



DEPARTMENT OF INFORMATION
TECHNOLIGY

Jayawikrama K.D.S.P. it21170720

Homomorphic Encryption for Data Processing

RP24_25J_003

Project Proposal Report

Sulaksha Punsara Jayawickrama – IT21170720

B.Sc. (Hons) Degree in Information Technology specialized in
Cyber Security


Department of Information Technology

Sri Lanka Institute of Information Technology

June 2024

Declaration

I declare that this is my own work, and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Group member name	Student ID	Signature
Jayawickrama K.D.S.P	IT21170720	

The above candidate is carrying out research for the undergraduate Dissertation under supervision of the undersigned.

.....

Signature of the supervisor

(Mr. Amila Senerathna)

.....

Date

.....

Signature of Co-supervisor

(Ms. Suranjini Silva)

.....

Date

Acknowledgement

I extend my sincere gratitude to my supervisor, Mr. Amila Senerathna, and co-supervisor, Ms. Suranjini Silva, for their invaluable guidance and support throughout this research study. I'm thankful to industry experts for sharing their expertise which helped me get a certain domain knowledge. Special thanks to my team members for their contributions, and to those who aided me willingly. Lastly, my heartfelt appreciation to my family for their constant love, assistance, and encouragement.

Abstract

This research focuses on the integration of advanced data classification and encryption methodologies within a Data Loss Prevention (DLP) tool. Machine learning (ML) algorithms are employed to classify data as sensitive or non-sensitive with high accuracy. Upon detecting sensitive information, the system deploys homomorphic encryption to secure the data, ensuring privacy while enabling secure operations on encrypted content. [1] Simultaneously, emails containing sensitive data are blocked from transmission and securely stored in the tool for administrative review. This approach fortifies organizational security by preventing unauthorized disclosure and maintaining compliance with data protection standards.

Keywords : Data Loss Prevention, Information Security, Digital Forensics, Machine learning, Data classification, Personnel identifiable information

Table of Contents

Contents

Declaration.....	3
Acknowledgement	4
Abstract	5
Table of Contents	6
List of Abbreviations.....	8
1. Introduction.....	9
2. Background and literature survey	10
3. Research Gap	11
4. Research Problem.....	12
5. Objectives	13
5.1. Main Objective	13
5.2. Sub Objectives.....	15
6. Methodology.....	18
6.1. System Architecture	18
6.2. Software solution	21
6.2.1. Requirements gathering and Analysis	22
6.2.2. Feasibility study	23
Dataset Preparation	23
6.2.3. Implementation	23
6.2.4. Testing.....	24
1.1. Tools and Techniques	24
1.1. Functional Requirements	25
1.1. System Requirements	26
1.2. User Requirements	26
Bibliography	27

List of Figures

Figure 1 - Machine Learning Workflow for Data Classification.

Figure 2 - Sensitive vs. Non-Sensitive Data Identification Process.

Figure 3 - Homomorphic Encryption Mechanism for Sensitive Data.

Figure 4 - Email Blocking Process for Sensitive Data Store

List of Abbreviations

Abbreviation	Description
DLP	Data Loss Prevention
SDLC	Software Development Life Cycle
PII	Personally Identifiable Information
ML	Machine Learning
HE	Homomorphic Encryption

1. Introduction

Data security is a critical aspect of modern organizational workflows, especially in the prevention of unauthorized access or leakage of sensitive information. In this context, data classification plays a pivotal role by categorizing information into sensitive and non-sensitive types. Leveraging machine learning (ML) techniques, this research ensures accurate and automated classification, minimizing human error and enhancing efficiency.

Once sensitive data is identified, the system employs homomorphic encryption to safeguard it. Homomorphic encryption allows computations to be performed on encrypted data without compromising its confidentiality, enabling secure processing while maintaining data privacy. Additionally, the tool proactively blocks emails containing sensitive data from being transmitted and securely stores the encrypted information in a centralized system for administrative review.
[2]

This integrated approach combines data classification, encryption, and secure storage, forming a robust framework that aligns with data protection regulations and strengthens the organization's overall data security posture.

2. Background and literature survey

In the evolving landscape of data security, the combination of advanced machine learning techniques and robust cryptographic methods has emerged as a cornerstone of effective Data Loss Prevention (DLP). This research focuses on integrating DeBERTa for data classification and homomorphic encryption for secure data handling, drawing on extensive prior work in these domains.

