-based techniques, the model enhances accuracy in detecting sensitive fields by analyzing the semantic meaning of the text. Once identified, masking techniques such as partial masking and character replacement are applied to hide confidential information while preserving readability. This ensures that the masked data remains useful for analytical and processing tasks. The approach also minimizes common issues such as over-masking and under-masking, improving overall data quality. Furthermore, the method demonstrates adaptability across different types of text datasets. Overall, the work highlights the effectiveness of intelligent data masking in modern text-based applications.

Diaz Asper et al. [2] introduced a privacy-preserving approach focused on masking sensitive information in speech data. Their work emphasizes protecting speaker identity by transforming identifiable voice characteristics while maintaining the original linguistic content. Although primarily applied to audio data, the concept aligns with data masking principles by concealing sensitive attributes without affecting data usability. The method ensures that personal identity cannot be inferred while still allowing meaningful analysis of the data. This approach highlights the importance of preserving data structure and utility during the masking process. It also demonstrates how masking techniques can be extended beyond structured data to multimedia datasets. By maintaining a balance between privacy protection and data usability, the system supports secure data sharing in sensitive environments. Overall, the work reinforces the role of data masking in safeguarding personal information across diverse data formats.

Ali and Ouda [3] proposed a data masking framework specifically designed for healthcare business intelligence systems to protect sensitive medical information. Their approach focuses on identifying critical data fields such as patient records, personal identifiers, financial details, and applying suitable masking techniques to ensure confidentiality. The framework utilizes methods such as substitution, shuffling and partial masking to transform sensitive data while preserving its structure and analytical value. This enables organizations to perform data analysis and reporting without exposing confidential information. Unlike traditional security mechanisms, the system emphasizes usability by maintaining data consistency across masked datasets. The framework also supports integration with existing healthcare data systems, making it practical for real-world deployment. Furthermore, it highlights the importance of balancing data privacy with data usability in sensitive domains. Overall, the work demonstrates an effective application of data masking in healthcare analytics environments.

Cunha et al. [4] presented a comprehensive survey of privacy-preserving techniques for heterogeneous data sources, highlighting the role of data masking in protecting sensitive information across different data types. Their study analyzes how masking techniques can be applied to structured, semi-structured and unstructured datasets to ensure confidentiality while maintaining usability. The work discusses various masking methods emphasizing their effectiveness in real-world scenarios. It also compares masking with other privacy techniques, showing that masking offers a practical balance between security and data utility. The survey further identifies challenges such as handling diverse data formats and maintaining consistency across large-scale systems. Additionally, it highlights the need for adaptable masking solutions that can work across multiple domains. Overall, the study provides valuable insights into the importance of data masking in modern data privacy frameworks.

Majeed and Lee [5] presented a detailed survey of anonymization techniques for privacy-preserving data publishing, with strong relevance to data masking approaches. Their work examines how sensitive information can be transformed using techniques to prevent identity disclosure. These methods align closely with data masking by ensuring that original data values are concealed while preserving overall data utility. The study highlights the importance of maintaining a balance between privacy protection and analytical usefulness in published datasets. It also discusses challenges such as data distortion and loss of accuracy when masking is not properly applied. Furthermore, the authors emphasize the need for adaptable masking strategies based on the type of data and application requirements. The survey provides a strong foundation for

understanding how masking techniques can be effectively used in real-world data publishing scenarios. Overall, the work reinforces the significance of data masking in protecting sensitive information.

Keswani et al. [6] proposed a secure framework for protecting sensitive data in encrypted databases by transforming critical information before access. Their approach emphasizes applying controlled data transformation techniques to sensitive fields so that original values are not directly exposed to users or systems. This concept aligns with data masking, where confidential data is modified while preserving its structure and usability. The framework ensures that even when data is accessed or processed, sensitive attributes remain protected through transformation mechanisms. It also supports secure data handling in environments where multiple users interact with the database. Additionally, the system maintains consistency across transformed datasets, which is essential for analysis and reporting. The study highlights the importance of integrating data protection techniques with secure storage systems. Overall, the work demonstrates how masking-oriented transformations can enhance data privacy in database environments.

