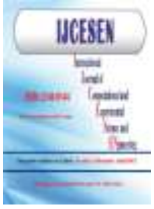
International Journal of Computational and Experimental Science and ENgineering
(IJCESEN)

Vol. 11-No.3 (2025) pp. 4424-4428
http://www.ijcesen.com

ISSN: 2149-9144



Research Article

Secure Multimedia Data Transmission Using AES-Based Encryption with Dummy Image Concealment for Attack Mitigation

A. Nesarani[1*], B. Ramesh[2], P. Keerthana[3], R. Deepa[4], T. Jayaprakash[5]

[1]Associate professor, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Vadeeswaram, Guntur, Andra Pradesh,India
*** Corresponding Author Email:** nesaraniabraham84@kluniversity.in – **ORCID:** 0000-0002-7604-9957

[2]Assistant professor, Department of Electronics and Communication Engineering, Dr.M.G.R. Educational and Research Institute, Maduravoyal, Chennai, Tamilnadu,India
Email: ramesh.ece@drmgrdu.ac.in - **ORCID:** 0009-0004-8491-6311

[3]Assistant Professor, Department of Information Technology, Sri Krishna College of Engineering and Technology, Coimbatore, Tamilnadu, India
Email: keerthanap@skcet.ac.in – **ORCID:** 0000-0002-7604-9957

[4] Professor, Department of Electronics and Communication Engineering, Nehru Institute of Engineering and Technology, Coimbatore, Tamilnadu, India
Email: ecedeepa@gmail.com – **ORCID:** 0000-0002-1099-2336

[5]Professor in Physics (S&H), Department of Science and Humanities, Nehru Institute of Technology Coimbatore, Tamilnadu, India
Email: nitjayaprakash@nehrucolleges.com – **ORCID:** 0000-0001-9698-8903

Article Info:

DOI: 10.22399/ijcesen.2508
Received : 15 March 2025
Accepted : 17 May 2025

Keywords

Federated Learning (FL),
Distributed Systems,
Differential Privacy,
Cryptography

Abstract:

In the image processing, while transferring multimedia data in the network there may occur some attacks. To prevent our encrypted data from attacks, we use Advanced Encryption Standard (AES). In this method we use a secured encrypted and decrypted data here; we explain with the help of two images for encryption the original image is covered inside the dummy image while decrypting data the original image will be displayed. If an attacks or intermediates a data the dummy image will be displayed. The original image is encrypted and decrypted with the help of secret key generation. No matters what the arrangements of permuted elements are, any permutation-only encrypted algorithm without change their histogram. The client can send and receive the data with full security.

1. Introduction

Image processing is a process in which the images were used for process. The images are performed some operations to secure a data. In which input image, like video frame or photography and output may be image or characteristics join together with that image. Image processing is processing of image using some mathematical operations by using any form of the process like signal processing for which the input is an image, a text image or a video, such as photography or video frames the output of image processing may be either image or set of characteristics or a numerical related to the images. Most of the image processing techniques involve deal with the certain way the images as a two-dimensional signal and applying standard signal-processing techniques to it. Image are also processed as three-dimensional signals where the third dimension being time. Image processing is closely related to the computer graphics and computer vision. In computer graphics, image are manually made from physical model of objects, the surroundings and effects of light, instead of being acquired from natural clips, as in most animated movies. Computer vision, on the other hand, is often considered high-level image processing out of which machine or computer or software design to decipher the physical content of image or a sequence of images. In modern sciences and technologies, images also gain much broader scopes due to the evergrowing importance of scientific visualization.

Image processing is technique to improve image quality by applying mathematical operations. Image processing projects works on pattern recognition concept which is used for object detection, classification and computer vision segmentation which requires some of the image processing algorithm or techniques. Many images are represented by 2D arrays, where each element stores information about a pixel in the image. Some image arrays have more dimensions to represent colour information or an image sequence. Research in computer communication network is underway to provide new methods for design, analysis, and optimization of increasingly complex and demanding communication. In genetics research, or real time multi-asset portfolio trading in finance.

