2022 5th International Conference on Computing and Informatics (ICCI) | 978-1-6654-9973-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICCI54321.2022.9756078

A New Approach of Cryptography for Data Encryption and Decryption

Khawja Imran Masud
Department of CSE
Dhaka University of Engineering & Technology, Gazipur
Gazipur, Bangladesh
Email: kimasud.cse@duet.ac.bd

Md Rakib Hasan
Department of CSE
Dhaka University of Engineering & Technology, Gazipur
Gazipur, Bangladesh
Email: bmrakibcseduet@gmail.com

MD. Mozammel Hoque
Department of CSE
Dhaka University of Engineering & Technology, Gazipur
Gazipur, Bangladesh
Email: mozammelmiru@gmail.com

Upel Dev Nath
Department of CSE
Dhaka University of Engineering & Technology, Gazipur
Gazipur, Bangladesh
Email: upelnath12@gmail.com

Md. Obaidur Rahman
Department of CSE
Dhaka University of Engineering & Technology, Gazipur
Gazipur, Bangladesh
Email: orahman@duet.ac.bd

***Abstract*—Nowadays in the modern digital world, everything is rapidly going to be fully dependent on internet communication. Effective use of the internet makes our life easier. The information that we share on the internet has great security risks and challenges in the present day. Cryptography is the solution to secure data from different security risks. To enhance the security of communication systems better cryptosystems technology is obvious in the area of cryptography. Our research focuses on data encryption and decryption technique for a better cryptosystem; where we have proposed a new approach that ensures better performance in comparison to the state-of-the-art solutions. In this work, after generating a unique key using random characters the plain text is encrypted into ciphertext. To do this encryption, a divide and circular left and right shift approach is followed, and conversely, the reverse is maintained for decryptions as well. According to the experimental results, our proposed algorithm provides better results using the chi-square test while comparing with different cryptography algorithms.**

Keywords—Data Encryption Standard, Advanced Encryption Standard, Rivest Shamir Adleman, Least Significant Bit, Most Significant Bit

I. Introduction

The necessity of data security with a corporation has been undergone due to the wide use of the web or online mechanism in today's organizations. With the arrival of computers in every field, the necessity for software tools for shielding files and other information stored on the pc became important. That's why whenever we like to secure our data from differing types of attacks, first we've to secure our computer with different security measures [1].

Cryptography is very necessary everywhere in communicating over any untrusted transmission system [2]. It's a very crucial sector in information system security, and, now in these modern technologies days, cryptography is worked everywhere from browsing the web to phone chat calls. The needy for a better cryptosystem keeps increasing because the enhancement of the recent generation of computers improved the backdated cryptosystems [3].

To enhance the cryptosystems, we have gone through different Cryptography algorithms and Cryptography techniques, which are the prospective candidates for the concerned issue [4]. However, to address the efficient security issues, we have motivated to design a new approach for data encryption-decryption that works with the ASCII value of plain text which performing with a unique secret key.

As an overview of our proposed technique; firstly, some random characters are generated which are chosen for generating the key. Upon having the random characters, a character array is formed, and converted into its corresponding ASCII equivalent binary value. Furthermore, from each of the 8 bits of a character, we separate four LSB bits and four MSB bits, and finally take their corresponding decimal values. Now, we have some pairs of digits where the first digit indicates the block position and the second digit of the pair indicates how many bits will be shifted circularly. To maximize the rotation and ensure more security the modulus of the first digit with 2 is derived, which ensures 0 or 1 in the first digit of the pair. Hence, the circular rotation is taken placed in each block from first position. Note that for each random character; we have some pairs of digits, actually which are our keys. Secondly, the plain text is divided into some blocks where each of the blocks are 10 bytes. Now, we generate our cipher text by using plain text as well keys those have already been generated at first step. To perform the data encryption, we divide the key into two portions, then a circular left shift operation is performed with the first portions of keys, and then circular right shift operation is performed with the rest portion of keys, and conversely, the reverse is maintained for decryptions as well.

To evaluate the performance of the proposed work, in the experiment we have calculated the time complexity of our proposed technique with respect to different cryptography algorithms. Also, we have verified the chi-square test and found the chi-square test value is higher than the other cryptography algorithm ensuring the maximum non-homogeneity between plain text and cipher text.

