International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 03, March-2022

A New Cryptosystem using Oscillation Model

U. Thirupalu
Research Scholar (PT)
Dept. of Computer Science,
S.V.U. College of CM&CS
Tirupati-A.P.-INDIA

E. Spandhana
Business Analyst
ADECCO India PVT Ltd
Bangalore
KARNAKA-INDIA

Dr. E. Kesavulu Reddy Ph.D, FCSRC (USA)
Assistant Professor
Dept. of Computer Science,
S.V.U. College of CM&CS, Tirupati.
A.P.-INDIA.

***Abstract*: Data Encryption techniques is used to avoid the unauthorized access original content of a data in communication between two parties, but the data can be recovered only through using a key known as decryption process. The objective of the encryption is to secure or save data from unauthorized access in term of inspecting or adapting the data. Encryption can be implemented occurs by using some substitute technique, shifting technique, or mathematical operations. By adapting these techniques, we can generate a different form of that data which can be difficult to understand by any person. The original data is referred to as the plaintext and the encrypted data as the cipher text. Several symmetric key base algorithms have been developed in the past year. We proposed a new technique diagonal transposition with 256 bits different key values and generation of wave as in the form of cipher with variable length matrix to reduce the time complexity of simple column transposition techniques.**

Keywords: - Substitution Cipher, Transposition Cipher, Encryption, Decryption, Diagonal transposition technique.

I. INTRODUCTION

Cryptography is the art of achieve security by encoding messages to make them non-readable [1]. It is a technique which allow human-being to encrypt the data in such a way that the decryption can be achieved without the aid of sender. Cryptography not only protects the information but also provides authentication to the user. As the network technology has been greatly advanced, there is a need to send much information via the Internet. Data can be read and understood without any special measures are called plaintext. Cryptography plays an important role uncertain communication over the network and it provides a best solution to offer the necessary fortification against the data intruders. Cryptography is the science of safeguarding data. Cryptography is way of implanting mathematics to encrypt and decrypt data. Cryptography provides you to store sensitive information or transmit it across the insecure networks so that cannot be read by anyone except the intend recipient.

Classical cryptanalysis involves an interesting combination of analytical reasoning, application of mathematical tools, pattern finding, patience, determination, and luck. Cryptanalysts are also called attackers. Cryptology embraces both cryptography and cryptanalysis [2][3]. During communication, the sender performs the encryption with the help of a shared secret key and the receiver performs the decryption. Cryptographic algorithms are broadly classified as Symmetric key cryptography and Asymmetric key cryptography.

A. *Symmetric Cryptography*

In the symmetric key encryption, same key is used for both encryption and decryption process. Symmetric algorithms more advantageous with low consuming with more computer science power and it works with high speed in encrypt them. The symmetric key encryption takes place in two modes either as the block ciphers or as the stream ciphers. The block cipher mode affords, whole data is divided into number of blocks. This is based on the block length and the key is provided for encryption. In the case of the stream ciphers the data is divided as small as single bits and randomized then the encryption takes place. Symmetric key cryptosystems are much faster than the asymmetric key cryptosystems [3]. In this paper we examine only called classical encryption techniques. There are two building blocks of all encryption techniques.

B. *Asymmetric Key Encryption*

Asymmetric key algorithms use two keys, a public key (PBK) and a private key (PRK) for encryption and decryption process. Each user has their own private key and public key. In this algorithm the sender (X) encrypts the plain text message (M) input into cipher text (C) by using the public key PBK of the receiver (Y), which is known to the parties involved in the communication. At the receiver end the cipher text is renewed back into the original plain text by applying the receiver's private key (PRK) because the receiver's private key and public key are related. So only the authorized person at the receiver end can decrypt the encrypted message. In public key algorithms key exchange is easier but encryption and decryption are complex and time consuming. Also, public key algorithms are vulnerable against chosen plain text attack. The main examples of Public key algorithms are RSA, ECC and Diffie-Hellman key exchange algorithm.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 03, March-2022

Public Key Private Key
Ciphertext Plaintext
Plaintext

II. TYPES OF ENCRYPTION TECHNIQUES

A. Substitution Techniques

This is one of the classical encryption techniques which involve the exchange of a cipher text symbol for a plain text symbol [4].This method substitutes the plain text bit patterns with cipher text bit patterns or plain text letters with other letters, numbers or symbols based on the key values. Examples of this method are Caesar cipher, Playfair cipher and Hill cipher.

