VOL. 12, NO. 3, FEBRUARY 2017 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

INTRODUCING DYNAMIC P-BOX AND S-BOX BASED ON MODULAR CALCULATION AND KEY ENCRYPTION FOR ADDING TO CURRENT CRYPTOGRAPHIC SYSTEMS AGAINST THE LINEAR AND DIFFERENTIAL CRYPTANALYSIS

M. Zobeiri and B. Mazloom-Nezhad Maybodi
Faculty of Electrical and Computer Engineering, ShahidBeheshti University, Tehran, Iran
E-Mail: m_zobeiri@sbu.ac.ir

ABSTRACT

In modern cryptography, the P-Boxes and S-boxes are used to make the relation between the plaintext and the cipher text difficult to realize. P-Box is a tool that changes the order of the input bits and they appear in the output. In this case, the key is order transmission of input bits in output bits.An S-box is a basic component which performs substitution. In this paper, a new method for the production of P-Box with the help of modular computing is introduced in a way that the order of the input bit transfer in output is the function of the encryption key.Next, development of this algorithm is used to make S-box structure.And also if the current symmetric encryption methods used, the P-Box and S-Box keysare derived from the key encryption method. In general, there is no need to increase the length of current key to use the P-Box and S-Box recommended.

Keywords: dynamic P-Box and S-box, modular calculations, prime numbers, key encryption, code breaking.

1. INTRODUCTION

Cryptography is the science of providing security for information. Today, cryptography is a cornerstone of the modern security technologies used to protect information and resources on both open and closed networks.

In modern cryptographic systems, the data in several stages (known as Round) are combined. Through using the powerful operator Xor they contribute in the process of combining other effective factors. Two of the easiest components of encryption which are in combination with other components are P-BOX and S-box. Powerfulness of each of these components can produce stronger encryption methods.

In most symmetric encryption methods permutation has a special position in the implementation of encryption algorithm. After data preparation, permutation procedure (similar to DES[1], etc.), is used as the first stage of data processing and also as the final processing in the delivery of encrypted blocks.

Substitution boxes (S-boxes) have been extensively used in almost all conventional cryptographic algorithms, such as data encryption standard (DES) and advanced encryption standard (AES[2]). S-boxes are the only non-linear components in these cryptosystems. The strength of cryptographic algorithms is determined by these non-linear S-boxes, so the construction of cryptographically strong S-boxes is important in the design of secure cryptosystems. S-boxes are usually designed using the non-linearity criteria inspired by linear and differential cryptanalysis. It is the objective of most designs to keep the maximum differential and linear approximation probabilities of an S-box as small as possible.

Substitution boxes give a DES-like cryptosystem with the perplexity property portrayed by Shannon in his excellent paper [3]. They are the main nonlinear part of these ciphers.

Some scholars made the current P-Box and S-Box tables more efficient [4-6], such as AEScryptography. Some researchers considered the permutation and its applications on other issues, and the new methods defined like: notion of mesh patterns permutation [7-9] and permutation polynomials [10, 11].

The introductory permutations and final permutations, permutations operations in different rounds, in most ways, the permutations are based on a fixed table form. This work, especially in preliminary and final permutations, does not make higher security in encryption process. According to Kirchhoff's Laws encryption algorithm should not be hidden. If someone wants this operation and other permutations, in the Round of encryption in fixed table, are prevented, the permutations should be identified separately so that at the time of decryption the counter operation is performed correctly. To do so, there is a requirement to increase the length of the required information of encryption and decryption(the encryption key).In this paper, a method of permutations, which is dependent on the encryption key, through the science of number theory and modular computational science will be introduced.

In symmetric encryption methods, at any stage, a block of data that should be encrypted is selected after implementing various stages of encryption, the encrypted form is calculated and the next block is selected again. In recommended method, the required permutation table of encryption stage of the process (Rounds) is calculated in terms of the key once. These tables will be fixed for all the blocks. This function will be presented in the second part. In the third part, making an S-box based on encryption key is discussed.

After the introduction of the proposed algorithms, in the fourth part the formation of the required subkeys in

creating permutation and substitution tables of proposed algorithms from the main key encryption is discussed. In generating subkey in this section, an algorithm is suggested, preventing form increasing the length of encryption key in permutation and substitution operation. In the fifth part, a suggestion will be presented to adding the proposed algorithms to current encryption method without changing their methods and the conclusions of the paper will be presented in section sixth.

