Analytical Study of AES and Proposed Variant with Enhance Block Length and Key Length

Sayed Tathir Abbas[1], Ravindera Kumar[2]

[1]Al-falah School of Engineering & Technology, Faridabad Haryana, India

[2]Al-falah School of Engineering & Technology, Faridabad Haryana, India

Abstract: ***Encryption and decryption are both methods used to ensure the secure passing of messages and other sensitive documents and information. The encryption process plays a major factor in our technology advanced lives. Encryption basically means to convert the message into code or scrambled form. Advanced Encryption Standard (AES) is a specification for the encryption of electronic data. It has been adopted by the U.S. government and is now used worldwide. AES is a symmetric-key algorithm, meaning the same key is used for both encrypting and decrypting the data. This paper defines the method to enhance the block and key length of the conventional AES.***

Keywords: Plaintext, Cipher text, key length, block length

1. Introduction

In the United States, AES was announced by National Institute of Standards and Technology (NIST) as U.S. FIPS PUB 197 (FIPS 197) on November 26, 2001 after a five-year standardization process in which fifteen competing designs were presented and evaluated before it was selected as the most suitable.AES is the first publicly accessible and open cipher approved by the National Security Agency (NSA) for top secret information when used in an NSA approved cryptographic module. AES is a symmetric block cipher with a block size of 128 bits. Key lengths can be 128 bits, 192 bits, or 256 bits called AES-128, AES-192 and AES-256 respectively. AES-128 uses 10 rounds, AES-192 uses 12 rounds, and AES-256 uses 14 rounds. In the proposed work the block length and key length are enhanced to two hundred.

2. Architecture

The architecture of this proposal is exactly same as the conventional system. The size of the input key and input data are different than conventional. Here, the block and key length has been enhanced to two hundred. Hence the data matrix used here is 5X5 matrix rather than 4X4 matrix as in conventional AES.

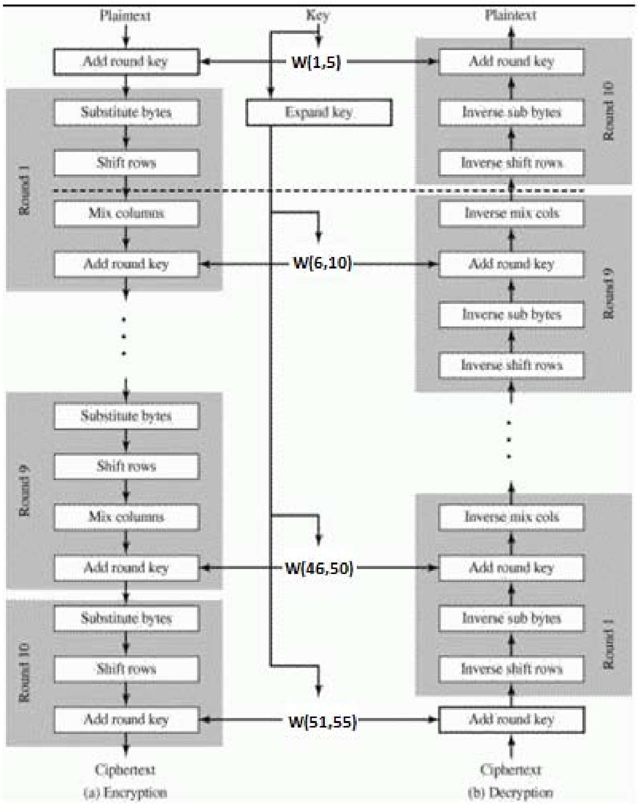

Figure 1: Architecture

3. Operations Performed

There are several transformations performed to extract the AES cipher. Various transformations are done on byte basis to build AES cipher. The operations performed here are quite different as compared to the conventional addition and multiplication. In AES the addition and multiplication are performed under the "Galois Field" operations. Galois Field is the finite field operation which is named after the brilliant young French mathematician who discovered them.

3.1 G.F Addition

The addition of two bits under GF addition is described in the Table 1.1.

