CRYPTOIMAGE: EFFICIENT CRYPTOSYSTEM FOR DATA HIDING IN HIGH DYNAMIC RANGE IMAGES

Balamurugan.V

1Assistant Professor, Department of CSE, S.A. Engineering College, Chennai, India

Abstract— Data preserving is one of the key challenges in these days. There are many network attacks, cyber attacks from which the sensitive data suffer from. Steganography techniques support data hiding inside an image to certain extent. It also falls short in maintaining the integrity when transmitted through the internet. “CryptoImage” solves this problem by embedding encrypted data onto an HDR image which is absolutely non-prone to attacks. We implement cipher techniques to modify the RGB value of each pixel and store the sensitive data onto it, thereby concealing the confidentiality of the message. Graphics Drawable Library (GD library) of PHP module present in PHP engine performs the conversion of HDR images into Base64 format in which the encrypted text is embedded and then stored onto the server. AES encryption is used to encrypt the original secret message into cipher text which is appended to the converted image file and then stored onto the server thereby concealing the security and confidentiality of the original message.

Keywords— Cryptosystem, Advanced Encryption Standard, Steganography, HDR images, Data hiding.

I. INTRODUCTION

Information security [1]– [3] over the systems is an essential test for specialists and PC engineers for a considerable length of time. Both the information concealing [3] and encryption procedures [3] are observed to be the fundamental instruments in information security However, use of former mechanism has been increasing recently due to some demerits have been found in the later mechanism. Steganography is a method of concealing data in advanced media. Rather than cryptography, it isn’t to shield others from knowing the shrouded data yet it can shield others from believing that the data even exists. Steganography turn out to be more imperative as more individuals join the internet insurgency[19]. The developing conceivable outcomes of present day interchanges require the extraordinary methods for security particularly on PC organize. The system security is ending up more essential as the quantity of information being trade on the web increments. Accordingly, the privacy and information respectability are required to secure against unapproved access and utilize. This has brought about a dangerous development of the field of data stowing away.

As a straightforward illustration, a sender might start with an innocuous image file and adjust the color of every 100th pixel to correspond to a letter in the alphabet, a change so subtle that someone not specifically looking for it is unlikely to notice it. Digital Steganography has three basic components. (a) Obtain the data to be hidden, i.e., secret message, (b) embed the secret message into the cover medium, i.e., images, sounds or videos, etc., and (c) lastly, obtain the stego-carrier to be sent.

There exist a lot of Steganography based information concealing systems which utilize LSB based calculations for concealing the mystery messages by installing it into cover media like pictures. In [7], creators proposed an information concealing system which depends on basic LSB [24] substitution strategy by choosing ideal quantities of k LSB substitution technique to take care of
the issue while $k$ is observed to be vast. Another work proposed in [8] utilizes dynamic programming way to deal with find ideal LSB which later implanted into picture.

II. RELATED WORK

In [11], the flipping priorities are computed by partitioned an image into $3\times3$ sub blocks and then modifying the total number of black pixels to be either odd or even embeds data bits. Choosing the randomness for the embedding locations creates poor visual effects despite the large capacity[15][16][17]. Improvements over the visual quality is made by choosing the edge pixels in their paper that is the problem in [12]. In [12], the proposed scheme uses a secret key and a weight matrix to protect the hidden data, it also uses a weight matrix to enhance the data hiding ratio. The operator XOR is adopted so that the keys can’t be compromised easily. In [5], the authors proposed a novel data hiding method by partitioning an image into blocks, where each block was repartitioned into overlapping sub-blocks. Each sub-block is connected with a level number according to its pattern, indicating power on visibility by assumed change of the central pixel in the sub-block. Data will be concealed by changing the central pixel in a sub-block. In [6], the LSB is the most popular Steganography method. It covers the secret message in the RGB image based on its binary coding. Figure 1 presents an example about pixel values and shows the secret message. LSB algorithm is used to hide the secret messages by using algorithm 1. LSB makes the changes in the image resolution quite clear as well as it is easy to attack [17].