2.1 Machine Learning in Data Classification

Machine learning (ML) has revolutionized the process of data classification, enabling systems to identify patterns and make decisions with minimal human intervention. Traditional models like Support Vector Machines (SVM) and Naïve Bayes classifiers were widely used for text classification tasks but often struggled with complex contextual understanding. DeBERTa (Decoding-enhanced BERT with disentangled attention) has introduced a paradigm shift by incorporating enhanced self-attention mechanisms and disentangled representations. [3]

DeBERTa's architecture excels in understanding subtle relationships between words and phrases, making it highly effective for identifying sensitive information, such as personally identifiable information (PII) or proprietary content, within unstructured data. Studies have highlighted the model's ability to outperform predecessors like BERT and RoBERTa in text classification tasks by improving semantic understanding and reducing ambiguity in predictions. [4]

2.2 Homomorphic Encryption

Homomorphic encryption offers a groundbreaking approach to data privacy by enabling computations on encrypted data without requiring decryption. This ensures that sensitive information remains secure throughout its lifecycle, even during processing. Existing research emphasizes its potential to mitigate risks associated with insider threats and external breaches. Fully homomorphic encryption (FHE), while computationally intensive, has demonstrated practical applications in fields such as secure cloud computing, where sensitive operations must be performed on encrypted datasets. [5]

The integration of homomorphic encryption with DLP tools allows organizations to enforce stringent data protection policies. For instance, sensitive data identified by the classification system can be encrypted before being transmitted or stored, ensuring compliance with regulations such as GDPR, HIPAA, and CCPA.

2.3 Addressing Data Loss Challenges

Organizations face significant challenges in preventing data loss, including the accidental or malicious transmission of sensitive information via unapproved channels such as email. Combining DeBERTa’s classification capabilities with homomorphic encryption ensures a multi-layered approach to data security. Sensitive data is automatically classified, blocked from unauthorized transmission, and securely stored in encrypted form, enabling real-time protection and streamlined auditing processes.

By building on the foundational principles of machine learning and cryptography, this research proposes a novel system that enhances existing DLP frameworks. It bridges the gap between accurate data classification and secure handling, offering a holistic solution to modern data security challenges.

3. Research Gap

As discussed in the above literature review, a notable gap can be seen in the existing systems. Below mentioned are the research gaps found.

Application Reference	Data Classification	Homomorphic Encryption	Real-Time detection	DLP integration
Reference 01	✓	✗	✗	✗
Reference 02	✓	✗	✓	✗
Reference 03	✓	✗	✗	✗
Reference 04	✓	✗	✓	✗
Proposed system	✓	✓	✓	✓

Complementary to the existing research in this area, this solution offers a fresh approach with two different CNN models working on two different scales-fine and coarse-for the detection of steganalysis, which allows analysts to detect hidden threats with high accuracy and to precisely locate the exact position by narrowing down the detection to specific image patterns across multiple scales. After extracting these features at multiple scales, a feature fusion mechanism will be applied in order to combine the outputs of both CNN models, hence coming up with a more robust representation against possible steganographic content. [6]

The detection will be performed by the system in real-time, hence being suitable for installations in DLP solutions. It also allows for a module for ranking the quality of the detection, which enables the elimination of false positives and increases the reliability of the detection. Threat categorization will also be possible, making it easier for users to filter the steganalysis results according to file formats or data type.

After the detection and categorization are done, the results will be presented through an intuitive web interface, easily integrable with any existing DLP system. The interface will further support automated workflows with customized notifications, thus efficiently managing and executing threat responses. In general, this solution will contribute to enhancing data security by embedding advanced steganalysis in DLP tools, improving not only the detection accuracy but also operational efficiency. [7]

4. Research Problem

The rapid growth of digital data has amplified the need for robust security mechanisms to prevent unauthorized access and leakage of sensitive information. In Data Loss Prevention (DLP) tools, the challenge lies in securely handling sensitive data without hindering operational efficiency. This research addresses the problem:

How can homomorphic encryption be integrated with data classification to ensure secure processing of sensitive data in a DLP tool?

Homomorphic encryption offers a transformative approach to secure data processing, allowing computations on encrypted data without requiring decryption. However, its practical application in DLP systems poses several challenges:

1. **Efficient Data Classification:** To ensure effective encryption, the system must accurately classify data as sensitive or non-sensitive. Misclassification can lead to either exposing sensitive information or encrypting non-critical data, increasing computational overhead unnecessarily. Advanced machine learning models, such as DeBERTa, are essential for high-precision classification.
2. **Prioritization of Data for Encryption:** Encrypting large datasets using homomorphic encryption can be computationally expensive. To optimize performance, the DLP tool should prioritize sensitive, smaller data segments for encryption while applying alternative protection measures (e.g., tokenization or hashing) to larger files.
3. **Integration with Real-Time Systems:** A key requirement is the seamless integration of data classification and encryption processes within real-time workflows. This includes blocking sensitive data in outbound communications (e.g., emails) and ensuring encrypted data remains accessible for authorized operations while maintaining its confidentiality.
4. **Centralized Monitoring and Management:** Managing both encrypted and non-encrypted data requires a centralized system capable of monitoring data flows, enforcing security policies, and providing visibility into potential risks. Such a system must also support auditing and compliance reporting to meet regulatory standards like GDPR, HIPAA, or CCPA.
5. **Scalability and Performance:** Homomorphic encryption, while secure, has computational limitations. Implementing it in a way that supports large-scale enterprise environments without significant performance trade-offs is a critical challenge that must be addressed.
6. **Compliance and Privacy Alignment:** Ensuring the approach aligns with global data protection regulations is vital. The integration must safeguard privacy, prevent unauthorized disclosures, and provide mechanisms to audit data handling practices.

