Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IMPACT FACTOR: 6.017

IJCSMC, Vol. 6, Issue. 6, June 2017, pg.328 – 331

LUCAS NUMBER RANDOMIZED LSB IMAGE STEGNOGRAPHY FOR SMALL MESSAGES

Shah Asif Bashir[1], Faizan Manzoor[2], Aaqib Manzoor[3], Zain Ashraf Wani[4], Shahid Mohi Ud Din[5]

[1]Department of CSE, SSM College of Engg and Tech, Baramulla, India

[2]Department of CSE, SSM College of Engg and Tech, Baramulla, India

[3]Department of CSE, SSM College of Engg and Tech, Baramulla, India

[4]Department of CSE, SSM College of Engg and Tech, Baramulla, India

[5]Department of CSE, SSM College of Engg and Tech, Baramulla, India

[1] shaharslan3339@gmail.com;[2] faizankhan_39@yahoo.co.in; [3] aaqu74@gmail.com;

[4] zainwani5@gmail.com; [5] shahidsb87@gmail.com

ABSTRACT --- This paper describes a novel – technique based on the combination of Lucas numbers and Fibonacci numbers for hiding data in digital images. This paper extends the LSB steganography technique to implement highly encrypted and highly randomized message embedding methodology.

Keywords --- LSB, Steganography, Lucas numbers.

I. INTRODUCTION

Steganography is a technique of hiding data in some other kind of data. Various techniques for steganography are present; the simplest one is the image steganography [1]. In image steganography, the data we want to send to the other user is embedded in a digital image using a predefined method and then this image is send to the receiver. On receiving the image, the receiver collects the data back from the image. In our system, we used the LSB method of image steganography and used the Lucas numbers and Fibonacci numbers to find the pixel in which we want to embed the data.

© 2017, IJCSMC All Rights Reserved

II. LSB METHOD

LSB (least significant bit) method is an approach of data hiding where we use the least significant bit of a pixel to insert the data. LSB uses the 8^{th} bit of every byte inside an image to hide the message. When using the 24-bit color image, each 8^{th} bit of Red, Green and Blue color bytes is used [2].

Consider an example to store the data (101) to be stored in a single pixel. If the pixel is as:

10010000 10101011 11101110

After applying the LSB method the bytes change to

1001000**1** 1010101**0** 1110111**1**

In our system we use the LSB method but instead of changing only a single bit we change the two least significant bits of each byte. Thus the above data can be embedded by our method as:

100100**10** 101010**1**1 11101110

III. LUCAS NUMBERS

These were developed by French mathematician, Eduardo Lucas (1842-1891). These numbers are the same as the numbers from the Fibonacci series but instead of starting from 0 and 1, it starts with 2 and 1. The Lucas number Ln can be found as:

$$L_n = L_{n-1} + L_{n-2} \quad \text{for } n>1 \tag{1}$$

$$L_0 = 2$$

$$L_{1 = 1}$$

IV. EMBEDDING METHODOLOGY

In this system, we first convert the message into the ASCII representation. Then we ask the user to give a password or a key to the system. The system uses this key and the inbuilt salt value to encrypt the message using the AES encryption. After the encryption is done, we take the image and based of the subtraction of Lucas number and the Fibonacci number we choose a pixel to embed the data using LSB method. But each two LSB bits of each byte is used to embed the data.

