International Journal of Communication and Information Technology

E-ISSN: 2707-6628
P-ISSN: 2707-661X
www.computersciencejournals.com/ijcit
IJCIT 2021; 2(1): 49-60
Received: 25-11-2020
Accepted: 28-12-2020

Mohammad Anwar Hossain
Department of CSE World University of Bangladesh Dhaka, Bangladesh

Ahsan Ullah
Department of CSE World University of Bangladesh Dhaka, Bangladesh

Md. Shakil Hossain
Department of CSE World University of Bangladesh Dhaka, Bangladesh

Sumaiya Begum
Department of CSE World University of Bangladesh Dhaka, Bangladesh

Md. Ibrahim
Department of CSE World University of Bangladesh Dhaka, Bangladesh

Corresponding Author:
Mohammad Anwar Hossain
Department of CSE World University of Bangladesh Dhaka, Bangladesh

Design and development of a novel symmetric algorithm for enhancing information security

Mohammad Anwar Hossain, Ahsan Ullah, Md. Shakil Hossain, Sumaiya Begum and Md. Ibrahim

DOI: https://doi.org/10.33545/2707661X.2021.v2.i1a.40

Abstract
Because of the tremendous rise in internet-based cybercrime, the safety of data is becoming increasingly important in order for the internet to continue providing its many features and benefits. The largest problem for data owners and service providers is ensuring the security and privacy of their data. Digital technology has become an integral element of our daily lives. Technology plays a vital part in everything from online shopping to online banking to government infrastructure. Cyber-attacks, on the other hand, are a blemish on the digital landscape. As a result, the authors devised a new symmetric algorithm that utilizes a private key and delivers a more scalable, secure, and speedy algorithm solution. As a result of this algorithm's efforts, the security hazards to confidential material will be greatly reduced. This study focuses mostly on the issues that data security faces when working with the most current technology.

Keywords: Data security, symmetric algorithm, 512bits, sha-512, hashing, key expansion

Introduction

Chapter-1: Information or data that is confidential, private, or sensitive can be protected use, misuse, disclosure or destruction by means of principles and methods that are designed and implemented [1].

Security initiatives have increased in tandem with the rise of knowledge as one of the 21st century's most valuable commodities. Because of the tremendous rise in internet-based cybercrime, the safety of data is becoming increasingly important in order for the internet to continue providing its many features and benefits. By employing cryptographic procedures, such as encrypting and decrypting, network security can be established. Triple DES, Two fish, Blowfish, AES, IDEA, MD5, and RSA are among the most secure encryption methods [2]. In this research, researchers are working to create a new symmetric algorithm that will be superior to existing ones. The authors of this research used a variety of permutations to make the system more complex and secure.

The reason for using a symmetric approach is that it is more efficient symmetric algorithms do not require as many CPU cycles as asymmetric algorithms, symmetric algorithms are preferable. Asymmetric algorithms tend to be slower, on the whole, than their symmetric counterparts in terms of speed.

Symmetric encryption systems in which both the sender and recipient of a communication have access to the same key are known as symmetric encryption. Strictly speaking, the use of the same key for both encoding and decoding is preferable to the use of a symmetric key. That's why you can't decipher a message until you get hold of the sender's secret key [3].

Chapter-2: Literature Review

Cybersecurity, or InfoSec, is the activity of preventing unwanted access to information To put it another way, data security is the discipline of ensuring that no one can get their hands on information without permission.

There are numerous reasons why organizations put in place information security measures. All firm information must be kept safe and secure by Information Security (InfoSec). It is common for information security, infrastructure security, cryptography, and vulnerability management to be implemented simultaneously. Many research works are being proposed to secure the data [2]. Rajat Goel, Ripu R Sinha and O.P Rishi (2011) proposed a hybrid algorithm for information security by the help of NDEA (Novel Data Encryption Algorithm), DES (Data Encryption Standard), IDEA (International Data Encryption Algorithm) and Fiestel Structure.

Encryption is performed using NDEA, DES, IDEA and Fiestel Structure. The suggested paper use a 128- bit key, a 64-bit block size, and eight rounds of computation. Plaintext and key sizes are both 512 bits; the algorithm uses a newer, method that is safer and takes less time than AES to encrypt the data [3].

Marwan Ali Albahar, olaymi Olawumi, keijo Haataja and Pekka Toivanen Proposed hybrid encryption algorithm based on AES, RSA and Twofish for Bluetooth encryption in order to improve the security of Bluetooth. First, the message is encrypted using AES with a 128-bit key, and then a second time using Two fish with the same 128-bit key to encrypt the message again. To ensure its security during transmission over the air, information will be encrypted using RSA and a 1024-bit key. The process of encrypting and subsequently decrypting data is a reversible one. The evaluated approach utilizes a more secure hash function like SHA-512, whereas the proposed article does not [4].

A.S.N Chakravarty and T. Anjikumar (2013) proposed a novel symmetric key cryptography using multiple random secret keys. Session keys and permanent keys are utilized in the proposed book to encrypt or decode user data during communication and to distribute the session key. Two systems are allowed to connect to each other, and the front-end processor encrypts the entire procedure. In this approach, the plaintext is broken down into blocks of a certain size, and each block of cipher text is generated with a length equal to the plaintext. This algorithm uses 256 characters. Each character in the secret key is treated as a separate unit of memory. A new symmetric method that is both more secure and less time consuming than AES and DES has been studied for its use of two distinct keys for encryption and decryption [5].

Ali M Alshahrani and Prof. Stuart Walker (2015) developed an algorithm to enhance block cipher security by using cubical technique. Although the method's key length is long, its most important characteristic is the complicated technique utilized to generate a key and the usage of two keys to generate the algorithm. This algorithm uses 512-bit plaintext and 1024-bit keys. In total, there are ten rounds, separated into two groups. Data can be encrypted and decrypted using the technique in question. SHA-512 is used as a hash function in the reviewed algorithm. When it comes to the total number of rounds, the new algorithm has less than the current one [6].

Ali M Alshahrani and Prof. Stuart Walker (2014) has developed an algorithm to enhance data security in less time than is required for some other systems. The technique primarily shifts a smart table of four quarters, each of which contains 64 bytes, in four separate rounds. A key of 64 bytes will be generated for each quarter in this table, with a total of 64 bytes in each quarter. The maximum key length generated by this technique is 2048 bits. A 256-byte table called the key container table is part of the algorithm. The author employed symmetric and asymmetric algorithms to complete this task. The proposed paper use secret key as SHA-256 [7].

