Military Technical College
Kobry El-Kobbah,
Cairo, Egypt

8th International Conference
on Electrical Engineering
ICEENG 2012

A Novel S-box Substitution Technique for AES Algorithm

Faiz Yousif Mohamed M. E. Gadallah Alaa Eldin Rohiem Ashraf Diaa Elbayoumy

Abstract— In AES, the S-box substitution of each byte is depends on the input byte value only. In this paper, a novel method for substituting each byte is introduced. In this method, the substitution of each byte is depending not only on its value but also depend on the input key and the derived subkeys. The keys are used to generate two parameters that are used for mapping the substituted value horizontally and vertically to a random location in the S-box. The first parameter, subkey plus the input value, are used to rotate the row horizontally to the new location. The second parameter, subkey plus substituted value result from first parameter, are used to rotate the column vertically to new location which is final substitution value. The required modifications are made for the inverse substitution saving the non-linearity relationship between the S-box and its inverse. In this technique, we can get up to 256! Substitutions values instead of 256 values as in normal AES algorithm which increase the encryption complexity and complicate the cryptanalysis process. The results show that this is achieved with negligible extra delay. The increase of the substitution domain achieved from this Technique doesn't require to construct a new S-box, or inverse S-box as it does in the dynamic S-box approaches in which the need for re-evaluation and proved against cryptanalysis after every new S-box construction is compulsory.

Keywords— AES, Avalanche effect, Cryptography, Key dependent S-box, NISTSts, SAC.

1. Introduction

In October 2000, after four years effort to replace the aging DES, NIST announced the selection of Rijndael as the proposed AES (NIST 2004). [5]. Standardization of AES was approved after public review and comments, and published a final standard FIPS PUB-197 in December 2001. The input to the encryption and decryption algorithms is a single 128-bit block and three key length alternatives 128, 192, or 256 bits. This key is then expanded into an array of key schedule words: each word is four bytes and the total key schedule is 44 words for the 128-bit key. The input data will be partitioned into a rectangular array of bytes, called a state, the key is similarly pictured as a rectangular array with 4 rows and N_k columns, where N_k is equal to the key size divided by 32. Rijndael round function operates on a state N_r times, where N_r is equal to the number of rounds that can be 10, 12 or 14 rounds, depending on N_b and N_k, where N_b is equal to the block size divided by 32. Rijndael round is composed of 4 transformations:

1. ByteSub: S-box substitution provides nonlinearity and confusion.
2. Shiftrow: rotations, provides inter-column diffusion.
3. MixColumn: linear combination provides inter-Byte diffusion.
4. AddRoundKey: round key bytes XOR into each byte provide confusion

Decryption is applying the operations in a reverse order with respect to the order of encryption. For more details refer to [3][6]. S-box Construction was attempted by many designers using a fixed irreducible polynomial for higher efficiency and a smaller footprint of the AES intellectual property (IP). For long-term use, however, the fixed irreducible polynomial has been proven to make the system's golden key obvious, thus increasing the decryption rate of confidential files. The decryption methods include side channel, time channel, and power side channel attacks. Some systems can even be decrypted by an inside job [3].

2. Background

Techniques for the construction of S-boxes must follow set of criteria, as given by [9-11][14]:

1. Maximization of the nonlinearity so as to provide resistance against linear attacks.
2. No fixed points (S-BOX[x] = x) or reverse fixed points (S-BOX[x] = x).
3. Complexity of the equivalent algebraic S-box description in GF2.
4. Increase the over all security(increase confusion not lead to decrease diffusion).
5. Satisfy the Strict Avalanche Criterion (SAC): If an S-box does not satisfy the SAC, some output bits will be dependent only on some input bits.
6. Output bits change independently i.e. satisfy the Bit Independence Criterion.

Many attempts were satisfied some of the above criteria and the more relevant works techniques were discussed below:

2.1 Key Dependant AES

Key Dependant AES (AES-KDS) is a technique attempts to make AES key dependent [7]. AES-KDS is block cipher in which the block length and the key length are specified according to AES specification: Encryption and decryption process, number of rounds, data and key size were chosen as AES specifications. But the S-box construction was attempt in the following fashion:

1- Initialize the S-box: The first column contains 0x00, 0x01,……, 0x0F. The second column contains 0x10, 0x11,… etc, and so on. Thus, the value of the Byte at column x and row y is [xy].

2- For each byte value of the key, k_i (for 0 =< I <= key length), for example, if the key length is 16 Byte, the first byte is k_1, then k_2 and so on. Examine the value of k_i, if (k_i mod 2) equals zero, run a pseudorandom generator for the value of k_i, otherwise run another, also for the value of k_i. Two linear congruence pseudorandom generators are used, called rand1 and rand2 that make use of the linear congruence parameters (Michael 2001) for ISO-C Standard and GNU-C respectively.

3- The last run value of the selected pseudorandom generator r is added to the mean of the key to introduce a loop counter value for the swapping loop. loopcounter = [mean (key) + r +1].

4- Again, use rand1 and rand2 to generate two byte values that serve as indexes into S-box to select two bytes to be swapped together. This operation continues until the loop counter ends.

5- Repeat steps 2, 3, and 5 until all the key byte values k_i (for 0 =< i <= key length) has been taken.

The inverse substitute byte transformation, called InvSubBytes, makes use of the inverse S-box. The inverse S-box is constructed by determining a substitution pair and replacing it with its inverse [7].

As shown from above discussions, the AES S-box is completely replaced by a new S-box. This eliminates completely Inverse S-box, which violates AES design and hence requires thorough analysis regarding its security, because AES S-box must be tested thoroughly for linear, differential and algebraic attacks.

In general all competing ideas chosen to create good S-box design have one or another outward appearance from the following shortcomings:

- Reconstruct the S-box every execution time decrease the system efficiency and complicates the hardware implementations [2,3,8].
- Repetitive security evaluation: When a new S-box generated then it need to be tested upon different types of attacks.
- The range of indexes mapping is very limited i.e. every index has only one picture from the substitution domain values. Construction of more than one S-box during one session leads to high overhead.

