CONSTRUCT MATHEMATICAL S-BOX FOR SYMMETRIC KEY CRYPTOSYSTEMS

[1]ALI H. KASHMAR, [2]TAREEF KAMIL MUSTAFA, [3]UHOOD AL-HASSANI

[1, 2, 3] University of Baghdad, College of Science, Baghdad, Iraq
E-mail: tareef_alshaibi@yahoo.com

Abstract - Advanced Encryption Standard (AES) is a block cipher system which widely used in cryptographic application. Substitution box (S-box), nonlinear substitution operation, is the main factor of the AESstrength.The reason of construction S-boxes for symmetric key cryptosystems is to yield good properties such as high nonlinearity andlow autocorrelation. The main objective of this research is suggesting mathematical approach to generate random S-boxessuitable for symmetric key cryptosystems. The fact that S-boxes are randomly key-dependent and unknown is the main strength of the new approach, since both linear and differential cryptanalysis requires known S-boxes. In this paper, briefly analyze of the AES algorithm, substation-boxes, linear and differential cryptanalysis are introduced. Moreover, a randomly key-dependent S-box, inverse S-box generation algorithm andthe significant properties of proposed S-boxes are described. Preliminary results show that our proposed algorithm has good cryptographic strength, fast, low complexity, and resistant against linear and differential cryptanalysis.

Keywords - Cryptography, Symmetric key,AES and RijndaelS-box.Cryptanalysis.

I. INTRODUCTION

Cryptography has an important role in the security of the data transmission and is the best method of data protection against passive and active fraud. The growing number communication users haveled to increasing demand for security measures to protect data transmitted over open channels. Acipher system is a set of reversible transformations from the set M of a plaintext into the C of a ciphertext. Eachtransformation depends on a secret key and the ciphering algorithm, carried out for the whole block. Two general principles of block ciphers are diffusion and confusion. Confusion is designed to obscure the relationship between the plaintext and ciphertext, while diffusion is supposed to spread the plaintext statistics through the ciphertext[1]. In most cipher systems the diffusion and confusion is achieved by means of round repetition. Repeating a single round contributes to cipher's simplicity. Modern block ciphers consist of four transformations: substitution, permutation, mixing and key adding[2]. The AES defined by NIST of the US has been accepted to replace DES as the new private key encryption. AES overpass DES in improved security because of lager key sizes. AES is suitable for 8 bit microprocessor platforms and 32 bit processors [3].It depends on the S-boxes, which are fixed and have no relation with the secret key. So only changeable parameter is the secret key. Since the only nonlinear component of AES is S-boxes, they are an important source of cryptographic strength. The design of S-boxes is an important problem for security becausethey form the only nonlinear operation in an encryption process and each successful linearization approximation can help break a few bits of the key. The use of key-dependent S-boxes in block cipher design has not been widely investigated in the literature. Research into S-box design has focused on determination of S-box properties that yield cryptographically strong ciphers, with the aim of selecting a small number of good S-boxes for use in a block cipher DES and CAST [2]. Some results have demonstrated that a randomly chosen S-box of sufficient size will have several of these desirable properties with high probability [4]. This paper outlines the work of the author's investigation into the design of new key dependent S-boxes. Other systems using key dependent S-boxes have been proposed in the past, the most well-known is Blowfish [5] and Khufu [6]. Each of these two systems uses the cryptosystem itself to generate the S-boxes. Preliminary results show that our proposed algorithm has good cryptographic strength with the added benefit that is resistant against linear and differential cryptanalysis, which require that the S-boxes be known. A new method to generate key S-boxes as a function of the secret key will be presented. In the next section, the AESalgorithm with Rijndale S-boxwas briefly introduced. A central part of the paper describes the construction of new S-boxand its inverse. In the following two sections, the paper gives thesignificant properties of proposed S-boxas well as results and discussions.Finally, the paper gives some conclusions.

II. THE AES ALGORITHM

Since Rijndael algorithm was chosen by NIST as an advanced encryption standard (AES) in 2000,[7] much attention has been attracted to it and many methods have been proposed to attack it [8],[9],and [10]. However, there has never been successful attack on the full AES up to now. As the only nonlinear operation of AES, S-box plays a crucial role against

