International Journal of Research in Advanced Engineering and Technology
ISSN: 2455-0876
www.engineeringresearchjournal.com
Volume 2; Issue 1; January 2016; Page No. 22-29

NASSCA: Novel and sturdy symmetric cryptographic algorithm

V Kumar, Balajee Maram, Satish Muppidi

[1]Asst. Prof., Dept. of Computer Science, Central University of Kerala, Kerala, India.

[2] Sr. Asst. Prof., Dept. of CSE, GMRIT, Rajam, India.

[3] Asst. Professor, Dept. of IT, GMRIT, Rajam, India.

Abstract
In this Research paper, a novel and strong symmetric cryptographic algorithm is proposed. NASSCA is based on several symmetric cryptographic algorithms. NASSCA is very simple that uses character Converting algorithm, Fibonacci Number Series, Lucas Number series and bitwise XOR. In NASSCA, shared secret files play a vital role in this Proposed Algorithm. The Sub-keys generation depends on 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 in each Round. Key Generation, encryption/decryption schemes of NASSCA are fast and difficult to predict by Cryptanalysts. The feature of the proposed system is Avalanche Effect is more than some of the existing algorithms.

Keywords: DRDP, Symmetric Cryptography, Block Key Cipher, Lucas, NASSCA

1. Introduction

Till now so many cryptography algorithms have been introduced and proposed. "bit-scramble" plays a vital role in cryptography algorithms. The traditional cryptography algorithms worked on bit-scramble, but it is not up to the benchmark. There is a need to increase the rate of bit-scrambling.

In best cryptography algorithms, bit-scramble plays an important role. It means, a small change in input data/key reflects more than half of the bits in output. This is called "Avalanche Effect". This is very important property for any cryptography algorithm.

Avalanche Effect is very important property in any cryptography algorithm. But in traditional cryptography algorithms, Avalanche Effect is very less. The technique of traditional cryptography algorithms is very effective but the parameter Avalanche Effect is not good. Implementation of cryptography algorithms with good Avalanche Effect is not so easy, because a single bit reflects more than half of the bits in output.

Avalanche Effect is an important parameter for any cryptography algorithms. Many algorithms are successful in Avalanche Effect. But rate of Avalanche Effect is up to 60% to 65% only. It's been solved but not up to the benchmark.

The proposed system worked on bit-manipulation concept and mathematical operations. Comparatively the proposed system is best algorithm with existing cryptography algorithm in terms of Avalanche Effect.

A. Cryptography

Cryptography is the study and practice of encoding data using transformation techniques so that it can only be decoded by specific users. In simpler words, it is a theory of secret writing. Cryptography is the systems involving two kinds of security problems: privacy and authentication. A privacy system prevents the extraction information by unauthorized parties from messages transmitted over a public channel, thus assuring the sender of a message that it is being read only by the intended recipient.

Types of Cryptography

There are two main types of cryptography:

a) Secret key cryptography
b) Public key cryptography

B. 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". This is also called Symmetric key cryptography.

C. Public-key Algorithms

Public key cryptography – In this method, there is a need of two keys. The first key is known as "Public-Key" and the Second key is known as "Secret-Key". The plain text is encrypted with private-key of sender (or public-key of receiver). In receiver side, the cipher text is decrypted with public-key of sender (or private-key of receiver). Here, it is not necessary for the sending and receiving users to share the common secret. Here the recipient should distribute his own public-key.

D. Confusion and Diffusion

Confusion: A technique that seeks to make the relationship between the statistics of the cipher-text and the value of the encryption keys as complex as possible. Cipher uses key and plaintext. It should be very difficult to find out the key even the attacker has large number of plaintext and cipher-text pairs that are produced by the same shared secret key. So each bit of cipher-text is based on key.

Diffusion: A technique that seeks to obscure the statistical structure of the plaintext by spreading out the influence of each individual plaintext digit over many cipher-text digits. 1-bit change in plain-text should reflect more than half of the bits in cipher-text.

E. Avalanche Effect

An important property of any encryption algorithm is that a small change in either the plain-text of the key must produce a significant change in cipher-text. 1-bit change either plain-text/key should reflects more than half of the bits in cipher-text are known as "Avalanche Effect".

$$\text{Avalanche Effect} = \frac{\text{Number of bits flipped in cipher-text}}{\text{Number of bits in cipher-text}}$$

F. Fibonacci sequence

Fibonacci2 (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... .

