International Journal of
Mathematics And its Applications
ISSN: 2347-1557

Int. J. Math. And Appl., **12(3)**(2024), 133–149
Available Online: http://ijmaa.in

Lattice-Based Cryptographic Scheme for Secure Blockchain Development and Financial Systems

Alex Musa[1], G. Udoaka Otobong[2], Michael Nsikan John[2,*]

[1]*Department of Mathematics and Statistics, University of Port Harcourt, Nigeria*
[2]*Department of Mathematics, Akwa Ibom State University, Nigeria*

Abstract

This paper introduces a robust and mathematically rigorous lattice-based cryptographic scheme to enhance the security of blockchain networks, with a focus on financial systems. With the advent of quantum computing, traditional cryptographic systems like RSA and ECC face vulnerabilities that lattice-based schemes can effectively mitigate. The proposed approach integrates lattice-based cryptography with group theory to ensure secure communication and data integrity within blockchain ecosystems. By leveraging a 4×4 matrix-based key exchange, encryption, and decryption mechanism, this scheme ensures quantum-resistant security for money transfers, demonstrated here with the Naira (₦) as a practical example. Theoretical group-theoretical properties are utilized to achieve secure cryptographic operations over matrices, providing a practical, scalable, and quantum-resistant approach for financial transactions in blockchain systems.

Keywords: Blockchain security; Lattice-based cryptography; Financial systems; Quantum resistance; Cryptographic schemes; Group theory; Resilience; Scalability.
2020 Mathematics Subject Classification: 94A60, 68P25, 11T71.

1. Introduction

Blockchain technology has revolutionized financial systems by providing a decentralized and immutable ledger for transactions [9]. However, the increasing computational power of quantum computers poses a significant threat to traditional cryptographic schemes used in blockchain networks, including RSA and elliptic curve cryptography (ECC) [10]. Lattice-based cryptography, which leverages the hardness of certain lattice problems, has shown promise in addressing these security vulnerabilities due to its strong resistance to quantum attacks [1,2]. This paper proposes a novel cryptographic scheme based on lattice theory and group theory, focusing on matrix operations to secure transactions. A detailed mathematical analysis of the scheme's key exchange, encryption,

*Corresponding author (storm4help1@gmail.com)

and decryption processes is presented, particularly using the Naira for practical applications in financial systems [9].

2. Preliminaries and Theoretical Background

2.1 Lattice-Based Cryptography

A lattice in n-dimensional space is a discrete subgroup of $\mathbb{R}^n$. Formally, given a basis $B = \{b_1, b_2, \ldots, b_n\}$, the lattice generated by B is defined as:

$$\mathcal{L}(B) = \left\{ \sum_{i=1}^{n} z_i b_i : z_i \in \mathbb{Z} \right\}$$

Lattice-based cryptography relies on hard problems like the Shortest Vector Problem (SVP) and Learning With Errors (LWE), which are conjectured to be resistant to both classical and quantum attacks [1,4].

2.2 Group Theory in Lattice Cryptography

Group theory plays a crucial role in the construction of cryptographic schemes. Let G be a group under a binary operation "$*$". A matrix-based group can be formed using $GL_n(\mathbb{Z})$, the general linear group of degree n over the integers, which comprises invertible $n \times n$ matrices. This structure enables secure operations like key exchange and encryption over lattice-based schemes [6,7].

2.3 The 4×4 Matrix Structure

The cryptographic scheme uses 4×4 matrices for secure key exchange, encryption, and decryption. Let A be a 4×4 matrix over $\mathbb{Z}_q$, where q is a prime modulus. The operations on these matrices (e.g., multiplication, inversion) are performed in the modular field $\mathbb{Z}_q$.

3. Key Exchange Protocol

3.1 Parameters and Setup

- **Prime Modulus** q: A large prime number, chosen to define the field $\mathbb{Z}_q$. All matrix operations will be performed modulo q.
- **Public Matrix** A: A known 4×4 matrix over $\mathbb{Z}_q$, accessible to both parties.
- **Private Matrices**:
  - **Alice's private matrix** S_A: A randomly generated 4×4 invertible matrix over $\mathbb{Z}_q$.
  - **Bob's private matrix** S_B: A similar randomly generated 4×4 invertible matrix over $\mathbb{Z}_q$.

