A New Symmetric Key Encryption Algorithm Using Images as Secret Keys

Conference Paper · December 2015

DOI: 10.1109/FIT.2015.12

Mohsin Shah
Hazara University
21 PUBLICATIONS **79** CITATIONS
SEE PROFILE

Zakir Khan
Hazara University
28 PUBLICATIONS **69** CITATIONS
SEE PROFILE

Toqeer Mahmood
University of Engineering and Technology, Taxila
30 PUBLICATIONS **145** CITATIONS
SEE PROFILE

2015 13th International Conference on Frontiers of Information Technology

A New Symmetric Key Encryption Algorithm using Images as Secret Keys

Mazhar Islam[1], Mohsin Shah[2], Zakir Khan[3], Toqeer Mahmood[4], Muhammad Jamil Khan[5]

[1,5] Department of Telecommunication Engineering, University of Engineering and Technology Taxila, Pakistan
Email: mazharislam87@gmail.com, muhammad.jamil@uettaxila.edu.pk

[2,3] Department of Information Technology, Hazara University Mansehra, Pakistan
Email: {syedmohsinshah, zakirk2012}@gmail.com

[4] Department of Computer Engineering, University of Engineering and Technology Taxila, Pakistan
Email: toqeer.mahmood@yahoo.com

Abstract—**Symmetric key cryptography is a common cryptographic technique using the same key at both the transmitter and receiver side. The main advantage of symmetric key encryption is its less computational cost compared to its counterpart-public key encryption. In this work a new symmetric key encryption scheme is proposed. Rather than using normal binary secret keys, our technique uses images as secret keys. Message letters are converted into their corresponding 8-bit binary codes. These 8-bit codes are scanned for image pixel values which are represented by the same 8-bit codes. When a match is found between the message 8-bit code and pixel values codes that location of the pixel is saved in a separate file. Instead of saving pixel locations as (x, y) coordinates, their locations are saved as one value column wise. After having all matches, pixels locations are transmitted as cipher text. The receiver side will scan the same image for those locations and will pick values at those locations which represent message codes. The main advantage of this scheme is its high security as its key size is very large.**

Keywords— symmetric key; encryption; image; cipher text

I. INTRODUCTION

Cryptography is a mechanism in which information is encrypted or transformed into some unreadable format called cipher text. Only the authorized user having the secret code can decrypt or decipher the received message. A number of encryption techniques are available in literature. All encryption algorithms can be divided in two major groups which are [15]: (1) Symmetric key encryption algorithms, and (2) Asymmetric key encryption algorithms.

In Symmetric key encryption or secret key encryption, only one key is used for both encryption and decryption. In asymmetric key encryption two keys are used i.e. public key and private key. This type of encryption is also called public key encryption. Symmetric key encryption algorithms are faster than asymmetric key encryption algorithms [1, 2]. Key should be exchanged between the communicating entities before the transmission of data. Key is one of the important factors of these algorithms. Weak keys can be easily attacked by the attackers as compared to longer keys which are difficult to break [3]. Symmetric key encryption algorithms are still widely used as powerful techniques in insecure communication channel [4]. Some widely used encryption schemes are discussed below.

DES was the first block cipher. IBM designed DES and it was adopted by national bureau of standard in 1977 [13] which is now called National institute of standard and technology (NIST) [12]. It was declared as an official Federal Information Processing Standard (FIPS) [5] in USA and it was used all over the world. DES algorithms take 64 bits plaintext as an input and transform it into 64 bits cipher text as output. The key length of DES is also 64 bits. DES is called a complex block cipher as it has 16 blocks of complex round ciphers and each block itself has a complex function [5].

One of the drawbacks of DES is that the key length was too short. To overcome this problem DES was enhanced and 3DES was proposed. The size of the key is increased to 192 bits instead of 64 bits while the block size remains the same which is 64 bits [6]. The encryption process of 3-DES is similar to DES encryption but to increase security level, DES rounds are applied three times [14]. DES has two versions available, (i) DES with two keys and (ii) DES with three keys. Many algorithms uses 3DES with three keys. In terms of performance DES is faster than 3DES [7].

Blowfish is a replacement algorithm for DES. It is also a block round cipher having a block size of 64 bits. Its key length varies from 32 bits to 448 bits [14]. By default 128 bits key length is used [14]. It is available free to all users and also it is license free. Rounds of blowfish algorithms varies having maximum 14 rounds [3].

