International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 06, June-2015

Security Enhanced Image Encryption using Password based AES Algorithm

Nandu Krishnan J
M. Tech Scholar,
Dept. of Electronics and Communication Engineering,
Mar Baselios College of Engineering and Technology,
Thiruvananthapuram, Kerala, India.

Roshny G. Kumar
Assistant Professor,
Dept. of Electronics and Communication Engineering,
Mar Baselios College of Engineering and Technology,
Thiruvananthapuram, Kerala, India.

***Abstract*—Encryption plays an important role in ensuring the security of the information being used. Nowadays information in the form of digital image is being used widely in many fields and needs to be protected from unauthorized recipients. For this purpose cryptography gives certain standard techniques which gives better security and performance to the systems for handling such confidential data. This work is based on the implementation of digital image encryption using Advanced Encryption Standard Algorithm (AES 128 bit) defined by National Institute of standard and technology (NIST) of United States. The main focus is to improve the security of the image being sent by preventing outside attacks. The algorithm is successfully implemented and simulated in MATLAB. The security analysis of the proposed method is studied through Histogram assessment, Key Sensitivity test and Password sensitivity test.**

***Index Terms*—AES, FIPS, Rcon, DES, ASCII, MATLAB,GF**

I. INTRODUCTION

Cryptography plays an important role in many electronic systems, to ensure the security of the confidential data being used especially when the medium used for the communication is unreliable and error prone. As a result the data needs to be encrypted. The raw data called plain text is transformed into a secret code called cipher text which is in an unintelligible form that will not be easily accessible by unintended recipients. Decryption is the reverse process where the encrypted data is converted back into its original form by using the same key that was used at the time of encryption. Thus the security and integrity of the system can be ensured.

As digital images play an important role in many fields like internet communication, medical imaging, military imaging systems, satellite imaging etc, it is essential to protect them from outside attacks for their reliable storage and transmission. The aim of this work is to implement the widely accepted Advanced Encryption Standard (AES) algorithm for digital image encryption. The main focus is to improve the security level and to analyze the proposed encryption scheme.

II. AES ALGORITHM

Advanced Encryption Standard (AES) is an approved cryptographic algorithm that can protect electronic data. AES is a type of symmetric key block cipher based on several rounds. There will be 10, 12, or 14 rounds, when the key length is 128, 192 or 256 bits, respectively. During encryption each round performs four transformations: Sub Bytes, Shift Rows, Mix Columns and Add Round Key, while the final round does not have the Mix Columns transformation [1].

A. Sub Bytes Transformation

This transformation involves a byte to byte nonlinear substitution where the substitute byte is obtained from a 16 x 16 look up table known as Substitution Box (S Box).To find the substitute byte for a given input byte:

- Input byte needs to be divided into two 4-bit patterns and find the corresponding integer value between 0 and 15.
- Represent these by their hex values 0 through F.
- One of the hex values is used as a row index and the other as a column index.
- From the S box lookup table, find the substitute byte after locating the corresponding row index and column index.
- Replace the corresponding data byte with the substitute byte.

B. Shift Rows Transformation

The transformation is made to the incoming state array in such a way that the first row of the state array is not shifted, second row is circularly shifted to the left by one byte, third row is circularly shifted to the left by two bytes and the last row is circularly shifted by three bytes to the left.

C. Mix Column Transformation

The transformation operates on the State matrix column-by-column individually. Here each byte of a column is replaced by a function of all the bytes in the same column as two times that byte, plus three times the next byte, plus the byte that comes next, plus the byte that follows. In simple matrix form this operation can be represented as [2]:

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} S_{0,0} & S_{0,1} & S_{0,2} & S_{0,3} \\ S_{1,0} & S_{1,1} & S_{1,2} & S_{1,3} \\ S_{2,0} & S_{2,1} & S_{2,2} & S_{2,3} \\ S_{3,0} & S_{3,1} & S_{3,2} & S_{3,3} \end{bmatrix} = \begin{bmatrix} S'_{0,0} & S'_{0,1} & S'_{0,2} & S'_{0,3} \\ S'_{1,0} & S'_{1,1} & S'_{1,2} & S'_{1,3} \\ S'_{2,0} & S'_{2,1} & S'_{2,2} & S'_{2,3} \\ S'_{3,0} & S'_{3,1} & S'_{3,2} & S'_{3,3} \end{bmatrix} \quad (1)$$

Here the additions and multiplications are performed in GF (2^8).