Prakash Kuppuswami and saeed Q.Y.AL Khalidi (2014) proposed an algorithm based on hybrid encryption and decryption technique using new public key algorithm and private key algorithm. The author uses a symmetric key technique to do this task. A two-way secure data encryption system is proposed by the authors to solve privacy, authentication, and privacy concerns of users. The encryption and decryption sequences are used in two separate algorithms in this algorithm. The proposed algorithm one public key cryptography based on linear block cipher another one is private key cryptography based on simple symmetric algorithm.

Chapter-3: Methodology

The process was broken down into five stages by the authors in this study. These phases are planning, requirement analysis, proposed algorithm, implementation, testing and result. Figure 1: Proposed Methodology.

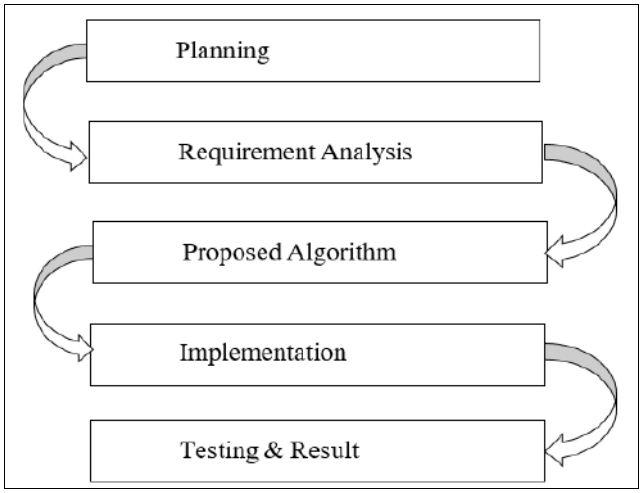

Fig 1: Proposed Methodology

Planning

Planning is the first step to a good research project. As a result, the researchers began this investigation with a clear strategy in mind. The plan contains the study topic and the method of doing the research. Prior to writing this article, the authors reviewed a wide range of studies on the same issue. Then, based on what they had learned from those papers, the writers came up with the title. During their review of the studies, the authors discovered that each one had its own set of restrictions. As a result, the researchers devised a strategy to circumvent these restrictions while yet maintaining the work's individuality.

Requirement Analysis

Every project has its own set of criteria, which vary depending on the task at hand. Resource requirements for the proposed project can be found in the project description.

A. System Requirements: System requirement can be isolated into two types:

1. Software requirement
2. Hardware requirement

⇒ Software Requirements

1. Language-Java
2. Environment –JDK and JRE
3. Operating System –Windows
4. Cloud Server
5. External Algorithm –AES, DES, SHA

⇒ Hardware Requirements

1. Laptop or Desktop with processor
2. USB cable

B. User Requirements: user requirement includes what the user expects from the system. For this, the user wants security of data including integrity, confidentiality and authentication.

Proposed Algorithm

Block wise, the algorithm proposed works. Cipher text is generated from a plain-text data block that is up to 512 bits in length. Numerous unique methods of encryption and decoding are included in this algorithm. The 512-bit key size is used for both encryption and decryption. The data is encrypted and decrypted in seven rounds by the algorithm. The data will be encrypted and decrypted using a unique key for each round. Authentication is also performed using a hash value generated by the system. Protecting data and information is the primary objective of this algorithm.

Implementation

Java is the programming language that was utilized to create the proposed algorithm. Java is the programming language of choice for the algorithm's design. The algorithm for encrypting data in Java is divided into two parts. An encrypted text and a hash code are generated after a 512-bit secret key is provided. Using the algorithm for decrypting data, the first stage is receiving the cipher text, the second step is obtaining the secret key, and the third step is receiving the hash code. After implementation, the authors proposed to use the technique for the protection of confidential information.

Testing and Result

Upon implementation, the authors discovered that they had achieved what they had sought. A new algorithm was then developed that met all of the criteria needed to improve internet security, compared to an existing algorithm that met only some of the criteria.

