Global Journal of Computer Science and Technology
Volume 11 Issue 21 Version 1.0 December 2011
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Novel Advent for Add-On Security by Magic Square Intrication

By S. Praveen Kumar, K. Naveen Kumar, S. Sreenadh, B. Aravind, K. Hemnath Kumar

GITAM University

Abstract - The efficiency of a cryptographic algorithm is based on its time taken for encryption / decryption and the way it produces diverse cipher text from a clear text. The RSA, the extensively used public key algorithm and other public key algorithms may not guarantee that the cipher text is copiously secured. As an alternative approach to handling ASCII characters in the cryptosystems, a magic square implementation is deliberated of in this work. It attempts to augment the efficiency by providing add-on security to the cryptosystem. This approach will boost the security due to its complexity in encryption because it deals with the magic square. Here, encryption / decryption is based on numerals generated by magic square rather than ASCII values. In this, we add the ASCII value and the numeral in the consequent magic square. Because to encrypt the plaintext characters, their ASCII values are taken and if a character occurs in numerous places in a plaintext there is a possibility of same cipher text is produced. To surmount the problem, this paper attempts to develop a technique in which a constant is added for the recurring character.

Keywords : *ICryptography, Encryption, Decryption, Cipher text, Secret key, Public key, Steganography, Magic Square, ASCII value.*

GJCST Classification : *K.6.5*

Strictly as per the compliance and regulations of:

Novel Advent for Add-On Security by Magic Square Intrication

S. Praveen Kumar [α], K. Naveen Kumar [Ω], S. Sreenadh[β], B. Aravind [Ψ], K. Hemanth Kumar[¥]

Abstract - The efficiency of a cryptographic algorithm is based on its time taken for encryption / decryption and the way it produces diverse cipher text from a clear text. The RSA, the extensively used public key algorithm and other public key algorithms may not guarantee that the cipher text is copiously secured. As an alternative approach to handling ASCII characters in the cryptosystems, a magic square implementation is deliberated of in this work. It attempts to augment the efficiency by providing add-on security to the cryptosystem. This approach will boost the security due to its complexity in encryption because it deals with the magic square.

Here, encryption / decryption is based on numerals generated by magic square rather than ASCII values. In this, we add the ASCII value and the numeral in the consequent magic square. Because to encrypt the plaintext characters, their ASCII values are taken and if a character occurs in numerous places in a plaintext there is a possibility of same cipher text is produced. To surmount the problem, this paper attempts to develop a technique in which a constant is added for the recurring character.

Thus, instead of taking ASCII values for the characters to encrypt, preferably dissimilar numerals representing the position of ASCII values are taken from magic square. This proposed work provides another layer of security to any public key algorithms such as RSA, Elgamal etc., since, this model is acting as a wrapper to a public key algorithm, it ensures that the security is enhanced.

Keywords : Cryptography, Encryption, Decryption, Cipher text, Secret key, Public key, Steganography, Magic Square, ASCII value.

I. Introduction

Cryptography : Cryptography is the science of securing data. Classical cryptanalysis involves an appealing amalgamation of analytical interpretation, application of mathematical tools and pattern finding. The method of disguising plaintext in such a way as to hide its substance is called encryption. Encrypting plaintext results in unreadable gibberish called cipher text. The process of reverting cipher text to its original plaintext is called decryption.

The benefit of cryptography is that it works in a combination with a key ,a word number or phrase to encrypt the plain text. The same plain text encrypts to different cipher text with different keys. A key is a value that works with a cryptographic algorithm to fabricate a precise cipher text. The bigger the Key the more secure the cipher text.

Algorithms : There are two modules of key-based algorithms, symmetric (or secret-key) and asymmetric (or public-key) algorithms.

There is a problem of key distribution in the case of symmetric algorithms is solved with help of public key in asymmetric algorithms. The concept of Public key is using a pair of keys , a public key which encrypts data ,and a corresponding private key for decryption.