III. PROBLEM DEFINITION

Given a HDR image of pixels $X$ to $Y$ where $X$ and $Y$ are the pixel ranges in X and Y axis, a secret message of ‘n’ bit length; let it be a string of alphanumeric characters, the algorithm should return an image with hidden encrypted text in it. As mentioned earlier, there are several HDR image formats. An uncompressed HDR image uses 96 bits to store as a RGB image, which is four times of the conventional LDR image. To avoid using such images which are not suitable for practical use, here choose one common HDR image format, called RGBE. This is the first efficient HDR format which was proposed by G. Ward [7] in 1991. The other encoding formats include OpenEXR [8], LogLuv [9], etc.

![Figure 1. Least Significant Bit Hiding Technique](image1.png)

![Figure 2. Illustration of RGBE format](image2.png)
IV. TECHNOLOGIES AND IMPLEMENTATION

A cryptographic technique has to be adapted to encode the bits of the pixel positions of the HDR images. The image should not be distorted so that it should not get attracted by the hackers. The system is designed in such a way that all the crypto images are given as news feed to all the registered users. If the user wants to decrypt the message within a particular HDR image, he has to send a request for the key. The user who posted that image, will send the secret key and OTP to the requested user. The system is developed in PHP with AJAX and JSON parsing for easier understanding and efficient data transmission.

![System Architecture of CryptoImage](image)

**Figure 3. System Architecture of CryptoImage**

4.1 PHP GD LIBRARY

The GD library is used for dynamic image creation. From PHP we use the GD library to create GIF, PNG or JPG images instantly from our code. If you are unsure if you have GD library, you can run `phpinfo()` to check that GD Support is enabled. The header() function is used to tell the browser which content type it will be sent.

```php
<?php
header("Content-type: image/png");
$handle = ImageCreate (130, 50) or die ("Cannot Create image");
$bg_color = ImageColorAllocate ($handle, 255, 0, 0);
$txt_color = ImageColorAllocate ($handle, 0, 0, 0);
ImageString ($handle, 5, 5, 18, "PHP.About.com", $txt_color);
ImagePng ($handle);
?>
```

- With this code, we are creating a PNG image. In our first line, the header, we set the content type. If we were creating a jpg or gif image, this would change accordingly.
- Next, we have the image handle. The two variables in `ImageCreate()` are the width and height of our rectangle, in that order. Our rectangle is 130 pixels wide, and 50 pixels high.
- Next, we set our background color. We use `ImageColorAllocate()` and have four parameters. The first is our handle, and the next three determine the color. They are the Red, Green and Blue values (in that order) and must be an integer between 0 and 255. In our example, we have chosen red.
- Next, we choose our text color, using the same format as our background color. We have chosen black.
- Now we enter the text we want to appear in our graphic using `ImageString()`. The first parameter is the handle. Then the font (1-5), starting X ordinate, starting Y ordinate, the text itself, and finally it's color. Finally, `ImagePng()` actually creates the PNG image.
4.2 AES 256 BIT ENCRYPTION IN PHP