2. INTRODUCTION OF PROPOSED ALGORITHM TO MAKE DYNAMIC P-BOX

In Figure-1, the permutation operation of one input block is shown. In cryptographic operations a selected block of inputs, numerous permutations and substitutions may be used in different rounds. In proposed algorithm, each of permutations based on encryption key is calculated and the table of these permutations for all the input block which should be coded, are fixed.

To create permutations, it is assumed that the input data can be binary or character or any other symbol, they are located in n house block, (each of the houses include a bit or a character or a symbol is but without reducing the whole issue, each house should be considered as including a bit). The output block contains permutations of the n house based on the encryption key. In this way, if the length of the permutation table, in terms of current encryption method which proposed algorithm is to be added, is not a unique number the selection of the length of the permutation table can be given to the user.

Different methods of encryption include P-Boxes or substitution tables with different lengths. The capability of different length of permutation table of the proposed algorithm made it applicable in all possible current encryption methods. Because of this capability even the entire entry can be considered as a block and this method can be applied to all the input data. When the length of the selected block is bigger we need more blocks of memory and higher operation volume to calculate the permutations of the required input block houses.

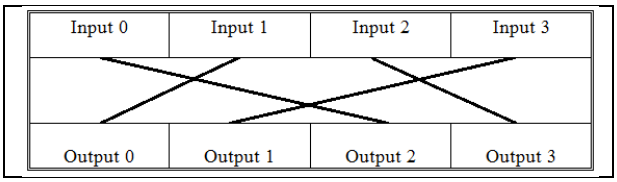

Figure-1. Showing the performance of a P-Box.

The length of the Input block to create permutations is considered as n. Two blocks of n houses are considered, one block as an input block another block as an output block. The content of this n house input block during the execution of the algorithm are located in the output block (Figure-2).The houses of input block are numbered from 0 to n-1 and at first the number of each input should be put into it.

Input Block	0	1	2	3	…	n-1
Home number	0	1	2	3		n-1

Figure-2. The Schematic presentation of an input block.

In each step, one input house will be selected according to the presented algorithm and its content is located in the first empty output block. The content of the next input houses will be transferred to the next house to the left side. For example, it is assumed that in the first step house 3 is selected, its content is located in the house zero of output (the first output house), after that input houses will be transferred from selected house to the next house in the left side (Figure-3).

Input Block(before beginning)								**Input Block(after beginning)**							
				select											
Content	0	1	2	3	…	n-1	⇒	Content	0	1	2	4	…	n-1	
No:	0	1	2	3	…	n-1		No:	0	1	2	3	…	n-2	n-1
Output Block(before beginning)								**Paste**	**Output Block(after beginning)**						
Content							⇒	Content	3						
No:	0	1	2	3	…	n-1		No:	0	1	2	3	…	n-2	n-1

Figure-3. The place of the first selected house in output block and transmission of other inputs.

According to the algorithm, this action is repeated n times to transfer all the input houses to the output houses; therefore, the order of the input permutation is obtained.

Algorithm implementation

1) Preparing the amounts of variable of the house selection.

1-1) in the implementation of this algorithm the user introduces $n_1$as the length of the input and four arbitrary numbers A', B', C', D' as the input of the function.

1-2) Consider A number equal to A'+ 1000 and B to number B'+1000, (it ensures the least amount of A and B is 1000)

1-3) Consider number C equals C'+100 and D equals D'+10 (this ensures the least amount of C equals100 and D equals 10)

1-4) Select Pc number as the largest prime number that Pc<Cand P = (Pc,…., 7,5,3,2) is defined to include all the prime numbers less than or equal Pc.

1-5) Select n number as the smallest prime number that is$n_1 \leq n$.

This action is employed to use modular computing features and reduce the volume of calculation.

1-6) the number m equal to the numbers of input block which are not selected yet to be inserted, at first$m = n_1$ after each step m will lose one unit.

2) The selection of input block house and its transmission to the output block

In step K, the number of selected input block house to be inserted in output block is calculated as follows:

Number of selected house=

$$\left[\left((A - P[k \bmod D])^{(B+P[k \bmod D])}\right) \bmod n\right] \bmod m \quad (1)$$

Where:

A, B, C and D are constant numbers that are chosen by user or program.

k is the number of step that the house is selected.

P[k] equals the amount of K p arrow.

n is the smallest prime number that is$n_1 \leq n$.

mis the numbers of input block which are not selected yet.

As the positioning of the prime number among numbers is not according to any order, the number of each selected house in comparison to the previous house is completely different and nonlinear. The presence of D will increase the level of nonlinearity. The sample of implication is present in example 1.