The above Related works shortcomings become a motivation for new design paradigm in which the above construction overheads must be eliminated or minimized, and mapping domain range must be increased for each S-box element.

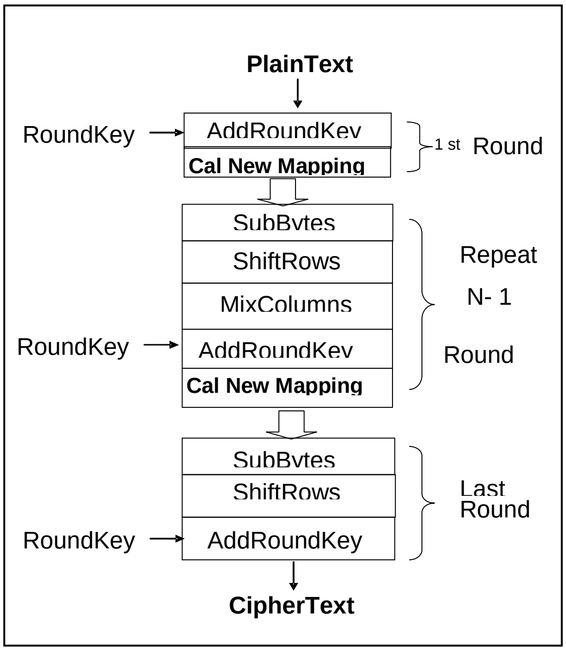
3. A Novel S-box of AES Algorithm Using Variable Mapping

AES with Variable Mapping S-box (VMS-AES) is block cipher algorithm in which the block length, the key length and the round functions are specified according to AES specification. Figure (1) shows the overall structure of VMS-AES.

A new function was added for Rijndael round transformation which represents a new substitution method in which the substitution is not depend only on input data, but it depend also on the input and derived subkeys. The inverse of the new function was designed to be used in decryption round transformations.

3.1 New S-box Construction

Four irreducible polynomials shown in table (1) were used to create new substitution tables (S-boxes) then they were evaluated so as the best one of them was built-in (VMS-AES). Table (2) and Table (3) describe the S-box and its inverse for the polynomial ($X^8+X^6+X^5+ X+1$), the other S-boxes and their inverses were appeared in appendix A.



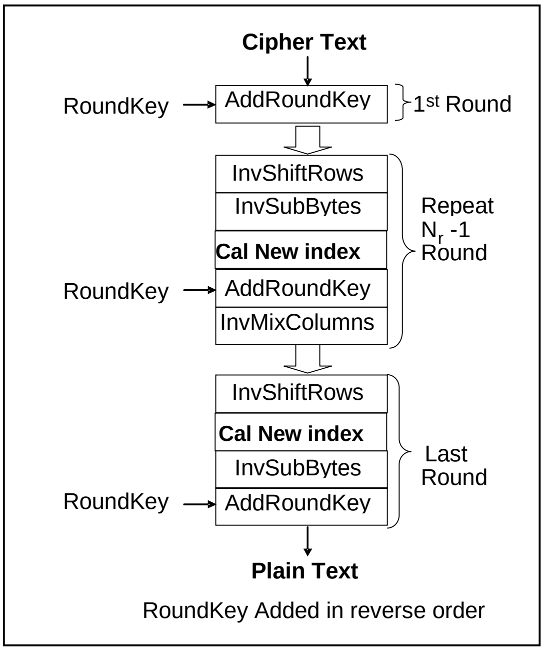


Figure (1a): VMS-AES encryption

Figure (1b): VMS-AES

decryption

Table (1) different polynomial used to create new S-boxes

No	polynomial
1.	$X^8 + X^6 + X^5 \quad X + 1$
2.	$X^8 + X^5 + X^4 + X^3 + X^2 + X + 1$
3.	$X^8 + X^6 + X^5 + X^2 + X + 1$
4.	$X^8 + X^6 + X^5 + X + 1$

Table (2) S-box generated from the polynomial (X^8+X^6+X^5+ X+1)

X

		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0	63	7c	e1	60	2f	62	e2	68	48	6c	e3	34	ae	29	e6	95
	1	Fb	ce	e9	8f	2e	0d	c5	53	85	4d	46	66	a1	a7	15	eb
	2	12	2B	b8	16	2b	a5	18	d6	c8	9d	54	79	3d	9f	76	65
	3	1d	17	74	0e	f1	31	ec	51	0f	8c	01	ab	58	43	2a	b0
	4	c3	db	52	6a	83	32	d9	8a	47	ea	00	30	d3	d2	b4	b7
	5	b6	81	11	4f	f8	b1	6e	02	41	7b	10	96	e4	8d	6d	5b
	6	5c	f5	59	4b	e5	b9	d5	40	27	06	4a	a9	a4	c4	77	21
Y	7	55	9e	99	56	5f	05	0a	37	f3	2c	7e	c0	c7	9c	87	14
	8	33	4e	3f	cc	f6	b3	e7	36	13	98	cb	af	3e	fd	97	86
	9	71	08	aa	c1	df	70	ca	1a	3b	a6	bb	dc	88	49	09	be
	A	89	93	12	ee	57	84	75	9a	a3	8b	07	dd	e8	bc	de	23
	B	7f	5d	6f	61	d7	67	94	f7	a0	38	19	bf	69	25	72	f0
	C	Fc	ba	28	f4	73	9b	7a	3a	20	cd	03	a2	35	d4	ff	5a
	D	4c	d0	d1	92	fa	24	0b	0c	80	04	bd	ed	64	ad	42	fe
	E	78	82	90	ac	1e	a8	f9	c6	7d	1f	50	6b	da	cf	44	ef
	F	26	39	c9	c2	e0	8e	b2	5e	3c	b5	91	45	1c	f2	d8	2d

