Bulletin of the *Transilvania* University of Braşov
Series III: Mathematics and Computer Science, Vol. 5(67), No. 1 - 2025, 225-240
https://doi.org/10.31926/but.mif.2025.5.67.1.17

FEISTEL INSPIRED NOVEL HYBRID CRYPTOSYSTEM FOR INFORMATION SECURE PROCESSING

Nuh AYDIN[1], Djilali BEHLOUL[2], Sara BENATMANE[*,3] and Anouar Ben MABROUK[4]

Abstract

In this paper, we propose a novel hybrid encryption algorithm that enhances security by integrating DNA encoding, the Rabin algorithm, a one-time pad (OTP), and a Feistel-inspired structure. The algorithm begins with the application of a DNA-based OTP key, used exclusively for a single secure communication session. In the second step, it combines public and private keys to generate a Rabin key. By incorporating a Feistel-inspired scheme and leveraging the inherent randomness of DNA sequences, the resulting ciphertext achieves a high level of security, making it significantly more challenging to decrypt without the private key.

2020 Mathematics Subject Classification: 94A60, 11T71, 14G50, 68P25.
Key words: cryptography, DNA encoding, Rabin algorithm, Feistel structure, secure information.

1 Introduction

In modern life, individuals, organisms, institutions and also machines exchange information on a daily basis. The major problem in such exchanges is the safety and security of the information. In many situations, it is crucial that the information exchanged cannot be accessed by external users other than the sender and the receiver. Cryptography is a well-established discipline concerned with the security of data and information. Many mathematical tools are employed in

[1] Department of Mathematics, Kenyon College, Gambier, OH 43022, e-mail: aydinn@kenyon.edu

[2] LATN Laboratory, Department of Mathematics, University of Sciences and Technology Houari Boumedienne, BP 32, El Alia, 16111, Bab Ezzouar, Algeria, e-mail: dbehloul@yahoo.fr

[3*] *Corresponding author*, LATN Laboratory, Department of Mathematics, University of Sciences and Technology Houari Boumedienne, BP 32, El Alia, 16111, Bab Ezzouar, Algeria, e-mail: benatmanesara34@gmail.com

[4] Department of Mathematics, Faculty of Science, University of Tabuk, King Faisal Road, 47512 Tabuk, Saudi Arabia, e-mail: anouar.benmabrouk@fsm.rnu.tn

cryptography to achieve this goal, including wavelets, fractals, number theory, finite fields, algebraic geometry, Clifford algebra, and max-plus algebra to name a few. For example, in [14] the author developed a Clifford wavelet-based algorithm to embed copyright data within a multiresolution domain. The technique achieved a high resistance to common attacks. In the same direction, the discrete wavelet transform was applied in [18] to develop an efficient algorithm for encrypting and decrypting messages by representing the ciphertext through the wavelet decomposition vector. The authors in [27] developed an algorithm that embeds text information into images for secure transmission. The embedded text is encrypted using the Advanced Encryption Standard algorithm, and inserted in some sub-band wavelet decomposed image. The authors in [28] proposed a three-phase encryption process starting by encoding a plaintext via ASCII characters and added diffusion, followed by a computation of some points relative to an elliptic curve to introduce randomness. Finally, the output is decomposed using a max-plus algebra-based wavelet transform to generate the ciphertext. A comprehensive security analysis, including entropy, correlation, key space, key sensitivity, plaintext sensitivity, encryption quality, and resistance to various attacks (ciphertext-only, known-plaintext, chosen-plaintext, and chosen-ciphertext), demonstrates the robustness of the scheme. The paper [8] introduced a cryptographic algorithm using type IVa max-plus wavelet transforms (MP-Wavelets). The encryption and decryption processes are based on the analysis and synthesis of these wavelets. The encryption key is determined by the number of channels, ensuring its product relative to the plaintext length. The decryption key includes the encryption key and a binary-encoded sequence of detail components, making brute-force attacks difficult. The algorithm primarily uses maximization and addition operations, ensuring efficiency with linear complexity relative to the plaintext size. Experiments demonstrate strong encryption quality, significant key space, and robustness against cryptanalysis and security threats.

Secret key (also known as symmetric key) cryptography uses a single key to both encrypt and decrypt the plaintext. One-time Pad, DES, and AES are a few well-known examples of secret key encryption schemes. Historically, all cryptosystems were symmetric key. Public key cryptography (PKC) was a revolutionary idea that was introduced in 1976 by Diffie and Helman [13]. It refers to cryptographic techniques that use a public key for encryption and a private key for decryption. The idea of PKC is based on the notion of a one-way function. Examples of PK cryptosystmes include RSA and elliptic curve cryptography (ECC) [1, 22]. The security of many modern public key cryptosystems depends on the computational complexity of certain number theoretical problems such as factorization of large integers and the discrete logarithm problem.