Example 1: If length of selected Input block is 8 (n1=8) and other parameters are (A'=200, B'=170, C'=90, D'=20), the process of permutation of input block by proposed method in this study presents below:

(A= A'+1000=1200, B= B'+1000=1170, C= C'+100=190, D=D'+10=30,$P_C = 181$, n=11)

$P[k] = \{181,179,177,173,167,163,157, \ldots ,3,2\}$

You can see the processing and result of example 1 in Table-1 and its p-box in Figure-4.

Table-1. Calculating the P-box of the example 1 in 8 steps.

Selected home Number = $\left[\left((A - P[kmodD])^{(B+P[kmodD])}\right)modn\right]modm$, n=11								
Input BlockBefore starting process								
Home Number	0	1	2	3	4	5	6	7
Contain	0	1	2	3	4	5	6	7
Output BlockBefore starting process								
Home Number	0	1	2	3	4	5	6	7
Contain								
Step 1								m=8
Selected home Number=$((1200 - P[0])^{(1170+P[0])}$ Mod 11) Mod 8 =7								
Input Block								
Home Number	0	1	2	3	4	5	6	7
Contain	0	1	2	3	4	5	6	7
Output Block								
Home Number	0	1	2	3	4	5	6	7
Contain	7							
Step 2								m=7
Selected home Number =$((1200 - P[1])^{(1170+P[1])}$ Mod 11) Mod 7 =5								
Input Block								
Home Number	0	1	2	3	4	5	6	7
Contain	0	1	2	3	4	5	6	E
Output Block								
Home Number	0	1	2	3	4	5	6	7
Contain	7	5						
Step 3								m=6
Selected home Number =$((1200 - P[2])^{(1170+P[2])}$ Mod 11) Mod 6 =0								
Input Block								
Home Number	0	1	2	3	4	5	6	7
Contain	0	1	2	3	4	6	E	E
Output Block								
Home Number	0	1	2	3	4	5	6	7
Contain	7	5	0					
Step 4								m=5
Selected home Number =$((1200 - P[3])^{(1170+P[3])}$ Mod 11) Mod 5 =4								
Input Block								
Home Number	0	1	2	3	4	5	6	7
Contain	1	2	3	4	6	E	E	E
Output Block								
Home Number	0	1	2	3	4	5	6	7
Contain	7	5	0	6				
Step 5								m=4
Selected home Number =$((1200 - P[4])^{(1170+P[4])}$ Mod 11) Mod 4 =2								
Input Block								
Home Number	0	1	2	3	4	5	6	7
Contain	1	2	3	4	E	E	E	E
Output Block								
Home Number	0	1	2	3	4	5	6	7
Contain	7	5	0	6	3			

Step 6									m=3								
Selected home Number =$((1200 - P[5])^{(1170+P[5])}$ Mod 11) Mod 3 =2																	
Input Block									Output Block								
Home Number	0	1	2	3	4	5	6	7	Home Number	0	1	2	3	4	5	6	7
Contain	1	2	4	E	E	E	E	E	Contain	7	5	0	6	3	4		
Step 7									m=2								
Selected home Number =$((1200 - P[6])^{(1170+P[6])}$ Mod 11) Mod 2 =0																	
Input Block									Output Block								
Home Number	0	1	2	3	4	5	6	7	Home Number	0	1	2	3	4	5	6	7
Contain	1	2	E	E	E	E	E	E	Contain	7	5	0	6	3	4	1	
Step 8									m=1								
Selected home Number =$((1200 - P[7])^{(1170+P[7])}$ Mod 11) Mod 1 =0																	
Input Block									Output Block								
Home Number	0	1	2	3	4	5	6	7	Home Number	0	1	2	3	4	5	6	7
Contain	2	E	E	E	E	E	E	E	Contain	7	5	0	6	3	4	1	2

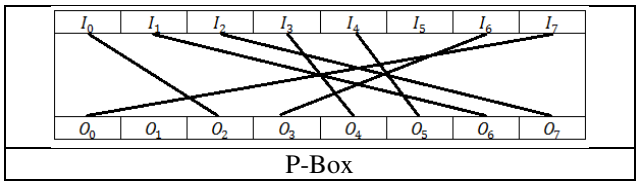

Figure-4. Showing the performance of P-Box in example 1.

