# Implementation, Performance and Security Analysis for CryptoBin Algorithm

Ahmed H. Eltengy[1], Samaa M. Shohieb[2], Ali E. Takieldeen[3], and Mohamed S. Ksasy[4]
*(Corresponding author: Ahmed H. Eltengy)*

Faculty of Engineering, Alexandria University[1]
22 El-Guish Road, El-Shatby, Alexandria 21526, Egypt
(Email: tengy_fox@yahoo.com)
Faculty of Computers and Information, Mansoura University[2]
Mansoura, Egypt
Faculty of Engineering, Delta University, Mansoura, Egypt[3]
Faculty of Engineering, Mansoura University, Mansoura, Egypt[4]
Mansoura, Egypt
*(Received Aug. 23, 2018; Revised and Accepted Nov. 13, 2018; First Online Jan. 22, 2019)*

## Abstract

With the fast evolution of digital data exchange, security is the main concern in today's world. It is important to secure data from uncertified access, security information becomes essential in information and data storage, and transmission over open networks such as the Internet. The traditional algorithms face some drawbacks of little key-space and poor security. This paper proposes a new way to encrypt data on the basis of the binary form which is considered the simplest form of data that is consisted of zero and one. This new system converts the target message into zero and one and then swap the bit value from one to zero and from zero to one by using mathematical equations built on the truth table in which the secret key and the target message are the main elements. This algorithm is characterized by a secret key that has an unlimited length and a sub-secret key added to the system. The use of the sub-secret key helps to generate a different encrypted message every time even if the same secret key, the sub-secret key, and the same plain-text are used, which increases the confidentiality and strength of the system. This system provides all the demands of secrecy and strength to confront the intruders with high efficiency and has high-security analysis such as key space analysis, statistical analysis. For example, if the secret key is chosen as 1 MB length that means a number of trials equal to 28388608 to estimate it, which is considered very large to be adequate to safeguard information and data that is encrypted by the proposed encryption system against any attacks. Therefore, the proposed system can be used to secure any software applications.

*Keywords: Encryption; Decryption; Information Security; cryptography*

## 1 Introduction

The science of cryptography is the science of coverage and verification of information. Often referred to as "the study of secret" when data exchanged over the Internet, networks or other media. It's the technique of protecting data and information from non-authorized access [7] by transforming it into a non-readable format, called ciphertext.

Only those who have the secret key of the encryption system can decrypt the encrypted message and return it to a readable format. It includes algorithms, protocols, and methodologies to secure prevent or delay unapproved access to sensitive information and to enable verifiability of every component in the communication. A cryptographic algorithm, which is also known as a cipher, could be the mathematical function or equation used for encryption and decryption [3].

Generally, data decryption process is similar to the data encryption process, but in a reversed way. Encryption/Decryption protects data and information from being hacked by the hacker [4]. Encryption/Decryption is a security system where cipher or encryption algorithms are executed together with a secret key to encrypt/decrypt data so that they are unreadable in the event that they are intercepted [6].

With a dramatic increase in the number of Internet users around the world, the need to protect data, information, and multimedia on the Internet has become a high priority. Most operations in governments, military installations, financial institutions, hospitals, and private companies deal heavily with data that is in the form of an image or multiple media, most encryption algorithms today are based on text-only data [19]. Encryption of Digital Image is a beanch of software encryption and has

become very important to prevent and thwart any attack on them to obtain information without prior authorization. Min-Shiang Hwang [11] proposed a new secure cryptographic system built on the Merkle-Hellman public key cryptographic system (knapsack public-key). This method proposes a new Permutation Combination Algorithm.

Gilhorta and Singh [18] proposed the plaintext is converted to a floating number in a range from 0 to 1 and then this floating number converted to binary code and by using a secret key it is converted to encrypted binary code. Animesh Hazra et al. [8] present a brief review of using DNA as a method of cryptography in real time implementation. Li-Chin Huang and Min-Shiang Hwang [9] proposed a study of data hiding in medical images.

