J. of Ramanujan Society of Mathematics and Mathematical Sciences
Vol. 12, No. 1 (2024), pp. 127-136

DOI: 10.56827/JRSMMS.2024.1201.11 **ISSN (Online): 2582-5461**

ISSN (Print): 2319-1023

ENHANCING COMMUNICATION SECURITY: A PRIVATE KEY CRYPTOSYSTEM EMPLOYING THE CHINESE REMAINDER THEOREM

Nagabharana N., L. Praveen Kumar and L. Jyotsna

Department of Mathematical and Computational Sciences,
Sri Sathya Sai University for Human Excellence,
Kalaburagi, Karnataka, INDIA

E-mail : nagabharana@sssuhe.ac.in, praveen.l@sssuhe.ac.in,
jyotsna.l@sssuhe.ac.in

(**Received:** *Aug. 30, 2024* **Accepted:** *Dec. 25, 2024* **Published:** *Dec. 30, 2024)*

Abstract: This paper presents a new private key cryptosystem utilizing the Chinese Remainder Theorem (CRT) for encryption and decryption. The system generates keys by selecting a specified number of distinct primes and a random integer 'a', which is chosen as a random number greater than all the selected primes and coprime to them, ensuring security without revealing the modulus. Encryption and decryption operations are performed within the modulus of each prime, enhancing data protection. The system's security relies solely on the integrity of prime numbers and the randomness of 'a', offering a promising solution for secure symmetric key cryptography.

Keywords and Phrases: Private key Cryptosystem, Modular arithmetic, Chinese remainder theorem and confidentiality.

2020 Mathematics Subject Classification: 11T71, 94A60.

1. Introduction

In the digital age, secure communication is paramount, driving the increasing importance of cryptography [8]. Cryptography, the art of sending messages in disguised form, plays a pivotal role in ensuring the confidentiality and integrity of

digital communications. Broadly classified into public and private key cryptosystems, each has its unique advantages and applications. Public key cryptosystems [4] offer enhanced security but come with a computational overhead, particularly in encryption and decryption processes. Consequently, for routine communication needs, private key cryptosystems are widely favored [12]. Public key cryptography finds its niche in key exchange, authentication, verification, and digital signatures [3, 7, 9].

Symmetric key cryptography plays a crucial role in private key cryptosystems, utilizing the same key for both encryption and decryption. Among the most notable algorithms are the Data Encryption Standard (DES) and the Advanced Encryption Standard (AES). DES, based on the Feistel network, was extensively adopted but became vulnerable to brute force attacks due to its limited key size [2]. AES, which employs a substitution-permutation network, supports multiple key lengths and offers enhanced resistance to cryptanalytic techniques [5]. Recent advancements in symmetric ciphers have introduced optimized designs for better efficiency and resistance to side-channel attacks. Additionally, block ciphers have been tailored for lightweight cryptography in resource-constrained environments, such as IoT, emphasizing energy efficiency and security [11]. Despite significant advancements, most symmetric key cryptosystems do not incorporate advanced mathematical frameworks like modular arithmetic, prime numbers, or computational number theory.

In this paper, we propose a novel private key cryptosystem where encryption and decryption are intricately tied to the Chinese Remainder Theorem (CRT). Our method hinges on the selection of a secure key represented as an ordered pair: one component comprises a set of k distinct primes, while the other is a random integer exceeding all primes and co-prime with each prime. We adopt the ASCII table for mapping message units, numerically representing each unit according to the ASCII table. Encryption entails dividing each message into k numbers, encrypting each under the modulo of a distinct prime. Decryption, utilizing CRT, aggregates these encrypted components into a single number, uniquely solvable modulo the product of all primes.

Importantly, our cryptosystem preserves the confidentiality of the modulus, treating it as part of the private key. Successful decryption necessitates correctness across all moduli, ensuring the security of our system. Scaling the number of primes enhances security but also increases computational complexity due to the block-like division of messages. Unlike existing symmetric ciphers, our system leverages modular arithmetic and ensures an expanded key space. The complexity of encryption is tied to the multiplication of two integers modulo a prime and

scales with the number of primes k. Decryption complexity, governed by CRT computation, is $O((\log n)^2)$, where n is the product of all k primes [7].

