Asymmetric Key based Secure Data Transfer Technique

Anupam Mondal[#], Joy Samadder[*], Ivy Mondal[$], Neha Majumder[@], Sudipta Sahana[#]
[#] Asst. Prof. Department of CSE, JIS College of Engineering
Kalyani, Nadia, 241235, W.B. India
[*] Department of IT, JIS College of Engineering
Kalyani, Nadia, 241235, W.B. India
[$]Department of CSE, JIS College of Engineering
Kalyani, Nadia, 241235, W.B. India
[@] Department of IT, JIS College of Engineering
Kalyani, Nadia, 241235, W.B. India

ABSTRACT

Data security is one of the important issues in network communication. Secure data transfer become more essential and important, as security is a major concern in the field of message transformation over internet. Data, that is likely to be kept hidden from all people except the authorized users, could not be sent as plain text. Each data has its own features; therefore different techniques are used to protect data from unauthorized access. In recent years, Cryptography and Steganography are two important areas of research that involve a number of applications. These two areas of research are important especially when reliable and secure information exchange is required. Cryptography is the technology that involves converting a message text into an unreadable cipher. Steganography is an art of embedding information in a cover image without causing statistically significant variations to the cover image, so a carrier is needed to transfer information. In this paper, we have tried to introduce a new approach to encrypt secret information which is based on the concept of triangularization. Since the encryption and decryption is done on a binary file by means of successful implementation of XOR operation and this standard should be effective on any type of data such as text or multimedia files.

Keywords- Security, Cryptography, Steganography, encryption, decryption, triangularization**.**

1. Introduction

The rapid growths of computer networks have allowed large files such as texts and digital images to be transferred over the internet and intranet. Data encryption is widely used to ensure security & privacy of the data. For the use of encryption on a data, we use private key as well as public key. But, the uniqueness of our algorithm is that there is no need to use any key, by which we can reduce the complexity of the algorithm to a great extent. We take a binary file as the plain text, which is to be sent to another user over the internet & intranet. We will do bit wise XOR operation on each bit of the file & generate data text. And also we are generating key text based on XOR operation on plain text. After that we are applying the bit wise XOR operation with data text & key text & generate a cipher text. And from the cipher text bit to convert cipher string and forward this cipher string to another user over the internet & intranet. When the receiver will get this cipher string, he or she will follow the almost same procedure with a little variation as that of the encryption process, i.e, he/she will perform bit wise XOR operation on the cipher string & converts into two part decrypted data text & also generate a plain string. After the operation is complete, he will take the left most bits of each operation and generate a binary data string. He/she will again perform bit wise XOR on the decrypted data string. When done, he will take the left most bits again from the results and construct a binary string. This binary string will be the same as the plain text, the algorithm and working of the algorithm is given in the next segments.

The paper is organized as follow. Section 2 describes the different types of encryption Techniques. In Section 3 we have introduced the Asymmetric Key Based Secure Data Transfer Technique followed by an example in Section 4. Finally, in Section 5 we have concluded our paper.

2. Related Work

To give more prospective about the performance of the compared algorithms, this section discusses the results obtained from other resources.

S.Z.S. Idrus et al. have proposed [1] a study of security measure level has been proposed for a web programming language to analyze four Web browsers. This study consider of measuring the performances of encryption process at the programming language's script with the Web browsers. This is followed by conducting tests simulation in order to obtain the best encryption algorithm versus Web browser.