The One-time Pad (OTP) is an encryption technique invented by Gilbert Vernam in 1917. It is based on an older encryption method called the Vigenère Cipher [6]. The OTP was improved by J. Mauborgne, who used random characters for the encryption key. In OTP, a message is encrypted by adding the secret key to the plaintext message. The key can only be used once.

Invented in 1979 by Michael Rabin [12, 26], the Rabin cryptosystem is a public key algorithm similar to the RSA. It is a variation of RSA with the fixed public encryption key $e = 2$ [15, 26]. The security of both cryptosystems is related to the integer factorization problem with the stronger connection in the case of the Rabin system. The Rabin cryptosystem requires simpler computations during the encryption phase, while the decryption step can use the Chinese Remainder Theorem to obtain the correct plaintext. The security of the Rabin cryptosystem is more closely tied to factoring than RSA. However, it has a disadvantage in decryption: for $N = pq$, the product of two prime numbers, squaring $\mod N$ is a four-to-one map, necessitating a rule to select the proper decryption result [7]. There are ways to choose the correct plaintext in the Rabin system, we provide an example in our proposed algorithm for determining the correct plaintext from the four possible results.

One of the recent methods in cryptography that can potentially increase information security is DNA cryptography. In relation to our work, DNA cryptography uses DNA sequences to hide data via various DNA technologies and biological techniques. Data hiding techniques have grown in popularity as a means of sending covert information in recent years. To prevent malicious intrusion and ensure a secure transmission, methods for data hiding based on DNA sequences have received a lot of attention. To address the issue of data hiding, numerous cryptographic techniques based on DNA have been presented. For example, the method suggested in [23] combines genetic engineering with cryptology technology to create an asymmetric encryption and signature cryptosystem. This approach is an investigation into biological cryptology.

The technique suggested in [10] presents a strategy based on DNA and the RSA cryptosystem, capable of providing an architectural foundation for the encryption and generation of digital signatures for all characters, simple text data, and text files. The process is broken down into four steps: key generation, data pre/post-processing, DNA encoding, and signature generation.

The technique suggested in [3] proposes a new cryptography method using symmetric key exchange, OTP, and DNA hybridization operations. The XOR operation with the OTP DNA sequence is used as an encryption technique. The proposed method used the matrix of the message and OTP DNA sequence to minimize the time complexity of encryption and decryption. The conclusion is that DNA cryptography could be combined with traditional cryptography to create a hybrid system with a higher level of security.

The technique given in [21] provides an enhanced cryptographic strategy based on DNA cryptography employing a Feistel-inspired structure that is based on symmetric key cryptography.

The method presented in [16] creates an improved cryptographic algorithm by combining current symmetric and asymmetric encryption. The proposed methodology leverages the strengths of various popular encryption schemes: AES's S-Box mapping, which makes the encryption difficult to predict, ElGamal encryption scheme, based on the difficulty of solving discrete logarithms, chaos-based security, which provides an avalanche effect through the use of multiple keys and RSA security, based on the difficulty of factorizing large integers.

The method outlined in [1] is a new secure way of concealing data based on DNA sequences. It has two rounds of encryption, similar to the established DES algorithm. The message is encrypted using two secret keys. The first key is created using elliptic curve cryptography (ECC) and the Gaussian kernel function (GKF), while the second key is generated using an arbitrary injective mapping on the second characters repeated in the first key.

Concealing information in the DNA sequence has drawn a lot of interest, and much research has been done to develop various innovative techniques. The goals of the new DNA cryptography algorithms that have been presented are to achieve the highest level of security during data transfer and to reduce the computational time of encryption and decryption. The primary goal of this paper is to create a novel hybrid cryptosystem that combines the strengths of each cipher to ensure stronger security. It does this by combining DNA, the Rabin algorithm, OTP, and a Feistel-inspired structure. The proposed system is a significant improvement over [4] in the following ways: we employ the Rabin cryptosystem instead of RSA because Rabin is potentially more secure than RSA [15, 17], and we add the Fiestel-inspired structure to our algorithm, which further enhances its security. The authors in [4] do not discuss the details of random key generation (generating truly random numbers/sequences is a major problem in cryptography) and state (in the abstract) that the system encrypts the binary key via an asymmetric cipher using the recipient's public key, but only the asymmetric cipher is used to encrypt the plaintext. Moreover, they do not explain how the private random binary key is communicated from the sender to the receiver, simply stating, "The sender sends it to the receiver". The need to securely send this private key is one of the major drawbacks of symmetric key cryptosystems, making them impractical in many situations. Finally, the conversion of the binary encrypted text by the authors into a DNA sequence and then into a complement of DNA provides no security benefit and only adds unnecessary computations to the encryption and decryption processes.