This research seeks to design and implement a framework that integrates advanced data classification techniques with homomorphic encryption in a DLP tool. By addressing the identified challenges, the system aims to enhance data security, protect sensitive information, and provide a scalable and efficient solution for modern organizations. [8]

5. Objectives

5.1. Main Objective

The primary objective of the "Homomorphic Encryption for Data Processing" component is to enhance data security by enabling secure computations on sensitive data without requiring decryption. This approach ensures that sensitive information remains confidential throughout its lifecycle, even during processing, while still allowing meaningful operations on encrypted data.

Achieving this objective involves multiple interdependent goals:

1. **Implementation of Robust Data Classification:** A critical step is to develop and implement a machine learning-driven classification system capable of accurately

distinguishing sensitive data from non-sensitive data. Leveraging advanced models such as DeBERTa, this system will focus on high precision and recall to minimize false positives and negatives. The classification system should be efficient and scalable to handle diverse and large datasets.

2. **Effective Management of Sensitive Data:** To address the computational challenges associated with homomorphic encryption, the system will prioritize and manage smaller, sensitive data sets for encryption. Larger files or less critical data may be processed using alternative security measures such as tokenization, hashing, or segmentation to optimize performance while maintaining adequate security.
3. **Seamless Integration with DLP Tools:** The integration of homomorphic encryption and data classification will enhance the functionality of the Data Loss Prevention (DLP) tool. This includes the ability to block the transmission of sensitive data through unauthorized channels such as emails, ensuring that the organization's data handling aligns with security policies.
4. **Enable Secure and Private Data Analysis:** One of the unique advantages of homomorphic encryption is its ability to perform computations on encrypted data. The objective includes enabling secure data analysis and reporting within the encrypted domain, providing insights while safeguarding data privacy.
5. **Ensure Compliance with Data Protection Regulations:** The system aims to comply with global data protection standards such as GDPR, HIPAA, and CCPA. By securing sensitive information and preventing unauthorized disclosures, the approach will align with regulatory requirements and strengthen the organization's trustworthiness.
6. **Centralized Monitoring and Auditing:** The objective also includes developing a centralized monitoring system for managing encrypted and non-encrypted data. This system will enforce security policies, provide visibility into data flows, and support compliance reporting.
7. **Scalability and Efficiency:** Lastly, the system must be designed for scalability and efficiency to support enterprise-level deployments. The goal is to achieve a balance between computational security and operational performance, ensuring practical usability across various organizational contexts.

By achieving these objectives, the research aims to establish a robust framework that enhances data security, ensures compliance, and supports advanced data analytics within the encrypted domain, ultimately strengthening the organization's overall data protection strategy.

5.2. Sub Objectives

To achieve the primary objective of enabling secure computations on sensitive data while maintaining privacy, the research focuses on the following key sub-objectives:

4.1 Implement Data Classification

Effective data classification forms the foundation of this research. The goal is to develop and deploy an advanced machine learning model, particularly utilizing the DeBERTa architecture, for classifying data as sensitive or non-sensitive. This sub-objective involves:

- **Training the Classification Model:** The first step is to train a robust machine learning model using large datasets with labeled sensitive and non-sensitive data. The model will utilize NLP techniques to process unstructured textual data and classify it based on predefined sensitivity categories. Given the complexity of modern data, the model must be fine-tuned for high precision, reducing false positives and negatives, which can compromise both security and efficiency.
- **Sensitivity Labeling and Granularity:** The classification system will incorporate multiple sensitivity levels, recognizing that not all sensitive data should be treated equally. For example, financial data may require higher levels of protection than internal communications. The model will dynamically adapt its sensitivity categorization, ensuring proper prioritization for encryption and secure handling.
- **Automation and Scalability:** To be effective across diverse use cases, the classification system should be scalable. This means automating the classification process, allowing the system to handle vast amounts of data without manual intervention. This scalability will be particularly important as the amount of data within organizations continues to grow exponentially.
- **Continuous Learning:** To address evolving threats and new types of sensitive data, the classification system will include mechanisms for continuous learning. It will regularly retrain with new data inputs, ensuring it adapts to emerging patterns and threats in the data landscape.