Chapter –4: Research Design and Analysis

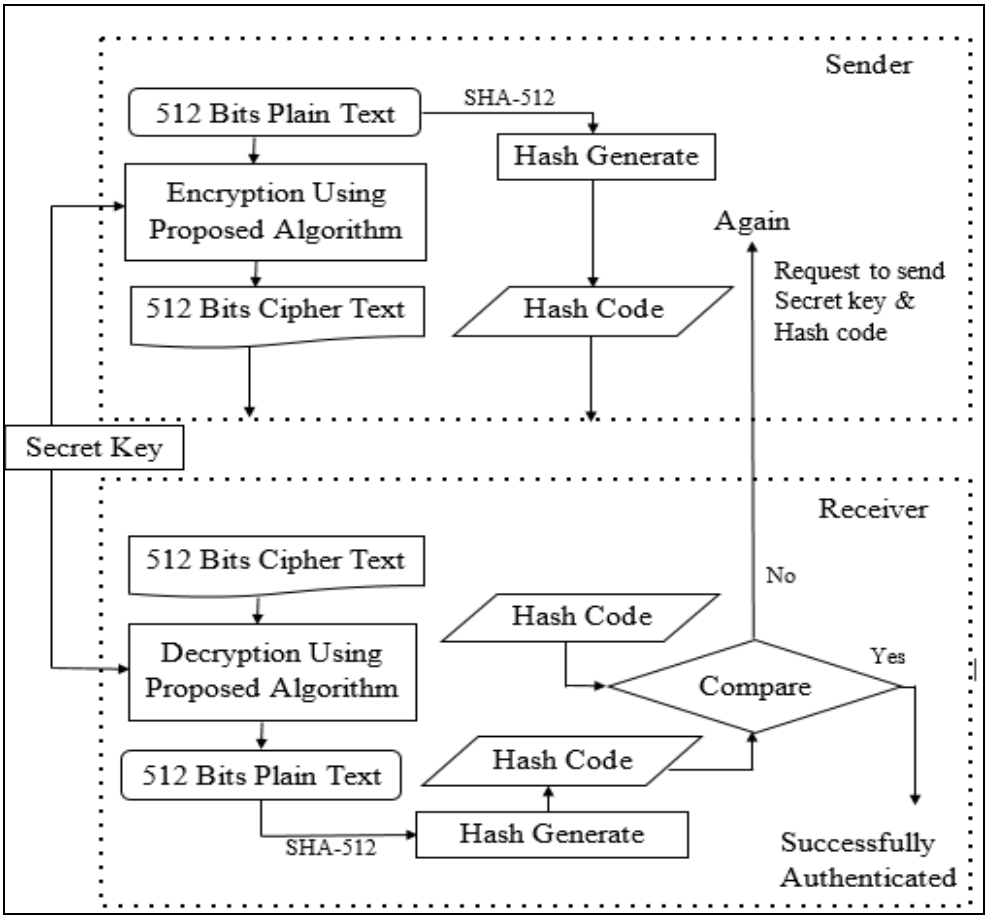

Fig 2: Flowchart of Encryption and Decryption Process Overview

Hashing

SHA 512 Logic

SHA 512 is a cryptographic hashing algorithm that takes a document with a maximum size of less than 2128 characters as argument and outputs a 512-bit message digest as output. The input is processed in 1024-bit blocks by the algorithm. The full processing of a message to produce a digest is depicted in. (“Stallings, 2016”)

Features of SHA-512 hashing algorithm-

- Plaintext Block Size = 1024 bits
- No. of Rounds/steps = 80
- Each Round – Qword = 64 bits
- Each Round- constant K Buffer – 8 buffers (A, B, C, D, E, F, G, H)
- Store Intermediate result
- Each buffer size – 64 bits (“tutorialspoint.com”)