Mohamed Rasslan et al. [15] presented a public model to execute any cryptographic algorithm by way of a parallel- pipelined design. Ali E. Takieldeen et al. [20] suggested a method of cryptography which uses the image as a public key and random integers as a private key which is used to permutate the image. Lihua Liu et al. [13] designed a cryptographic system of private broadcast encryption to encrypt a plaintext or a message for multi recipients and hide the recipients identities. Cheng-Chi Lee and Min-Shiang Hwang [12] designed a new convertible authenticated encryption scheme built on the ElGamal cryptosystem. Said Bouchkaren and Saiida Lazaar [2] proposed some tests concentrate on the randomness of tests and on differential cryptanalysis Managed on the CAES (Cellular automata Encryption System).

In this paper, a modified cryptographic algorithm system based on binary codes (0,1) is designed by using mathematical equations [5]. The main idea of this algorithm is converting 0-bit value to 1-bit value and 1-bit value to 0-bit value by using a mathematical equation depends on the bit values of secret key and target message by using logic functions. The target message is divided into bytes each of which is composed of 8 bits. The secret key length is modified to be equal to the target message length. Changing the value of the bits depends on a truth table in which the secret key and target message act as main elements. Each person who receives the message has their own sub-secret key. This key is composed of two parts; the first part is a value that points to where the first place of a dummy bit is added to each byte in the encrypted message, and the second part is also a value that points to where the second place of a dummy bit is added to each byte in the encrypted message.

The values of these two dummy bits are generated randomly by the system. Finally, the system generates a different encrypted message every time even if the same secret key, the sub-secret key, and the same plaintext are used several times because of each byte contains two bits have random values. The decryption procedure is similar to the encryption procedure in processing but in reverse order starts by removing the previously generated random dummy bits and then decrypting the message. The proposed algorithm system can regenerate the original binary data byte with no loss or lack of data during and after the encryption or decryption process. By using unlimited secret keys length, and a sub-secret key is owned by each person who receives the encrypted message, the algorithm is more secure and it's hard to guess the key value or be attacked.

The proposed algorithm has been tested and compared with other recent algorithm and it was fast, simple and flexible enough. Validation of the new algorithm security requirements have been applied and it has been suitable for using it in many software applications.
The second section demonstrates the proposed algorithm (CryptoBin), the third section discusses the architecture of the algorithm system, the proves of the strength, the performance and security analysis for CryptoBin based system and its results. Finally, the conclusion will be introduced.

## 2 Proposed Work

The cryptographic algorithm system for binary codes which discussed and published after many tests and trials to attack it found that it has some drawbacks and weakness in the secret key system, and should be improved. This article will perform a study of the CryptoBin algorithm (ours) and try to explain its strength and resistance to attacks. So that a new method for secure communication of information and multimedia encryption proposed here. This technique contains advantages of both multimedia (Audio, Image, and video) cryptography and normal encryption data. This article is used to achieve and solve the problem of the weakness and drawbacks of the former system that mentioned before and it strengthens the secret key to be difficult to break. This cryptographic algorithm system is called CryptoBin which deals with Binary codes (0,1) bits.

The proposed secret key is a binary number which characterized by an unlimited size of bits, the bits can be less, equal or greater than the target message (plain text). The algorithm system compares the bit value of the secret key with the bit value of the target message and generates a new encrypted message that has an equal length of the target message. The algorithm system compares the bit value of the secret key and the target message using logical equations based on a given truth table, resulting in the encrypted message. For example, if the bit value of the secret key is equal to "1", the bit value of the target message will change from "1" to "0" or "0" to "1", else if the bit value of the secret key is "0", the bit value of the target message will not change and be as it "0" is "0" and "1" is "1".

### 2.1 Architecture of The Algorithm System

The CryptoBin algorithm consists of a secret key, plain message, and a truth table. The secret key and the plain

message are binary numbers, and the truth table controls the output bit's value (an encrypted bit).

**Encryption System.**

For example:

plain text = "Hello World"

binary input = "010010000110010101101100011011000111100100000010101110110111011100100110110001100100" secret key = "723" in decimal form

secret key = "1011010011" in binary form

To obtain a modified secret key to be equal to the number of bits of the target message it should be repeated.