Rao et al. [7] presented a survey on privacy preservation techniques in big data analytics, highlighting the importance of data masking in handling large-scale sensitive datasets. Their study discusses how masking techniques can be applied to transform critical data fields before analysis to prevent exposure of confidential information. Methods such as data perturbation and partial masking are emphasized as effective approaches for protecting Personally Identifiable Information. The work also addresses challenges in maintaining data utility while applying masking in high-volume and high-velocity data environments. It highlights the need for scalable masking solutions that can efficiently process large datasets without affecting performance. Additionally, the study stresses the importance of integrating masking techniques into data processing pipelines. Overall, the work demonstrates the role of data masking in ensuring privacy in big data systems.

Rapsik and Kvet [8] explored techniques for improving cybersecurity through data transformation methods, with emphasis on protecting sensitive information using data masking approaches. Their work highlights how masking techniques can be applied to conceal confidential data and reduce the risk of data breaches and unauthorized access. Methods such as character replacement, partial masking, and format-preserving masking are discussed as effective ways to secure sensitive attributes. The study emphasizes that masking helps maintain the usability of data while preventing direct exposure of original values. It also discusses the importance of applying masking in systems where data is shared across multiple platforms or users. Furthermore, the approach supports secure data handling without compromising system functionality. The work demonstrates that data masking plays a key role in strengthening overall cybersecurity frameworks.

Sweeney [9] introduced a privacy protection model that focuses on preventing the identification of individuals within a dataset by transforming sensitive attributes. This concept can be closely related to data masking, where original values are modified to ensure that individual records cannot be uniquely distinguished. The approach emphasizes grouping and transforming data in a way that preserves overall dataset structure while hiding confidential information. Such transformation techniques support masking objectives by reducing the risk of identity disclosure. The model highlights the importance of balancing privacy protection with data usability in analytical processes. It also demonstrates how structured data can be safely shared after applying appropriate transformations. Furthermore, the approach provides a foundation for developing advanced masking techniques. Overall, the work contributes to the understanding of privacy-preserving transformations in data systems.

Dwork [10] proposed a strong privacy-preserving model that protects sensitive information by applying controlled data transformations before analysis. This concept aligns with data masking principles, where original data values are modified to prevent direct identification of individuals. The approach introduces techniques that ensure privacy protection while still allowing useful insights to be derived from the data. By transforming sensitive attributes, the system minimizes the risk of exposing confidential information during data processing. The method also supports maintaining the statistical properties of the dataset, which is essential for accurate analysis. It highlights the importance of applying systematic data transformation techniques in privacy-sensitive environments. Additionally, the approach provides a theoretical foundation for modern data masking strategies. Overall, the work strengthens the role of transformation-based methods in ensuring data privacy.

III. PROPOSED METHODOLOGY

The proposed system is designed to enhance the protection of sensitive information stored in organizational databases by automatically detecting and securing PII. The system integrates database connectivity, automated PII detection, configurable data masking mechanisms and secure data export within a unified framework. By combining rule-based detection techniques with multiple data masking strategies such as partial masking, character replacement, data perturbation, and etc… the system ensures that sensitive information is effectively concealed while maintaining data usability. This enables the dataset to be safely used for development, testing and analytical purposes without exposing confidential information. The overall workflow of the system consists of several stages, including data acquisition, PII detection, masking configuration, data transformation using masking techniques and secure export of protected datasets.

a. System Overview

The architecture consists of key modules including Data Acquisition, PII Detection, Masking Configuration, Data Transformation and Secure Data Export. Each module performs a specific function, contributing to the overall effectiveness of the system. The PII Detection module identifies sensitive fields using rule-based techniques, while the Masking module applies appropriate methods to data. The system ensures that protected datasets retain their structure and can be safely used for development, testing and analytical purposes.