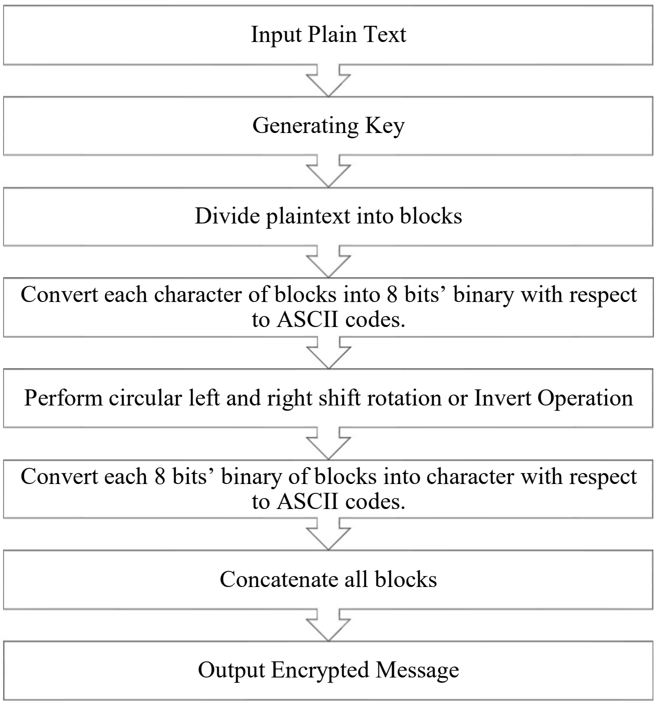
The rest of parts of the paper are organized as follows: Section 2 discuss the related works, Section 3 represents the proposed technique, and in Section 4 we have shown the result analysis. Finally, Section 5 concludes the paper.

II. Related Work

The field of cryptography is extremely intriguing and exciting for people who wish to hide data [5]. One of the foremost use of cryptography is that the RSA cryptography techniques which generates two individual separate keys, namely public key and private key [6]. Such cryptosystems are very popular because of the extent of security it provides in opposition to any quiet attack also brute force attack [7] as calculating the factors of a very large prime which needs several months too for the mightiest computes of the modern age [5]. One of the most popular and effective symmetric key cryptosystems DES is developed to be implemented in hardware systems, although it works on software systems slowly [8]. Such cryptosystems use some simple logical operations and it encrypts and decrypts data with a block cipher and it shows weaker cryptographic capability [8]. Triple DES is another cryptosystems which is the updated version of the DES [9]. Even triple DES has been performing well, there is another cryptosystem AES which superseded most of the existing cryptosystem before 2002 [9]. A fixed key is used to data encrypt and decrypt which means AES is an algorithm based on symmetric-key that provides a great speed on data encryption and decryption [10]. There exist several types of cryptography algorithms; however, the binary field of encryption technology is interesting due to encrypting plain text in a form of machine language resulting in very good security performance [3]. To keep the encryption technique ahead a lot of cryptosystems have been developed in recent decades and this game is going on.

III. Proposed Technique

Fig. 1. and Fig.2. shows the flowchart and process for our proposed data encryption and decryption system. Data encryption is applied to plain text which generates an encrypted text is called cipher text, then the encrypted text is fed to the receiver end to decrypt and get the original plain text by data decryption system.