Table 1: GF (2) Addition

+	0	1
0	0	1
1	1	0

Using the above mentioned table, two byte addition is done in the following manner. Given A = {a_7, a_6, a_5, a_4, a_3, a_2, a_1, a_0} added to B = {b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0} results to C = {c_7, c_6, c_5, c_4, c_3, c_2, c_1, c_0}.

C7=a7+b7	C6=A6+b6	C5=A5+b5	C4=A4+b4
C3=a3+b3	C2=A2+b2	C1=a1+b1	C0=a0+b0

3.2 G.F Multiplication

The Multiplication in this field is much more difficult and harder to understand. The first step in multiplying two field elements is to multiply their corresponding polynomials just as in algebra (except that the coefficients are only **0** or **1**, and **1 + 1 = 0**). The result would be up to a degree **14** polynomial too big to fit into one byte. A finite field now makes use of a fixed degree eight irreducible polynomial (a polynomial that cannot be factored into the product of two simpler polynomials). For the AES the polynomial used is the following

$$C(x) = x^8 + x^4 + x^3 + x + 1$$

The intermediate product of the two polynomials must be divided by **C(x)**. The remainder from this division is the desired product.

4. AES Rounds

All rounds in proposed AES are identical to the conventional one. The total transformation takes place in one AES round is explained below.

Common Round (State, Round Key)
Byte Sub (State);
Shift Row (State);
Mix Column (State);
Add Round Key (State, Round Key);

Last Round (State, Round Key)
Byte Sub (State);
Shift Row (State);
Add Round Key (State, Round Key);

4.1 Sub Byte

Sub Byte adds confusion by processing each byte through an S-Box. An S-Box is a substitution table, where one byte is substituted for another, based on a substitution algorithm. The S-Box is as

Table 2: S-Box

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xa	xb	xc	xd	xe	xf
0x	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
1x	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2x	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3x	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4x	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5x	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6x	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7x	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8x	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9x	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
ax	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
bx	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
cx	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
dx	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
ex	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
fx	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

To complete an S-Box operation on an example string of "ABC," take the hexadecimal value of each byte. ASCII "A" = hex 0x42, "B" = 0x43 and "C" = 0x44. Look up the first (left) hex digit in the S-Box column and the second in the S-Box row. 0x42 becomes 0x2c; 0x43 becomes 0x1a, and 0x44 becomes 0x1b.

4.2 Shift Row

Shift Row provides diffusion by mixing data within rows. Row zero of the State is not shifted, row 1 is shifted 1 byte, row 2 is shifted 2 bytes, and row 3 is shifted 3 bytes, as shown in the fig 2.

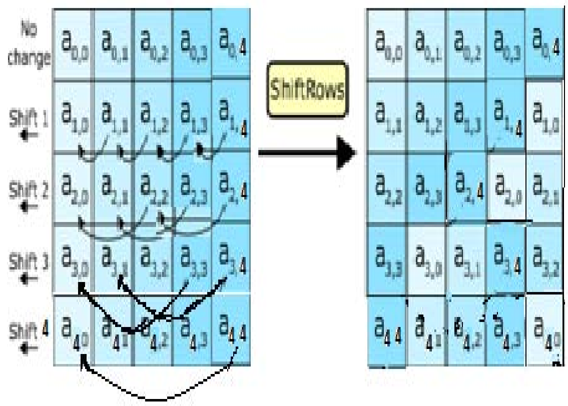

Figure 2: Illustration of Shift Row transformation

4.3 Mix Column

Mix column provides diffusion by mixing data within columns. The 4 bytes of each column in the State are treated as a 4-byte number and transformed to another 4-byte number via finite field mathematics, as shown in fig 3. Here c(x) is the polynomial which is given as