II. Literature Review

Steganography is the knack of writing concealed messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message, a form of security through obscurity. DES is the archetypal block cipher — an algorithm that takes a fixed-length string of plaintext bits and transforms it through a series of complicated operations into another cipher text bit string of the same length.PGP is often used for signing, encrypting and decrypting texts, E-mails, files, directories and whole disk partitions to increase the security of e-mail communications. It was created by Philip Zimmermann in 1991.

Gopinanath Ganapathy, and K. Mani (2009) developed "Add-on Security Model for Public-key Cryptosystem based on Magic Square Implementation". This paper discusses how encryption and decryption is done by using RSA cryptosystem along with the elements of doubly even magic square.

R. L. Rivest (1996) developed "Hand book of Applied Cryptography". In this, emphasis is on those that are both (believed to be) secure and practically useful.

Objective : The objective of the method is to provide a secure way of sending a message using some standards and to overcome from man in the middle attack, spoofing etc.

Using Magic square for add-on security : In our approach , add on security is provided by dealing with magic square. We encrypt a file of any length using a magic square. We append each element of magic square to each character in the file until the end of the

Author [α] : IT department, GIT, GITAM University 09989590690.
E-mail : spkmtech@gmail.com
Author [Ω] : IT department, GIT, GITAM University 09963778239.
E-mail : nkumarkuppili@gmail.com
Author [β] : IT department, GIT, GITAM University 09491552723.
E-mail : Sreenadh.sadasivuni@gmail.com
Author [Ψ]: IT department, GIT, GITAM University 09676067261.
E-mail : aravind.barla22@gmail.com
Author [¥] : IT Dept, GIT, GITAM University 9494329294,
E-mail : hemanth903@gmail.com

file. The size of the file may surpass the number of elements in the magic square. If number of characters is auxiliary than the number of elements then we start totaling from the first element of the magic square. . We browse the file, which is to be encrypted and a password which is known only to sender and receiver. Encryption is done, and a copy of encrypted file is saved automatically. For decryption we browse the file and same password is given. A copy of decrypted (original) file is also saved. It is a reverse process of encryption in which instead of adding elements we will subtract elements. In this way, add-on security is provided by dealing with magic square.

Magic Square : A magic square of order a is an arrangement of a2 numbers, usually distinct integers, in a square, such that the a numbers in all rows, all columns, and both diagonals sum to the same constant. A normal magic square contains the integers from 1 to a2. The term "magic square" is also sometimes used to refer to any of various types of word square.

Given an $a \times a$ normal magic square, suppose S is the number that each row, column and diagonal must add up to. Then since there are a rows the sum of all the numbers in the magic square must be . But the numbers being added are 1, 2, 3, ... a^2 , and so 1 + 2 + 3 + ...+ a^2 = a x S. In summation notation, $\sum_{i=1}^{n^2} i = a.S$. Using the formula for this sum, we have $a.S = \frac{a^2(a^2+1)}{2}$ and then solving for S gives $S = \frac{a(a^2+1)}{2}$. Thus, a 3×3 normal magic square must have its rows, columns and diagonals adding to

$$S = \frac{3\,(3^2+1)}{2} = \frac{30}{2} = 15$$

In the same way for a 4×4 square S= 34, Benjamin Franklin's 8×8 to S = 260, and so on.

The magic sum for an $a \times a$ normal magic square can be found by filling the square with the numbers 1, 2, 3, ... a^2 . first going across the top row, then the second row, and so on -- and then adding the numbers along either of the diagonals. For instance, to find the magic sum of a 4×4 normal magic square, we form the following square:

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

and then compute 1 + 6 + 11 + 16 = 34. See if you can figure out why this works for any $a \times a$ normal magic square. As defined above, a normal magic square uses the numbers 1, 2, 3 , . .. a^2 . Some people relax this restriction to permit any positive integers, calling the resulting square simply a magic square -- without the adjective "normal." It is easy obtain magic squares of this generalized type from an existing magic square. One way is simply to multiply every number used by some positive constant, and/or add a positive constant to every number used. It should be easy for you to see why this will always work.A more interesting problem, however, concerns the process of creating normal magic squares. For odd values of a, there is a simple procedure for constructing a normal magic square.