B. Transposition Techniques

The process of mapping plain text letters to cipher text letters is accomplished by performing some permutation on the original plain text letters [4]. This method can be made more secure by execution more than one transposition operation. Examples of transposition ciphers are Rail fence technique and columnar transposition.

III. STREAM CIPHERS

Stream ciphers achieve encryption and decryption on stream of plain text and cipher text, usually one bit or byte at a time. Sometimes stream ciphers operate on one 32-bit word [5]. Stream ciphers are more appropriate for real time applications such as multimedia. The examples of stream ciphers are A5 and RC4.

IV. BLOCK CIPHERS

Block ciphers implement encryption and decryption on blocks of plain text and cipher text, usually a block size of 64 bits. Sometimes block size is more than 64 bits [5]. Linear cryptanalysis is one of the widely used attacks on block ciphers. The examples of block ciphers are DES, AES and Blowfish.

A. Mon Alphabetic Cipher

This is one of the substitution techniques. This method maps the plain text alphabet to cipher text alphabet, that is a single cipher alphabet is used per message [22]. Example of this method is DES.

B. Polyalphabetic Cipher

This method is also a substitution technique. This technique uses a set of related mono alphabetic substitution rules based on the key value and the key determines the rule chosen for a given transformation [12]. The examples of this method are Vigenere cipher
and Vernam cipher.

V. RELATIVE WORK

A new approach to encrypt secret information based on the introducing concept of triangularization [6]. Since the encryption and decryption is done on a binary file by means of XOR operation it is effective on any type of data such as text or multimedia files. Modify the plain text to a cipher text using cryptography process [7]. Some Boolean algebraic operations are used. The cipher text is further suppressing the inside a cover media of image. Cryptanalysis and Steganalysis methods of recovering data at receiver side are also exposed. An algorithm is proposed for considering encryption as a critical security measure for protecting data privacy [8]. The entire process is achieved by considering binary data to cover all kinds of data in the field of Computer Science thus ensuring data security irrespective of what information is being transmitted. Focused primarily on the idea wherein the location of data is encrypted along with the data itself by means of a single Location Encryption Algorithm [9]. The security is further boosted using the Confirmation Code. Proposed a process of embedding to accomplish data hiding under the transformation (DWT and (IDWT) of cover image and to obtain privacy by using Huffman encoding [10].

Introduced a technique where Caesar cipher and Rail Fence cipher technique are combined to eliminate their respective fundamental weaknesses, and produce a cipher text that is hard to crack [11]. When Caesar cipher substitution and Rail fence transposition techniques are used individually, cipher text obtained is easy to crack. This talk will present a perspective on combination of techniques substitution and transposition. Combining Caesar cipher with Rail fence technique can eliminate their fundamental weakness and produce a cipher text that is hard to crack.

VI. EXISTING SYSTEM

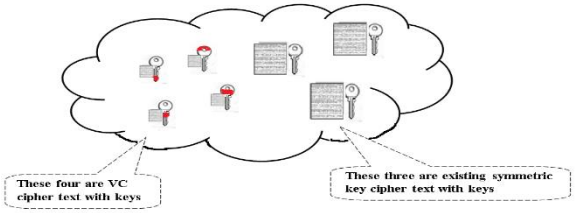
A. Transposition techniques

In the cryptography system, a transposition cipher is a method of encryption by changing the position of plain text into different position. In this technique, the character or group of characters are shifted into different positions. That is the order of units is changed mathematically and gets the cipher text. There are several techniques. They are:
Rail fence technique: It is the technique which rotates the position of plain text into cipher text. For example the message UNIVERSITY is positioned and gets the cipher like the following.