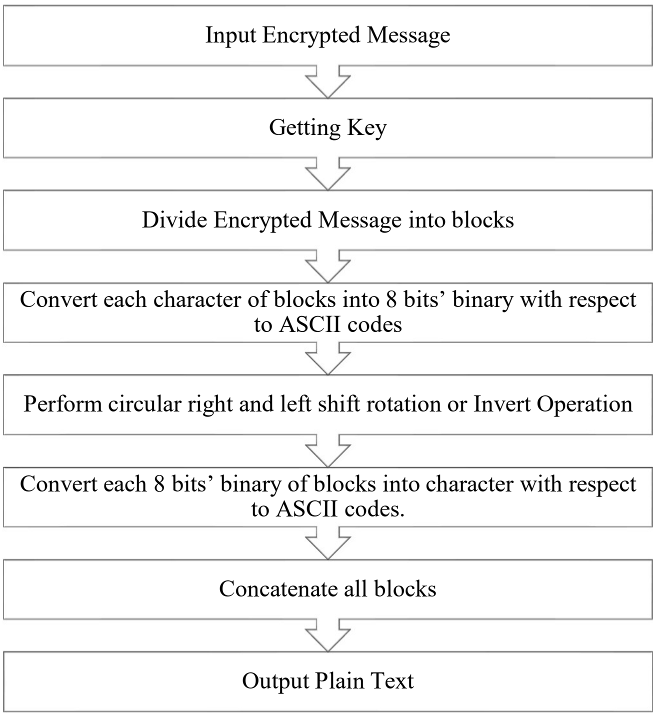


Input Plain Text
Generating Key
Divide plaintext into blocks
Convert each character of blocks into 8 bits' binary with respect to ASCII codes.
Perform circular left and right shift rotation or Invert Operation
Convert each 8 bits' binary of blocks into character with respect to ASCII codes.
Concatenate all blocks
Output Encrypted Message

Fig. 1. Flow chart of the proposed data encryption technique.

We generate a unique secret key where some random characters have been taken and then after some pre-processing, we get the key. Then data encryption is being applied with respect to the key that has been generated.

Finally, in Fig. 2. shows the data decryption techniques which is the reverse process of the encryptions system.

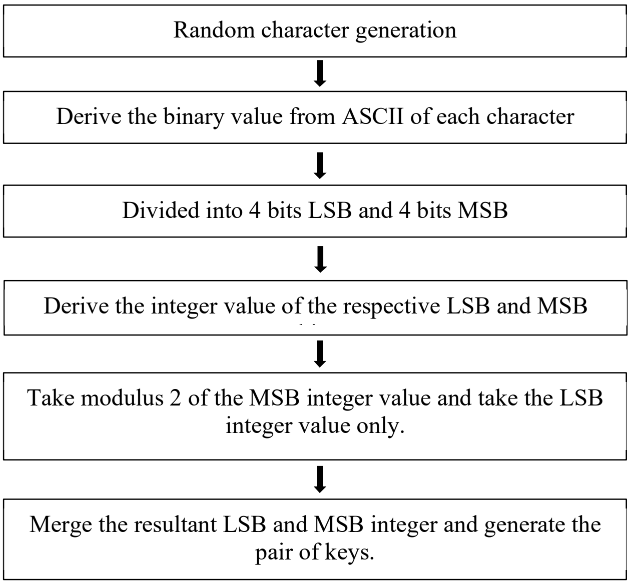


Input Encrypted Message
Getting Key
Divide Encrypted Message into blocks
Convert each character of blocks into 8 bits' binary with respect to ASCII codes
Perform circular right and left shift rotation or Invert Operation
Convert each 8 bits' binary of blocks into character with respect to ASCII codes.
Concatenate all blocks
Output Plain Text

Fig. 2. Flow chart of the proposed data encryption technique.

A. *Generating Key*

We generate the key from a random character generator and then pre-process the character and produce a strong key that ensures reliable security for the desired system. A generated random character is not less than 208 bits. Now we randomly select at least 4 characters of 32 bits for the key generation. After getting the characters we divided them into an array of characters.



Random character generation
Derive the binary value from ASCII of each character
Divided into 4 bits LSB and 4 bits MSB
Derive the integer value of the respective LSB and MSB
Take modulus 2 of the MSB integer value and take the LSB integer value only.
Merge the resultant LSB and MSB integer and generate the pair of keys.

Fig. 3. Flow chart of the proposed key generation technique.

The whole process of key generation is given below.

Random Character = ADYX . . . . n

Array of Character = [A, D, Y, X . . . . n]

ALGORITHM 1: PROPOSED ALGORITHM FOR KEY GENERATION.

```
Input: S ← {Random string of n characters}
Output: K[n] ← {Array of the nth generated keys}
Generate random string S
for i ← 0 to n-1 do
    Divide S into nth characters: C[i] ← S[i]
    Compute ASCII value for each character: A[i] ← C[i]
    for j ← 0 to 7 do
        Compute binary bits for A[i]
        Store binary bits to B[i][j]
    end for
    Divide the 4 bits LSB and MSB
    Compute MSB_val from B[i][0..3]
    Compute LSB_val from from B[i][4..7]
    MSB_val ← MSB_val % 2
    KEY ← Merge MSB_val, LSB_val
    K[i]  ← KEY
end for
```

Now each character is converted to its corresponding ASCII value and derive the binary number from the respective ASCII values of taken characters. Now we separate the 4 bits of LSB and 4 bits of MSB of each binary numbers. After that, we derive the corresponding integer value of the 4-bit LSB's and 4-bit of MSB's then formed an array of each character with such two digits' pairs. Now we modulus the first digit of each pair with 2 and the second digit of each pair will remain the same. We will use the first pair as a character position of a block; that's why we modulus the digit with 2 which always confirms the maximum rotation of the block while data encryption and decryption will be performed. These pair of digits in the array are the final key of our proposed technique. We have two digits' pairs; the first digit denotes the character positions of the corresponding block of plain text and the second digit denotes that how many bits will be circularly shifted.