b. Authentication (RBAC)

The Authentication module is responsible for ensuring secure access to the system by validating user identity and controlling permissions based on predefined roles. The system implements Role-Based Access Control (RBAC), where users are assigned specific roles such as administrator or standard user, each with different levels of access. This ensures that only authorized users can upload datasets, configure masking rules or export protected data. The module manages login credentials and enforces secure authentication mechanisms to prevent unauthorized access. It also restricts sensitive operations based on user privileges, enhancing overall system security. By controlling access to critical functionalities, this module ensures that data masking processes are performed only by authorized personnel. The Authentication module plays a vital role in maintaining data security and system integrity.

c. Project Creation

The Project Creation module allows users to organize and manage datasets by creating separate projects within the system. Each project acts as an independent workspace where users can upload data, apply masking configurations, and perform data transformations. This modular approach helps in handling multiple datasets efficiently without mixing configurations or results. Users can define project-specific settings, including naming, data source selection, and masking preferences. The module ensures that all operations such as PII detection and data masking are performed within the selected project context. It also helps in maintaining proper data organization and traceability of processed datasets. By providing a structured environment, this module improves usability and workflow management. Overall, the Project Creation module enhances the flexibility and scalability of the system.

d. Database Connection

The system focuses on establishing a secure connection with the target database and collecting the necessary data for further processing. The system enables users to configure database connection parameters through the application interface. Once the connection is successfully established, the system retrieves database schema information and sample records required for analysis.

During this stage, several important operations are performed, including:

- Database Configuration: The user provides database details such as database type, host address, port number, database name and authentication credentials through the system interface.
- Connection Validation: The django verifies the provided credentials and establishes a secure connection with the database server to ensure that the database is accessible.
- Schema Retrieval: After a successful connection, the system fetches metadata information such as available tables, column names and data types present in the database.
- Sample Data Collection: The system retrieves a small set of sample records from the selected tables to analyze the structure and content of stored data.
- Dataset Preparation: The collected metadata and sample records are organized and prepared as the input dataset for the next stage of the system, which performs automated PII detection.

e. PII Detection

The system performs automated detection by analyzing the structure of database tables, column names and sample data patterns. A rule-based detection approach is used to recognize sensitive attributes that may contain personal information. The detection module evaluates multiple characteristics of the data before classifying a column as sensitive. The following operations are performed during the PII detection process:

- Column Name Analysis: The system examines column names to identify keywords commonly associated with sensitive information, such as *name, email, phone, address, password* and *ID number*.
- Pattern Recognition: The system analyzes sample data values to detect specific patterns, such as email formats, phone number structures or identification numbers.
- Rule-Based Matching: Predefined detection rules are applied to compare column attributes with known PII indicators stored in the system.
- Confidence Evaluation**:** Each detected field is assigned a confidence level based on how strongly the data matches predefined PII patterns.
- Sensitive Field Identification**:** Columns that satisfy the detection criteria are marked as PII fields and are forwarded to the privacy protection configuration stage.

f. Data Masking Module

The Data Masking module is responsible for protecting sensitive information by transforming identified PII fields into a secure and non-sensitive format. Based on the detection results, the system applies configurable masking techniques such as partial masking, character replacement, substitution and many more. This ensures that the original

data is concealed while maintaining its structure and usability for analysis and testing. Users can customize masking rules according to the type and sensitivity of the data, providing flexibility in implementation. The module processes the dataset efficiently and generates a masked version without altering the overall data format. It also ensures consistency across records, which is essential for maintaining data integrity. By preventing direct exposure of confidential information, this module plays a key role in enforcing data privacy.