U I E S T
N V R I Y
Plain text: UNIVERSITY

VII. PROPOSED SYSTEM

This is more beneficial for both the sender (**Pal**) and receiver (**Milky**) who exchanges the information easily without confusion, and the third party (**Cracker**) will not hack the information because the key length and cipher text length is not matched. In most cases it produces tiny cipher text. For this task, we perform *six* functional operations. They are:



International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 03, March-2022

A. Sorting Out the Plaintext (SOP).
B. Drop Dupe Chars (DDC).
C. Diagonal Transpositions (DT).
D. Key Generation (KG).
E. Bit Substitution (BS).
F. Oscillation Fetching (OF).

A. Sorting Out the Plaintext (SOP)

Sorting is a process of arranging the plaintext elements into an order. This is one of the key concepts to arrange the elements in an order to find out the repeated (duplicate) letters easily to drop out at next section. To sort the plaintext elements, we use on of the sorting techniques i.e. selection sort. For instance, the original plaintext The sorted text index positions are made as an IndKey (indexed key) which is sent as a partial key to the receiver (**Milky**) from the sender (**Pal**). The algorithm which sort the plaintext is as follows.

Algorithm Sort Text(list)
Input : list, the string of characters.
Output : index, act as index key.
Steps:
1. Set MIN to location 0
2. Repeat step 3 through step 5 until list is sorted
3. Search the minimum element (character) in the list
4. Swap the character and index with location MIN
5. Increment MIN to point to next character
6. Return index
7. Stop

Figure 1.1 Algorithm to sort the plain text

B. Drop Dupe Chars (DDC)

The 'Drop Dupe Chars' method drops the repeated characters from the original plain text. Before dropping the duplicate characters, we find out the number of duplicates (repeated characters). Then count the number of duplicates and add that number to the character in the new string. For example, the original plaintext is "1234567890" the operated DDC text is "x00x01x02x03x04x05x06x07x08x09". The length between these two texts are: the original plaintext is 10, the operated DDC plaintext is 30 because every numeric entry point of view the algorithm is padded 'x0' character to the number. Here we observed the operated DDC text is greater than the original plaintext. The following figure shows the process of increased DDC text. Here the DDC Plain text is acts as an intermediate text between the original plaintext and the cipher

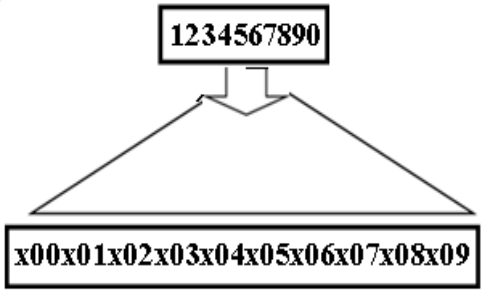


Figure 1.1 DDC Plain text with increased text.

The algorithm is used to implement DDC is as follows.

Algorithm DropDupeChars (InChars)
Input : InChars is an array of characters of plain text.
Output : varyString is a new string which is varying in size of inChars (plaintext).
Step 1 : n ← inChars.length
Step 2 : Repeat step 3 through step 5 for i ← 1 to n
Step 3 : if inChar[i] = digit then
Add x0 and inChar[i] to varySring
Step 4 : else
Add inChar[i] to varyString
Step 5 : if any duplicates then
Count and add number to varyString
Step 6 : return varyString
Step 7 : Stop

Figure 1.2 Algorithm for Drop Dupe Chars

C. Diagonal Transposition (DT)

The Diagonal Transposition is the primary unit to perform all operations of our proposed algorithm. To meet this demand, first we create square matrix as block and divide the matrix into three logical parts: first part is diagonal part, second one is upper triangle and third one is lower triangle diagonals. This logical part is shown in the Figure 1.8 with colours; **Red** represents diagonal, **Green** represents upper diagonal and, **Blue** represents lower diagonal. For 9-character (C) filling diagonal transposition is as follows.