S.Hirani concluded in [2] that AES is faster and more efficient than other encryption algorithms. Increasing the key size by 64 bits of AES leads to increase in energy consumption about 8% without any data transfer. The difference is not noticeable. Reducing the number of rounds leads to power savings but it makes the protocol insecure for AES and should be avoided. Diaa Salama Abdul. Elminaam et al. [3] have compared the various encryption algorithms with different settings for each algorithm such as different sizes of data blocks, different data types, battery power consumption, different key size and finally encryption/decryption speed. In [4], the authors show that these algorithms satisfies the avalanche effect as in other traditional en-cryption algorithms. These algorithms are also shown to be vulnerable to different types of attacks as shown in [5]. Now it becomes clear that a new algorithm which takes into consideration the error nature of wireless channels is critically needed. M. Haleem et al. [6] introduced what is called opportunistic encryption, in which they encrypt the data with longer keys which implies more security whenever the SNR of the channel is higher so that the probability of error at the received cipher text will be lower than for lower SNR values and hence higher security can be used. They also used forward error correction (FEC) codes to protect encrypted packets from bit errors. They also assumed perfect knowledge about the channel in order to use opportunistic encryption. Using there new encryption technique, the authors showed that the throughput of the system was more utilized than for that of using fixed encryption of AES. Data Encryption Standard, was the first encryption standard to be recommended by NIST (National Institute of Standards and Technology).DES is (64 bits key size with 64 bits block size). Since that time, many attacks and methods recorded the weaknesses of DES, which made it an insecure block cipher [7],[8]. *3DES (Triple DES)* is an enhancement of DES; it is 64 bit block size with 192 bits key size. In this standard the encryption method is similar to the one in the original DES but applied 3 times to increase the encryption level and the average safe time. It is a known fact that 3DES is slower than other block cipher methods [7]. *Blowfish* is block cipher 64-bit block - can be used as a replacement for the DES algorithm. It takes a variable length key, ranging from 32 bits to 448 bits; default 128 bits. Blowfish is unpatented, license-free, and is available free for all uses. Blowfish has variants of 14 rounds or less. Blowfish is successor to Twofish [9]. *AES* is a block cipher .It has variable key length of 128, 192, or 256 bits; default 256. it encrypts data blocks of 128 bits in 10, 12 and 14 round depending on the key size. AES encryption is fast and flexible [7], [10].

In this paper we have taken a binary file as a plain text & we first encrypt it & then decrypt it by which we can ensure the security of the data string.

3. Proposed Algorithm

In this section along with key generation algorithm we have proposed our encryption & decryption algorithm

A. Encryption Algorithm

Step 1: Taken an Input string (IS)

Step 2: From this input string we are taking character one by one and represent as an ASCII value

Step 3: Implement triangularization method on this 8 bits binary data (IBD) and taken right most 8 bits string, known as data text (DT)

Step 4: Then generate a key text (KT) from the initial 8 bits binary data (BD) by using key generating algorithm

Step 5: Applying bitwise XOR operation on the data text (DT) and key text (KT) generate cipher text (CT)

Step 6: We then convert the cipher text into its equivalent ASCII value and then we have sent its corresponding characters towards the receiver end

B. Decryption Algorithm

Step 1: Convert the sent data into its equivalent ASCII Value and then change them into their 8 bit binary values

Step 2: Implement triangularization method on the cipher text (CT) and taken right most 8 bits string, known as decryption data text (DDT).

Step 3: Just doing the reverse of decryption data text (DDT) and generate plain text (PT)

Step 4: Convert this 8 bits plain text (PT) to ASCII value.

Step 5: From this ASCII value to Character

Step 6: Now those characters are concatenating and generate a string, this string is actually our input string

C. Key generating Algorithm

Step 1: Taken an 8 bits input binary data (IBD)

Step 2: Generate the every bit of the key by using the following steps:

Step I: 1st and 8th bits value is XOR of 1st to 7th bit position value of IBD

Step II: 2nd and 7th bits value is XOR of 2nd, 4th, 6th and 7th bit position value of IBD

Step III: 3rd and 6th bits value is XOR of 3rd, 4th, 6th and 7th bit position value of IBD

Step IV: 4th and 5th bits value is XOR of 4th, 5th, 6th and 7th bit position value of IBD

Step 3: Now arrange those bits in ascending order and develop the key text (KT)

4. Example

Now, with the use of a character string we will show how this algorithm actually works.

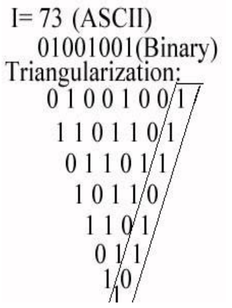
I. Encryption Process

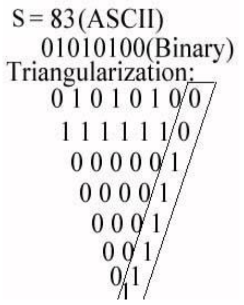
Step1: We take a Character String 'IS'.