2. Related Works

In cryptograph, they are designed an algorithm to give a secured communication between the network transaction. They specify a multimedia, Image Scrambling Encryption Algorithm (ISEA) has been successfully, in terms of the number of required plaintexts, the complexity of the attacks, and the storage space needed. As different scrambling elements may have dramatically different effects on the sensible information of the plaintext, different multimedia scrambling encryption algorithm may possess totally different strength against cipher text-only attacks. The secure type of the binary image scrambling encryption algorithm, against cipher text- only attacks and known/chosen-plaintext attacks was studied comprehensively. Just as previous cryptanalytic works on the class of permutation-only encryption algorithm have shown, secret scrambling operations are incapable of providing a sufficiently high level of security against known/chosen- plaintext attacks alone.

3. Proposed

The image processing is the process of the executing the input image and the output image, by using advanced encryption algorithm. The original name of AES is Rijndael.

The AES operates based on blocks that are 128-bits in length. There are actually 3 variants of the Rijndael cipher, each of which uses a different key length. The permissible key lengths are 128, 192, and 256 bits.

3.1 Mathematical Preliminaries of Aes

The fundamental unit of blocks of operations are operated upon is a byte, that is, 8 bits. Bytes are thought of in two different ways in Rijndael. Let the byte be given in terms of its bits as $b_7, b_6, \ldots, b_0$. We may think of each bit as an element in GF (2), the finite field of two elements. First, one may think of a byte as a vector, $(b_7, b_6,.., b_0)$ in GF $(2)^8$. Second, one may think of a byte as an element of GF (2^8), in the following way: Consider the polynomial ring GF (2) [X]. We may mod out by any polynomial to produce a factor ring. If this polynomial is irreducible, and of degree n, then the resulting factor ring is isomorphic to G (2n). In AES, we mod out by the irreducible polynomial $X^8 + X^4 + X^3 + X + 1$, and so obtain a representation for G (2^8). A byte is then represented in G (2^8) by the polynomial $b_7X^7 + b_6X^6 +. . .+b_0$. It is also convenient to refer to bytes (in either setting) by their hexadecimal representations. Of course, we may then define polynomial rings over G (2^8). Later on, the ring G (2^8) $[Y]/(Y^4 + 1)$ we be used. We note that while this is not a field (as $Y^4 + 1$ is not irreducible in $G(2^8)[Y]$, being equal to $(Y + 1)^4$), elements are invertible if they are cop rime to $Y^4 + 1$, that is, if they are not divisible by $Y + 1$.

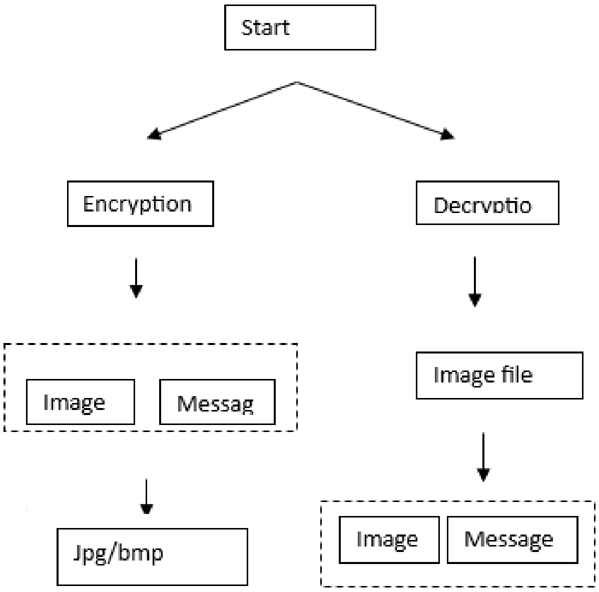


Figure 1. *Architecture diagram*

3.2 The States

For simplicity, we limit ourselves to describing AES with a 128-bit key. The other variants are essentially the same. Operations are done on intermediate results known as the state. The state is 128-bits long. We think of the state as divided into 16 bytes, $a_{(i, j)}$ where 0 _ i, j _ 3. We think of these 16 bytes as an array, or matrix, with 4 rows and 4 columns, like so state starts out as the 128-bit input. We operate on the state by performing successive rounds. A round is made up of three parts: application of the S-box, linear diffusion, and sub key addition. We discuss each part below.