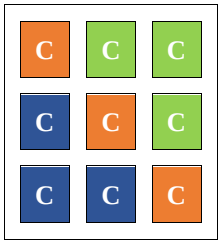


Figure 1.2 Diagonal representation for 9 elements

This process to generate greater cipher than the DDC generated plaintext. After creating the suitable matrix, the DDC character are filled and implemented the diagonal transposition technique in the following manner. For character (C) filling diagonal transposition is as follows

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 03, March-2022

.

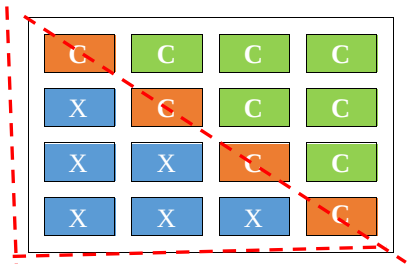


Figure 1.3 Diagonal representation for more than 9 elements

The algorithm diagonal transposition is as follows.

Algorithm Diagonal Transpositions (inString, N)
Input : inString is a string of characters, N is the size of the block.
Output : Block, contains data in diagonal manner.
Steps:

1. Pos = 0
2. Repeat step 3 through step 4 for i=0 to N-1 do
3. Block[i][i] = inString[Pos]
4. Pos = Pos + 1
5. Repeat step 6 through step 8 for i=0 to N-1 do
6. Repeat step 7 through teep 8 for j=i to N-1 do
7. Blck[j-1][j]=inString[Pos]
8. Pos = Pos + 1
9. Repeat step 10 through step 12 for i=0 to N-1 do
10. Repeat step 11 through step 12 for j=i to N-1 do
11. Block[j][j-i]=inString[Pos]
12. Pos = Pos + 1
13. Return Block
14. Stop

Figure 1.3Algorithm of diagonal transposition

D. Key Generation (KG)

The proposed algorithm is the symmetric key algorithm, so that the sender (**Pal**) and receiver (**Milky**) must share the common key in a secure mode. The proposed method follows two basic steps in sequence are: the *primary key* and *indexed key*. The combination of these two keys becomes the common key (Main Key). The primary key is used to perform all primary operations on the specific DDC plaintext like Bit Substitution, Oscillation Fetching, etc. The indexed key is used to get back the original plaintext from UnSort method at receiver (Milky) end. The primary key is generally entered by the sender (Pal) as any kind of text. This text should be changed as numbers like 1, 2, 3, etc. This key is related to square matrix (block) column size. The indexed key is generated by the SOP (Sorting Out the Plaintext) method. This method should have generated the *indexed* position of every character of the plaintext. Finally, both the primary and indexed keys are merged and generated as a large key. This is the main key to distribute both the parties. Finally, this process causes most confusion and diffusion to the attacker.

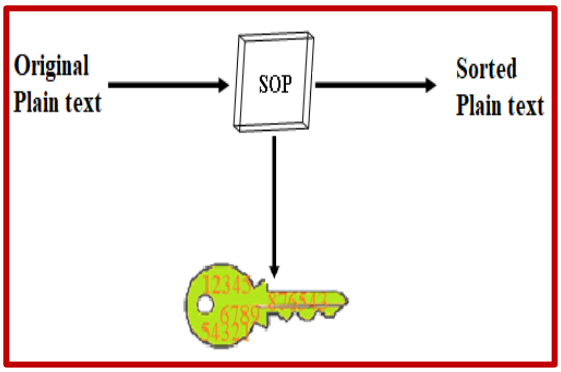


Figure 1.4. SOP generated indexed key (IndKey)

E. Bit Substitution (BS)

We substitute the bits of byte in Bits Substitution. This process should be follows from right to left. For example, the bit 0 is replaced by 1 and 1 is replaced by 0 and vice versa. Through this system, the individual bits of a byte is changed based on the key positions If the key value is greater than 8, the positions are started from 1 onwards because the greater value are divisible by 8 for getting remainder value. \For instance, the plaintext 'U' is used to change the different kinds of cipher symbols that is depending on the key positions. The binary value of the said plaintext "U" is 01010101. Then implementing with different key values, we get different symbols that are shown in the following way.