2 Preliminaries

2.1 DNA in cryptography

DNA is the genetic makeup of nearly all living things, ranging in complexity from tiny viruses to intricate humans. With two strands running counter-parallel, DNA has a double helix structure. Nucleotides are tiny, continuous polymers that makeup DNA. A nitrogenous base, a sugar with five carbons, and a phosphate group are the three parts of each nucleotide. There are four distinct nucleotides, depending on the sort of nitrogenous base they contain. Adenine (A), Cytosine (C), Thiamine (T), and Guanine (G) are the four distinct bases. All organisms' vast and intricate information is stored in DNA using simply this basis.

By creating hydrogen bonds with one another to keep the two strands of DNA connected, the structure of DNA strands. While C and G create bonds with one another, A makes a hydrogen bond with T [32]. Before 1994, it was thought that

DNA exclusively contained biological information. However, Adleman's solution to the NP-complete Hamiltonian path problem of seven vertices disproved this notion [2]. Since then, DNA has been also employed as a tool for computation [24]. The four letters A, C, T, and G make up the DNA language, which is used in DNA computing.

DNA's capacity for computation is now applied to cryptography as well. DNA cryptography is a promising area that, if used correctly, might put other areas of cryptography in a much tougher competitive position [11]. One of the key benefits of DNA in cryptography is its potential to generate large volumes of truly random numbers quickly for practical needs [25, 29].

2.2 Rabin cryptosystem

In 1979, Rabin offered an alternative to RSA with public-key exponent 2 [17]. Utilizing this exponent 2 has the following benefits over using greater exponents:

(i) A lighter computational burden.

(ii) It is known that factoring the RSA modulus N (we will refer to this problem as FACTOR) is equivalent to computing square roots $\mod N$, i.e., solving (1) below [15]. On the other hand, for the RSA cryptosystem, while the factorization of N is sufficient to break it, it is not known whether it is also necessary. Hence, the security of the Rabin cryptosystem is more closely related to the integer factorization problem than RSA.

However, the Rabin cryptosystem also has some disadvantages. The main one is the fact that square roots $\mod N$ are not unique. When decrypting a ciphertext, we need to find a way to pick the correct plaintext among the four possible candidates. Moreover, it is vulnerable to the chosen plain-text attack.

Definition 1. *[17] Let $S = \mathbb{N}$ or $S = \{pq;\, p, q \text{ primes} \equiv 3 \mod 4\}$. The computational problem SQRT-MOD-N is: Given $N \in S$ and $y \in \mathbb{Z}/N\mathbb{Z}$ to output if y is not a square modulo N, or a solution x to $x^2 \equiv y \pmod N$.*

Lemma 1. *[17] SQRT-MOD- N is equivalent to FACTOR (is the problem of decomposing a composite number N into its prime factors.)*

The key generation of the Rabin algorithm acts as follows,

1) Generate two very large prime numbers, p and q, that satisfy

$$p \neq q \qquad\qquad p \equiv q \equiv 3\ (\mathrm{mod} 4).$$

2) Calculate the value of $n = pq$.

3) Publish n as public key and save p and q as private keys.

The encryption of the Rabin algorithm follows the following steps

1) Obtain the public key n.

2) Convert the clear text m to ASCII. Then, transform it to a binary sequence, double the binary sequence by duplicating it, and return the binary value to decimal.

3) Encrypt m by squaring,
$$C = m^2 \mod n. \tag{1}$$

4) Send C to the recipient.

Now, with Rabin's decryption method, there are four potential outcomes, but only one of them is the correct plaintext. The steps of Rabin algorithm decryption are resumed as follows.

1) Take C from the sender.

2) Use the extended Euclidean algorithm to find y_p and y_q. (Here, $\gcd(p, q) = 1$, and $py_p + qy_q = 1$.)

3) Calculate m_p and m_q as
$$m_p = C^{\frac{p+1}{4}} \bmod p \quad \text{and} \quad m_q = C^{\frac{q+1}{4}} \bmod q.$$

4) Calculate the values of the variables v and w as
$$v = y_p \times p \times m_q \quad \text{and} \quad w = y_q \times q \times m_p.$$

5) Calculate the values of r, s, t and u using Chinese Remainder Theorem (CRT) as
$$\begin{aligned} r &= (v + w) \bmod n, \\ s &= (v - w) \bmod n, \\ t &= (-v + w) \bmod n, \\ u &= (-v - w) \bmod n. \end{aligned}$$

6) Convert r, s, t, and u into binary.

7) Determine the original message from the values of r, s, t, and u.

3 Main results: The hybrid algorithm

In this section, we explain our novel hybrid cryptosystem. The proposed algorithm consists of a combination of Rabin's cryptosystem, binary one-time pad, DNA cryptography, and a Feistel-inspired scheme for the encryption of plaintexts. This algorithm uses both symmetric and asymmetric keys. It follows the steps below.