G. Lucas Numbers

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:

Ln = Ln-1 + Ln-2 for n > 2, L1 = 2, L2 = 1 for the initial terms L1 = 1 and L2 = 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)

H. The Double-Reflecting Data Perturbation Method

The Double-Reflecting Data Perturbation Method7 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 ($1 \le j \le n$) 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.

2. Database creation for Security Enhancement

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. Literature survey

The proposed concept in [1] provides architecture for confidentiality of text data with good Avalanche-Effect in public domain that can be useful in various software applications like banking, medical, government organization, defense and many more.

The proposed algorithm [2] is not fully depended on secret key and for the same plain text it produces different cipher text using the same secret key which reduces the probability of various attacks.

The proposed algorithm [3] has better Avalanche Effect than any of the other existing algorithms and hence can be incorporated in the process of encryption of any plain text.

Triple SV [4] has a way better Avalanche Effect than any of the other existing algorithms and hence can be incorporated in the process of encryption of any plain text. The high avalanche ratio and a key size of 112 bits ensure sound security from brute force attacks.

This paper [5] presents a performance evaluation of selected symmetric encryption algorithms. The selected algorithms are AES, DES, and DES in Feistal structure, AES in Feistal structure and Hybrid AES-DES [21] structure. The performance evaluation has been done based on parameters: Avalanche Effect [22], Throughput, CPU Usage, Encryption and Decryption Time. The Avalanche Effect is better in Hybrid AES-DES algorithm than other algorithms.

The avalanche effect exhibited by blowfish algorithm [6] is very strong. Approximately 50% cipher-text bits differ after every round. Also I can see that avalanche effect is stronger when plaintext is changed than the change in key.

The performance of proposed algorithm (Enhancement of AES algorithm) in [7] is evaluated using Avalanche Effect due to one bit variation in plaintext (before being mapped in various binary codes) keeping encryption key constant in a binary code and Avalanche Effect due to one bit variation in encryption key (before being mapped in various binary codes) keeping plaintext constant in a binary code. This leads significant increase in Avalanche Effect of AES Algorithm

The classical ciphers [8] like Playfair cipher, Vigenere Cipher, Caesar Cipher etc. have very less Avalanche Effect and hence cannot be used for encryption of confidential messages. The modern encryption techniques are better than classical ciphers as they have higher Avalanche Effect.

In the proposed algorithm [9], we have mapped input plaintext and encryption key into various binary codes, instead of giving plaintext directly to the DES algorithm. This leads significant increase in Avalanche Effect of encryption algorithm. We got maximum avalanche effect of 44/64, when key is mapped in Gray code and Data is mapped in 5421.

The proposed algorithm [10] has better Avalanche Effect as well as execution time than any of the other compeering algorithms and hence can be incorporated in the process of encryption of any plain text.

The AES [11] provides a reasonably high level of security with efficient implementation, and it is likely to remain a strong algorithm for some time to come. This paper presents the implementation of AES algorithm which also shows the Avalanche effect is good.

The key avalanche effect [12] produced on data blocks in diffusion rounds has improved in AES with Matrix based key generation procedure.
E-AES [13] performs better when compared with the existing AES is reflected by the corresponding Avalanche Effect of both the existing and proposed E-AES.

4. Proposed system

NASSCA is proposed algorithm and is derived from existing techniques. Merely it uses DRDP conversion method, bitwise XOR, Number system. NASSCA Sub-Key Generation, Final-Key Generation, encryption and decryption are explained in the following:

A. Round-Key Generation

The details of key generation in NASSCA are as follows:
Here the key generation is based on 32 shared secret files. From 32 shared secret files, the Round-Key generation is as follows:
Round-Keys (K1 ….. k16): 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.

B. 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.

C. NASSCA encryption

NASSCA Encryption is explained in the following:

Round Function

The Pseudo code for single round is given below:

1) Each Round takes Converted Block after DRDP method.
2) Each Round takes 256-bit Round-Key from 32-Shared-Secret-Files
3) Function F
4) Output of Function F is divided into two halves i.e A, B.
5) 256-bit Round-Key is converted according to DRDP method and divided into two halves i.e C, D.
6) E:= A $\oplus$ D
7) F:= B $\oplus$ C
8) Concatenate E and F

The steps in Function F:

1) Function F takes Converted Block after DRDP method. Take it as ‘A’
2) Take the 256-bit Round-Key
3) Calculates the Intermediate-Key using Fibonacci Series in the following way:

The Fibonacci Series: 1,1,2,3,5,8,13,21,34,55,89,144,… so on. It picks the character from the Round-Key which is having the index value is equal to (Fibonacci Number) mod 32.
The index values are 1,1,2,3,5,8,13,21,2,23,25,16, … so on. So it takes the characters from the Round-Key are Round-Key [1], Round-Key [1], Round-Key [2], … up to 32 characters. Here the characters should be unique.