Table (3) Inverse S-box generated from the polynomial (X^8+X^6+X^5+ X+1)

		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
	0	4a	3a	57	Ca	d9	75	69	aa	91	9e	76	D6	d7	15	33	38
	1	5a	52	a2	88	7f	1e	23	31	26	ba	97	21	fc	30	e4	e9
	2	c8	6f	20	Af	d5	bd	f0	68	c2	0d	3e	24	79	ff	14	04
	3	4b	35	45	80	0b	cc	87	77	b9	f1	c7	98	f8	2c	8c	82
	4	67	58	de	3d	ee	fb	1a	48	08	9d	6a	63	d0	19	81	53
	5	Ea	37	42	17	2a	70	73	a4	3c	62	cf	5f	60	b1	f7	74
Y	6	03	B2	b3	05	00	dc	2f	1b	b5	07	bc	43	eb	09	5e	56
	7	95	90	be	c4	32	a6	2e	6e	e0	2b	c6	59	01	e8	7a	b0
	8	d8	51	e1	44	a5	18	8f	7e	9c	a0	47	A9	39	5d	f5	13
	9	e2	fa	d3	a1	b6	0f	5b	8e	89	72	a7	C5	7d	29	71	2d
	A	b8	1c	cb	a8	6c	25	99	1d	e5	6b	92	3b	e3	dd	0c	8b
	B	3f	55	f6	85	4e	f9	50	4f	22	65	c1	9a	ad	da	9f	bb
	C	7b	93	f3	40	6d	16	e7	7c	28	f2	96	8a	83	c9	11	ed
	D	d1	D2	4d	4c	cd	66	27	b4	fe	46	ec	41	9b	ab	ae	94
	E	f4	02	06	0a	5c	64	0e	86	ac	12	49	1f	36	db	a3	ef
	F	Bf	34	fd	78	c3	61	84	b7	54	e6	d4	10	c0	8d	df	ce

3.2 AES vs. VMS-AES

Table (4) represents the main difference between VMS-AES and AES

Table (4)VMS-AES vs AES

The item	AES	VMS-AES
Data Block length	128 Bit	Same
Key length	128/192/256 Bit	Same
Number of rounds	10/12/14 round	Same
Round functions	ByteSub, ShiftRow, MixColoum, AddroundKey	Same + CalNewMapping, CalNewIndex
Last round	Has no MixColoum	Same
S-box	Fixed (one to one mapping function)	New polynomial + New mapping method
Mapping to S-box	Depend on the State Byte	Depend on the State Byte and on the Keys

3.3 VMS-AES S-box mapping Calculation

S-box substitution represents the main difference between VMS-AES proposal and the standard AES. In VMS-AES the forward substitute byte transformation, operate as AES SubBytes function except a shift parameter value would be calculated and added to the index parameter, so as to shift the substitution to new secret location, new because it shift from the specific static location (as in AES) to another location varied from Byte to Byte and this variation does not depend only on the left most and right most of the Byte but it depend also on the SHIFT parameter. The location is secret because it depends on a secret key, and here the substitution stage provides another degree of security in addition of the nonlinearity and confusion which provides from AES style. Figure (2) illustrates the substitution and the inverse substitution processes in VMS-AES. Figure (3) shows the mapping range for one byte from the plain text state. S is another parameter that could be sending within the packet payload for the application not sensitive to QoS like the file transfer. S parameter used to rotate SHIFT contents to the right direction. During decryption S is used to reverse the rotation direction. The above simple rotations could increase the encryption complexity and complicate the cryptanalysis process.

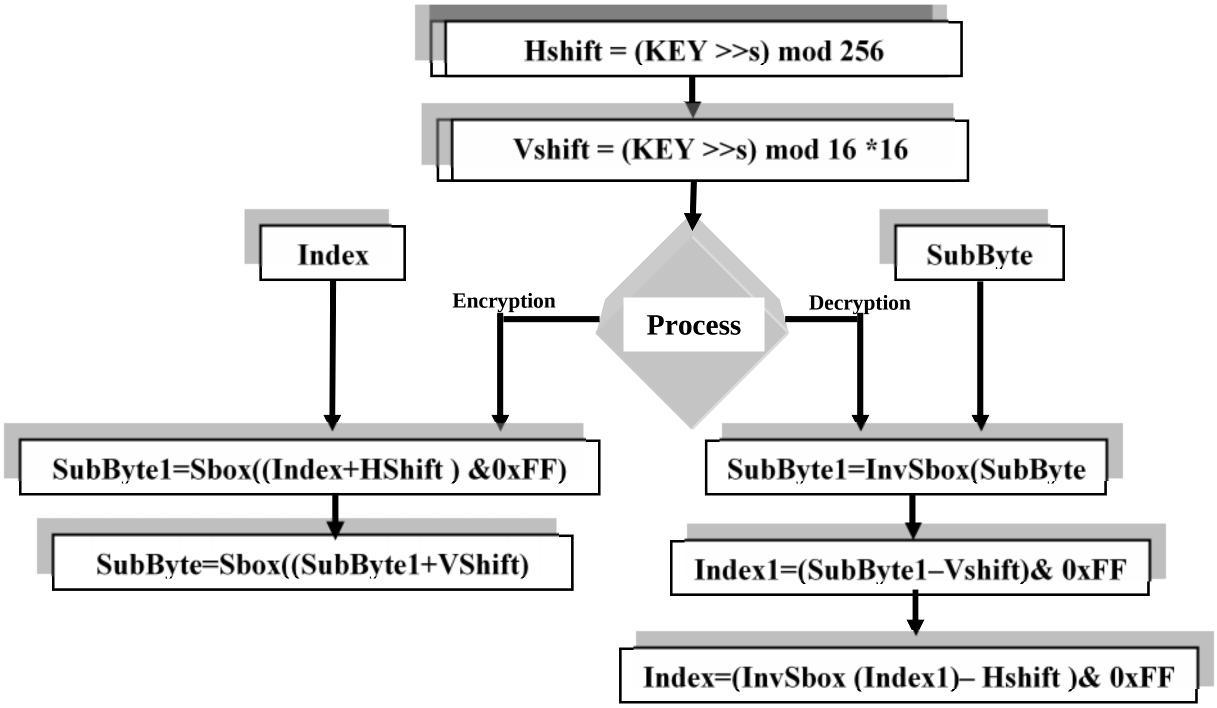

Hshift = (KEY >>s) mod 256
Vshift = (KEY >>s) mod 16 *16
Index
SubByte
Encryption
Process
Decryption
SubByte1=Sbox((Index+HShift) &0xFF)
SubByte=Sbox((SubByte1+VShift)
SubByte1=InvSbox(SubByte
Index1=(SubByte1–Vshift)& 0xFF
Index=(InvSbox (Index1)– Hshift)& 0xFF