3.1 The key generation step

In this step, the sender generates a random DNA key, with size depending eventually on one of the plaintext messages, as

$$\text{size(DNA key)} = 13 \times \text{size(plaintext)}.$$

Convert the random DNA key to a binary sequence via Table 1 (called the random binary key) and send it to the receiver using the method described in [5].

Table 1: Binary codes [3]

Nucleotide	A	C	G	T
Code	00	01	10	11

.

At the same time, the receiver generates a Rabin key and sends only the public key to the sender.

3.2 The encryption algorithm step

In the encryption process, the sender has the plaintext, the DNA key, and the Rabin public key. The sender's encryption algorithm works as follows.

1. To establish the precise output when using the Rabin cryptosystem to decrypt, we perform a sort of disambiguation diagram in this stage. The sender and receiver choose a secret spy to be inserted before each character of the plaintext before starting the encryption process. The encryption is applied on the new plaintext.

2. Convert the new form of the message via the ASCII table.

3. Concatenate the numerical values two-by-two.

4. Apply the Rabin encryption algorithm to each ASCII value and compute the Rabin cipher.

5. Convert each digit of the Rabin cipher to a binary plaintext of 26 digits.

6. Perform XOR operation between binary cipher and random binary key, called XOR cipher.

7. Reorder every 26 digits by Feistel-inspired structure, where it is divided into two equal parts (l_0, r_0), and the XOR operation is performed between the two parts, with the left part l_1 containing the value of r_0 as is, and the XOR result moving towards the second part r_1. The pieces r_1 and l_1 have now been concatenated.

The sender sends this ciphertext to the receiver.

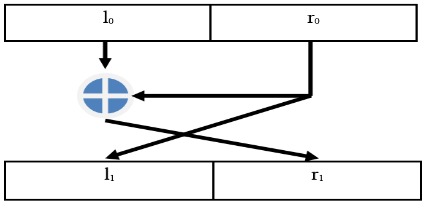

Figure 1: Feistel-inspired scheme for reordering binary plaintext.

3.3 The decryption algorithm step

The receiver's algorithm is just the opposite of the sender's algorithm. It works as follows.

1. Apply a concept of diffusion by using two Feistel-inspired schemes for each of the 26 binary digits.
2. Perform XOR operation between the binary key and Feistel decryption key.
3. Convert every 26 binary cipher to the corresponding decimal number.
4. Decrypt each decimal number with the Rabin decryption key.
5. Select the square root that starts with the ASCII character corresponding to the spy that the sender and receiver chose at the beginning from the four distinct square roots that were obtained.
6. Remove the ASCII number associated with each selected square root's spy.
7. Convert each ASCII to its corresponding character.

4 Some illustrative implementations

In this section, two implementations are conducted in order to compare our hybrid algorithm with the classical case and the Rabin cryptosystem.

4.1 Example 1: A Rabin's cryptosystem case

4.1.1 The key generation step

1. Consider the prime numbers $p = 167$ and $q = 127$. It is easily verified that
$$167 \equiv 3 \bmod 4 \quad \text{and} \quad 127 \equiv 3 \bmod 4.$$
2. Compute $n = pq = 21209$.
3. Publish n as public key, and save p and q as private keys.

4.1.2 The encryption algorithm

1. Obtain the public key, $n = 21209$.

2. Let the plaintext $m = H = 72$. Converted to a binary sequence, it becomes

$$m = 1001000_2.$$

Duplicating the binary sequence we get

$$m = 1001000 \mid 1001000_2.$$

Reconverting to the decimal yields

$$m = 9288_{10}.$$

3. Encrypting the plaintext m by the Rabin cryptosystem gives

$$C = m^2 \mod n = 9288^2 \mod 21209 = 9941.$$

4. Send C to the recipient.

4.1.3 The decryption algorithm

1. The ciphertext received is $C = 9941$.

2. Using the extended Euclidean algorithm, we find $y_p = 54$ and $y_q = -71$.

3. We get the values $m_p = 64$ and $m_q = 17$.

4. The values of the variables v and w are $v = 153306$ and $w = -577088$.

5 and 6. Next, we obtain

$$\begin{aligned} r &= 398_{10} = 110001110_2, \\ s &= 11921_{10} = 10111010010001_2, \\ t &= 9288_{10} = 10010001001000_2, \\ u &= 20811_{10} = 101000101001011_2. \end{aligned}$$

7. The only case yielding the original message is

$$t = 9288_{10} = 1001000 \mid 1001000_2 \text{ (left=right)}.$$

Consequently,

$$m = 1001000_2 = 72_{10} = H.$$

4.2 Example 2: A case of the new hybrid cryptosystem

Consider a sender that wants to communicate a piece of information in the form of a secret text such as “SUCCESS” to a receiver. The size of the plaintext is 7. We will apply our hybrid cryptosystem to encrypt and decrypt this piece of information.