various attacks. S-boxes are the most important and only nonlinear component of a block cipher since diffusion and confusion properties, which are related with the security of cryptographic algorithms, are added to a block cipher by S-boxes. So, bijective S-boxes play an important role in the design of symmetric ciphers. To date, the techniques for the construction of the S-boxes have included pseudorandom generation, finite field inversion, power mapping and heuristic techniques. From these techniques, the use of finite field operation in the construction of an S-box yields linear approximation and difference distribution tables in which the entries are close to uniform. Therefore, this provides security against differential and linear attacks. Some of the desirable cryptographic properties are completeness, strictavalanchecriterion, and bit independence [11]. An $n \times n$ S-box, $S(x): GF(2^n) \rightarrow GF(2^n)$, maps an n-bit input to an n-bit output and can be viewed as consisting of n Boolean functions. This type of S-boxes, one of which is the AES S-box and maps an 8-bit input to an 8-bit output, has been used in most ciphers in the literature. The S-box is a simple table lookup with integer values defined by a 16×16matrix, which contains all possible 256 bytes. Each individual byte is mapped into a new byte in the following way: the left most 4-bits are used as a row value and the right most 4-bits are used as column value. This row and column values serve as indices (0 to 15) into the S-boxes to select a unique 8-bit output value. It is to be noted that the entries are given in decimal forms. These S-boxes also facilitate the cipher not go into spurious (all zero) state after key initialization.Rijndeal S-boxis invertible, and can be constructed by composing two transformations, the algorithm of S-box shown in the following steps[12],[13]:

1. Inputs look up table S-box (16×16)bytes.
2. Compute the multiplicative inverse for each byte in the finite field GF(2^8); the multiplicative inverse for byte{00} is itself.
3. The output produce by applying the following affine transformation over GF(2):

$$b_i' = b_i \oplus b_{(i+4) \bmod 8} \oplus b_{(i+5) \bmod 8} \oplus b_{(i+6) \bmod 8} \oplus b_{(i+7) \bmod 8} \oplus c_i \quad (2.1)$$

Where b_i ,$0 \leq i < 8$, is the i^{th}bit of a byte b, and $c_i$is the i^{th}bit of a byte cwith the value {63} or {01100011}. A prime on a variable (e.g., b') gives an indication that the variable, everywhere, requires updating with the value on the right.

The affine transformation byte of the S-box can be represented in matrix form as:

$$\begin{bmatrix} b_0' \\ b_1' \\ b_2' \\ b_3' \\ b_4' \\ b_5' \\ b_6' \\ b_7' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \quad (2.2)$$

The Rijndeal S-boxis shown in Table (2.1)

Table (2.1)Rijndeal S-box

		Y															
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
	0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
	1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
	2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
	3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
	4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
	5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
	6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
X	**7**	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
	8	CD	0C	13	EC	5F	97	44	17	CA	A7	7E	3D	64	5D	19	73
	9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	08	D8
	A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	EA	79
	B	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
	C	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
	D	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
	E	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
	F	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

III. THE NEW S-BOX CONSTRUCTION

The new S-box [.] was derived from the Rijndale S-box in a key-dependent fashion. The S-box that is deploying a small quantity of memory was implemented in the design of a new algorithm to have limitations on the quantity of ROM available. Hence this algorithm used S-box containing only 256 entries. This S-box is a simple lookup table that contained a permutation of all possible 256 eight-values. The proposed algorithm design introduced private key S-box used as an appropriate shuffler for bytes. The S-box [.] was derived from the Rijndale S-box in a key-dependent fashion. Because S-box is secretive, therefore, the opponent no prior information about the input to the S-box. Four bytes(32-bit)secret key was employed to update S-box process. The following equation was used to initialize the private S-box:

$$S[i] = Rij_{(S-box)}[\cdots Rij_{(S-box)}[Rij_{(S-box)}[i \oplus k_0] \oplus k_1] \oplus \ldots \oplus k3, i=0,\cdots,255. \quad (3.1)$$

where $Rij_{(S-box)}$ is refers to the fixed Rijndael S-box, which is represented one-dimensional as demonstrated in Table 2.1. A strong diffusion can be produced from the selection of the S-box because there is a dependency between the output bit and the input bit of the S-box. The pseudo code for this transformation in the encryption process was explained as

```
SubBytes (byte state [4, 4])
        begin
  for ( int r = 0; r < 4; r++)
  for ( int c = 0; c < 4; c++)
state [ r, c ] = Sbox [state [ r , c ] ]
        end
        end
        end
```

Inverse S-box was utilized for decipherment. Similar alteration was applied but with the use of inverse substitution S-box. The inverse S-box is represented as one-dimensional of Table 2.1 which was found as the forward S-box, explained as

```
for i = 0 to 256
invSbox [ Sbox [ i ] ] = i
```

The algorithm ofconstructed a new S-box shown in the following steps:

1. Input Rijndeal S-box (16×16)bytes and key S-box (4-bytes).
2. New S-box $= f(Rij_{(S-box)}, key_{(S-box)})$
3. $S(i) = Rij_{(S-box)}[\ldots Rij_{(S-box)}\{Rij_{(S-box)}(i \oplus k_0) \oplus k_1\} \ldots \oplus k_3]$, for $i = 0, 1, 2, \ldots, 255$
4. The output a new S-box (16×16).