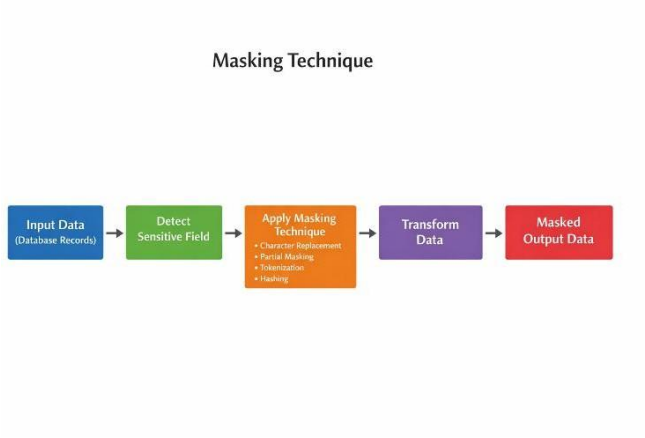

Fig 1. Data Masking Techniques

1. Input Data

This stage represents the original dataset obtained from the connected database. The system retrieves records containing various attributes such as names, email addresses, phone numbers, and identification numbers. These records act as the input data for the privacy protection process.

2. Detect Sensitive Field

The system analyzes the dataset to identify fields that contain Personally Identifiable Information. Detection is performed using rule-based techniques that examine column names, data patterns and predefined keywords. Identified sensitive fields are then marked for protection.

3. Apply Masking Technique

Once sensitive fields are detected, appropriate masking techniques are selected and applied. Methods such as character replacement, partial masking, tokenization and hashing etc… are used to hide or transform sensitive values. This ensures that confidential information is protected while maintaining the usability of the dataset.

4. Transform Data

During this stage, the selected masking technique is applied to each sensitive value in the dataset. The system processes records and replaces original sensitive values with their masked or transformed versions. The structure and format of the dataset are preserved to maintain data consistency.

5. Masked Output Data

The final stage produces the protected dataset containing masked values instead of the original sensitive information. This masked data can be safely used for development, testing, analytics or sharing purposes without exposing confidential data.

The Techniques applied here are:

1. Partial Masking- Partial masking is implemented using string slicing and replacement logic, where selected portions of the data are retained while the remaining characters are replaced with a masking symbol.
   It follows a rule-based approach such as retain first n and last m characters, mask the rest.
2. Character Replacement- Character replacement uses a fixed pattern substitution algorithm, where each character (or selected positions) is replaced with a constant symbol like * or X. This is typically implemented using iterative traversal or regex-based replacement functions.
3. Substitution- Substitution is implemented using a mapping or lookup table mechanism, where original values are replaced with pre-generated or randomly generated dummy values. It may also use random data generators with format constraints to maintain realistic outputs.
4. Data Perturbation - Data perturbation is implemented using a noise addition algorithm, where random values are added or subtracted within a defined range. Mathematically, it follows new_value = original_value ± random_noise, ensuring statistical consistency.
5. Date Masking- Date masking uses a date transformation algorithm, where specific components (day/month/year) are modified or generalized.
   It is implemented using date parsing functions and controlled shifting or truncation logic.
6. Shuffling- Shuffling is performed using a random permutation algorithm, commonly the Fisher–Yates shuffle, to rearrange values within a dataset column.
   This breaks direct associations while preserving the overall data distribution.
7. Tokenization-Tokenization is implemented using a random token generation algorithm, where sensitive data is replaced with unique identifiers (tokens).
   A secure mapping table (token vault) is maintained to store the relationship between original data and generated tokens.

g. Output Generation

The Output Generation module is responsible for producing and exporting the masked dataset after the application of data masking techniques. Once the transformation process is completed, the system generates a secure version of the dataset in which all sensitive fields are properly masked. The module supports multiple output formats such as CSV, JSON and Excel, allowing flexibility for different use cases.

