International Journal of Scientific World, 3 (2) (2015) 244-254
www.sciencepubco.com/index.php/IJSW
©Science Publishing Corporation
doi: 10.14419/ijsw.v3i2.5111
Research Paper

NARSKCA: Novel and robust symmetric key cryptography algorithm

Balajee Maram [1]*, Y Ramesh Kumar[2], K Lakshmana Rao[3]

[1] *Sr. Asst. Prof., Dept. of CSE, GMRIT, Rajam, India*
[2] *Asso. Prof, Dept. of CSE, Avanthi college of Engineering and Technology, Visakhapatnam*
[3] *Asst. Prof., Dept. of CSE, GMRIT, Rajam, India*
**Corresponding author E-mail: balajee.journal@outlook.com*

Copyright © 2015 Balajee Maram et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

In this research paper, a novel and strong symmetric key cryptography algorithm is proposed. NARSKCA is based on several symmetric cryptographic algorithms. NARSKCA is very simple that uses character converting algorithm, Fibonacci Number Series, Lucas Number series and bitwise XOR. In NARSKCA, 32 files are shared-secret files plays a vital role in this Proposed Algorithm. The Sub-keys are generated from those 32 shared-secret files which are useful in different rounds of Encryption and Decryption Process. The most important feature is the calculation of the final key from the Sub-Keys for each Text-Block. Key Generation, encryption/decryption schemes of NARSKCA are fast and difficult to predict by Cryptanalysts.

Keywords: *DRDP; Symmetric Key Cryptography; Block Cipher; Lucas; Narskca.*

1. Introduction

In this Modern Computing and Communication Technology, whenever any sensitive data transmitted over a public channel, the hacker/attacker could get the message. So there is a need to give protection to the sensitive data which is transmitted through the Internet. For protecting these transmissions, the old and modern cryptographic methods are very useful.

In every cryptographic algorithm, there is a need to either shared secret-key or private/public key pair. The strength of the encryption is based on the length of the key. When the length of the key is very small then easy to crack the key as well as decrypt the data. The required keys should be shared in symmetric cryptography, and the public key should be published in asymmetric cryptography techniques.

In Symmetric Cryptographic Algorithms, a single key is shared between Sender and Recipient. In most of the Symmetric Cryptographic Algorithms (DES, 3-DES, AES, IDEA, RC4 and RC5) the sizes of the key are 56-bit, 128-bit or 256-bit only. In public key cryptography, sender uses a public key of the receiver, known to everyone, to encrypt the message and receiver use his private key, known only to him, to decrypt the message. RSA is one of the most famous public-key algorithm, which is based on Diffie-Hellman Key Exchange [1]. Most of the keys crack able through one of the following techniques:

- Brute-Force Attack
- Differential Cryptanalysis
- Linear Cryptanalysis

In this Research Paper, NARSKCA is proposed the alternative technique for Symmetric Cryptography. In this method, it provides more and more security to the data through 32-shared-secret-files. NARSKCA have the following characteristics:

- It is a symmetric block cipher.

- The Architecture of the NARSKCA is very simple that uses bitwise XOR, DRDP Converting method, Fibonacci & Lucas Number Series.
- It uses variable sizes of 32-Shared-Secret-Files for providing security. User can choose the size of 32-Shared-Secret-Files according to the need and desired level of security.
- For each and every Plain-Text 256-bit Block, it generates Sub-Keys and Final-Keys. It is one of the important features of NARSKCA.
- The generation of Final-Key for each Input-Block is depends on Fibonacci and Lucas Number Series.

This Research Paper is organized as follows: Literature review is covered in Section 2. Proposed algorithm, which includes generation of Round-Key and Final-Key for each Round and the encryption/decryption schemes, is covered in section 3. NARSKCA Encryption process is covered in Section 4. NARSKCA Decryption process is covered in Section 5. Analysis is covered in Section 6. Security-Level is discussed in Section 7. Finally, the conclusion is covered in Section 8.

2. Literature review

2.1. Data communication

Data Communication is the process for conveying the information by the exchange of information, messages, feelings, thoughts, etc. Like this; two persons can exchange their needs, desires, perceptions and knowledge. This is called Data Communication.

In Data Communication, there is a need of a sender, a recipient and a message. Here the presence of Recipient is optional i.e present or absent. Here the intention of the sender to send data to the recipient. If the recipient is in online, then he will receive the message instantly. Otherwise, the message will be stored in Message-Queue/Inbox. The Communication is called successful when both the persons exchange their information successfully [2].

Communicating with others involves three primary steps:

- Thought: First, information exists in the mind of the sender. This can be a concept, idea, information, or feelings.
- Encoding/Encryption: A message is sent to a recipient in words or other symbols.
- Decoding/Decryption: When the recipient translates the words, symbols or cipher into the information that a person can understand.