4) Now the size of Intermediate-Key is variable to different Blocks.
5) The Final-Key generation is based on Lucas Number Series: 1, 3, 4, 7, 11, 18, 29, 47, 76, 123, 199, 322, 521, 843, 1364, 2207, 3571, 5778, 9349, 15127, 24476, 39603, 64079, … so on.
6) It picks the characters from the Intermediate-Key which is having the index value is equal to (Lucas Number) mod (size of Intermediate-Key)
7) Now the Final-Key is generated and its length is 256-bit. Take it as ‘B’
8) Apply bitwise XOR operation on Step 1 and Step 7 Blocks. C=A $\oplus$ B

Algorithm for NASSCA Encryption Process

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

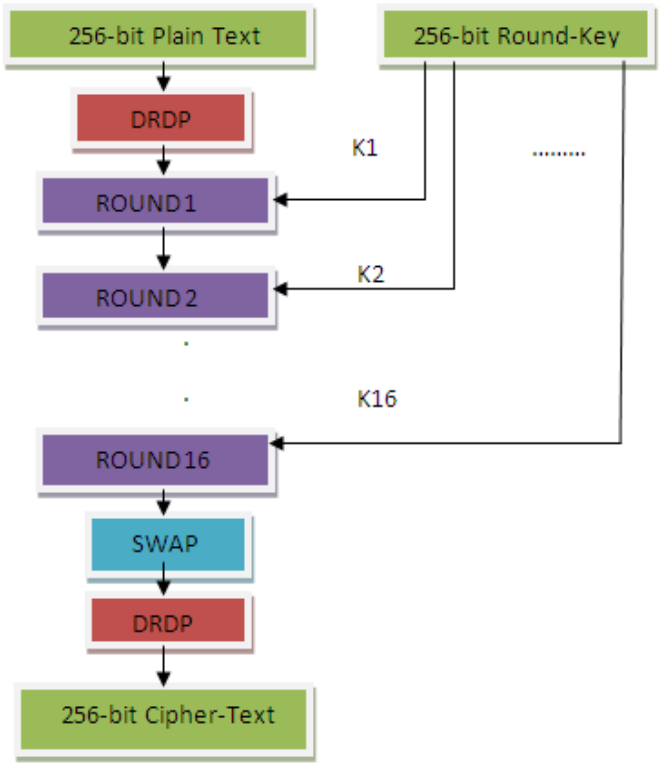


Fig 1: Overview of Encryption Process

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 16 Round Operations on Step 3 Input-Block in the following way:

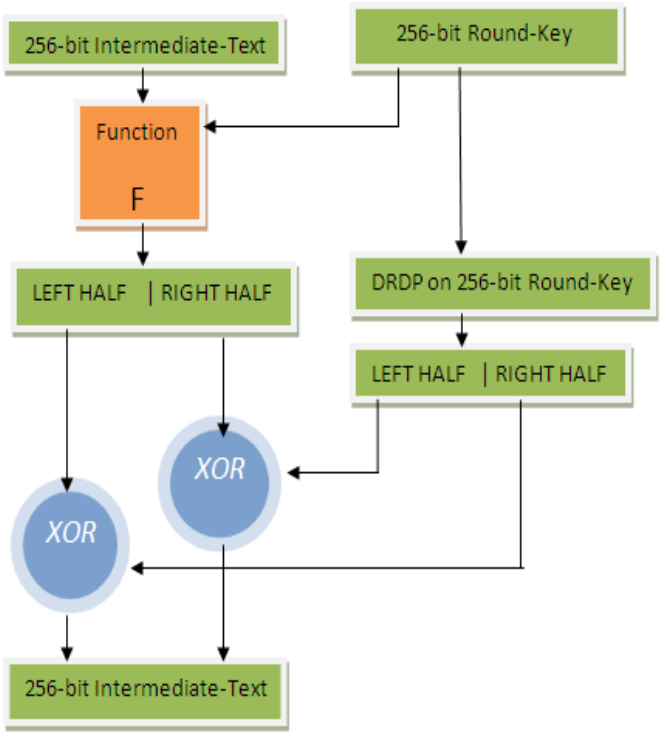


Fig 2: Overview of Round 1 .. 16

Here the Function F is defined in the following way:

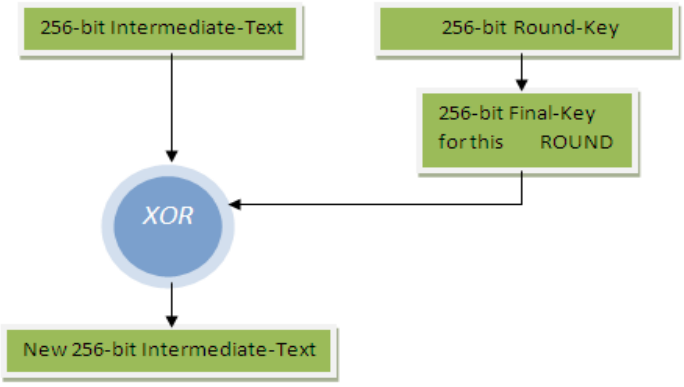


Fig 3: Functionality of F

Step 5: After completion of 16 Rounds, the Intermediate-Text is swapped in the following way:

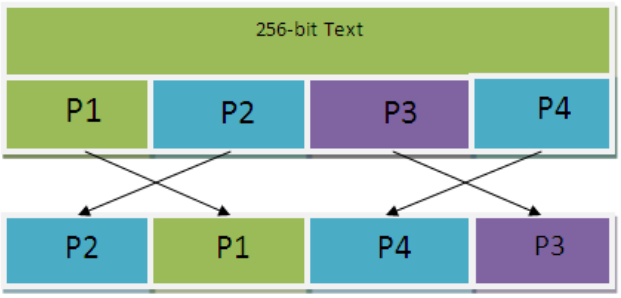


Fig 4: Swapping of 256-bit Data

Step 6: Apply DRDP Character Converting Method on Step 5 Data-Block
Step 7: Now it gives Cipher-Block.

D. NASSCA decryption process

The NASSCA Decryption Process is in the following way:

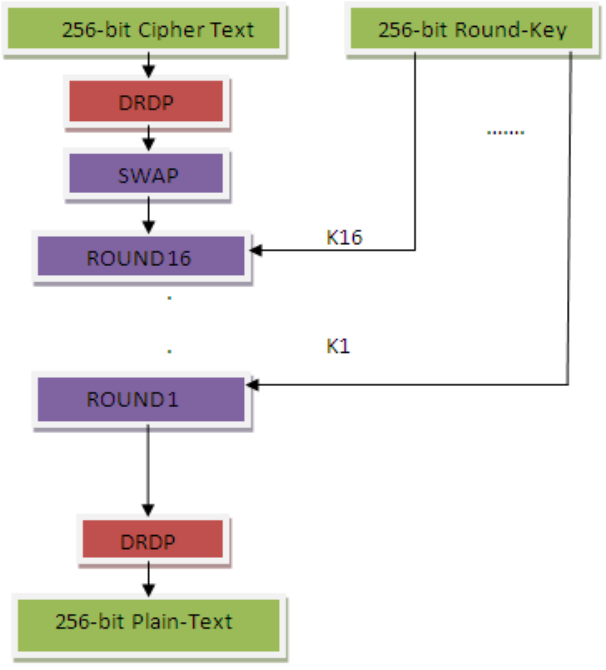


Fig 5: Overview of Decryption Process

Step 1: It takes the Cipher-Block as Input
Step 2: Apply DRDP Character Converting technique on the Block
Step 3: Apply swap operation in the following way:

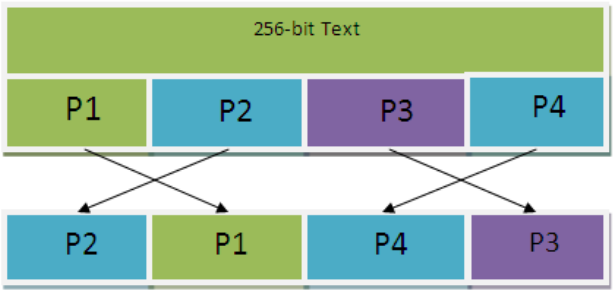


Fig 6: Swapping of 256-bit Data

Step 4: Apply Round 16 to Round 1 operations in the following way:

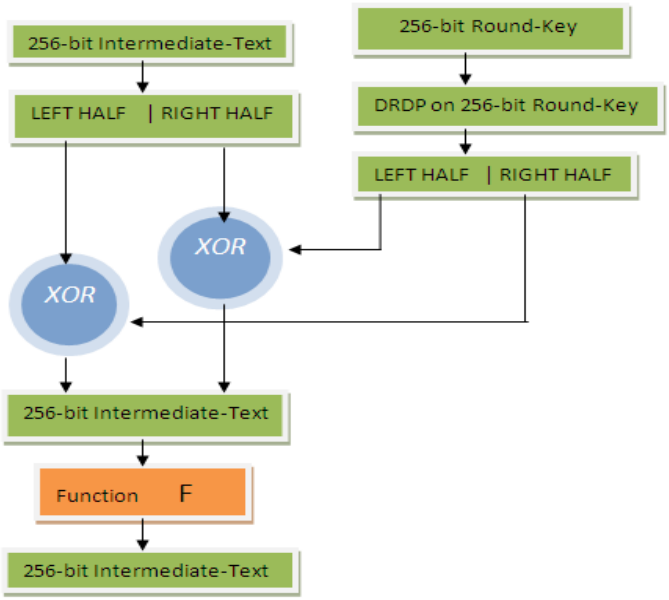


Fig 7: *Overview of* Round 16..1

The functionality of Function F:

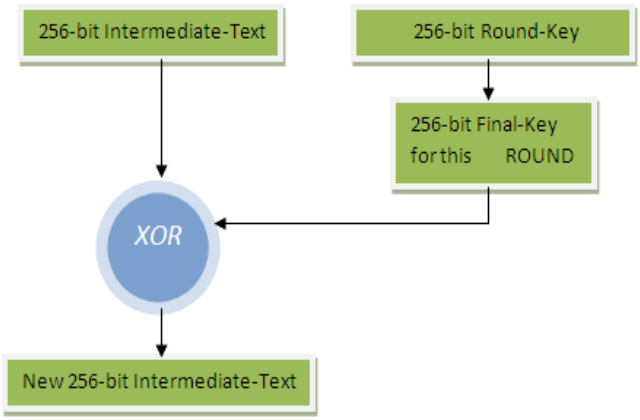


Fig 8: *Functionality of F*

Step 5: Apply DRDP Character Converting technique
Step 6: Now it outputs the 256-bit Plain-Text

5. Analysis

In this Section, the proposed algorithm NASSCA is analyzed.

A. Performance analysis between DES, TDES, AES, IDEA & NASSCA (new)

Table 1: Performance analysis of DES, TDES, AES, IDEA & NASSCA (new)

Algorithm	Enc-Time(ms)
DES	1866
TDES	1923
AES	1696
IDEA	1759
NASSCA(New)	2000

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

B. Analysis on Avalanche Effect

Table 2: Comparison of Avalanche-Effect between proposed and existing algorithms

Encryption Technique	Avalanche Effect (%)
This Proposed system (Average)	***58.6***
DES	54.68
AES-RC4	52.34
Original AES	46.88
Blowfish	28.71
Playfair Cipher	6.25
Vigenere Cipher	3.13
Caesar Cipher	1.56

6. 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 16 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:

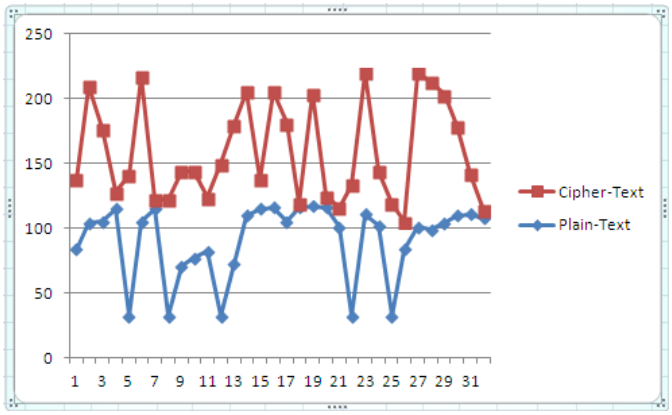
$$A = \frac{VAR(A - A')}{VAR(A)}$$

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 4: Relation between Plain-Text and Cipher-Text

Plain-Text	84	104	105	115	32	105	115	32	71	77	82	32	73	110	115	116
	105	116	117	116	101	32	111	102	32	84	101	99	104	110	111	108
Cipher-Text	54	106	72	13	109	112	7	90	73	67	42	118	107	96	23	90
	76	3	87	9	15	102	109	42	87	21	119	114	99	69	31	6



