Paper: ASAT-13-CE-24

13th International Conference on
AEROSPACE SCIENCES & AVIATION TECHNOLOGY,
ASAT- 13, May 26 – 28, 2009, E-Mail: asat@mtc.edu.eg
Military Technical College, Kobry Elkobbah, Cairo, Egypt
Tel : +(202) 24025292 – 24036138, Fax: +(202) 22621908

Generation of AES Key Dependent S-Boxes using RC4 Algorithm

I. Abd-ElGhafar*, A. Rohiem*, A. Diaa*, F. Mohammed**

Abstract: The increase use of computer and communication system by industry and organizations has increased the risk of theft of proprietary information. Although these threats require a variety of countermeasure, encryption process is a primary method of protecting valuable electronic information. The encryption process also needs to be dynamic in order to face new technique and more advance methods used by cryptanalysis. Substitution box (S-box) is keystone of modern symmetric cryptosystem .They bring nonlinearity to cryptosystem and strengthen their cryptographic security. In this paper RC4 algorithm which is well known stream cipher is used to generate S-box for advance encryption standard (AES). The generated S-boxes are more dynamic and key dependant which will increase the complexity and also make the differential and linear cryptanalysis (DC&LC) more difficult. Various randomness tests are applied to the customized AES (AES-RC4) algorithm and the results shown that the new design pass all tests which proven its security.

Keywords: Advanced Encryption Algorithm (AES), Cryptosystem, Advance encryption standard, RC4, Differential cryptanalysis (DC), Linear cryptanalysis (LC).

1. Introduction

Rijndael is a block cipher developed by Joan Daemen and Vincent Rijmen. The algorithm is flexible in that any combination of data and key size of 128, 192, and 256 bits are supported. However, AES only allows the data length to be 128 bits while conserving the property of supporting three different key lengths. AES can be divided into four basic operation blocks which operates on array of bytes, organized as a 4×4 matrix called the state. Four basic steps, called layers consist of the ByteSub Transformation, the ShiftRow Transformation, the Mix-Column Transformation, and AddRoundKey .

i- **The ByteSub Transformation:** Non-linear byte substitution which is composed of multiplicative inverse and affine transformation.
ii- **The ShiftRow Transformation:** Linear diffusion process, operating on individual rows. Depending on the row location, offset of left shift varies from zero to three bytes.
iii- **The MixColumn Transformation:** Matrix multiplication over $GF(2^8)$ .Column vector is multiplied with a fixed matrix where the bytes are treated as a polynomials rather than numbers.
iv- **AddRoundKey:** Simple byte XOR operation with the round key.

* Egyptian Armed Forces
** PGS Student, Sudanese Armed Forces

These four layer steps describe one round of AES. A 128 bit round key, used in AddRoundKey operation, is generated by the key schedule. Sub-keys are derived from the original user key by XOR operation of two previous columns. For columns that are in multiples of four, the process involves additional round constants, S-box, and shift operation. Excluding the first and the last round, AES encryption round executes nine iterations. First round of the encryption step performs XOR with the original key and the last round skips MixColumn layer.

All four layers described above have corresponding inverse operations such that the decryption is simply the reverse order operations of these inverse transformations .Note that the constant matrix for the MixColumn multiplication used in the decryption process consist of higher values.

Rijindael is considered to be the fastest algorithms in terms of the critical path between the plaintext and the ciphertext, due to this and other security features AES is selected by 803.11i group to replace WEP in wireless networks. In this paper we introduce a new method for S-box generation .This method is increase the complexity of S-box generation, thus increase the diffusion and also make the differential and linear cryptanalysis (DC) and (LC) attacks more difficult. The rest of paper is organized as follow. Section 2 we introduce the method of construction of S-box with AES- like algorithm. Section 3 new S-box structures is proposed. We analyzed the cryptographic characteristics of AES-RC4 S-box and compared it with AES S-box in section 4. Section 5 is conclusion.