In 3-DES the key size was increased to add security but the whole process became very slow. So National Institute of standard and technology recommended AES. It is also called Rijndael algorithm, recommended after a competition held in 1997 to select the best encryption algorithm [5]. It is available in three different variants on the basis of its key length. It has 128, 192 or 256 bits key [6]. It has also multiple rounds and number of rounds depends upon key length [13]. For a key size of 128 bits, there are 10 rounds [16]. For 192 bit key size, 12 rounds are used and for 256 bits key length 14 rounds are used [6]. Security level of AES is much better than DES and 3 DES [5-7]. In AES key can only be braked if the attacker tries

978-1-4673-9666-0/15 $31.00 © 2015 IEEE
DOI 10.1109/FIT.2015.12

IEEE computer society

all the bit combination which is a difficult process [5]. As compared to DES and 3DES, AES is flexible and fast [6].

II. LITERATURE REVIEW

Encryption algorithms were compared by Hirani in [8] and he concluded that AES has an edge over other encryption algorithms in terms of efficiency and speed. In case of data transmission different schemes have a minor performance difference. In [9] study of different symmetric key algorithms is performed. Files of variable sizes and contents were encrypted and performance was measured. It was concluded that the performance of blowfish was better compared to other encryption algorithms. It also concluded that compared to DES and 3DES, performance of AES was better. In addition throughput of DES was triple as compared to 3DES. Time consumption of RC2 was not good as compared to other algorithms. Different symmetric key algorithms were compared in terms of energy consumption performance in [10]. 5 MB file was taken and after 600 encryptions, it was found that 3-DES consumed 65% of battery power and further encryptions cannot be done because battery was consumed. In [11] a comparison is performed among six encryption algorithms. Each algorithm was tested for different setting such as data blocks having different sizes, multiple types of data, CPU time and different size of key. Two different hardware platforms were used for testing. The result favored blowfish in terms of efficiency and also favored AES over DES and 3DES. Hirani increased the key size of AES by 64 bits [8]. He concluded that energy consumption was increased by 8% with no data transfer. Power was saved by reduction of rounds which resulted in insecure AES.

New techniques involving both cryptography and steganography are the focus of research now a days. As cryptographic keys are very long and has to be protected from unauthorized access, a new technique generating secret keys directly from image properties was proposed by [17]. The image used for the generation of secret key is shared through some secret means between the sender and receiver. So it is very difficult for the attacker to know the secret key unless he has the same image and the algorithm used for the generation of secret key. For securing images an efficient technique using image to generate secret key is proposed by [18]. In this work an image is divided into group of pixels. The binary values of pixel intensity are calculated for each pixel of the image. These binary values are converted into their corresponding decimal values. All the pixel having the same decimal values are grouped together. Root mean square (RMS) value of each group is calculated and secret key is generated using these RMS values. The generated key is used to encrypt digital images over the channel.

In [19] a threshold of color levels is set to embed secret bits in the least significant bits (LSB) of cover image pixels. This technique exploits weakness in Human Visualization System (HVS) for information hiding. Bit complementation is used to hide secret bits in images [20]. Secret bits are compared with the LSB of pixels of cover image. Secret bits are embedded when they have a match with the LSB of cover image pixels. Otherwise, complements of secret bits are replaced with the LSB. This technique embeds a high payload of secret bits but lacks complexity.

The rest of the paper is organized as follows. Section III describes the encoding and decoding in detail. Experimental results are given in Section IV and conclusion in Section V.

III. PROPOSED METHOD

A. Sender Side

All images whose pixel values can represent all characters can be used as secret key. Character probability is checked for any image that is to be used as secret key. If any character is missing in secret key image, we have two option to deal with: (1) Reject image, (2) Modify image by inserting pixel having value equal to the character value. After selecting image as secret key, read the secret message that is to be transmitted. If the characters of the secret message are present in image in the form of pixel values, find the location of every character. There can be multiple locations for one character, take any random location. After finding all the locations of characters, write these locations in a separate file. Instead of transmitting secret message file, transmit the file having location information of characters. Only that receiver will be able to read the message if it has the same secret key image. The encryption process is shown in Fig. 1.

