A NOVEL APPROACH FOR AES ALGORITHM IN IMAGE ENCRYPTİON

Sai Charan Dhatrika [1], Deepika Puvvula [2], S.Venu Gopal [3]

[1,3] *Assistant Professor, Dept of CSE, Bharat Institute of Engineering & Technology (BIET), Ibrahimpatnam, (India)*

[2] *Assistant Professor, Dept of CSE, ANITS Engineering College, Visakhapatnam, (India)*

ABSTRACT

Today with the tremendous development of various technologies like multimedia, research on security is becoming more important. In providing security Cryptography places a very crucial role. Even though there are many cryptographic algorithms to provide security, they are not up to the satisfactory level of the users. So there was a need for research on inventing new algorithms or modifying existing algorithms. In this paper we proposed an enhanced AES algorithm for image encryption which can be used to encrypt using AES-128 bit key. The proposed modifications in this paper are: repositioning the image pixels to break the correlation between them, randomization of key and hiding the key value into the encrypted digital image. So the proposed method provides more security.

Keywords: *Advanced Encryption Standard, Image Encryption, Key hide.*

I. INTRODUCTION

With the rapid development of multimedia technology multimedia data like images, videos, audios are used in various applications like entertainments, education, advertisements, and politics. There are different types of encryption algorithms available like AES, DES, Blowfish etc[1]. These algorithms are very good at encrypting text data but coming to multimedia data these data are large in volumes and also there is high redundancy. For example, the image shown in Fig:1(a) below is encrypted by AES algorithm directly(ECB mode) and the resultant image is shown in Fig1(b). We can say that Fig1(b) is still intelligible.

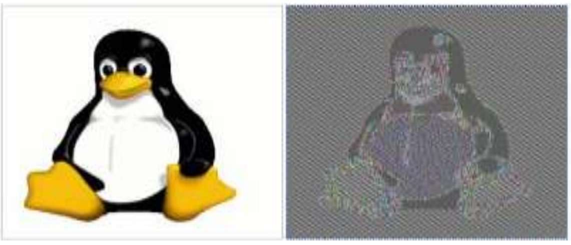


Figure 1: (a) Original image (b) Cipher image

Hence the security is low. This is because the correlation between the adjacent pixels in an image cannot be break by AES algorithm. In real time applications we need better encryption algorithm so we go for new encryption algorithms or modification to existing algorithms. In this paper we introduced a new encryption algorithm as a modification to AES algorithm. The modification is mainly focused on breaking the correlation between the image pixels by shifting pixel position, randomization of key and hiding the key into the encrypted digital image. For multimedia data the correlation between the image pixels is too high, AES cannot break this relation between pixels. In our enhanced AES algorithm we break the correlation between the pixels by shifting the pixel position row wise and column wise. In our proposed method we randomize the key values also.

II. ADVANCE ENCRYPTION STANDARD

The Advanced Encryption Standard (AES) was published by NIST (National institute of standards and Technology) in 2001.AES is a Symmetric block cipher intended to replace DES for commercial applications [7]. It uses a 128-bit block size and a key size of 128,192 or 256 bits, this standard is based on the Rijndael algorithm, a symmetric block cipher. The AES algorithm used three different key lengths; these three are referred to as “AES-128”, “AES-192” and “AES-256”.The AES algorithm is divided into four different steps, which are executed in a sequential manner by forming rounds. Depending upon the key length the number of rounds will vary. The below figure shows the parameters of AES algorithm.

2.1 AES Parameters

Key size(word/bytes/bits)	4/16/128	6/24/192	8/32/256
Plain text block size (words/bytes/bits)	4/16/128	4/16/128	4/16/128
Number of rounds	10	12	14
Round key size (words/bytes/bits)	4/16/128	4/16/128	4/16/128
Expanded key size (word/bytes)	44/176	52/208	60/240

Figure 2: AES parameters

As I said earlier AES is a Symmetric block cipher, which means that: 1) AES works by repeating the same defined steps multiple times. 2) AES is a secret key algorithm.3) AES operates on a fixed number of bytes. AES as well as most encryption algorithms are reversible. This means that almost the same steps are performed to complete both encryption and decryption in reverse order. The key we are using in this algorithm is expanded into individual sub keys, one sub key per each round. This process is called Key Expansion. The operations performed on fixed number of bytes in AES algorithm are classified as below:

- ADD ROUND KEY
- BYTES SUBSTITUTION
- SHIFT ROWS
- MIX COLUMN

Here I discussed all these operations one by one. At first I discussed the key expansion of AES algorithm.