Fig. (5.5): The Substitution and the inverse Substitution in VMS-AES

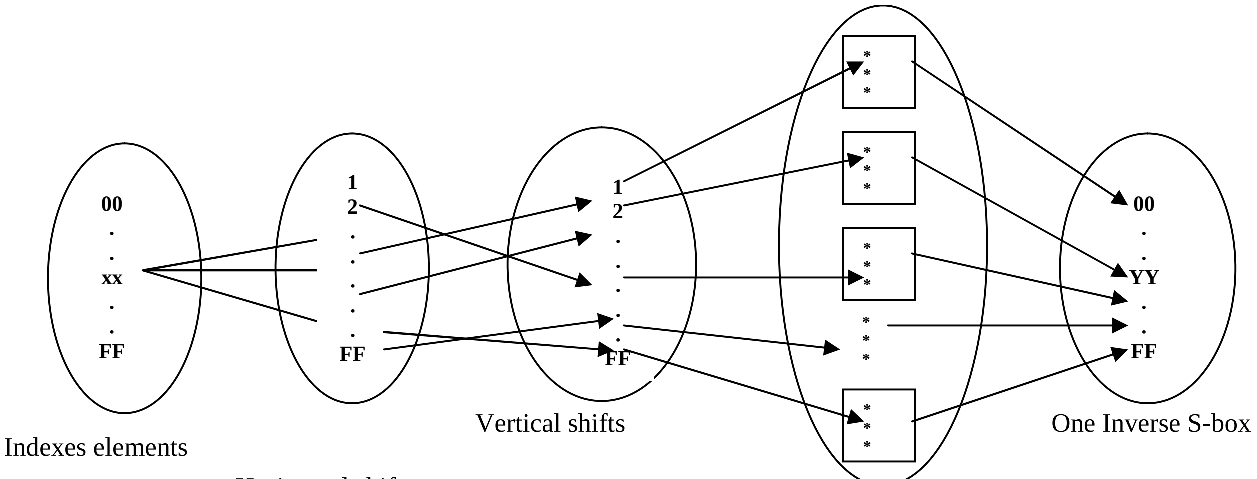


Figure (3): The Mapping range for one byte (index) in VMS-AES

4. VMS-AES S-box Evaluation

To evaluate VMS-AES the following verification methods were applied:

1. NIST statistical suite.
2. Correlation Coefficient.
3. Avalanche effect.
4. Strict avalanche criterion.
5. Measurement of Encryption Quality.
6. Mathematical demonstration
7. Timing and quality of service aspects.

4.1 NIST Statistical suite

The National Institute of Standards and Technology (NIST) develop a Test Suite as a statistical package consisting of 16 tests that were developed to test the randomness of (arbitrarily long) binary sequences produced by either hardware or software based Cryptographic random or pseudorandom number generators. These tests focus on a variety of different types of non randomness that could exist in a sequence. Some tests are decomposable into a variety of subtests [12].

A number of files (with different types) were encrypted using (VMS-AES, AES) algorithms. The encrypted files were entered as inputs parameters to the 16 tests of NIST statistical suite. The average values of the statistical tests for both algorithms were shown in Figure (4). The tests were applied to a different number of rounds for VMS-AES, and this feature clarifies the VMS-AES flexibility. Note that both proposals (AES &VMS-AES) were pass all NIST Statistical tests.