Encryption Process

Encryption Process Flow Chart

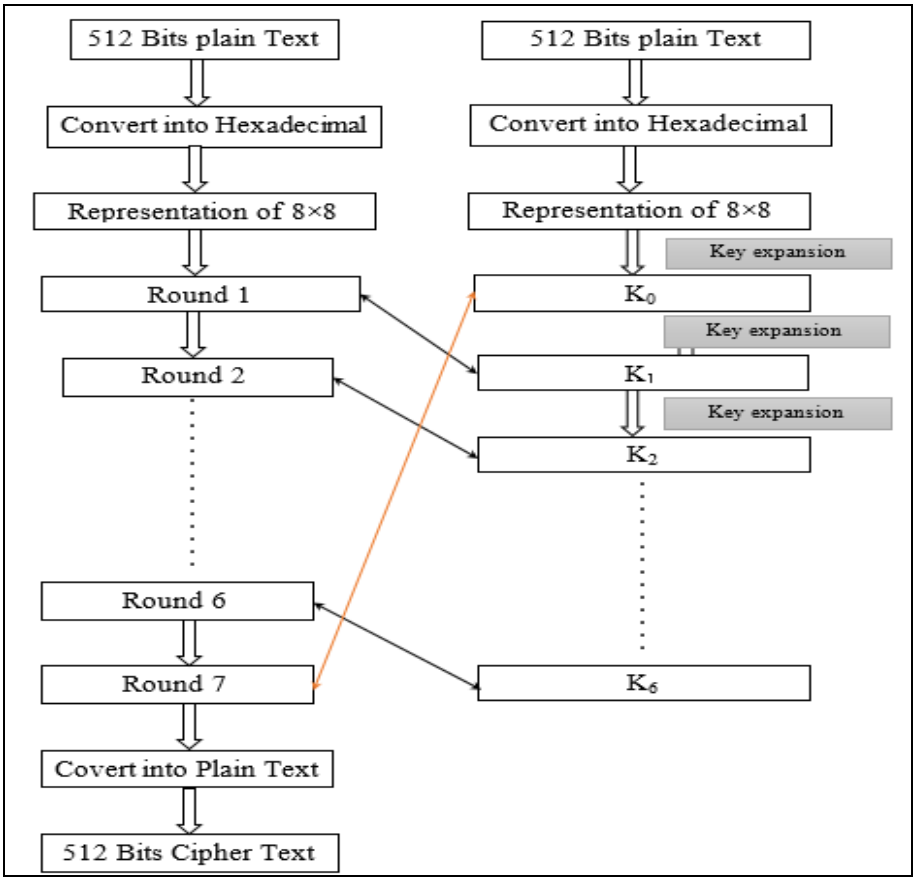

Fig 3: Encryption Process Flowchart

Encryption process Round Function Block diagram

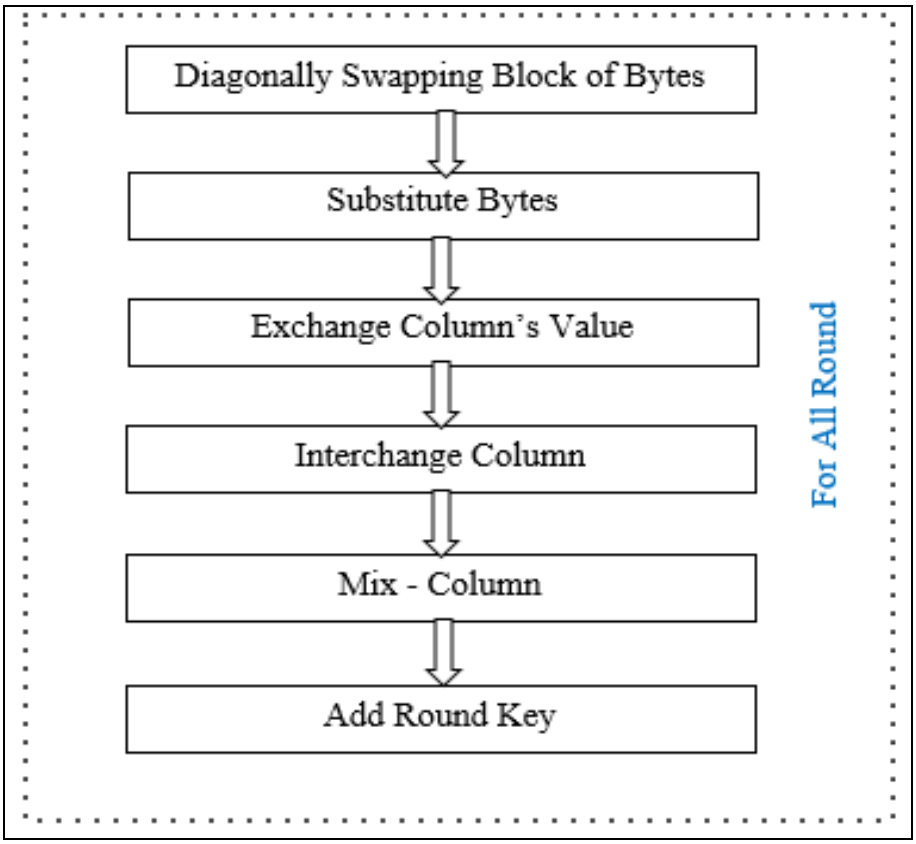

Fig 4: Round Function of Encryption Process

Block diagram of Encryption & Decryption process of Key Expansion

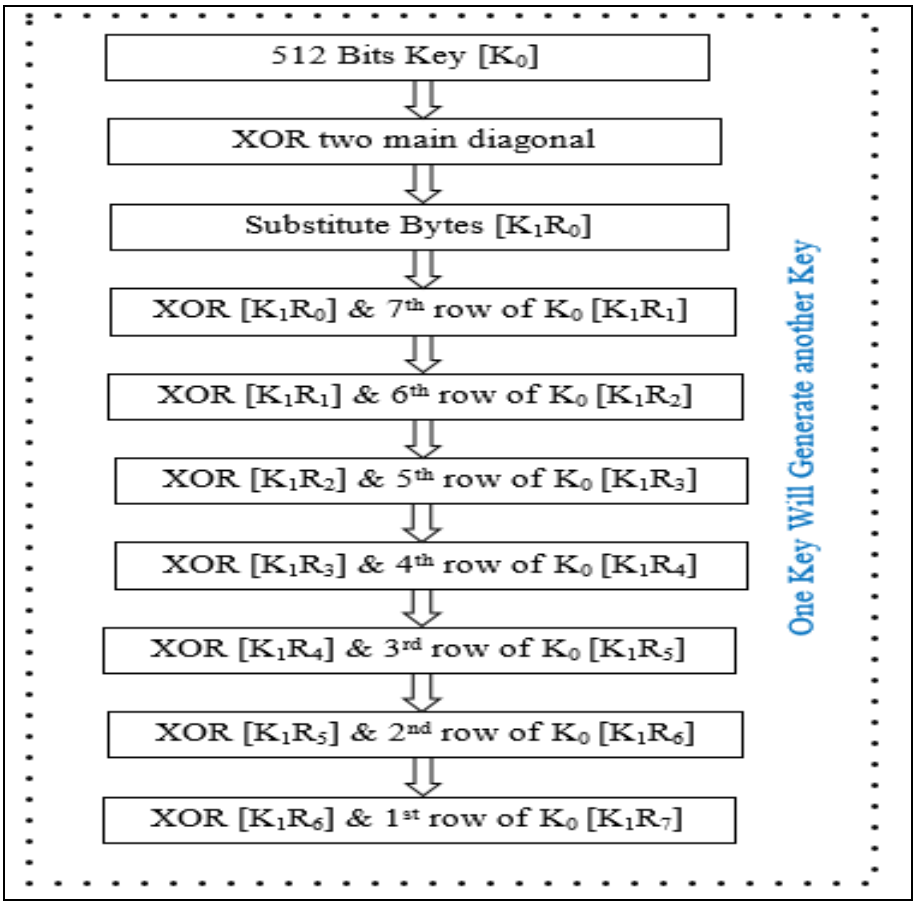

Fig 5: Block diagram of Encryption & Decryption process Key Expansion

Encryption Process Description

1. Take an input or plaintext message of any size.
2. Convert the messages characters into Hexadecimal equivalent.
3. Generate 8*8 block matrix. Which is denoted by M.

Table 1: Generation of 8*8 block matrix

M =

R_C	0	1	2	3	4	5	6	7
0	A0,0	A0,1	A0,2	A0,3	A0,4	A0,5	A0,6	A0,7
1	A1,0	A1,1	A1,2	A1,3	A1,4	A1,5	A1,6	A1,7
2	A2,0	A2,1	A2,2	A2,3	A2,4	A2,5	A2,6	A2,7
3	A3,0	A3,1	A3,2	A3,3	A3,4	A3,5	A3,6	A3,7
4	A4,0	A4,1	A4,2	A4,3	A4,4	A4,5	A4,6	A4,7
5	A5,0	A5,1	A5,2	A5,3	A5,4	A5,5	A5,6	A5,7
6	A6,0	A6,1	A6,2	A6,3	A6,4	A6,5	A6,6	A6,7
7	A7,0	A7,1	A7,2	A7,3	A7,4	A7,5	A7,6	A7,7

4. Perform Diagonally Swapping Block of Bytes of bytes like:

✓ At first this 8*8 block matrix will divided in to 4 block

Table 2: Swapping Block of Bytes

R_C	0	1	2	3	4	5	6	7
0	A0,0	A0,1	A0,2	A0,3	A0,4	A0,5	A0,6	A0,7
1	A1,0	A1,1	A1,2	A1,3	A1,4	A1,5	A1,6	A1,7
2	A2,0	A2,1	A2,2	A2,3	A2,4	A2,5	A2,6	A2,7
3	A3,0	A3,1	A3,2	A3,3	A3,4	A3,5	A3,6	A3,7
4	A4,0	A4,1	A4,2	A4,3	A4,4	A4,5	A4,6	A4,7
5	A5,0	A5,1	A5,2	A5,3	A5,4	A5,5	A5,6	A5,7
6	A6,0	A6,1	A6,2	A6,3	A6,4	A6,5	A6,6	A6,7
7	A7,0	A7,1	A7,2	A7,3	A7,4	A7,5	A7,6	A7,7

Here, let assume Block of red = 1 Block of yellow = 2
Block of green = 3
Block of blue = 4
Then, swap the block 1 with 4 & 2 with 3.