4.2 Ensure Efficient Data Processing

Once sensitive data has been identified, it must be processed securely and efficiently. This sub-objective focuses on optimizing the performance of the system while maintaining the highest level of security, particularly by implementing homomorphic encryption. The key components are:

- **Data Prioritization:** Not all data within an organization needs the same level of encryption. Homomorphic encryption is computationally expensive, so this sub-objective emphasizes efficient data processing by prioritizing smaller, more sensitive datasets for encryption. Larger files that do not contain sensitive information may be handled differently, using alternative security techniques such as hashing or tokenization to reduce the computational burden on the system.
- **Optimizing Encryption:** Homomorphic encryption allows computations on encrypted data, but it can introduce performance challenges. Efficient methods to minimize the processing time of encrypted data without compromising security are crucial. Techniques like batching operations and parallelizing computations can help mitigate the performance overhead.
- **Seamless Integration with Encryption Mechanisms:** The encryption process needs to work without disrupting the overall workflow. The tool will integrate encryption seamlessly into the data flow, ensuring that sensitive data is encrypted before any operations are performed, and maintaining its encrypted state throughout the processing stages. This includes encryption during storage, transit, and while undergoing analysis or computations.
- **Performance Trade-offs:** The system will need to balance security with efficiency. By optimizing the use of homomorphic encryption for sensitive data while employing other methods for non-sensitive data, the overall system can achieve a practical balance between security and performance. [9]

4.3 Integrate with DLP Tool

The integration of the secure processing pipeline into an existing DLP tool is critical for providing a comprehensive data protection solution. This sub-objective focuses on ensuring that sensitive data is properly handled, monitored, and blocked from unauthorized transmission while maintaining operational efficiency.

- **Secure Transmission and Blocking:** One of the core functionalities of a DLP tool is preventing sensitive data from being leaked through unapproved channels, such as email or external file transfers. Upon identifying sensitive data, the tool will block any attempts to send it through these channels, ensuring that no unencrypted sensitive data leaves the organization. Emails or files containing encrypted sensitive data should be securely stored within the DLP system.
- **Real-Time Data Monitoring:** The integration will include continuous monitoring of data flows across the organization, providing real-time alerts whenever sensitive data is detected in unapproved locations or communications. The system will offer a comprehensive view of data movement, providing administrators with the ability to take immediate action if a potential breach occurs.
- **Compliance and Reporting:** Integration with the DLP tool will ensure that data protection policies are enforced consistently across the organization. The tool will generate detailed reports for auditing and compliance purposes, making it easier to demonstrate adherence to global data protection standards such as GDPR, HIPAA, and CCPA.
- **User and Access Management:** To strengthen security, the tool will include robust user and access management features. Only authorized personnel will be able to access, decrypt, or interact with sensitive data, while the system will log all actions for audit trails. This integration will help enforce policies regarding data access and ensure that any handling of sensitive data is done in compliance with organizational security standards.

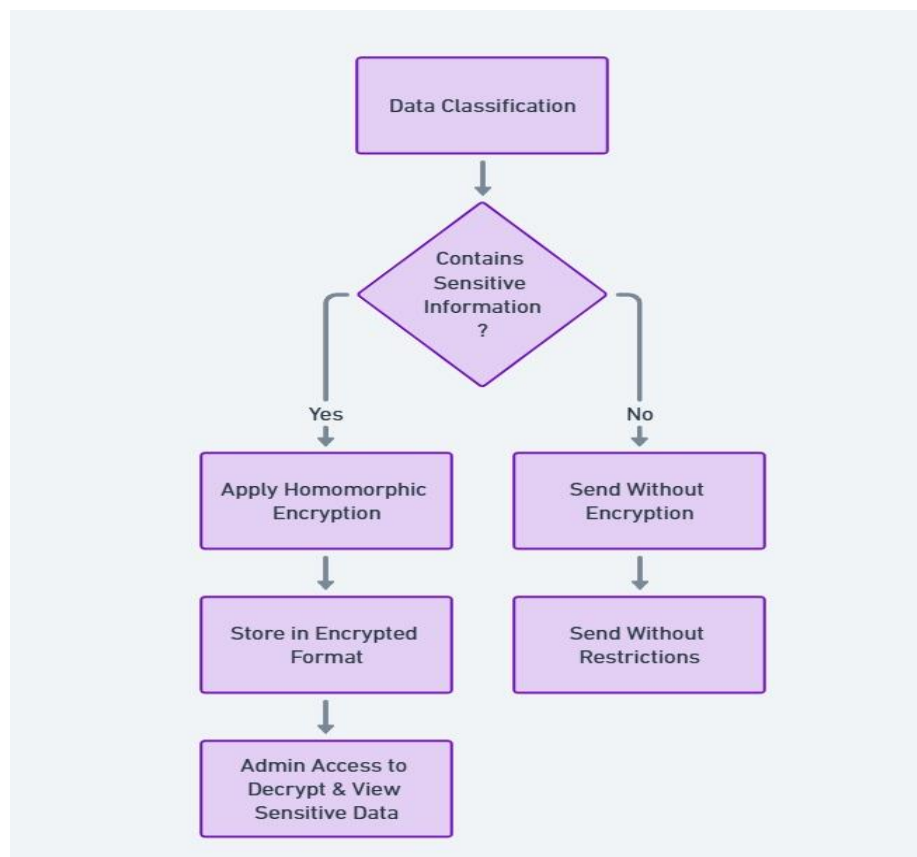
By addressing these sub-objectives—implementing data classification, ensuring efficient data processing, and integrating with the DLP tool—the research aims to create a comprehensive

