Volume 4, Issue 7, July 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Efficient Text Data Encryption System to Optimize Execution Time and Data Security

Charru
M. Tech Scholar
Computer Science & Engineering
GZS PTU Campus, Bathinda, India

Paramjeet Singh
Assistant Professor
Computer Science & Engineering
GZS PTU Campus, Bathinda, india

Shaveta Rani
Assistant Professor
Computer Science & Engineering
GZS PTU Campus, Bathinda, india

Abstract- Network security is becoming more and more crucial as the volume of data being exchanged on the internet increases. A more practical way to protect information is to alter it so that only an authorized receiver can understand it. Security of data and telecommunication can be done by a technique called cryptography. So we can say that cryptography is an effective way of protecting sensitive information by the way that prevents intruder from reading it. In this paper we have developed a new cryptography algorithm which is based on symmetric key encryption method. In symmetric key encryption same key is used for encryption and decryption. This algorithm uses ASCII values of input data and key of variable length to encrypt the data. Further logical operation like exclusive-OR is performed on encrypted data to increase the level of security. The aim of the paper is to develop stronger encryption system that results in minimum execution time and maximum security to encrypt the plain text messages. Experimental results show that proposed algorithm is very secured and efficient.

Keywords - ASCII, Ciphertext, Cryptography, Decryption, Encryption, Plaintext, Security, XOR.

I. INTRODUCTION

In the data communication network, the security of the data has now become a very important issue. From one computer to another computer no confidential message should be sent in raw form as it may be intercept by hacker at any time and they can divert it to some wrong destination. To avoid any disaster, any type of confidential data must be protected from any unwanted intruder. Now day's network security and cryptography is an emerging research area to protect any kind of hacking problems. In this area the programmers are trying to develop some strong encryption algorithm in such a way that no intruder can intercept the encrypted message.

Cryptography is the art and science of study of techniques for secure communication. In cryptography system encryption and decryption techniques are used to convert the plain text or original message to cipher text and cipher text to plaintext respectively. Encryption is considered as the subset of cryptography. Encryption algorithms may be symmetric or asymmetric. These techniques are used to provide security at higher levels.

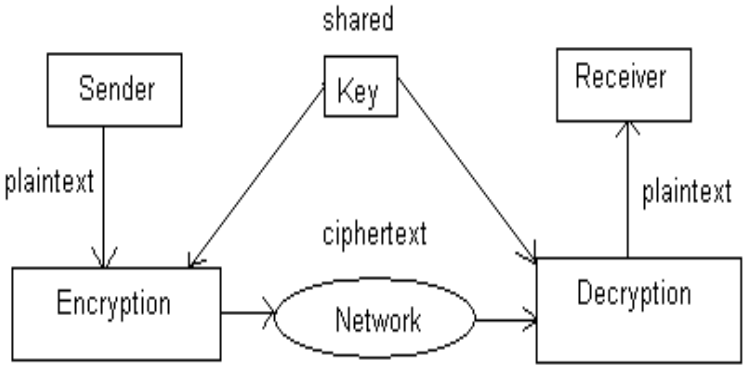

Fig. 1 Symmetric key encryption method

In symmetric key encryption method the secret key is used and it should be make available to sender and receiver both and no one else. Symmetric key encryption algorithms are DES, 3DES, Blowfish, RC2, RC5 and AES et al. In asymmetric key encryption the key used is not secret. The key is public that can be shared by anyone but the decryption key should be make

available to the receiver only. Asymmetric key encryption algorithms are RSA, Digital signature algorithm, Diffie Helman, ElGamal, ECC et al. The main advantage of symmetric key encryption is that management of the key is very simple and easy because only one key is needed. Cryptosystem involves four aspects of security: authenticity, confidentiality, integrity, and non-repudiation services. Symmetric Key encryption ensures confidentiality of the data while asymmetric key encryption can provide all aspect of security except privacy. A cryptography algorithm is good if the technique used can not be easily traced and the hit & trial method is the only method to trace it for all possible and available techniques. Cryptography covered areas related to banking transactions, storing of confidential information, wireless communication, e-mail, web transactions, faxes and phone calls.