But in Secure Data Communication, there is a need to apply Cryptographic algorithms. But all algorithms are based on either private-key or public-key.

2.2. Types of cryptography

There are three main types of cryptography:

- Secret key cryptography
- Public key cryptography

Secret key cryptography – in this method, the data is encrypted and decrypted using a “shared secret” key. This type of encryption scheme is also known as symmetric key encryption. Here there is a need to share one common key, which is known as “Secret-Key." In this paper, we have studied a number of such symmetric key algorithms and selected one of them for reference in the proposed algorithm.

DES

Data Encryption Standard (DES) was recommended by NIST (National Institute of Standards and Technology) [3]. It was developed by an IBM in 1974. Here the Data-Block size is 64-bit. The size of the key is 56-bit. It supports initial permutation and sixteen rounds for processing Data-Blocks.

Triple DES

This is an alternative algorithm to DES. TDES uses two or three keys. The main feature is that there are no known practical attacks to it. Its key-size is 56X3=168-bits.

RC2

RC2 is one of the private key algorithm developed by Rivest. It uses 64-bit Plain-Text and a variable key size that varies from 8 to be 1024-bits. This algorithm can be developed on 16-bit microprocessors.

Advance Encryption Standard (AES)

AES is developed by Joan Daemen and Vincent Rijmen. Here the block size is 128 bits. It uses three different key lengths: 128, 192 and 256 bits. Here the number of rounds depends on the key length: 10 rounds for a 16-byte key, 12 rounds for a 24-byte key, and 14 rounds for a 32-byte key. The first rounds consist of four distinct transformation functions: Sub Bytes, Shift Rows, Mix Columns, and Add Round Key. The final round contains only three transformations.

Blowfish algorithm is a block cipher developed by Bruce Schneier in 1993 with the following characteristics: Speed, Compactness, Simplicity and Variably Secured. The main operation is XOR. It also uses 32-bits to 448-bits variable key lengths. It is a 16-round Feistel cipher and uses large key-dependent S-boxes.

- Prakash Kuppuswamy, Dr.Saeed Q Y Al-Khalidi, in the year 2012 proposed an algorithm based on Modulo 37 [4]. This algorithm uses two keys: k1=positive number, K2=negative number, find the inverse of both using modulo 37, giving k1', K2'.
- A Symmetric Key Cryptographic Algorithm by Ayushi, 2010[5]. It Generate the ASCII value of the letter and corresponding binary value and reversing the binary. There is no standard key generation method.
- An Efficient Developed New Symmetric Key Cryptography Algorithm For Information Security By Suyash Verma, Rajnish Choubey, Roopali Soni, July 2012[6]. It is a block cipher that divides data into blocks of equal length and then encrypts each block using a special mathematical set of functions known as key.
- A Modified Approach For Symmetric Key Cryptography Based On Blowfish Algorithm By Monika Agarwal, Pradeep Mishra, August 2012[7]. It is a 64 bit block cipher with a variable key length.

2.3. Fibonacci sequence [8]

Fibonacci[2] (1170-1230) introduced Arabic numerals to Europe. His theorem gives a sequence (the Fibonacci sequence) in which "each number is the sum of the two preceding numbers". Thus, the sequence progresses: 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597... .

2.4. Lucas numbers [9]

Francois-Edouard-Anatole Lucas is the French mathematician, professor. Lucas is studied the Fibonacci sequence and proposed Lucas sequence. Lucas Series is the sequence of numbers 1, 3, 4, 7, 11, 18, 29, 47, given with the following formula:

$L_n = L_{n-1} + L_{n-2}$ for $n > 2$, $L_1 = 2$, $L_2 = 1$

For the initial terms $L_1 = 1$ and $L_2 = 3$.
Example: LUCAS [2] NUMBER SEQUENCE which are
1, 3, 4, 7, 11, 18, 29, 47, 76, 123, 199, 322, 521, 843, 1364, 2207, 3571, 5778, 9349, 15127, 24476, 39603, 64079 (23 numbers from LUCAS3 NUMBER SERIES)

2.5. The double-reflecting data perturbation method [10]

The Double-Reflecting Data Perturbation Method[7] denoted by DRDP reverberates the original data by x-axis and y-axis to achieve the perturbed data for some confidential attribute. In this method, the randomization function plays a very crucial rule, and if the function is not properly chosen it May degrade the clustering quality. The distortion operation performed to the confidential attribute is given by

$o\rho_j = \rho_{Aj} + (\rho_{Aj-aj}) = 2\,\rho_{Aj-aj}$.