solution that both protects sensitive information and allows organizations to meet their security and regulatory compliance requirements.

6. Methodology

6.1. System Architecture

The projects' main goal is to provide an overall solution based on data loss prevention which facilitates in mitigating sensitive data leakage or data leakage attacks in the organization.



The architecture of the proposed system focuses on integrating data classification, homomorphic encryption, and DLP functionalities to create a robust and secure data processing pipeline. This system architecture is designed to ensure that sensitive data is securely processed, classified, and

stored, preventing unauthorized disclosure while maintaining operational efficiency. The architecture can be broken down into the following key components:

5.1 Data Input Layer

The Data Input Layer is responsible for collecting and receiving data from various sources within the organization. This layer interacts with multiple systems such as email servers, file systems, cloud storage, and databases. It serves as the entry point for data into the system, where data is either structured (e.g., database records) or unstructured (e.g., emails, text files).

- **Data Collection:** This layer collects and aggregates data from multiple sources to be processed. It includes the tools that scan incoming and outgoing communications, as well as document and database records for potential sensitive information.
- **Real-Time Data Processing:** The system is designed to operate in real time, allowing for continuous monitoring of data flows and immediate classification and encryption actions.

5.2 Data Classification Layer

The Data Classification Layer is the core of the system where sensitive data is identified and categorized using advanced machine learning models such as DeBERTa. This layer ensures that data is classified into predefined categories, such as sensitive or non-sensitive, based on its content, context, and sensitivity.

- **Machine Learning Model Integration:** The DeBERTa model is utilized here to analyze the content of incoming data and classify it based on predefined sensitivity labels. The model will identify sensitive information such as personally identifiable information (PII), financial data, intellectual property, and confidential business data. [10]
- **Granularity of Classification:** Data can be classified at different levels of sensitivity, allowing the system to apply varying degrees of protection based on the nature of the data. For example, highly sensitive data might be encrypted with homomorphic encryption, while less critical data might use other protective measures.
- **Continuous Learning and Adaptation:** The system will have a feedback loop where newly labeled data will be used to improve the accuracy of the classification model over time, ensuring that the model stays up-to-date with emerging patterns and threats.

5.3 Encryption Layer

Once data is classified as sensitive, the next step is to secure the data using homomorphic encryption. This layer ensures that computations can be performed on encrypted data without needing to decrypt it, preserving the confidentiality and integrity of sensitive information.

- **Homomorphic Encryption Application:** Homomorphic encryption algorithms are applied to the sensitive data identified by the classification system. This encryption

allows the system to perform operations like searches, queries, and analysis on encrypted data, ensuring that even in case of a breach, the data remains secure.

- **Prioritization of Sensitive Data:** Sensitive data that needs to be encrypted will be prioritized over non-sensitive data. This ensures that the encryption process is computationally efficient, focusing on smaller, more critical data sets while applying less computationally intensive measures to larger files or non-sensitive data.
- **Alternative Protection Mechanisms:** For larger files or data that does not need the same level of encryption, the system will apply alternative security measures such as tokenization or hashing to maintain security without the performance overhead associated with homomorphic encryption.

5.4 DLP Integration Layer

The DLP Integration Layer ensures that the data is monitored and controlled based on security policies. It enforces rules regarding the transmission of sensitive data, ensuring compliance with organizational policies and regulatory requirements.

