Eman Mohammed Mahmoud, Ahmed Abd El Hafez, Talaat A. Elgarf, AbdelhalimZekry / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 3, Issue 1, January -February 2013, pp.1662-1670

Dynamic AES-128 with Key-Dependent S-box

Eman Mohammed Mahmoud*, Ahmed Abd El Hafez, Talaat A. Elgarf***, AbdelhalimZekry*****

*(Modern Academy of Engineering and Technology, Cairo, Egypt)
** (MTC, Cairo, Egypt)
*** (Higher Technological Institute, Cairo, Egypt)
**** (Faculty of Engineering, Ain shams University, Cairo, Egypt)

Abstract

Advanced Encryption Standard (AES) block cipher system is widely used in cryptographic applications. The main core of AES block cipher is the substitution table or S-Box. This S-box is used to provide confusion capability for AES. The aim of this paper is to design dynamic S-box which depends on the secret key. The parameters of the new created S-BOXes have characteristics equal to those in the original algorithm AES. This algorithm is suitable to exchange keys on insecure communication channels in order to achieve secure communications.

In this paper, a dynamic AES-128 with key dependent S-box is designed and implemented. Also, the quality of the implemented S-boxes is experimentally investigated. Also, the designed AES is compared with original AES in terms of security analysis, and simulation time.

Key words-**Advanced encryption standard (AES), dynamic S-box, S-box, security analysis**

1. INTRODUCTION

Advanced encryption standard (AES) is one of the widely used symmetric encryption algorithm. It is an encryption standard adopted by the US government. It is available in many different encryption packages. AES is first publicly accessible and open cipher approved by the National Security Agency (NSA) for top secret information.

AES was developed by two Belgian cryptographers, Joan and Vincent Rijmen. AES uses the Rijndael block cipher. AES is a very resilient algorithm that has shown resistance to all known cryptographic attacks so far. AES algorithm is a symmetric block cipher that can process data blocks of 128 bits using cryptographic keys of 128, 192 and 256 bits. Hereafter encryption/decryption with a cipher key of 128, 192, or 256 bits is denoted AES-128, AES-192, AES-256, respectively. [1] ,[2],[3].

AES-128, AES-192, AES-256 process the data block in, respectively, 10, 12, or 14 iterations of a pre-defined sequence of transformations, which are also called "rounds" (AES rounds) for short. The rounds are identical except for the last one, which slightly differs from the others (by skipping one of the transformations).

The rounds operate on two 128-bit inputs: "State" and "Round key". Each round from 1 to 10/12/14 uses a different Round key. The 10/12/14 round keys are derived from the cipher key by the "Key Expansion" Algorithm. This algorithm is independent of the processed data, and can be carried out independently of the encryption/decryption phase.

The data block is processed serially as follows: initially, the input data block is XOR-ed with the first 128 bits of the cipher key to generate the "State". This step is also referred to as "Round 0" which is using round key #0 (round key #0 is the first 128 bits of the cipher key). Subsequently, the State is serially passed through 10/12/14 rounds where the result of the last round is the encrypted (decrypted) block.

Each processing round involves four steps: 1. Substitute bytes – Uses an S-box to perform a byte by byte substitution of the block, 2. Shift rows – A simple permutation, 3. Mix column – A substitution method where data in each column from the shift row step is multiplied by the algorithm's matrix and 4. Add round key – The key for the processing round is XORed with the data [4].

This paper implements a 128-bit plaintext, and produces a 128-bit ciphertext under the control of a 128 bit secret key. This secret key is also used to generate an initial state of a pseudo random (PN) sequence generator. The output of PN generator is used to apply a dynamic permutation on the standard S-box. This step enables AES cipher to produce unexpected ciphertext due to S-box

modification [5], [6]. Correlation factor between standard AES and proposed one are calculated. Also avalanche effect test are applied to standard AES and designed one. Finally compare simulation time needed for different massage length.

