International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT206662

# Encryption Model Converting Secret Message to a Single Character

**I. Maria Vincy*[1], S. S. Dhenakaran[2]**

*[1]M.Phil Research Scholar, Department of Computer Science, Alagappa University, Karaikudi, Tamilnadu, India
[2]Professor, Department of Computer Science, Alagappa University, Karaikudi, Tamilnadu, India

**Article Info**
Volume 6, Issue 6
Page Number: 316-322
Publication Issue :
November-December-2020

**Article History**
Accepted : 15 Dec 2020
Published : 28 Dec 2020

## ABSTRACT

A lot of information is shared across the internet in day-by-day activities, which leads someone or hackers to access other personnel's secure information. In these cases, encryption is a well-known solution for ensuring internet data confidentiality and privacy. In order to make it impossible for someone to decode the information, encryption hides original information in an unintelligible form of information. This paper proposed a refined cryptographic algorithm to secure confidentiality information. The resultant of the proposed encryption algorithm generates a single character, which may be an alphabet or number, or special character. The key size required may be the size of the input information.

**Keywords :** Encryption, Message Size, Encryption Algorithm, Single Character, Key Generator, Decryption

## I. INTRODUCTION

Encryption is a great idea to share secure information between users running on the Internet. Data security and privacy of information is generally a concern for Internet users. The Encryption Principle allows users to protect the accuracy and privacy of their personal information. This paper suggested a new algorithm that would reduce the hidden message to a single character. The size of the message is decreased by 99 percent, which is why this work helps to save a lot of storage space.

## II. LITERATURE REVIEW

Bangera et al., 2017 [1] have implemented a new encryption algorithm to provide better security and produced output in waveforms so no one can modify the original information .Panda & Chattopadhyay, 2017 [2] have defined a new hybrid RSA algorithm, in which these algorithms produced a key value (public, private) is depended on four prime numbers.

Joshi et al., 2015[3] have studied a new algorithm to prevent Brute force attack and cryptanalytic attacks. The purpose of this algorithm decreases the size of the cipher text and out the complexity of decryption and the key to the cipher text along. This algorithm

will set aside less effort for encryption contrasted with existing algorithms.

Agarwal & Pal, 2017[4] have experimented with a new encryption algorithm for secure message communication. This algorithm utilizes a technique based on symmetric key s for delivering key values.

KumarPandey et al., 2013 [5] have developed enhanced symmetric key algorithms using the public key and remain difficult to break the original information. This new symmetric algorithm utilizes a key size as 512 bits this algorithm effective for the large quantity of information over existing algorithms. The existing algorithm only appropriate for small quantities of information securely transferred.

Sadhu Narayana et al., 2019 [6] have proposed a new KAN algorithm for secure data sharing. This KAN with RSA algorithm uses graph methods for the message. This will give more security to ensure the sharing of information.

Suyash Verma in 2012 [7] has explored a new algorithm for securing information. This algorithm protects against Brute Force attacks using the key length as 128 bits in the process of encryption. The proposed algorithm maybe given good outcomes as compared to existing encryption algorithms and it is also a time consuming method. Kumari, n.d. In 2019[8] have developed a modify RSA cryptosystem using graph plotting in this algorithm the encrypted message will be a draw on graphs and then converted into image and also encrypted key will be selected by the sender before graph plotting, then same key is send to the receiver side to decrypt the original text. These algorithms provide better security for the encrypted message stored in the form of an image in the cloud.

Kumar & Chaudhary, 2016 [9] has examined a changed RSA cryptosystem for Data Encryption and Decryption depended on n Prime number and Bit stuffing. In these algorithms using n number of prime numbers enhance security if we use then large numbers of values are not easily factored and bit stuffing.

## III. PROBLEM DEFINITION

Traditional cryptosystems typically take up a lot of room for cypher text. Many studies have argued algorithms to reduce cypher text space. Their works have been decreased by 60% of the storage space. The issue with this paper further reduced the storage space for cypher text. It is intended to produce a single character for secret details of the cypher text. It attempts to reduce the data storage and reduces the cypher text by 99 percent.

## IV. PROPOSED ENCRYPTION ALGORITHM

### A. Encryption Algorithm

**Step 1:** Message converted into ASCII Binary 8bit values and stored in an array list called encrypt word.
**Step2:** Count the message size, and size may be odd or even.
**Step 3:** If the size is odd, store the last index position of 8 bit values in array list called valuesforAND and remaining values of two consecutive 8 bit values are XOR and stored the result in arraylist called valuesforXOR or message size is even the consecutive values of two 8 bit values are XOR and produced a result stored in valuesforXOR.
**Step 4:** Again count the size of valuesforXOR, the size more than one, check the size is odd or even. If the size is odd, store last index position to valuestoRemain,or even repeat the process of XOR between two consecutive 8 bit values until the length of valuesforXOR becomes one.
**Step 5:** valuesforXOR size becomes1, already stored odd positions 8 bit values in valuesforAND and valuesforXOR are AND operation between them and produced 8 bit values stored in the finaloutput.
**Step 6:**(key generation and transfer)now generating a key for decryption. If the message size is odd, consider the XOR result of two ASCII bit values is a key and the first operand used in XOR ASCII binary