Binary								Decimal	Symbol
0	1	**0**	1	**0**	1	**0**	1	170	U

If the key value is = 1, the changes are made for the above example is shown in the following.

Key	Binary								Decimal
	1	0	**1**	1	**1**	1	**1**	1	170

Symbol ª

1

Here, every bit is changed i.e. '0' as '1' and '1' as '0'. Because key is 1. So that the cipher decimal is 170 and symbol is 'ª'.

If the key value is = 2, the changes are made for the plaintext U is shown in the following.

Key	Binary								Decimal	Symbol
2	**1**	0	**1**	1	**1**	1	**1**	1	235	ÿ

Here, every 2nd bit only is changed i.e. '0' as '1' and '1' as '0'. Because the key is 2. So that the cipher decimal is 255 and symbol is 'ÿ' etc,,
Hence key value is = 9, there is no 9th bit in the said ASCII Byte format. So that for 9, we generate the value 1 because modulo operation is done for key 9 mod 8. This process

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 03, March-2022

continuous for the new generated key value like 10, 11, 12, and so on.

VIII. OSCILLATION MODEL

A. Oscillation Fetching (OF)

It is motion of an Object that regularly repents itself, back and forth, over the same path. This is another technique which we use to get the characters either for sending or receiving the parties. This is also the confusion technique for the cryptanalysts who does not recognize the cipher text easily.

If the key values are odd, then the fetching process is:

$$A_{ij}$$

If the key values are even, then the fetching process i$A_{ji,}$ The algorithm which is used to support the oscillation model is as follows

B. Procedure Oscillation : Encryption

1. inChars : array of characters
2. Key : array of values
3. size←key.length
4. For i←1 to size
5. Find key (sequentially) and repeats step 7 and 9 randomly
6. For j←0 to size-1
   cipherString←cipherString + inChars[key][j]
7. End for
8. For j←size-1 down to 0
   cipherString←cipherString + inChars[key][j]
9. End for
10. End for
11. Return cipherString
12. End Procedure

1.4. Algorithm for Oscillation Model

The Oscillation algorithm gets the cipher characters from the block in a top-down and bottom-up approach. This process should be done by the relation of key positions. The character which is fetched from inChars array then added to the cipherString and should return to the **Pal** to send it to the **Milky** as a complete cipher text.

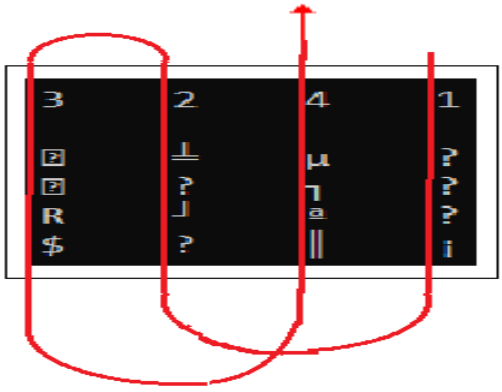
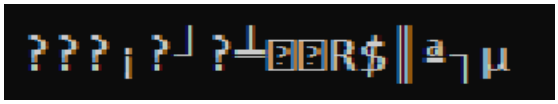


Figure 1.5. Oscillation Fetching from the block

The cipher text which we get to send in exact manner said above is as follows.