Array of decimal value = [41, 44, 59, 58, ... n]

Key = [01, 04, 19, 18, ... n]

TABLE I. KEY GENERATION OF THE PROPOSED TECHNIQUE.

A		D		X		Y	
65		68		89		88	
01000001		01000100		01011001		01011000	
0100	0001	0100	0100	0101	1001	0101	1000
4	1	4	4	5	9	5	8
4%2=0	1	4%2 =0	4	5%2 =1	9	5%2 =1	8
01		04		19		18	

For simple demonstration, we have taken only four pairs of characters and generate the key. Table 1. shows the detailed calculation of our unique key generation.

B. Data Encryption Technique

At first we divide the plain text is into blocks where each block size should be 10 bytes. Number of block for plain text is derived by:

Ceiling (length (TEXT)/10)

Let's take a plain text "The World Is a Book" for demonstrating the encryption technique.

PLAIN_TEXT = "The World Is a Book"

BLOCK_1 = "The World "

BLOCK_2 = "Is a Book_"

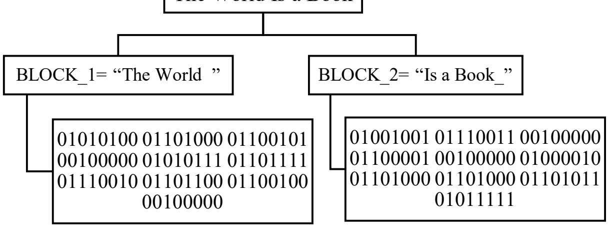
Note that, if the last block size is less than the selected block than extra padding is to be added. Here, a total of one padding byte is added for the taken plain text.

Secondly, the array of keys is divided into two groups and for the first group of keys a circular left shift operation is performed with respect to the key generated, and for the second group circular right shift operation is performed. In the blocks each of the character of plain text is 8 bits, so we can maximum shift 8-bit rotation if we get 0 or 9 for performing shift operation then simply inverted the bit instead of shifting.

ALGORITHM 2: PROPOSED ALGORITHM FOR DATA ENCRYPTION.

```
Input: PLAIN_TEXT.txt
Output: CIPHER_TEXT.txt
Divide input into BLOCK_1, BLOCK_2, ......, BLOCK_N
If BLOCK_N < 10 bytes then
    add extra padding "_"
end if
Compute binary bits for each block.
for i ← 0 to round(n/2)-1 do
    K_1[i] ← K[i]
end for
for j ← round(n/2) to n do
    K_2[j] ← K[j]
end for
for i ← 0 to length(K_1)-1 do
    key_number ← K_1[i]
    first_digit ← key_number / 10
    last_digit ← key_number % 10
    if last_digit > 8 then
        Invert all bits of each block from first_digit position
    else
        r ← 0
        while r ≠ last_digit do
            Circular Left Shift from the first_digit position
            r++
        end while
    end if
end for
for i ← 0 to length(K_2)-1 do
    key_number ← K_2[i]
    first_digit ← key_number / 10
    last_digit ← key_number % 10
    if last_digit > 8 then
        Invert all bits of each block from first_digit position
    else
        r ← 0
        while r ≠ last_digit do
            Circular Right Shift from the first_digit position
            r++
        end while
    end if
end for
Compute the ASCII character of each block.
Concatenate all blocks and generate CIPHER_TEXT.txt
```