D. Add Round Key Transformation

This transformation is a simple bit wise XOR operation between the state matrix and the corresponding round key generated from the key scheduling module. This operation is also performed in decryption stage where each round performs four transformations: Inverse Shift Rows,

Inverse Sub Bytes, Inverse Mix Columns and Add Round Key, and the last round do not have the Inverse Mix Columns transformation.

E. Inverse Sub Bytes Transformation

This transformation step operates in a similar way to Sub Bytes Transformation step. The substitute byte is obtained from a 16 x 16 look up table known as Inverse Substitution Box (Inv S Box).

F. Inverse Shift Row Transformation

Here the transformation performs the reverse operations done on Shift Row transformation where the first row of the state array is not shifted, second row is circularly shifted to the right by one byte, third row is circularly shifted to the right by two bytes and last row is circularly shifted by three bytes to the right.

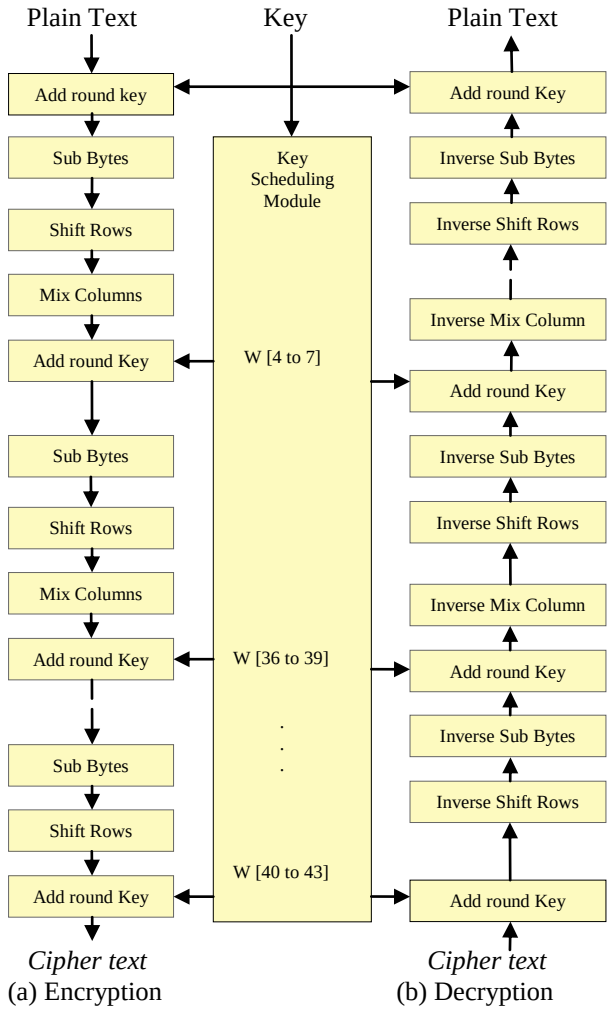


Fig.1.AES Encryption/Decryption procedure

G. Inverse Mix Column Transformation

The Inverse Mix Column transformation operates on the State matrix column-by-column individually. In simple matrix form this stage can be represented as [2]:

$$\begin{bmatrix} 0E & 0B & 0D & 09 \\ 09 & 0E & 0B & 0D \\ 0D & 09 & 0E & 0B \\ 0B & 0D & 09 & 0E \end{bmatrix} \begin{bmatrix} S_{0,0} & S_{0,1} & S_{0,2} & S_{0,3} \\ S_{1,0} & S_{1,1} & S_{1,2} & S_{1,3} \\ S_{2,0} & S_{2,1} & S_{2,2} & S_{2,3} \\ S_{3,0} & S_{3,1} & S_{3,2} & S_{3,3} \end{bmatrix} = \begin{bmatrix} S'_{0,0} & S'_{0,1} & S'_{0,2} & S'_{0,3} \\ S'_{1,0} & S'_{1,1} & S'_{1,2} & S'_{1,3} \\ S'_{2,0} & S'_{2,1} & S'_{2,2} & S'_{2,3} \\ S'_{3,0} & S'_{3,1} & S'_{3,2} & S'_{3,3} \end{bmatrix} \quad (2)$$

Here the additions and multiplications are performed in GF (2^8).