Modifiedsecretkey="1011010011101101001110110100111011010011101101001110110100111011010011101101001110100"

sub-secret key = "010110"

By using the sub-secret key, the system will generate random values of two dummy bits and add them to the encrypted message. The first part of the sub-secret key points to the position of first dummy bit will be added to each byte of the encrypted message, the second part of the sub-secret key points to the position of the second dummy bit will be added to each byte of the encrypted message. In this case, the first dummy bit position will be the third bit, the second dummy bit position is the seventh bit with random values.

Truth Table =

| Key | Msg | Enc Msg |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

The common idea of Truth Table that if the key's bit value is "0" then the encrypted bit value will be unchanged, else if the key's bit value is "1" as shown in Figure 1 then the encrypted bit value will be changed from "1" to "0" or "0" to "1".

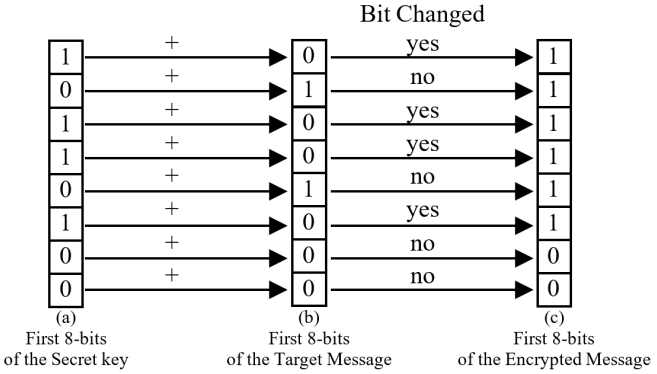



Figure 1: One-byte encryption

**The Plain Message**

| Bit no. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | |
| 13 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 24 |
| 25 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 36 |
| 37 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 48 |
| 49 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 60 |
| 61 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 72 |
| 73 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 84 |
| 85 | 0 | 1 | 0 | 0 | 88 | | | | | | | | |

**The Secret Key**

| Bit no. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | |
| 13 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 24 |
| 25 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 36 |
| 37 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 48 |
| 49 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 60 |
| 61 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 72 |
| 73 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 84 |
| 85 | 0 | 1 | 0 | 0 | 88 | | | | | | | | |

**The Encrypted Message**

| Bit no. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | |
| 13 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 24 |
| 25 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 36 |
| 37 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 48 |
| 49 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 60 |
| 61 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 72 |
| 73 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 84 |
| 85 | 0 | 0 | 0 | 0 | 88 | | | | | | | | |

**The sub-secret key is "010110"**

**Encrypted Message After adding Dummy Bits**

| Bit no. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | |
| 13 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 24 |
| 25 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 36 |
| 37 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 48 |
| 49 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 60 |
| 61 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 72 |
| 73 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 84 |
| 85 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 96 |
| 97 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 108 |
| 109 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 117 | | | |

EncryptedBinary="1111110100010010100000110110110100110000101010111100110010100101011111010101001010000011011001001011111111101100100"

**Decryption System.**

For Decrypting the same example:

EncryptedBinary="11111100100010000101011100100010101111001001010010111010010101000011110010111111111010000"
secret key = "723" in decimal form
Secret key = "1011010011" in binary form
sub-secret key = "010110"

By using the sub-secret key, the system will search for the two dummy bits that have already been added to the encrypted message, then remove them from that message. The first part of the sub-secret key points to the position of the first dummy, the second part of the sub-secret key points to the position of the second dummy bit that has been added to each byte of the encrypted message. In this case, the first dummy bit position will be the third bit, the second dummy bit position is the seventh bit.

To obtain a modified secret key to be equal to the number of bits of the target message it should be repeated.

Modifiedsecretkey="10110100111011010011101101001110111010011101101001110111010011101101001110111010011101101001110110100"

By using the same truth table in the decryption process the bit value of 0 changed to 1 and 1 to 0 according to the rules which added in the truth table and Figure 2 shows a sample of a ciphered 8-bits changed to decrypted 8-bits.