II. EXISTING WORK

I Encryption system with self generated key of fixed length

Udepal Singh et. al proposed, a technique which is based on ASCII values of input data. This algorithm is based on symmetric key encryption approach. In this system they used a random method to generate the fixed key of 4 characters. The encryption algorithm used by the author overcome the problem of above discussed algorithm by using fixed length key instead of key length size same as that of length of input data. Further various transformations are applied to encrypt the data with the help of generated random key. To decrypt the data reverse transformations are applied on encrypted data. After the completion of encryption and decryption process encryption and decryption time calculated respectively. In future, this existing system can be further improved by use of variable length key and Unicode values. To accept the system globally, the system must be able to decrypt the sentential form of data [1]. Below table shows the execution time of different data set values.

TABLE I: Execution Time

Input data (Plain text)	**Time in microseconds**
2	849
4	926
6	1299
8	1366
10	1784

II Encryption system with key length equal to input length

A. mathur proposed, an algorithm based on ASCII values of characters in the plaintext. This algorithm used the secret key for the process of encryption and decryption, so named as "Symmetric encryption algorithm". The main focus of the proposed algorithm is that the algorithm operates only when the length of input data and the length of the key used are same. The key used in the system is entered through user manually. Also the proposed algorithm takes more execution time. In future the system can be further improved by using variable key length in comparison with input size [3]. Execution time of encryption/decryption process is described in tabular form as shown below.

TABLE II: Execution Time

Input data (Plain text)	**Time in microseconds**
2	3220
4	3679
6	3861
8	4748
10	5543

III. PROPOSED WORK

In this section we are presenting a new and more secured symmetric key encryption algorithm. In our proposed technique, we are developing a new method to generate the key from the data to be input with the help of some random code. Existing approaches works only with fixed length key that can be easily decrypted with the help of brute force attack. Also the existing system has the moderate level of security because it encrypts the data only one time by using fixed length key. Our proposed system using key of variable length which is depends upon the size of input data that is to be encrypted. Our proposed system generates the variable length key from the data to be encrypted with the help of some random code. Suppose if the size of the generated key is of four bit than it takes two characters from the random method and two characters from the input data to ensure security of the key. To increase the level of security further exclusive-OR operation is performed on encrypted data. To decrypt the encrypted data one has to know exactly what the key character contains.

Steps of an algorithm to perform encryption:

1. Input the original text or plain text and store it.
2. Get the ASCII values for each characters of the input string.
3. Find the minimum ASCII value from the input data.
4. Apply the modulus operation on each ASCII value with the minimum value find in step 3. i.e. (ASCII content % minimum value) and store the resultants in modcontent array.
5. Generate a key of variable length depending upon the length of the plain text.
6. Now, get the ASCII values of the generated key.
7. Find the minimum ASCII value from the step 6.
8. Apply the modulus operation on key ASCII values with the minimum value find in step 7.
9. Perform the right shift the key one time.
10. Now add minimum ASCII value from step 3 to mod key values to obtain the final key.
11. Add each modcontent of data to the final key find in step 10.
12. Generate the encrypted text from the ASCII values obtained from step 11.
13. Apply XOR operation on encrypted text to obtain the final ciphertext.

Steps of an algorithm to perform decryption:

1. Input the ciphertext.
2. Apply Reverse XOR operation on ciphertext.
3. Find out the minimum value from ASCII values of each character of ciphertext and stored in mincipher.
4. Find the ASCII values and minimum value of final key.
5. Calculate the difference of ASCII values of ciphertext and ASCII values of final key.
6. Add the mincipher to the each value of the difference to generate the plaintext ASCII values.
7. Obtain the plaintext with the help of ASCII values.

IV. PERFORMANCE FACTORS

In this section, the following some of the performance factors such as key length management, security of the data against attacks, security levels and the acceptance of text data in sentential form are described [6].