It ensures that the structure, schema and data consistency are preserved during the export process. Users can download the masked dataset or use it directly. The module also verifies that no original sensitive data is exposed in the output. By providing a reliable and secure export mechanism, this module ensures safe data sharing and usability. Overall, the Output Generation module plays a crucial role in delivering protected datasets for practical applications. module ensures safe data sharing and usability. Overall, the Output Generation module plays a crucial role in delivering protected datasets for practical applications.

IV. CONCLUSION

This work presents a data masking–based privacy enforcement framework for securing PII in structured datasets. The system integrates rule-based PII detection with configurable masking algorithms to transform sensitive attributes while preserving schema consistency and data utility. Techniques such as partial masking, character replacement, substitution, and format-preserving masking are systematically applied to ensure that original values are irreversibly concealed without affecting downstream usability. The proposed architecture, built using a Django backend and React interface, enables efficient data ingestion, rule configuration, and transformation workflows within a unified environment. The masking process is designed to maintain referential integrity and structural constraints, allowing the masked datasets to be seamlessly utilized in development, testing, and analytical pipelines.

Experimental evaluation indicates that the system achieves effective privacy protection with minimal impact on data usability. By emphasizing transformation-based security over traditional access control mechanisms, the proposed approach demonstrates a scalable and practical solution for privacy preservation. Overall, this work highlights data masking as a robust technique for enforcing data privacy in modern data management systems.

ACKNOWLEDGMENT

The authors would like to express their sincere gratitude to their project guide and faculty members of the Department of Computer Science and Engineering for their continuous support, valuable guidance, and encouragement throughout this research work.

The authors also extend their thanks to their external guide, Mr. T. Siva, from Cybersecurity Nerds Lab, for providing valuable insights, technical guidance and necessary resources that contributed to the successful completion of this work.

Additionally, the authors acknowledge the use of publicly available datasets and tools that supported the development and evaluation of the proposed system.

REFERENCES

[1] B. Manzanares and D. Sanchez, “Improving Text Anonymization through Pre-trained Large Language Models,” *Journal of Information Security and Applications*, vol. 65, pp. 1–12, 2023.

[2] C. Diaz-Asper, M. Todisco, and N. Evans, “Speaker Anonymization for Privacy Protection in Clinical Speech Data,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 195–205, 2022.

[3] O. Ali and A. Ouda, “Data Masking Framework for Healthcare Business Intelligence Systems,” *International Journal of Information Management*, vol. 52, pp. 102–110, 2020.

[4] M. Cunha, J. Mendes, and P. Silva, “Privacy-Preserving Techniques for Heterogeneous Data Sources: A Comprehensive Survey,” *ACM Computing Surveys*, vol. 55, no. 3, pp. 1–36, 2023.

[5] A. Majeed and S. Lee, “Anonymization Techniques for Privacy Preserving Data Publishing: A Survey,” *IEEE Access*, vol. 9, pp. 27863–27878, 2021.

[6] M. Keswani, A. Kaul, S. Braghin, N. Hilohan, and S. Antonatos, “Secure k-Anonymization Framework for Encrypted Databases,” *Proceedings of the IEEE International Conference on Data Engineering Workshops*, pp. 89–96, 2021.

[7] P. R. M. Rao, S. Murali Krishna, and A. P. Siva Kumar, “Privacy Preservation Techniques in Big Data Analytics: A Survey,” *Journal of Big Data*, vol. 6, no. 1, pp. 1–23, 2019.

[8] J. Rapsik and M. Kvet, “Improving Cybersecurity through Anonymization Techniques,” *International Journal of Computer Science and Network Security*, vol. 21, no. 7, pp. 45–52, 2021.

[9] L. Sweeney, “k-Anonymity: A Model for Protecting Privacy,” *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 5, pp. 557–570, 2002.

[10] C. Dwork, “Differential Privacy,” *Proceedings of the International Colloquium on Automata, Languages and Programming*, Springer, pp. 1–12, 20