INDIAN INSTITUTE OF INFORMATION TECHNOLOGY

Security Engineering

Advanced Secure Messaging System: A Unified Approach Using AES and RSA

Under the Guidance of

Dr. Suvadip Hazra

B.Jagadish(20bcs032)

7th July, 2024

Certificate

It is certified that the work contained in the project report titled " Advanced Secure Messaging System:A Unified Approach Using AES and RSA " by Bhukya Jagadish (20bcs032) has been carried out under my/our supervision and that this work has not been submitted elsewhere for a degree.

Signature of Supervisor(s)

Name(s)

Department(s)

(Month, Year)

Abstract

This report explores the development of an advanced secure messaging system that combines the strengths of Advanced Encryption Standard (AES) and Rivest-Shamir-Adleman (RSA) algorithms. The unified approach ensures efficient and secure communication, addressing the growing need for data confidentiality, integrity, and authenticity in the digital age.

Introduction

In the digital age, the need for secure communication has become paramount. With increasing threats from cyber-attacks and data breaches, ensuring the confidentiality and integrity of messages is critical. This project explores an advanced secure messaging system that leverages the strengths of both Advanced Encryption Standard (AES) and Rivest-Shamir-Adleman (RSA) algorithms to provide a robust and secure communication framework.

Objectives

The main objectives of this project are:

1. To develop a secure messaging system using AES and RSA algorithms.
2. To ensure the confidentiality, integrity, and authenticity of messages.
3. To leverage the efficiency of AES for data encryption and the security of RSA for key exchange.
4. To provide a detailed implementation guide for the unified approach.

Methodology

The methodology for developing the advanced secure messaging system includes the following steps:

1. Research on AES and RSA algorithms.
2. Designing the system architecture combining AES and RSA.
3. Implementing the key generation, key exchange, encryption, and decryption processes.
4. Testing the system for security and efficiency.
5. Analyzing the results and making necessary improvement

Detailed Description of AES and RSA

Advanced Encryption Standard (AES):

- Symmetric Encryption: AES is a symmetric key algorithm, meaning the same key is used for both encryption and decryption.
- Block Cipher: AES operates on fixed-size blocks of data (128 bits) and supports key sizes of 128, 192, and 256 bits.
- Security and Efficiency: Known for its speed and security, AES is widely used in various applications including SSL/TLS for secure web browsing.

Rivest-Shamir-Adleman (RSA):

- Asymmetric Encryption: RSA is an asymmetric key algorithm, using a pair of keys—a public key for encryption and a private key for decryption.
- Key Exchange: RSA is often used for secure key exchange and digital signatures.
- Security: RSA's security is based on the computational difficulty of factoring large integers.

Unified Approach Using AES and RSA

Combining Strengths:

- Efficiency of AES: AES is efficient for encrypting large amounts of data.
- Security of RSA: RSA provides secure key distribution, ensuring that the AES key can be safely shared between communicating parties.

Process Workflow:

1. Key Generation: RSA keys (public and private) are generated by both the sender and the receiver. A symmetric AES key is generated for each communication session.
2. Key Exchange: The sender encrypts the AES key using the receiver's RSA public key and sends it to the receiver. The receiver decrypts the AES key using their RSA private key.
3. Message Encryption and Transmission: The sender encrypts the message using the AES key. The encrypted message is transmitted to the receiver.
4. Message Decryption: The receiver decrypts the message using the shared AES key.

Security Analysis Confidentiality:

- The use of AES ensures that the message content remains confidential.
- The AES key exchange via RSA ensures that only the intended recipient can decrypt the key and hence the message

Integrity:

- Digital signatures can be used to ensure the message has not been tampered with.
- The sender signs the message with their RSA private key, and the receiver can verify it using the sender's RSA public key.

Authentication:

- Ensures that the message is from the legitimate sender.
- Public key infrastructure (PKI) can be used to manage and distribute RSA keys securely.

Implementation

Technologies and Tools:

- Programming Languages: Python, Java, or C++.
- Libraries: PyCryptodome (Python), Bouncy Castle (Java), OpenSSL (C++).
- Development Environment: Integrated Development Environments (IDEs) such as PyCharm, IntelliJ IDEA, or Visual Studio Code.

Steps:

1. RSA Key Generation: Use cryptographic libraries to generate RSA key pairs.
2. AES Key Generation: Generate a random AES key for each session.
3. Encrypt AES Key with RSA: Encrypt the AES key with the receiver's RSA public key.
4. Encrypt Message with AES: Encrypt the message using the AES key.
5. Decrypt AES Key: The receiver decrypts the AES key with their RSA private key.
6. Decrypt Message: The receiver decrypts the message using the AES key.

Conclusion

By combining AES and RSA, this secure messaging system leverages the strengths of both symmetric and asymmetric encryption, providing a secure and efficient communication solution. This unified approach ensures the confidentiality, integrity, and authenticity of messages, making it suitable for various secure communication applications.This outlines implementation of an advanced secure messaging system using AES and RSA encryption. I had included key generation, encryption/decryption functions, data visualization using word clouds, and demonstrated the integration of Rouge score and F1 score calculation for text evaluation tasks. Each section provides clear code snippets and examples to illustrate its functionality within the context of the project.

Code:

Mess_sys_20bcs032_SE.ipynb

References

- NIST AES: https://csrc.nist.gov/publications/detail/fips/197/final
- RSA Algorithm: https://en.wikipedia.org/wiki/RSA_(cryptosystem)
- Cryptographic Key Management: https://csrc.nist.gov/projects/key-managemen

- Shamir, A., Adleman, L., and Rivest, R. L. (1978). a technique for getting public-key cryptosystems and digital signatures. ACM Communications, 21(2), 120–126.
- In 1999, Daemen, J., and Rijmen, V. The Rijndael AES Proposal. NIST's Proposal for AES.
- Ferguson, N., Kohno, T., and Schneier, B. (2010). Design Principles and Real-World Applications of Cryptography Engineering.
- Wiley Dworkin (2001) Methods and Techniques for Suggested Block Cipher Modes of Operation. 800-38A, NIST Special Publication.
- 2001 saw the National Institute of Standards and Technology. Advanced Encryption Standard (AES), FIPS PUB 197.
- Technology and Standards Institute (NIST) (2002). Digital Signature Standard (DSS), 32FIPS PUB 186-2.