By leveraging the Chinese Remainder Theorem, our cryptosystem offers a promising avenue for secure communication, balancing security and computational efficiency in private key cryptography. Additionally, a Python program has been authored to implement our cryptographic system, specifically designed for facilitating communication.

2. Preliminaries

In this section, we lay the groundwork by discussing fundamental concepts in modular arithmetic and the Chinese Remainder Theorem (CRT), which are crucial for understanding our proposed cryptosystem [1, 10].

2.1. Modulo Arithmetic

Modulo arithmetic, also known as clock arithmetic, is a fundamental concept in mathematics where numbers"wrap around" upon reaching a certain value called the modulus. In modulo arithmetic, two integers a and b are said to be congruent modulo n, denoted as $a \equiv b \pmod{n}$, if their difference is divisible by n. For example, $11 \equiv 3 \pmod{4}$ because $(11 - 3) = 8$ is divisible by 4.

2.2. Co-prime Numbers

Two integers are said to be co-prime if their greatest common divisor (GCD) is 1. In other words, they have no common factors other than 1. Co-prime numbers play a crucial role in various number theoretic concepts, including the Chinese Remainder Theorem.

2.3. Chinese Remainder Theorem (CRT)

The Chinese Remainder Theorem (CRT) is a fundamental theorem in number theory that provides a solution to a system of simultaneous congruences.

Consider a system of congruences:

$$\begin{cases} x \equiv a_1 \pmod{m_1} \\ x \equiv a_2 \pmod{m_2} \\ \vdots \\ x \equiv a_k \pmod{m_k} \end{cases}$$

where $m_1, m_2, \ldots, m_k$ are pairwise co-prime integers. CRT states that this system has a unique solution modulo $M = m_1 \cdot m_2 \cdots m_k$, where M is the product of the moduli.

2.4. Finding Solutions using CRT

To find the solution to the system of congruences (in the above) using CRT, we first compute $M = m_1 \cdot m_2 \cdots m_k$. Then, for each congruence, we compute $M_i = \frac{M}{m_i}$ and find the modular inverse y_i of M_i modulo m_i. Finally, the solution x is given by:

$$x = \sum_{i=1}^{k} a_i \cdot y_i \cdot M_i \pmod{M}$$

These concepts form the basis of our proposed private key cryptosystem, which we will elucidate in subsequent sections.

3. A Novel Private Key Cryptosystem Based on CRT

This section introduces a novel private key cryptosystem algorithm leveraging the Chinese Remainder Theorem (CRT). The algorithm encompasses key generation, encryption, and decryption processes. For encryption and decryption, a random integer a is utilized alongside a set of k distinct primes $p_1, p_2, \cdots, p_k$ as moduli. The integer a is chosen to be greater than all primes and co-prime to them, forming the private key. During encryption, plaintext message units are divided into k integers, resulting in a ciphertext consisting of k numbers or a block of length k , each smaller than the respective prime p_i, where $1 \leq i \leq k$ respectively. Decryption involves utilizing the private key, an ordered pair comprising k primes and a, alongside CRT to obtain a unique solution modulo the product of primes, that unique number gives the original plaintext message unit. Additionally, numerical values for each message unit are determined using the ASCII table. Furthermore, a Python program based on our algorithm is developed for communication purposes.

3.1. Algorithm

Key Generation:

First, choose k distinct primes $p_1, p_2, ..., p_k$ such that $p_1 < p_2 < \cdots < p_k$. Also, choose an integer a such that $a > p_k$ and $\gcd(a, p_i) = 1$ for all i. Therefore, the key in this system is $((p_1, p_2, ..., p_k), a)$.

Encryption:

Suppose m is the numerical value of the message unit chosen according to the ASCII table. Then, m is encrypted as follows:

$$\begin{aligned} m \cdot a &\equiv b_1 \pmod{p_1} \\ m \cdot a &\equiv b_2 \pmod{p_2} \\ &\vdots \\ m \cdot a &\equiv b_k \pmod{p_k} \end{aligned}$$

The ciphertext will be $(b_1, b_2, ..., b_k)$.