3. MAKING DYNAMIC S-BOX BASED ON ENCRYPTION KEY

In cryptography, an S-box (substitution-box) is a basic component of symmetric key algorithms which performs substitution. In block ciphers, they are typically used to obscure the relationship between the key and the cipher text. In general, an S-box takes some number of input bits and transforms them into some number of output bits. An S-box can be implemented as a lookup table. Fixed tables are normally used, as in the Data Encryption Standard (DES), in this paper; the tables are generated dynamically from the key. One of the ways to make an S-box is using some decoders, encoders, and P-boxes.In Figure-5, an S-box which performed by these tools, is shown. In table 2, relationship between inputs and outputs of this S-box is shown.

In Structure of some S-boxes, like Figure-5, is used P-box, so the proposed method in section 2 can be used for formation these S-boxes.

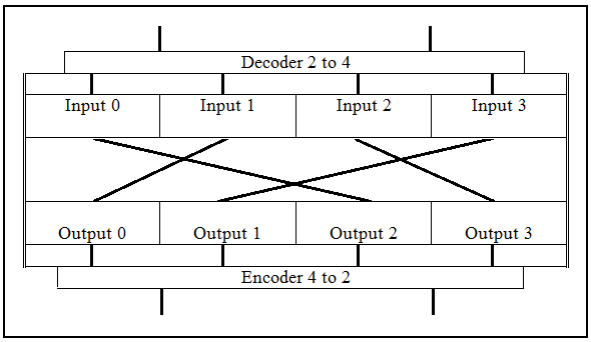

Figure-5. Showing the performance of an S-Box.

Table-2. Performance of S-Box in Figure-5.

Input of S-box	S-Box	Output of S-box
00		10
01		00
10		11
11		00

On the other hands, table of an S-Box is turning a data into another data. This changing should be the extremity of nonlinear. The S-box design criteria were eventually published after the public rediscovery of differential cryptanalysis, showing that they had been carefully tuned to increase resistance against this specific attack. In this paper, a method of making dynamic S-Box is introduced by modular calculations. In proposed algorithm, each of substitutions based on encryption key is calculated, then the table of these substitutions for all the input block which should be coded, are fixed.

The length of the Input block to create substitution is considered as n. Eight bits of data (or above), which should change to new amount, locates in any house of input block. For each house, a number from 0 to 126 is selected to add the content of this house. Next, the new content is divided by 128, and then the remainder

of division replaced the previous content. According to the algorithm, this action is repeated n times to calculate all of outputs. Therefore, the order of the input substitution is obtained. The S-Box key is derived from the key encryption method. Two variables of $C'$and D', which used in section 2, are used in this section again.

Algorithm implementation

1) Preparing the amounts of variable of the house.

1-1) In the implementation of this algorithm the user introduces n as the length of the input and two arbitrary numbers C', D' as the input of the function.

1-2) Consider number C equals C'+100 and D equals D'+10 (this ensures the least amount of C equals 100 and D equals 10)

2) The amount which should add to the content of kth house is calculated as follows:

$$\text{Additive amount}=[(C-k)^{(D+K)} mod\ 127] \quad (2)$$

Number 127 is a prime number, this number is employed to use modular computing features and reduce the volume of calculation.

3) Adding the additive amount to the content of kth house, next, put the remainder of this result by 128 instead of last amount.

The sample of implication is present in example 2.

Example 2: If length of the input block is 4 and parameters are (C'=90, D'=20), the process of substitution of input block by proposed method in this study presents below:

(C= C'+100=190, D= D'+10=30)

You can see the processing and result of example 2 in Table-3.

Table-3. Calculating the S-box of the example 2.

Initial Input	**a**	**b**	**c**	**d**
k	**1**	**2**	**3**	**4**
Additive amount	189^{31} mod 127	188^{32} mod 127	187^{33} mod 127	186^{34} mod 127
	81	76	117	22
Output	(a+81) mod 128	(b+76) mod 128	(c+117) mod 128	(d+22) mod 128

4. GENERATING ENCRYPTION KEY

As stated in section 2, to use algorithm of this study, the amounts of A', B', C', D' and $n_1$are needed. And the encryption key should be extracted for encryption and decryption. The users of this algorithm can consider an appropriate numerical range for these parameters. Also they can increase the length of the encryption key, to put the intended amounts of content comparably.

$$\text{New Key= Old Key + (Parameters of Method)} \quad (3)$$

This action will increase the length of the encryption key which does not make any practical problem. Some of the users do not want to increase the length of the encryption key while they are interested in substituting permutation algorithm instead of fixed tables of encryption permutation algorithm. The following way is suggested:

Concerning user's permutation algorithm table, the amount of n1 is fixed; therefore, it does not need to be located in the encryption key. In modern encryption methods, the length of the used keys is more than 100 bits. The first 100 bit is chosen as Key 1 (or each 100 bits that the users can be selected) and we consider the amounts of A', B', C' and D' according to the Figure-6.