- **Sensitive Data Monitoring and Blocking:** Once data has been classified and encrypted, the DLP system monitors data movement, ensuring that sensitive data is not transmitted to unauthorized recipients. It intercepts and blocks any unauthorized attempts to send sensitive data through email, external file-sharing services, or other unapproved channels.
- **Encryption Handling:** If sensitive data needs to be sent, the system will block the transmission of unencrypted data and allow encrypted data to pass through securely. For outgoing emails or files containing sensitive data, the DLP system will store the encrypted data within a secure repository or handle it as per the organization's security policy.
- **Real-Time Alerts and Notifications:** The DLP system sends real-time alerts and notifications to administrators whenever a potential data breach is detected or an attempt to transmit sensitive data occurs. These alerts help in rapid incident response and investigation.

5.5 Monitoring and Auditing Layer

The Monitoring and Auditing Layer is responsible for overseeing the entire data security process, providing visibility into data flows, and ensuring compliance with data protection regulations. This layer plays a critical role in tracking, logging, and reporting on sensitive data handling.

- **Real-Time Monitoring:** The monitoring system continuously observes data movements, analyzing and recording every interaction with sensitive data. It keeps track of encrypted

and non-encrypted data transactions, ensuring that proper security measures are applied at every stage.

- **Audit Trails:** The system will generate detailed audit logs of all actions taken with sensitive data, including encryption, transmission attempts, and data access. These logs are crucial for compliance audits and forensic investigations in case of a security incident.
- **Compliance Reporting:** The system will provide reports aligned with global data protection regulations like GDPR, HIPAA, and CCPA. These reports will allow the organization to demonstrate its adherence to security policies and legal requirements regarding sensitive data.

5.7 Centralized Management and Control Dashboard

A centralized management dashboard will serve as the user interface for administrators to configure the system, monitor real-time activities, and generate reports. It will provide a comprehensive view of the entire data security pipeline, from classification to encryption and DLP enforcement.

- **Visualization:** The dashboard will visualize data flows, alerts, encryption statuses, and system health, providing administrators with actionable insights into the system's performance and security posture.
- **Policy Configuration:** Administrators can define, modify, and enforce security policies from the dashboard. This includes setting rules for classification, encryption, data blocking, and alert management.
- **Incident Response Tools:** In case of a security incident, the dashboard will provide tools for investigating and responding to threats in real-time. It will facilitate the investigation of potential breaches and the analysis of encrypted data when necessary.

6.2. Software solution

Development will be done iteratively; the DLP tool's Agile Software Development Lifecycle using SCRUM methodology will be followed for the development of the steganalysis component. This approach gives much emphasis to continuous integration and collaboration between developers and stakeholders for the efficiency and adaptability of the component.