Decryption:

The ciphertext $(b_1, b_2, ..., b_k)$ is decrypted as follows using the key:

First, compute the inverse of a modulo $p_1, p_2, ..., p_k$, denoted as $a_1, a_2, ..., a_k$ respectively. Then, solve the following congruence by CRT:

$$\begin{aligned} x &\equiv a_1 b_1 \pmod{p_1} \\ x &\equiv a_2 b_2 \pmod{p_2} \\ &\vdots \\ x &\equiv a_k b_k \pmod{p_k} \end{aligned} \tag{1}$$

We will get a unique solution modulo $p_1 p_2 ... p_k$, the product of primes. The unique solution is the numerical value m of the original message.

Decryption will work as the solution for the each congruence in (1) is m and CRT gives the unique solution modulo the product $p_1 p_2 ... p_k$. As the highest numerical value in the ASCII table is 127, and we choose primes such that the product is more than 127, our decryption will work. If we want to accommodate more characters, we can choose primes accordingly.

Note that the algorithm will work not only for prime numbers but also for any k positive integers that are pairwise coprime. Additionally, the reason for choosing 'a' greater than all the primes is to avoid the case where $m \cdot a$ is less than any p_i. If m is less than any p_i, then there is a chance that $m \cdot a = b_i$ for some i (in the case of $a \not> p_k$). If the ciphertext unit $(b_1, b_2, \cdots, b_k)$ is revealed, with the knowledge of b_i and all possible values of m, we can find the value of a. This reveals part of the key, and with the value of a, we can find the original message as $m \cdot a = b_i$. Another reason to choose 'a' coprime to each p_i is that in decryption, we have to find the inverse of a modulo each p_i.

Regarding security, it depends entirely on the choice of primes as moduli and the random integer a. One advantage is that the modulus is not revealed, as it is part of the private key. Knowledge of all primes is necessary for decrypting the message, as CRT gives the original message if all primes are correct. Additionally, knowledge of the random integer a is also necessary; without it, even with the correct primes, CRT will not give the original message. Therefore, for security reasons, if we fix the primes, we can randomly change the value of a. Then, there is no need to change the entire key every time; sometimes, changing the random value of a suffices.

Furthermore, each message unit is divided into blocks of length k, where k represents the number of primes. Therefore, the number of primes used in the

system is also kept confidential. If a full message consists of l units, then the ciphertext numerical consists of $k \times l$ numbers. When observing the ciphertext, the recipient first divides it into blocks of length k. Thus, the size of the block also plays an important role in security.

3.2. Complexity

The complexity of the proposed cryptosystem arises from the encryption and decryption processes, which are inherently tied to modular arithmetic and the Chinese Remainder Theorem [6] [7].

Encryption: During encryption, we choose k distinct primes and perform modular arithmetic operations for each message unit m, multiplied with a, under the modulus of each prime. Since a is greater than all the k primes, the computational complexity of the encryption is $O(k \cdot (\log a)^2)$.

Decryption: The decryption process is predominantly determined by solving the CRT problem. If n is the product of all k primes, the complexity of solving the CRT is $O((\log n)^2)$. This ensures that decryption is computationally efficient while maintaining the security of the cryptosystem.

Overall, the cryptosystem achieves a balance between computational efficiency and cryptographic strength, making it suitable for secure communication in various applications.

Example 1. Choose three primes $p_1 = 11$, $p_2 = 17$, and $p_3 = 41$, and $a = 45$. Therefore, the key is $((11, 17, 41), 45)$ and $k = 3$.
Consider the message ``KRISHNA". The numerical values according to the ASCII table are: $K \to 75, R \to 82, I \to 73, S \to 83, H \to 72, N \to 78$ and $A \to 65$.

Encryption of the message KRISHNA is as follows:

- For K: 75 and $a = 45$, then $75 \times 45 \equiv 9 \pmod{11}$, $75 \times 45 \equiv 9 \pmod{17}$, and $75 \times 45 \equiv 13 \pmod{41}$. So, for K, the corresponding ciphertext numericals are $(9, 9, 13)$.