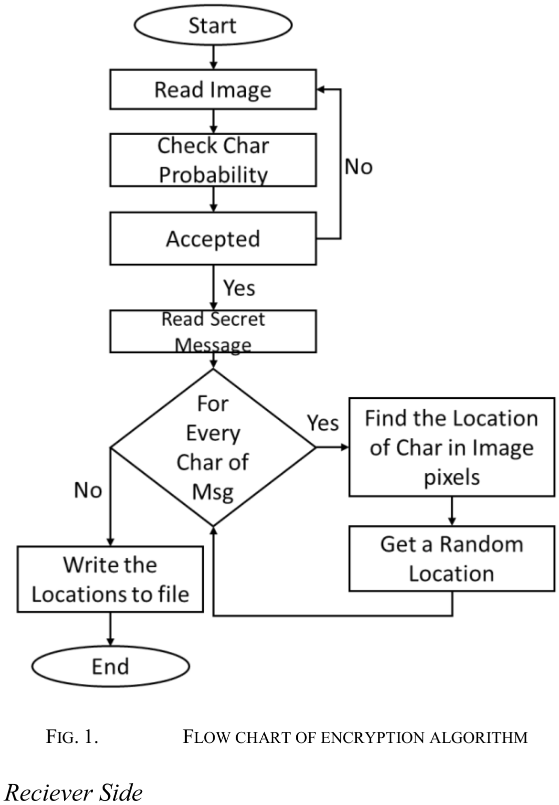

FIG. 1. FLOW CHART OF ENCRYPTION ALGORITHM

B. Reciever Side

On receiving the file having locations of the characters, read the same image used as secret key. Read all the secret locations and for every location, find the corresponding pixel value. After finding pixel values for all of the received locations, convert those pixels values to corresponding characters. Write these characters to a separate file. The decryption process is shown in the Fig. 2.

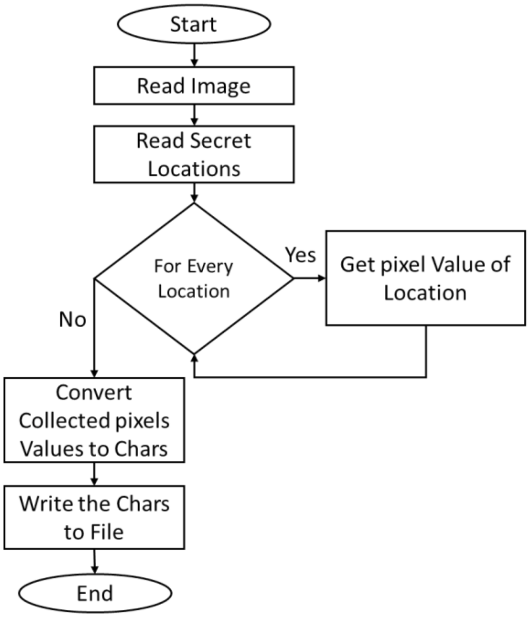

FIG. 2. FLOW CHART OF DECRYPTION ALGORITHM

C. Description with the help of example

We assume that image as a secret key is exchanged securely between sender and receiver before the transmission. Suppose plaintext "HELLO" is to be transmitted. The following steps are performed in the transmission process.

- ASCII values for each character is taken and then converted to 8 bit binary codes.

Character	ASCII	Binary code
H	72	01001000
E	69	01000101
L	76	01001100
L	76	01001100
O	79	01001111

- Image pixel values are scanned to find these ASCII codes or their binary code. Fig. 3 shows pixel value of an image.

  After image is completely scanned, pixel values equal to ASCII for "HELLO" are shown in Fig. 3.

- Next step is to find pixel locations as (x, y) coordinates, and save them as value column wise. We assign column wise values to (x, y) coordinates as shown in Fig. 4.

  Column wise values for the word "HELLO" are stored, converted to binary and then transmitted as cipher text.

32	45	112	48	233	123	11	21
255	253	115	195	245	222	125	227
127	111	255	112	135	28	110	228
38	113	69	173	189	37	198	125
22	137	46	148	76	25	112	79
72	23	136	24	185	36	187	127
77	47	73	77	154	76	169	140
82	49	48	221	247	49	187	140

FIG. 3. PIXEL VIEW OF IMAGE

1	9	17	25	33	41	49	57
2	10	18	26	34	42	50	58
3	11	19	27	35	43	51	59
4	12	20	28	36	44	52	60
5	13	21	29	37	45	53	61
6	14	22	30	38	46	54	62
7	15	23	31	39	47	55	63
8	16	24	32	40	48	56	64