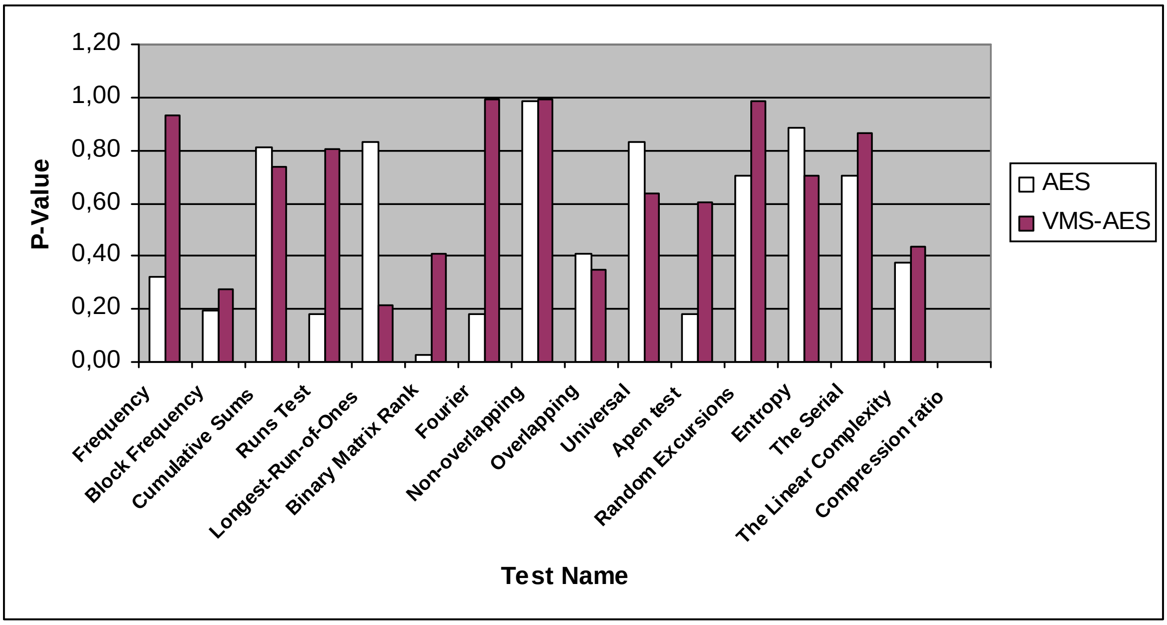
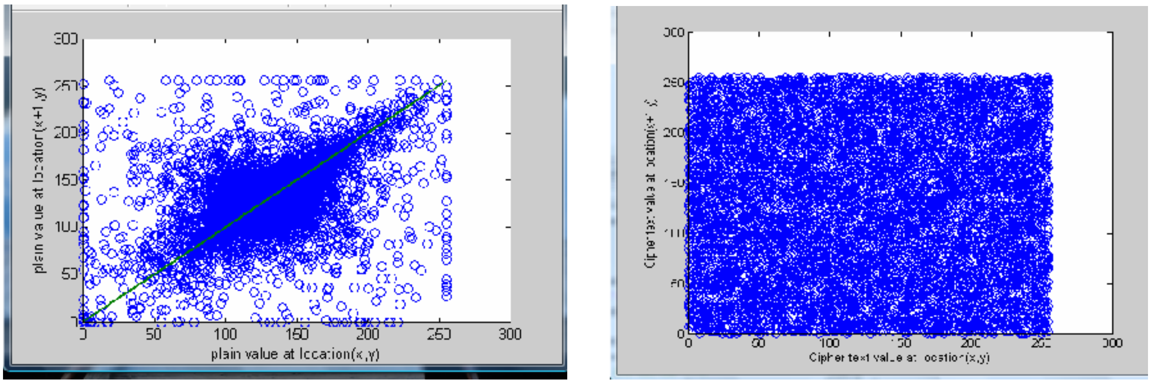


Figure (4): VMS-AES vs AES Statistical tests average p - values

4.2 Correlation Coefficient

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.

Figures below show the correlation distribution of two horizontally adjacent codes in the plaintext/ciphertext for VMS-AES block cipher. Matlab package is used to calculate the Correlation Coefficient in a numerical form and it had a value of (-0.07491) which represent the departure of the plain and cipher text from independence.



(a) (b)

Figure (5) Correlation Coefficient for (a) plain text, (b) cipher text

4.3 Avalanche effect

Avalanche effect is a characteristic of an encryption algorithm in which a small change in the plaintext or key give rise to large change in the ciphertext (more than half)[13].

Avalanche effect measurement subsystem was designed and called in recursive mode to calculate the average value of avalanche effect for 20 millions experiments with different random keys used to encrypt a fixed plain text by VMS-AES and AES algorithms simultaneously. Figure (6) shows the average results for different number of rounds. The tenth down to the third round pass Avalanche test.

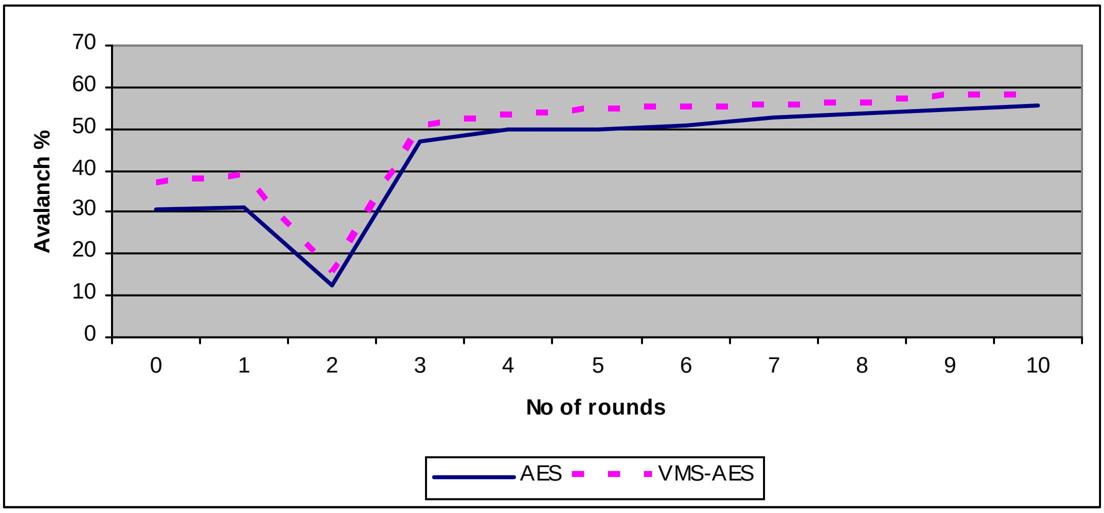


Figure (6): Avalanche effect percentage for VMS-AES & AES

VMS-AES was executed to encrypt a constant plain text and constant key data with all possible S-boxes that could be generated due to the change of shift parameter value. Different cipher text for every shift value was given. The avalanche effect was calculated and its value was equal or greater than **70%** for all different situations.