Final Cipher text

C. Encryption

Message encryption is one of the processes, which convert the message (plaintext) into scramble (cipher) text. The method BitSubs() is used to convert the plaintext into ciphertext. It takes the message as input form the source node usually sender Pal, then process and produce the required output to the destination node usually receiver Milky as cipher text. This process is shown in the following diagram

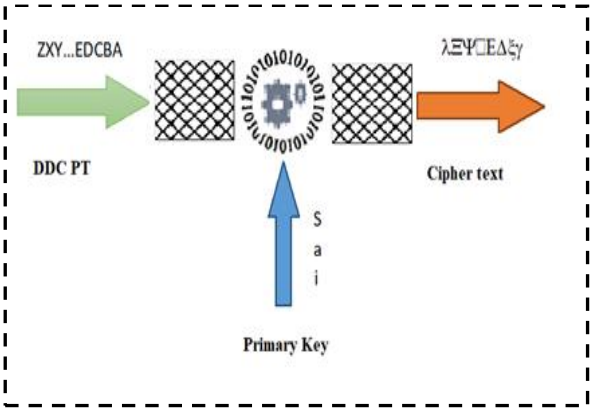


Figure 1.6. Passing of Message Encryption

After sending the message by the sender (through various methods), the method BitSubs () of VC performs the operation on plaintext with the help of the key. While processing the bits, and bytes of the message are replaced with other bits or bytes. Then return the processed text as ciphertext usually send to the related destiny. For example, the message "**In the car, at the end of the war and surely in the roar I am alive.**" is send by the source node with the key value "**cipher**". Then the message is reduced as a DDCPT is like "**16,.I2a7cd2e7fh4i3l2mn4o2r5st5uvwy**" and **SOP** generates the IndKey. Then the sequence of process is shown below.

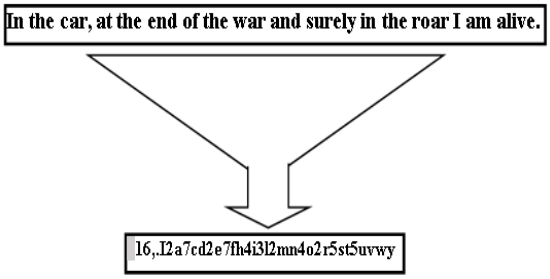


Figure 1.7. DDC reduced plain text

The indexed key (IndKey) which is generated by the SOP is shown in the following table. This key is related to the above plaintext. This key is different for different kind of

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 03, March-2022

plaintexts. Both IndKey and Primary Key are merged as a Main key and send to the **Milky** as a core key.

IndKey	2 6 11 14 18 22 25 29 33 37 44 47 51 56 58 61 10 67 0 57 31 8 34 12 54 59 62 7 21 36 28 5 41 50 19 66 17 24 16 4 49 27 45 64 42 63 60 20 1 46 35 23 53 52 9 55 40 32 38 48 13 15 26 3 39 65 30 43
Primary Key	**cipher** (1 4 5 3 2 6)

Table 1.1. Key Generation

The bit stream which is related to the given DDC plaintext is as follows.

10101000000101010010011011010011100001
00011010010001011001110001110010001100
10010100010001110010100110101001110101
00011010010111100111100100100110011000
01001100000100101110010101001010001001
00100100001001100011111010000100010110
00111000101110111101010100010110011011
11111101011101

Figure 1.8. Plaintext bits generation

The bit stream which is related to the ciphertext is as follows. We know this process performed by applying the above operations (Encryption). This bit stream changes according to the given key

10101000000101010010011011010011100
00010001101001000101100111000111000
10001100100101000100011100101001100
10100111010100011010010111100111100
01001001100110000100110000010010110
10010101001010001001001001000010010
10001111101000010001011000111000100
11101111010101000101100110111111111
01011101

Figure 5.20 Cipher text bits generation

The related ciphertext of the above stream is as follows.

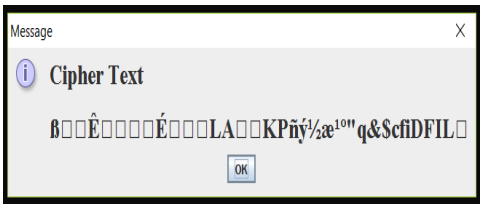


Figure 5.21 Cipher text characters

D. Decryption

Message decryption is the process of converting the ciphertext into plaintext. It is similar to encryption,

simply it is the reverse process of encryption. Here, the ciphertext is placed into the block (matrix) in a diagonal position and performs OscillationDes, bitSubs, getElements, pickDroppedChars, and unSort methods of VC to get plaintext by passing different key to the related algorithms. The algorithms of decryption oscillation is as follows.