1. **1. Key length management:** In the encryption processing the key management is a considerable and important aspect. The key used by our proposed algorithm is of variable length which is more secured than fixed length key.
2. **2. Security level:** By levels of security, we mean we can go up to two, three or n levels. We measured it by generating the key from the data to be encrypted with the help of some random code to improve security. Further the generated cipher text is again encrypted using exclusive-OR operation to increase the level of security.
3. **3. Acceptance of sentential form:** The existing system that we have studied taken plain text in characters form only but the proposed system accept plain text in sentential form to be accepted globally
4. **4. Security Issues:** Cryptography security means whether encryption approach is secure against brute force and different plaintext-cipher text attack? In our analysis we measure cryptography security in three levels: low, moderate and high. As analyzed our proposed system shows less vulnerability to above attacks.

TABLE III: Performance Factors

Factor Analysis	**Encryption system with key size equal to input size**	**Encryption algorithm with self generated key**	**Proposed Algorithm**
Execution time	More	Moderate	Less

Key length value	Fixed	Fixed	Variable
Security level	Moderate	Moderate	High
Acceptance of sentence form	No	No	Yes
Security Issues	Moderate	Moderate	High

V. RESULTS & DISCUSSION

The below figures and table represent the implementation of proposed algorithm on a number of different text data values and sizes of a wide range. The performance matrices are encryption and decryption time. The encryption time is defined as the time that an encryption algorithm takes to generate a ciphertext from input data or plaintext and decryption time is defined as the time that a decryption algorithm takes to generate plain text form cipher text. In Fig. 2, the author used an input data of variable lengths with key of variable length to generate the ciphertext and corresponding execution time is calculated. Further the generated ciphertext is again encrypted using exclusive-OR operation to increase the level of security only. In Fig. 3, it has been shown that for input text of size greater than ten bits, the corresponding key generated is of six bits and also this result shows that proposed system accepted sentential form of data. In our analysis, we computed that total execution time that covers both encryption and decryption time by our proposed algorithm is less as compared to above discussed existing encryption systems. All the simulation has been conducted using visual C# to implement the proposed algorithm. The simulation results are shown in tabular and graphical form respectively.