5. Perform substitute bytes (S-BOX) with Rijndael S-Box Here the bytes will be substitute from Rijndael S-Box cell by cell

Table 3: Rijndael S-Box cell by cell

R_C	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

6. Perform "Exchange Column's Value"

a) Here A0,0 will take place of A1,0; A1,0 will take place of A6,0; and A6,0 will take place of A0,0
b) A1,1 will take place of A2,1; A2,1 will take place of A5,1 ; and A5,1 will take place of A1,1
c) A2,2 will take place of A3,2; A3,2 will take place of A4,2; and A4,2 will take place of A2,2
d) A3,3 will take place of A4,3 and A4,3 will take place of A3,3
e) A4,4 will take place of A5,4; A5,4 will take place of A2,4; and A2,4 will take place of A4,4
f) A5,5 will take place of A6,5; A6,5 will take place of A1,5; and A1,5 will take place of A5,5
g) A6,6 will take place of A7,6; A7,6 will take place of A0,6; and A0,6 will take place of A6,6
h) A7,7 will take place of A0,7 and A0,7 will take place of A7,7

7. Interchange Column

- The sequence of interchange columns are 0 -> 2 -> 4 -> 6 -> 1 -> 3 -> 5 -> 7 -> 0
- That means the values of 0th column will take place of 2nd column; the values of 2nd column will take place of 4th column; the values of 4th column will take place of 6th column and so on.

8. Mix Column

Mix Column Predefined matrix:

Table 4: Predefined Mix Column Matrix

R_C	0	1	2	3	4	5	6	7
0	02	01	03	01	01	01	01	01
1	01	03	01	01	01	01	01	02
2	03	01	01	01	01	01	02	01
3	01	01	01	01	01	02	01	03
4	01	01	01	01	02	01	03	01
5	01	01	01	02	01	03	01	01
6	01	01	02	01	03	01	01	01
7	01	02	01	03	01	01	01	01

9. Add Round Key

In this stage the key matrix will be XOR with the resultant matrix of Mix-Column. In this algorithm authors will use 7 separate keys for 7 rounds.

Key Expansion Process Description

1. Take a plain text as a key which is 64 bytes (fixed)
2. Convert the messages characters into Hexadecimal equivalent.
3. Generate 8*8 block matrix. Which is denoted by K0.
4. Take two main diagonal from left to right of K0 and XOR them

Table 5: Generation 8*8 block matrix for Key Expansion

M = / K0 =	R_C	0	1	2	3	4	5	6	7
	0	A0,0	A0,1	A0,2	A0,3	A0,4	A0,5	A0,6	A0,7
	1	A1,0	A1,1	A1,2	A1,3	A1,4	A1,5	A1,6	A1,7
	2	A2,0	A2,1	A2,2	A2,3	A2,4	A2,5	A2,6	A2,7
	3	A3,0	A3,1	A3,2	A3,3	A3,4	A3,5	A3,6	A3,7
	4	A4,0	A4,1	A4,2	A4,3	A4,4	A4,5	A4,6	A4,7
	5	A5,0	A5,1	A5,2	A5,3	A5,4	A5,5	A5,6	A5,7
	6	A6,0	A6,1	A6,2	A6,3	A6,4	A6,5	A6,6	A6,7
	7	A7,0	A7,1	A7,2	A7,3	A7,4	A7,5	A7,6	A7,7

5. Perform substitute bytes (S-BOX) with Rijndael S-box of result of step 4 and this is the 0th row of another 8*8 block matrix which is denoted by K1.
6. Perform XOR operation between 7th row of K0 and result of step 5 which is the 1st row of K1 matrix.
7. Perform XOR operation between 6th row of K0 and result of step 6 which is the 2nd row of K1 matrix.
8. Perform XOR operation between 5th row of K0 and result of step 7 which is the 3rd row of K1 matrix.
9. Perform XOR operation between 4th row of K0 and result of step 8 which is the 4th row of K1 matrix
10. Perform XOR operation between 3rd row of K0 and result of step 9 which is the 5th row of K1 matrix.
11. Perform XOR operation between 2nd row of K0 and result of step 10 which is the 6th row of K1 matrix.
12. Perform XOR operation between 1st row of K0 and result of step 11 which is the 7th row o K1 matrix.

And with this 8 row, an 8*8 block matrix will be generated. Actually this matrix is K1.

Table 6: Generation Key 1(K1) from K0

K1 =	R_C	0	1	2	3	4	5	6	7
	0	B0,0	B0,1	B0,2	B0,3	B0,4	B0,5	B0,6	B0,7
	1	B1,0	B1,1	B1,2	B1,3	B1,4	B1,5	B1,6	B1,7
	2	B2,0	B2,1	B2,2	B2,3	B2,4	B2,5	B2,6	B2,7
	3	B3,0	B3,1	B3,2	B3,3	B3,4	B3,5	B3,6	B3,7
	4	B4,0	B4,1	B4,2	B4,3	B4,4	B4,5	B4,6	B4,7
	5	B5,0	B5,1	B5,2	B5,3	B5,4	B5,5	B5,6	B5,7
	6	B6,0	B6,1	B6,2	B6,3	B6,4	B6,5	B6,6	B6,7
	7	B7,0	B7,1	B7,2	B7,3	B7,4	B7,5	B7,6	B7,7

By following the same process the rest of the key will be generated. One key will generate another key. Here K0 generated K1 and then K1 will generate K2, K2 will generate K3 and so on.