The rest of the paper is organized as follows. In section 2, the S-box is discussed. In section 3, the proposed dynamic AES is described. Security analysis is presented in section 4. In section 5 a comparison between the dynamic AES with key dependent S-box and the standard AES is done. Finally, the conclusion is given in Section 6.

2. S-BOX GENERATOR

The S-box has the task of minimizing the susceptibility of the algorithm to methods of linear and differential cryptanalysis and to algebraic attacks. In addition to the requirement of complexity, the S-box function must be invertible; it must have no fixed points $S(a) = a$ or complementary fixed points $S(a) = \bar{a}$; and it must also execute rapidly and be easy to implement [7], [8].

S-box 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 of the byte 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 [9]. Table 1 represents S-box. During encryption each value of the state is replaced with the corresponding S-box value

For decryption the S-box must be used backwards. During decryption each value in the state is replaced with the corresponding inverse of the S-box. The inverted S-box appears in Table 2.

Table 1: AES standard 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	2B	76	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
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

Table 2: standard 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

3. THE PROPOSED DYNAMIC AES-128 WITH KEY DEPENDENT S-BOX

The proposed dynamic AES-128 with key dependent S-box is based on permutes the standard S-box under control of AES secret key. This secret key is manipulated and then applied to the PN generator as initial state. ThePN generator and AES secret key are used to generate two permutation sequences of length 16 hexadecimal values. These sequencesare used to arrange S-box vectors. This proposed algorithm leads toincrease the complexity and makes the differential and linear cryptanalysis more difficult.

3.1 The proposed PN sequence generator

The pseudo noise PN generator is responsible for generating perfect random sequence. Secure dynamic S-box permutation is based on this random sequence. The proposed generator consists of three maximal length linear feedback shift registers (LFSR) with thirty one, nineteen and fourteen taps. The feedback functions are chosen primitive to achieve a maximum period for each register [10].The feedback functions of the LFSRs are:

$$f_1 = X^{14} + X^{10} + X^{6} + X + 1 \quad (1)$$

$$f_2 = X^{19} + X^{5} + X^{2} + X^{1} + 1 \quad (2)$$

$$f_3 = X^{31} + X^{3} + 1 \quad (3)$$

Fig. 1 represents one possible PN sequence generator. The outputs of these LFSRs are connected through XOR gate. The period of this PN sequence is

$P_1 = (2^{14} - 1) * (2^{19} - 1) * (2^{31} - 1) = 1.8446 * 10^{19}$.

The output is divided to 128 bits blocks. Each block is used to change S-box dynamically.

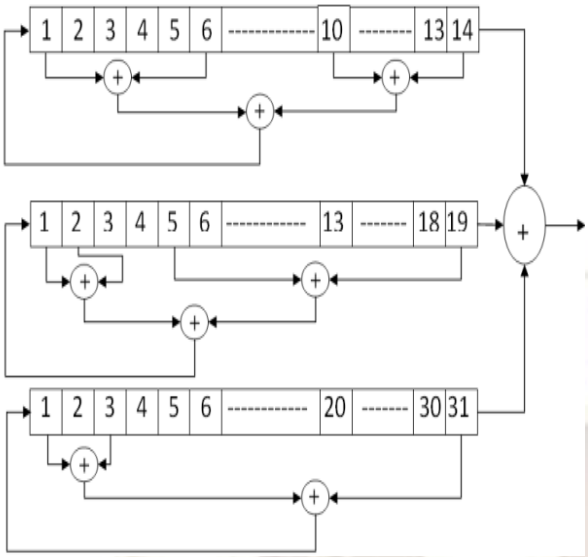

Figure 1: PN sequence generator.

The key length of this PN generator is (14+19+31) 64 so this generator needs 64 initial values. The AES secret key is reshaped to two vector of 64 bit length and these two vectors are Xored with each other and the result is fed to PN generator as initial state.