2. The ByteSub Transformation layer

This layer uses S-box to perform the byte substitute operation. AES defines a 16x16 matrix of byte values, called an S-box as given in Table (1) that contains a permutation of all possible 256 8-bit values. Each individual byte of state is mapped into a new byte in the following way: The leftmost 4 bits are used as a row value and the rightmost 4 bits are used as a column value. These row and column values serve as indexes into the S-box to select a unique 8-bit output value. For example, the hexadecimal value {95} references row 9, column 5 of the S-box, which contains the value {2a}:

The S-box is constructed in the following fashion:

(1) Initialize the S-box with the byte values in ascending order row by row. Thus, the value of the byte at row x, column y is {xy}
(2) Map each byte in the S-box to its multiplicative inverse in the finite field GF (2^8) the value {00} is mapped to itself.
(3) Consider that each byte in the S-box consists of 8 bits labeled (b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0). Apply the affine transformation to each bit of each byte in the S-box:

$$b_i' = b_i \oplus b_{(i+4) \bmod 8} \oplus b_{(i+5) \bmod 8} \oplus b_{(i+6) \bmod 8} \oplus b_{(i+7) \bmod 8} \oplus c_i \quad (1)$$

where c_i is the i-th bit of byte c with the value {63}, that is, ($c_7c_6c_5c_4c_3c_2c_1c_0$) = (01100011). The prime $(')$ indicates that the variable is to be updated by the value on the right.

The AES standard depicts this transformation in matrix form as follows:

$$\begin{pmatrix} b_0' \\ b'_1 \\ b'_2 \\ b'_3 \\ b'_4 \\ b'_5 \\ b'_6 \\ b'_7 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} \quad (2)$$

Each element in the product matrix is the bitwise XOR of elements of one row and one column. Further, the final addition, shown in equation (2), is a bitwise XOR, the inverse S-box is obtained by taking the inverse of equation (2), affine transformation followed by taking the multiplicative inverse in GF (2^8) given in Table (2). As an example, consider the input value {95}. The multiplicative inverse in GF (2^8) is $\{95\}^{-1}$ = {8a}, which is 10001010 in binary. Using equation (2), the result is {2A}

Table 1. S-box

	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	3B	52	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
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	C4	A7	7E	3D	64	5D	19	73
9	60	81	4F	DC	22	**2A**	90	88	46	EE	B8	14	DE	5E	0B	DB
A	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	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

3. Generation of AES Key Dependent S-Boxes using RC4 Algorithm

The new s-box design using well know RC4 stream cipher algorithm, RC4 is designed in 1987 by Ron Rivest, RC4 is variable key size stream cipher with byte oriented operation. The algorithm is based on the use of a random permutation of 256 bit state. It used in WEP and SSL/TLS (secure socket layer/transport layer security). The key length is variable from 1 to 256 byte and used to initialize a-256 state vectors. All times the state contains a permutation of all 8-bits numbers from 0 to 255. In this design we use the key schedule algorithm to produce that permutation to generate our S-box. The AES-RC4 S-box is constructed as the following steps:

Table 2. Inverse S-box

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	52	09	6A	D5	30	36	A5	38	BF	40	A3	9E	81	F3	D7	FB
1	7C	E3	39	82	9B	2F	FF	87	34	8E	43	44	C4	DE	E9	CB
2	54	7B	94	32	A6	C2	23	3D	EE	4C	**95**	0B	42	FA	C3	4E
3	08	2E	A1	66	28	D9	24	B2	76	5B	A2	49	6D	8B	D1	25
4	72	F8	F6	64	86	68	98	16	D4	A4	5C	CC	5D	65	B6	92
5	6C	70	48	50	FD	ED	B9	DA	5E	15	46	57	A7	8D	9D	84
6	90	D8	AB	00	8C	BC	D3	0A	F7	E4	58	05	B8	B3	45	06
7	D0	2C	1E	8F	CA	3F	0F	02	C1	AF	BD	03	01	13	8A	6B
8	3A	91	11	41	4F	67	DC	EA	97	F2	CF	CE	F0	B4	E6	73
9	96	AC	74	22	E7	AD	35	85	E2	F9	37	E8	1C	75	DF	6E
A	47	F1	1A	71	1D	29	C5	89	6F	B7	62	0E	AA	18	BE	1B
B	FC	56	3E	4B	C6	D2	79	20	9A	DB	C0	FE	78	CD	5A	F4
C	1F	DD	A8	33	88	07	C7	31	B1	12	10	59	27	80	EC	5F
D	60	51	7F	A9	19	B5	4A	0D	2D	E5	7A	9F	93	C9	9C	EF
E	A0	E0	3B	4D	AE	2A	F5	B0	C8	EB	BB	3C	83	53	99	61
F	17	2B	04	7E	BA	77	D6	26	E1	69	14	63	55	21	0C	7D