Truth Table =

| Key | Msg | Enc Msg |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

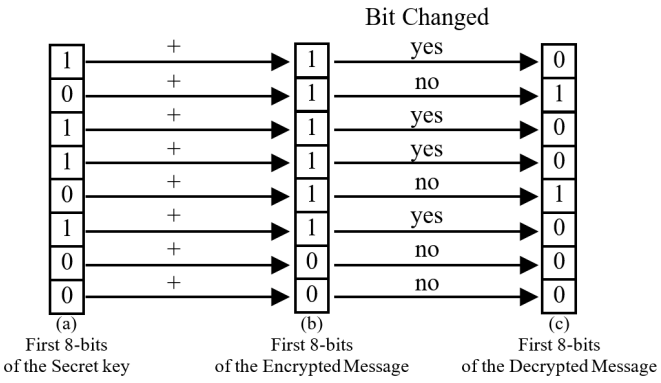



Figure 2: One-byte decryption

**The Encrypted Message Including Dummy Bits**

| Bit no. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | |
| 13 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 24 |
| 25 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 36 |
| 37 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 48 |
| 49 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 60 |
| 61 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 72 |
| 73 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 84 |
| 85 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 96 |
| 97 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 108 |
| 109 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 117 | | | |

**The sub-secret key is "010110"**
**The Encrypted Message After Removing Dummy Bits**

| Bit no. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | |
| 13 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 24 |
| 25 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 36 |
| 37 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 48 |
| 49 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 60 |
| 61 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 72 |
| 73 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 84 |
| 85 | 0 | 0 | 0 | 0 | 88 | | | | | | | | |

**The Secret Key**

| Bit no. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | |
| 13 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 24 |
| 25 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 36 |
| 37 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 48 |
| 49 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 60 |
| 61 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 72 |
| 73 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 84 |
| 85 | 0 | 1 | 0 | 0 | 88 | | | | | | | | |

**The Decrypted message**

| Bit no. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | |
| 13 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 24 |
| 25 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 36 |
| 37 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 48 |
| 49 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 60 |
| 61 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 72 |
| 73 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 84 |
| 85 | 0 | 1 | 0 | 0 | 88 | | | | | | | | |

Decryptedbinaryoutput="01001000011001010110110001101100011011110010000001010111011011110111001001100110110001100100"
Decrypted text = "Hello World"

### 2.2 CryptoBin Implementation

Now we propose the CryptoBin algorithm encryption by using a simple coding language such as VB.NET. For simplicity, we use a simple series of binary codes for plaintext and secret key as shown in Algorithm 1, and 2.

**Algorithm 1** CryptoBin Algorithm (Encryption)

```
'Encryption Process
'plain text = "Hello World"
Diminputstr="01001000011001010110110001101100011011110010000001010111011011110111001001101100011001100"
Dim key = "1011010011" '723 in decimal form
Dim keybit = ""
Dim txtbit = ""
Dim resbit = ""
Dim result = ""
Dim keylength
For x = 0 To inputstr.Length - 1 Step key.Length
keylength = key.Length
'if no. of bits of plaintext length < no. of bits of key length
If (inputstr.Length) - x < key.Length Then keylength = (inputstr.Length) - x
For n = 1 To keylength
txtbit = Mid(inputstr, x + n, 1)
keybit = Mid(key, n, 1)
If keybit = "0" And txtbit = "0" Then
resbit = "0"
ElseIf keybit = "0" And txtbit = "1" Then
resbit = "1"
ElseIf keybit = "1" And txtbit = "0" Then
resbit = "1"
ElseIf keybit = "1" And txtbit = "1" Then
resbit = "0"
End If
result = result + resbit
Next n
Next x
```

Encryptedresult="11111100100010000101011100100010101111001001010010111010010101000011110010111111111010000"

Decryptedresult="01001000011001010110110001101100011011110010000001010111011011110111001001101100011001100"

We can use this code for Image Encryption also. For example; we use an image file named "m.png" and the encrypted file named "m-Encrypt.png", Figure 3 shows the image before and after encryption.