3.2 Permutation block

The permutation block is used to produce two permutation sequences S1 and S2. These sequences are responsible for rearranging the standard S-box column-wise and row-wise. This operation is based on AES-128 secret key and the generated PN sequence. The AES secret key is Xored with generated PN sequence. The result is converted to 32 hexadecimal values. The first 16 values are denoted by S1 and the other 16 values are denoted by S2. It is important to ensure that permutation values (S1 and S2) are normally distributed between (0, F). No repetitions are accepted. If S1 or S2 contains any repeated values, these repeated values are discarded and then the missing number is added to the sequence to ensure that all the S-box indexes are mapped.

3.3 The Key Dependent S-Boxes

The key dependent S-box bases on permute or rearrange standard S-box column-wise and row-wise interchangeable. S_1 vector is used to rearrange columns of standard S-box. S_2 vector is used to rearrange standard S-box rows.

Table 3 represents the key dependent S-box steps. Table 4, 5 represent the implemented S-box and its corresponding inverse S-box respectively when used secret key is: "B9B5ED7585C8B15D7454ED271AA3A3A3"

Table 3: Key Dependent S-Box Steps

Steps	Corresponding sequence (hexadecimal form)
Secret key	B9B5ED7585C8B15D7454ED271AA3A3A3
Initial state	C0155C55ADF95E8C
PN sequence	0B5AFCEC075DF07DBFEE5CFA9B8D2F52
Permutation sequence	B2EF119982954120CBBAB1DD812E8CF1
S_1 before arrangement	B2EF119982954120
S_1 after arrangement	B2EF198540367ACD
S_2 before arrangement	CBBAB1DD812E8CF1
S_2 after arrangement	CBA1D82EF0345679

Table 4: Key Dependent S-Box

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

Table 5: Key Dependent Inverse S-Box

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

4. SECURITY ANALYSIS

In order to ensure that implemented dynamic AES-128 with key dependent S-box is secure, some cryptographic tests must be applied. To facilitate interpretation of the experimental results, a brief description is given, to make the analysis of the tests' output understandable [11].

4.1 Randomness tests

Randomness tests are used to ensure randomness properties of the outputs corresponding to the implemented algorithm. Image Histogram test can be used for that purpose.

4.2 Avalanche effect

It is a desirable property of any encryption algorithm, which a small change in either the plaintext or the key should produce a significant change in the cipher text. In, particular a change in one bit of the plaintext or one bit of the key should produce a change in many bits of the cipher texts. 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.

4.3 Correlation factor

Correlation coefficient is a number between -1 and 1 which measures the degree to which two variables are linearly related. The correlation is 1 in the case of an increasing linear relationship, −1 in the case of a decreasing linear relationship, and some value in between in all other cases, indicating the degree of linear dependence between the variables. If the variables are independent then the correlation is 0.

4.4 Simulation time

The time required by the algorithm for processing completely a particular length of data is called the simulation time. It depends on the processor speed, complexity of the algorithm etc. The smallest value of simulation time is desired.

5. EVALUATION OF IMPLEMENTED AES WITH KEY DEPENDENT S-BOX

To simulate the dynamic AES with dependent S-box, a MATLAB (2011) script was implemented for both AES and dynamic AES. The key was fixed for both algorithms. Anumber of data blocks were encrypted. Several tests have been conducted to observe the performance of dynamic AES. Below experimental results are achieved:

5.1 Image Histogram test

In this test (moon.tif) image is encrypted using the same key for different S-boxes. The histogram for these encrypted images is plotted. Fig.2 shows the original image and its histogram. Fig.3 represents the encrypted image using standard AES algorithm with itshistogram. Fig. 4 shows the encrypted image using dynamic AES with key dependent S-box algorithm and its histogram.

The encrypted images represent the randomness properties of both algorithms. The histogram of the two ciphered image is nearly the same and fairly uniform and significantly different from the original image, therefore, it does not provide any indication to employ statistical attack.