- **Random Noise Matrices**:
  - **Alice's noise matrix** R_A: A random perturbation matrix over $\mathbb{Z}_q$, adding extra security to the protocol [8].
  - **Bob's noise matrix** R_B: A random perturbation matrix over $\mathbb{Z}_q$.

3.2 Key Exchange Protocol Steps

The key exchange protocol involves both Alice and Bob generating public keys from their private matrices and the public matrix A. Then, they exchange their public keys to compute a shared secret key.

Step 1: Private Key Selection Both Alice and Bob select their private key matrices:

$$S_A = \begin{bmatrix} s_{11} & s_{12} & s_{13} & s_{14} \\ s_{21} & s_{22} & s_{23} & s_{24} \\ s_{31} & s_{32} & s_{33} & s_{34} \\ s_{41} & s_{42} & s_{43} & s_{44} \end{bmatrix}, \quad S_B = \begin{bmatrix} s'_{11} & s'_{12} & s'_{13} & s'_{14} \\ s'_{21} & s'_{22} & s'_{23} & s'_{24} \\ s'_{31} & s'_{32} & s'_{33} & s'_{34} \\ s'_{41} & s'_{42} & s'_{43} & s'_{44} \end{bmatrix}$$

where $s_{ij}, s'_{ij} \in \mathbb{Z}_q$. These matrices are kept secret and are not shared between Alice and Bob.

Step 2: Public Key Generation

- **Alice's Public Key Generation**: Alice uses her private key S_A, the public matrix A, and her noise matrix R_A to compute her public key:

  $$P_A = A \cdot S_A + R_A \mod q$$

  where:

  $$R_A = \begin{bmatrix} r_{11} & r_{12} & r_{13} & r_{14} \\ r_{21} & r_{22} & r_{23} & r_{24} \\ r_{31} & r_{32} & r_{33} & r_{34} \\ r_{41} & r_{42} & r_{43} & r_{44} \end{bmatrix}$$

  with elements $r_{ij} \in \mathbb{Z}_q$.

  The resulting public key P_A is:

  $$P_A = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \\ p_{41} & p_{42} & p_{43} & p_{44} \end{bmatrix}, \quad p_{ij} \in \mathbb{Z}_q$$

- **Bob's Public Key Generation**: Similarly, Bob generates his public key using his private key S_B,

the public matrix A, and his noise matrix R_B:

$$P_B = A \cdot S_B + R_B \mod q$$

where:

$$R_B = \begin{bmatrix} r'_{11} & r'_{12} & r'_{13} & r'_{14} \\ r'_{21} & r'_{22} & r'_{23} & r'_{24} \\ r'_{31} & r'_{32} & r'_{33} & r'_{34} \\ r'_{41} & r'_{42} & r'_{43} & r'_{44} \end{bmatrix}$$

The resulting public key P_B is:

$$P_B = \begin{bmatrix} p'_{11} & p'_{12} & p'_{13} & p'_{14} \\ p'_{21} & p'_{22} & p'_{23} & p'_{24} \\ p'_{31} & p'_{32} & p'_{33} & p'_{34} \\ p'_{41} & p'_{42} & p'_{43} & p'_{44} \end{bmatrix}, \quad p'_{ij} \in \mathbb{Z}_q$$

Step 3: Key Exchange and Shared Secret Generation

- **Alice Computes the Shared Secret Key**:

$$K_A = P_B \cdot S_A \mod q = (A \cdot S_B + R_B) \cdot S_A \mod q = A \cdot (S_B \cdot S_A) + R_B \cdot S_A \mod q$$

- **Bob Computes the Shared Secret Key**:

$$K_B = P_A \cdot S_B \mod q = (A \cdot S_A + R_A) \cdot S_B \mod q = A \cdot (S_A \cdot S_B) + R_A \cdot S_B \mod q$$

Since matrix multiplication is associative and distributive, both Alice and Bob obtain the same shared secret key:

$$K_A = K_B = A \cdot (S_B \cdot S_A) + R_B \cdot S_A \mod q = A \cdot (S_A \cdot S_B) + R_A \cdot S_B \mod q$$

The shared secret key K is a 4×4 matrix that both Alice and Bob can use for secure encryption and decryption in their communications.

3.3 Example of Key Exchange Protocol

Let's work through an example with specific matrices:

- Let $q = 12289$ (a large prime number).