H. Key Scheduling Module

In this module 44 words are produced using the 128 bit input key, and gives each 4 word key to the concerned add round key function as shown in Fig.1 .The four columns of the initial state matrix forms 4 words [w_0 to w_3].This set of 4 words [w_0 to w_3] is used to generate all the remaining words. This process is known as key expansion. For each iteration, the last word is given to a function named g where the following transformations take place.

- Rot Word where a one byte circular shift on a word is performed.
- Sub Word where a Sub Bytes transformation is made on the incoming bytes.
- Results of step 1 and 2 are XORed with the round constant. The round constant is a word which is defined as Rcon (k).

Rcon (k) = (RC (K), 0, 0, 0); RC (1) =1; RC (k) =2∗ RC(k − 1)

III. Proposed Method of Image Encryption

A cryptographic system should be designed in such a way that no outsiders can attack the system. But a major attack that has been reported to many complex crypto systems is Brute force attack which is a kind of key guessing attack which tries to find the original key by trying all possible combinations of letters, numbers and symbols till the correct combination is obtained.

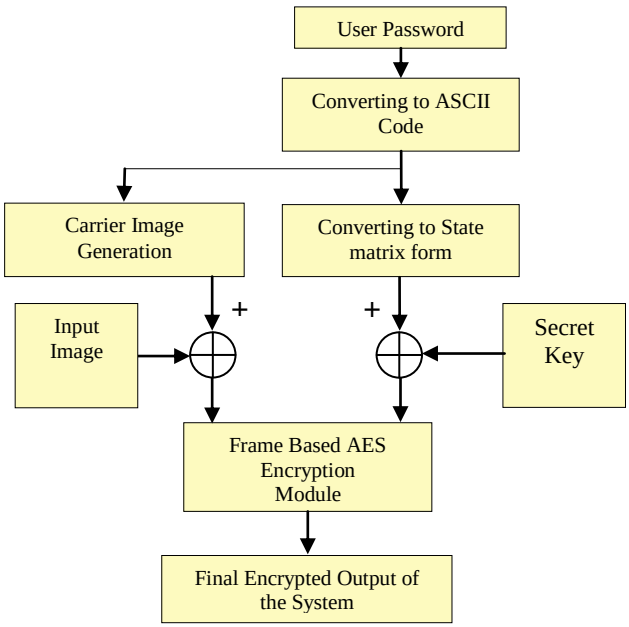


Fig.2. Proposed Encryption scheme

Larger the key size, more the time and combinations needed to crack the system. It has been reported that the Data Encryption Standard (DES) faced this attack some years ago, and was the reason for replacing DES with AES as it supports larger key sizes.

Since AES algorithm uses a key length of 128 bits, the possible number of combinations for the key searching will be 2^{128} which is equal to 3.4 x10 ^{38}. Here we can see the exponential increase in possible combinations compared to the 56 bit key used in DES. Even with a super computer, it will take 1 billion years to crack the 128 bit AES key using brute force attack [3].So the security of AES is more than DES. This work is aimed to improve the security level of existing AES (128 bit) by incorporating a secret password based processing along with the usual way of AES encryption of 128 bit Data and Key. American Standard Code for Information Interchange (ASCII) is the character encoding technique where each character is mapped into a numerical value as described in the standard ASCII table. As the user password contains characters as well as numbers the ASCII method of encoding can be effectively used in this work to get their equivalent numerical values. The ASCII numerical equivalent of the user password is then used for two purposes.

- Carrier Image Generation.
- Mixing with the original input key.