0	1	…	28	29	30	31	…	58	59	60	61	…	88	89	90	91	…	99
A'					B'					C'					D'			

Figure-6. Calculation of permutation key parameters from current bits of encryption key.

If the users have t distinct permutation table in his algorithm, he needst defined permutation key. We considerkey number 1 as the main algorithm current 100 bits' key. For generating key number 2 we use proposed algorithm of this study and calculate number 1 key permutation for $n_1 = 100$andA' ‹B' ‹C' and D'. Then, it equals key number 2.

$$\text{Key 2 = PPA (Key 1, Key 1)} \quad (4)$$

Where, PPA(x,y) function is proposed permutation algorithm and x is input text and y is key of permutation. The above procedure should be observed to generate key i:

$$\text{Key (i)= PPA (Key (i-1), Key (i-1))} \quad (5)$$

In this section, the proposed lengths of A', B', C' and D'are arbitrary. And the user can employ the other

lengths concerning the length of the main algorithm key and the speed of employed hardware for encryption and decryption.

5. ADDING THE PROPOSED ALGORITHM TO THE CURRENT ENCRYPTION METHODS WITHOUT CHANGING THEM

As stated in previous sections, the producers of encryption algorithms without changing the length of encryption key can use proposed permutation algorithm instead of permutation tables.

Most current cryptographic systems have initial permutation in beginning and inverse initial permutation in end of algorithm; by fixed tables.This work does not make higher security in encryption process. For dynamic permutation tables, proposed algorithm can be used instead of initial and inverse initial permutation at least, and based on section 4; users can introduce its 100 first bits of current key for these permutation keys.

But if someone does not want to change these fixed tables, to increase the security of his encryption method, based on section 4, he can add an introductory permutation according to proposed algorithm before encryption operation and one reverse permutation at the end of the encryption operation as shown in Figure-7.

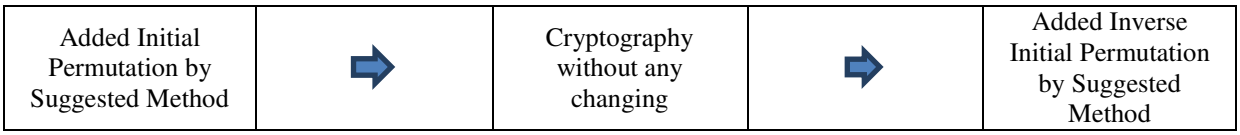

Figure-7. Adding proposed permutation method to current encryption method without changing the main body of previous algorithm.

6. CONCLUSIONS

P-Box and S-Box are essential components of symmetric and modern encryption systems but they are not effective in isolation. Each symmetric encryption system should have equipped with two features diffusion and confusion at the highest level. In most encryption methods permutation and substitution tables are fixed. In this paper, two methods, without changing the length of current encryption key, were introduced which can turn tables from static form to the dynamic form and dependent on the encryption key. This proposed method can increase the level of security which is the ultimate aim of these algorithms.

Discussing about attacks

Linear cryptanalysis and differential cryptanalysis are two known and classic ways of attacking symmetric encryption systems.

The linear cryptanalysis

With the assumption of availability of a large size of encrypted blocks and the equivalent unencrypted blocks, the linear cryptanalysis is achieved. Code cracking tries to find a linear relation between the bits of the input text, the encrypted content and the main key. This method which was introduced by Matsui[12], is based on linear approximation of P-box and S-Box tables. For linear approximation the analyzer (code cracking) should begin with the system input and identify every path the inputs effect on before the last loop and do the approximate linearization of P-Box and S-Box tables. Following with the use of the encrypted and unencrypted text blocks, approaches the approximation to reality and discovers the whole key or part of the key. In the suggested algorithm, with correlation between P-Box tables and the key, it is impossible to do the approximation and linear attack in this method would be negated.

The differential cryptanalysis[13]

This method is based on how the differences between two blocks of the plain text (even as tiny as a bit) effect on the cipher text and how different the encrypted results of these two blocks are. In specific situations this relation can reveal some bits of the key.

In this method all the possible changes are exerted to the inputs and the changes to the output are recorded. Statistically analyzing the results and forming the differential distribution table, the effective parameters for attacking the system are extracted and then using enough encrypted texts and the equivalent unencrypted texts the true key is calculated.

