DESIGNING A SECURE ENCRYPTION TECHNIQUE FOR WEB BASED APPLICATIONS

Lalitha Ruby[1], Rahul Johari[2]

[1]ECE, Indira Gandhi Technical University for Women formerly (IGIT)/GGSIP University, (India)

[2]CSE, University School of Information and Communication Technology (USICT)/GGSIP University, (India)

ABSTRACT

In today's scenario of cyber-attacks like phishing, man in the middle attacks and system compromises, it is difficult to ensure the secure data transfer. In order to ensure confidentiality there is a requirement to encrypt the data. So, an encryption and decryption technique module has been developed to convert the plain text to cipher text of binary data stream after two iterations, by converting text into ASCII format and adding random pattern to it in the first iteration and converting the pattern into binary format in the second iteration and saved as the encrypted file in sender's desk and this encrypted file is transferred through web to the receiver. By running the same module at the receiver end the encrypted file can be transformed back into original text and saved as decrypted file in the receiver desk. The designed logic has been tested successfully with the file containing plain text in the form of only alphabets, numbers and alphanumeric characters . Since the logic implemented is custom made, there is remote possibility of breaking the encryption by an intruder, since the logic will be a secret known only within the organisation.

I INTRODUCTION

In today's information world every organization is in need to protect its data, which is considered as its valuable asset. Organizations are spread across states and sometimes across countries. In such cases they make use of Internet as a backbone to carry out their operations including sensitive data transfer. The onus of protection lies with the organization/data owners especially if data of the customers are involved. There is a need to protect customer data as mandated by various security controls. Online fraud costs more than $100 billion of global economy in 2013. So, it is important to ensure secure data transfer. Basically Data Security means protecting its confidentiality, integrity, and availability. The consequences of a failure to protect all three of these aspects include business losses, customer loss, legal liability, and loss of company's goodwill.

An organization have to pay a heavy price in case of compromise of data, especially customer data as per the latest IT Act and other related laws of all the countries. Here if we define the scope only to a telecom to focus on the consequences of security attacks in case of Telecom Company like:

(a) Stealing the subscriber database to capture the market.

(b) Stealing the address, location details of the subscriber.

(c) CDR(Call Details Records) Pattern stealing by the competitors to plan for better marketing strategy

(d) Stealing of reconciliation of CDR's and producing a false claim.

(e) Stealing of postpaid billing data to trap the credit card details of customer

(f) Stealing the location, call details of big shots to plan an evil attack

1.1 Related Work

In [1] authors bring out the nuances of encryption and decryption and discusses implementation of ASCII conversion as part of encryption process. In this paper a secret key is generated subsequent to ASCII conversion, by implementing logic of finding the mod of the ASCII value of the sequence of character, and subsequently generation of key which is converted to binary and back to ASCII. This logic although is logically strong encryption, however is not feasible to implement for conversion of larger files which are required to be taken as input and stored in encrypted format. The format of input files also are restricted in this type of conversion. However the encryption technique discussed is strong and stable.

In [2] authors have considered multimedia data stream as plain text to be transformed into cipher text and have proposed a new block cipher based on randomized key of size n × n where n is the block size and the block undergoes n2 iterations with the plaintext. Every iteration generates the pseudo cipher text. The encryption process generates the ciphertext C with the help of the randomized key. The decryption apply the key in reverse order on the cipher text, to get back the plain text. This work deals with the problem of efficient multimedia data encryption.

In [3] A block cipher technique for security of data and computer networks is proposed. The technique can be used for text, binary and hexadecimal information. It can be placed in any one of the network layers. It is based on changing the system parameters, starting with the block length, including the number of processing rounds, the used permutation, substitution and arrangement boxes, and ending with a disturbance XOR sequence which is XORed with the final cipher-text block. This makes the system looks like a one-time pad system. These keys are indirectly generated from a text key string either inputted from the keyboard or read from a file. This happens in a delicate way using two input key numbers L1 and L2 which indicate the orders of the generated keys. The generated keys are used to make all the used parameters changeable from one block to another and from one 8-bit combination to the next. This is done using ElGamal discrete logarithm pseudo-random sequence generators in a special way. Compared with existing techniques, the proposed method offers good properties

II PROPOSED WORK

Methodology Adopted : The Entire process of Encryption and decryption has been accomplished in Java platform. First the already existing Caesar Cipher is tried to covert plain text to cipher text and a self encryption and decryption technique is developed to convert the same plain text to cipher text and the the transformed decrypted file from both the programs has been compared . The plain text used for transformation to cipher text had alphanumeric text.