Convert plain text to binary data:→



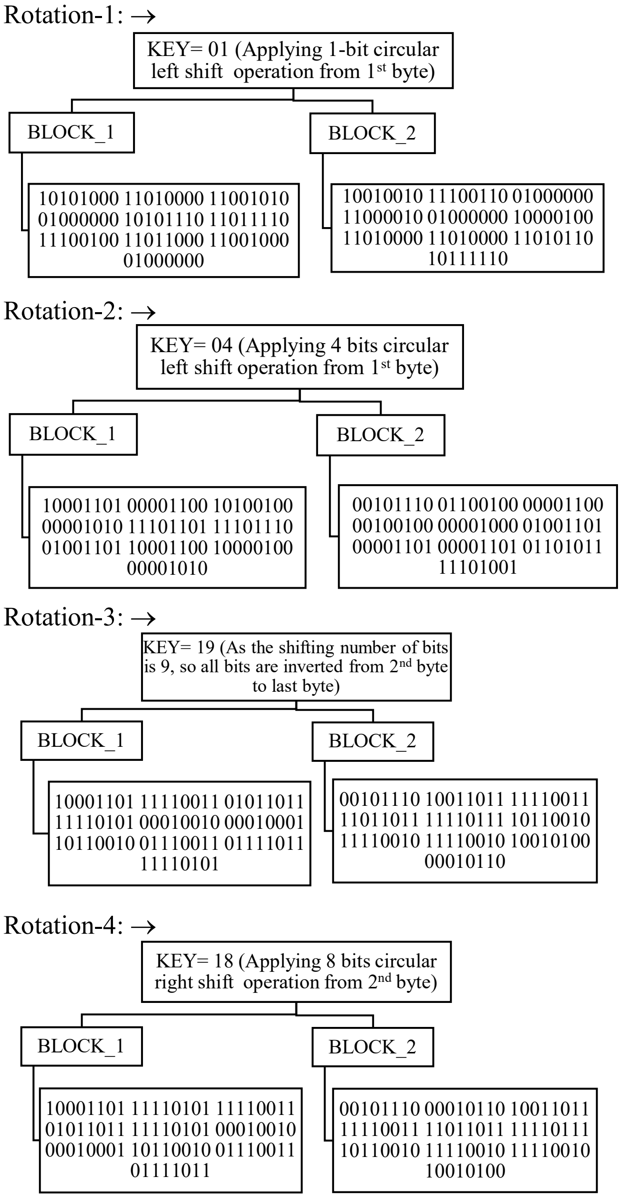


Fig. 4. Flow and processes of data encryption technique.

We divide plain text into two blocks and corresponding binary bit has shown in Fig. 4. Upon getting the keyword from Table 1. circular left shift operation is performed with respect to first group of keys and circular right shift operation is performed with respect to the second group keys.

Finally, after completed all rotation concatenated two blocks and we have the desired cipher text.

BLOCK_1: ì§¾[§ÑÑ▓s{
BLOCK_2: .Ñø¾█‚▓‗ö
CIPHER_TEXT: ì§¾[§ÑÑ▓s{.Ñø¾█‚▓‗ö

C. Data Decryption Technique

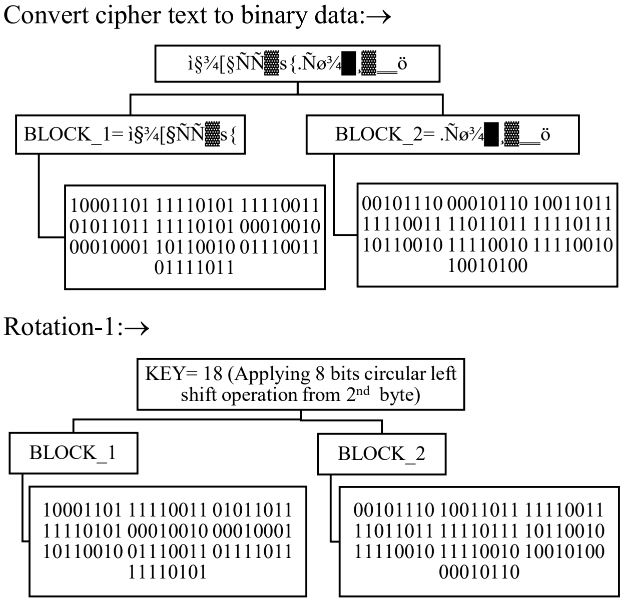
Our decryption technique follows the reverse process of encryption technique. At first the cipher text is divided into blocks of 10 bytes. As we know the key, we follow the key reversely.

CIPHER_TEXT: ì§¾[§ÑÑ▓s{.Ñø¾█‚▓‗ö
BLOCK_1: ì§¾[§ÑÑ▓s{
BLOCK_2: .Ñø¾█‚▓‗ö

Secondly, circular left, right shift operation and inverting operation is performed with respect to key as like as the encryption method. Fig. 5 shows the flow and process of our proposed data decryption technique.

