JSS MAHAVIDYAPEETHA
JSS SCIENCE AND TECHNOLOGY UNIVERSITY
SRI JAYACHAMARAJENDRA COLLEGE OF ENGINEERING
- Constituent College of JSS Science and Technology University
- Approved by AICTE
- Governed by the Grant-in-Aid Rules of Government of Karnataka
- Identified as lead institution for World Bank Assistance under TEQIP Scheme
DIAMOND JUBILEE YEAR :: 1963 - 2023

"RABIN CRYPTOSYSTEM"

Algorithm implementation report in fulfilment of curriculum prescribed for the Cryptography and Network Security (20CS552) course for the award of degree of

Bachelor of Engineering
In
Computer Science and Engineering
By

Abhishek M	01JST21CS003
Amith K Kumble	01JST21CS014
Gagandeep D	01JST21CS039
Vikhyat G Gowda	01JST21CS171
Vidisha Nanjappa B	01JCE21CS119
Saanvi Gopinath	01JST21CS122

Submitted To
Prof. Swetha P. M
Assistant Professor
Dept. of CSE, JSSSTU, Mysore

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING - 2023

TABLE OF CONTENTS

1 Introduction

The Rabin cryptosystem is a public-key cryptographic algorithm developed by Michael Rabin in the late 1970s. It is based on the difficulty of factorizing large composite numbers into their prime factors, a problem believed to be computationally hard. Unlike other popular encryption schemes like RSA, which relies on the difficulty of factoring the product of two large primes, the Rabin cryptosystem encrypts and decrypts messages by exploiting the difficulty of extracting square roots modulo a composite number that is the product of two large primes.

At its core, the Rabin cryptosystem involves generating a public key from the product of two distinct prime numbers and using this key to encrypt plaintext messages into ciphertext. Decryption is achieved by utilizing the knowledge of the prime factors of the composite modulus, allowing for the retrieval of the original plaintext from the ciphertext. Its security relies on the challenge of finding the square roots modulo a composite number without knowing its prime factors, making it a potentially secure encryption scheme in the realm of public-key cryptography.

Despite its robustness against factorization-based attacks, the Rabin cryptosystem has seen less widespread adoption compared to algorithms like RSA, partly due to its vulnerability to attacks like the Chosen Ciphertext Attack (CCA). However, its simplicity and reliance on square roots as opposed to the more complex mathematical operations in other encryption schemes make it an area of ongoing research and interest in the cryptographic community.

2 Working Procedure

2.1 Key Generation

- **Prime Number Selection:** Choose two distinct large prime numbers, p and q, typically of similar bit lengths. These primes should remain secret.
- **Modulus Calculation:** Compute the modulus $n = p \times q$. This modulus forms part of the public key.

2.2 Encryption

- **Message Preparation:** To encrypt a plaintext message M, it must first be converted into a number in the range from $0\ to\ n-1$.

- **Encryption Process:** Compute the ciphertext *C* by squaring the plaintext message M and taking the result modulo n. $C = M^2\ mod\ n$

  The resulting ciphertext *C is* then transmitted to the intended recipient.

2.3 Decryption

- **Private Key** Utilization: To decrypt the received ciphertext C and retrieve the original message M, the recipient uses their knowledge of the prime factors p and q (the private key).

- **Square Root Calculation:** Calculate the four potential square roots of $C\ modulo\ n$ using the Chinese Remainder Theorem (CRT):

  - $\boldsymbol{M_1 = \sqrt{C}\ mod\ p}$
  - $\boldsymbol{M_2 = -\sqrt{C}\ mod\ p}$
  - $\boldsymbol{M_3 = \sqrt{C}\ mod\ q}$
  - $\boldsymbol{M_4 = -\sqrt{C}\ mod\ q}$

- **CRT Combination:** Use the CRT to combine these results to obtain four potential solutions for $\boldsymbol{M}$**.**

- **Plaintext Determination:** Out of the four possibilities, two pairs should match, representing the original plaintext message M (bearing in mind that $M\ and\ -M$ are considered equivalent due to the squaring operation in the encryption process).

3 Implementation

Rabin cryptosystem is implemented in Java, accompanied by a Swing-based graphical user interface (GUI) and following the Model-View-Controller (MVC) architecture, embodies a structured and user-friendly platform for encryption and decryption operations. This section delves into the specifics of the

Rabin cryptosystem's implementation in Java, highlighting the utilization of Swing for the GUI, incorporating Java's inbuilt utilities like SecureRandom and BigInteger, and adhering to the MVC design pattern.

3.1 Utilizing Java's SecureRandom and BigInteger

- **SecureRandom for Key Generation:** The SecureRandom class from Java's security package is employed to generate secure and unpredictable prime numbers, which serve as the basis for creating the cryptographic keys in the Rabin cryptosystem. This class ensures the provision of cryptographically strong random numbers, vital for robust key generation and system security.