Decryption Process

Decryption Process Flow Chart

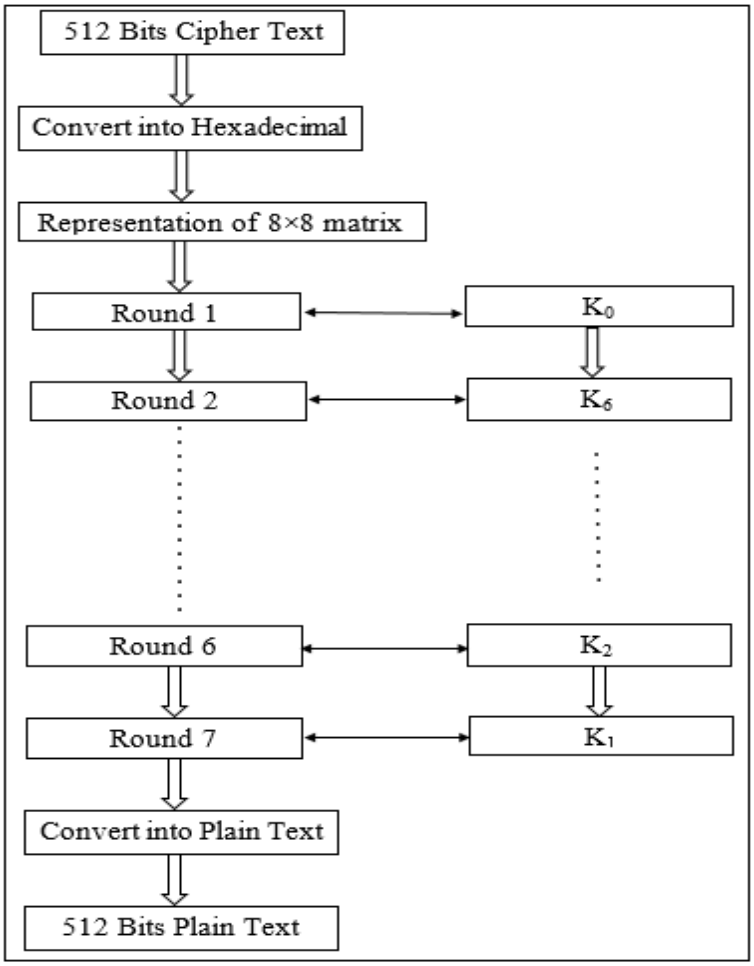

Fig 6: Decryption Process Flow Chart

Block diagram of Decryption Process Round Function

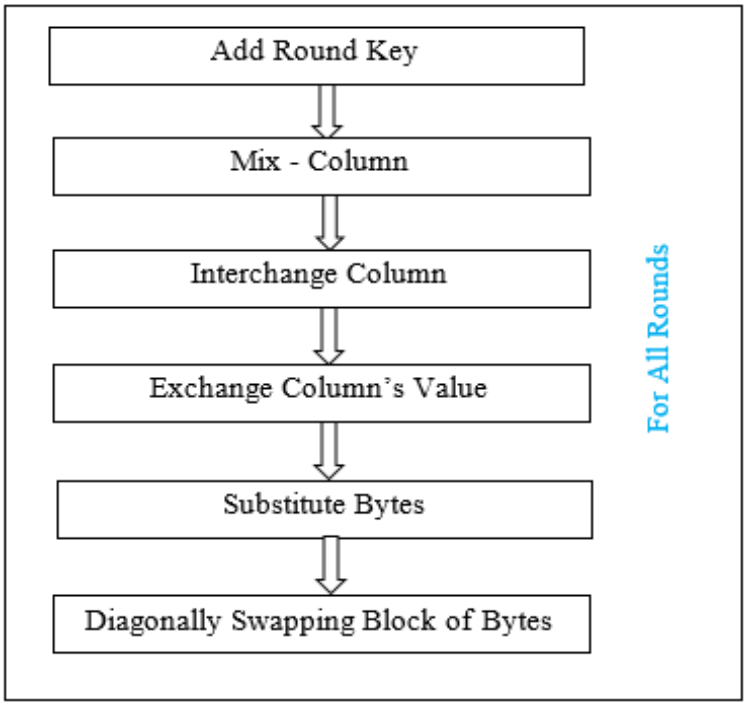

Fig 7: Decryption process Round Function

Decryption Process Description

1. Received the cipher text from the encryption process.
2. Convert into Hexadecimal equivalent.
3. Represent in 8*8 block matrix. Which is denoted by Md

Table 7: Representation of 8*8 block matrix from encrypted test

$M_d =$

R_C	0	1	2	3	4	5	6	7
0	A0,0	A0,1	A0,2	A0,3	A0,4	A0,5	A0,6	A0,7
1	A1,0	A1,1	A1,2	A1,3	A1,4	A1,5	A1,6	A1,7
2	A2,0	A2,1	A2,2	A2,3	A2,4	A2,5	A2,6	A2,7
3	A3,0	A3,1	A3,2	A3,3	A3,4	A3,5	A3,6	A3,7
4	A4,0	A4,1	A4,2	A4,3	A4,4	A4,5	A4,6	A4,7
5	A5,0	A5,1	A5,2	A5,3	A5,4	A5,5	A5,6	A5,7
6	A6,0	A6,1	A6,2	A6,3	A6,4	A6,5	A6,6	A6,7
7	A7,0	A7,1	A7,2	A7,3	A7,4	A7,5	A7,6	A7,7

4. Perform Add Round Key

In this stage the key matrix will be XOR with the resultant matrix of Mix-Column. In this algorithm authors will use 7 separate keys for 7 rounds. For Decryption the Key Expansion Processes are same as before Key Expansion Processes.

5. Perform Mix-Column

Here this mix column calculations are same as encryption process description Mix-column. But the predefine matrix is different.

Table 8: Inverse Mix Column Predefined Matrix

R_C	0	1	2	3	4	5	6	7
0	0E	01	09	01	0D	01	0B	01
1	01	09	01	0D	01	0B	01	0E
2	09	01	0D	01	0B	01	0E	01
3	01	0D	01	0B	01	0E	01	09
4	0D	01	0B	01	0E	01	09	01
5	01	0B	01	0E	01	09	01	0D
6	0B	01	0E	01	09	01	0D	01
7	01	0E	01	09	01	0D	01	0B

6. Perform Interchange Column

- The sequence of interchange columns are 0 -> 7 -> 5 ->3 -> 1 -> 6 -> 4 -> 2 -> 0
- That means the values of 0th column will take place of 7th column; the values of 7th column will take place of 5th column; the values of 5th column will take place of 3rd column and so on.

7. Perform Exchange Column's Value

a) Here A0,0 will take place of A6,0 ; A6,0 will take place of A1,0 ; and A1,0 will take place of A0,0
b) A1,1 will take place of A5,1 ; A5,1 will take place of A2,1 ; and A2,1 will take place of A1,1
c) A2,2 will take place of A4,2 ; A4,2 will take place of A3,2 ; and A3,2 will take place of A2,2
d) A3,3 will take place of A4,3 and A4,3 will take place of A3,3
e) A4,4 will take place of A2,4 ; A2,4 will take place of A5,4 ; and A5,4 will take place of A4,4
f) A5,5 will take place of A1,5 ; A1,5 will take place of A6,5 ; and A6,5 will take place of A5,5
g) A6,6 will take place of A0,6 ; A0,6 will take place of A7,6 ; and A7,6 will take place of A6,6

A7, 7 will take place of A0,7 and A0,7 will take place of A7, 7 i)