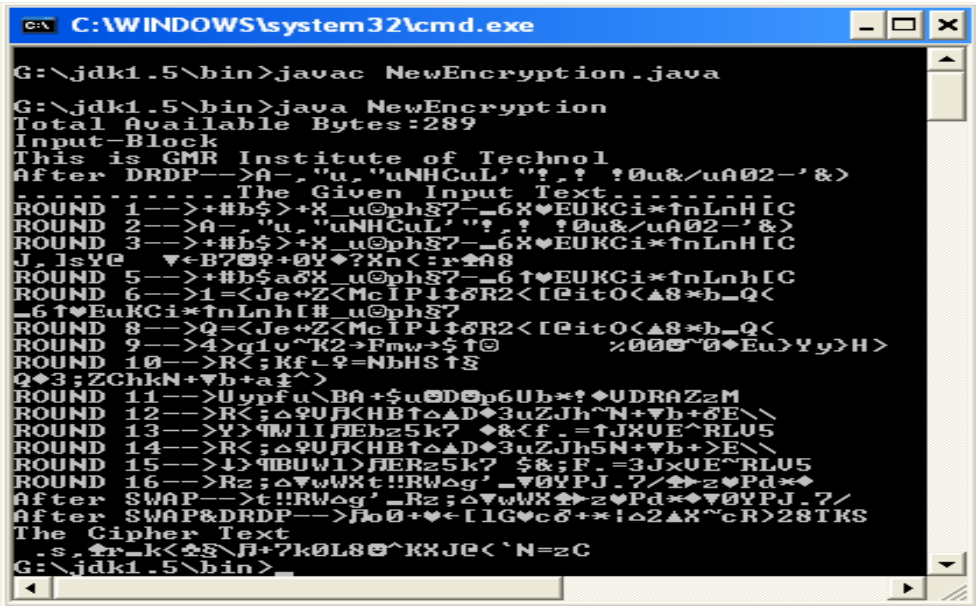
7. Illustration

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:



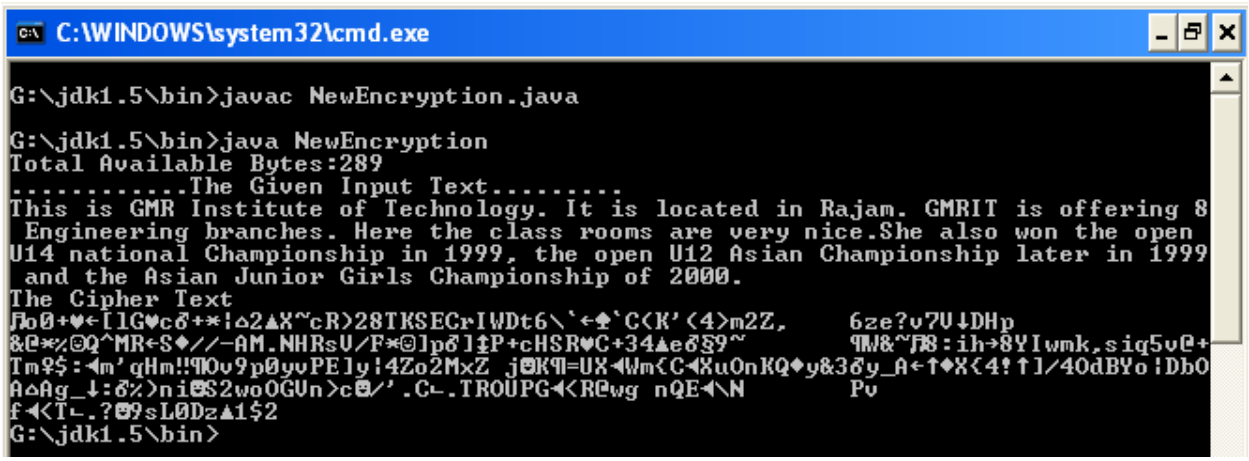
Here the first character 'T' is converted in encryption process into the following way:

A,+,A,+,J,+,1,_,Q,4,R,Q,U,R,Y,R, ↓,R,t, ♫, .