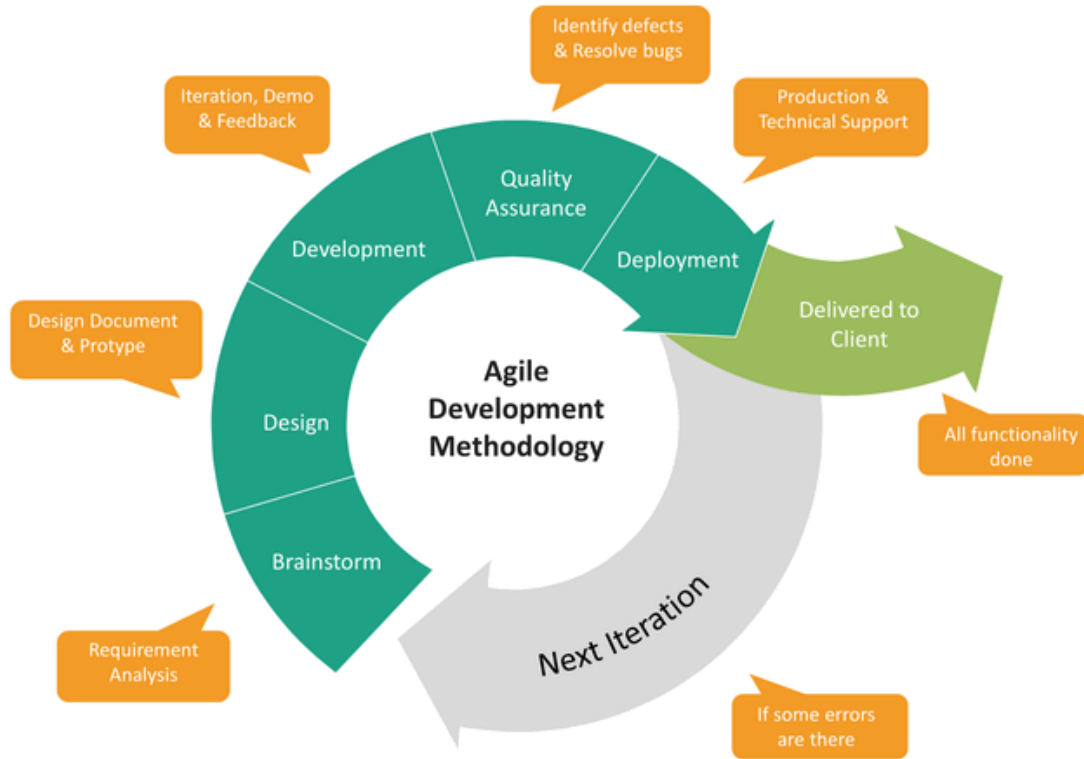


Figure 1-Agile methodology

6.2.1. Requirements gathering and Analysis

This stage is crucial to ensuring the integration of homomorphic encryption and data classification into the DLP tool meets its objectives. Key stakeholders, including cybersecurity experts, data engineers, and compliance officers, will need to be identified to gather requirements. Functional requirements for the data classification system include accurate identification of sensitive data, the ability to classify data based on its content, and integration with the DLP tool to block sensitive data from being transmitted. The non-functional requirements include the accuracy of the classification system, real-time processing, and ensuring scalability to handle large datasets efficiently.

Additionally, the homomorphic encryption requirements include the secure encryption of sensitive data, enabling computations on encrypted data, and minimizing computational overhead. Business and compliance requirements will focus on ensuring the protection of sensitive information, such as personal data and financial records, while adhering to industry standards and regulations like GDPR and HIPAA.

The datasets needed for training the data classification model (e.g., labeled data containing sensitive and non-sensitive information) and evaluating encryption methods will be identified. The feedback from cybersecurity experts will be gathered to align the design with security best practices. This structured process will provide a foundation for implementing a data classification system with homomorphic encryption within the DLP tool.

alignment with industry best practices. This structured process lays the ground for an effective and accurate steganalysis system that could be integrated within the DLP tool.

6.2.2. Feasibility study

A feasibility study will be conducted to assess both technical and economic factors in implementing the homomorphic encryption and data classification components. The development team will evaluate the computational resources required for performing data classification in real-time and applying homomorphic encryption. This involves assessing the performance of machine learning models used for classification and encryption, ensuring the system can efficiently handle large datasets without compromising speed. The study will also consider the cost-effectiveness of implementing homomorphic encryption and classify data, ensuring the solution fits within the project's budget.

Dataset Preparation

The dataset preparation will include collecting and labeling data to train the data classification system, with a focus on identifying sensitive and non-sensitive information. This dataset could include emails, documents, financial records, or other forms of communication. Additionally, data related to encryption methods will be prepared to ensure that sensitive data can be securely encrypted using homomorphic encryption. A critical part of the dataset preparation is the training of machine learning models to identify the characteristics of sensitive data for accurate classification and encryption.

6.2.3. Implementation

- **Model Training:** Develop and train a machine learning model (e.g., using DeBERTa) to classify data as sensitive or non-sensitive. The model will be trained on a labeled dataset that includes a variety of sensitive and non-sensitive data types.
- **Homomorphic Encryption Integration:** Implement the homomorphic encryption component, ensuring that sensitive data identified by the classification system is encrypted before transmission. The system will perform operations on encrypted data without the need to decrypt it, maintaining data privacy.
- **Real-Time Integration:** Integrate the trained model with the DLP tool to classify and encrypt data in real time. Sensitive data will be encrypted and blocked from unauthorized transmission, while non-sensitive data will be processed normally.

6.2.4. Testing

- **Unit Testing:** Test individual components of the data classification model, including the machine learning model and encryption algorithms, to ensure accuracy and efficiency.
- **Integration Testing:** Ensure the data classification system works seamlessly with the DLP tool, including accurate identification and encryption of sensitive data.
- **System Testing:** Test the overall system with real-time data to ensure the homomorphic encryption is applied correctly, sensitive data is classified accurately, and the DLP tool prevents unauthorized transmission.
- **User Acceptance Testing (UAT):** Security analysts and other end users will test the system to ensure it meets operational needs, such as accurate data classification, secure encryption, and blocking of unauthorized data transmissions.

1.1. Tools and Techniques

Tools

- **Python:** As the primary programming language for implementing machine learning models and system integration.
- **Anaconda:** For managing the machine learning environment and dependencies, especially during model development.
- **PyCharm:** For Python development, providing a robust Integrated Development Environment (IDE).
- **Flask/Django:** Tentatively for developing the web-based interface to visualize steganalysis detection results.
- **Heroku/AWS/GCP:** For cloud deployment, allowing for scalability and easy integration with a DLP system.
- **Jupyter Notebooks:** For training and experimenting with different machine learning models.
- **GitHub:** For version control and collaboration during the development process.