4.4 Strict Avalanche Criterion

The concepts of completeness and the avalanche effect can be combined to define a new property which we shall call the strict avalanche criterion. If a cryptographic function is to satisfy the strict avalanche criterion, then each output bit should change with a probability of one half whenever a single input bit is complemented. A more precise definition of the criterion is as follows. Consider X and X_i , two n-bit, binary plaintext vectors, such that X and Xi differ only in bit i, $1 \leq i \leq n$. Let V_i =Y Y_i where Y = f (X), Y_i = f (X_i) and f is the cryptographic transformation, under consideration. The value of bit i in V_j (either a 1 or a 0) is added to element $a_{i, j}$ in the m x n dependence matrix A. This procedure is repeated for a large number, r, of randomly generated plaintext vectors X, and each element in A is divided by r. Then each $a_{i,j}$ gives the strength of the relationship between plaintext bit j and ciphertext bit i. A value of 1 indicates that whenever bit j is complemented in the plaintext then the ciphertext bit i will also change its value, while a value of 0 indicates that the ciphertext bit is completely independent of the plaintext bit. If all elements in the matrix have a nonzero value then the cryptographic transformation is complete, and if it is to satisfy the strict avalanche criterion, every element must have a value close to one half. Therefore, completeness is a necessary condition if the strict avalanche criterion is to be met.

The strict avalanche criterion matrix given in Table (5) and Figure (7) indicates that each element in the strict avalanche matrix has a value close to one-half. So, the new proposal satisfies the strict avalanche criterion.

Table (5): VMS-AES strict Avalanche criterion matrix

0.500030	**0.499822**	**0.499683**	**0.500310**	**0.499879**	**0.499992**	**0.499602**	**0.499757**
0.500028	**0.499822**	**0.499683**	**0.500310**	**0.499880**	**0.499992**	**0.499601**	**0.499757**
0.500028	**0.499822**	**0.499683**	**0.500310**	**0.499879**	**0.499992**	**0.499602**	**0.499757**
0.500030	**0.499822**	**0.499683**	**0.500310**	**0.499880**	**0.499992**	**0.499602**	**0.499758**
0.500030	**0.499822**	**0.499683**	**0.500311**	**0.499879**	**0.499992**	**0.499602**	**0.499757**
0.500030	**0.499822**	**0.499683**	**0.500311**	**0.499880**	**0.499992**	**0.499602**	**0.499756**
0.500030	**0.499822**	**0.499683**	**0.500311**	**0.499879**	**0.499992**	**0.499602**	**0.499758**
0.500030	**0.499822**	**0.499683**	**0.500311**	**0.499880**	**0.499992**	**0.499602**	**0.499757**

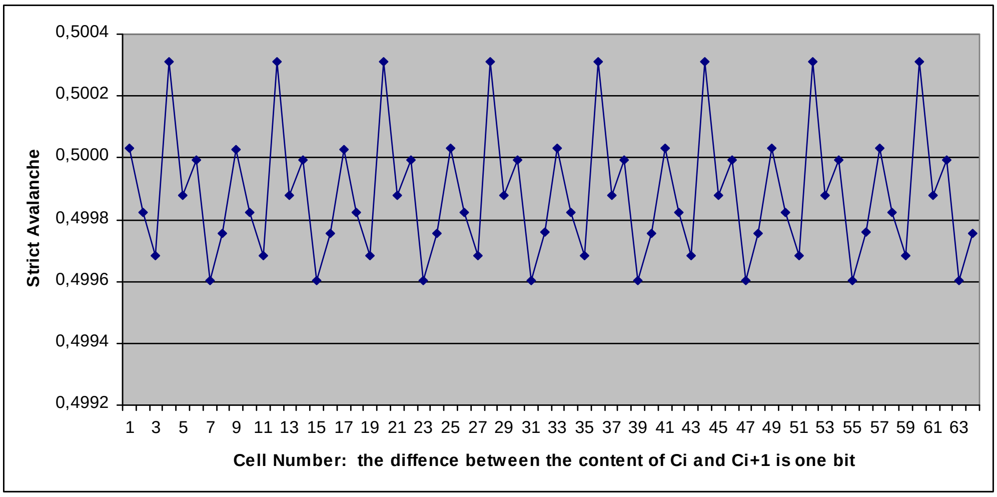


Figure (7): VMS-AES strict Avalanche criterion

4.5 Measurement of Encryption Quality

Encryption quality may be expressed as the deviation between the original and encrypted output. Let F, F′ denote the original (plaintext) and the encrypted output (ciphertext) respectively, each of size M samples with L different samples. F(x,y),F(x,y) Є{0,.., L− 1} are the different samples of F , F′ .We will define HL(F) as the number of occurrence for each sample L in the original (plaintext), and HL(F') as the number of occurrence for each sample L in the (ciphertext). The encryption quality represents the average number of changes to each sample and it can be expressed mathematically as Following :

$$\text{Encryption Quality} = \frac{\sum_{L=0}^{255} \left| H_L(F') - H_L(F) \right|}{256}$$

Figure (8) illustrates results of Encryption quality (for 1.2 G.byte plaintext and ciphertext files) of AES and its comparison with AES-VMS. From the results we can conclude that the modification to AES will not affect the Encryption Quality of the cipher in any way.