1. First initialize as follow after given the input key

*/*initialization*/*
For i=0 to 255
S[i]=I;
T[i]=k[i mod keylenght];
J=0;
For i=0 to 255
J=(j+s[i]+t[i])mod 256;
Swap (s[i],s[j])

2. The output of step one gives us 256 different values, all these values depend on the input key. This means that if we change one byte value from the input key we get another different 256 values .This feature help us to construct 256! S-boxes depend on input key.

3. Take affine transformation for the produced values, the affine transformation is used here, as apply in original S-box, to avoid any fixed points and to make the new S-box invertible. Tables (3) and (4) give an example of AES-RC4 S-box and its inverse when key equal to (0123456789ABCDEF) is applied.

4. Security Analysis

In order to measure the degree of security of AES-RC4 S-box, some cryptographic tests must be applied such as randomness test, avalanche criteria and bit independence criteria (BIC) test [1]. In this section we analyze AES-RC4 S-box using these tests.

4.1 Avalanche test

The avalanche effect property is very important for encryption algorithm. This property can be seen when changing one bit in plaintext and then watching the change in the outcome of at least half of the bits in the ciphertext.One purpose for the avalanche effect is that by changing only one bit there is large change then it is harder to perform an analysis of ciphertext, when trying to come up with an attack. First we start calculate avalanche effect for AES S-box. To perform the test we change plaintext bit to"01"instead of"00"and "11"instead of "01"the result obtained is 0.4688 and 0.5078 for avalanche test .We apply the test for AES-RC4 S-box and the result is 0.5235 and 0.5078 respectively which prove that AES-RC4 pass avalanche test. The results are given in Tables (5) and (6).

Table 3. AES-RC4 S-box

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	9A	E8	FE	F8	D0	6E	4A	EA	DC	F3	51	7C	99	4C	89	FF
1	53	63	27	44	11	01	08	D4	F5	F2	39	00	2C	0E	F0	06
2	3D	C8	9F	02	36	22	94	C6	2F	16	D5	E3	A8	4B	81	D1
3	15	2b	5A	D2	24	EF	41	A4	AC	12	AB	5F	35	2A	87	1B
4	BB	D8	45	90	1A	59	A0	64	31	18	23	CD	93	9D	A1	0D
5	4D	91	B7	C0	B1	20	74	09	66	ED	B9	68	D3	A9	6A	03
6	FA	B8	26	82	A7	E0	1F	CB	6A	19	43	3B	71	FC	8F	56
7	8d	32	30	AF	B0	F7	4E	57	7E	76	0C	88	97	BD	75	A6
8	8b	B5	21	38	CC	0A	DF	85	A5	3C	6C	47	C4	E4	CF	14
9	1D	B3	33	84	60	DD	73	F9	65	D9	13	EB	0F	7B	34	69
a	AD	2E	9B	3E	40	92	DA	96	F1	6D	58	78	0B	DE	70	86
b	7A	25	B4	B2	6b	BC	B6	29	C9	72	42	1E	10	05	04	FB
c	DB	BF	77	8C	54	E1	55	C2	F2	F6	1C	FD	46	98	95	5D
d	5b	AE	67	C5	9C	CA	F4	C3	EC	2D	A3	E5	E6	37	E7	8A
e	BA	83	E9	C1	A2	07	CE	3A	61	AA	9E	C7	D6	8E	82	7D
f	E2	7F	EE	FC	48	49	17	50	4F	3F	5E	80	79	62	BE	D7