Advanced Encryption Standard (AES) is a specification for the encryption of electronic data established by the U.S. National Institute of Standards and Technology (NIST) in 2001. Also referenced as Rijndael, it is one of the most powerful encryption algorithms available today. AES can be described as an iterative, symmetric block cipher. It is iterative because the encryption process is done in multiple rounds, symmetric because it uses the same key for encryption and decryption and block cipher because it performs operations on blocks of data. Possible key lengths are 128, 192 and 256 bits (192 or 256-bit keys are required for top secret data). The number of iterations (rounds) is 10, 12 or 14, depending on the length of the key. Blocks of data are always 128 bits long. That class implements AES-128 encryption (AES-256 can be used by changing the key size inside the class), PKCS7 padding and authenticates messages with HMAC-SHA256. It requires PHP 5.4 or newer and OpenSSL PHP extension.

```php
$plaintext = 'My secret message 1234';
$password = '3sc3RLrpd17';
$method = 'aes-256-cbc';

AES ENCRYPTION (Secret Text To Encrypted Text)
Input: Secret Text and Secret Key
Output: Encrypted Text Algorithm
Step 1: Concat Key 8 Times and choose first 8 characters from KEY as (c1,c2,c3,c4, c5,c6,c7,c8)
Step 2: Convert each character (o/p of Step1) into ASCII and sum and divide from 1000 and reminder of this process is our Final Key of three digit number (N1, N2, N3)
Step 3: according to these three numbers (N1, N2, N3) we change our original message into cipher message we use shift encryption and every 1,4,7,… character shift by N1 every 2,5,8,… character shift by N2 every 3,6,9,… character shift by N3 and we get cipher text

CRYPTOIMAGE ALGORITHM (EMBEDDING)
Input: Secret Text, Secret Key, Image
Output: Stego Image
Step 1: For Placing Data into images pixel first we will calculate DataPositionArray by concatenating of ASCII of N1, N2 and N3.
Step 2: Choose first 20 characters from DataPositionArray
Step 3: Count 1 in this 20 character long DataPositionArray
Step 4: Calculate size of Cipher Text (StrLen) in Bytes Strlen=StringLength(ciphertext) * 8
Step 5: Now SizeOfImage( Length X Height ) should be greater than Show The Minimum Image Size ( Length X Height ) as that [(CountOne/20)*(1 /4)* Length*Height]) > StrLen Convert cipher text into array of bits and then make 4 X 4 pixel Block
Step 6: Convert LSB Bit of block Diagonal pixel After inserting all cipher text bit we get final image.

V. EXPERIMENTAL RESULTS AND FINDINGS

The proposed work is implemented on XAMPP server in a Windows 10 machine. The PHP code with GD library is residing at thehtdocs folder of XAMPP directory. MySQL server is supporting the back end storage as mentioned in the architecture diagram. The image is converted to base64 format. The secret key is encrypted and embedded inside the image which is present in base64 format after the delimiter #. Now the image is proceeded to get stored inside the server. Now if any hacker tries to compromise the server in retrieving the data, he/she can only get the encrypted format of the secret message in base64 format, which cannot be decrypted because it is encrypted with AES 256 bit encryption, thereby making the system mode secure.
In our experiments, we used 32 Bytes English text document and 512X512 lenna image. Figure 4.1 illustrates a 512X512 original image and we can hide 256 bits in the original image which is shown in Figure 4.1. The stegoimage is shown in Figure 4.2 are embedded in it using the proposed techniques. The pixels in image are very approximately uniformly distributed; therefore, it is very difficult to detect the hidden data by naked eyes.

![Figure 4.1 Original image](image1.png)  ![Figure 4.2 stego image](image2.png)  ![Figure 4.3 Histogram of red color and Stego images](image3.png)

Table 1. Test Results For 512x512x24 Color Images

| Comparison of LSB and Our Algorithm | Lena | Baboon |
|-------------------------------------|------|--------|
| Capacity (bits)                     | LSB  | Our Algorithm | LSB  | Our Algorithm |
|-------------------------------------|------|----------------|------|----------------|
|                                     | 788  | 812            | 570  | 598            |
| Robustness                          | .8   | .7             | 1.6  | 1.4            |

Note that there is no noise in all of tests since the proposed algorithm does not use modulo-256 addition. The embedding capacity can range from 512 to 1024 bits for the purpose of authentication, and it can be adjusted by changing the block size for other applications.