2.1.1 Key Expansion

The AES algorithm takes a fixed key K, and performs a key Expansion routine by using Rijndael’s key schedule to generate a key schedule[8]. When the key length is 128 bit, then the Key Expansion generates a total 11 sub-key arrays of 128 bits, denoted Wi and the first sub-key is the initial key. We need previous sub-key, two tables, RCon and S-Box to generate the sub-keys.

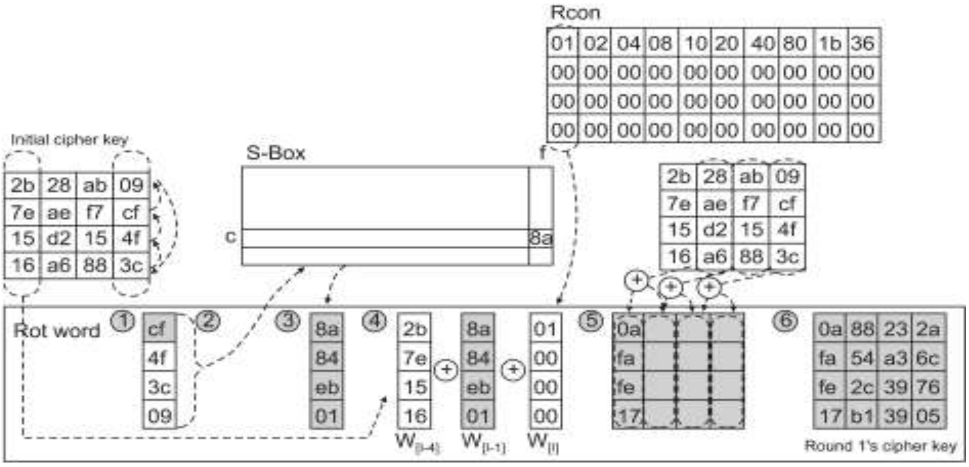


Figure 3: Key expansion operation

2.1.2 Add Round Key

In the AddRoundkey step, the sub-key is combined with the state. For each round, a sub-key is derived from the fixed key by using Rijndael's key schedule; size of the each sub-key is 128 bits. The sub-key is added by combining each byte of the state with the corresponding byte of the sub-key using bitwise XOR as shown in below Figure 4:

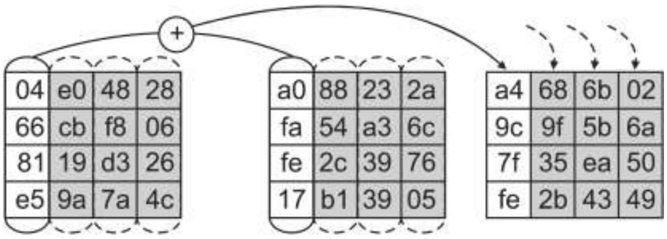
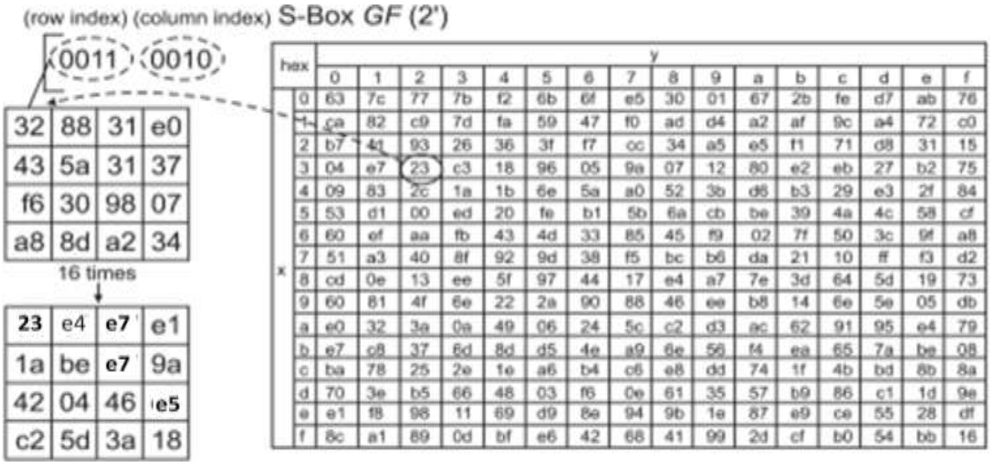


Figure 4:Add Round Key operation.