- Similarly, for the remaining units R, I, S, H, N, A, the ciphertext numericals are $(5, 1, 0)$, $(7, 4, 5)$, $(6, 12, 4)$, $(6, 10, 1)$, $(1, 8, 25)$, $(10, 1, 14)$, respectively.

Therefore, the numerical ciphertext for the total message is:

$$[9, 9, 13, 5, 1, 0, 7, 4, 5, 6, 12, 4, 6, 10, 1, 1, 8, 25, 10, 1, 14]$$

The decryption for the above ciphertext is as follows:

- As there are three primes, divide the above ciphertext into blocks of size three: $(9, 9, 13)$, $(5, 1, 0)$, $(7, 4, 5)$, $(6, 12, 4)$, $(6, 10, 1)$, $(1, 8, 25)$, $(10, 1, 14)$.

- Take the first block $(9, 9, 13)$ and find the multiplicative inverse of $a = 45$ modulo 11, 17, and 41, which are 1, 14, and 31, respectively. Then consider the congruence:

$$\begin{aligned} x &\equiv 1 \times 9 \pmod{11} \\ x &\equiv 14 \times 9 \pmod{17} \\ x &\equiv 31 \times 13 \pmod{41} \end{aligned}$$

  That is,

$$\begin{aligned} x &\equiv 9 \pmod{11} \\ x &\equiv 7 \pmod{17} \\ x &\equiv 34 \pmod{41}. \end{aligned}$$

- Apply CRT to get the unique solution modulo $11 \times 17 \times 41 = 7667$.
- Let $m_1 = 11$, $m_2 = 17$, and $m_3 = 41$, and $c_1 = 9$, $c_2 = 7$, and $c_3 = 34$. Then, compute $M = m_1 \times m_2 \times m_3 = 7667$ and $M_1 = M/m_1 = 697$, $M_2 = M/m_2 = 451$, and $M_3 = M/m_3 = 187$.
- The inverse of $M_i \bmod m_i$ is y_i for $i = 1, 2, 3$. Then $y_1 = 3$, $y_2 = 2$, and $y_3 = 25$.
- The unique solution is $c_1 \times y_1 \times M_1 + c_2 \times y_2 \times M_2 + c_3 \times y_3 \times M_3 \pmod{7667}$, which evaluates to 75 and $75 \to K$.
- If we apply the same procedure for the remaining ciphertext blocks, then the corresponding plaintext for the blocks $(5, 1, 0)$, $(7, 4, 5)$, $(6, 12, 4)$, $(6, 10, 1)$, $(1, 8, 25)$, $(10, 1, 14)$ is `R`, `I`, `S`, `H`, `N`, `A`, respectively.

4. Python Implementation for Communication

Here is the Python implementation for communication using our algorithm:

```
def encrypt(msg, a, primes):
    numerical_identities = [ord(char) for char in msg]

    ciphertext = []
    for num in numerical_identities:
        ci = [(num * a) % prime for prime in primes]
        ciphertext.extend(ci)

    return ciphertext

def chinese_remainder_theorem(congruences):
```

```
    M = 1
    for _, mi in congruences:
        M *= mi

    result = 0
    for ai, mi in congruences:
        Mi = M // mi
        Mi_inv = pow(Mi, -1, mi)
        result += ai * Mi * Mi_inv

    return result % M

def decrypt(ciphertext, a, primes):
    block_size = len(primes)
    decrypted_msg = ""

    for i in range(0, len(ciphertext), block_size):
        block = ciphertext[i:i + block_size]

        congruences = [(block[j] * pow(a, -1, primes[j]), primes[j]) for j in range(block_size)]
        m = chinese_remainder_theorem(congruences)

        decrypted_msg += chr(m % 128) if m >= 0 and m <= 127 else ' '

    return decrypted_msg

def main():
    num_primes = int(input("Enter the number of primes: "))
    primes = [int(input(f"Enter prime {i+1}: ")) for i in range(num_primes)]

    a = int(input("Enter a: "))

    operation = input("Choose an operation (encrypt/decrypt): ").lower()

    if operation == 'encrypt':
        original_message = input("Enter the original message: ")
        numerical_ciphertext = encrypt(original_message, a, primes)
        print("Numerical Ciphertext:", numerical_ciphertext)

    elif operation == 'decrypt':
        user_input_ciphertext = input("Enter numerical values of ciphertext (comma-separated): ")
        numerical_ciphertext = [int(num) for num in user_input_ciphertext.split(',')]
        decrypted_msg = decrypt(numerical_ciphertext, a, primes)
        print("\nDecrypted Message:", decrypted_msg)

    else:
        print("Invalid operation. Please choose either 'encrypt' or 'decrypt'.")

if __name__ == "__main__":
    main()
```