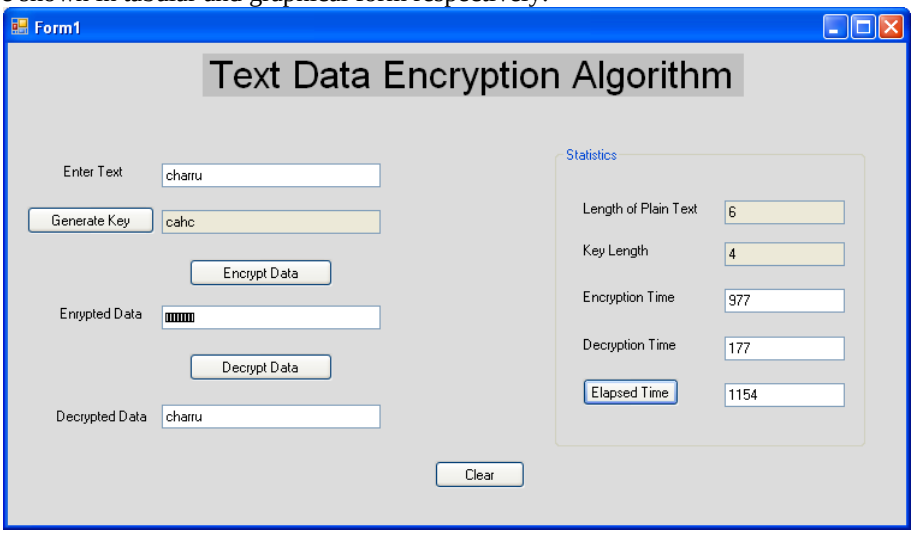

Fig. 2 Output screen for 4 characters input

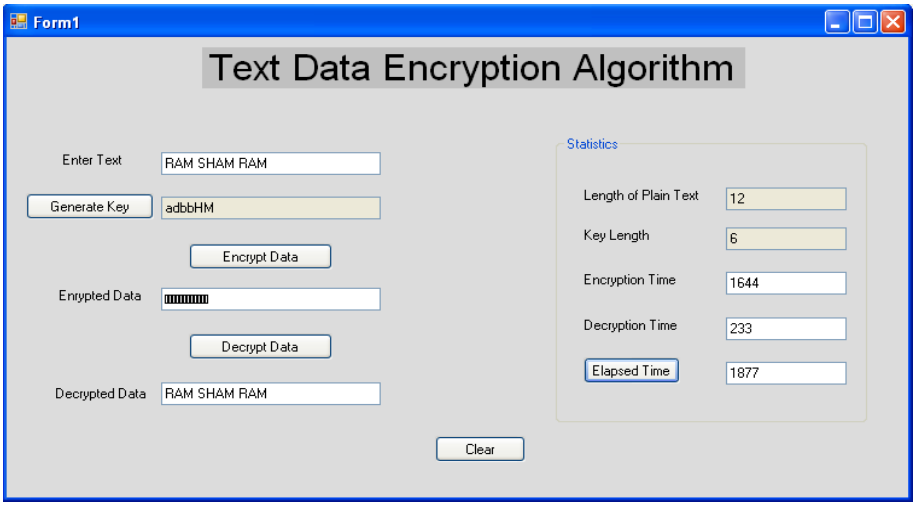

Fig. 3 Output screen for 12 characters input that also shows acceptance of sentential form of data

TABLE IV: Execution Time (in microseconds) Comparison between Existing Algorithms with Proposed Algorithm

Input data (Plain Text) Size	**Encryption system with key length equal to input length**	**Encryption system with self generated key of fixed length**	**Proposed Algorithm**
2	3220	849	683
4	3679	926	702
6	3861	1278	1154
8	4748	1349	1240
10	5543	1752	1649

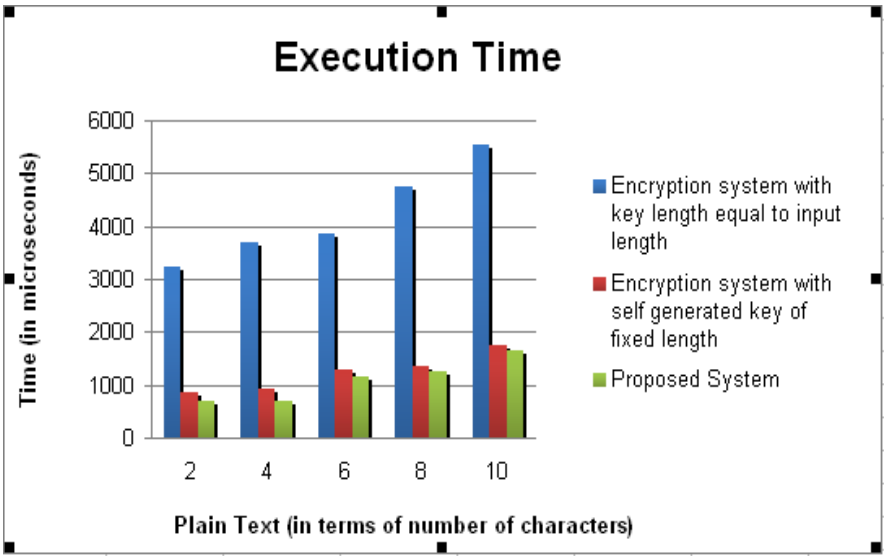

Fig. 4: Graph showing execution time varies with no. of characters

Graphical representation of the above described table is shown in Fig. 4 for execution time of "**Encryption system with key length equal to input length [3]**" and "**Encryption system with self generated key of fixed length [1]**", respectively and is for "**Proposed Algorithm**". According to the graph, there is tendency that total execution time for proposed algorithm, and compared algorithms increases with text data size. The results shows that time needed for the encryption/decryption process by Proposed Algorithm is lesser than encryption/decryption time of compared algorithms.