2.1.3 Sub Bytes

In the SubBytes step, each byte in the matrix is replaced with a SubByte using S-Box. This operation provides the non linearity in the cipher. The S-Box is derived from the multiplicative inverse over **GF**(2^8), known to have good non-linearity properties. For example the state matrix value represents the row and column indexes of S-box. Here in the below figure state matrix value 32 represents the value of S-box at 3rd row and 2nd column, so it substitutes 32 with 23.



hex	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
0	63	7c	77	7b	f2	6b	6f	e5	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	4f	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	e7	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	fe	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	60	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	0e	13	ee	5f	97	44	17	e4	a7	7e	3d	64	5d	19	73
9	60	81	4f	6e	22	2a	90	88	46	ee	b8	14	6e	5e	05	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	6e	56	f4	ea	65	7a	be	08
c	ba	78	25	2e	1e	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	cf	b0	54	bb	16

Figure 5: Sub Bytes operation

2.1.4 Shift Rows

The Shift Rows step operates on the rows of the states; it cyclically shifts the bytes in each row by a certain offset. The first row is left unchanged. Each byte in the three rows of the states is cyclically shifted over 1, 2 and 3 bytes respectively as shown below in Figure 5:

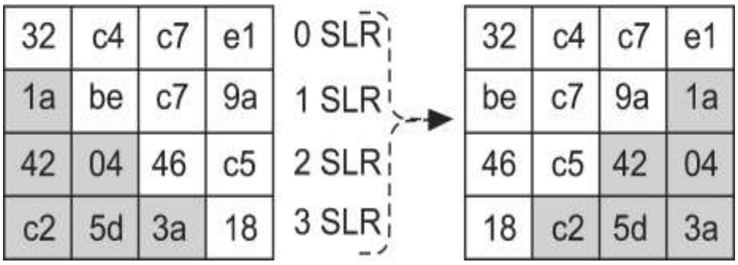


Figure 6: Shift Rows operation.

E: Mix Columns

In the MixColumns Step, the four bytes of each column of the state are combined using an invertible linear transform. It takes four bytes as input and outputs four bytes, where each input byte affects all four output bytes shown in Fig. 5. During this operation, each column is multiplied by the known matrix that is:

$$\begin{bmatrix} 2 & 3 & 1 & 1 \\ 1 & 2 & 3 & 1 \\ 1 & 1 & 2 & 3 \\ 3 & 1 & 1 & 2 \end{bmatrix}$$

Each column is treated as polynomials over **GF**(2^8) and then multiplied by a fixed polynomial c(x) modulo x^4+1 given by

$$\mathbf{C(x) = \{03\}x^3 + \{01\}x^2 + \{01\}x + \{02\}}$$

The coefficients are displayed in their hexadecimal equivalent of the binary representation. In the MixColumns Step, each column of the state is multiplied with a fixed polynomial c(x).

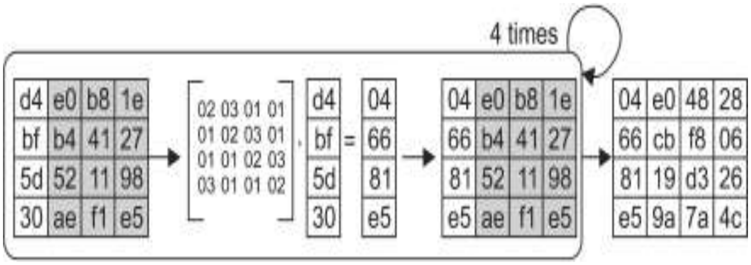


Figure 7: Mix Column Operation

III. LITERATURE SURVEY

3.1 High Definition Image Encryption Algorithm Based on AES Modification, 2014

Salim Muhsin Wadi and Nasharuddin Zainal analyze the Advanced Encryption Standard and in their image encryption technique they add two modifications to AES algorithm to improve the performance and decreasing the hardware requirements. First modification was conducted using MixColumn transformation in 5 rounds instead of 10 rounds, and the second modification was instead of using S-box and Inverse S-box as in original AES algorithm they used only one simple S-box for encryption and decryption.