Step2: Then we have converted the above string into its ASCII Value which is 73 and 83 respectively

Step3: Then again converting them to their corresponding 8 bit binary code to 01001001and 01010100

Step4: Implementing XOR operation on the binary codes sequentially we get,





The Data Text (DT) generated is 11101101 and 00111111

Step5: Using the key generating algorithm the key text(KT) generated is 01011010 and 11000011

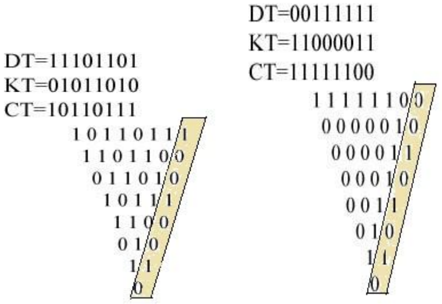
Step6: Doing XOR operation on Data Text(DT) and Key Text (KT) we have got 10110111 and 11111100 which is the required Cipher Text(CT)

Step7: Converting the cipher text to its equivalent ASCII Value we have got 'À' and '³' and the receiver will receive 'À³' instead of 'IS'

II. Decryption Process

Step1: Convert 'À³' into its corresponding ASCII Value to 183 and 252 and their corresponding binary conversion is 10110111 and 11111100 respectively.

Step2: Doing XOR operation on the above values we get the decryption data text(DDT)



The decryption data text generated is 10010010 and 00101010

Step3: Taking the reverse of DDT we get 01001001 and 01010100 which is our original plain text in binary format.

Step4: By converting the following plain text into its ASCII Value we get 73 and 83

Step5: Changing this ASCII Code to Character we get 'I' and 'S'

Step6: Now concatenating these characters we get 'IS' which is our original character string.

5. Conclusion

This encryption operation is done on binary data. So this encryption standard holds for any types of data such as image file, character file or sound file. Here we have converted the given data into its corresponding ASCII Code and then into binary. After simultaneous encryption we have got the desired cipher text which we can send to the receiver. Another advantage that we have kept in mind while designing the algorithm is that there is no limit on the size of the input binary string. This means that large data can also be encrypted easily. Also after decryption the binary value is converted to its ASCII Code and thus we get the desired secured data. But the most significant positivity of this new algorithm is that, there is no need of a key to encrypt and decrypt the data. This fact reduces the complexity of the algorithm, when using large data, to a great extent.

6. References

[1] S.Z.S. Idrus,S.A.Aljunid,S.M.Asi, "Performance Analysis of Encryption Algorithms Text Length Size on Web Browsers," IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.1, January 2008 ,PP 20-25.

[2] S.Hirani, "Energy Consumption of Encryption Schemes in Wireless Devices Thesis," university of Pittsburgh, April 9,2003. Retrieved October 1, 2008, at: portal.acm.org/ citation.cfm?id=383768

[3] Diaa Salama Abdul. Elminaam, Hatem Mohamed Abdul Kader and Mohie Mohamed Hadhoud, "Performance Evaluation of Symmetric Encryption Algorithms", International Journal of Computer Science and Network Security, VOL.8 No.12, December 2008.

[4] Sedat Akleylek, "On the avalanche effect of MISTY1, KASUMI and KASUMI-R," Master's thesis, Middle East Technical University, Feb 2008

[5] O. Dunkelman, N. Keller and A. Shamir, "A Practical-Time Attack on the A5/3 Cryptosystem Used in Third Generation GSM Telephony," Cryptology ePrint Archive, Report 2010/013, Feb 2010

[6] M. Haleem, C. Chetan, R. Chandramouli and P. Subbalakshmi, "Opportunistic Encryption: A rade-Off between Security and Throughput nWireless Networks," IEEE Transaction on Dependaple and Secure Computing, vol. 4, no. 4, pp.313-324, Oct 2007

[7] W.Stallings, "Cryptography and Network Security 4th Ed," Prentice Hall , 2005,PP. 58

[8] Coppersmith, D. "The Data Encryption Standard (DES) and Its Strength Against Attacks."I BM Journal of Research and Development, May 1994,pp. 243

[9] **B**ruce Schneier. The Blowfish Encryption Algorithm Retrieved October 25, 2008, www.schneier.com blowfish.html

[10] Daemen, J., and Rijmen, V. "Rijndael: The Advanced Encryption Standard."D r. Dobb's Journal, March 2001,PP. 137-139