- **BigInteger for Arithmetic Operations:** The BigInteger class in Java's math package offers support for handling large integer values required in the Rabin cryptosystem's computations. Operations involving large prime numbers, modular arithmetic, squaring, and extracting square roots modulo a composite number are efficiently managed using the BigInteger class, ensuring accurate cryptographic calculations.

3.2 Model-View-Controller (MVC) Architecture

- **Model: Core Cryptographic Operations:** The Model encapsulates the core functionalities of the Rabin cryptosystem, including key generation, encryption, and decryption algorithms. It utilizes the SecureRandom and BigInteger utilities for secure key generation and cryptographic computations. The Model operates independently of the UI, maintaining the integrity of cryptographic operations.

- **View: Graphical User Interface:** The View component, developed using Java Swing, represents the GUI elements visible to users. It comprises interactive components and layout designs, providing a user-friendly platform for inputting data and triggering encryption/decryption processes.

- **Controller: Managing User Interactions**: The Controller serves as an intermediary between the View and the Model. It handles user inputs from the GUI, triggers corresponding cryptographic operations via the Model, and updates the View with the resulting output, ensuring seamless

synchronization between user interactions and cryptographic functionalities.

4 Trade-offs

4.1 Advantages

- **Security Against Factorization Attacks:** The Rabin cryptosystem relies on the computational complexity of extracting square roots modulo a composite number. Its security is linked to the difficulty of factoring the modulus into its prime factors, making it resistant to attacks based on prime factorization. If factoring large numbers remains computationally hard, Rabin encryption can be secure.

- **Simplicity in Encryption and Decryption:** The encryption and decryption processes in the Rabin cryptosystem are relatively straightforward compared to some other public-key cryptographic algorithms. It involves squaring for encryption and computing square roots for decryption, making it conceptually simpler in its operations.

- **No Padding Requirement:** Unlike RSA, which often requires padding schemes to prevent certain attacks (like the padding oracle attack), the Rabin cryptosystem doesn't mandate specific padding techniques. This simplicity can lead to potentially fewer vulnerabilities related to padding schemes.

4.2 Disadvantages

- **Non-Uniqueness of Decryption:** During decryption, the Rabin cryptosystem can produce multiple possible plaintexts due to the existence of four potential square roots modulo the composite modulus n. Extra information or protocols may be necessary to identify the correct plaintext, potentially complicating the decryption process.

- **Vulnerability to Chosen Ciphertext Attacks (CCA):** The Rabin cryptosystem is susceptible to certain attacks, including the Chosen Ciphertext Attack (CCA). This vulnerability poses a risk in scenarios where adversaries have access to a decryption oracle, allowing them to decrypt

chosen ciphertexts and potentially gain information about the secret key or the plaintext.

- **Less Adoption and Standardization:** Compared to widely adopted algorithms like RSA, the Rabin cryptosystem has seen limited real-world implementation and standardization. Its susceptibility to certain attacks, coupled with the presence of more established alternatives, has hindered its widespread adoption in practical applications.

- **Efficiency Concerns in Decryption:** The process of obtaining the original plaintext from the ciphertext in the Rabin cryptosystem, particularly when using the Chinese Remainder Theorem (CRT) to compute square roots, can be computationally intensive and less efficient compared to some other encryption schemes, impacting performance in certain scenarios.

5 Conclusion

In conclusion, the Rabin cryptosystem presents a unique approach to public-key cryptography, leveraging the complexity of extracting square roots modulo a composite number as the foundation for encryption and decryption. Its reliance on the difficulty of factoring large numbers offers a potential avenue for robust security, particularly against attacks based on prime factorization. However, the Rabin cryptosystem also carries inherent limitations and vulnerabilities that impact its widespread adoption and practical implementation.

While offering relative simplicity in its encryption and decryption processes compared to some other cryptographic algorithms, the Rabin cryptosystem faces challenges regarding the non-uniqueness of decryption, potentially resulting in multiple plausible plaintexts. Addressing this issue may require additional protocols or information to discern the correct message, adding complexity to the decryption process.

Moreover, vulnerabilities such as susceptibility to chosen ciphertext attacks (CCA) and its less established presence in standardized cryptographic practices have hindered its broader adoption in real-world applications. Efficiency concerns, particularly in the decryption phase, further contribute to the considerations when evaluating its practicality in certain contexts.

Despite these limitations, ongoing research continues to explore the potential of the Rabin cryptosystem and its variations. Enhancing its security against known vulnerabilities, addressing efficiency concerns, and exploring its applicability in specific use cases remain areas of interest within the cryptographic community.

Ultimately, while the Rabin cryptosystem demonstrates intriguing cryptographic principles and a different paradigm compared to widely adopted schemes like RSA, careful evaluation of its strengths and weaknesses is essential when considering its utilization in practical cryptographic implementations.

6 References

- **Cryptography & Network Security** Forouzan, Behrouz A. McGraw-Hill Education, 2018.
- **Java Swing Documentation:** https://docs.oracle.com/javase/tutorial/uiswing/