3.2 Modified Advanced Encryption Standard, 2014

Pravin Kawle, Avinash Hiwase, Gautam Bagde, Ekant Tekam and Rahul Kalbande analyze the Advanced Encryption Standard (AES) and modify it, to reduce the calculation of algorithm and for improving the encryption performance. In modified AES algorithm instead of using Mixcolumn they use permutation on data. Modified-AES algorithm is a fast lightweight encryption algorithm for security of multimedia data. All above advantages make algorithm highly suitable for the images and plaintext transfer as well, than the AES algorithm.

3.3 Enhanced Image Encryption Techniques Using Modified Advanced Encryption Standard, 2012

Faisal Riaz, Sumira Hameed, Imran Shafi, Rakshanada Kausar and Anil Ahmed study the AES algorithm and provide a modification to the Existing AES algorithm to increase the speed of the AES algorithm they proposed Selective Image Encryption technique.

3.4 A new modified version of Advanced Encryption Standard based algorithm for image encryption, 2010

S.H.Kamali, R.Shakerian M.Hedayati and M.Rahmani analyze and present a modification to the Advanced Encryption Standard (MAES) to reflect a high level security and better image encryption. The modification is done by adjusting the ShiftRow Transformation. Detailed results in terms of security analysis and implementation are given. Experimental results verify and prove that the proposed modification to image cryptosystem is highly secure from the cryptographic viewpoint. The results also prove that with a comparison to original AES encryption algorithm the modified algorithm gives better encryption results in terms of security against statistical attacks.

3.5 A Modified AES Based Algorithm for Image Encryption, 2007

M.Zeghid, M.Machhout, L.Khriji, A.Baganne, and R.Tourki analyze the Advanced Encryption Standard (AES), and add a key stream generator (A5/1, W7) to AES to ensure improving the encryption performance; mainly for images characterized by reduced entropy.

IV. PROPOSED METHOD

In our proposed algorithm, we just change the AES to be more efficient and secure. For multimedia image the correlation values are too high which cannot be removed by AES algorithm. So we need to break this close relation between these adjacent pixels. It can be achieved by repositioning the pixel values. So we shifted the pixel values row wise and column wise. This shifting operation is done so many times and we get our desired cipher image which is not intelligible. In our proposed algorithm we by randomized the key values and shifted the pixel values. Here we generate key values depending on the mouse position on the screen. When the key length size is 128bits, at that moment we need 16 values from the mouse position on the screen. So, we take 8 mouse position values. In one position there is x position value and y position value. We derived this mouse position values and we get our desired key values. Then we have done our encryption by using these key values In our proposed method we randomized the key value which is essential part of the decryption. If we do not know the key value then we cannot decrypt the cipher image. So we need to send this key value with encrypted image. So with this key value and cipher image we generate a new image which is our desired cipher image. To send the key along with encrypted image we use some techniques like digital watermarking and cryptographic techniques.

4.1 DETAILED ALGORITHM

4.1.1 Sender Side Algorithm

STEP 1: The image file is taken as the input.

STEP 2: The pixels are right shifted along row wise and column wise to break the correlation between the adjacent pixels and thus the 1st level cipher image can be obtained.

STEP 3: Key value is generated randomly.

STEP 4: A key Expansion routine is to be performed by using Rijndael's key schedule to generate a key schedule. When the key length is 128 bit, then the Key Expansion generates a total 11 sub-key arrays of 128 bits, denoted Wi and the first sub-key is the initial key.

STEP 5: The sub-key is combined with the state. This step is known as AddRouudkey Step. For each round, a sub-key is derived from the fixed key by using Rijndael's key schedule. Size of the each sub-key is of 128 bits.

STEP 6: Each byte in the matrix is replaced with a SubByte using S-Box. This operation provides the non linearity in the cipher. The S-Box is derived from the multiplicative inverse over **GF**(2^8) to have good non-linearity properties.

STEP 7: The bytes in each row are cyclically left shifted by a certain offset where the first row is left unchanged.

STEP 8: The four bytes of each column of the state are combined using an invertible linear transform. It takes four bytes as input and outputs four bytes, where each input byte affects all four output bytes.

STEP 9: The AddRouudkey Step operation is again repeated.

STEP 10: Step 6 to Step 9 operations are to be repeated for eight times and in the final round step 6, step 7 and step 9 operations will be repeated.

STEP 11: The second level cipher image is then generated by using the key values and the 1st level cipher image, which can be transferred through public-purpose communication channel.

4.1.2 Receiver Side Algorithm

STEP 1: Upon receiving the cipher image the decryption or extraction process can be started. At first, from the received 2nd level cipher image the key value to be extracted.