FIG. 4. PIXEL COORDINATE REPRESENTATION OF IMAGE

- The relationship between plaintext and cipher text is as follow:

Character	ASCII	Binary code	Coordinate value	Coordinate Binary
H	72	01001000	6	00000110
E	69	01000101	20	00010010
L	76	01001100	37	00100101
L	76	01001100	47	00101111
O	79	01001111	61	00111101

Plain Text	H	E	L	L	O
Plain Text	01001000	01000101	01001100	01001100	01001111
Cipher Text	00000110	00010010	00100101	00101111	00111101

On the receiver side Cipher text received is

00000110 00010010 00100101 00101111 00111101

The received 8-bit chunks of bits are the locations of the plain text 'HELLO'. Again image will be scanned to find out the pixel values at these coordinates. Pixel values at the above received locations are

01001000 01000101 01001100 01001100 01001111

These binary pixel values are converted to corresponding characters and saved in separate file. The characters received are:

H E L L O

Which is the same message as transmitted by the sender.

IV. Experimental Results

The proposed technique is implemented using MATLAB on Pentium IV 2.4 GHz CPU for calculating the computational cost of the technique. Several files of different sizes are taken and encrypted and decrypted. For implementation of the proposed technique a grayscale image as shown in Fig. 5 is used as a cover image. Encryption time is defined as the time taken by any algorithm to convert plain text into cipher text. Encryption time is used in finding throughput of encryption algorithm which gives the speed of encryption algorithm. Throughput is calculated by dividing total bytes of plaintext by the total encryption time. Execution time (Encryption + Decryption) of the proposed technique is compared with different existing symmetric encryption techniques in the Table 1. The comparison of Throughput performance of each algorithm is also shown in Fig. 6.

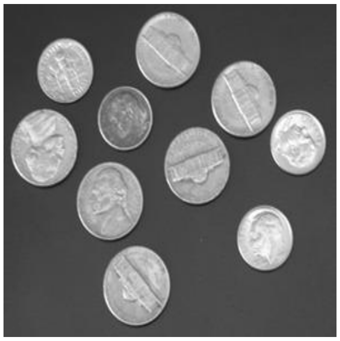

FIG. 5. Cover Image Coin.Png

TABLE 1. Comparative execution time (in millisecond) of Symmetric algorithms with different packet size

Input Size (KB)	AES	3DES	DES	Proposed Technique
98	119	107	79	7.18
118	96	99	75	8.62
200	150	138	106	14.77
494	188	188	119	36.52
642	313	254	156	47.6
1388	352	373	264	102.6
1798	429	470	392	132.3
1926	372	460	407	142.6
10690.56	1892	2301	2079	792.8
14620.672	2248	2887	2648	1084.1
Throughput (MB/Sec)	**5.3**	**4.5**	**5.2**	**136.06**

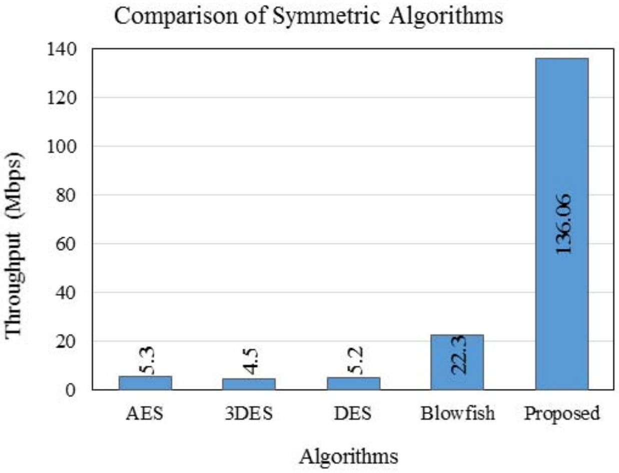

FIG. 6. Throughput of each encryption algorithm

From Table 1 and Fig. 6, it is concluded that the performance of proposed technique is better than AES, 3DES, and DES. If average throughput of encryption and decryption is considered, the proposed technique is much better than the existing symmetric key techniques.

V. Conclusion

In this paper, a new symmetric key encryption technique is implemented using image as secret key. The pixel locations of characters of message text are transmitted as cipher text instead of transmitting actual message text. On receiving side, receiver extracts actual text by using the locations of secret key image pixels. Security and computational analysis is presented which showed that the proposed technique has low computational cost and high security because of the huge key size. The computational cost of the proposed technique is calculated for different message file sizes and compared with the computational costs of the state of the art encryption techniques available in literature. Computational analysis shows that the proposed technique is well ahead of the existing techniques.