value is a key value, and then . If the message size is odd, consider the XOR result of two ASCII bit values is a key and the first operand used in XOR ASCII binary value is a key value Example{valuesforXOR, first operand used in XOR} and {valuesforAnd, final output} or message size is even, only consider the XOR result of two ASCII bits value is a key and the first operand used in XOR ASCII binary value is a keyvalue. Key size depends on 50% of an input message.

**Step 7:** final output is converted into ASCII equivalent character (cipher text).

### B.Decryption Algorithm

**Step 1**: decrypting the message using a key value pair and cipher text.

**Step 2:(**if message size is odd**)**cipher text consider as a key find the keyvalue then key and value pair are NAND and result stored in array list called decryptword again result take as key find the key value if key has key value then these two values are XOR, otherwise key has no key value that key stored in decryptword,and also key value consider as a key check the key value if it's have key value, the key value stored in other value.

**Step 3:(**if message size is even)ciphertext consider as a key find the keyvalue then key and value pair are XOR and result stored in array list called decryptword, again result take as key find the key value if key has key value then these two values are XOR , otherwise key has no key value that key stored in decryptword and also key value consider as a key check the key value if it's have key value, the key value stored in other value.

**Step 4**: decryptword values converted into ASCII equivalent value and take reverse the decryptword values finally Original message found.

### C. Example

### i.Encryption

**Step 1**:Message "welcome".

| w | e | l | c | o | m | e |
|---|---|---|---|---|---|---|

**Step 2:** Encryptword values as,

| Encryptword | | | | | | |
|---|---|---|---|---|---|---|
| 1110 111 | 1100 101 | 1101 100 | 1100 011 | 1101 111 | 1101 101 | 1100 101 |

**Step 3:**message size is odd, so last index valuesare stored in valuesforAND.

| valuesforAND |
|---|
| 1100101 |

Otherwise message size is even valuesforAND have no use.

**Step 4:** EncryptwordXOR of two consecutive bit values and produce results stored in valuesforXOR .

| valuesforXOR | | |
|---|---|---|
| 10010 | 1111 | 0000010 |

**Step 5 :** Count the valuesforXOR size it becomes onego to next step, otherwise again XOR of valuesforXOR values and produced result stored in valuesforXOR .and also check valuesforXOR size it may be odd or even if it's odd store last index values into valuestoRemain.

| valuesforXOR | valuestoRemain |
|---|---|
| 11101 | 0000010 |

**Step 6 :** valuesforXOR values and valuestoRemain values are XOR produced results and are stored into ResultOfWord.

| ValuesforXOR | valuestoRemain | ResultOfWord |
|---|---|---|
| 11101 | 0000010 | 111111 |

ValuesforAND values AND operation on ResultOfWord values,

| ValuesforAND | ResultOfWord | Final output |
|---|---|---|
| 00101 | 111111 | 0000101 |

Now, Generate a key consider the XOR result of two ASCII bit values is a key and the first operand used in XOR ASCII binary value is a key value, and then valuesforAND is a key and the final output is a key value.

| Key | Key Value |
|---|---|
| 10010 | 1110111 |
| 11101 | 10010 |
| 01111 | 1101100 |
| 00010 | 1101111 |
| 10111 | 100100 |
| 11111 | 11101 |
| 0000101(cipher text) | 1100101 |

**Step 7:** Final output 8 bit binary value converts into equivalent ASCII character and storesthe result in EncryptedArray.EncryptedArray ASCII values from (00000000 to 00011111) are symbolized as question marks. If final output may be from this range your cipher text will be question mark otherwise out of these range equivalent ASCII character return in EncryptedArray.

| EncryptedArray |
|---|
| ? |

"?" is the encrypted text of your message"welcome".