STEP 2: A key Expansion routine is to be performed by using Rijndael's key schedule to generate a key schedule. When the key length is 128 bit, then the Key Expansion generates a total 11 sub-key arrays of 128 bits, denoted Wi and the first sub-key is the initial key.

STEP 3: The sub-key is combined with the state. This step is known as Inverse AddRouudkey Step. For each round, a sub-key is derived from the fixed key by using Rijndael's key schedule. Size of the each sub-key is of 128 bits.

STEP 4: The bytes in each row are cyclically right shifted by a certain offset where the first row is left unchanged.

STEP 5: Each byte in the matrix is replaced with a SubByte using S-Box. This operation provides the non linearity in the cipher. The S-Box is derived from the multiplicative inverse over **GF**(2^8) to have good non-linearity properties.

STEP 6: The Inverse AddRouudkey Step operation is again repeated.

STEP 7: The four bytes of each column of the state are combined using an invertible linear transform. It takes four bytes as input and outputs four bytes, where each input byte affects all four output bytes.

STEP 8: Step 4 to Step 7 operations are to be repeated for eight times and in the final round, step 4, step 5 and step 6 operations will be repeated.

STEP 9: The pixel values are left shifted along row wise and column wise for repositioning the pixel values.

STEP 10: Finally the original image can be obtained.

V. RESULTS

Besides giving high authentication ability and good robustness, this proposed scheme provides good recoverability. If we use our proposed method we get the results as shown below .Here the cipher images are totally invisible. The decrypted images are shown as below in Figure8:

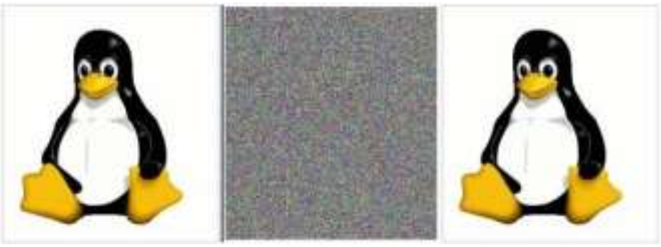


Figure 8 a) original image b)cipher image c)decrypted image

VI. CONCLUSION

In this paper we modified the version of AES and we proposed a new algorithm. The modification is done by randomizing the key values and repositioning the pixel values. We have shown that the proposed cryptosystem gives better encryption results in terms of security against statistical attacks. Even though it gives good security against statistical analysis it takes more time. So we propose that to reduce the time complexity one should reduce the number of rounds in AES algorithm.

REFERENCES

[1] Subijit Mondal, Subhashis Maitra, "Data security-modified AES algorithm and its applications", ACM SIGARCH Computer Architecture News,volume-42,issue-2, 2014.

[2] Salim Muhsin Wadi, Nasharuddin Zainal, "High Definition Image Encryption Algorithm Based on AES Modification", Wireless Personal Communications,volume-79,issue-2, 2014.

[3] Pravin Kawle, Avinash Hiwase, Gautam Bagde, Ekant Tekam, Rahul Kalbande, “Modified Advanced Encryption Standard”, International Journal of Soft Computing and Engineering (IJSCE), volume-4,issue-1, 2014.

[4] Faisal Riaz, Sumira Hameed, Imran Shafi, Rakshanada Kausar, Anil Ahmed,Enhanced Image Encryption Techniques Using Modified Advanced Encryption Standard”, Emerging Trends and Applications in Information Communication Technologies Communications in Computer and Information Science.volume-281, 2012.

[5] Kamali, S.H. ; Qazvin Branch, Islamic Azad Univ., Qazvin, Iran ; Shakerian, R. ; Hedayati, M. ; Rahmani, M,”A new modified version of Advanced Encryption Standard based algorithm for image encryption”, Electronics and Information Engineering (ICEIE), International Conference On volume-1.2010.

[6] Zeghid, M.; Machhout, M.; Khriji, L.; Baganne, A.; Tourki, R. “A Modified AES Based Algorithm for Image Encryption**”,** International Journal of Computer Science & Engineering. volume-1,issue-1,2007.

[7] Abdulkarim Amer Shtewi, Bahaa Eldin M.Hasan, Abd El Farat.A.Hegazy,”An Efficient Modified Advanced Encryption Standard (MAES) Adapted for Image Cryptosystems” International Journal of Computer and Network Security, VOL.10No.2, February 2010,p 226-232.

[8] http://csrc.nist.gov/archive/aes/rijndael/Rijndael-ammended.pdf.