**Algorithm 2** CryptoBin Algorithm (Decryption)

```
'Decryption Process
'Encryptedtext="11111100100010000101011100100010101111001001010010111010010101000011110010111111111010000"
Diminputstr="11111100100010000101011100100010101111001001010010111010010101000011110010111111111010000"
Dim key = "1011010011" '723 in decimal form
Dim keybit = ""
Dim txtbit = ""
Dim resbit = ""
Dim result = ""
Dim keylength
For x = 0 To inputstr.Length - 1 Step key.Length
keylength = key.Length
'if no. of bits of plaintext length < no. of bits of key length
If (inputstr.Length) - x < key.Length Then
keylength = (inputstr.Length) - x
For n = 1 To keylength
txtbit = Mid(inputstr, x + n, 1)
keybit = Mid(key, n, 1)
If keybit = "0" And txtbit = "0" Then
resbit = "0"
ElseIf keybit = "0" And txtbit = "1" Then
resbit = "1"
ElseIf keybit = "1" And txtbit = "0" Then
resbit = "1"
ElseIf keybit = "1" And txtbit = "1" Then
resbit = "0"
End If
result = result + resbit
Next n
Next x
```

(a) Original image (b) Encrypted image

Figure 3: Image before and after encryption

## 3 Performance and Security Analysis

For designing a very good encryption system, it should be resisting all kinds of common attacks such as brute-force attacks, the man in middle attack, dictionary at-

tack, side channel attack, cipher-text attack, and various attacks. Some of the security analysis techniques can perform on the CryptoBin encrypting system while the statistical analysis and key space are included.

The security analysis of the proposed CryptoBin encryption for image encryption will be discussed in this section, such as Histogram Analysis, Correlation between plain images and cipher images, Information Entropy, and Key Space Analysis to prove that the proposed encryption system is effective, safe and more secure against all common attacks. Experiments are executed by using the "Matlab" software. The key parameter for example; $\text{key})_{\text{Decimal}} = 723$ or $\text{key})_{\text{Binary}} = 1011010011$. This parameter must be kept secret. The same key is used to decrypt the cipher-images.

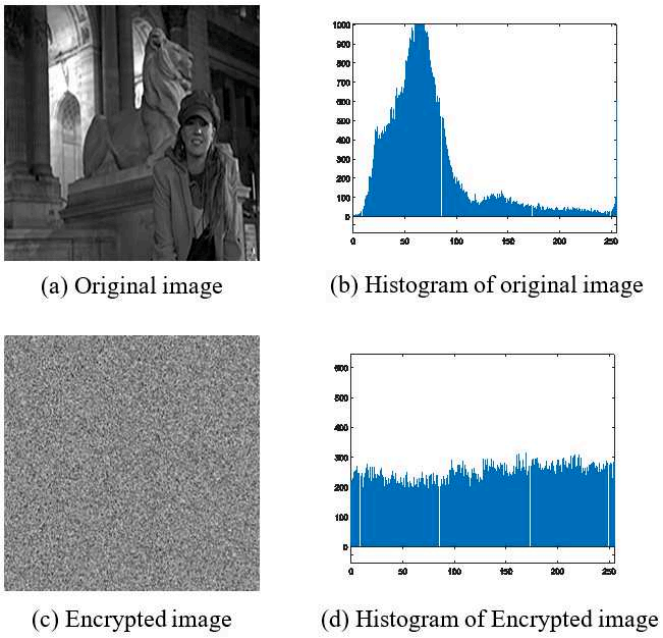
## 3.1 Statistical Analysis

To demonstrate the strength of the proposed encryption system, a statistical analysis was performed showing superior confusion characteristics and also diffusion characteristics in the nature of strong resistance against all kinds of statistical attacks. This is done by the study of Histogram Analysis, Correlation, Key Space Analysis, and Information Entropy between the plain images and ciphered images [17]. Applying the statistical analysis on the CryptoBin system demonstrated the properties of diffusion and the superior confusion of the system that effectively protect from statistical attacks. these results will be shown by the histogram tests on the plain and the ciphered images.

### 3.1.1 Histogram Analysis

Two techniques of confusion and diffusion may be used, as Shannon pointed out, to defeat any strong attacks depending on the statistical analysis. Histogram test is one of Shannon methods and it is applied to ciphered images. We have a grey-scale image (256X256) has different contents, and we calculate its histogram which shows the distribution of pixel intensities of the image. The attacker uses frequency analysis to obtain the secret key or the plain-pixels. This attack type is called a statistical attack. To prevent that statistical attack, the histogram of the original image and histogram of the encrypted image shouldn't have a statistical similarity. Therefore, the histogram of the encrypted image should be relatively flat or with a uniform statistical distribution, indicating the strength and quality of the encryption system [10].