8. Perform Substitute Bytes (S-Box) with Rijndael Inverse S-Box

9. Perform Diagonally Swapping Block of Bytes

- At first this 8*8 block matrix will divided in to 4 block of bytes like:

Table 9: Swapping Block of Bytes

R_C	0	1	2	3	4	5	6	7
0	A0,0	A0,1	A0,2	A0,3	A0,4	A0,5	A0,6	A0,7
1	A1,0	A1,1	A1,2	A1,3	A1,4	A1,5	A1,6	A1,7
2	A2,0	A2,1	A2,2	A2,3	A2,4	A2,5	A2,6	A2,7
3	A3,0	A3,1	A3,2	A3,3	A3,4	A3,5	A3,6	A3,7
4	A4,0	A4,1	A4,2	A4,3	A4,4	A4,5	A4,6	A4,7
5	A5,0	A5,1	A5,2	A5,3	A5,4	A5,5	A5,6	A5,7
6	A6,0	A6,1	A6,2	A6,3	A6,4	A6,5	A6,6	A6,7
7	A7,0	A7,1	A7,2	A7,3	A7,4	A7,5	A7,6	A7,7

Here, let assume Block of red = 1
Block of yellow = 2
Block of green = 3
Block of blue = 4 Then, swap the block 1 with 4 & 2 with 3.

Security Analysis

As the internet is increasingly being used for the transfer of private and confidential data, it is critical that the key used to encrypt such data be robust and secure. If we utilize an alphanumeric key with ten characters, we can get around this. Counting the upper and lower cases adds up to 26+26=52, and if we count the numeric digits adds up to 62. There are altogether 26 alphabets in English. For a 10-character key, the number of possible combinations is 6210 or 8.39*1017 or 8.4 quintillion. If a computer were to hack a 10-digit password, it would take nearly 257201646.091 years. It will take a supercomputer 800,000,000 seconds, or 133333333.333 minutes, or 2222222.22222 hours, to crack the code. This computation is based on a 10-digit key, but if we use a 64-digit key and include special characters, it will take a long time to crack the key, that seems nearly unachievable.

Implementation the Algorithm in JAVA

Starting of the program

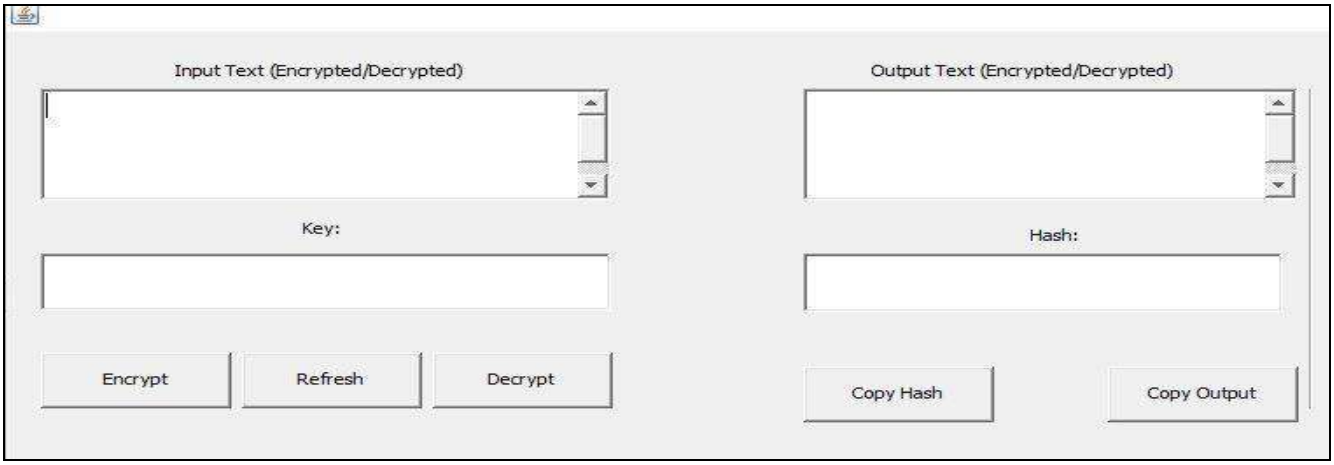

Fig 8: Interface of the Algorithm in JAVA

To encrypt a text, need to type text in “Input Text (Encrypted/Decrypted)” panel. Then had to give a secret key of 512 bits in “Key:” panel. Then had to click on “Encrypt” butto.

Chapter – 5: Result Discussion

The result has tested by –

- Windows 10 Pro 64-bit
- Intel® Core (TM) i3 – 6100U CPU @ 2.30 GHz 2.30 GHz
- 4 GB RAM

Table 10: Algorithm analysis with same key and same message size

Data			Key size	Runtimes (milliseconds)			
Type	No	Size		Encryption		Decryption	
					Average		Average
Numeric	1	512 - bits	512 - bits	442.783ms	442.329ms	172.286ms	171.748ms
	2	512 - bits	512 -bits	439.692ms		168.323ms	
	3	512 - bits	512 - bits	444.513ms		174.636ms	
Alphabetic	1	512 - bits	512 - bits	495.405ms	494.775ms	220.402ms	220.999ms
	2	512 - bits	512 - bits	491.307ms		224.269ms	
	3	512 - bits	512 - bits	497.613ms		218.326ms	
Alphanumeric	1	512 - bits	512 - bits	433.721ms	433.781ms	177.923ms	178.558ms
	2	512 - bits	512 - bits	436.343ms		175.968ms	
	3	512 - bits	512 - bits	431.279ms		181.782ms	

Table 11: Comparison of Proposed Algorithm with AES, DES

Data		Key Size	Run Time (milliseconds)	
Algorithm	Message Size		Encryption	Decryption
Proposed Algorithm	512 – bits	512 - bits	442.783ms	172.286ms
	1024 – bits		1159.604ms	386.667ms
AES	512 – bits	128 – bits	672.351ms	1.637ms
	1024 – bits		1444.269ms	481.589
DES	512 – bits	64 - bits	5.967ms	1.671ms
	1024 – bits		14.683ms	4.894ms