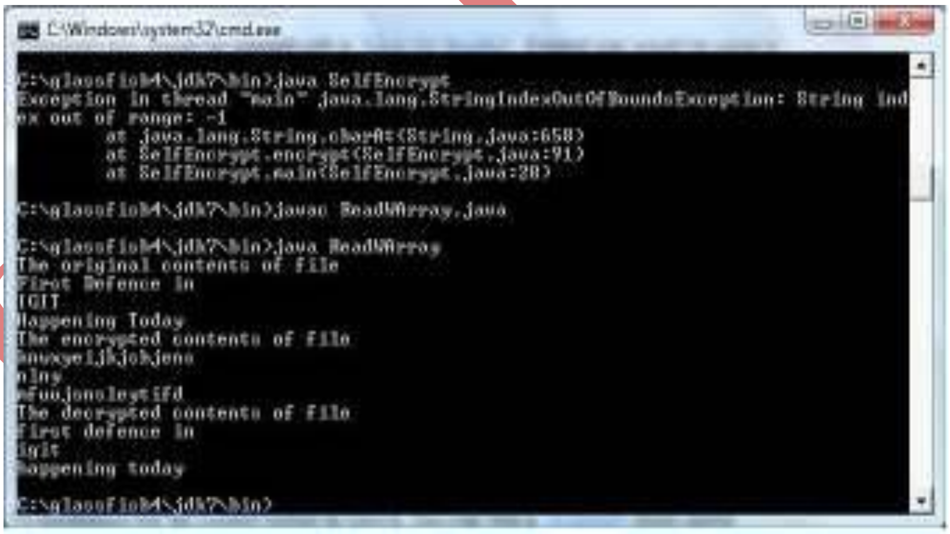
2.1 Step I

In the step 1 it was planned to Script a program to implement Caesar Cipher encryption technique which is a symmetric Algorithm Caesar Cipher, is one of the simplest and most widely known encryption techniques. It is a type of substitution cipher in which each letter in the plaintext is replaced by a letter some fixed number of positions down the alphabet. For example, with a left shift of 3, D would be replaced by A, E would become B, and so on. The method is named after Julius Caesar, who used it in his private correspondence.

2.1.1 The process involved for step 1

1. A program is developed to read the contents of the file.
2. Stored the contents of the file in an array.
3. Read the contents from the array.
4. Printed the same.
5. Applied the encryption technique: Caesar Cipher
6. Printed the original and encrypted contents of the file.
7. After applying the decryption technique the original script was reproduced but failed for the special characters.

   Given below is the snapshot of the running program



```
C:\glassfish4\jdk7\bin>java SelfEncrypt
Exception in thread "main" java.lang.StringIndexOutOfBoundsException: String ind
ex out of range: -1
        at java.lang.String.charAt(String.java:658)
        at SelfEncrypt.encrypt(SelfEncrypt.java:91)
        at SelfEncrypt.main(SelfEncrypt.java:28)

C:\glassfish4\jdk7\bin>javac ReadWArray.java

C:\glassfish4\jdk7\bin>java ReadWArray
The original contents of file
First Defence in
IGIT
Happening Today
The encrypted contents of file
bnuxyeljkjohjeno
nlny
mfuojonoleytifd
The decrypted contents of file
first defence in
igit
happening today

C:\glassfish4\jdk7\bin>
```

2.2 Step II

In order to develop the self encryption and decryption technique, it was proposed to encrypt in the following manner:

1. The original contents of the file are stored in a String array.
2. The contents are read character by character.

3. Each and every character is replaced by its ASCII code.
4. A random pattern is added to the ASCII code.

2.2.1 Similarly the Decryption is done as per the following steps

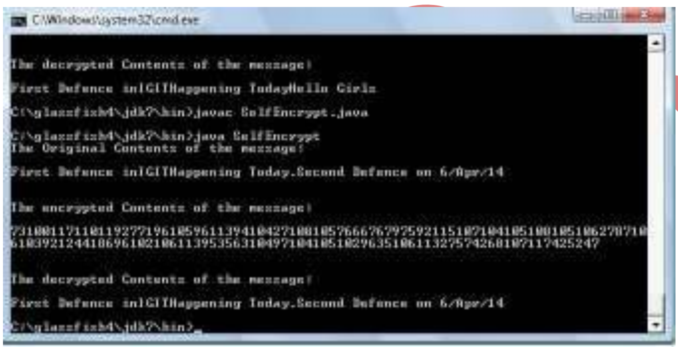
1. The random pattern is then subtracted from the integer number to get the ASCII code.
2. The ASCII code is replaced by the respective character and stored in a String array
3. Now the decrypted original contents are read and displayed.