**VI. CONCLUSION AND FUTURE ENHANCEMENT**

The proposed system Cryptoimage works in such a way that it is concealed from the hackers that the original HDR image will not look distorted so that it is not prone to hacks. The system works with considerably medium pixel images also because embedding the information inside the HDR image is easy since there is a lot of pixels in it. CryptoImage converts the HDR image into bytes and adds the encrypted information from the AES 256 bit library. The retrieval of the original message is also safe because it asks for 16bit key from the registered email upon approval. The project is completely open source because there is no proprietary software involved in it. We have used XAMPP for server, PHP GD library for converting the HDR image into bytes, AES library from GitHub, HTML and CSS for page display. The proposed work will be extended in future in such a way that a cloud will be integrated to it for anywhere anytime retrieval of data. AES library should be replaced with some other advanced cryptographic standards such as ECC (Elliptical curve cryptography). The server load capacity should be increased.

**REFERENCES**

[1] Yun-Te Lin, Chung-Ming Wang, Wei-Sung Chen, Fang-Pang Lin, and Woei Lin, “A Novel Data Hiding Algorithm for High Dynamic Range Images”, IEEE Transactions on Multimedia, Volume 19, No. 1, January 2017
[2] Jing-Ya Yan, Tzung-Her Chen, Chih-Hung Lin, “Encryption in HRD images for RGBE format”, 2013 international conference on Intelligent information hiding and multimedia signal processing, Vol 5, pp – 493 - 496

[3] Vassilios Solachidis, Emmanuele Mioriana, Patrizio Campisi, Banterle, “HRD Image watermarking based on bracketing decomposition”, IEEE transactions on Digital Video and Forensics, DSP 2013 – pp.978-1-4673-5807-1/13

[4] Abdelhamid Awad Attaby a, Mona F.M. Mursi Ahmed b, Abdelwahab K. Alsammak , “Data hiding inside JPEG images with high resistance to steganalysis using a novel technique: DCT-M3” 2017 Elsevier Science Direct Journal

[5] Abdullah Hamid, “Stega Image a Technique to Hide Data within Image File, Using Image Steganography and Encryption”, International Journal Of Engineering And Computer Science ISSN: 2319-7242 Volume 6 Issue 3 March 2017, pp. 20417-20426

[6] Chandreeye Maiti, T.-h. Kim et al., ”Data Hiding in Images Using Some Efficient Steganography Techniques”, (Eds.): SIP 2011, CCIS 260, pp. 195–203, 2011. © Springer-Verlag Berlin Heidelberg 2011

[7] Priya Bharti, Rupali Soni, “A New Approach of Data Hiding in Images using Cryptography and Steganography”, International Journal of Computer Applications (0975 – 8887) Volume 58– No.18, November 2012

[8] Minati Mishra, Priyadarshini M-ishra, “DIGITAL IMAGE DATA HIDING TECHNIQUES: A COMPARATIVE STUDY”, ANSVESA, 7(2), 105-115, 2012

[9] Hsi-Chun Alister Wang, “Data Hiding Techniques for Printed Binary Images”, © 2010 IEEE

[10]Navdepp, “Hide Text in Images Using Steganography and a Review of Methods and Approach for Secure Stegnography”, International Journal of Research in IT & Management, Volume 6, Issue 5, May, 2016.

[11]Padmapriya, Sandhya, et.al., “Stego integrated Image encryption using row and column indexing”, International Conference on Computer Communication and Informatics (ICCCI -2017), Jan. 05 – 07, 2017 IEEE.

[12]Qibin Sun’, Shuiming, “A Crypto Signature Scheme for Image Authentication over Wireless Channel”, IEEE International Conference on Multimedia and Expo, 2014, IEEE.

[13]Radha, Rachel, “Enhanced Blend of Image Steganography and Cryptography”, International Conference on Computing Methodologies and Communication (ICCMC), 2017 IEEE.

[14]Yi-Fei Tan1, Xiaoning Guo1, Wooi-Nee Tan1, “Reversible Data Hiding using Reference Points Coding”, 2017, IEEE.

[15]Rahat Zaheer, Rohit Gaur, “A Literature Survey on Various Approaches of Data hiding in Images”, International Conference on Innovations in information Embedded and Communication Systems (ICIECS), 2017 IEEE.

[16]Ms. Prachi D. Rathod, “Secure Bank Transaction using Data Hiding Mechanisms”, International Conference on Innovations in Information, Embedded and Communication Systems (ICIECS), 2017.

[17]Koushik Pal, Subhajit Koley, “A New Combined Crypto-Watermarking Technique using RSA Algorithm and Discrete Cosine Transform to Retrieve Embedded EPR from Noisy Bio-Medical Images”, IEEE 1st International Conference on Condition Assessment Techniques in Electrical Systems, 2013.

[18]Tarek Hadje, “A New Image Crypto-compression System”, S - PIHT-PSCS”, pp. 706 – 711, 2014 IEEE.

[19]Koushik Pal, Goutam Ghosh, “GUI based Smart Breast Cancer Identification System for Mammographic Images through 2nd Level Secured Combined Crypto-Watermarking”, International Conference on Bioinformatics and Biomedicine, 2014, IEEE.