Table 4. The inverse S-box

	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	1B	15	23	5F	BE	BD	1F	E5	16	57	85	AC	7A	4F	1D	9C
1	BC	14	39	9A	8F	30	29	F6	49	69	44	3F	CA	90	BB	66
2	55	82	25	4A	34	B1	62	12	EE	B7	3D	31	1C	D9	A1	28
3	72	48	71	92	9E	3C	24	DD	83	1A	E7	6B	89	20	A3	F9
4	A4	36	BA	6A	13	42	CC	8B	F4	F5	06	2D	0D	50	76	F8
5	F7	0A	C8	10	C4	C6	6F	77	AA	45	32	D0	6D	CF	FA	3B
6	94	E8	FD	11	47	98	58	D2	5B	9F	68	B4	8A	A9	05	5E
7	AE	6C	B9	96	56	7E	79	C2	AB	FC	B0	9D	0B	EF	78	F1
8	FB	2E	63	E1	93	87	AF	3E	7B	0E	DF	80	C3	70	ED	6E
9	43	51	A5	4C	26	CE	A7	7C	CD	0C	00	A2	D4	4D	EA	22
a	46	4E	E4	DA	37	88	7F	64	2C	5D	E9	3A	38	A0	D1	73
b	74	54	B3	91	B2	81	B6	52	61	5A	E0	40	B5	7D	FE	C1
c	53	E3	C7	D7	8C	D3	27	EB	21	B8	D5	67	84	4B	E6	8E
d	04	2F	33	5C	17	2A	EC	FF	41	99	A6	C0	08	95	AD	86
e	65	C5	F0	2B	8D	DB	DC	DE	01	E2	07	9B	D8	59	F2	35
f	1E	A8	19	09	D6	18	C9	75	03	97	60	BF	F3	CB	02	0F

4.2 Bit independence criteria (BIC) test

A second property which would seem desirable for any cryptographic transformation is that, for given set of avalanche vectors generated by the complementing of single plaintext bit, all the avalanche variable should be pairwise independent. In order to measure the degree of independence between a pair of avalanche variable, we calculate their correlation coefficient, if its zero it mean that the variable are independent, if its 1 that mean stronger positive correlation and -1 is stronger negative correlation the value obtained for AES S-BOX is -0.1530 and 0.2969 as given in Table (5) and for AES-RC4 S-box is 0.4688 for bit independence test given in Table (6),the AES-RC4 results is good comparing with AES S-box results.

Table 5. Avalanche effect and BIC for AES S-box

Input data	Output data	AVALANCHE TEST	BIC
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00	7D,F7,6B,0C,1A,B8,99,B3,3E,42,F0,47,69,1B,54,6F	0.4688	-0.1530
80,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00	F6,C7,1E,ED,C3D9,96,B1,83,CB,56,8D,15,68,EB,06		
11,11,11,11,11,11,11,11,11,11,11,11,11,11,11,11	98,AC,21,A7,EF,17,17,16,BF,CB,B6,8E,B8,5E,7F,C8	0.5078	0.2969
01,11,11,11,11,11,11,11,11,11,11,11,11,11,11,11	96,63,13,AE,F5,B4,48,52,B6,30,18,7A,BB,C8,62,C2		

Table 6. Avalanche effect and BIC for AES-RC4 S-box

Input data	Output data	AVALANCHE TEST	BIC
00,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00	A9,23,41,A8,1F,FF,EB,80,07,F3,C0,61,79,53,CA,EE	0.5234	-0.5953
80,00,00,00,00,00,00,00,00,00,00,00,00,00,00,00	3C,C3,19,3C,F0,B6,3D,E9,3B,96,EC,9E,3B,C0,FD,7C		
11,11,11,11,11,11,11,11,11,11,11,11,11,11,11,11	1D,BA,92,8F,49,FC,58,29,B8,ED,48,96,4E,74,01,E4	0.5078	-0.1543
01,11,11,11,11,11,11,11,11,11,11,11,11,11,11,11	FA,B6,58,C6,07,81,EE,E4,5D,43,40,B5,1D,7E,BE,FC		

4.3 Randomness test

In this test we use CrypTool [10] to test randomness of AES-RC4 S-box and comparing it with AES S-box using same inputs for both S-boxes , the two algorithms pass all test but with different test values as given in Table (7).