Figure 4 show histograms of image 'm.png' before and after encryption. Histogram of the encrypted image looks relatively flat and with a uniform statistical distribution and distinctive from the histogram of the original image. Based on the experiment results above, the encryption process turned out to return a noisy image, and also the histograms of the previously encrypted image are very similar to the uniform distribution, distinctive from the original image and no statistical similarity to the original



(a) Original image (b) Histogram of original image

(c) Encrypted image (d) Histogram of Encrypted image

Figure 4: Histograms of the plain image and ciphered image

one is contained. The flat histogram in encrypted images can make an attacker's task very difficult to infer pixel values or secret keys using a statistical attack. This corresponds to the ideal security set by Shannon, and the encryption system resists against the known attacks [1].

### 3.1.2 Correlation between Plain and Cipher Images

Correlation is some of a wide class of statistical relationships involving dependence, though in keeping usage it usually identifies how close two variables are to presenting a linear relationship together. We have analyzed the correlation between horizontally, vertically, and diagonally adjacent pixels in a wide range of normal images as well as their encrypted images. The correlation coefficient analysis indicates the partnership among pixels in the cipher image [14]. In the newest scheme, the correlation among adjacent pixels is less than that of the original image. This low correlation value between the original images and their encryption indicates less resemblance between them, which supplies more resistant to attacks. The Statistical correlation is a measure that states the effectiveness of the linear relationship between two random variables. Let 'a' and 'b' are two random variables, each consisting of n elements, the correlation coefficient of both random variables is calculated by the Equations (1,2,3, and 4):

$$r_{\text{ab}} = \frac{cov(a,b)}{\sqrt{D(a)D(b)}} \quad (1)$$

$$D(a) = \frac{1}{n}\sum_{i=1}^{n}[a_{\text{i}} - E(a)]^2 \quad (2)$$

$$cov(a,b) = \frac{1}{n}\sum_{i=1}^{n}[a_i - E(a)][b_i - E(b)] \quad (3)$$

$$cov(a,b) = \frac{1}{n}\sum_{i=1}^{n} a_i. \quad (4)$$

Table 1: Correlation between both adjacent pixels in the plain and ciphered images

| | Plain image | Ciphered image |
|---|---|---|
| Diagonal | 0.9358 | -0.4020 |
| Horizontal | 0.9427 | 0.0082 |
| Vertical | 0.9858 | -0.0005 |

From Table 1 results and the correlation charts, we noticed that there is a negligible correlation between both adjacent pixels in the ciphered image. But both adjacent pixels in the original image are extremely correlated. Correlation in the encrypted images is exceptionally little or insignificant while the suggested encrypting scheme is utilized. Therefore, the suggested encryption system has a great change and substitution properties.

#### 3.1.3 Information Entropy

The information entropy is simply the average (expected) amount of the information from the event or how much information there's in an event. In general, the more uncertain or random the event is, the more information it will contain. It was founded in 1949 by Claude E. Shannon [16]. Entropy test is the other one of Shannon methods and it is applied to ciphered images, the indicator of randomness is the information entropy that can be calculated from the following Equation (5).

$$H(x) = -\sum_{i=1}^{2^N-1} P(x_i) log_2[P(x_i)] \quad (5)$$

The entropy amounts the random value or average uncertainty in xi where P(xi) is how much information from one instance of the random variable xi. If all symbols have the same probability then the information entropy will be H(x) = 8, while x = ($x_0$,$x_1$,$x_2$, ..., $x_{28}$ -1) and P($x_i$) = 1/28 (i=0, 1, ..., 255), that matches the ideal case. Basically, the scrambled images information entropies are less set alongside to the perfect case. The expected entropy of the scrambled image is close to the perfect case in order to create a great image encryption scheme. We may consider the image to be more random somehow if the information entropy is closer to 8. Table 2 shows the plain image entropy value and its equivalent ciphered image entropy value.