E. Procedure Oscillation : Decryption

Procedure OscillationDec
Step1: cipherText : array of characters
Step2: keyPositions : array of key positions
Step3: charPosition ← cipherText.length-1
Step4: size ← cipherText.length
Step5: For i←1 to size
Step6: Find key (reverses) and repeats step 7 and 8 randomly
Step7: For j←0 to size-1

cipherBlock[key][j]←cipherText(charPosition)
charPosition←charPosition-1
End for
Step8: For j←size-1 down to 0

cipherBlock[key][j]←cipherText(charPosition)

charPosition←charPosition-1
End for
End for
Step9: Return cipherBlock
End Procedure

Figure 1.5. Pseudo code of reverse Oscillation

The OscillationDec algorithm, is an exact reverse process of the oscillation in the encryption process.

F. UnSortText algorithm

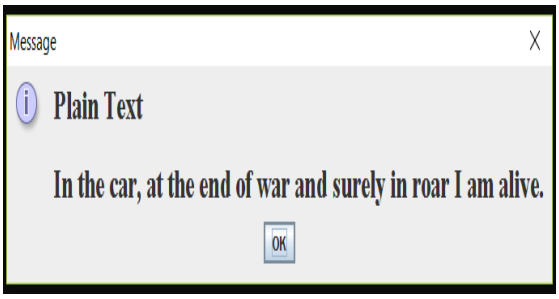
The UnSortText algorithm, unsorts the elements. It simply place the letters of the string into previous positions to produce the actual plaintext at sender's end. Here the IndKey place an important role to position the characters in previous positions. The related java code to implement the above algorithm to unsort the elements and get the actual string which is exactly sent by **Pal** to **Milky**.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 03, March-2022

```
Procedure UnSortText
        IndKey : array of characters
        size← IndKey. Length
        For i←1 to size
PlainText←PlainText+ unSort[IndKey[i]]
        End for
        Return plaintext,  End.
```

Figure 1.9. Pseudo code to unsort the characters

After unsorting pick dropped character, the plain text message is as follows.



IX. CONCLUSION

In this paper we introduce new technique i.e. oscillation fetching provides secure communication between two parties. This system allows characters, prime numbers or composite numbers to prevent the unauthorized access.

AUTHORS INFORMATION

1.U.Thirupalu is joining as a Part-Time Research Scholar in the department of computer science , S.V.University College of CM & CS, Tirupati. He pursuing PhD under the guidance of Dr.E.Kesavulu Reddy in the Dept. of Computer Science, S.V.University college of CM&CS, Tirupati.

2.E.Spandhana is working as a Business Analyst in ADECCO Pvt Ltd, Bangalore, Karnataka.