**ii.Decryption**

**Step 1:** Take encrypted text and key values as input

| EncryptedArray |
|---|
| ? |

Encrypted text converted into ASCII equivalent binary 8bit value and considered as a key.

| Key | Value |
|---|---|
| 10010 | 1110111 |
| 11101 | 10010 |
| 01111 | 1101100 |
| 00010 | 1101111 |
| 10111 | 100100 |
| 11111 | 11101 |
| 0000101(cipher text) | 1100101 |

**Step 2:**Cipher text value consider as key find the keyvalue pair are NAND and result take as key if key has keyvalue again the keyvalue pair are XOR otherwise key has no keyvalue the result is stored in decryptword,and also keyvalue consider as a key, if its have a key value the key and keyvalue pair are XOR and result take as key otherwise keyvalue stored in decryptword.

| Key | Keyvalue | XNOR result |
|---|---|---|
| Xnor | 1100101 | 11111 |

The result 1111 is key if it has keyvalue 11101,and also keyvalue 1100101 considered as a key, it does haveno keyvalue so 1100101 is stored in decryptword.

| Decryptword |
|---|
| 1100101 |

**Step 3:** The result hasa key value so the key value pair is XOR and takes the result.

| Key | Key value | XOR result |
|---|---|---|
| 11111 | 11101 | 00010 |

The result 00010 is a key if it has keyvalue 1101111 and keyvalue 11101 consider as a key it also has a keyvalue so 10010 is stored in othervalue.

**Step 4:** The result has a key value so the key value pair is XOR and takes the result.

| Key | Keyvalue | XOR result |
|---|---|---|
| 00010 | 1101111 | 1101101 |

The result 1100101 is key if it does not have any keyvalue so result is stored in decryptword ,and also keyvalue 1101111 consider it as a key it does not have keyvalue so the result is stored in decryptword.

| Decryptword | | |
|---|---|---|
| 1100101 | 1101101 | 1101111 |

**Step 5:** Previous step key and keyvalue pair have no keyvalue so check othervalues have any value. If it hasa value considersit as a key.

| Key | Keyvalue | XOR result |
|---|---|---|
| 11101 | 10010 | 1111 |

The result 1111 is key if it has keyvalue 1101100 and also keyvalue 10010 considered as a key it has keyvalue so 1110111 is stored in othervalue.

**Step 6:** The result hasa key value so the key value pair is XOR and takes the result.

| Key | Key value | XOR result |
|---|---|---|
| 1111 | 1101100 | 1100011 |

the result 1100011 is key if it does not have any keyvalue so result is stored in decryptword ,and also keyvalue 1100011 considered as a key it does not have keyvalue the result is stored in decryptword.

| Decryptword | | | | |
|---|---|---|---|---|
| 1100101 | 1101101 | 1101111 | 1100011 | 1101100 |

**Step 7:** Previous step key and keyvalue pair have no keyvalue so take key from othervalues 10010 have keyvalue as 1110111.

| Key | Key value | XOR result |
|---|---|---|
| 10010 | 1110111 | 1100101 |

The result 1100101 is key if its does not have any keyvalue so result is stored in decryptword ,and also keyvalue 1110111 consider as a key it does not have keyvalue and also othervalue is empty finally the result is stored in decryptword.

| Decryptword | | | | | | |
|---|---|---|---|---|---|---|
| 11001 01 | 1101 101 | 1101 111 | 110 001 1 | 110 110 0 | 1100 101 | 1110 111 |

Reverse of decryptword

| Decryptword | | | | | | |
|---|---|---|---|---|---|---|
| 11101 11 | 1100 101 | 1101 100 | 1100 011 | 110 111 1 | 110 110 1 | 1100 101 |

**Step 8:**Decryptword convert into equivalent ASCII characters,

| Decryptword | | | | | | |
|---|---|---|---|---|---|---|
| w | e | l | c | o | m | e |

## V. EXPERIMENTAL RESULTS

This algorithm is tested with a few types of input information namely a single word, a sentence,and paragraph.

TABLE I

TESTED STRINGS

| Input Text | Cipher Text |
|---|---|
| HIgOoDdAy | ) |
| WElCoMe TO tHe WoRlD. | = |
| You cant go back and alter the start,where you are ChaNGe The EnD | C |