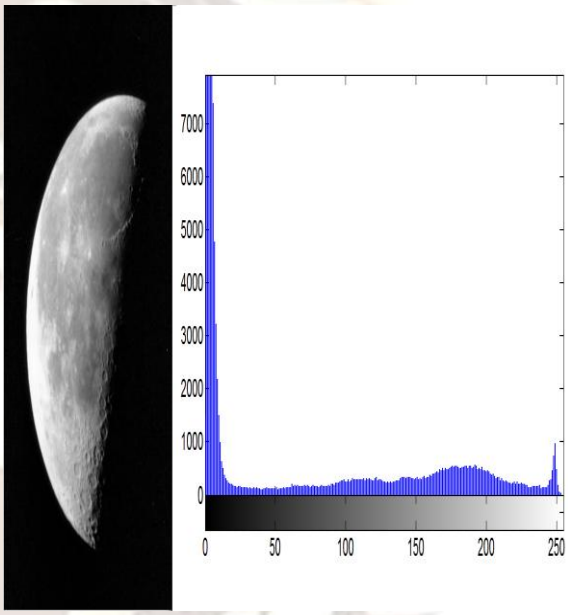

Figure 2: Original image & its histogram.

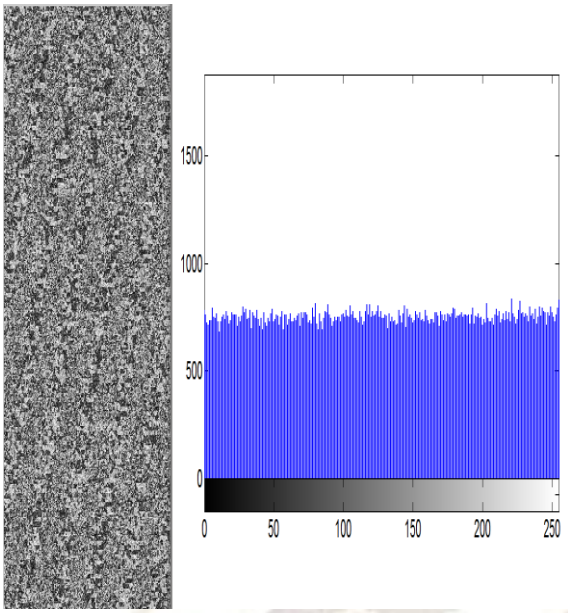

Figure 3: Standard AES encrypted image & its histogram.

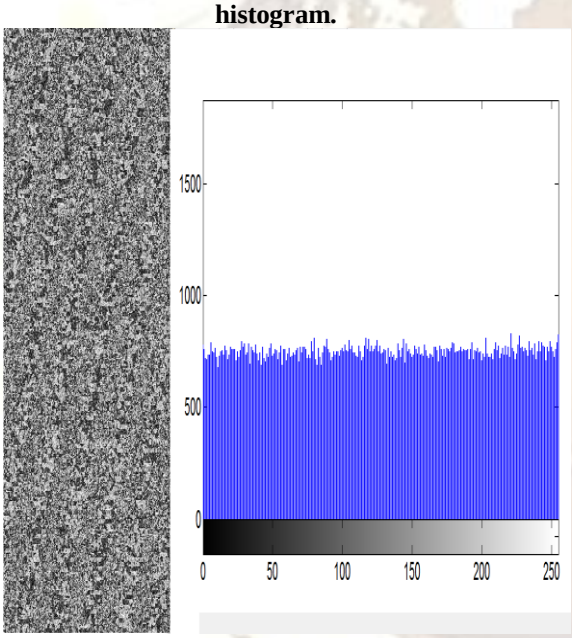

Figure 4: Dynamic AES encrypted image & its histogram.

5.2 Avalanche effect due to one bit change in plaintext

It is a primary design objective to apply avalanche effect on the implemented algorithm. Fig.5, 6 represent the avalanche effect of the standard algorithm and proposedone respectively due one bit change in the plaintext.The implemented algorithm represents avalanche effect lies between 45% and 63%, which means that it is difficult to cryptanalysis to make predictions about the input, being given only the output. The standard AES-128 and proposed AES have nearly the same avalanche effect.