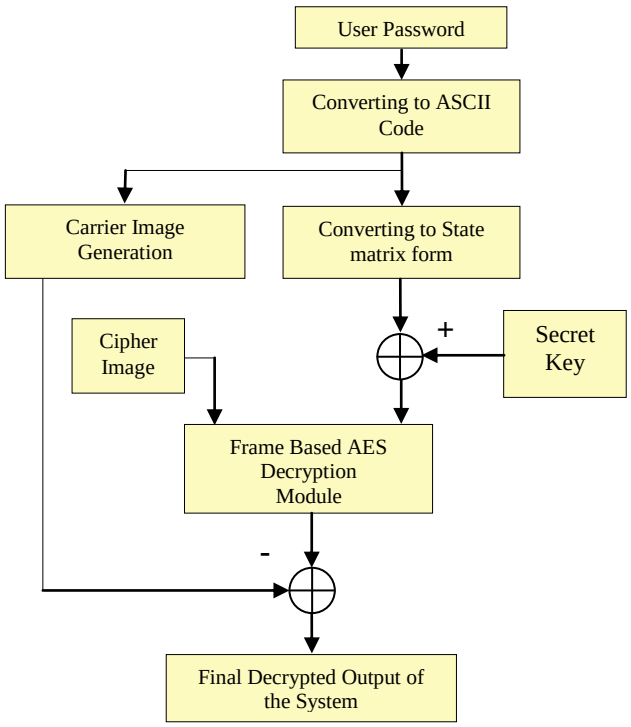
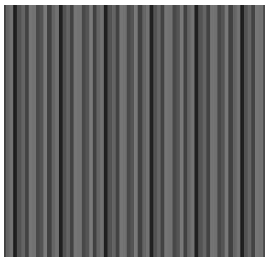


Fig.3. Proposed Decryption Scheme

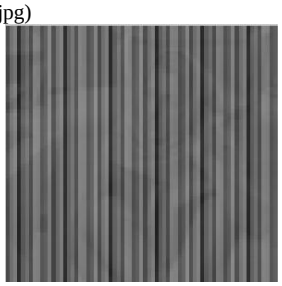
Once the password is entered, the ASCII equivalent vector equal to the length of the password is computed and is rearranged in the form of a matrix equal to the size of original image. If the length of the vector is less than the size of the input image then the same vector is repeated until the length becomes equal to size of original image. And finally the carrier image is created. This carrier image is then mixed with the input image in order to hide the originality of the image which is to be encrypted. The ASCII vector generated initially is also used for mixing with the original key. This is mainly done to make the encryption key sensitive to the secret password. If the length of the vector is less than 16 then the same vector is repeated until the length is become equal to 16 i.e. 16 bytes. The AES Encryption Module now accepts a carrier mixed image and a 128 bit mixed key as input. Since AES Algorithm is standardized to have only 128 bits of data, the image is divided into different frames of size equal to that of a state matrix. So each 4 x 4 frame will be sent to the encryption process and the corresponding results are stored inside another array. After the whole process is completed the final encrypted image is obtained as the output.

IV. Simulation Results

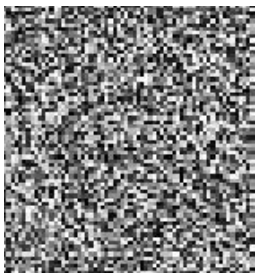
(a) Test Image 1 (barbara.jpg)



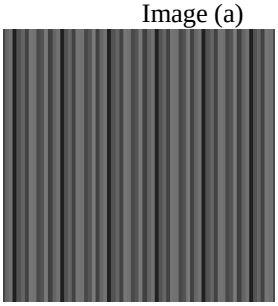
(b) Carrier image for the Password‘Aes SeCurITy'



(c) Carrier mixed with original Image (a)



(d) Encrypted image of (c)



(e) Carrier Image at Decryption using original password

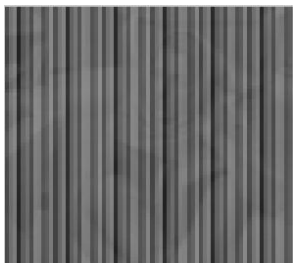


Image after AES Decryption of (d)