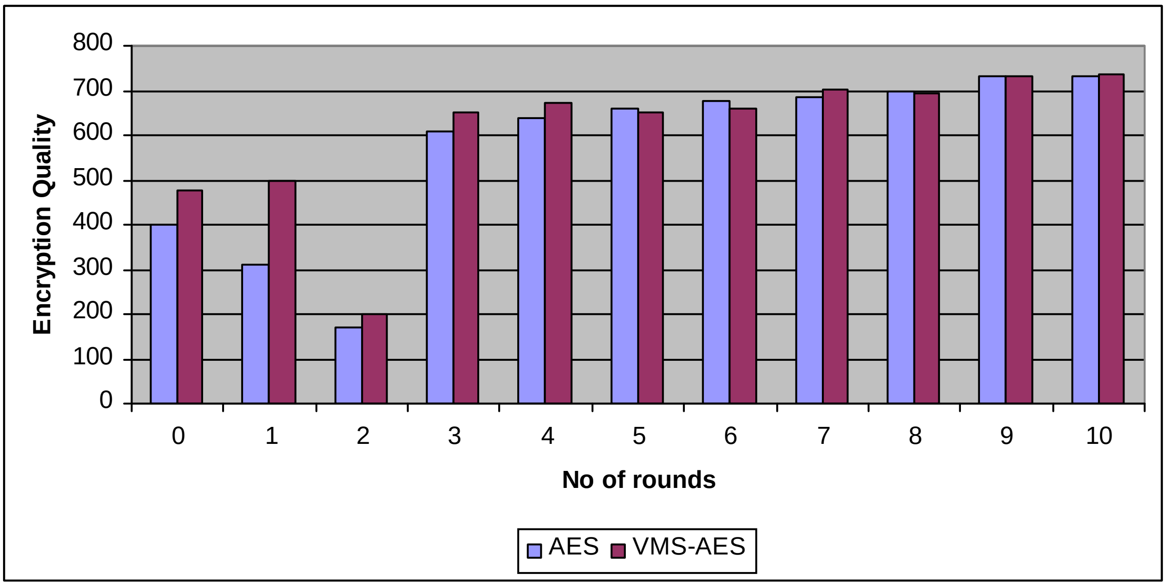


Figure (8): AES & VMS-AES Measurement of Encryption Quality

4.6 Mathematical demonstration

Refer to Figure (9) two types of permutations could be mentioned: the horizontal rotation (in ***dash blue lines***) and vertical rotation (in Continuous green) were give the global moving ability to all S-box elements.

D. Green [15] defined permutation as: an arrangement or sequence of selection of elements from single set, the order in which the elements are selected or arranged is significant. In general the permutation P(n,k) indicate the number of arrangements that can be formed by selecting k elements from a set of n elements[16]. The observed pattern it may be written as:

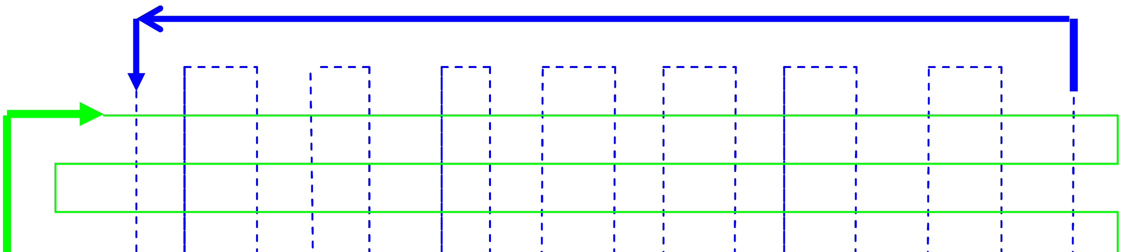
$$P(n,k)=n(n-1)(n-2)\ldots\ldots\ldots\ldots\ldots\ldots(n-k+1);$$

There is special case when a permutation uses all elements of set, in general P(n,n) is equal the product of integers n through 1. The following notation could be used [17]:

$$P(n,n)=n(n-1)\ldots\ldots\ldots\ldots\ldots 3*2*1= n!$$

Form the above mathematical notations it concludes that the number of different ways that could be used to re-arrange the original S-box are equal P(256,256)=(256!). Refer to Figure (9) the number of different substitutions for one byte during one session could be reached 256! i.e. that means every byte has one substitution from every virtual S-box and this gives the clear picture of the difficulty involved in the cryptanalysis.

Since VMS-AES apply all different probabilities that could be used to construct a new S-box so it could be conclude that any other solution can be assumed as a subset of S-boxes which VMS-AES has being already applied.



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

Figure (9): VMS-AES Virsual S-box rotations (Horizontal in greenlines & Vertical in dash blue lines)

AES encryption stage consists three different sets: the Plain text set **Ƥ**, substitution elements set **X** (S-box) and the inverse S-box set **I**. All the mentioned sets have the same domain {00,….,FF}, so we can say:

Ƥ= **X**= **I**= {00,…,FF}; With the same cardinality

ň(**Ƥ**)= ň(**X**)= ň(**I**)= 256;

For any element (p) belongs to the plain text **Ƥ** there exist one and only one picture x from S-box substitution group **X** in one-to-one correspondence relationship, which can be denoted as follows:

S: **Ƥ** → **X** | $\forall p \in$ **Ƥ** $\exists x \in$ **X** | S(p)=x --------------> (1)

where S is the S-box substitution function. The power set of S is the set of all subsets of S could be expressed as follows

$\rho(S) = \{(0,63),\ldots.,(FF,16)\}$ ----------------------> (2)

In VMS-AES the new plain text group **N** represents the plain text elements after add the shift value.

$\mathbf{N}=\{n_1,n_2,\ldots,n_n\}$

Since $n_i= (p_i+\text{shift}) \bmod 256$; where p_i is the correspondence plain text element then we can say

$0 \leq n_i \geq 255$

And this lead to unambiguous that the **N** domain in Hexadecimal is: **N**={00,…,FF}; And this is implies that **(N,Ƥ,X,I)** have the same power set.

N=Ƥ=X=I;

For any element n belong to the plain text **N** there exist one and only one picture x from S-box substitution group **X** such that R(p)=x; R is VMS-AES function which used for S-box substitution and can be denoted as follows:

$R: \mathbf{N} \rightarrow \mathbf{X} \mid \forall n\in\mathbf{N}\ \exists x\in\mathbf{X} \mid R\,(p)=x$ -------------> (3)

To prove that R =S let n∈**N** arbitrarily chosen element then

Since n∈**N** and $0 \leq n_i \geq 255$ this implies there exist $p\in\mathcal{P}$ such that p=n. from (3) R(n) =x then R (p)= x such that x∈**X**. Since n is arbitrarily chosen then S(n) =S(p) for all values of n. So The power set of S is equal to The power set of R

$\rho\,(S)=\rho\,(R\,)$ ----------------------> (4) with different arrangement.

This is lead to S=R.

In AES Decryption: For any element x belong to the S-box substitution group **X** there exist one and only one picture x' from Inverse S-box substitution group **I** such that S'(x)=p; S' is the Inverse S-box substitution function which can be denoted as

$S' : \mathbf{X} \rightarrow \mathbf{I} \mid \forall x\in\mathbf{X}\ \exists p\in\mathcal{P} \mid S'(x)=p$ -------------> (5)

The power set of S is the set of all subsets of S' i.e.