4.3 Step 1: Generation of key pairs

A random DNA key is generated, whose size is equal to $13 \times 7 = 91$. For example,

G T T T G G G G T T C C A A T C C A T T T A G A T C A C C
C G C C G G G G T T G C C T T A C G A C A G A A T T T A
T A A A C T G C A G C T T T A C A T T A C T A A T C C G C
T T C T G

This DNA key is converted to a binary sequence via Table 1. We get the sequence

1 0 1 1 1 1 1 1 1 0 1 0 1 0 1 0 1 1 1 1 0 1 0 1 0 0 0 0 1 1 0 1 0 1 0 0 1
1 1 1 1 1 0 0 1 0 0 0 1 1 0 1 0 0 0 1 0 1 0 1 1 0 0 1 0 1 1 0 1 0 1 0 1 0
1 1 1 1 1 0 0 1 0 1 1 1 1 1 0 0 0 1 1 0 0 0 0 1 0 0 1 0 0 0 0 0 1 1 1 1 1
1 0 0 1 1 0 0 0 0 0 0 0 1 1 1 1 0 0 1 0 0 1 0 0 1 1 1 1 1 1 1 0 0 0 1 0 0
1 1 1 1 0 0 0 1 1 1 0 0 0 0 1 1 0 1 0 1 1 0 0 1 1 1 1 1 0 1 1 1 1 0

The sender sends this key to the receiver using the method described in [5]. The receiver generates the Rabin key. Let the private key be $(p, q) = (6911, 6947)$, and the public key $n = 48010717$. The receiver sends just $n = 48010717$ as the public key to the sender.

4.4 Step 2: The encryption

1. The sender chooses a spy (for example "$*$") and inserts it before each letter in the plaintext "SUCCESS" to get

$$* \mathrm{S} * \mathrm{U} * \mathrm{C} * \mathrm{C} * \mathrm{E} * \mathrm{S} * \mathrm{S}$$

2. This text is converted to the corresponding ASCII representation as

42 83 42 117 42 99 42 99 42 101 42 115 42 115

3. Concatenating these numerical values two-by-two, we obtain

4283 42117 4299 4299 42101 42115 42115.

4. Applying the Rabin encryption, we get the ciphertext as

18344089 45455877 18481401 18481401 44108389 45287413 45287413.

5. Now every component of the Rabin ciphertext is converted to a 26-bit binary sequence to produce a binary cipher as

0 1 0 0 0 1 0 1 1 1 1 1 1 0 1 0 0 0 1 0 0 1 1 0 0 1 1 0 1 0 1 1 0 1 0
1 1 0 0 1 1 0 1 0 0 0 0 0 0 1 0 1 0 1 0 0 0 1 1 0 1 0 0 1 0 0 0 0 0 0
1 1 1 1 1 0 0 1 0 1 0 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 1 1
0 1 0 1 0 0 0 0 1 0 0 0 0 1 0 1 0 0 1 1 0 0 1 0 1 1 0 1 0 1 1 0 0 1 1
0 0 0 0 0 1 1 1 1 1 1 1 0 1 0 1 1 0 1 0 1 1 0 0 1 1 0 0 0 0 0 1 1 1 1
1 1 1 0 1 0 1

6. Now XOR operation is performed between binary cipher and binary key to yield the sequence

   1 1 1 1 1 0 1 0 0 1 0 1 0 0 0 0 1 1 0 1 0 0 1 1 0 1 1 0 0 1 1 0 0 0 0
   1 0 1 1 0 0 1 1 0 1 0 0 0 1 0 0 0 0 1 0 1 0 0 1 1 0 0 0 1 0 1 1 0 1 0
   0 1 0 1 0 1 1 0 1 1 0 1 0 0 0 1 0 1 0 0 0 1 1 0 0 0 1 0 1 1 0 0 0 1 1
   0 0 1 0 1 1 1 0 1 1 1 0 0 1 0 1 0 0 0 0 1 1 1 0 0 1 0 0 0 1 0 1 1 0 0
   1 1 0 0 0 0 1 1 0 0 0 0 0 1 0 0 0 1 1 0 1 1 1 1 1 0 0 1 1 0 0 0 0 0 0
   0 1 0 1 0 1 1

7. Reordering every 26 digits by Feistel-inspired structure, we obtain

   0 0 0 1 1 0 1 0 0 1 1 0 1 1 1 1 0 0 0 0 0 0 0 1 1 1 0 0 1 1 0 1 0 0 0
   1 0 0 0 1 0 1 0 1 1 0 0 0 0 0 1 1 1 1 0 1 0 0 1 0 1 0 1 1 0 1 0 0 0 0
   0 0 1 1 0 1 0 0 1 1 0 0 0 1 0 1 1 0 0 0 1 0 0 0 1 0 1 0 0 1 1 0 0 1 0
   1 0 1 0 0 0 0 1 1 1 0 0 1 1 0 0 0 1 1 1 1 0 0 1 0 0 0 0 1 1 0 0 0 0 0
   1 0 0 1 0 0 1 0 0 1 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 1 0 1 1 0 1 1 0 1 1
   1 0 1 1 0 0 0

The sender sends this ciphertext to the receiver.
The receiver receives the ciphertext and runs the receiver's side algorithm.