ALGORITHM 3: PROPOSED ALGORITHM FOR DATA DECRYPTION.

```
Input: CIPHER_TEXT.txt
Output: PLAIN_TEXT.txt
Divide input into BLOCK_1, BLOCK_2, ……, BLOCK_N
Compute binary bits for each block.
for i ← 0 to round(n/2)-1 do
    K_1[i] ← K[i]
end for
for j ← round(n/2) to n do
    K_2[j] ← K[j]
end for
for i ← length(K_2)-1 to 0 do
    key_number ← K_2[i]
    first_digit ← key_number / 10
    last_digit ← key_number % 10
    if last_digit > 8 then
        Invert all bits of each block from first_digit position
    else
        r ← 0
        while r ≠ last_digit do
            Circular Left Shift from the first_digit position
            r++
        end while
    end if
end for
for i ← length(K_1)-1 to 0 do
    key_number ← K_1[i]
    first_digit ← key_number / 10
    last_digit ← key_number % 10
    if last_digit > 8 then
        Invert all bits of each block from first_digit position
    else
        r ← 0
        while r ≠ last_digit do
            Circular Right Shift from the first_digit position
            r++
        end while
    end if
end for
Compute ASCII character of each block remove padding.
Concatenate all blocks and generate PLAIN_TEXT.txt
```



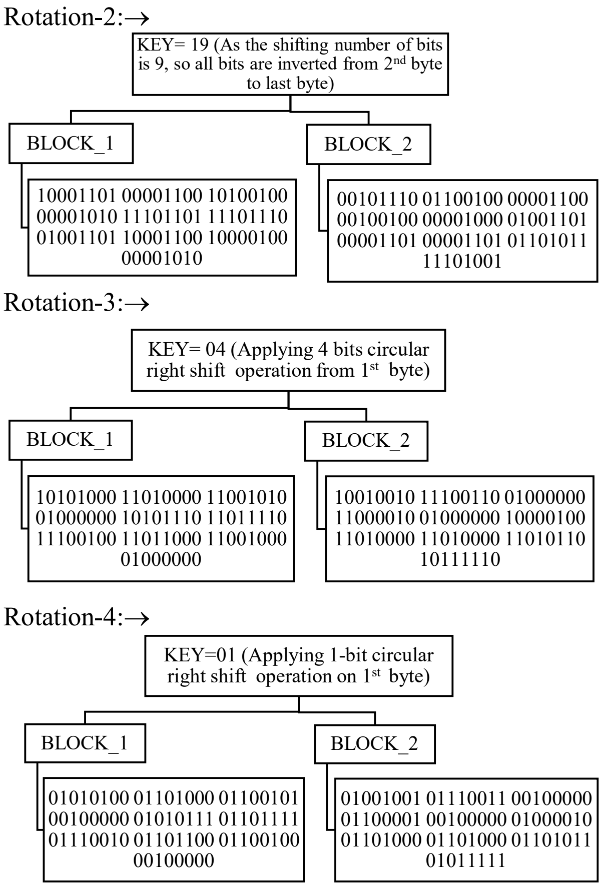


Fig. 5. Flow and processes of data decryption technique.

Now finally, we concatenated two blocks and get the original plain text.

BLOCK_1 = “The World ” & BLOCK_2 = “Is a Book_”
Now after removing padding and concatenated two blocks we get the plain text “The World Is a Book”

IV. Result Analysis

In this section, we have described the performance analysis of our proposed strategy. We discussed encryption time, decryption time, and chi-Square test analysis. We analyze the proposed strategies’ time complexity and also analyze the time complexity of the AES, RSA, and Algorithm [3] data encryptions and decryption systems. We observe that AES encryption and decryption is relatively very fast than other algorithms although such types of speed vary to another versions of AES. We know that RSA is an encryption system which used public-key it follows some pair of keys for data encryption and decryption [11]. We have applied our proposed algorithm on few files with different size. Data encryption and decryption time complexity analysis for our proposed technique is shown in Table 2 and Table 3.

A. Encryption and Decryption Time Analysis

TABLE II. Time Analysis of the Data Encryption.