3.Dr.E. Kesavulu Reddy

Department of Computer Science, S. V. University College of CM & CS, Tirupati, Andhra Pradesh-517502, India. He is working as a Senior Assistant Professor in the Department of Computer Science, Sri Venkateswara University College of Commerce Management and Computer Science, Tirupati, Andhra Pradesh-India. He received master of computer Applications on 2002 from S.V.University, Tirupati, Andhra Pradesh, India. He completed Master of Philosophy in Computer Science on 2006 from Madurai Kamraja University, Madurai, Tamilnadu, and Doctor of Philosophy in Computer Science 2012 from S.V.University, Tirupati, Andhra Pradesh, India. His research interest includes Elliptic Curve CryptographyNetwork Security, Data Mining in the Computer science. He had published 50 papers in various International Journals. He had attended and presented 50 papers in various International and National conferences. He had organized two National Conferences i.e. "National Conference on Information Security & Internet of Things (ISIoT-2K19) 20-21, December 2019, and National Conference on Information Security & Data Security in Cloud Computing (ISDSCC2K21) 29-30 April 2021. A PhD student has been awarded under his Supervision and one Submitted during 2014 to 2020. He received Dr. Surveypalli. Radhakrishna Life –Time Achievement National Award with Gold Medal, Memento and Certificate from IRDP Group of Journals, Chennai on 30th May 2018. He was Honored with "Fellow of Computer Science Research Council (FCSRC)" from Open Association of Research Society from Global Journals, U.S.A) on 31st January 2019 for the performance of published research work in the world. He was awarded with "Best Outstanding Researcher 2020" International Award and Best Outstanding Scientists 2020 with Gold Medal, Memento and Certificate from Kamaraj Institute of Higher Education Thane, Madurai-Tamilnadu. Also, he received "Best Outstanding Scientists 2021" International award from International Scientists on Science, Engineering & Medicine 2021, VDGOODTM TECHNOLOGY FACTORY, Coimbatore, Tamilnadu, India

REFERENCES

[1] Ayushi, "A Symmetric Key Cryptographic Algorithm", International Journal of Computer Applications, ISSN: 0975 – 8887, Vol. 1, No. 15, 2010.

[2] Sanket A. Ubhad, Prof. Nilesh Chaubey, Prof. Shyam P. DubeyASCII Based Cryptography Using Matrix Operation, Palindrome Range, Unique id International Journal of Computer Science and Mobile Computing IJCSMC, Vol. 4, Issue. 8, August 2015.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 03, March-2022

[3] Mohammad Shahnawaz Nasir, Prakash Kuppuswamy "Implementation of Biometric Security using Hybrid Combination of RSA and Simple Symmetric Key Algorithm" International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol. 1, Issue 8, October 2013.

[4] Van Tilborg, HenkC. A.;Jajodia, Sushil, eds. Encyclopedia of Cryptography and Security. Springer.ISBN 978-1-4419-5905-8., p. 455, 2011.

[5] Anupam Mondal, Joy Samadder, Ivy Mondal, Neha Majumder, SudiptaSahana, Asymmetric Key based Secure Data Transfer Technique", International Conference on Computing, Communication and Sensor Network (CCSN) 2012.

[6] SudiptaSahan a, Abhipsa Kundu, "Diagonal Block Steganography Based Enhanced Auxiliary Key Crypting for Secure Data Transfer", International Journal of Advanced Research in Computer Engineering & Technology(IJARCET) Volume 3 Issue 10, October 2014.

[7] MonalisaDey, Dhirendra Prasad Yadav, Sanik Kumar Mahata, Anupam Mondal, SudiptaSahana, "An Improved Approach of Cryptography using Triangulation andMSB Iteration Technique" Special Issue of International Journal of Computer Applications (0975 – 8887) International Conference on Computing, Communication and Sensor Network (CCSN) 2012.

[8] Soupayan Dutta, Soumya KantaDey, SudiptaSahana, "Implementation of Location Encryption Algorithm for Data Flow in Database Systems Ensuring Enhanced Security Management", International Journal of Innovations in Engineering and Technology(IJIET),Volume 6, Issue 4 April 2016, ISSN:2319-1058.

[9] A. Nag, S. Biswas, D. Sarkar, P. P.Sarkar, "A Novel Technique for Image Steganography Based on DWT and Huffman Encoding," International Journal of Computer Science and Security, (IJCSS), Volume (4): Issue (6),pp. 497-610,2011.

[10] Ajit Singh, Aarti Nandal, swati Malik, "Implementation of Caesar Cipher with Rail Fence for Enhancing Data Security", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 12, December 2012, ISSN: 2277 128X.

[11] IJARCSSE-"Study of Cryptography and its Techniques"-Ajit Singh, Madhu pahai, Annumalik, Volume 3, Issue 6, June 2013.

[12] M. J. B.Robshaw, "Block Ciphers", Technical Report, RSA Laboratories, Number TR –601, July 1994.