This program prompts the user to enter the number of primes, the primes themselves, the value of a, and the operation (encryption or decryption). Then, it performs the chosen operation accordingly.

5. Conclusion

In conclusion, we have introduced a novel private key cryptosystem based on the Chinese Remainder Theorem (CRT). Our system employs a unique approach to key generation, encryption, and decryption processes. By utilizing a set of distinct primes and a random integer grater than and co-prime to each prime, alongside ASCII numerical values for message units, we have demonstrated a robust method for securing digital communication. The Chinese Remainder Theorem plays a pivotal role in aggregating encrypted components into a single number during decryption, ensuring confidentiality and integrity without revealing the modulus. With an expanded key space and computationally efficient encryption and decryption processes, our cryptosystem strikes a balance between security and performance. Also, we have provided a Python implementation of our algorithm for practical communication purposes. The security of our system relies on the secrecy of the primes and the random integer, offering a balance between computational efficiency and cryptographic strength. As future work, this approach can be extended from primes to integers with co-prime conditions and additional constraints to achieve greater security.

However, the proposed cryptosystem, while secure, faces higher computational complexity in encryption and decryption due to modular arithmetic and the Chinese Remainder Theorem. Additionally, managing a key structure involving k distinct primes poses challenges. Scalability issues, such as longer ciphertexts and greater processing times with increasing k, and potential implementation vulnerabilities further limit the practicality of the cryptosystem for certain applications.

Overall, our proposed cryptosystem presents a promising avenue for secure communication in the digital age, emphasizing the importance of leveraging mathematical principles in cryptographic design.

References

[1] Apostol, T. M., Introduction to Analytic Number Theory, Springer Science & Business Media, 1976.

[2] Biryukov, Alex, and Christophe De Cannière, Data encryption standard (DES), Encyclopedia of cryptography and security, (2011), 295-301.

[3] Boneh, D., & Shoup, V., A Graduate Course in Applied Cryptography, Retrieved from, 2000, `https://crypto.stanford.edu/~dabo/cryptobook/`

[4] Hoffstein, J., Pipher, J., & Silverman, J. H., An Introduction to Mathematical Cryptography, Springer Science & Business Media, 2014.

[5] Joan, Daemen, and Rijmen Vincent, The design of Rijndael: AES-the advanced encryption standard, Information Security and Cryptography, 196 (2002).

[6] Koblitz, Neal, A course in number theory and cryptography, Vol. 114, Springer Science & Business Media, 1994.

[7] Menezes, A. J., Van Oorschot, P. C., & Vanstone, S. A., Handbook of Applied Cryptography, CRC press, 1996.

[8] Rivest, R. L., Shamir, A., & Adleman, L. M., A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2) (1978), 120-126.

[9] Schneier, B., Applied Cryptography: Protocols, Algorithms, and Source Code in C. John Wiley & Sons, 1996.

[10] Shoup, V., A Computational Introduction to Number Theory and Algebra, Retrieved from, 2008, `https://shoup.net/ntb/`

[11] Singh, Saurabh, et al., Advanced lightweight encryption algorithms for IoT devices: survey, challenges and solutions, Journal of Ambient Intelligence and Humanized Computing, (2024), 1-18.

[12] Stinson, D. R., Cryptography: Theory and Practice, CRC press, 2005.