Example

In the following, give an example to construct the new S-box. Suppose we have

Key S-box $(k_0 k_1 k_2 k_3)$ = 61 43 64 46,(hexadecimal) where k_0 = 61 (0110 0001), k_1 = 43(0100 0011),k_2 = 64(0110 0100), and k_3 = 46(0100 0110)

In order to construct the elements of a new S-box. Applied equation (3.1)

To find the first element $S(00)$, where i = 00 (0000 0000),

$Rij_{(S-box)}(00 \oplus k_0) =$
$Rij_{(S-box)}(0000\ 0000 \oplus 0110\ 0001) =$
$Rij_{(S-box)}(0110\ 0001) = Rij_{(S-box)}(61) = EF,$
$Rij_{(S-box)}(EF \oplus k_1) =$
$Rij_{(S-box)}(1110\ 1111 \oplus 0100\ 0011) =$
$Rij_{(S-box)}(1010\ 1100) = Rij_{(S-box)}(AC) = 91,$
$Rij_{(S-box)}(91 \oplus k_2)$
$= Rij_{(S-box)}(1001\ 0001 \oplus 0110\ 0100)$
$= Rij_{(S-box)}(1111\ 0101) = Rij_{(S-box)}(F5) = E6,$
$Rij_{(S-box)}(E6 \oplus k_3) =$
$Rij_{(S-box)}(1110\ 0110 \oplus 0100\ 0110) =$
$Rij_{(S-box)}(1010\ 0000) = Rij_{(S-box)}(A0) = E0,$

Thus the first element of the new S-box represents$E0$

		Y														
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0	E0															
	1																
	2																
	3																
	4																
	5																
	6																
X	7																
	8																
	9																
	A																
	B																
	C																
	D																
	E																
	F																

To find the second element$S(01)$,

where i = 01 (0000 0001)

$Rij_{(S-box)}(01 \oplus k_0) = Rij_{(S-box)}(0000\ 0001 \oplus 0110\ 0001) = Rij_{(S-box)}(0110\ 0000) = Rij_{(S-box)}(60) = D0$,

$Rij_{(S-box)}(D0 \oplus k_1) = Rij_{(S-box)}(1101\ 0000 \oplus 0100\ 0011) = Rij_{(S-box)}(1001\ 0011) = Rij_{(S-box)}(93) = DC$,

$Rij_{(S-box)}(DC \oplus k_2) = Rij_{(S-box)}(1101\ 1100 \oplus 0110\ 0100) = Rij_{(S-box)}(1011\ 1000) = Rij_{(S-box)}(B8) = 6C$,

$Rij_{(S-box)}(6C \oplus k_3) = Rij_{(S-box)}(0110\ 1100 \oplus 0100\ 0110) = Rij_{(S-box)}(0010\ 1010) = Rij_{(S-box)}(2A) = E5$, thus, the second element of the new S-box represents E5

		Y														
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0	E0	E5														
	1																
	2																
	3																
	4																
	5																
	6																
X	7																
	8																
	9																
	A																
	B																
	C																
	D																
	E																
	F																

And so on, by the same wayconstruct all elements of a new S-box, as shown in Table (3.1)

