Problem Statement - 5

Cryptography Simulation with mbedTLS/OpenSSL Library Usage and User Interaction

Developing an interactive simulation tool using C code and mbedTLS/OpenSSL libraries to facilitate hands-on learning and experimentation with cryptographic techniques. The tool will allow users to explore encryption, decryption, key generation, and various cryptographic algorithms such as RSA, AES, DES, and SHA. The simulation will provide a user-friendly interface, educational content on security concepts, and interactive elements for users to experiment with different encryption parameters. The objective is to create a valuable educational resource for students and professionals in the field of cybersecurity, enhancing their understanding of how cryptography secures digital communication and data.

Unique Idea

Our solution involves creating a desktop application with a graphical user interface (GUI) for interactive cryptography simulation using C code and mbedTLS/OpenSSL libraries. Users will be able to explore cryptographic techniques such as encryption, decryption, key generation, and various algorithms like RSA, AES, DES, and SHA through a user-friendly interface. The application will allow users to input text or data, select encryption methods, view encrypted results, and decrypt back to the original content. Additionally, the GUI will provide educational content on security concepts, interactive elements for experimenting with encryption parameters, and a visually engaging platform for learning cryptography. This innovative approach aims to enhance user interaction, understanding, and experimentation with cryptographic concepts in a practical and accessible manner.

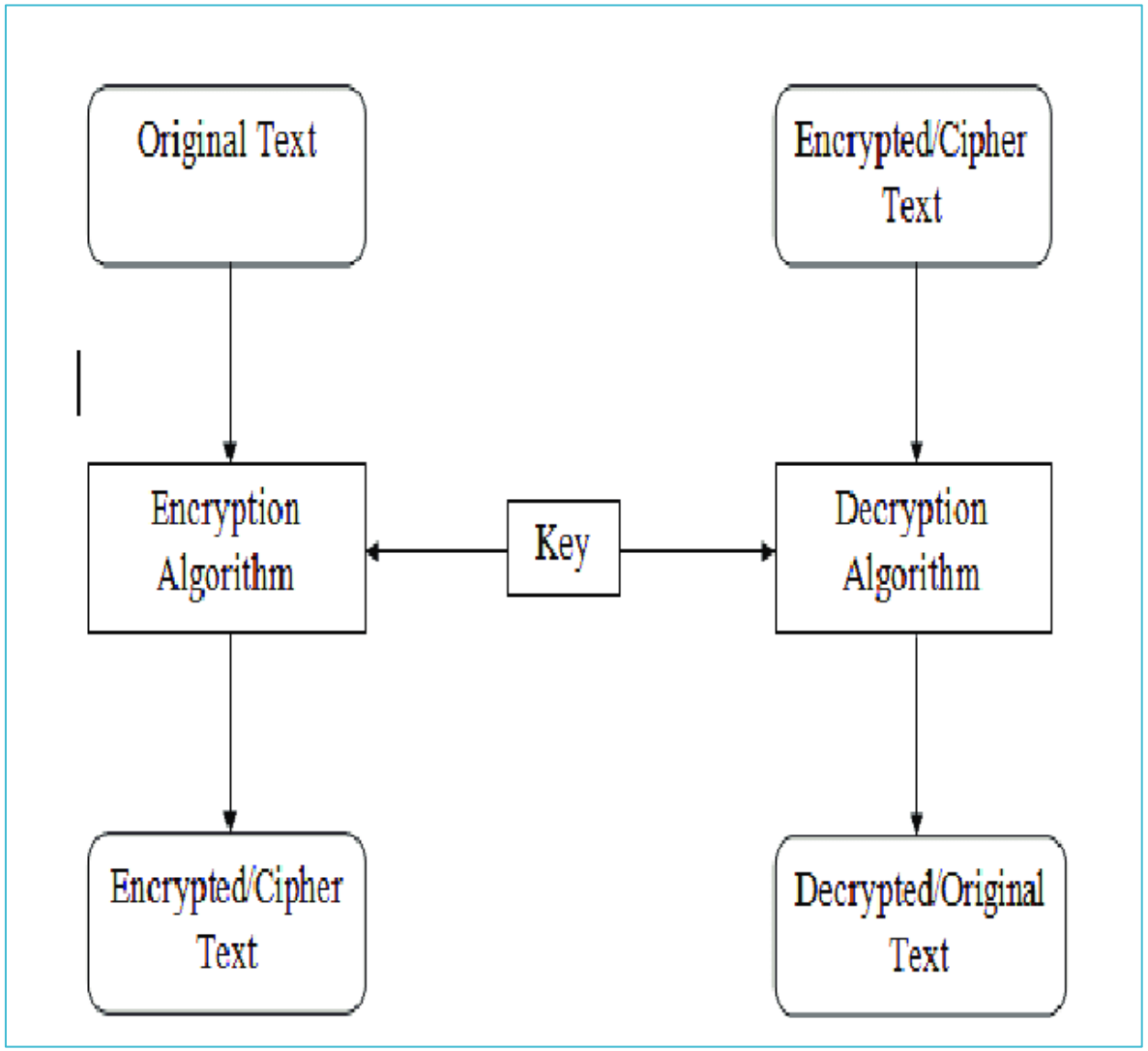
Features Offered for Encryption and Decryption