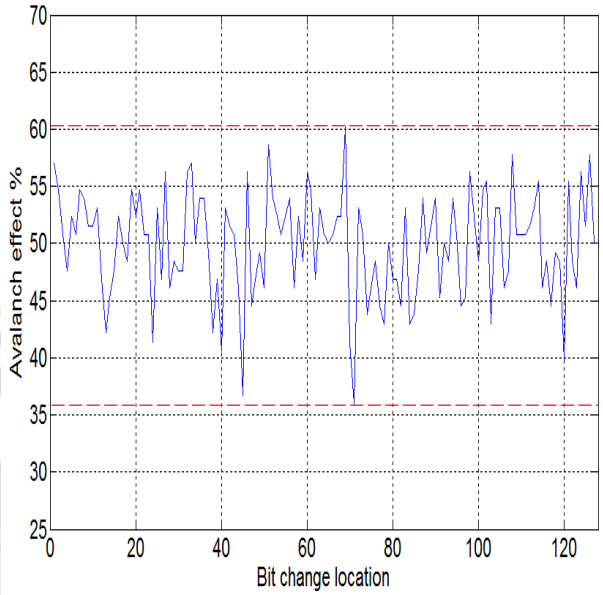

Figure 5: Avalanche effects of standard AES.

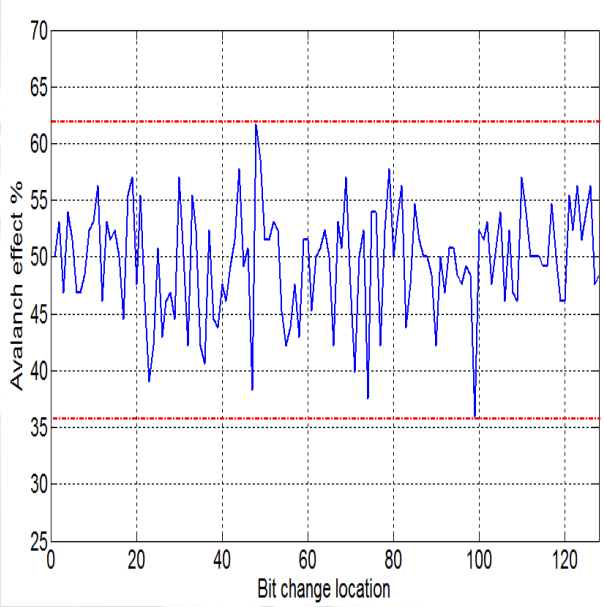

Figure 6: Avalanche effects of dynamic AES with key dependent S-box.

Table 6 represents samples of the standard and the proposed AES-128 ciphertext due to one bit variation in the plaintext.