Table 3.1the new S-box key in hexadecimal

		Y														
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0	E0	E5	38	DE	4B	EC	05	70	F7	3C	96	B5	30	4F	AF	1D
	1	3E	66	ED	44	BC	41	0A	34	FC	6F	BD	07	80	F8	Db	Ac
	2	39	A3	75	D7	26	55	B9	32	Cc	E8	C1	F9	D4	9D	10	58
	3	A9	3A	3D	D5	47	5E	14	7d	24	B2	9B	2C	73	2B	3F	Cb
	4	23	16	11	65	48	9E	4E	6a	78	C6	62	84	F0	8E	6E	51
	5	49	Be	2A	1F	2E	67	88	Bf	Dc	22	68	60	7E	Aa	D3	E1
	6	1C	B7	7A	B1	59	0E	A1	C0	Fa	79	Ea	5a	9C	18	8A	53
X	7	57	45	E7	69	89	4d	A0	F6	Ca	A8	87	46	74	7F	13	29
	8	Cf	72	6B	17	Ba	D6	1B	8F	40	D8	E4	2d	D0	20	37	1E
	9	42	9F	E6	64	Ce	Ab	50	E9	97	E2	06	3b	43	08	63	B6
	A	77	C9	71	A4	Eb	36	B3	6C	28	A5	F5	54	61	33	Ca	93
	B	00	A2	D2	C8	Da	25	D1	19	Fe	31	Ff	E3	6d	Fb	09	95
	C	Cd	35	A7	0D	Fd	27	76	Ae	8D	7B	C2	91	01	Bc	15	83
	D	5D	0B	F4	F2	99	Ee	5B	F1	Ef	D9	Df	0F	F3	B8	52	0C
	E	5C	56	98	86	9A	C5	4C	92	81	Dd	8B	4a	7C	5F	90	82
	F	A6	02	2F	Bb	Ad	21	85	C7	94	04	C3	12	B0	B4	03	1A

The inverse of a new S-box

Inverse S-box was utilized for decipherment. Similar alteration was applied but with the use of inverse substitution S-box. The inverse S-box is represented as one-dimensional of Table 3.1 which was found as the forward S-box, explained as

$$\text{for } i = 0 \text{ to } 256$$
$$invS-box[S-box[i]] = i$$

Table 3.2the inverse of a new S-box key in hexadecimal

		Y														
		0	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**
	0	B0	Ce	F1	Fe	F9	06	9A	1B	9D	Be	16	D1	Df	C3	65	Db
	1	2E	42	FB	7E	36	Ce	41	83	6D	B7	Ff	86	60	0F	8F	53
	2	8D	F5	59	40	38	B5	24	C5	A8	7F	52	3D	3B	8B	54	F2
	3	0C	B9	27	Ad	17	C1	A5	8E	02	20	31	9B	09	32	10	3E
	4	88	15	90	9C	13	71	7B	34	44	50	Eb	04	E6	75	46	0D
	5	96	4F	De	6F	Ab	25	E1	70	2F	64	6B	D6	E0	D0	35	Ed
	6	5B	Ae	4A	9E	93	43	11	55	5A	73	47	82	A7	Bc	4E	19
X	**7**	07	A2	81	3C	7C	22	C6	A0	48	69	62	C9	Ec	37	5C	7D
	8	1C	E8	Ef	Cf	4B	F6	E3	7A	56	74	6E	Ea	14	C8	4D	87
	9	Ee	Cb	E7	Af	F8	Bf	0A	98	E2	D4	E4	3A	6C	2D	45	91
	A	76	66	B1	21	A3	A9	F0	C2	79	30	5D	95	1F	F4	C7	0e
	B	Fe	63	39	A6	Fd	0b	9F	61	Dd	26	84	F3	Cd	1A	51	57
	C	67	2A	Ca	Fa	78	E5	49	F7	B3	A1	Ae	3F	2B	C0	94	80
	D	8C	B6	B2	5E	2C	33	85	23	89	D9	B4	1E	58	E9	03	Da
	E	00	5F	99	Bb	8A	01	92	72	29	97	6A	A4	05	12	D5	D8
	F	4C	D7	D3	Dc	D2	Aa	77	08	1D	2B	68	Bd	18	C4	B8	Ba

IV. SIGNIFICANT PROPERTIES OF PROPOSED S-BOX

1. The utilization of S-box dependent on secret key ensured great difficulties relative to attackers. Because the S-box was unrecognizable for the attacker, the number of probabilities faces the attacker would increase.
2. A strong diffusion can be produced from the selection of the S-box because there is a dependency between the output bit and the input bit of the S-box.
3. Unique S-box (weak collision resistant): no two different keys have the same S-box elements. In the mathematical step, the function makes a unique value for each input block.This means that the new S-box resistant against differential cryptanalysis, which require that the S-boxes be known.
4. Efficiency: calculating S-box table is easy to used and simple implementation according to comparing results of different algorithms.
5. Strict Avalanche Criterion (SAC): when change one bit/byte of input, the S-box value should be change more than half bits/bytes [14]. In mathematical step of S-box, the previous byte of new block will be changing all the next bytes.
6. Pseudo randomness: in the proposed S-box, the keys is difficult to cryptanalysis attacks (randomness) according to:
a) Key values generated from strong PRNG and used only once (as one time pad).
b) S-box table has the same frequency for each key.
7. Key size: the size of the key block has a great effect on the algorithm, the key size should be minimal size with maximum period, the block size of the propose algorithm is 128-bits.