Table 7. Randomness tests for AES and AES-RC4

Test type	Maximum test value	Normal AES algorithm		AES-RC4 algorithm	
Frequency test	6.63	0.06	Pass	0.35	Pass
Poker test	18.4	5.0	Pass	8.19	Pass
Long run	34	22	Pass	20	Pass
run	13.2	2.02	Pass	5.45	Pass
serial	9.21	0.39	Pass	6.91	Pass

4.4 Image histogram test

In this test we run encryption of bmp image (babes.bmp)with the same key for different S-boxes then take the histogram for both image after encryption, fig(1) is the original image then is encrypted and decrypted with AES algorithm shown at the left hand(a) and also encrypted and decrypted using customize AES-RC4 algorithm (b)we can see the randomness of both algorithms in encrypted images .The histogram figs (2) is also taken for the two encrypted images at the left (a)for AES algorithm and at right hand(b) for our customized AES-RC4 taken for both respectively. We can see that the histogram of the two ciphered image is nearly the same and fairly uniform and significant different from the original image, there for, it does not provide any indication to employ any statistical attack.

Plaintext

AES encryption output
(ciphertext)

AES-RC4 encryption output
(ciphertext)

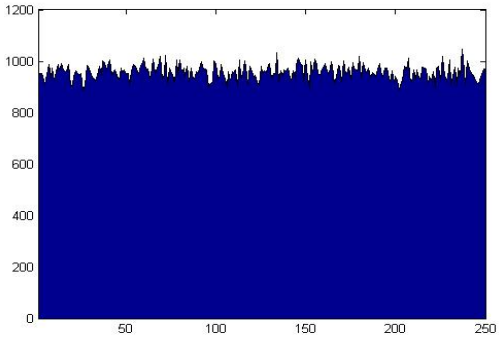
(a)AES decryption
(Recovered plaintext)

(b)AES-RC4 decryption
(Recovered plaintext)

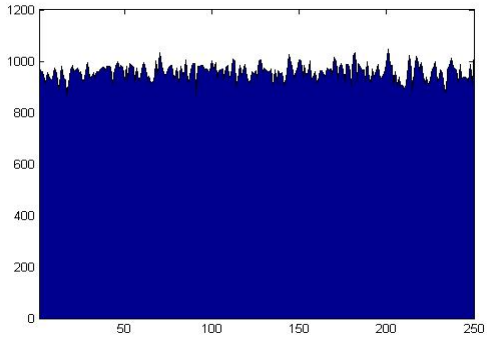
Fig. 1 Encryption and decryption using:
(a) AES algorithm, (b) AES-RC4 algorithm

Ciphertext using AES

Ciphertext using AES_RC4



(a) AES histogram



(b) AES-RC4 histogram

Fig. 2 Histograms of AES and AES-RC4 algorithms

5. Conclusion

In this paper a novel method for constructing cryptographically strong variable S-box dependant key is introduced. The proposed S-box passes the avalanche, bit independence tests and randomness tests which are important features for strong S-boxes to produce more confusion to the encryption process. We have other feature that the generated S-box are key dependent and can produce new S-box in every session so the encryption process being more dynamics. Also we can use two keys one for constructing S-box and other for encryption.

References

[1] A.F Webster and S.E Travares, "On The Design of S-boxes," Queen's university Kingston, Springer-verlag ,Canada 1998 .

[2] F. Fahmy and G. Salama, "A proposal for Key-dependant AES ," 3rd International Conference: Sciences of Electronic ,Technologies of Information and Telecommunications(SETIT) ,TUNISIA March 2005.

[3] Muhammad Asim, "Efficient and Simple Method for Designing Chaotic S-boxes," Electronic and Telecommunications Research Journal, University of Technology Petronas, Malaysia February 2008.

[4] M.Zeghid ,"A Modified AES Based Algorithm for Image Encryption," International Journal of Computer and Engineering VOL. 1.2003.

[5] Eltayeb Salih,"An Optimized Implementation of S-box using Residues of Prime Numbers," International Journal of Computer Science and Network Security (Vol. 8), April 2008.

[6] Melek,"Avalanche and Bit Independence Properties for the Ensembles of Randomly Chosen n×n S-boxes," Journal of Electric Engineering (VOL.9), Turkey, 2001.
[7] V.Ch.Venkaiah,"Variation to S-box and Mixcolumn Transformation of AES," International Institute of Information and Technology (Gachibowli), Hyderabad, India.
[8] Carlos Cid, "Algebraic Aspect of Advance Encryption Standard," 1st edition Springer, 2006.
[9] W. Stallings, "Cryptography and Network Security," 3rd edition, Pearson Education, 2003.
[10] CrypTool, Version 1.4.20 for Win32, May 2008B Beta 03 http://www.cryptool.org