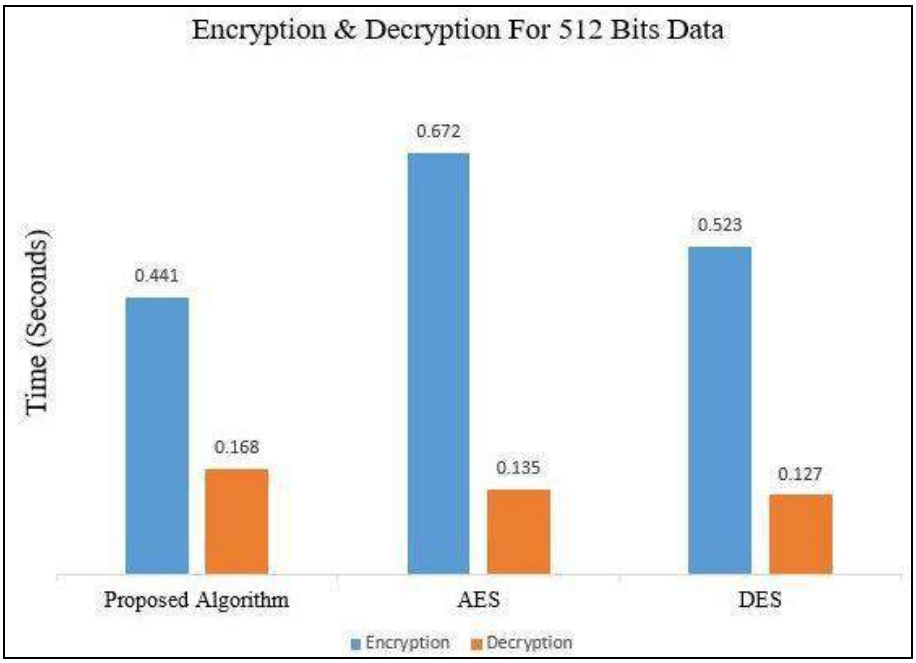

Fig 9: Encryption and decryption times for 512 bits data using the proposed algorithms, AES and DES, as represented graphically.

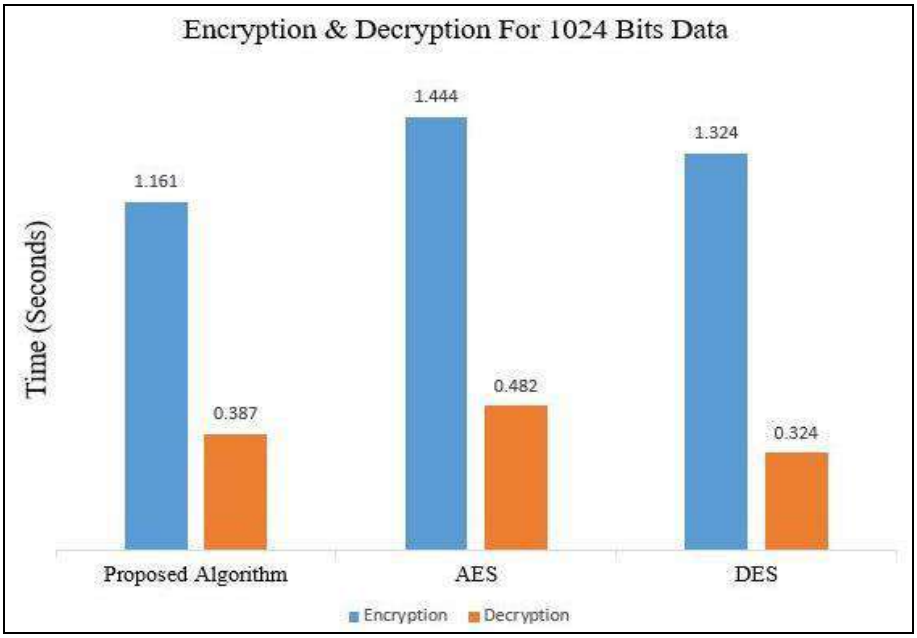

Fig 10: Encryption and decryption time graphs for 1024 bits data using the proposed algorithms, AES and DES.

Chapter 6: Conclusion

Conclusion

The algorithm's primary goal is to protect confidential data.. The authors employed a symmetric algorithm for this aim. Encryption and decryption can now be done with ease thanks to their improved algorithm. Block-wise, the algorithm works. Data can be encrypted using up to 512 bits of data at a time using this algorithm. The data is encrypted and decrypted in seven rounds by the algorithm. Because the technique uses 512-bit keys, it is more secure. To store and encrypt private data, symmetric algorithms are often used all around the world, which is why the authors chose to utilize one for their technique. Confidentiality and integrity of sensitive data are ensured using the method developed by the authors

Limitations

i. It works on text format data only.
ii. Key and Hash code exchange is not much secure.

Future Works

i. Audio, video, image and file encryption.
ii. Providing better security on Key and Hash code exchange.

Acknowledgments

This paper and the research behind it would not have been possible without the exceptional support of my supervisor, Ahsan Ullah. His enthusiasm, knowledge and exacting attention to detail have been an inspiration and kept my work. Md Shakil Hossain, Sumaiya begum and Md Ibrahim, my colleagues at World University of Bangladesh, have also looked over my transcriptions and answered with unfailing patience numerous questions about the Paper. I am grateful to all of those with whom I have had the pleasure to work during this.

References

1. https://www.csoonline.com/article/3513899/what-is-information-security-definition- principles-and-jobs.html
2. Rajat Goel, Ripu Sinha R, Rishi OP. Novel data encryption algorithm, IJCSI International journal of computer science 2011, July 8(4). ISSN (1694-0814).
3. Marwan Ali Albahar, Olaymi Olawumi, Keijo Haataja, Pekka Toivanen. Novel hybrid encryption algorithm based on AES, RSA and Twofish for Bluetooth encryption journal of information security. 2018;9:168-176. ISSN(2153-1242).
4. Chakravarty ASN, Anjikumar T. A novel symmetric key cryptography using multiple random secret keys, International journal of computer application, 2013;18:16.
5. Ali Alshahrani M, Stuart Walker. New approach in symmetric block cipher security using a new cubical technique, International Journal of Computer Science and Information Technology. 2015;7:1.
6. Ali Alshahrani M, Stuart Walker. Implement a novel symmetric block cipher algorithm. 2014;4:4.
7. Prakash Kuppuswami, Saeed Khalidi QYAL. An algorithm based on hybrid encryption and decryption technique using new public key algorithm and private key algorithm, 2014.