- Let the public matrix A be:

$$A = \begin{bmatrix} 123 & 456 & 789 & 321 \\ 234 & 567 & 890 & 432 \\ 345 & 678 & 901 & 543 \\ 456 & 789 & 123 & 654 \end{bmatrix}$$

- Alice's private key matrix:

$$S_A = \begin{bmatrix} 2 & 3 & 5 & 7 \\ 11 & 13 & 17 & 19 \\ 23 & 29 & 31 & 37 \\ 41 & 43 & 47 & 53 \end{bmatrix}$$

- Bob's private key matrix:

$$S_B = \begin{bmatrix} 61 & 67 & 71 & 73 \\ 79 & 83 & 89 & 97 \\ 101 & 103 & 107 & 109 \\ 113 & 127 & 131 & 137 \end{bmatrix}$$

- Alice's noise matrix:

$$R_A = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

- Bob's noise matrix:

$$R_B = \begin{bmatrix} 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \\ 2 & 2 & 2 & 2 \end{bmatrix}$$

- Alice's public key:

$$P_A = A \cdot S_A + R_A \mod q$$

- Bob's public key:

$$P_B = A \cdot S_B + R_B \mod q$$

- Shared Secret Key:
  - Alice computes:

$$K_A = P_B \cdot S_A \mod q$$

– Bob computes:

$$K_B = P_A \cdot S_B \mod q$$

Both K_A and K_B will be equal, thus establishing a shared secret key matrix for secure communication.

4. Encryption and Decryption Using 4×4 Matrices

4.1 Encryption Scheme

Let M be the message matrix (a 4×4 matrix) containing elements representing the message in block form. The encryption process is as follows:

- **Ciphertext Generation**: The ciphertext matrix C is generated using the shared secret key K:

$$C = M \cdot K + E$$

where E is a small error matrix to ensure security against lattice attacks [15].

4.2 Decryption Scheme

To recover the original message matrix M, the recipient uses the shared secret key K:

$$M = (C - E) \cdot K^{-1}$$

4.3 Example: Naira Money Transfer

Suppose M represents a money transfer amount in Naira. Assume:

$$M = \begin{bmatrix} 500 & 0 & 0 & 0 \\ 0 & 1000 & 0 & 0 \\ 0 & 0 & 2000 & 0 \\ 0 & 0 & 0 & 1500 \end{bmatrix}$$

If K is a shared secret key matrix, then the ciphertext for secure transfer is computed as:

$$C = M \cdot K + E$$

For a secure transaction, only the authorized recipient with K^{-1} can decrypt C to recover M.

5. Group-Theoretical Operations and Examples

The proposed lattice-based cryptographic scheme leverages group theory to provide secure operations for encryption and decryption. In this context, the use of matrices over modular arithmetic forms a group structure, allowing secure communication through operations like matrix multiplication and inversion. Here, we explore how group theory facilitates cryptographic functions using a 4×4 matrix structure and modular arithmetic, as well as examples of operations with alphabetic representations for secure messaging.

5.1 Group Theory and Matrix Operations

Group Properties in Cryptography

A **group** is defined as a set G with a binary operation "$*$" satisfying the following properties:

1. **Closure**: For any $a, b \in G$, the result of the operation $a * b \in G$.
2. **Associativity**: For any $a, b, c \in G$, $(a * b) * c = a * (b * c)$.
3. **Identity**: There exists an element $e \in G$ such that for any $a \in G$, $a * e = e * a = a$.
4. **Inverses**: For every $a \in G$, there exists an element $a^{-1} \in G$ such that $a * a^{-1} = a^{-1} * a = e$.

In this cryptographic scheme, the group under consideration is the set of invertible 4×4 matrices over the finite field $\mathbb{Z}_q$, denoted as $GL_4(\mathbb{Z}_q)$. The group operation is matrix multiplication modulo q [13].

Matrix Groups over $\mathbb{Z}_q$

Let q be a prime modulus, and consider a 4×4 matrix M over $\mathbb{Z}_q$. The set of all invertible matrices forms a group:

- **Identity matrix**: The identity element I_4 is:

$$I_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

  which satisfies $M \cdot I_4 = I_4 \cdot M = M$ for any matrix M.