Where Aj ($1 \leq j \leq n$) is a confidential attribute and a j ($1 \leq j \leq n$) is an instance of Aj. ρ_{Aj} is defined by the following formula

$\rho_{Aj} = |(\max Aj + \min Aj)/2|$

Where max Aj and min Aj are respectively the maximum value and minimum value of attribute Aj. The 'student' relational database before and after applying DRDP is shown in the following Table:

Table 1: Example on DRDP Method

S.No	Roll No	Name	Marks	Distorted Marks
1	101	Rohan Raj	78	92
2	102	Shashank Raj	89	81
3	103	Prudvi Raj	92	78
4	104	Dhan Raj	82	88
5	105	Mythily Raj	80	90

2.6. Database creation for security enhancement [11]

This phase is implemented for security enhancement. In the proposed experimental setup both the sender and receiver has shared 32 files. Both the sender and receiver should use same database and a file with both users should be of same name. The Primary goal is to provide protection in data communication through Internet. In such environment, the suitable algorithms should be used which provides security to our sensitive data. For data security, many approaches have been adopted.
These 32-shared-Secret-Files are used for generating and supplying Round-Key for different Rounds in Encryption/Decryption Process.

3. Proposed system

NARSKCA is based on various cryptography algorithms. It uses different techniques like DRDP Converting technique, bitwise XOR, Fibonacci & Lucas Number Series. NARSKCA uses 32-Shared-Secret-Files between sender and Recipient. NARSKCA Sub-Key Generation, Final-Key Generation, encryption and decryption are describes in this section.
In this Research Paper following parameters are used as input for algorithm:
r: Number of encryption rounds.
s: Size of the Plain-Block
K_R: Round-Key for round
K_F: Final-Key for Function F
K_1, K_2: Round Keys for encryption and decryption process

3.1. Round-key generation

The details of the different types of keys used in NARSKCA:
Master Key Files: In NARSKCA, there is a need to share 32-Shared-Secret-Files between Sender and Recipient. Here the size of the files may equal or not. Here there is not restriction on the size of the files. When the sizes are random then it gives more security to our Data. The 1st character in all 32-Shared-Secret-Files formed as a Round-Key for 1st Round. The 2nd character in all 32-Shared-Secret-Files formed as a Round-Key for 2nd Round and so on.
Round-Keys (K_1 ….. k_{16}): In Encryption and Decryption process, 16 Rounds are used for processing the given Input Block. The Round-Key generation is as follows:

- For 1st Block & 1st Round, it takes 1st character in all 32-Shared-Secret-Files.
- For 1st Block & 2nd Round, it takes 2nd character in all 32-Shared-Secret-Files.
- For 1st Block & 3rd Round, it takes 3rd character in all 32-Shared-Secret-Files.
- For 2nd Block & 1st Round, it takes 17th character in all 32-Shared-Secret-Files.
- For 2nd Block & 2nd Round, it takes 18th character in all 32-Shared-Secret-Files.
- For 3rd Block & 1st Round, it takes 33rd character in all 32-Shared-Secret-Files.
- For 3rd Block & 2nd Round, it takes 34th character in all 32-Shared-Secret-Files.

And so on.
During the Round-Key Generation, if all characters are over in any file then it starts from 1st character in respective file. In this way, it simply generates infinite number of Round-Keys in both Encryption and Decryption Process.
Example: The Round-Key Generation for 1st Block and 1st Round in Encryption/ Decryption.
32-Shared-Secret-Files

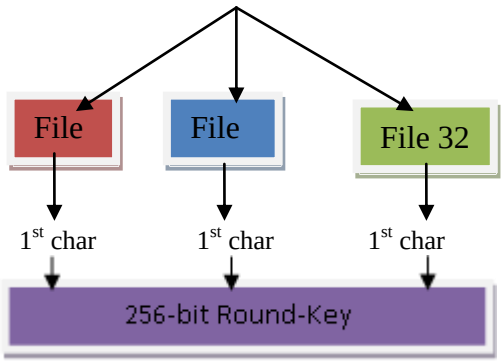


Fig. 1: Generation of Round-Key.

3.2. Final-key generation for round

The Final-Key for Round is based on the Round-Key. Here the size of Round-Key is 256-bit (32-Character). Here the Final-Key Generation for Round is depends on 2 phases.
Phase 1: From the Round-Key, it selects some of the characters using the Fibonacci Number Series: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610 … and so on. The Intermediate-Key generation is in the following way:
Step 1: Initialize f0=0, f1=1 for generating Fibonacci Number Series
Step 2: Initialize count=1
Step 3: Select the character from the Round-Key based on the index = f1 Mod 32
Step 4: If the character is repeated in the Intermediate-Key then this character is included to the Intermediate-Key. Otherwise it ignores the character.
Step 5: f2=f0+f1; f0=f1; f1= f2;
Step 6: if count not equal to 32 then go to Step 3.
Step 6: Intermediate-Key has been generated and its length is not fixed. For different Rounds, it may have different lengths.
Phase 2: This phase is based on the Lucas Number Series. The Final-Key Generation is (based on the Intermediate-Key) as follows:
Step 1: Initialize count=1, l0=2, l1=1
Step 2: Select the character from the Intermediate-Key based on Index=l1 mod size_of_Intermediate_key
Step 3: l2=l0+l1; l0=l1; l1=l2; count=count+1
Step 4: if count not equal to 32 then go to Step 2.
Step 5: 256-bit Final-Key has been generated.