Table 6 samples of Ciphertext due to one bit change in plaintext

Bit variation index	Plaintext	Ciphertext of standard AES	Ciphertext of dynamic AES
0	A07B00321C11759D0FDE340234384BC9	6EC1C564A5B556ED8DA6FF494B1942FE	7E2FE5CC639EDAAA7569FCF8EA677525
1	B07B00321C11759D0FDE340234384BC9	BEB94AF37F95ED11CFA3B1A3F60BB459	84EFC77A08B5ADC953FA55C488894A1B
2	807B00321C11759D0FDE340234384BC9	5EC53DF412699470424786D670F9F5AA	9E8AA518D01F56E7C7144BAB35B24E7E
6	A27B00321C11759D0FDE340234384BC9	351D0C6C16C89C55D4D36C6FE9150D99	603B330D10CAA0C466B7902D41AC68CC
28	A07B00B21C11759D0FDE340234384BC9	144EA67A93D3DC5E46F049BFB1A3C05D	DAA4A5B39CDBD63496A6827FC14196DB
36	A07B00329C11759D0FDE340234384BC9	1063C43BFB2105EEE778CBDB18BB40BD	9D3073946EDEBA6C5CE14758F21FDD19
47	A07B00321C15759D0FDE340234384BC9	64D83BDF4BB4B8A013950F12B8454406	1CACF4B3E4C9CBFC87A3906E90ABE634
50	A07B00321C11559D0FDE340234384BC9	94EC1376857385743910737A5FD42947	A1F1C967633CD5F2B6BBF39396987191
69	A07B00321C11759D0EDE340234384BC9	5AC5A1A91DD855AE24E673F318F3C197	3CAC61953C9349F06F33F7AFF72D430B
70	A07B00321C11759D0DDE340234384BC9	248B9FADDFD15AC4E88A93DC39AF8EF4	70C9C64E723957C5B0177D68C1D1B9F8
71	A07B00321C11759D0BDE340234384BC9	73A849A6E2BCA104AE1E5C70E832C2DE	D49E87264C95F8ADD21E6AD5B0DD023A
85	A07B00321C11759D0FDE350234384BC9	1FDA4608CD964B68E369C2E65F63E647	2E39EC1E04A41AE531B594EF1BB2E954
89	A07B00321C11759D0FDE341234384BC9	6F5C52C0F338E285A6392C078E9F8F80	078E516CD64ADAD835EF5A974F1C11CD
96	A07B00321C11759D0FDE340A34384BC9	32E47D9A932EE7A0B799A3734ECEF939	49D2815C10EE644B2DDE1F4DE8443DD5
97	A07B00321C11759D0FDE340224384BC9	2A65F6A733AB70AB3FD717F4B0B723D6	0DFDAF5E0618D35EA7BC8C6C27BBE11D
114	A07B00321C11759D0FDE340234386BC9	F3E47315CB0CC4BA65437863A63EDC2A	F876BABD3989263185EE64E7F81CD7D7
127	A07B00321C11759D0FDE340234384BCD	95B1FFBC83E62B7974079CC974906B1A	42159AC44B559ADC77A94D799AAA9631
128	A07B00321C11759D0FDE340234384BC1	B0A172630ED34B8624EC94A5996FBCAE	CE238E22D156E9CC5E63651E57A49CFF

5.3 Avalanche effect due to one bit change in secret key

Avalanche effect due to one bit variation in the secret key keeping plaintext constant is applied to verify the dynamic properties proposed AES with key dependent S-box algorithm. Consider the 128 bit length secret key in the hexadecimal form. The key is {AC05EB8D2006F5C3175B73706FD5A73A} and its corresponding S-Box is representing in table 4.

When only one bit of secret key is changed a 248 states are changed in corresponding S-box. Thus approximately 96.8 % of second S-Box is changed. The changed key is {BC05EB8D2006F5C3175B73706FD5A 73A} and its related S-Box is represent in table 7.

Table 8 represents samples of the proposed AES-128 ciphertext due to one bit change in the ciphertext.