4.5 Step 3: The decryption

First receive the ciphertext

0 0 0 1 1 0 1 0 0 1 1 0 1 1 1 1 0 0 0 0 0 0 0 1 1 1 0 0 1 1 0 1 0 0 0 1 0
0 0 1 0 1 0 1 1 0 0 0 0 0 1 1 1 1 0 1 0 0 1 0 1 0 1 1 0 1 0 0 0 0 0 0 1 1
0 1 0 0 1 1 0 0 0 1 0 1 1 0 0 0 1 0 0 0 1 0 1 0 0 1 1 0 0 1 0 1 0 1 0 0 0
0 1 1 1 0 0 1 1 0 0 0 1 1 1 1 0 0 1 0 0 0 0 1 1 0 0 0 0 0 1 0 0 1 0 0 1 0
0 1 1 0 0 0 1 0 0 0 0 0 0 0 0 1 0 1 0 1 1 0 1 1 0 1 1 1 0 1 1 0 0 0

1. Applying the diffusion by using two Feistel-inspired schemes, we get the binary key

   1 1 1 1 1 0 1 0 0 1 0 1 0 0 0 0 1 1 0 1 0 0 1 1 0 1 1 0 0 1 1 0 0 0 0
   1 0 1 1 0 0 1 1 0 1 0 0 0 1 0 0 0 0 1 0 1 0 0 1 1 0 0 0 1 0 1 1 0 1 0
   0 1 0 1 0 1 1 0 1 1 0 1 0 0 0 1 0 1 0 0 0 1 1 0 0 0 1 0 1 1 0 0 0 1 1
   0 0 1 0 1 1 1 0 1 1 1 0 0 1 0 1 0 0 0 0 1 1 1 0 0 1 0 0 0 1 0 1 1 0 0
   1 1 0 0 0 0 1 1 0 0 0 0 0 1 0 0 0 1 1 0 1 1 1 1 1 0 0 1 1 0 0 0 0 0 0
   0 1 0 1 0 1 1

2. Performing XOR operation between the binary key and the Feistel decryption, we get the sequence

   0 1 0 0 0 1 0 1 1 1 1 1 1 0 1 0 0 0 1 0 0 1 1 0 0 1 1 0 1 0 1 1 0 1 0
   1 1 0 0 1 1 0 1 0 0 0 0 0 0 1 0 1 0 1 0 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0
   1 1 1 1 1 0 0 1 0 1 0 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 1 1

0 1 0 1 0 0 0 0 1 0 0 0 0 1 0 1 0 0 1 1 0 0 1 0 1 1 0 1 0 1 1 0 0 1 1
0 0 0 0 0 1 1 1 1 1 1 1 0 1 0 1 1 0 1 0 1 1 0 0 1 1 0 0 0 0 0 1 1 1 1
1 1 1 0 1 0 1

3. Next, converting every 26 binary cipher to the corresponding decimal number, we get

   18344089 45455877 18481401 18481401 44108389 45287413 45287413

4. Decrypting each decimal number by the Rabin decryption key, we obtain

1018545	4283	46992172	48006434
47968600	8210919	42117	39799798
4299	6346910	48006418	41663807
4299	6346910	48006418	41663807
13539284	47968616	34471433	42101
47968602	2875625	42115	45135092
47968602	2875625	42115	45135092

5. Choosing the segments starting with 42 in each row, we obtain the sequence

   4283 42117 4299 4299 42101 42115 42115

6. Next, by removing the digit 42 from each segment, we obtain the text

   83 117 99 99 101 115 115

7. Finally, converting each ASCII to its corresponding character, we get our original text "SUCCESS".

5 Method Analysis

Despite many differences between DNA cryptography and conventional cryptography, both satisfy the same cryptographic requirements. The security of the proposed encryption is dependent on three levels: The first level depends on the DNA OTP key. It is well known that the OTP is unconditionally secure if the key is truly random, has the specified length, is never completely or partially reused, and is kept in complete secrecy. Given these conditions, no adversary can obtain it. If a brute force attack is used to try to break the algorithm, the chances of finding the right combination are 1 in $4^{(m\times 26/2)}$, where m is the message length. The likelihood of obtaining the correct combination for a message of length 10 is therefore 1 in $4^{130} = 2^{260}$ combinations. It is a very remote possibility. The second level of security is the computational difficulty of factoring an integer of the Rabin key-sharing system. The Rabin cryptosystem benefits from the fact that the underlying problem for security has been shown to be equivalent to the integer factorization problem [17] which currently is not known to be the case for the