Techniques

- **Machine Learning (ML) for Data Classification:** Used to classify data as sensitive or non-sensitive based on predefined features.
- **Homomorphic Encryption:** Encrypts sensitive data, allowing secure computations without decryption.
- **Data Classification:** Labels data as sensitive or non-sensitive using ML models, ensuring sensitive data is encrypted and protected.
- **Real-Time Data Processing:** Classifies and encrypts data in real time, ensuring timely protection.
- **Model Evaluation:** Uses techniques like cross-validation and hyperparameter tuning to optimize classification accuracy.
- **Integration with DLP Tool:** Ensures encrypted data is securely handled and blocked from unauthorized transmission.
- **Testing and Validation:** Includes unit testing, integration testing, and user acceptance testing to ensure functionality and accuracy.

1.1. Functional Requirements

- **Data Classification:** The system must classify data into sensitive and non-sensitive categories based on predefined rules or machine learning models.
- **Homomorphic Encryption:** Sensitive data identified by the classification system must be encrypted using homomorphic encryption to ensure privacy while allowing computations.
- **Real-Time Processing:** The system must classify and encrypt data in real-time to ensure sensitive data is protected before transmission.
- **Data Blocking:** The system must prevent the transmission of sensitive data by blocking it from being sent via email or other communication channels.
- **Monitoring and Alerts:** The system must provide real-time alerts to administrators when sensitive data is detected and blocked.
- **Data Storage:** The system must store sensitive data securely in encrypted form for compliance and auditing purposes.

1.1. System Requirements

The system requirements outline the necessary software, hardware, and resources needed to ensure the proper functioning of the Homomorphic Encryption component within the DLP solution

- **Python:** The primary programming language used for implementing machine learning models and system logic.
- **Anaconda:** To manage virtual environments and handle dependencies for the machine learning frameworks.
- **PyCharm or VS Code:** For code development and testing in a well-supported Integrated Development Environment (IDE).
- **Flask or Django:** (Optional) Tentative frameworks for developing the web-based user interface for visualization and reporting.

1.2. User Requirements

- **Security Analysts:** The system should provide real-time detection and classification of sensitive data. Analysts need immediate alerts when sensitive data is identified, encrypted, or blocked. A user-friendly dashboard is required to monitor real-time data classifications, encryption status, and alerts. Detailed reports should offer insights into classified data, encryption activities, and actions taken by the system. Analysts should be able to configure alert notifications (e.g., email or SMS) for critical events and set thresholds for classification confidence levels to prioritize high-risk data.
- **Incident Response Teams:** The system must automatically block the transmission of sensitive data once it is classified and encrypted, preventing potential data leaks. Detailed logs of sensitive data handling and encryption actions should be available for incident response teams to analyze and review threats. The system must integrate with existing incident response workflows, offering customizable actions, such as quarantining sensitive files or notifying specific personnel, to streamline response processes.
- **System Administrators:** Administrators should have control over system settings, including detection parameters for data classification and encryption thresholds. They need the ability to manage user roles and permissions to ensure appropriate access control. The system should provide an intuitive interface for administrators to manage updates, patches, and troubleshooting processes, ensuring smooth and continuous operation.
- **Compliance Officers:** The system should generate tamper-proof audit logs that track the classification, encryption, and blocking of sensitive data, ensuring compliance with data protection regulations. Exportable reports should be available for compliance audits,

detailing detected incidents, encryption actions, and responses, to demonstrate adherence to legal and regulatory standards.

Bibliography

- [1] "wikipedia.org," [Online]. Available: https://en.wikipedia.org/wiki/Homomorphic_encryption.
- [2] C. Brook, 24 08 2024. [Online]. Available: <https://www.digitalguardian.com/blog/data-classification-examples-help-you-classify-your-sensitive-data>.
- [3] 18 12 2022. [Online]. Available: <https://ieeexplore.ieee.org/document/10016592>.
- [4] "DeBERTa," [Online]. Available: <https://paperswithcode.com/method/deberta>.
- [5] 12 2023. [Online]. Available: <https://www.internetsociety.org/resources/doc/2023/homomorphic-encryption/>.
- [6] M. Savva, "Data Loss Prevention Tools," p. 01, 2014.
- [7] S. Al-Fedaghi, "A Conceptual Foundation for Data Loss Prevention," 2011.
- [8] L. G.*, R. N.** and C. J.*, "Methodology for Data Loss Prevention Technology Evaluation for," Sheffield.
- [9] M. Vaidya, "Design and Analysis of Large Data Processing Techniques".
- [10] P. Gao, Z. Han and F. Wan, "Big Data Processing and Application Research," [Online]. Available: <https://ieeexplore.ieee.org/document/9425958>.