Name of the file	Size of the file in bytes	Data Encryption Time in Seconds			
		RSA	AES	Algorithm [3]	Proposed Technique
Sample_1.txt	262144	0.828	0.558	4.74	3.077
Sample_2.txt	524288	3.182	0.604	9.291	6.404
Sample_3.txt	1048576	13.502	0.65	18.959	10.2

TABLE III. Time Analysis of the Data Decryption.

Name of the file	Size of the file in bytes	Data Decryption Time in Secounds			
		RSA	AES	Algorithm [3]	Proposed Algorithm
Sample_1.txt	262144	0.91	0.359	4.753	4.67
Sample_2.txt	524288	3.948	0.421	9.441	8.778
Sample_3.txt	1048576	10.469	0.58	19.59	15.77

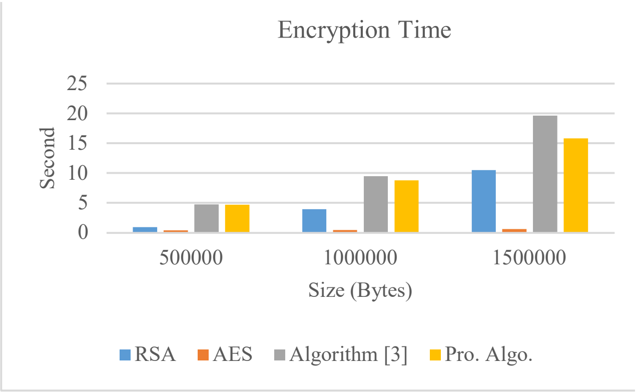


Fig. 6. Encryption time analysis of proposed algorithm.

Fig. 6. and Fig. 7. represent the graphical representation of the time response of RSA, AES, Algorithm [3], and Proposed techniques. Here we observe that our time response is better than the proposed algorithm in [3].

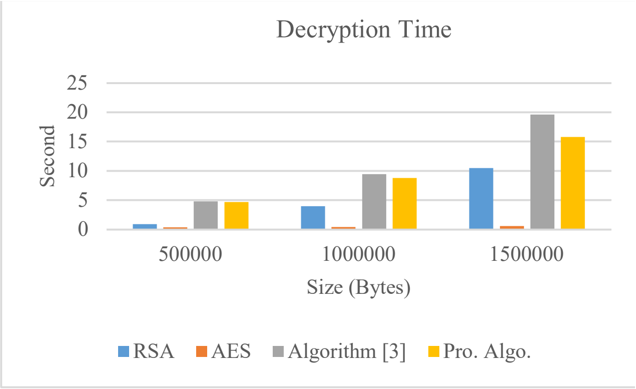


Fig. 7. Decryption time analysis of proposed algorithm.

B. Chi-square Test

Chi-square test is a mathematical derivation that's utilized in main statistics and different comparison techniques which differentiate between expected data values and observed values [12]. It's wont to determine how closely actual data fit with expected data. The worth of chi-square will help us to urge the solution to the question on the importance of the difference in expected and observed data statistically. A little chi-square value will tell us that any differences in actual and expected data are thanks to some usual chance.

$$x^2 = \sum \frac{(Observed\ value - Expected\ value)^2}{Expected\ value}$$

Best Chi-Square test values show the non-homogeneity of the plain text files and respective cipher text files. We apply Chi-Square tests to our Proposed Algorithm, RSA, AES cryptosystems, and Algorithm [3]. Our proposed algorithm has the best chi-square test values which prove that the non-homogeneity of our proposed approaches is greater than all other techniques that we have analyzed.

TABLE IV. CHI-SQUARE TEST ANALYSIS

File Name(.txt)	Chi-Square value			
	RSA	AES	Algorithm [3]	Proposed Algorithm
Sample_1.text	141	141	147	151
Sample_2.text	282	284	295	302
Sample_3.text	564	567	590	608

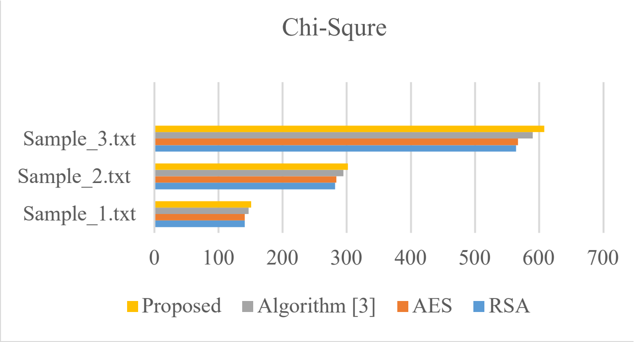


Fig. 8. Chi-square test analysis of our proposed algorithm.