4. NARSKCA encryption

- NARSKCA Encryption is depends in the following phases:
- DRDP Character Converting Technique
- Left Circular Shift
- Swap

4.1. DRDP-character converting technique

Here all 32-characters are converted according the Double-Reflecting-Data-Perturbation Method in the following way:
The distortion operation performed to the confidential attribute is given by
$o\rho j = \rho Aj + (\rho Aj - aj) = 2\,\rho Aj - aj$.
Where Aj ($1 \le j \le n$) is a confidential attribute and a j ($1 \le j \le n$) is an instance of Aj. ρAj is defined by the following formula

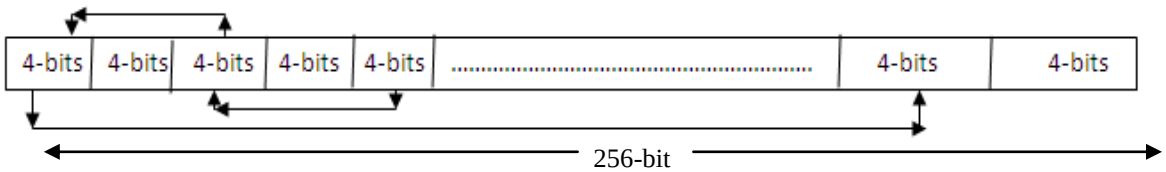
$$\rho Aj = |(\max Aj + \min Aj)/2|$$

Where max Aj and min Aj are respectively the maximum value and minimum value of attribute Aj.

4.2. Round function

In Encryption and Decryption Processes, two Rounds are included. The Pseudo code for single round is given below:

1) Each Round takes Converted Block after DRDP method.
2) 256-bit text is divided into 4-bit blocks. The odd numbered 4-bit blocks are Circular Rotate Left in the following way:



3) Apply DRDP character converting technique on the above text.

4.3. Algorithm for NARSKCA encryption process

The Pictorial representation of the NARSKCA Encryption Process is in the following way:

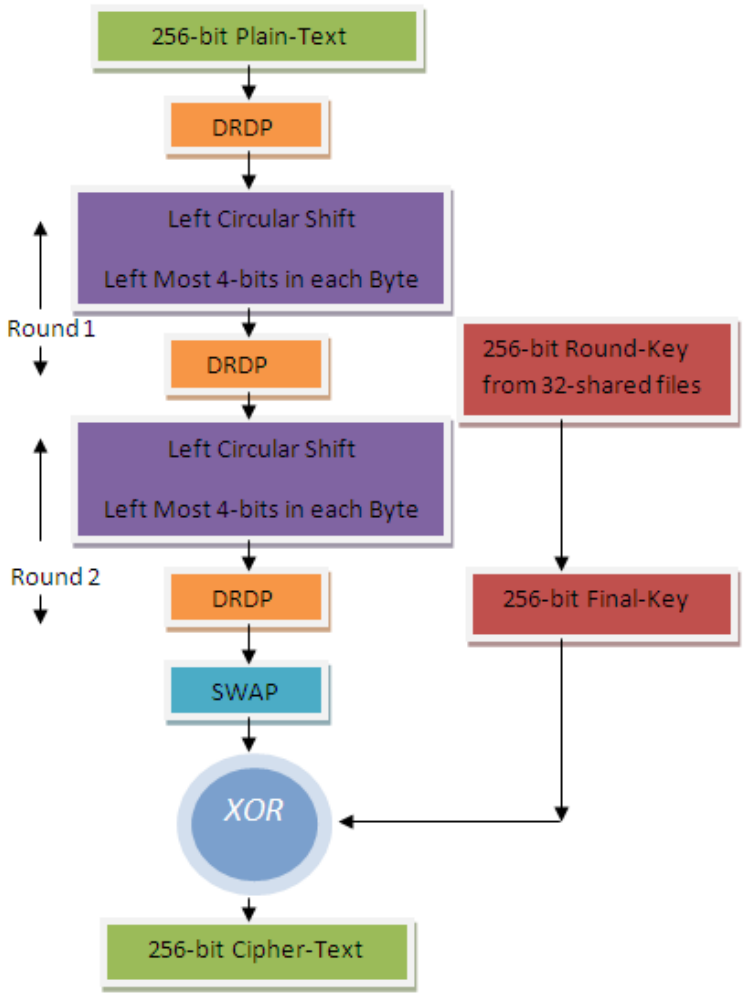


Fig. 2: Overview of Encryption Process.