- **Matrix inverse**: Each matrix $M \in GL_4(\mathbb{Z}_q)$ has an inverse M^{-1}, such that:

$$M \cdot M^{-1} = M^{-1} \cdot M = I_4$$

These properties enable secure operations like encryption and decryption, as the group structure ensures that transformations (e.g., encryption) can be reversed (e.g., decryption).

5.2 Secure Messaging with Alphabet Matrices

In this section, we illustrate how alphabetic characters can be represented as matrices for encryption and decryption within the group structure. Each letter of the alphabet is encoded as a distinct 4×4 matrix over $\mathbb{Z}_q$.

Matrix Representation of Alphabets

Let's map each letter of the English alphabet to a unique 4×4 matrix modulo q. For example:

- A be represented as:

$$A \mapsto \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- B be represented as:

$$B \mapsto \begin{bmatrix} 2 & 1 & 0 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 0 & 2 & 1 \\ 1 & 0 & 0 & 2 \end{bmatrix}$$

- C be represented as:

$$C \mapsto \begin{bmatrix} 3 & 0 & 1 & 0 \\ 0 & 3 & 0 & 1 \\ 1 & 0 & 3 & 0 \\ 0 & 1 & 0 & 3 \end{bmatrix}$$

- Similarly, other letters $D, E, \ldots, Z$ are represented as different 4×4 matrices over $\mathbb{Z}_q$.

Group Operations on Alphabet Matrices

To encrypt a message, each letter of the plaintext is mapped to its corresponding matrix and then transformed using the shared secret key matrix K. The group operation used for encryption is matrix multiplication modulo q.

Example of Encrypting the Word "ABC" Suppose we wish to encrypt the word "ABC" using the shared secret key matrix:

$$K = \begin{bmatrix} 7 & 3 & 5 & 2 \\ 4 & 6 & 7 & 1 \\ 9 & 2 & 8 & 5 \\ 3 & 1 & 2 & 4 \end{bmatrix}$$

The matrices for "A," "B," and "C" are:

$$M_A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad M_B = \begin{bmatrix} 2 & 1 & 0 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 0 & 2 & 1 \\ 1 & 0 & 0 & 2 \end{bmatrix}, \quad M_C = \begin{bmatrix} 3 & 0 & 1 & 0 \\ 0 & 3 & 0 & 1 \\ 1 & 0 & 3 & 0 \\ 0 & 1 & 0 & 3 \end{bmatrix}$$

The ciphertext matrices are obtained by multiplying each letter's matrix by the shared key K, modulo q:

- **Encryption of "A"**:

$$C_A = M_A \cdot K \mod q = \begin{bmatrix} 7 & 3 & 5 & 2 \\ 4 & 6 & 7 & 1 \\ 9 & 2 & 8 & 5 \\ 3 & 1 & 2 & 4 \end{bmatrix}$$

- **Encryption of "B"**:

$$C_B = M_B \cdot K \mod q = \begin{bmatrix} 18 & 9 & 7 & 2 \\ 8 & 20 & 13 & 1 \\ 9 & 2 & 8 & 6 \\ 17 & 3 & 10 & 8 \end{bmatrix}$$

- **Encryption of "C"**:

$$C_C = M_C \cdot K \mod q = \begin{bmatrix} 21 & 9 & 20 & 2 \\ 12 & 18 & 7 & 5 \\ 17 & 3 & 8 & 15 \\ 3 & 7 & 6 & 12 \end{bmatrix}$$

These ciphertext matrices C_A, C_B, and C_C correspond to the encrypted version of "ABC".

Decryption Process

To decrypt the ciphertext, the receiver uses the shared key K and its inverse K^{-1}. For a ciphertext matrix C:

$$M = C \cdot K^{-1} \mod q$$

This reverses the encryption process and recovers the original matrix representation of the plaintext message.

Decryption of "C_A" (for "A")

$$M_A = C_A \cdot K^{-1} \mod q = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The same process applies to decrypt C_B and C_C, recovering the matrices for "B" and "C," respectively.

5.3 Group-Theoretical Security Benefits

The group structure and modular arithmetic used in lattice-based cryptography provide robust security features:

- **High dimensionality**: Operations occur in high-dimensional matrix spaces, making brute-force attacks computationally infeasible.
- **Non-commutativity**: Matrix multiplication is non-commutative, providing additional complexity and security.
- **Quantum resistance**: Lattice-based problems like the Shortest Vector Problem (SVP) are believed to be resistant to quantum attacks [11,12].