$$C(x) = x^8 + x^4 + x^3 + x + 1$$

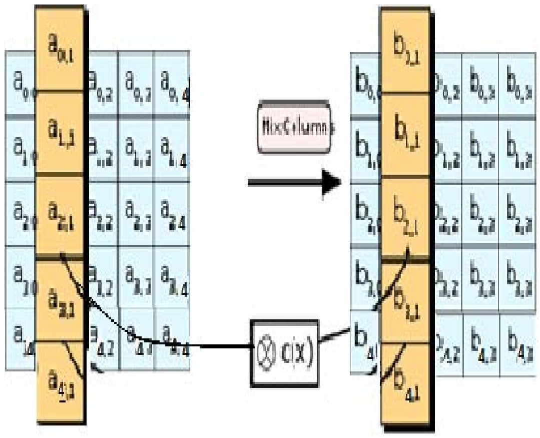

Figure 3: Mix Column transformation

4.4 Add Round Key

In this operation, a Round Key is applied to the State by a simple bitwise EX-OR. The Round Key is derived from the Cipher Key by means of the key schedule. The transformation that consists of EXOR a Round Key to the State is denoted by: Add Round Key (State, Round Key).

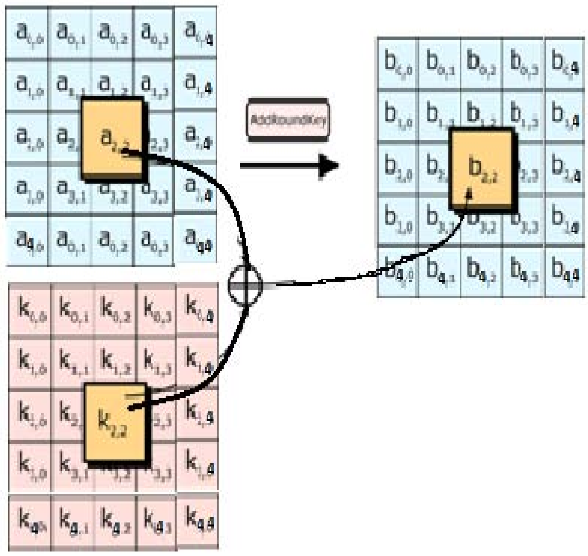

Figure 4: Round Key Addition

5. Conclusion

The main application of proposed system is in the places where we can vary the block size according to our need or channel. As we know that any complemented bit in received signal than to transmitted may cause the whole block to vanish. So, we can reduce and increase the block size according to our channel, which can also be named as opportunistic encryption. The proposed model has enhanced the block size as compared to conventional 128 bits. The proposed algorithm can be used in Geographic Information System (GIS) and Satellite Communication when huge data need to be transferred securely. AES with its all variants and proposed model is implemented in Matlab.

References

[1] A.S.Tanenbaum, Computer Networks, Fourth Edition PHI

[2] B.A.Forouzan, Data communication and Networking, Fourth Edition McGraw Hill

[3] Stalling W. Cryptography and Network Security Third Edition Pearson Education.

[4] The Laws of cryptography The finite field GF(2^8) by Neal R. Wagner 2001.

[5] Advanced Encryption Standard Eric Conrad

[6] Schneier B. and Whiting D., Performance Comparison of AES Finalist, 2000

[7] C. Shannon, Communication theory of secrecy systems, Bell Systems Technical Journal, vol. 28, 1949

[8] J. Daemen and V. Rijmen, The Design of Rijndael: AES – The Advanced Encryption Standard. Springer-Verlag, 2002

Authors Profile

Sayed Tathir Abbas received the degree B.Tech. in Electronics and communication Engineering from Veera College of Engineering in 2009 and Pursuing M.Tech in Electronics and communication from Al-Falah School of Engineering and Technology. He has worked in Dr. K. N. Modi Institute of Engineering and Technology in 2009 and joined R.V.Insitute of Technology in 2010.

Ravinder Kumar is working as an Assistant Professor in Al-Falah School of Engineering and Technology**.** He received the degree of B.Tech. and M.Tech in Electronics and communication in 2004 and 2010 respectively. He has more than 9 years of teaching and research experience.