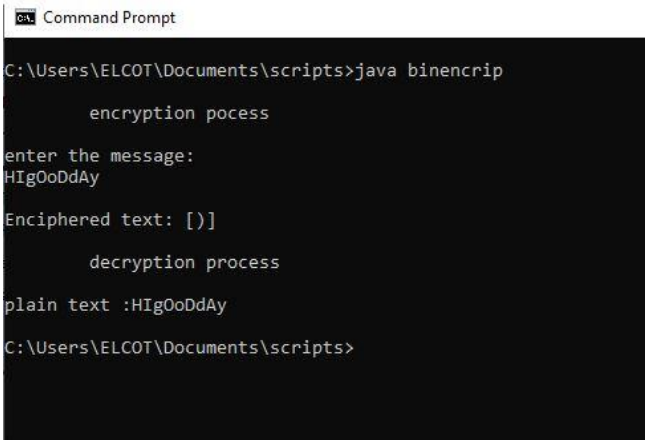


**Fig.1.**Shows snapshot for "HIgOoDdAy" string. Here entered message is welcome. Encrypted text is" )".

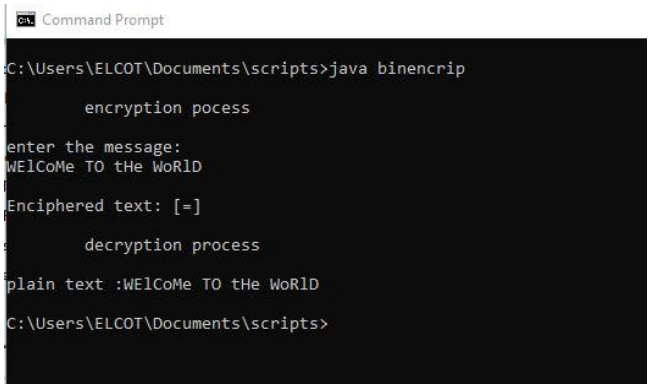


**Fig.2.**Shows snapshot for encrypted message"WElCoMe TO tHe WoRlD" string. Here Encrypted text is "=".

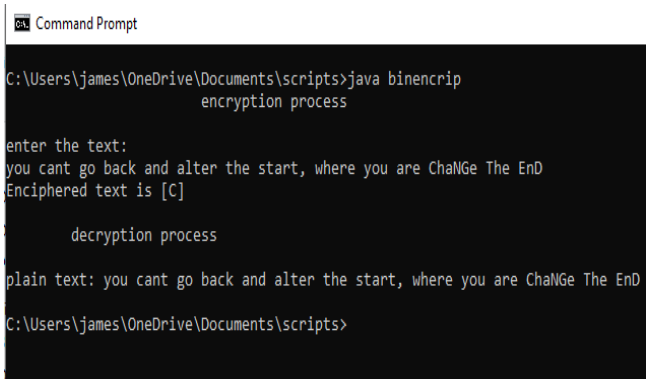


**Fig.3.** Shows snapshot for encrypted message "You cant go back and alter the start,where you are ChaNGe The EnD" string. Here Encrypted text is "C".

## V. CONCLUSION

In this paper, a new encryption algorithm is pr oposed to hide information-making output a single alphanumeric character. The basic XOR and AND operations are played with ASCII values to implement the proposed work.

## VI. ACKNOWLEDGEMENTS

This article has been written with the financial support of RUSA-Phase 2.0 grant sanctioned vide letter No.F.24-51/2014-U, policy(TNMulti-Gen), Dept. of Edu.Govt. of India, dt.09.10.2018.

## VII. REFERENCES

[1]. Bangera, K. N., Reddy, N. V. S., Paddambail, Y., & Shivaprasad, G. (2017). Multilayer security using RSA cryptography and dual audio steganography. 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 492–495. https://doi.org/10.1109/RTEICT.2017.8256645

[2]. Panda, P. K., & Chattopadhyay, S. (2017). A hybrid security algorithm for RSA cryptosystem. 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), 1–6. https://doi.org/10.1109/ICACCS.2017.8014644

[3]. Joshi, A., Wazid, M., & Goudar, R. H. (2015). An Efficient Cryptographic Scheme for Text Message Protection Against Brute Force and Cryptanalytic Attacks. Procedia Computer Science, 48, 360–366. https://doi.org/10.1016/j.procs.2015.04.194

[4]. Agrawal, E., & Pal, P. (2017). A Secure and Fast Approach for Encryption and Decryption of Message Communication. International Journal of Engineering Science and Computing, 7, 5.

[5]. KumarPandey, K., Rangari, V., & Kumar Sinha, S. (2013). An Enhanced Symmetric Key Cryptography Algorithm to Improve Data Security. International Journal of Computer Applications, 74(20), 29–33. https://doi.org/10.5120/13028-0215

[6]. Sadhu Narayana Naidu. (2019). Secure Sharing of Data using an Algorithm Namely KAN. International Journal of Innovative Technology and Exploring Engineering, 8(9), 2959–2966. https://doi.org/10.35940/ijitee.I8523.078919

[7]. Roopali soni, S. V., Rajnish Choubey. (n.d.). An Efficient Developed New Symmetric Key Cryptography Algorithm for Information Security. International Journal of Emerging Technology and Advanced Engineering.

[8]. Kumari, M. (n.d.). DATA ENCRYPTION AND DECRYPTION USING GRAPH PLOTTING. 11.

[9]. Kumar, N., & Chaudhary, P. (2016). Implementation of Modified RSA Cryptosystem for Data Encryption and Decryption based on n Prime number and Bit Stuffing. Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies - ICTCS '16, 1–6. https://doi.org/10.1145/2905055.2905180

**Cite this article as :**

I. Maria Vincy, S. S. Dhenakaran, "Encryption Model Converting Secret Message to a Single Character", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 6 Issue 6, pp. 316-322, November-December 2020. Available at doi
: https://doi.org/10.32628/CSEIT206662
Journal URL : http://ijsrcseit.com/CSEIT206662