Step 1: Divide the Input-Text into 256-bit Blocks i.e P [0], P [1]…, P [n]
Step 2: Take 256-bit Input-Block
Step 3: Apply DRDP Converting Technique on Step 2 Input-Block
Step 4: Perform 2 Round Operations on Step 3 Input-Block in the following way:

- Divide 256-bit Block into 4-bit chunks. Make odd numbered chunks Left Circular Shift.
- Apply DRDP Character Converting technique.

Step 5: The Intermediate-Text is swapped in the following way:

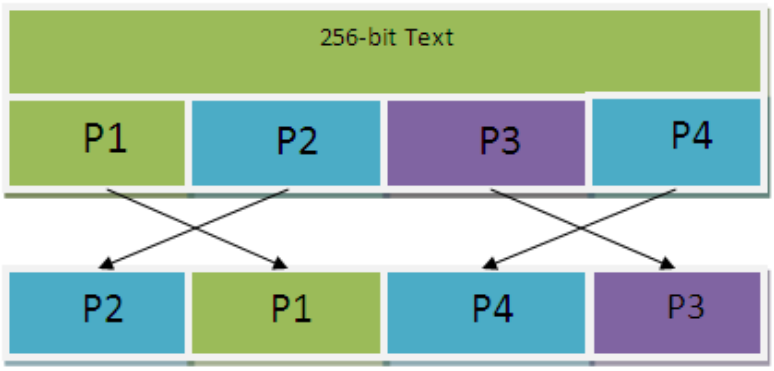


Fig. 3: Swapping of 256-Bit Data.

Step 6: Apply bitwise XOR on Step 5 Data-Block and Final-Key, which is calculated from Round-Key.
Step 7: Now it gives Cipher-Block.

5. NARSKCA decryption process

The NARSKCA Decryption Process is in the following way:

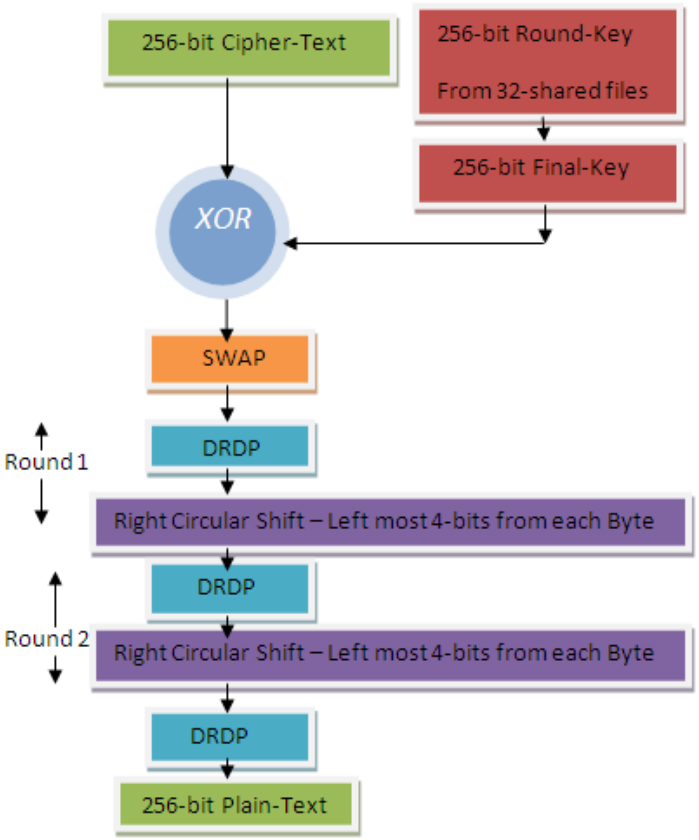


Fig. 4: Overview of Decryption Process.

Step 1: It takes the Cipher-Block as Input
Step 2: Apply bitwise XOR on Cipher-Block and Final-Key which is generated from 256-bit Round-Key.
Step 3: Apply swap operation in the following way:

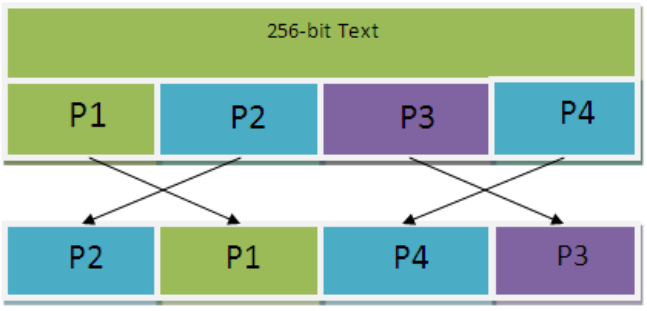


Fig. 5: Swapping of 256-Bit Data.

Step 4: Apply DRDP character converting technique on step 3 Text-Block.
Step 5: Perform 2 Round Operations on Step 4 Input-Block in the following way:

- Divide 256-bit Block into 4-bit chunks. Make odd numbered chunks Right Circular Shift.
- Apply DRDP Character Converting technique.