:

$$\begin{pmatrix} a(0,0)\ a(0,1)\ a(0,2)\ a(0,3)\ldots \\ a(1,0)\ a(1,1)\ a(1,2)\ a(1,3)\ldots \\ a(2,0)\ a(2,1)\ a(2,2)\ a(2,3)\ldots \\ a(3,0)\ a(3,1)\ a(3,2)\ a(3,3)\ldots \end{pmatrix}$$

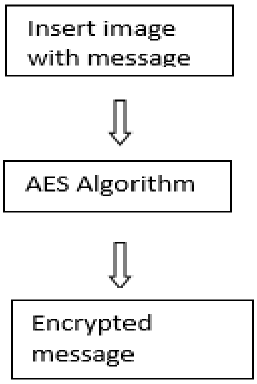


Figure 2. *Crypto Modules*

3.3 The S-Box

S-boxes, or substitution boxes, are common in block ciphers. These are objective functions on the blocks that are, ideally, highly non-linear. Much of the security of block ciphers can be thought of as ‘residing’ in their S-boxes. In AES, the S-box has a relatively simple form. The S-box is the same in every round, and it acts independently on each byte. It has two parts. For the first part, we think of each byte as living in GF (2^8). We then simply apply the ‘patched inverse’. This sends a byte a to a^{-1}. If a is non-zero, and send sit to 0 if it is zero. This can also be expressed as sending a→a^{254}. This inversion is actually optimal with respect to several measures of non-linearity, and non-linearity is important to protect against several common families of attack.

$$F \qquad \begin{pmatrix} 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1 \\ 1\ 1\ 0\ 0\ 0\ 1\ 1\ 1 \\ 1\ 1\ 1\ 0\ 0\ 0\ 1\ 1 \\ 1\ 1\ 1\ 1\ 0\ 0\ 0\ 1 \\ 1\ 1\ 1\ 1\ 1\ 0\ 0\ 0 \\ 0\ 1\ 1\ 1\ 1\ 1\ 0\ 0 \\ 0\ 0\ 1\ 1\ 1\ 1\ 1\ 0 \\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 1 \end{pmatrix}$$

or the second part, we apply an affine (over GF(2)) transformation. Think of the byte a as a vector in GF$(2)^8$. Consider the invertible matrix A, Much like inversion, the structure of A is relatively simple, successively shifting the prior row by 1. If we define the vector v 2 GF $(2)^8$ to be (1, 1, 0, 0, 0, 1, 1, 0), then the second half of the S-box sends a byte a to A · a + v^T. As a whole then, the action of the S-box is $a_{(i,\ j)} \rightarrow A.\ a_{(i,\ j)}^{-1} + v^T$.

3.4. Linear Diffusion

Next, we apply two different linear maps to ‘mix’ the state. The first map is the Row Shift. Here, we simply shift the rows around. The first row is unchanged; the second row is shifted to the left by 1, the second by 2, the third by 3. Graphically, if the state after the S-box step is denoted by the matrix ($a_{(i,\ j)}$), the new state is

$$\begin{pmatrix} a(0,0)\ a(0,1)\ a(0,2)\ a(0,3)\ldots \\ a(1,1)\ a(1,2)\ a(1,3)\ a(1,0)\ldots \\ a(2,2)\ a(2,3)\ a(2,0)\ a(2,1)\ldots \\ a(3,3)\ a(3,0)\ a(3,1)\ a(3,2)\ldots \end{pmatrix}$$

The second step, the Mix Column transformation, not surprisingly mixes the columns. We do more, however, than just move around bytes within the columns. We interpret the bytes of each column as the coefficients of a polynomial in GF(2^8)[Y]/(Y^4 + 1). Then, we multiply each column by the polynomial ‘03’Y^3 + ‘02’Y^2 + ‘01’Y + ‘02’ (which is invertible in our ring) and reduce appropriately.

Each step, and hence their composition, is linear, whether viewed over GF(2) or GF(28). Note that in the last round, for reasons of efficiency in decrypting, we will leave out the column mixing.

3.5 Subkey Addition

From the original key, we produce a succession of 128-bit keys, by means of a key schedule. The details of the key schedule need not concern us here; we simply note that later round keys are produced from earlier round keys by applications of the S-box above and by XOR ring prior round keys together. Each 128-bit round key may then be divided into bytes, and the bytes placed in a 4x4 matrix. We refer to the (i, j)th byte of the mth round key by $k_{m,(i,j)}$. Then in round m we replace byte a of the current state with $a_{(i,j)} \oplus k_{m,(i,j)}$.