In our approach, we add each element of magic square to each character in the file until the end of the file. The size of the file may exceed the number of elements in the magic square. If number of characters is more than the number of elements then we start adding from the first element of the magic square.

Code snippet:

Encryption:

In the beneath code we can see the processing of the magic square and the encryption of the data by using it. First the magic square serial (password) is processed and then using the serial given the data is encrypted by totaling each element of the magic square to each character in the file until the end of the file.

```
for (int i = 2; i <= n*n; i++)
 if (magic[(row + 1) % n][(col + 1) % n] == 0)
row = (row + 1) % n; col = (col + 1) % n;
 else
 row = (row - 1 + n) % n;   // don't change col
 magic[row][col] = i;
for(int i=0;i<n;i++)
for(int j=0;j<n;j++)
c=fis.read();
if(c!=-1)
 s=(char)(c+magic[i][j]);
fos.write(s); l++;
if(l==(n*n))
i=-1; l=0;
fos.close(); fis.close();
 i=n; j=n;
```

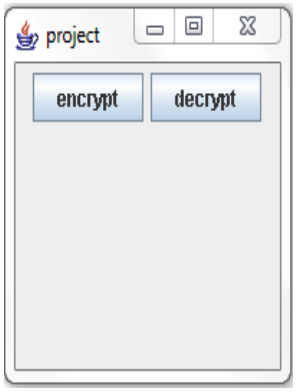
Decryption:

The following code decrypts the cipher into Plain text.

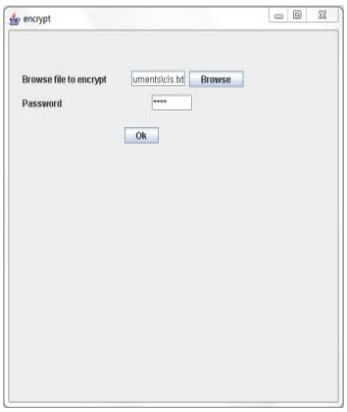
During the process of decryption the magic square serial (password) is to be entered ,if the password matches with the password given during encryption then the cipher is converted into plain text.

```
for (int i = 2; i <= n*n; i++)
 if (magic[(row + 1) % n][(col + 1) % n] == 0)
row = (row + 1) % n; col = (col + 1) % n;
 else
 row = (row - 1 + n) % n;   // don't change col
 magic[row][col] = i;
for(int i=0;i<n;i++)
for(int j=0;j<n;j++)
c=fis.read()
if(c!=-1)
s=(char)(c-magic[i][j]);
fos.write(s) l++;
if(l==(n*n))
i=-1; l=0;
else
fos.close(); fis.close();
i=n; j=n;
```