References

[1]. J.J. Amador and R.W. Green, "Symmetric-key block cipher for image and text cryptography," International Journal of Imaging Systems and Technology, vol. 15, pp. 178-188, 2005.

[2]. A. Ramesh and A. Suruliandi, "Performance analysis of encryption algorithms for Information Security," International Conference on Circuits, Power and Computing Technologies, 2013, pp. 840-844

[3]. M. Umaparvathi and D.K. Varughese, "Evaluation of symmetric encryption algorithms for MANETs," IEEE International Conference on Computational Intelligence and Computing Research, 2010, pp. 1-3

[4]. D. Trinca, "Sequential and parallel cascaded convolutional encryption with local propagation: Toward future directions in symmetric cryptography," Third International Conference on Information Technology: New Generations, 2006, pp. 464-469

[5]. O. Verma, R. Agarwal, D. Dafouti, and S. Tyagi, "Peformance analysis of data encryption algorithms," 3rd International

Conference on Electronics Computer Technology 2011, pp. 399-403

[6]. S.R. Masadeh, S. Aljawarneh, N. Turab, and A. M. Abuerrub, "A comparison of data encryption algorithms with the proposed algorithm: Wireless security," Sixth International Conference on Networked Computing and Advanced Information Management, 2010, pp. 341-345

[7]. D.S.A. Elminaam, H. M. Abdual-Kader, and M.M. Hadhoud, "Evaluating The Performance of Symmetric Encryption Algorithms," International Journal of Network Security, vol. 10, pp 216-222, 2010

[8]. S.A. Hirani, "Energy Consumption of Encryption Schemes in Wireless Devices" PhD Thesis, University of Pittsburgh, 2003.

[9]. A. Nadeem and M. Y. Javed, "A performance comparison of data encryption algorithms," First international conference on Information and communication technologies, 2005, pp. 84-89.

[10]. P. Krishnamurthy and N. Ruangchaijatupon, "Encryption and Power Consumption in Wireless LANs-N," Third IEEE Workshop on Wireless LANs-Newton, Massachusetts, 2001.

[11]. G. Ramesh and R. Umarani, "A Comparative Study of Six Most Common Symmetric Encryption Algorithms across Different Platforms," International Journal of Computer Applications, vol. 46, pp. 6-9, 2012.

[12]. W. Stallings, Network security essentials: applications and standards: Pearson Education India, 2007.

[13]. A.K. Mandal, C. Parakash, and A. Tiwari, "Performance evaluation of cryptographic algorithms: DES and AES," IEEE Students' Conference on Electrical, Electronics and Computer Science, 2012, pp. 1-5.

[14]. S.P. Singh and R. Maini, "Comparison of data encryption algorithms," International Journal of Computer Science and Communication, vol. 2, pp. 125-127, 2011.

[15]. M. Agrawal and P. Mishra, "A comparative survey on symmetric key encryption techniques," International Journal on Computer Science and Engineering, vol. 4, pp. 877-882, 2012.

[16]. T. Hoang, "An efficient FPGA implementation of the Advanced Encryption Standard algorithm," IEEE International Conference on Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2012, pp. 1-4.

[17]. T.S. Barhoom, Z.M. Abusilimiyeh, “A Novel Cryptography Method Based on Image for Key Generation”, IEEE Proceedings on the Palestinian International Conference on Information and Communication Technology, 2013, pp. 71-76.

[18]. A. Sahu, Y. Bahendwar, S. Verma, P. Verma, “Proposed Method of Cryptographic Key Generation for securing Digital Image”. International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 2(10), 2012, pp. 285-291.

[19]. Z. Khan, M. Shah, M. Naeem, T. Mahmood, S.N.A. Khan, N. Amin, D. Shahzad, “Threshold based Steganography: A Novel Technique for Improved Payload and SNR”, International Arab Journal of Information Technology [Online].

[20]. Z. Khan, M. Shah, M. Naeem, D. Shahzad, T. Mahmood, “LSB Steganography using Bits Comlementation”, International Conference on Chemical Engineering and Advanced Computational Technologies (ICCEACT), November 24-25, 2014 Pretoria (South Africa).