### 3.2 Keyspace Analysis

For the encryption scheme to be so effective, it must be sensitive to the secret keys. The key space size has to be big enough to prevent and stop the brutal attacks [21]. In this case, the size of the key space is unlimited. The results of the experiments showed that CryptoBin is quite sensitive to the secret key. Table 3 shows the CryptoBin is sensitive to the secret keys. As visible once the secret key is changed a little the correlation coefficients become absolutely different.

Table 2: Entropy values for original and encrypted images

| Image | Entropy value |
|---|---|
| Original Image(plain Image) | 7.1200 |
| Encrypted Image (Cipher Image) | 7.9919 |

Table 3: Correlation values for the image by using different secret keys

| Correlation | Vertical | Horizontal |
|---|---|---|
| Original Image | 0.9858 | 0.9427 |
| Encrypted Image (key1) | 0.0160 | -0.0174 |
| Encrypted Image (key2) | 0.0785 | 0.0711 |
| Encrypted Image (key3) | 0.0068 | 0.0078 |

## 4 Conclusion

An advanced approach for a cryptographic system using binary codes based on (0,1) called CryptoBin is proposed. This new algorithm depends on converting the target message into zero and one and then swap the bit value from one to zero and from zero to one by using mathematical equations. This new system has a secret key, this secret key has an unlimited length and a sub-secret key added to the system. The sub-secret key is used to generate a different encrypted message every time even if the same secret key, the sub-secret key, and the same plaintext are used several times, that increased the confidentiality and strength of the system. To prove the effectiveness of the proposed encryption system Histogram Analysis, Correlation between plain images and cipher images, Information Entropy, and Key Space Analysis has been tested. The demands of secrecy and strength to confront the intruders with high efficiency have been achieved and the new system introduced high-security analysis. The data was encrypted by the proposed encryption system against any attacks. The proposed system can be used to secure any software applications.

## References

[1] L. Bi, S. Dai, and B. Hu, "Normalized unconditional e-security of private-key encryption," *Entropy*, vol. 19, no. 3, p. 100, 2017.

[2] S. Bouchkaren and S. Lazaar, "Caes cryptosystem: Advanced security tests and results," *International Journal of Network Security*, vol. 20, no. 1, pp. 177–183, 2018.

[3] S. Dey and R. Ghosh, "A review of cryptographic properties of s-boxes with generation and analysis of crypto secure s-boxes," *International Journal of Electronics and Information Engineering*, vol. 8, no. 1, pp. 49–73, 2018.

[4] N. K. El Abbadi, S. T. Abaas, and A. A. Alaziz, "New image encryption algorithm based on diffie-hellman and singular value decomposition," *matrix*, vol. 55, no. 89, p. 144, 2016.

[5] A. H. Eltengy, S. M. Shohieb, M. S. Ksasy, and A. E. TakielDeen, "A new advanced cryptographic algorithm system for binary codes by means of mathematical equation," *ICIC Express Letters*, vol. 12, no. 2, pp. 300–308, 2018.

[6] A. H. Eltengy, A. E. Takieldeen, and H. M. Elbakry, "Implementation of a hybrid encryption scheme for sms/multimedia messages on android," *International Journal of Computer Applications*, vol. 85, no. 2, pp. 300–308, 2014.

[7] A. H. Eltengy, A. E. TakielDeen, and H. M. Elbakry, "Implementation of an encryption scheme for voice calls," *International Journal of Computer Applications*, vol. 144, no. 2, pp. 300–308, 2016.

[8] A. Hazra, S. Ghosh, and S. Jash, "Review on dna based cryptographic techniques," *International Journal of Network Security*, vol. 20, no. 6, pp. 1093–1104, 2018.

[9] L.-C. Huang, L.-Y. Tseng, and M.-S. Hwang, "The study of data hiding in medical images." *International Journal of Network Security*, vol. 14, no. 6, pp. 301–309, 2012.

[10] L.-C. Huang, L.-Y. Tseng, and M.-S. Hwang, "A reversible data hiding method by histogram shifting in high quality medical images," *Journal of Systems and Software*, vol. 86, no. 3, pp. 716–727, 2013.