Table 7 S-box for secret key {A9B5ED7585C8B15D7454ED271AA3A3A3}

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

Table 8 samples of Ciphertext due to one bit change in plaintext

Bit variation index	Plaintext	Ciphertext of dynamic AES	Avalanche effect
0	B9B5ED7585C8B15D7454ED271AA3A3A3	15F45A3A2D49C92B8BF8EF5BEC04B9A7	0
1	A9B5ED7585C8B15D7454ED271AA3A3A3	D2777E23A8672F3090CBB74EBABA99FB	51.5625
2	99B5ED7585C8B15D7454ED271AA3A3A3	A0FDDF50BF9A71F9D71AFB3A77F7B427	50.7813
16	B9BDED7585C8B15D7454ED271AA3A3A3	6757B9A4EB6BE21D50CDCBD81CDEA1A5	50.7813
26	B9B5FD7585C8B15D7454ED271AA3A3A3	3E4BB1E483B10FD3F07F341BE9A6C3C1	52.3438
42	B9B5ED7585E8B15D7454ED271AA3A3A3	15549737275CE656ADB6967A13639C3E	45.3125
44	B9B5ED758548B15D7454ED271AA3A3A3	E945BFCD7E221754D81A3882F01C836C	53.9063
59	B9B5ED7585C8B11D7454ED271AA3A3A3	2FFF460C4885DF2ED340FAD563766A29	52.3438
60	B9B5ED7585C8B1DD7454ED271AA3A3A3	068856D9E045A6ABD08D21CF5BCE8C18	59.3750
74	B9B5ED7585C8B15D7474ED271AA3A3A3	898E12B8EAAE9A5A545653146B84E2E0	50.0000
76	B9B5ED7585C8B15D74D4ED271AA3A3A3	4098DA2E8BAF9CE0F892DAFBB08CB363	57.0313
88	B9B5ED7585C8B15D7454E5271AA3A3A3	6B4DAE8676686B85331CA1EE16454340	50.7813
89	B9B5ED7585C8B15D7454ED371AA3A3A3	61216AC3B5A7722EE4F290A71EFE0C26	55.4688
97	B9B5ED7585C8B15D7454ED270AA3A3A3	C68D5FE7552B64362037CC1950C13598	42.9688
98	B9B5ED7585C8B15D7454ED273AA3A3A3	E0905686E5A06D1F6E32B4AF2BA7FDB5	45.3125
110	B9B5ED7585C8B15D7454ED271AA1A3A3	09A3A29F1D95D7715F6AFAFB490BF97F	42.9688
113	B9B5ED7585C8B15D7454ED271AA3B3A3	9A41EEADA4866C613A6040AE3F87961B	55.4688
120	B9B5ED7585C8B15D7454ED271AA3ABA3	FCA8DA5555B9674D7D7FD0143EA7AFBA	51.5625
127	B9B5ED7585C8B15D7454ED271AA3A3A7	0D25FFB79F8132A9588E1199E63F5339	50.0000
128	B9B5ED7585C8B15D7454ED271AA3A3AB	48B88039994106FD6CC5793ED1D16008	57.0313

Fig.7 represents the avalanche effect of the proposed AES-128 due to one bit change in secret key. It represents avalanche effect lies between 41% and 61%, which means that it is difficult to make predictions about the input, being given only the output. This reflects the immunity of our algorithm to linear and differential cryptanalysis.

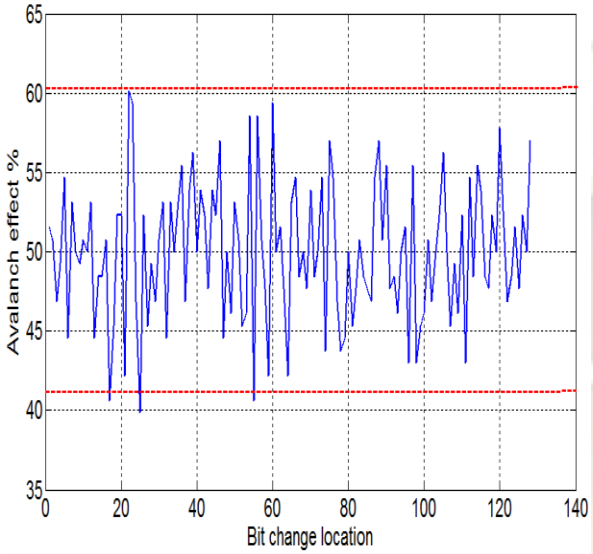

Figure 7: Avalanche effect due two one bit variations in secret key

5.4 Correlation factor

Fig. 8 represents the correlation factor between standard AES and implemented one. As correlation factor lies between -0.3 and 0.3 (ie. near zero) this means the two algorithms are independent on each other.