Step 6: Now it outputs the 256-bit Plain-Text

6. Analysis

In this Section, the proposed algorithm NARSKCA is analyzed. The proposed system is analyzed with the help of the following environment:
The CONFIGURATION of the Computer System where this proposed algorithm has been executed:

- Processor: Intel Core 2 Duo E7500@2.93GHz
- RAM: 2 GB
- Operating System: MS Windows XP
- Hard-Disk: 500 GB
- Java software Jdk 1.5

The following advantages have been identified from the proposed system NARSKCA:

1) NARSKCA is simple in nature and secure form timing attacks. Here the similar Blocks will take different timings for encryption/decryption. Because the Round-Key is different for different Rounds and Blocks.
2) It provides more security for Data. Because it generates different Round-Keys for the similar types of Blocks in encryption/decryption.
3) According to our results, the proposed encryption/decryption processes satisfies strong avalanche effect. For one bit change in plain text, MSEA changes more than 50% bits of cipher text after 12 rounds.
4) In NARSKCA, DRDP Character Converting Technique provides strong diffusion by creating the nonlinearity in message.
5) In NARSKCA, the Round-Key and Final-Key generation provides strong confusion.
6) Here we can't apply and compare with method "STANDARD FRENQUENCY DISTRIBUTION for ENGLISH"
7) Here the Round-Keys are depends on 32-Shared-Secret-Files. So not possible to predict the Round-Keys as well as Final-Keys.
8) Performance analysis with respect to Number of Rounds:

Table 2: NARSKCA Rounds/Encryption-Time (Ms)/ Decryption-Time (Ms)

Number of Rounds	Enc-Time(ms)	Dec-Time(ms)
16	12688	12759
8	6447	6525
4	3215	3282
2	1657	1698

The corresponding Graph is as follows:

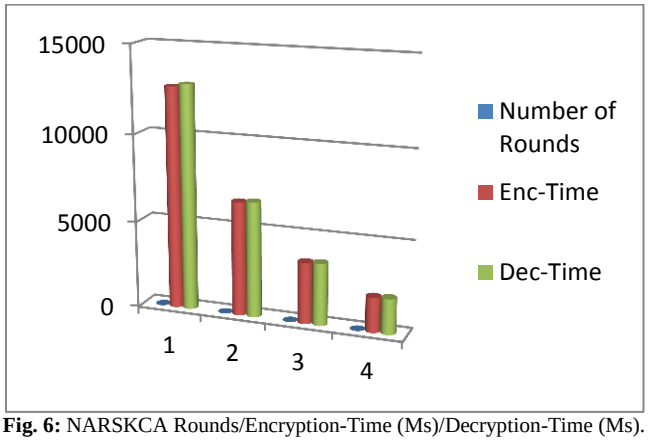


Fig. 6: NARSKCA Rounds/Encryption-Time (Ms)/Decryption-Time (Ms).

9) Performance analysis between DES, TDES, AES, IDEA & NARSKCA (new).

Table 3: Performance Analysis of DES, TDES, AES, IDEA & NARSKCA (New)

Algorithm	Enc-Time(ms)
DES	1543
TDES	1701
AES	1387
IDEA	1484
NARSKCA(New)	1557

The Graphical representation is as follows:

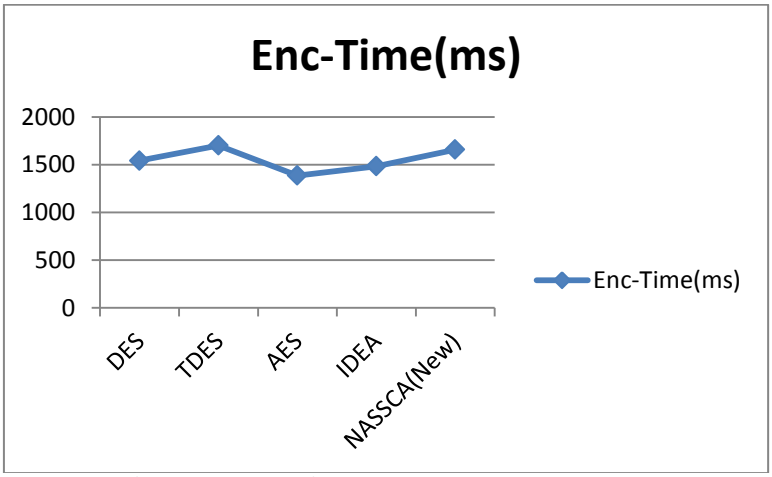


Fig. 7: Performance Analysis of DES, TDES, AES, IDEA & NARSKCA (New).

In NARSKCA (Proposed), the encryption/decryption time is more. But it provides more security to the Data.