V. RESULTS AND DISCUSSIONS

There are three main aspects of results to introduce the efficiency and robust of the propose S-box. These aspects are fast, complexity and secure.
1. Fast: the execution time of proposed S-box is a criterion to check the S-box efficiency. The execution time computed by calculating the arrange time of multi execution time for the similar S-boxes. The execution time is based on the complexity of algorithm and size of dataset. Table (5.1) shows the compression between the execution time for a propose S-box and some identical algorithm.

Table (5.1) the execution time for construction S-box [15]

Algorithm	Mean execution time (MS)	Standard Deviation (MS)
SHA-128	15.1	11.27
DES	10.3	0.64
AES	25.8	32.77
3DES	20.3	10.75
propose	8.9	1.3

Comparing the propose S-box based on fastens aspect shows without any doubt that, the execution time results for the propose S-box is faster than identical algorithms.

2. Less Complexity: the less complexityof the propose S-box come from:

a) Generate S-box based on secure key

b) Substitution step of the propose S-box is based on mathematical function.

3. Secure: the security result is one of the important matric that should be providing in the design of any S-box.There are many parameters used to demonstrate the security of the propose S-box:

a) Construction S-box based on secure key. The S-box table is comparing with security parameters such as balanced output, SAC and hamming distance.

b) Generate secure and robust sub-key for S-box depend on a strong and efficient PRNG, moreover, the output of PRNG pass statistical tests for randomness.

c) As a results, the proposed S-box resistant against some security attacks such as frequency analysis, known plaintext attack, linear, differential and statistical attacks.

CONCLUSION

A new S-box depends on mathematical approach was suggested to increase resistantagainst security analysis such as linear and differential attack,where new S-box was unknown, does not depended on exclusive-or operation and multiplicative inverse. Unlike static original S-box, it is a good idea to design a new S-box and its inverse based on mathematical approach because it dependent on secret key in order to add additional random and dynamic properties to S-box. The proposed S-box design using complexity of its algebraic expression in GF(2^8) as well as it does not contain linear combinations of input bits and linear combination of output bits that means resistant against linear attack.The results show that the propose S-box was efficiency, less complex and more secure than identical algorithms.

REFERENCES

[1] Shannon, C. E. 1949. Communication Theory of Secrecy Systems. Bell Technical Journal, 28(4): 656-715.

[2] Menezes, A. J., van Oorschot P., & Vanstone, S. 2006. Handbook of Applied Cryptography. Florida: CRC Press.

[3] Su, C. P., Lin, T. F., Huang, C. T. & Wu, C. W. 2003. “A High-throughput Low-cost AES Processor”. IEEE Communications Magazine 41(12): 86–91.

[4] Keliher, L. 2003. “ Linear Cryptanalysis of Substitution-Permutation Network” PhD Thesis, Queen’s University Kingston, Ontario, Canada.

[5] Schneier, B. “Applied Cryptography”, John Wiley & Sons, Inc. 1996.

[6] Merkle, R.C. 1990. “Fast Software Encryption Functions. Advances in Cryptology—CRYPTO '90. Santa Barbara, California: Springer-Verlag. pp. 476–501”.

[7] Advanced Encryption Standard (AES). Federal Information Processing Standards Publication 197, 2001.

[8] Biryukov, A; Dunkelman, O; Keller, N; Khovratovich, D& Shamir, A. 2009. "Key Recovery Attacks of Practical Complexity on AES Variants With Up To 10 Rounds".

[9] Bogdanov, A; Khovratovich, D& Rechberger, C. 2011. "Biclique Cryptanalysis of the Full AES" .

[10] Biham, E., Dunkelman, O. & Keller, N. 2006. “Related Key Impossible Differential Attacks on 8-round AES-192”. Lecture Notes in Computer Science (3860): 21-33.

[11] Balaee Maram K. and Gnansekar .M, 2016.“ Evalution of Key Dependent S-box based Data Security Algorithm using Hamming Distance and Balanced output”.

[12] Daemen, J. & Rijmen V. 2000. AES Proposed: Rijndael. Available at http://www.east.kulenven,ae.be/~rijmen/rigndael.

[13] Daemen, J. & Rijmen, V. 2002. The Design of Rijndael: AES the Advanced Encryption Standard. Berlin: Springer.

[14] Balaee Maram, K. and Gnanasekar,M. 2008 “ Efficient and simple Method for Designing Chaotic S-box”, ETRI Journal, vol, 30,10-172, 2008.

[15] Stallings, W. 2003. Cryptography and Network Security: Principal and Practice. New Jersey: Prentice-Hall.

★★★