IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
Vol. 4, Issue 12, December 2015

ASCII Based Text Encryption and Decryption With Check For Data Integrity

Aniket Olkar[1]

Student, IT Department, PVPIT, Pune, India [1]

Abstract: Data security is of high concern when it comes to data-transfer over the network. With increased attacks on the ciphertexts, the need to improve the ciphers has increased. Not only the data but also the key used for the cryptographic process needs to be secured as all the operations needed to retransform the ciphertext back to original plaintext depend on that key. These needs have led to many new algorithms for encryption as well as for key transfer mechanisms. In this paper, a method is proposed for improving encryption using various mathematical and logical operations and providing a check for data integrity.

Keywords: Cryptography, Symmetric Encryption, Randomness, Secure Key Transfer, Data Integrity.

I. INTRODUCTION

Cryptography means writing or study of codes. It involves various techniques used to scramble the plaintext and then unscramble it to plain original form. Two most prominent encryption methods are Symmetric Encryption and Asymmetric Encryption. Encryption techniques involve use of various mathematical as well as logical functions used to make the text unreadable. Decryption usually involves reverse of encryption operations to obtain the original plaintext.

1.Asymmetric Encryption –
This method makes use of two different keys. One key is used for encrypting the data and another is used for decrypting the data. Any one key from the pair can be used for encryption and the other for decryption.

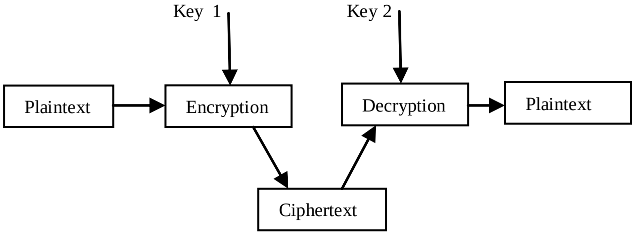

Fig. 1 Asymmetric (Two Key) Encryption

2.Symmetric Encryption –
This method makes use of a single key. Client and server both agree upon a key using techniques like Diffie-Hellman algorithm and then the communication proceeds using that key. Unlike asymmetric encryption, in this method, decryption can be done only with the key which is used for encrypting the data.

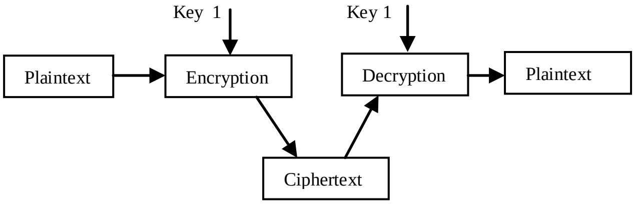

Fig. 2 Symmetric (Single Key) Encryption

3.One Time Pad –
One Time Pad, also known as OTP, is a collection of randomly generated characters which is also the only cipher known to be mathematically unbreakable. Use of OTP in any cipher techniques can help make encryption stronger and unbreakable.

4.XOR Cipher –
Xor cipher uses the same known logical xor operation. Xor cipher is preferred in many complex encryption techniques over other known logical operations due to its property, which is, it has no data leakage. Use of Xor cipher is also seen in many hashing techniques for example - in the SHA-2 family.

5.Modular Arithmetic –
Modular arithmetic is a very well known mathematical technique which is many times used for wrapping around. In this technique, the output value obtained is always less than a predefined value, also known as modulus.

II. PROPOSED SYSTEM

The main focus of the proposed system is to increase the randomness of the generated ciphertext so that it is difficult to break the cipher. Another point on which the propsed system focuses is the way in which symmetric key is transferred. The proposed system secures transfer of such confidential data by providing a way to check the data integrity. In the propsed system, the actual key is not transferred, rather the OTP on which the operations are performed and key is generated is transferred.

OTP is generated based on the stream of input characters and then a set of mathematical and logical operations are performed on it to generate a key for encryption. The same OTP is transferred along with an integer value which is result of xor of OTP, key and other used values along with a predefined constant. At receiving end, again the xor value is generated from OTP, key and other transferred values and xored with transferred xored integer value. If it gives the predefined constant then it can be made sure that data has not been altered.

DOI 10.17148/IJARCCE.2015.412109

III.ALGORITHM

1.Initial Step –

Both the client and server agree upon a constant integer number, lets say xorConstant. Diffie Hellman algorithm can be used for this purpose.

2.Encryption Process –

a)Take message as input.
b)For each word in the message, calculate the ASCII values and add them till single digit number number is formed. (eg:- abc => 97+98+99 => 6)
c)For each such generated single digit, generate an OTP having length corressponding to the generated single digit.
d)Generate ASCII values for each of the OTP.
e)From each OTP's ASCII values, find pair of minimum and maximum value.
f)A key is generated by performing xor operation on each pair of minimum and maximum value. Hence, different key is generated for each word in the message.
g)For each character in the original plaintext message, divide its ASCII value by 26 and get the remainder. Divide the key generated for that word by 26, get the remainder and add the previous value to it. Later 73 is added to the result and this value is stored in different array, say ciphertext array. The generated key is incremented by 1 for each succeeding character in the plaintext word.
h)All the integer values from ciphertext array are converted into characters and the obtained result is the ciphertext.
i)The quotient generated after dividing plaintext character ASCII value by 26 is stored in quotient array and this array is maintained for every word in the message.
j)Now an integer value is calculated which is obtained by performing xor on ASCII values of OTP, generated keys and the values from quotient array. The obtained value is further xored with the predefined xorConstant.
k)Now, the quotient array, OTP, ciphertext and the xored value are transferred over the network.