7. Security level

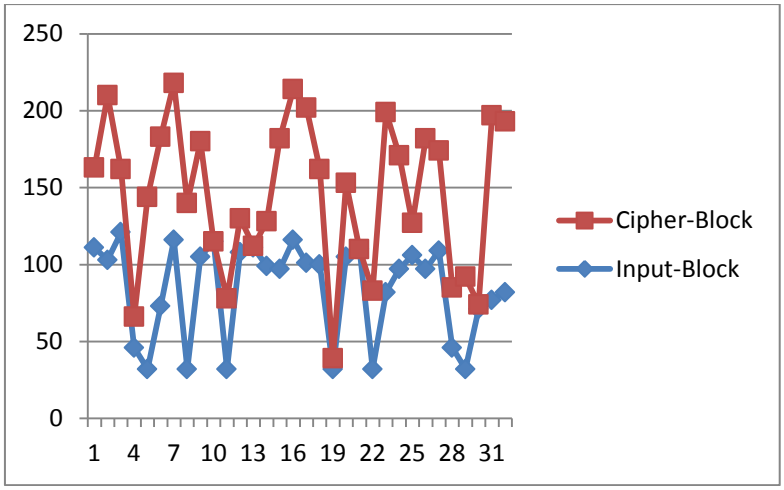
In proposed method, all the characters in the sentence are converted based on Double Reflecting Data Perturbation Method (DRDP). The 256-bit data is processed in 2 Rounds. The converted Data go to Swap and DRDP. Here the privacy of data is measured by the variance between the actual and the perturbed values which is given by the following formula:

$$A = \frac{VAR(A-A')}{VAR(A)}$$

It has been analyzed that the privacy or the security level of the confidential data is improved a lot by the proposed method for Encryption and Decryption [12], [13].
The relation between Plain-Text and Cipher-Text is as follows (Some Input-Block):

Table 3: Avalanche Effect between Plain-Text and Cipher-Text

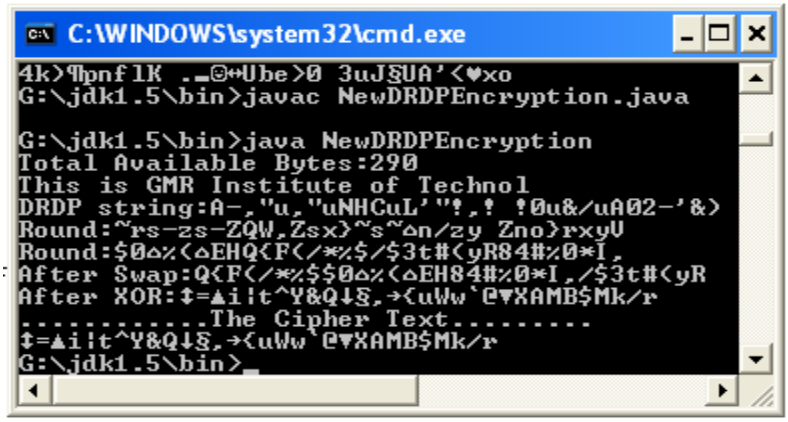
Input-Block	111	103	121	46	32	73	116	32	105	115	32	108	111	99	97	116
	101	100	32	105	110	32	82	97	106	97	109	46	32	71	77	82
Cipher-Block	52	107	41	20	112	110	102	108	75	0	46	22	1	29	85	98
	101	62	7	48	0	51	117	74	21	85	65	39	60	3	120	111



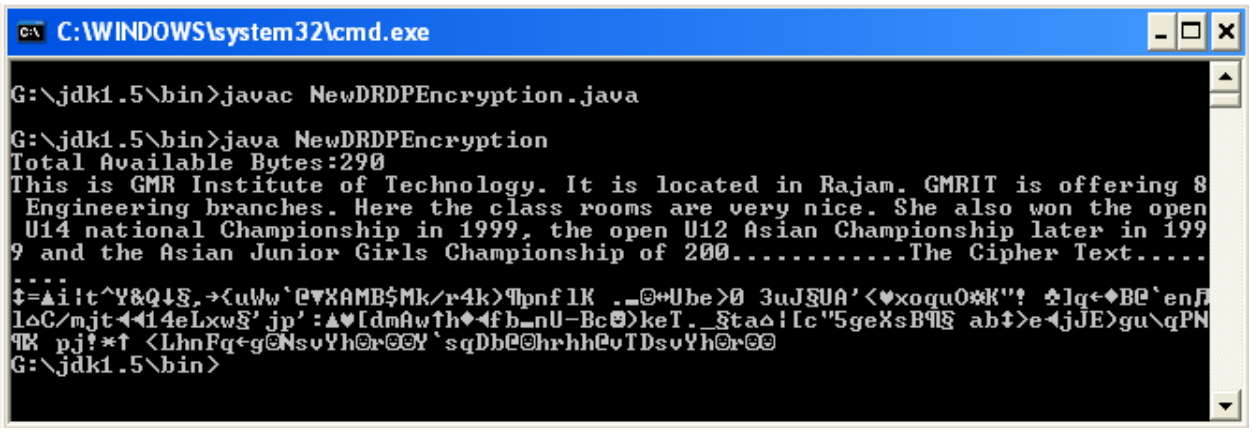
8. Illustration

Step 1: Initialize 32-Shared-Secret-Files between sender and recipient.
Step 2: Take Input-Text:

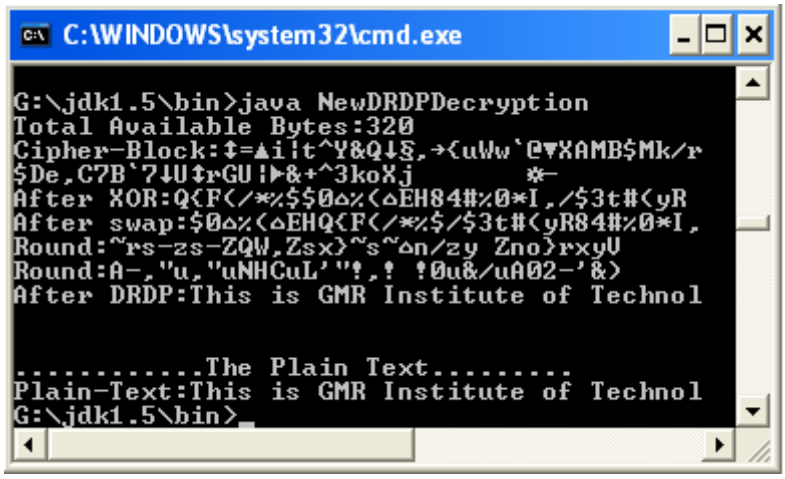
"This is GMR Institute of Technology. It is located in Rajam. GMRIT is offering 8 Engineering branches. Here the class rooms are very nice. She also won the open U14 national Championship in 1999, the open U12 Asian Championship later in 1999 and the Asian Junior Girls Championship of 2000."
Step 3: The first 256-bit Input-Block "This is GMR Institute of Technol" is encrypted in the following way:



Here the first character 'T' is converted in encryption process into the following way:
A, ~, $, Q, ↕
So the Input Character 'T' is not repeated in the encryption/decryption process.
Step 4: The Cipher-Text of first Input-Block is "↕=▲i|t^Y&Q↓§, → {uWw`@▼XAMB$Mk/r"
Step 5: The Cipher-Text of the entire file is



Step 6: The above Cipher-Text is decrypted in the following steps.
Step 7: The first Cipher-Block is decrypted in the following way:



Step 8: After decryption process, the first Plain-Text is "This is GMR Institute of Technol"

References

[1] W. Diffie and M. Hellman, "New directions in cryptography", IEEE Transaction on Information Theory, 1976, pp. 644–654. http://dx.doi.org/10.1109/TIT.1976.1055638.

[2] http://en.wikipedia.org/wiki/Communication.

[3] "Data Encryption Standard," Federal Information Processing Standards Publication No. 46, National Bureau of Standards, January 15, 1977.

[4] Prakash Kuppuswamy, Dr. Saeed Q Y Al-Khalidi, "Implementation of Security through Simple Symmetric Key Algorithm Based On Modulo 37", International Journal of Computers & Technology, ISSN: 2277-3061, Volume 3, OCT 2012.

[5] Ayushi, "A Symmetric Key Cryptographic Algorithm" International Journal of Computer Applications (0975-8887), Volume 1, 2010.

[6] Suyash Verma, Rajnish Choubey, Roopali Soni, "An Efficient Developed New Symmetric Key Cryptography Algorithm for Information Security", July 2012.

[7] Monika Agarwal, Pradeep Mishra, "A Modified Approach for Symmetric Key ryptography Based On Blowfish Algorithm", August 2012.

[8] http://protea.worldonline.co.za/fibon.htm.

[9] http://milan.milanovic.org/ math /english/fibo/ fibo3.html.

[10] A. Viji Amutha Mary, Dr. T. Jebarajan, A NovelData Perturbation Technique with higher Security, IJCET, 3(2): 126-132 (2012).

[11] B. Santhi, K.S. Ravichandran, A.P. Arun and L. Chakkarapani "A Novel Cryptographic Key Generation Method Using Image Features", Research Journal of Information Technology 4(2): 88-92, 2012, ISSN: 2041-3114.

[12] BALAJEE MARAM, Dr CHALLA NARASIMHAM, "Double Reflecting Data Perturbation Method for Information Security", OJCST, Dec' 2012, Vol: 5, No.2, Pgs: 283-288.

[13] http://www.cse.unr.edu/~bebis/CS302/image_info.html.