VI. CONCLUSION & FUTURE SCOPE

From the results, we analyzed that our proposed encryption algorithm producing better results as compared to existing encryption algorithms. Hence the time for encryption and decryption of our proposed algorithm is lesser than existing approaches. As the complexity of the encryption algorithm increases, security also increases but speed decreases. If any user emphasis on security then they can use our proposed method. The main advantage is that it is having variable length key approach to make it difficult for intruder to identify. And the generated key is unpredictable till it is not generated. So the security aspect is high and the execution time is lesser as compared to above discussed existing encryption systems. The system can be further extended to encrypt the multimedia data such as audio files, video files and images etc.

ACKNOWLEDGMENT

I am highly thankful to Dr. Paramjeet Singh for his invaluable guidance and suggestions. I am very much grateful to him for giving constant encouragement and support. I would also like to thank and express Dr. Shaveta Rani for providing the resources to proceed with the research work. Heartfelt gratitude to Computer Science & Engineering Department for their help and support to carried out this work successfully.

REFERENCES

[1] Udepal Singh, Upasna Garg, "An ASCII value based text data encryption system", International Journal of Scientific and Research Publications (IJSRP), Volume 3, Issue 11, November 2013, ISSN 2250-3153.

[2] V. Vasudha Rani, K. Kanaka Vardhini, "Secure and efficient key ciphering through ASCII codes", International Journal of Systems, Algorithms & Applications(IJSAA), Volume 3, Issue ICRAET 13, March 2013, ISSN Online: 2277-2677.

[3] A. Mathur, "An ASCII value based data encryption algorithm and its comparison with other symmetric data encryption algorithms", International Journal on Computer Science and Engineering (IJCSE), Vol. 4, No. 09, Sep 2012.

[4] E.Thambiraja, G.Ramesh, Dr. R.Umarani, "A Survey on Various Most Common Encryption Techniques", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 7, July 2012, ISSN: 2277 128X.

[5] G. Gupta, R. Chawla, "Review on Encryption Ciphers of Cryptography in Network Security", International Journal of Advanced Research in Computer Science and Software Engineering (IJARSSSE), Volume 2, Issue 7, July 2012, ISSN: 2277 128X.

[6] AL. Jeeva, Dr. V. Palanisamy, K. Kanagaram, "Comparative analysis of performance efficiency and security measures of some encryption algorithms", International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622, Vol. 2, Issue 3, May-Jun 2012, pp.3033-3037.

[7] Md. Mizanur Rahman, "Any File Encryption by Translating ASCII Value of Characters", International Journal of Advanced Research in Computer Science(IJARCS), Volume 3, No. 2, March-April 2012, ISSN No. 0976-5697.

[8] V. Gupta, G. Singh, R. Gupta, "Advance cryptography algorithm for improving data security", International Journal of Advanced Research in Computer Science and Software Engineering, volume 2, Issue 1, January 2012, ISSN: 2277 128X.

[9] Pranab Garg, Jaswinder Singh Dilawari, "A Review Paper on Cryptography and Significance of Key Length", International Journal of Computer Science and Communication Engineering, IJCSCE Special issue on "Emerging Trends in Engineering" ICETIE 2012.

[10] CISSP All-in-One Certification Exam Guide, ch-08: Cryptography.

[11] D. Sravan Kumar, CH. Suneetha, A.Chandrasekhar, "A Block Cipher using Rotation and Logical XOR operations", International Journal of Computer Science, Volume 8: Issue 6,No 1, November 2011.

[12] Gurjeevan Singh, Ashwani Kumar Singla, K.S. Sandha, "Throughput Analysis of Various Encryption Algorithms", International Journal of Computer Science and Technology (IJECT), Vol. 2, Issue 3, September 2011, ISSN : 2230-7109 (Online) | ISSN : 2230-9543 (Print).

[13] Gary C. Kessler, "An Overview of Cryptography", May 1998, an article available at www.garykessler.net/library/crypto.html.

[14] Behrouz A. Forouzan, "Data communication and networking", second edition update, copyright © 2001, 1998 by the Tata Mc Graw-Hill companies.

[15] Majdi Al-qdah, Lin Yi Hui, "Simple Encryption/Decryption Application", International Journal of Computer Science and Security, Volume (1): Issue (1).