3.Decryption Process –

a)On receiving the OTP for each plaintext word, its ASCII values are generated and a key for each plaintext word is recreated like in encryption process by xor operation on minimum and maximum ASCII values.
b)Now again, an integer value is calculated which is obtained by performing xor on ASCII values of OTP, generated keys and the values from quotient array.
c)This obtained value is xored with received xor value from encryption process. If the result of xor is the value of predefined xorConstant then its means that the transmitted data has not been altered and the further decryption process can be continued or else retransmission of data with new OTP is required.
d)Now, if the data is unaltered, the received ciphertext is converted into its corressponding ASCII values and then stored in an array.
e)Now, reverse operation of step g. from encryption process is performed on the received ciphertext.
f)The obtained result from above step is converted into characters and the plaintext message is obtained.

4.Data Integrity Check –

In the proposed model, the data which is transmitted over the network is the OTP, ciphertext, quotient array and the xor value. What is needed at the other end of communication end is to check whether the received data has its integrity maintained or not, whether any modification to the data is made or not. To facilitate this check, xor value is used. At the other end, received xor value is xored with the recalculated xor value from the OTP, quotient array and key values. The obtained result should be the value of predefined xorConstant value on which both the communicating parties agreed upon before the communication began. If the value matches, we can say that the transferred data is not tampered with or else ask the sender for retransmission with change of OTP.

This method helps maintain and validate the data integrity. Also, if the validation fails, further steps are not executed resulting in reduced computational costs.

IV.RESULTS

Consider a plaintext message :- I like cryptography

Few of the many ciphertexts generated using proposed algorithm are as follows –

1. ^ USVk oemejfykuQJ\
2. s `^av mckchdwisibZ
3. n ZX[p aW_W\Xk]g]Vh
4. o SQTi rhphmibT^TM
5. k [Y\q h^f^c_rdnd]o
6. j ^_t tjrjUQdV`Voa
7. l YWZo lbjbgcvhrhas
8. _ MKNc vlZRWSfXbXQc
9. p b`cx cYaY^Zm_i_Xj

Above mentioned ciphertexts are a subset of a huge collection of ciphertexts that can be generated using the proposed system.

The proposed system provides randomized results which makes pattern detection in the ciphertexts impossible.

V. MERITS

The proposed system has following advantages –

1.It provides huge number of ciphertexts for the same plaintext message.
2.Generated OTP is not always equal to the length of the plaintext word and thus the computation required for generating it is reduced.
3.It provides a check for data integrity to detect modification of transmitted data over the network.
4.This system can be used not only for single-word plaintext but also for messages with n number of words.

VI. FUTURE SCOPE

The existing system can be further improved to reduce the space complexity as this system needs additional arrays for storage and computational purposes. The system can be extended to support other types of data along with the textual data.

VII. CONCLUSION

Thus, using the proposed algorithm not only can the randomness in the generated ciphertexts can be increased but also the check for any possible data modification can be made available.

The proposed algorithm makes pattern detection in the ciphertext impossible thus avoiding attacks like known-plaintext attacks. Thus, the proposed system will help in making the encryption process strong and in effectively securing the required data.

ACKNOWLEDGEMENT

I would like to thank **Prof. S.C.Chaudhari**, IT Dept, PVPIT, and express gratitude for supporting me.

REFERENCES

[1] Charru, Paramjeet Singh and Shaveta Rani, “Efficient Text Data Encryption System to Optimize Execution Time and Data Security”, IJARCSSE, Volume 4, Issue 7, July 2014.

[2] Udepal Singh and Upasna Garg, “An ASCII value based text data encryption System”, IJSRP, Volume 3, Issue 11, November 2013.

[3] Satyajeet R. Shinge and Rahul Patil, “An Encryption Algorithm Based on ASCII Value of Data”, IJCSIT, Vol. 5 (6) , 2014.

[4] R. Venkateswaran and Dr. V. Sundaram, “Information Security:Text Encryption and Decryption with Poly Substitution Method and Combining the Features of Cryptography”, IJCA, Volume 3 – No.7, June 2010.

[5] Vineet Sukhraliya, Sumit Chaudhary and Sangeeta Solanki, “Encryption and Decryption Algorithm using ASCII values with substitution array Approach”, IJARCCE, Vol. 2, Issue 8, August 2013.

[6] Yashpalsingh Rajput, Dnyaneshwar Naik and Charudatt Mane, “An Improved Cryptographic Technique to Encrypt Text using Double Encryption”, IJCA, Volume 86 – No 6, January 2014.

[7] Nehal Kandele and Shrikant Tiwari, “New Cryptography Method Using Dynamic Base Transformation:DBTC Symmetric Key Algorithm”, IJCTEE, Volume 2, Issue 4, July 2012.

DOI 10.17148/IJARCCE.2015.412109