6. Secure Money Transfer Scenarios Using Blockchain

In this section, we provide detailed mathematical solutions for two practical scenarios of secure money transfer within Nigeria and internationally using a lattice-based cryptographic scheme within a blockchain environment. The scenarios demonstrate how amounts in Naira (₦) are encrypted and securely transmitted using shared key matrices in a blockchain setting.

6.1 Secure Money Transfer Within Nigeria

Consider that Alice wants to securely send 500,000 Naira (₦500,000) to Bob within Nigeria through a blockchain-enabled platform. The cryptographic protocol uses lattice-based key exchange and encryption for secure transfer.

Key Components

- **Transaction Matrix** M: The transfer amount is represented as a 4×4 matrix over $\mathbb{Z}_q$, where q is a large prime number. Here, the matrix M represents the block data for the transfer:

$$M = \begin{bmatrix} 500000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

  This matrix encodes the transfer amount with padding zeros.

- **Shared Secret Key** K: The shared key is derived through a lattice-based key exchange between Alice and Bob (as described in Section 3). Let the shared secret key matrix be:

$$K = \begin{bmatrix} 7 & 3 & 5 & 2 \\ 4 & 6 & 7 & 1 \\ 9 & 2 & 8 & 5 \\ 3 & 1 & 2 & 4 \end{bmatrix}$$

- **Error Matrix** E: An error matrix E is used to introduce noise to the ciphertext for security against potential lattice attacks:

$$E = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Encryption Process

To securely transmit the amount M, Alice encrypts M using the shared key K and the error matrix E. The encrypted ciphertext matrix C is calculated as follows:

$$C = M \cdot K + E \mod q$$

Let's perform the matrix multiplication step-by-step:

Matrix Multiplication $M \cdot K$

$$M \cdot K = \begin{bmatrix} 500000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 7 & 3 & 5 & 2 \\ 4 & 6 & 7 & 1 \\ 9 & 2 & 8 & 5 \\ 3 & 1 & 2 & 4 \end{bmatrix}$$

The result is:

$$M \cdot K = \begin{bmatrix} 500000 \cdot 7 & 500000 \cdot 3 & 500000 \cdot 5 & 500000 \cdot 2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 3500000 & 1500000 & 2500000 & 1000000 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Adding Error Matrix E

$$C = M \cdot K + E = \begin{bmatrix} 3500000 & 1500000 & 2500000 & 1000000 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

The resulting ciphertext matrix is:

$$C = \begin{bmatrix} 3500001 & 1500001 & 2500001 & 1000001 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Transmission Over Blockchain

The encrypted matrix C is transmitted over the blockchain. Due to its cryptographic encoding, the transfer remains secure, ensuring data confidentiality and integrity.

Decryption by Bob

Upon receiving C, Bob decrypts the ciphertext using the shared secret key K and its inverse K^{-1}:

1. **Subtract the Error Matrix**:

$$C - E = \begin{bmatrix} 3500001 & 1500001 & 2500001 & 1000001 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} - \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Resulting in:

$$C - E = \begin{bmatrix} 3500000 & 1500000 & 2500000 & 1000000 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

2. **Multiply by the Inverse of** K:

$$M = (C - E) \cdot K^{-1} \mod q$$

Since the inverse of K will effectively reverse the transformation, Bob recovers the original matrix:

$$M = \begin{bmatrix} 500000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Thus, Bob knows that the transferred amount is 500,000 Naira.

6.2 Secure Money Transfer Outside Nigeria (International Transfer)

Now, suppose Alice wants to securely send 2 million Naira (₦2,000,000) to Bob, who is located outside Nigeria, e.g., in the USA.