RSA problem (because to break RSA it suffificent to either solve the integer factorization problem, or efficiently compute e^{th} roots $\mod n$), making the Rabin cryptosystem potentially more secure than the RSA [30]. The Rabin encryption method is more efficient because it only requires computing squares $\mod n$ whereas RSA requires computing e^{th} powers $\mod n$ [6]. The Rabin cryptosystem is secure against specific plaintext attacks [30] but it can be cracked using chosen ciphertext attack, allowing the attacker to obtain the private key. The main flaw in the Rabin cryptosystem, which has kept it from being widely adopted in practice, is that decoding yields three false results in addition to the true one, making it necessary to guess the true result [6]. Guessing is usually not hard if the plaintext represents a text message, however, if the plaintext is meant to represent a numerical value, the issue becomes one that needs to be solved via a disambiguation method. To solve this issue, redundancy in the message is required (three solutions to the problem are described in [17]), or alternatively, extra bits must be sent, one can choose plaintexts with unique structures or add padding. The third level of security is provided by the inclusion of a Feistel-inspired structure in our suggested algorithm that improves the algorithm in terms of its security parameter [21]. We employ the Feistel-inspired structure for each of the 26 digits. This makes things more complicated by adding some confusion and diffusion, which prevents the adversary from using any kind of brute force attack.

6 Conclusion

Applications of DNA cryptography are expanding quickly. The main approach of DNA cryptography is based on biochemical techniques for data encryption based on DNA sequences, which leads to more effective new algorithms by integrating the characteristics of both biological and conventional cryptography [33]. The goals of DNA cryptography algorithms are to increase data transmission security to the highest level possible and to decrease the computational complexity of encryption and decryption. In this work, we propose a new encryption technique based on DNA, Rabin cryptosystem, OTP, and Feistel-inspired structure to ensure higher security. Due to DNA's random nature [9], the use of DNA sequences in OTP makes the algorithm strong enough to fend off attacks. Given that DNA sequences are truly random and they are available on millions of websites, it is practically impossible to determine the random sequence utilized in encryption. The use of the Rabin cryptosystem potentially achieves higher security than RSA due to its closer connection to the integer factorization problem. The idea of replacing the binary sequences based on the Feistel structure also makes the proposed algorithm safer. Key sharing is a big disadvantage if the random binary key needs to be shared between the sender and the receiver but with the use of the method given in [5], we have overcome this problem. Since the original data is never communicated in an open manner, the suggested method is more secure for data transfer over the Internet.

References

[1] Abd El-Latif, E.I. and Moussa, M.I., *Information hiding using artificial DNA sequences based on Gaussian kernel function,* J. Inf. Optim. Sci. **40** (2019), no. 6, 1181-1194.

[2] Adleman, L.M., *Molecular computation of solutions to combinatorial problems,* Science **266** (1994), no. 5187, 1021-1024.

[3] Anwar, T., Kumar, A. and Paul, S., *DNA cryptography based on symmetric key exchange,* International Journal of Engineering and Technology **7**, no. 3, 938-950.

[4] Begum, M., Ferdush, J. and Moazzam, G.M., *A hybrid cryptosystem using DNA, OTP and RSA,* International Journal of Computer Applications **172** (2017), no. 8, 30-33.

[5] Bouroubi, S., Charchali, F. and Benyehia, N., *Set partitions solution for sharing secret key,* Rom. J. Math. Comput. Sci. **9** (2019), 78–86.

[6] Buchmann, J., *Introduction to cryptography*, Texts in Mathematics **335**, Springer, New York, 2004.

[7] Budiman, M.A., Rachmawati, D. and Utami, R., *The cryptanalysis of the Rabin public key algorithm using the Fermat factorization method,* In Journal of Physics: Conference Series, IOP Publishing, **1235** (2019), no. 1, 012084.

[8] Cahyono, J., Adzkiya, D. and Davvaz, B., *A cryptographic algorithm using wavelet transforms over max-plus algebra,* Journal of King Saud University-Computer and Information Sciences **34**, (2022), no. 3, 627-635.

[9] Chen, J., *A DNA-based, biomolecular cryptography design,* IEEE International Symposium on Circuits and Systems (ISCAS), **3** (2003), III-III.