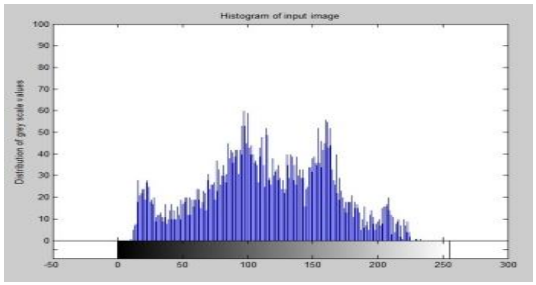
(f)

(g) Final Decryption after removing carrier

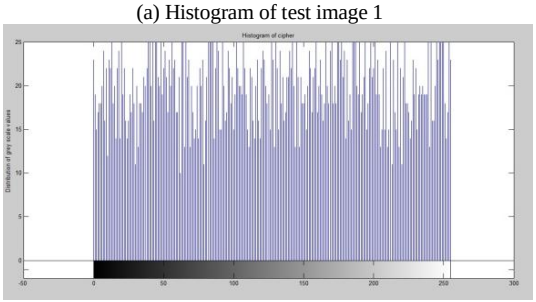
Fig.4.Test Results for the proposed scheme

Key used is 13FD6A6A4505998D58AABA2E335F331F.
Secret password used is ‘Aes SeCurITy'

The simulation of the proposed algorithm is carried out in MATLAB. The test image used is barbara.jpg. The user password can be of any length and can include numbers, characters or symbols. Once the password is entered, the ASCII vector of the same will be repeated continuously until the length of the vector becomes equal to the size of the test image. Thus carrier image is produced as seen in Fig.4.(b) .The ASCII vector of the password is mixed with the input key and the new key is used for AES encryption. Then the carrier image is added with the original image. From Fig .4. (c), it is clear that the originality of the test image is hidden inside the carrier image. This image is then given to the encryption unit with the new modified key for final encryption. The final output will be as shown in Fig.4. (d).The decryption involves the reverse operation in chronological order, where the AES decryption is done with the mixed key and then subtracting the carrier from the resulting image. The same testing is done for another image cameraman.tif and the results are observed. Thus if an attack in the form of repeated key guessing (brute force attack)occurs, the attacker gets back the image (c) from the transmitted cipher (d) using one particular computer generated key combination, but the importance of this proposed method is that the originality of the attacker image is hidden by the password and the attacker's attempt fails. So in this proposed system the time to crack is increased compared to the normal time needed to crack the AES system, without making any change in the length of the key to be used. Thus security of the system is more.



(a) Histogram of test image 1



(b) Histogram of cipher of test image1

Fig.5.Histograms obtained for the proposed scheme

V. SECURITY ANALYSIS

A. Histogram Assessment

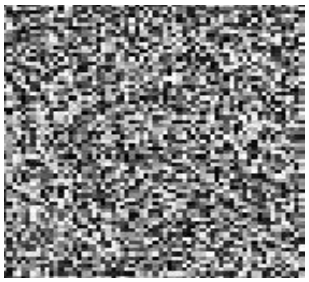
The histogram of the encrypted image as seen in the Fig.5 (b) is fairly uniform and is significantly different from the histogram of the original image as seen in the Fig.5 (a). Therefore, there is no chance of any statistical attack on the image which is considered and no loss in image quality after the encryption and decryption.

B. Key Sensitivity Test

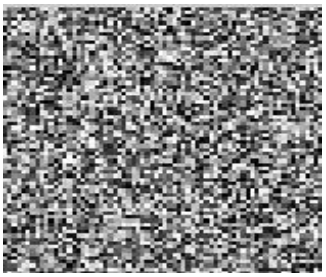
Key Sensitivity is one of the major parameters in ensuring the security of any cryptosystem. A small change in the key should bring about a significant or large change in output.