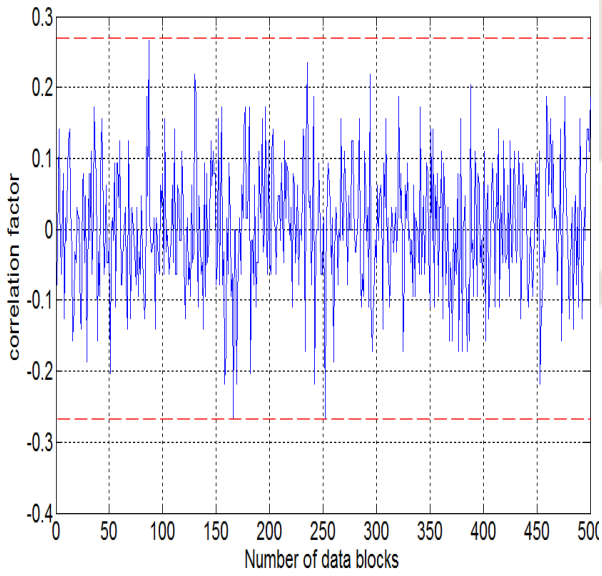

Figure 8: Correlation factor test.

5.5 Simulation time

The simulation time taken by two different algorithms due to 500 data blocks is recorded in sec in table 9. It is clear that standard AES is faster than dynamic AES.The difference in simulated time is corresponding to S-box dynamic change

Table 9: Simulation Time Comparison

	Standard AES	Dynamic AES	Difference
Simulation time in sec	2.963207	2.960076	0.0031

6. CONCLUSION

This paper presents a new approach to generate a dynamic AES with key-dependent S-boxes. It was established that for any change of the secret key, the structure of the S-box will be changed essentially. The performance of this approach is tested. For that purpose, the correlation ratio between standard AES and modified one is calculated. In order to validate the efficiency of the dynamic S-boxes, images were encrypted by the AES algorithm, and then the standard S-box was replaced by dynamic S-box. All the histograms were uniform and comparable. This leads to improve the AES security.

REFERENCES

[1] Rhee, Man Young. "Internet Security Cryptographic Principles, Algorithms and Protocols". England: John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester,West Sussex PO19 8SQ, 2003.

[2] Welschenbach and Michael. "Cryptography in C and C++". second . GraceWong,Michael Welschenbach, 2005.

[3] Boudriga Noureddine "SECURITY of mobile communications" [Book]. - [s.l.] : Taylor and Francis Group, LLC, 2010 .

[4] Joan Daernen Vincent Rijrnen "The Design of Rijndael {AES - The Advanced Encryption Standard}" [Book]. - Verlag Berlin Heidelberg : Springer-, 2002.

[5] I. Abd-ElGhafar A. Rohiem, A. Diaa, F. Mohammed "Generation of AES Key Dependent S-Boxes using RC4 Algorithm" [Conference] // "AEROSPACE SCIENCES & AVIATION TECHNOLOGY ASAT-13". - cairo : "Military Technical College", 2009.

[6] Sharma Himani Agrawal and Monisha "Implementation and analysis of various

symmetric cryptosystems" [Journal] // Indian Journal of Science and Technology. - India : Indian Society for Education and Environment (iSee), December 2010. - 12 : Vol. 3. - pp. 1173-1176.

[7] Faiz Yousif Mohammad Alaa Eldin Rohiem, Ashraf Diaa Elbayoumy "A Novel S-box of AES Algorithm Using Variable Mapping Technique" [Conference] // "AEROSPACE SCIENCES & AVIATION TECHNOLOGY". - cairo : Military Technical College, 2009.

[8] Stallings William "Cryptography and Network Security Principles and Practices" [Book]. - [s.l.] : Prentice Hall, 2006.

[9] Kazys KAZLAUSKAS Jaunius KAZLAUSKAS Key-Dependent S-Box Generation in AES Block cipher system [Conference] // INFORMATICA. - Vilnius : Institute of Mathematics and Informatics, 2009. - Vol. 20. - pp. 23–34.

[10] Mark Goresky, and Andrew Klapper" Pseudo noise Sequences Based on Algebraic Feedback Shift Registers", IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 52, NO. 4, April 2006

[11] Razi Hosseinkhani,H. Haj Seyyed Javadi Using Cipher Key to Generate Dynamic S-Box in AES Cipher System [Journal]. - 2012. - Issue (1): Vol. (6). - pp. 19-28.