© 2017, IJCSMC All Rights Reserved

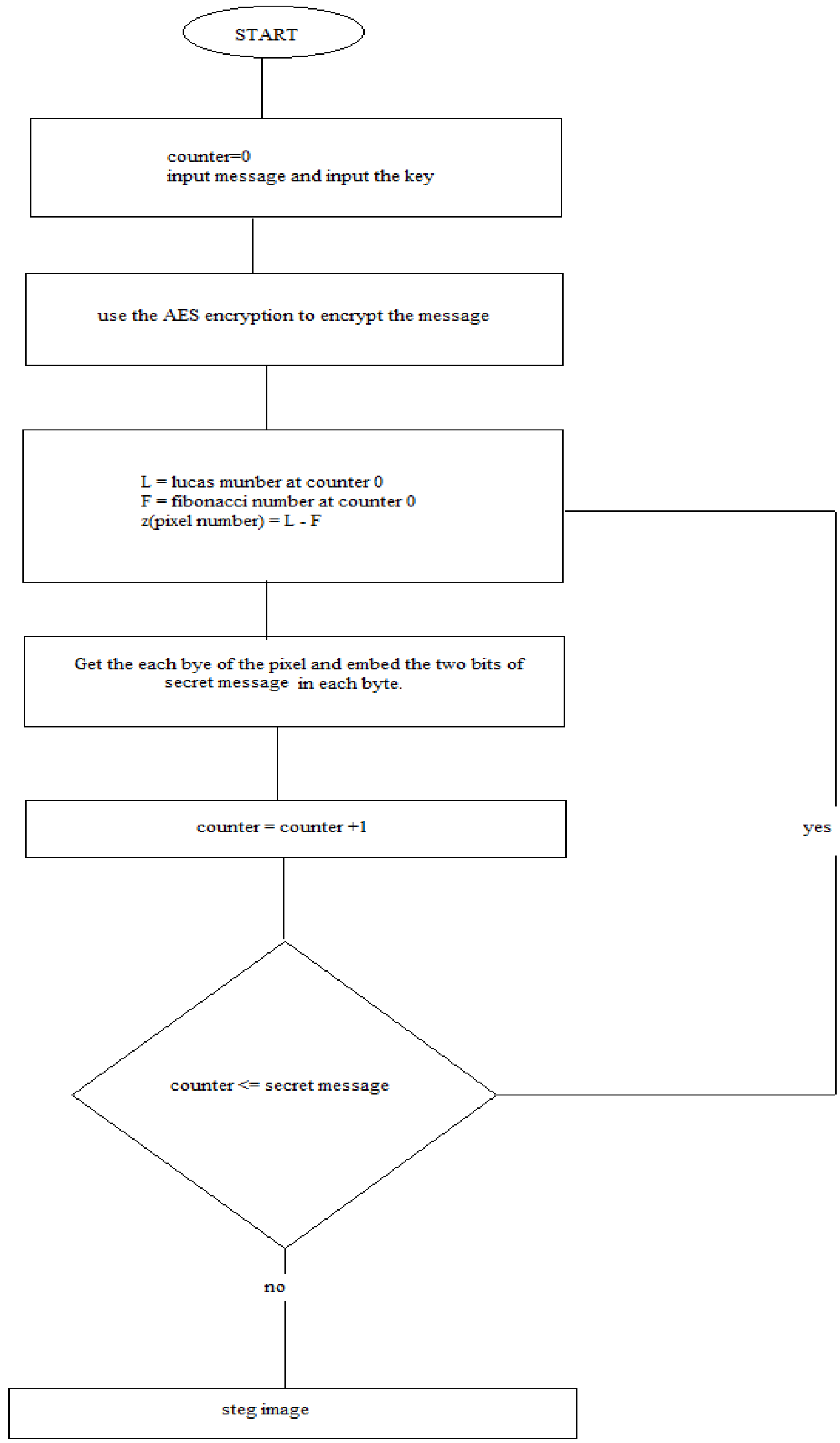

Fig 1: Flow Chart of embedding methodology

© 2017, IJCSMC All Rights Reserved

V. DATA RECOVERY METHOD

Every user who wants to get the message has to have the secret key that was used during embedding. Here we first collect the data from each pixel using same method to find the pixel used during embedding process and then store it in the container. The AES decryption is then applied to the collected data to get the binary data. The binary data is then converted to the output message using ASCII conversion.

VI. RESULT AND CONCLUSION

The message that we want to send has to be less than 120 words else the message is not embedded fully in an image of 1080 x 1320. For large size images, the larger messages can be embedded but the limitation of using Lucas numbers for choosing a random pixel to embedded data is that with larger values of “N”, the Lucas number doubles in size. Although we are using Fibonacci numbers to bring the Lucas number down, it works only for few values of “N”

The experimental result showed that the technique is effective in integrating the hidden message without any significant distortion to original image. But for the large size message, more than 500 words distortion of the significant proportion was found

ACKNOWLEDGEMENT

Words are not just enough to express our gratitude but we take this opportunity to express our profound sense of gratitude and respect to all those who helped us throughout the duration of this paper. First of all we are very thankful to our **HOD Yasmeen Bhat** for her regular support and guidance. We are also very thankful to Allah for providing us such a great opportunity. We feel privileged to offer our sincere thanks and deep sense of gratitude to our college for expressing confidence in us by letting us work on a paper of this magnitude and providing support, help & encouragement in implementing this paper.

REFERENCES

[1] Anderson, R.J. & Petitcolas, F.A.P., “On the limits of steganography”, IEEE journal of selected areas in communication, May 1998.

[2] Silman, J., “Steganography and Steganalysis: An overview”, SANS Institute, 2001.

© 2017, IJCSMC All Rights Reserved