- Encryption Algorithms: Implement RSA, AES, DES, and SHA for data encryption.
- Decryption Functionality: Decrypt data back to its original form.
- Key Generation: Generate cryptographic keys for encryption and decryption.
- Input Data Handling: Allow user input for encryption.
- Encryption Method Selection: Choose encryption method (e.g., RSA, AES).
- Visual Representation: Visual feedback on encryption and decryption processes.
- User-Friendly Interface: Intuitive GUI for interaction.
- Educational Content: Include materials on encryption, decryption, and security concepts.
- Experimentation: Allow parameter and algorithm experimentation for practical cryptography experience.

Process Flow

- Key Generation: Cryptographic keys are generated for encryption and decryption processes.
- User Input: User provides text or data to be encrypted or selects encrypted data to be decrypted.
- Encryption: The input data is encrypted using the chosen algorithm and keys.
- Decryption: The encrypted data is decrypted back to its original form using the decryption algorithm and keys.
- Educational Content: Provide information on encryption, decryption, and security concepts for user reference.

Architecture Diagram



Technologies Used

- Programming Language: C is Utilized for implementing cryptographic algorithms and handling encryption/decryption processes.
- Libraries: mbedTLS/OpenSSL: Used for cryptographic functions such as RSA, AES, DES, and SHA implementations.
- User Interface: Graphical User Interface (GUI) Provides interaction for users to input data and select encryption/decryption options.
- Development Tools: Integrated Development Environment (IDE): Such as Visual Studio Code
- Version Control: Git For version control and collaboration on the project codebase.
- By leveraging these technologies, the project can effectively implement the interactive simulation of cryptographic techniques using mbedTLS/OpenSSL, enabling users to experiment with encryption, decryption, key generation, and various cryptographic algorithms.

Team Members And Contribution

Name	Contribution
Ojas Shashikant Hedau	Lead Developer: Responsible for implementing cryptographic algorithms, key generation, and ensuring accuracy in encryption/decryption processes.
Sahil Dinesh Patil	User Interface Developer: Designs and develops the user interface for data input, algorithm selection, and visual feedback.
Om Shashikant Hedau	Documentation Specialist: Creates detailed documentation on cryptographic algorithms, educational content, and ensures the simulation's educational value.

Conclusion

Working on the encryption and decryption simulation project alongside a skilled team, we are confident in our ability to create an interactive tool that will serve as a valuable educational resource for cryptography enthusiasts. With each team member playing a crucial role in algorithm implementation, user interface design, and documentation creation, we are well-equipped to deliver a comprehensive solution that enhances understanding and experimentation in the field of cybersecurity. By combining our expertise and collaborating effectively, we are on track to develop a tool that not only educates users on cryptographic techniques but also fosters a deeper appreciation for the importance of secure communication in the digital world.