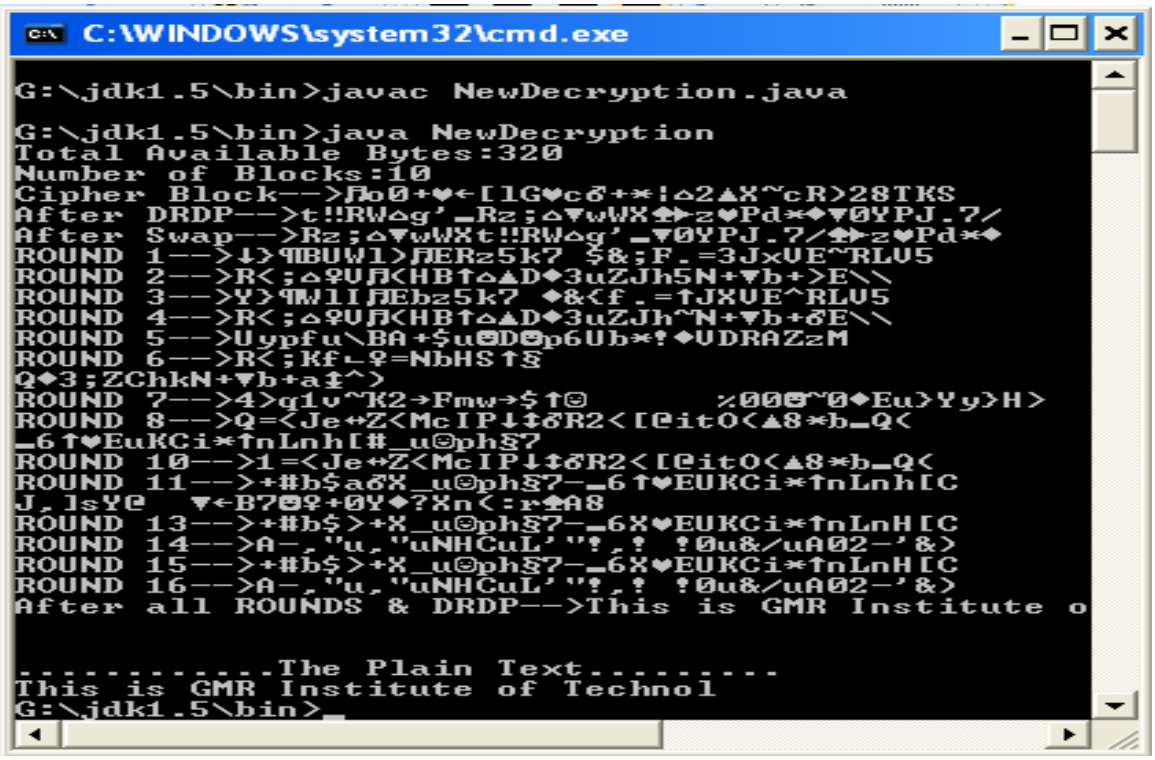
So the Input Character 'T' is not repeated in the encryption/decryption process.

Step 4: The Cipher-Text of first Input-Block is ".s,♠r▬k<♣§\♫+7k0L8☻^KXJ@(`N=zC"

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".

8. Conclusion

The proposed system adopts some of the best methods from the existing systems. It provides infinite number of Round-Keys for processing Encryption and Decryption process. It uses mathematical operations like bitwise XOR and DRDP character conversion technique. It strongly supports Confusion and Diffusion. Comparing with other symmetric key cryptography algorithms, the proposed algorithm NASSCA takes more time for encryption and decryption. When number of Rounds reduced then Encryption and Decryption Time will be reduced. But it will maintain consistent Security-Level. Comparing with existing algorithms, the Avalanche Effect is best in the proposed algorithm. In near future, this will be extended to Images using UNICODE.