$\rho\,(S')=\{(63,0),\ldots,(16,FF)\}$ --------------------> (6)

VMS-AES uses the same inverse S-box substitution group **I** for all different S-box permutation and this is lead to

- ρ (R')= ρ (S') with the same arrangement; where R' is the inverse S-box function for all S-boxes of VMS-AES. So

  $R\,'(x)=S'(x)\ \forall\ x\in\ \mathbf{X}$
- Since ρ (S) (AES Substitution function) is has been proved against differential and linear cryptanalysis, and has good diffusive properties then we can say ρ (R) has the same properties for all possible shifts. This point is the main difference between VMS-AES and the other dynamic S-box creation since the last need to construct a new inverse S-box for every new S-box which implies the need of re-evaluation and proved against cryptanalysis after every new S-box creation.
- For more clarification Figure (9) stand for the relation between the new S-boxes and their inverse in dynamic S-box construction paradigm (lines in black). While the dash lines in Blue show that VMS-AES has only one inverse S-box use with all new (Virtual) S-boxes that were generated due every permutation caused by every index shift which varies by every derived key.

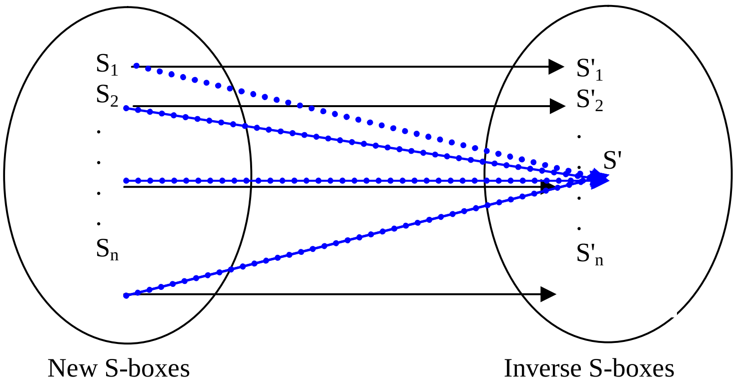


Figure (10) :The relationship between new S-boxes and Inverse s-box for dynamic construction(in black)and VMS-AES (in dash blue)

4.7 Timing and quality of service aspects.

VMS-AES require little extra time for encryption and decryption. The added operations in encryption are to the calculate of horizontal and vertical shifts then found the total shift to be added to the index before substitution process, and all these operation not contain any complex calculations like multiplication or division. Here only two XOR operations and two "+" operations and hence consume less time. Decryption process does not have any extra stage compared to AES, but one subtraction operation is carried out during the InvSubBytes function; the extra time take for this is also negligible.

5. Conclusions

VMS-AES does not contradict the security, simplicity and easy hardware implementation of AES. The security of AES was improved by employing variable mapping substitution that depends on the secret key which increases the mapping domain. As shown from the results, decreasing the number of VMS-AES rounds to be 7 instead of 10 would not affect the security and would increase the QoS for the applications that are sensitive to QoS like VoIP.

VMS-AES not need to construct a new inverse S-box for every new S-box permutation which implies there is no need to re-evaluation and proved against cryptanalysis after every new S-box creation. VMS-AES apply all different probabilities that could be used to construct a new S-box so it could be conclude that any other solution can be assumed as a special case of VMS-AES.

References

[1] D. Canright,"A Very Compact S-box for AES",Naval Postgraduate School, Monterey CA 93943, USA.

[2] B.preneel,V.rijmen,"Comments by NESSIE project on the AES finalists,NIST,May 2000

[3] Ming-Haw Jing," High Aberrance AES System Using a Reconstructable Function Core Generator",I-Shou University, TAIWAN.

[4] Faiz Yousif, Alaa Eldin Rohiem, A. Elbayoumy, "Security evaluation of VoIP cryptographic algorithms" ,Proceedings of the 6th ICEENG Conference on Electrical Engineering, 27-29 May, 2008.

[5] Susan Landau, " Communications Security for the Twenty-first Century: The Advanced Encryption Standard",Notice of the AMS 451, APRIL 2000.

[6] F. Granelli and G. Boato," A novel methodology for analysis of the computational complexity of block ciphers: Rijndael, Camellia and Shacal-2 compared", DIT - University of Trento,ive 14, I-38050 Trento (Italy).

[7] A. Fahmy, "A proposal For A key-dependent AES", Proceedings of 3rd International Conference: Sciences of Electronic, SETIT 2005, March 27-31, 2005 – TUNISIA.

[8] Rohiem, Elagooz, Dahshan H.,"Anovel approach for designing the S-box of advance ecryption standard using chaotic map", Radio science conference, NRSC May 2005.

[9] G. Boato, "Bijective S-box Applications", Trento (Italy).

[10] Bruce Schneier, "APPLIED CRYPTOGRAPHY.PDF", John Wiley & Sons ,ISBN: 0471128457, 01/01/96

[11] Alireza,Farmarz," A structure for fast data encryption",Isfahan,Iran,Int. J. Contemp. Math. Sciences, Vol. 2, 2007, no. 29, 1401 - 1424

[12] NIST, "A Statistical Test Suite for Random and Pseudorandom Generators for Cryptographic Applications", NIST Special Publication 800-22, 2003.

[13] William Stallings, "Cryptography and Network Security," Prentice Hall, 2008.

[14] Jennifer Seberry Xian, "Systematic Generation of Cryptographically Robust S-boxes", Wollongong University AUSTRALIA,May 1996.

[15] Michael D.Greenberg,"Advance engineering Mathematics",Prentice Hall,2nd edition 1998.

[16] Gareth Williams,Howard,"Finite Mathematics",WCB,1988.

[17] Howard L. Rolf,"Finite Mathematics",Thomson,6th edition,2005.