Key Components

- **Transaction Matrix** M:

$$M = \begin{bmatrix} 2000000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

- **Shared Secret Key** K:

$$K = \begin{bmatrix} 7 & 3 & 5 & 2 \\ 4 & 6 & 7 & 1 \\ 9 & 2 & 8 & 5 \\ 3 & 1 & 2 & 4 \end{bmatrix}$$

- **Error Matrix** E:

$$E = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Encryption Process

The encryption process for international transfer is the same:

- **Matrix Multiplication**:

$$M \cdot K = \begin{bmatrix} 2000000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 7 & 3 & 5 & 2 \\ 4 & 6 & 7 & 1 \\ 9 & 2 & 8 & 5 \\ 3 & 1 & 2 & 4 \end{bmatrix}$$

The result is:

$$M \cdot K = \begin{bmatrix} 14000000 & 6000000 & 10000000 & 4000000 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

- **Adding Error Matrix**:

$$C = M \cdot K + E = \begin{bmatrix} 14000000 & 6000000 & 10000000 & 4000000 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

The resulting ciphertext is:

$$C = \begin{bmatrix} 14000001 & 6000001 & 10000001 & 4000001 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

Transmission Over Blockchain

The ciphertext C is securely transmitted over the blockchain, providing a secure and verifiable means of transferring funds internationally.

Decryption by Bob

Upon receiving the ciphertext C, Bob decrypts it:

- **Subtract Error Matrix**:

$$C - E = \begin{bmatrix} 14000001 & 6000001 & 10000001 & 4000001 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix} - \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

  Resulting in:

$$C - E = \begin{bmatrix} 14000000 & 6000000 & 10000000 & 4000000 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

- **Multiply by Inverse of** K:

$$M = (C - E) \cdot K^{-1} \mod q$$

  The inverse operation recovers:

$$M = \begin{bmatrix} 2000000 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

  Thus, Bob retrieves the original amount of 2 million Naira.

Conversion to Local Currency

After decryption, Bob can convert the amount from Naira to his local currency (e.g., USD) based on the current exchange rate.

7. Security Analysis

The proposed scheme's security relies on the hardness of lattice problems such as SVP and LWE [14]. Due to the high dimensionality and noise involved in lattice operations, it is computationally infeasible

for an attacker to recover the secret key without solving these problems.

8. Conclusion

The integration of lattice-based cryptographic schemes in blockchain networks presents a promising solution to quantum-resistant security. By utilizing a 4×4 matrix structure for key exchange, encryption, and decryption, this paper outlines a secure and efficient cryptographic scheme for financial systems. The use of group theory enhances the complexity and security of the scheme, ensuring that transactions, particularly those involving the transfer of money like the Naira, remain secure against quantum attacks.

Acknowledgment

The authors thank their respective institutions for providing the necessary support to carry out this research.

References

[1] D. Micciancio and O. Regev, *Lattice-based cryptography*, Mathematics of Information Security, (2004), 131-173.

[2] O. Regev, *On lattices, learning with errors, random linear codes, and cryptography*. Journal of the ACM, 56(6)(2009), 1-40.

[3] M. Zhandry, *How to construct quantum random functions*, FOCS 2012: IEEE 53rd Annual Symposium on Foundations of Computer Science, (2012), 30-39.

[4] C. Peikert, *A decade of lattice cryptography*, Foundations and Trends in Theoretical Computer Science, 10(4)(2016), 283-424.

[5] D. Micciancio and O. Regev, *Worst-case to average-case reductions based on Gaussian measures*, SIAM Journal on Computing, 37(1)(2007), 267-302.

[6] J. P. Serre, *Linear representations of finite groups*, Vol. 42, Springer Science & Business Media, (1977).

[7] T. W. Hungerford, *Algebra*, Springer Science & Business Media, (1980).

[8] J. Buchmann, E. Dahmen and M. Schneider, *Merkle tree traversal revisited*, In PQCrypto 2008, Springer, Berlin, (2008), 67-78.

[9] S. Nakamoto, *Bitcoin: A peer-to-peer electronic cash system*, (2008).

[10] P. W. Shor, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM Review, 41(2)(1997), 303-332.

[11] D. Micciancio, *Efficient reductions among lattice problems*, In Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, (2008), 84-93.

[12] C. Gentry, C. Peikert and V. Vaikuntanathan, *Trapdoors for hard lattices and new cryptographic constructions*, In Proceedings of the 40th annual ACM symposium on Theory of computing, (2010), 197-206.

[13] S. Lang, Algebra, Springer Science & Business Media, (2002).

[14] V. Lyubashevsky, C. Peikert and O. Regev, *On ideal lattices and learning with errors over rings*, Journal of the ACM, 60(6)(2013), 1-35.

[15] L. Lyu and A. Mishra, *Lattice-based public key cryptography in hardware: Challenges and opportunities*, In 2019 26th International Conference on High Performance Computing, Data, and Analytics (HiPC), (2019), 153-162.