9. References

1. Prof.Gajendra Singh, Preeti Shukla, "Design and Development of New symmetric Cryptography Protocol to Improve Text Security" published by International Journal of Advanced Research in Computer Science and Software Engineering, 2014;4(11), ISSN: 2277 128X.
2. Debajit Sensarma, Samar Sen Sarma, "GMDES: a graph based modified data encryption standard algorithm with enhanced security" published by IJRET: International Journal of Research in Engineering and Technology eISSN: 2319-1163 | pISSN: 2321-7308, 2014, 03(03)
3. Sriram Ramanujam and Marimuthu Karuppiah, "Designing an algorithm with high Avalanche Effect" published by IJCSNS International Journal of Computer Science and Network Security, 2011; 11:1.
4. Rajdeep Chakraborty, Sonam Agarwal, Sridipta Misra, Vineet Khemka, Sunit Kr Agarwal, J. K. Mandal, "Triple SV: A Bit Level Symmetric Block Cipher Having High Avalanche Effect" published by (IJACSA) International Journal of Advanced Computer Science and Applications, 2011; 2:7.
5. VikasKaul SK Narayankhedkar AdityaPatil, "Enhanced Data Encryption Algorithmfor Next Generation Networks" published by International Journal of Applied Information Systems (IJAIS) – ISSN: 2249-0868.
6. Manisha S. Mahindrakar. "Evaluation of Blowfish Algorithm based on Avalanche Effect" published by International Journal of Innovations in Engineering and Technology (IJIET), 2014; 4(1), ISSN: 2319–1058.
7. Chandra Prakash Dewangan, Shashikant Agrawal. "A Novel Approach to Improve Avalanche Effect of AES Algorithm" published by International Journal of Advanced Research in Computer Engineering & Technology. 2012; 1(8)
8. Ganesh Patidar, Nitin Agrawal, Sitendra Tarmakar, "A block based Encryption Model to improve Avalanche Effect for data Security" published by International Journal of Scientific and Research Publications. 2013; 3(1), ISSN 2250-3153.
9. Akash Kumar Mandal, Mrs. Archana Tiwari. "Analysis of Avalanche Effect in Plaintext of DES using Binary Codes" published by International Journal of Emerging Trends and Technology in Computer Science(IJETTCS), 2012; 1(3).
10. Parvez khan Pathan, Basant Verma. "Hyper Secure Cryptographic Algorithm to Improve Avalanche Effect for Data Security" published by International Journal of

Computer Technology and Electronics Engineering (IJCTEE) Volume 1, Issue 2.
11. Jayant P. Bhoge, Dr. Prashant N. Chatur, "Avalanche Effect of AES Algorithm" published by (IJCSIT) International Journal of Computer Science and Information Technologies, 2014; 5(3):3101–3103.
12. Paul AJ, Mythili P, Paulose Jacob K. "Matrix based Key Generation to Enhance Key Avalanche in Advanced Encryption Standard" published by International Journal of Computer Applications® (IJCA).
13. Ajeet Singh. "A New Approach to Enhance Avalanche Effect in AES to Improve Computer Security" published by Information Technology & Software Engineering, Volume 5 Issue 1.
14. Diffie W, Hellman M. "New directions in cryptography", IEEE Transaction on Information Theory, 1976, 644–654.
15. http://en.wikipedia.org/wiki/Communication
16. "Data Encryption Standard," Federal Information Processing Standards Publication No. 46, National Bureau of Standards, January 15, 1977.
17. Prakash Kuppuswamy, Dr. Saeed QY Al-Khalidi. "Implementation of Security through Simple Symmetric Key Algorithm Based On Modulo 37", International Journal of Computers & Technology, ISSN: 2277-3061, 2012, 3.
18. Ayushi. "A Symmetric Key Cryptographic Algorithm" International Journal of Computer Applications (0975-8887), 2010, 1.
19. Suyash Verma, Rajnish Choubey, Roopali Soni. "An Efficient Developed New Symmetric Key Cryptography Algorithm for Information Security", July 2012.
20. Monika Agarwal, Pradeep Mishra. "A Modified Approach for Symmetric Key ryptography Based On Blowfish Algorithm", August 2012.
21. http://protea.worldonline.co.za/fibon.htm
22. http://milan.milanovic.org/ math /english/fibo/ fibo3.html
23. A. Viji Amutha Mary, Dr. T. Jebarajan, A Novel Data Perturbation Technique with higher Security, IJCET, 2012; 3(2):126-132.
24. Santhi B, Ravichandran KS, Arun AP, Chakkarapani L. "A Novel Cryptographic Key Generation Method Using Image Features", Research Journal of Information Technology. 2012; 4(2):88-92. ISSN: 2041-3114.
25. Balajee Maram, Dr. Challa Narasimham. "Double Reflecting Data Perturbation Method for Information Security", OJCST, Dec' 2012; 5(2):283-288.
26. http://www.cse.unr.edu/~bebis/CS302/image_info.html.
27. Balajee Maram, Y Ramesh Kumar, Lakshmana Rao K. "NARSKCA: Novel and robust symmetric key cryptography algorithm", International Journal of Scientific World. 2015; 3(2):244-254.
Doi: 10.14419/ijsw.v3i2.5111