From Fig. 8. we easily claim that our proposed technique has provided the best value as compared to the RSA, AES, and Algorithm [3]. Experiment results show that our proposed algorithm has the highest chi-square test value compared to other algorithms, so the encryption and decryption technique is secure to be used.

V. CONCLUSION

In our research work, we have proposed a new approach for data encryption and decryption for a cryptosystem. Our proposed technique is an efficient approach for data encryption and decryption for textual data or messages. It can generate a unique key based on random character generation and use the circular left or right shift method for data encryption and decryption which provides safer data on the internet communication system as well as for organization, personal, and business purposes, etc. Our proposed approach ensures a better cryptosystem than the other cryptosystem and ensures maximum data security. The usability of our proposed technique is that it could be encrypts the keys of other encryption system and can be used on such applications where data is small i.e., cell phones, Smart Tags, RFID, sensor network, etc. By the utilization of this system, communication is often more reliable and trusted. It has provided the best privacy and security rather than the RSA, AES, and Algorithm [3]. Although experiment on large files, our proposed algorithm time complexity is greater than the AES, RSA though it's better than the Algorithm [3]. The experiment result shows that our proposed approach has the highest chi-square test value compared to other algorithms, so we state that our approach is relatively secure to be used.

REFERENCES

[1] A. A. Danasingh, “Performance Analysis of Data Encryption Algorithms for Secure Data Transmission,” Int. J. Sci. Adv. Res. Technol., vol. 2, pp. 388–390, 2016.

[2] A. Zaru and M. Khan, “General summary of cryptography,” Int. J. Eng. Res. Appl., vol. 08, no. 02, pp. 68–71, 2018, doi: 10.9790/9622-080206871.

[3] D. Pradhan, S. Som, and A. Rana, “Cryptography Encryption Technique Using Circular Bit Rotation in Binary Field,” ICRITO 2020 - IEEE 8th Int. Conf. Reliab. Infocom Technol. Optim. (Trends Futur. Dir., pp. 815–818, 2020, doi: 10.1109/ICRITO48877.2020.9197845.

[4] A. Gupta and N. Walia, “Cryptography Algorithms: A Review,” Int. J. Eng. Dev. Res. 2321-9939, vol. 2, p. 1667, 2014.

[5] S. Nisha and M. Farik, “RSA Public Key Cryptography Algorithm – A Review,” Int. J. Sci. Technol. Res., vol. 6, pp. 187–191, 2017.

[6] X. Zhou and X. Tang, “Research and implementation of RSA algorithm for encryption and decryption,” in Proceedings of 2011 6th International Forum on Strategic Technology, 2011, vol. 2, pp. 1118–1121, doi: 10.1109/IFOST.2011.6021216.

[7] L. Bošnjak, J. Sres, and B. Brumen, “Brute-force and dictionary attack on hashed real-world passwords,” 2018, pp. 1161–1166, doi: 10.23919/MIPRO.2018.8400211.

[8] K. Rabah, “Theory and Implementation of Data Encryption Standard: A Review,” Inf. Technol. J., vol. 4, no. 4, pp. 307–325, 2005, doi: 10.3923/itj.2005.307.325.

[9] S. Heron, “Advanced Encryption Standard (AES),” Netw. Secur., vol. 2009, no. 12, pp. 8–12, 2009, doi: 10.1016/S1353-4858(10)70006-4.

[10] A. Laad and K. Sawant, “A Literature Review of Various Techniques to Perform Encryption and Decryption of Data,” in 2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT), 2021, pp. 696–699, doi: 10.1109/CSNT51715.2021.9509595.

[11] A. S. Prerna Mahajan Dr., “A Study of Encryption Algorithms AES, DES and RSA for Security,” Glob. J. Comput. Sci. Technol. Vol 13, No 15-E Glob. J. Comput. Sci. Technol., Dec. 2013, [Online]. Available: https://computerresearch.org/index.php/computer/article/view/272.

[12] R. Singhal and R. Rana, “Chi-square test and its application in hypothesis testing,” J. Pract. Cardiovasc. Sci., vol. 1, 2015, doi: 10.4103/2395-5414.157577.