3.6 Xor Opetations

The AES algorithm is then as follows. Put the input into the state. We start with this because any actions before the first (or after the last) use of the key are pointless, as they are publicly known and so can be undone by an attacker. Then, apply 10 of the above rounds, skipping the column mixing on the last round (but proceeding to a final key XOR in that round). The resulting state is the cipher text.

3.7. Hidden Phases

In hidden phase, we use two images, then we segregate one image as original image and another one as dummy images. This techniques was implemented using Most Significant Bit(MSB). MSB-based information hiding algorithm has the Feature of high robustness, but it may easily lead to changes in picture quality. In order to improve the security and robustness of hidden information in the image, the algorithm must be improved.

4. Attacks

4.1. Algorthim Attack

Algorithmic attacks are in some ways much more difficult to perform because they generally require an extremely high degree of knowledge in mathematics. Rather than going after the entire key space, the code breaker will try and find flaws in the algorithm that causes it to be reduced to a problem of decreased complexity.

4.2. Birthday Attack

A brute-force attack used to find collisions. It gets its name from the surprising result that the probability of two or more people in a group of 23 sharing the same birthday is greater than 1/2.

4.3. Brute Force Attack

Brute Force Attack is a form of attack in which each possibility is tried until success is obtained. Typically, a cipher text is deciphered under different keys until plaintext is recognized.

4.4 Chosen Plaintext Attack

A form of cryptanalysis where the cryptanalyst may choose the plaintext to be encrypted.

4.5. Ciphertext–Only Attack

A cryptanalysis where has some cipher text but nothing else. Modern cryptosystems are not weak against cipher text-only attacks; however, in practice it is often possible to guess the plaintext, as many types of messages have fixed format headers. For example, many classical attacks use frequency analysis of the cipher text; however, this does not work well against modern ciphers.

4.6. Known Plaintext Attack

A cryptanalysis where knows both the plaintext and the associated cipher text.

5. Results

5.1. Output

Figure 3. *Original image*

Figure 4. *Crypt image*

6. Conclusion

In this paper we have explained that the secured multimedia data files were studied comprehensively. Just as works on the class of permutation-only encryption algorithms have shown, secret operations are capable of providing a secured high level of security against attacks alone. Correlation existing in multimedia data may be used to support some specific attacks and enhance breaking performance; the size of each independent domain should be carefully checked to obtain the expected security requirement; No matter what the permuted elements are, any permutation-only.

Author Statements:

- **Ethical approval:** The conducted research is not related to either human or animal use.
- **Conflict of interest:** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper
- **Acknowledgement:** The authors declare that they have nobody or no-company to acknowledge.
- **Author contributions:** The authors declare that they have equal right on this paper.
- **Funding information:** The authors declare that there is no funding to be acknowledged.
- **Data availability statement:** The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

References

[1] Hayder Raheem Hashima, Irtifaa Abdalkadum Neamaab (2014). Image Encryption and Decryption in A Modification of ElGamal Cryptosystem in MATLAB. hayderr.almuswi@uokufa.edu.iq. Basic and Applied Research (IJSBAR) Vol 14(2), 141-147.

[2] Jyotika Kapur (2013). Security using image processing jyotikakapur18@gmail.co, baregar1611@gmail.com (IJMIT) Vol.5(2), May.

[3] H.B.Kekre, Archana Athawale and Pallavi N. Halarnkar (2009). Polynomial Transformation to improve Capacity of Cover Image for Information Hiding in Multiple LSBs, International Journal of Engineering Research and Industrial Applications (IJERIA), Ascent Publications, Vol 2, March, Pune.

[4] G. Ye (2010), Image scrambling encryption algorithm of pixel bit based on chaos map, Pattern Recognition Letters, vol. 31(5), 347–354.

[5] L. Zhao, A. Adhikari, D. Xiao, and K. Sakurai (2012), On the security analysis of an image scrambling encryption of pixel bit and its improved scheme based on self-correlation encryption, Communications ion Nonlinear Science and Numerical Simulation, vol. 17(8), 3303–3327.

[6] X. Ge, B. Lu, F. Liu, and D. Gong(2016), An image encryption algorithm based on information hiding, International Journal of Bifurcation and Chaos, vol. 26, p. art. no. 1650192.

[7] A. Jolfaei, X.-W. Wu, and V. Muthukkumarasamy(2016), On the security of permutation-only image encryption schemes, IEEE Transactions on Information Forensics and Security, vol. 11(2), 235–246.