- Key (K1) is: 13FD6A6A4505998D58AABA2E335F331F.
- Key (K2) is: 03FD6A6A4505998D58AABA2E335F331F. (1bit change compared to K1)
- Secret password used is: 'Aes SeCurITy'

(a) Test Image 1



(b) Cipher image using K1



(c) Decrypted image using K2 (d) Difference between b and c

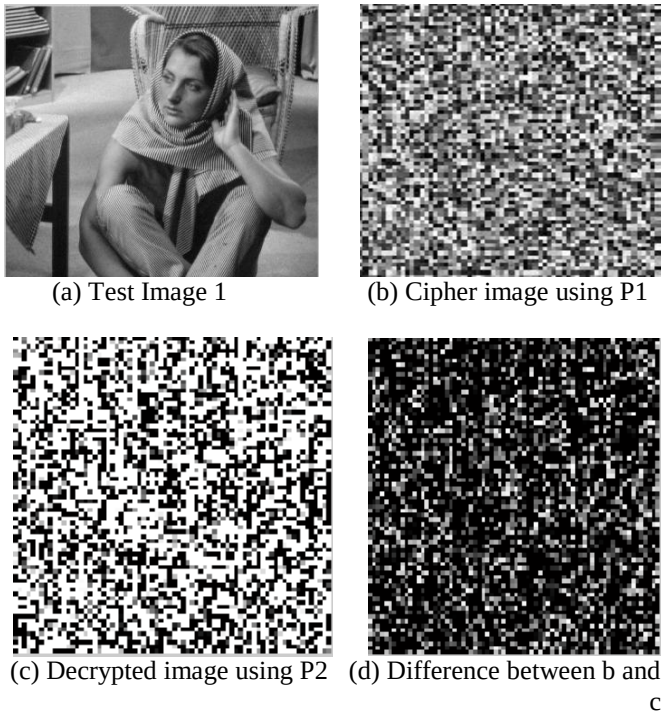
Fig.6.Key sensitivity test for proposed scheme

From Fig.6. it is clear that image decrypted using 1 bit change in the original key is not at all giving any true information about the original image. The difference between the actual image to be obtained at decryption and the image obtained after decryption with 1 bit changed key differs in pixels by 99.7%.So security is assured for the proposed algorithm as key sensitivity is high.

C. Password Sensitivity Test

A small change in the password should also bring out a significant change in output.

- Key is: 13FD6A6A4505998D58AABA2E335F331F
- Secret password P1 is: ‘Aes SeCurITy'
- Secret password P2 is: ‘aes SeCurITy'



(a) Test Image 1 (b) Cipher image using P1

(c) Decrypted image using P2 (d) Difference between b and c

Fig.7.Password sensitivity test results for the proposed scheme

From Fig.7. it is clear that image decrypted using a slight change in the original password is not at all giving any true information about the original image. The difference between the image to be obtained at decryption and the image obtained after decryption with slight password change differs in pixels by 99.96%.So security is assured for the proposed algorithm as password sensitivity is also high.

VI. Conclusion

A new Password based image encryption method using the Advanced Encryption Standard (AES) algorithm is proposed. It is very essential to secure the data being used especially when it is confidential in nature.AES has been adopted by many Government and businesses firms nowadays to secure their data. If somehow the key used for encryption is cracked by an attacker, the security of the whole system will fail. For such situations the proposed method of adding a password along with normal AES Encryption will make the system more secure and the time needed to crack the system increases further. The security strength analysis shows that the proposed method yields good results as desirable to a good cryptosystem.

References

[1] M. Pitchaiah, Philemon Daniel, and Praveen, “Implementation of Advanced Encryption Standard Algorithm”, International Journal of Scientific & Engineering Research, vol. 3, March 2012.

[2] AVI KaK, “The Advanced Encryption Standard lecture notes on Computer and Network Security”, October 18, 2014.

[3] How secure is AES against Brute force attacks? http://www.eetimes.com/document.asp?doc_id=1279619

[4] Deep Desai, Appoorv Prasad, Jackson Crasto, “Chaos-Based System for Image Encryption”, International Journal of Computer Science and Information Technologies, vol. 3(4), 2012.

[5] P. Radhadevi, P. Kalpana, K, “Secure Image Encryption Using AES”, International Journal of Research in Engineering and Technology (IJRET), Vol 1, October 2012.

[6] Kamel Faraoun, “Chaos-Based Key Stream Generator Based on Multiple Maps Combinations and its Application to Images Encryption”, The International Arab Journal of Information Technology, Vol 7, July 2010.