We have the respective functions in Java that helps us to achieve the above proposed algorithm. Since this technique is not available publicly as Caesar Cipher. Definitely this is more secure than the Caesar Cipher technique.

Plain Text: Second Defence on 6/Apr/14 Pattern used:+3,-5,+3,-5

CipherText:869610210611395356310497104105102963510611327574268107117425247

Given below is the snapshot of the running program:



```
The decrypted Contents of the message:
First Defence inIGITHappening TodayHello Girls
C:\glassfishv4\jdk7\bin>javac SelfEncrypt.java

C:\glassfishv4\jdk7\bin>java SelfEncrypt
The Original Contents of the message:
First Defence inIGITHappening Today.Second Defence on 6/Apr/14

The encrypted Contents of the message:
73100117110119277196105961139410427100105766676797592115107104105100105106278710
61039212441069610210611395356310497104105102963510611327574268107117425247

The decrypted Contents of the message:
First Defence inIGITHappening Today.Second Defence on 6/Apr/14
C:\glassfishv4\jdk7\bin>_
```

Decryption Logic:-

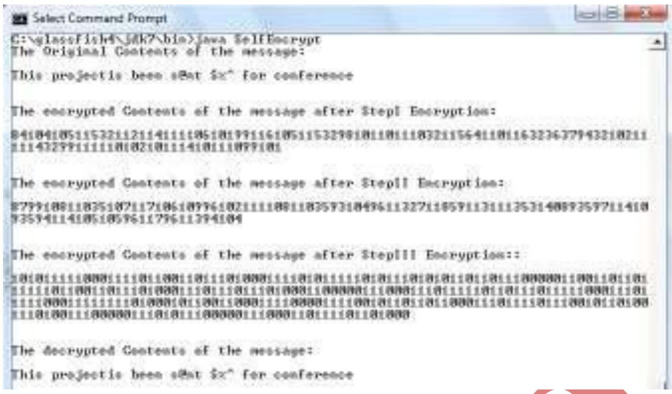
The random pattern is then subtracted from the integer number to get the ASCII code. The ASCII code is replaced by the respective character and stored in a String array. Now the decrypted original contents are read and displayed.

2.3 Step III

The encryption of converting the ASCII text to binary text has been accomplished This will be the final encrypted message.

The binary text is then again decrypted back to ASCII .Then the further reversing of the applied logic is been done to get back the original message.

2.3.1 The snapshot of the same is as follows:-



```
C:\glassfish4\jdk7\bin>java SelfEncrypt
The Original Contents of the message:

This projectis been s@nt $x" for conference


The encrypted Contents of the message after StepI Encryption:

84104105115321121141111061019911610511532981011011103211564110116323637943210211
1114329911111010210111141011109910 1

The encrypted Contents of the message after StepII Encryption:

87991001103510711710610979610211110011035973104961132711057113111253140093597114 10
93594114105105961179611394104

The encrypted Contents of the message after StepIII Encryption::

10101111100011110110011011101000111101011111010111010101101101110000001100110110 1
11111011001101110100011101110111010001100000111000111011111011011101111100011101
11110001111111101000101100110001111000011110010110110110001110111101110010110100
1110100111000000111010111000000111000110111110110100 0

The decrypted Contents of the message:

This projectis been s@nt $x" for conference
```

III CONCLUSION

In the current work we were able to write a code to implement custom made algorithm to encrypt and decrypt data. Although it is a symmetric encryption algorithm it is a more secure algorithm since the logic is not available in public domain. Moreover since the coding is done in Java it is platform independent and lightweight and special characters cannot be addressed in Caesar cipher whereas it is taken care in custom made encryption. These type of encryption programs have their applicability in day to day operations of various organizations which operate from multiple locations.

REFERENCE

Journal Papers:

[1] An ASCII value based data encryption algorithm and its comparison with other symmetric data encryption algorithms , International Journal on Computer Science and Engineering (IJCSE) ISSN : 0975-3397 Vol. 4 No. 09 Sep 2012 1650-1657.

[2] Aruljothi, S. ; Venkatesulu, M.Symmetric Key Cryptosystem Based on Randomized Block Cipher, Future Information Technology (FutureTech), 2010 5th International Conference

[3]Rahouma, K.H.A block cipher technique for security of data and computer networks ,Publication Year: 1999 ,