[11] M.-S. Hwang, C.-C. Lee, and S.-F. Tzeng, "A new knapsack public-key cryptosystem based on permutation combination algorithm," *Information Journal of Applied Mathematics and Computer Sciences*, vol. 5, no. 1, pp. 33–38, 2009.

[12] C.-C. Lee, M.-S. Hwang, and S.-F. Tzeng, "A new convertible authenticated encryption scheme based on the elgamal cryptosystem," *International Journal of Foundations of Computer Science*, vol. 20, no. 02, pp. 351–359, 2009.

[13] L. Liu, Y. Li, Z. Cao, and Z. Chen, "One private broadcast encryption scheme revisited," *International Journal of Electronics and Information Engineering*, vol. 7, no. 2, pp. 88–95, 2017.

[14] N. K. Pareek, "Design and analysis of a novel digital image encryption scheme," *arXiv preprint arXiv:1204.1603*, pp. 300–308, 2012.

[15] M. Rasslan, G. Elkabbany, and H. Aslan, "New generic design to expedite asymmetric cryptosystems using three-levels of parallelism," *International Journal of Network Security*, vol. 20, no. 2, pp. 371–380, 2018.

[16] R. A. Rodríguez, A. M. Herrera, Á. Quirós, M. J. Fernández-Rodríguez, J. D. Delgado, A. Jiménez-Rodríguez, J. M. Fernández-Palacios, R. Otto, C. G. Escudero, T. C. Luhrs *et al.*, "Exploring the spontaneous contribution of claude e. shannon to eco-evolutionary theory," *Ecological modelling*, vol. 327, pp. 57–64, 2016.

[17] T. Shah, I. Hussain, M. A. Gondal, and H. Mahmood, "Statistical analysis of s-box in image encryption applications based on majority logic criterion," *International Journal of Physical Sciences*, vol. 6, no. 16, pp. 4110–4127, 2011.

[18] A. Singh and R. Gilhotra, "Data security using private key encryption system based on arithmetic coding," *International Journal of Network Security & Its Applications*, vol. 3, no. 3, pp. 58–67, 2011.

[19] A. E. Takieldeen, E. El-Badawy, S. Gobran *et al.*, "Digital image encryption based on RSA algorithm," *IOSR Journal of Electronics and Communication Engineering*, vol. 9, no. 1, pp. 69–73, 2014.

[20] A. E. Takieldeen, M. A. Shawky, H. M. Elkamchouchi, I. M. Fouda, and M. M. Khalil, "A new image encryption algorithm combining the meaning of location with output feedback mode," in *13th IEEE APCA International Conference on Control and Soft Computing (CONTROLO'18)*, pp. 521–525, 2018.

[21] H. Zhu and R. Wang, "A survey to design privacy preserving protocol using chaos cryptography," *International Journal of Network Security*, vol. 20, no. 2, pp. 313–322, 2018.

## Biography

**Ahmed H. Eltengy** is currently a PhD candidate at Mansoura University, Computer Science Department. He received M. D. in Computer Sciences (2014). He has a lot of publications in various international journals (i.e. "International Journal of Scientific & Engineering Research", "International Journal of Computer Applications", and "ICIC Express Letters"). His interests of research include Software and Hardware Security Programming, Microcontroller and Embedded Systems.

**Dr. Samaa M. Shohieb** is a professor assistant in computer information systems department, faculty of Computers and Information, Mansoura University, Egypt. She's interested in Human-computer Interaction integrated with E-society and designing creative ICT solutions for diverse users with specified capabilities. She is an editorial board of many international Journals including inderscience, and Elsevier.

**Dr. Ali E. Takieldeen** (IEEE Senior Member) received the PhD degree in Electronics and Communications Engineering in "Encryption and Data Security in Digital Communication Systems". He has a lot of publications in var-

ious international journals and conferences. His current research interests are in multimedia processing, wireless communication systems, Microcontroller and Field Programmable Gate Array (FPGA) applications.

**Prof. Mohamed S. Ksasy** holds Ph. D. in Electronics and Communications (1992). He received M. Sc. In Electronics and Communications (1985). He is a Member of the Institute of Electrical & Electronics Engineers (IEEE), Member of the International Journal of Computers Applications (IJCA), and Member of The Arab Control Systems Association (ACSA).