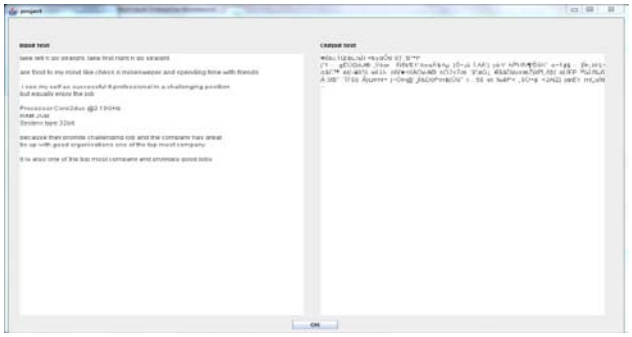
III. Result



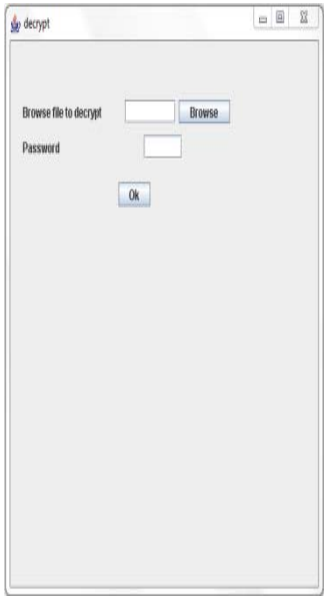
Choose encrypt or decrypt



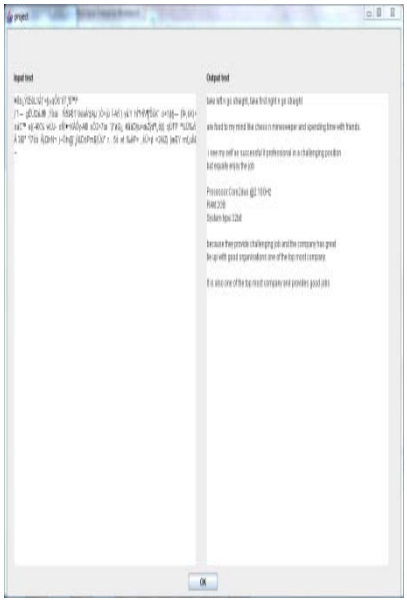
Browse the file for Encryption and enter password (magic square sequence)



Encryption from plain text to cipher



Browse file for Decryption and enter the password specified during encyption



Decryption from cipher to plain text

IV. Conclusion & Future Enhancement

An alternative advance to existing ASCII based cryptosystem a number based approach is thought of an implemented. This methodology will add on one more layer of security, by adding numerals of the magic square for the text. It provides security to the files in PCs and also can be transferred through pen drives. We can apply this algorithm for any type of files like .txt, .doc, .jpeg etc. The size of the file is very small to carry and is simply a jar file.

Further we can feed into any public key algorithms like RSA, ElGamal etc. Thus it provides add on security to existing algorithms.

References References Referencias

1. Gopinanadh Ganapathi, and K.Mani, "Add on security model for public key Cryptosystem based on magic square implementation", India, 2009.
2. R. L. Rivest, "Hand book of Applied Cryptography", 1996.
3. Herbert Schildt, "The Complete Reference", Seventh Edition, Tata McGraw-Hill Publishing Company Limited.
4. Patrick Naughton, Michael Morrison, :The Java Handbook", Publisher: Osborne/ McGraw-Hill.
5. Benson, W.H. & Jacoby, O. ; New Recreations with Magic Squares. Dover Publications Inc. New York.
6. Gauthier, N. 'Singular matrices applied to 3x3 Magic Squares'. Math. Gazette.81, (1997).
7. Thompson, A.C. ; 'Odd Magic Powers'. American Math. Monthly, 101(4),(April 1994).
8. Ollerenshaw, K. & Bondi, H. ; 'Magic Squares of Order 4'. Phil. Trans. Royal Soc. London.
9. Dudeney, H.E. ; The Canterbury Puzzles (and other curious problems). T. nelson & sons Publishers. 2nd Edition. 1927.
10. Gardner, M. 'Mathematical games: a breakthrough in magic squares and the first perfect magic cube.' Scientific American.
11. Fletcher, T.J.; Linear Algebra through its Applications. Van Nostrand Reinhold,New York. 1972.
12. Ward, James E. III; 'Vector spaces of Magic Squares.', Math. Magazine .
13. van den Essen, A. ; 'Magic Squares and Linear Algebra', American Math.Monthly.
14. Heinrich, C.J.; 'Magic Squares and Linear Algebra', American Math. Monthly.
15. Leinbach C. & Pountney D.C. 'Appropriate use of Computer Algebra Systems in Teaching Mathematics' Pennsylvania Council of Teachers of Mathematics.