In the suggested algorithms, P-Box and S-box are function of the key. These dependencies cause the differential distribution table and the effective parameters not to correctly form to attack. So the differential cryptanalysis would be negate too.

Applications

Recently, P-Box and S-Box have new usages. The utilization of P-Box and S-box become popular in image ciphers as a main approach to performing permutation and substitution (Image cryptosystem)[14-16].

Image steganography is one of the methods that used for protection of the images. In image steganography, a secret image is embedded into a host image in such a way that the unauthorized users cannot recognize the presence of the embedded image in resultant image. Permutation is used also in steganography [17-21].

REFERENCES

[1] National Bureau of Standards: Data Encryption Standard, US. Department of Commerce, FIPS pub.46. January 1977.

[2] J. Daemen and V. Rijmen, T. D.o.R. 2002. AES - The Advanced Encryption Standard. Springer-Verlag.

[3] Shannon C. 1990. Communication theory of secrecy system. Bell Syst. Tech. J. 28: 656-715.

[4] A. H. Alkhaldi, I. H., M. A. Gondal. 2015. A novel design for the construction of safe S-boxes based on TDERC sequence. Alexandria Engineering Journal. 54: 65-69.

[5] Das S. 2014. Generation of AES-like 8-bit Random S-box and Comparative Study on Randomness of Corresponding Ciphertexts with Other 8-bit AES-S-boxes. Intelligent Computing, networking and Informatics Advances in Intelligent Systems and Computing, Springer. 243: 303-318.

[6] J. Cui, L.H., H. Zhong, C. Chang and W. Yang. 2011. An ImprovedAES s-box and its Performance analysis. International Journal ofInnovative Computing, Information and Control. 7(5): 2291-2302.

[7] P. Brändén, A.C. 2011. Mesh patterns and the expansion of permutation statistics as sums of permutation patterns. Electron. J. Combin. 18(2): 5-14.

[8] S. Avgustinovich, S.K., A. Valyuzhenich. 2013. Avoidance of boxed mesh patterns on permutations. Discrete Applied Mathematics. 161: 43-51.

[9] S. Kitaev J.R. 2015. (a, b)-rectangle patterns in permutations and words. Discrete Applied Mathematics. 186: 128-146.

[10] Laigle-Chapuy, Y., Permutation polynomials and applications to coding theory. Finite Fields Appl, 2007. 13: p. 58-70.

[11] G. Khachatrian M. K. 2015. Permutation polynomials and a new public-key encryption. Discrete Applied Mathematics.

[12] M.Matsui. 1994. Linear Cryptanalysis Method for DES Cipher advances in Cryptology EUROCRYPT. Springer-Verlag.

[13] Biham S. 1990. Differential Cryptanalysis of DES-like Cryptosystems. Technical Report CS90-16 (Weizmann Institute of Science).

[14] I. Hussain, T. S., M. Gondal. 2012. Image encryption algorithm based on PGL(2,GF (28)) S-boxes and TD-ERCSchaotic sequence. Nonlinear Dyn.

[15] Ye G.D., Wong K.W. 2012. An efficient chaotic image encryption algorithm based on a generalized Arnold map. Nonlinear Dyn. 64(4): 2079-2087.

[16] Xiao Y.Z.D. 2013. Cryptanalysis of S-box-only chaotic image ciphers against chosen plaintext attack. Nonlinear Dyn. 72: 752-756.

[17] I.Hussain N.A.A., T. Shah. 2014. Stego optical encryption based on chaotic S-box transformation. Optics & Laser Technology. 61: 50-56.

[18] Liu Z, L.Q., Dai J, Sun X, Ahmad MA. 2009. A new kind of double image encryption by using a cutting spectrum in the 1-D fractional Fourier transform domains. Opt. Express. 282: 1536-1540.

[19] Chen W, S. G., Chen X. 2013. High-flexibility optical encryption via aperture movement. Opt. Express. 21: 680-691.

[20] Rudolf M., StanićLoknar N., ŽiljakStanimirović I. 2015. Infrared steganography with individual screening shapes applied to postage stamps with security features. Technical Gazette. 22(4): 939-945.

[21] V. Vaithiyanathan, B. K., Anishin Raj M. M., M. Rajasekhar Reddy, Priyanka S. and K. Abinaya. 2015. AN Amalgamated Approach of Cryptography and Steganography Using IWT and Random Pixel Selection for Secure Transmision. ARPN Journal of Engineering and Applied Sciences. 10(6): 2352-2357.