[10] Chouhan, D.S. and Mahajan, R.P., *An architectural framework for encryption and generation of digital signature using DNA cryptography,* International Conference on Computing for Sustainable Global Development (INDIACom) IEEE, (2014), 743-748.

[11] Cui, G., Li, C., Li, H. and Li, X., *DNA computing and its application to information security field,* Fifth international conference on natural computation, IEEE, **6** (2009), 148-152.

[12] Delfs, H., Knebl, H. and Knebl, H., *Introduction to cryptography*, **2**, Springer, Heidelberg, 2002.

[13] Diffie, W. and Hellman, M.E., *New directions in cryptography,* IEEE Transactions on Information Theory **22** (1976), no. 6, 644-654.

[14] Elhamzi, W., Jallouli, M. and Bouteraa, Y., *High efficiency crypto-watermarking system based on Clifford-multiwavelet for 3D meshes security,* Computers, Materials and Continua, **73** (2022), no. 2, 4329–4347.

[15] Elia, M., Piva, M. and Schipani, D., *The Rabin cryptosystem revisited,* Applicable Algebra in Engineering, Communication and Computing, **26** (2015), 251-275.

[16] Enriquez, M. and Arboleda, E., *Enhanced hybrid algorithm of secure and fast chaos-based, AES, RSA and ElGamal cryptosystems,* Indian Journal of Science and Technology, **10** (2017), no. 27, 1-14.

[17] Galbraith, S.D., *Mathematics of public key cryptography,* Cambridge University Press, 2012, 485-512.

[18] Goswami, D., Rahman, N., Biswas, J., Koul, A., Tamang, R. L. and Bhattacharjee, D.A., *A discrete wavelet transform based cryptographic algorithm,* IJCSNS **11** (2011), no. 4, 178.

[19] Hirabayashi, M., Kojima, H. and Oiwa, K., *Design of true random one-time pads in DNA XOR cryptosystem,* In Natural Computing: 4th International Workshop on Natural Computing Himeji, Japan, Proceedings, Springer Japan, 2009, 174-183.

[20] Katz, J. and Lindell, Y., *Introduction to modern cryptography: principles and protocols*, Chapman and hall/CRC Press (Cryptography and Network Security Series) 2007.

[21] Kaundal, A.K. and Verma, A.K., *Extending feistel structure to DNA cryptography,* Journal of Discrete Mathematical Sciences and Cryptography **18** (2015), no. 4, 349-362.

[22] Kim, C.H., Kwon, S. and Hong, C.P., *FPGA implementation of high performance elliptic curve cryptographic processor over GF (2163),* Journal of Systems Architecture, **54** (2008), no. 10, 893-900.

[23] Lai, X., Lu, M., Qin, L., Han, J. and Fang, X. *Asymmetric encryption and signature method with DNA technology,* Science China Information Sciences **53** (2010), 506-514.

[24] Leier, A., Richter, C., Banzhaf, W. and Rauhe, H., *Cryptography with DNA binary strands,* Biosystems **57** (2000), no. 1, 13-22.

[25] Meiser, L.C., Koch, J., Antkowiak, P.L., Stark, W. J., Heckel, R. and Grass, R.N., *DNA synthesis for true random number generation,* Nature communications **11** (2020), no. 1, Article ID 5869.

[26] Rabin, M.O., *Digitalized signatures and public-key functions as intractable as factorization*, Massachusetts Institute of Technology, Cambridge 1979.

[27] Reddy, M.I.S. and Kumar, A.S., *Secured data transmission using wavelet based steganography and cryptography by using AES algorithm,* Procedia Computer Science **85** (2016), 62-69.

[28] Sattar, K.A., Haider, T., Hayat, U. and Bustamante, M.D., *An efficient and secure cryptographic algorithm using elliptic curves and Max-plus algebra-based wavelet transform,* Applied Sciences **13** (2023), no. 14, Article ID 8385.

[29] Siddaramappa, V. and Ramesh, K.B., *True random number generation based on DNA molecule genetic information (DNA-TRNG),* Cryptology ePrint Archive (2020).

[30] Srivastava, A. K. and Mathur, A., *The rabin cryptosystem and analysis in measure of Chinese reminder theorem,* International Journal of Scientific and Research Publications **3** (2013), no. 6, 1-4.

[31] Varma, P.S. and Raju, K.G., *Cryptography based on DNA using random key generation scheme,* International Journal of Science Engineering and Advance Technology (IJSEAT) **2** (2014), no. 7, 168-175.

[32] Xiao, G., Lu, M., Qin, L. and Lai, X., *New field of cryptography: DNA cryptography* Chinese Science Bulletin **51** (2006), 1413-1420.

[33] Yang, J., Ma, J., Liu, S. and Zhang, C., *A molecular cryptography model based on structures of DNA self-assembly,* Chinese Science Bulletin **59** (2014), 1192-1198.