Message hiding using ECC and Blowfish

Theja G.D[1]
Student, (M.Tech)
Department of Computer science AIT
Chikkamagaluru, Karnataka

S.J. Prashantha[2]
B.E, M.Tech
Department of Computer science AIT
Chikkamagaluru, Karnataka

Abstract: **The paper presents an idea to transfer the data securely using an encryption mechanism involving data and image encryption. In this mechanism, first the data is encrypted using Elliptic curve cryptography (ECC) then embedded into an image. Then the image is encrypted using blowfish algorithm, and this final combination can be transferred via any transfer medium to the authorized recipient. On the receiver side the exact reverse of the encryption is done, the original message is obtained using decryption of ECC and Blowfish algorithms. This mechanism provides a lossless data recovery and message size sent is equal to image size.**

Keywords—Reversible Data hiding; Secret message; Encryption; Embedding; Decryption, ECC, Blowfish.

I. INTRODUCTION

Due to the rapid development in the field of information processing, it is very essential to send secret messages in a confidential way. In recent years the field of information processing has attracted interests among the researches. In many fields such as military services, cloud computing and medical services there is a needs to send the secret messages to a remote server for future use. Service providers are not trusted by many of the users hence it is very necessary to encrypt the message before sending it to the receiver. Thus processing is done in the encrypted domain by the service provider.

Reversible data hiding is a method in which we embed the secret message into a cover image at the sender side and recover the exact message at the recipient side. Data hiding is used in the applications such as military, medical, private messaging and espionage. Small distortion because of message embedding is tolerable in many of the applications. To recover the original message without any loss of information is the desirable property in many of the fields such as medical, military and legal. The method of reversible message hiding in which the original message can be recovered exactly has attracted interests from the community. Original message can be recovered without any distortion using reversible message hiding.

Separable reversible data hiding (RDH) scheme for encrypted images is proposed in the existing method. In separable reversible data hiding method for encrypted images user divides the image into blocks and embeds one bit into each block by tossing three least significant bits (LSB) of half the pixels in the block. On the receiver side data loss has occurred when message retrieved from image, is the drawback of the existing method. To overcome this problem, the proposed system gives a solution, the cover image is encoded by using stream cipher to select the most significant bits
(MSB‘s) and compressing those MSB‘s to make space for hiding the secret message as shown in figure 1.

In the proposed system a grey scale image is taken as an cover image to hide the secret message, and the cover image is encoded by using stream cipher to select the most significant bits (MSB‘s) and compressing those MSB‘s to make space for hiding the secret message . Read the secret message from the user, encrypt that message and embed the message into the image in the space where compression of bits are done and lastly encrypt the embedded image. The main aim of the proposed system is to design and implementing a system by selecting an input image and embedding the encrypted secret message into that image using data hiding technique. The proposed method aims to increase the embedding payload that is inserted into the image. The idea is inspired by the distributed source coding, where we encode the selected bits of the image after stream cipher is done. The objective of this system is to enhance the embedding payload and to extract and reconstruct the hidden message without any data loss at the receiver side.

II. MOTIVATION

To overcome the drawback of inseparability in the previous work, a reversible message hiding scheme was suggested, in which we recover the original message and original image at the receiver side. It is very essential to send secret messages in a confidential way, for an instance, consider

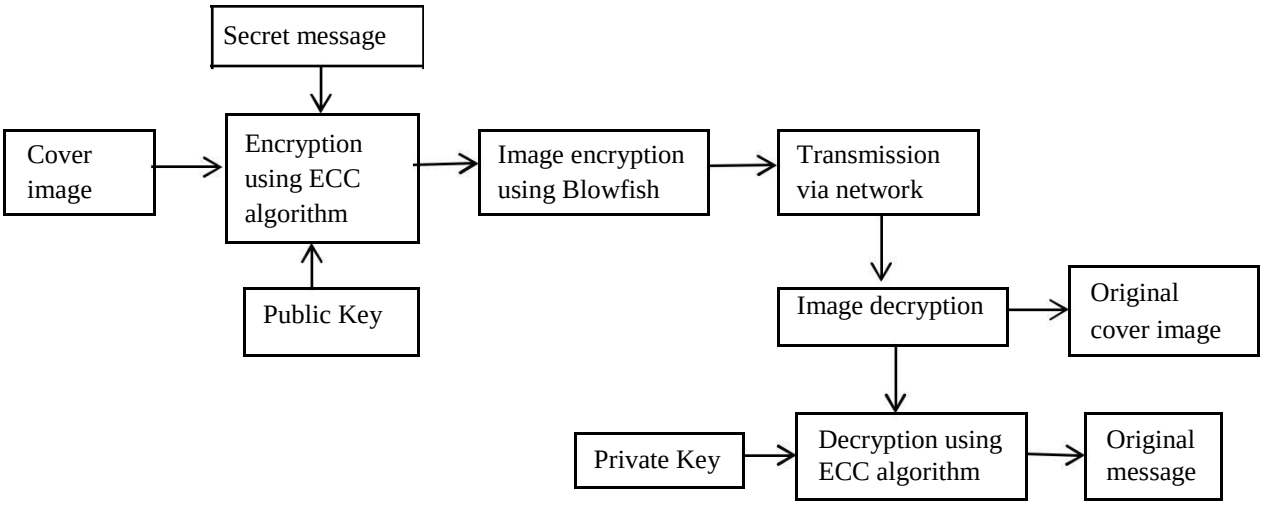

Figure 1: Reversible data hiding system

military services where secret messaging is very necessary, if they want to send some secret message to the government then they can encrypt the message and then embed the message in an encrypted image and send the message to the government, in these cases data hiding plays a very important role.

III. RELATED WORK

In this section related survey about message hiding, reversible message hiding and algorithm used for message hiding is briefly described.

A. Reversible Message Hiding

By reserving room before encryption [3] is a traditional reversible message hiding scheme and hence it is easy for the user to embed the message into the encrypted image. The proposed method can accomplish real reversibility that is; data extraction and image recovery are free of any error. Existing methods implement RDH in encrypted images by vacating room after encryption, as opposed to which we proposed by reserving room before encryption. The message hider makes use of the space emptied out in previous stage to make message hiding process efficient. The proposed method can take advantage of all traditional reversible message hiding techniques for plain images and accomplish excellent performance without loss of information.

The work divides an encrypted image into blocks, and each block carries one bit by tossing three Least Significant Bits of a set of pre-defined pixels [4]. Based on the block smoothness the data extraction and image recovery can be examined. To decrease the error rate of extracted-bits it adopts a better scheme for measuring the smoothness of blocks, and uses the side-match scheme.

The proposed method offers better performance based on the experimental results. The extraction and recovery of messages are performed according to the descending order of the absolute smoothness difference between two candidate blocks. To further reduce the error rate the side match technique is employed.

B. *Elliptic Curve Cryptography* (ECC)

In the implementation of elliptic curve cryptography (ECC) the plaintext is encoding using the public key and decoded using private key [6]. In ECC encryption and decryption methods the characters in the message are converted into ASCII values and are plotted on ECC curve. Every character in the secret message that is the ASCII values for each of these characters are mapped to the elliptic curve, a curve with minimum of 128 points are selected so that each point on the curve is fixed to the ASCII value. Different values of ECC parameters take different amount of time. The Execution time is constant for decoding different values of a, b, p. Compared to encoding the execution time for decoding is negligible.

C. Blowfish

Describes [5] the Purpose of parallel implementation of Blowfish cryptography algorithm is to improve the speed up of encryption and decryption so that large files also can be communicated on the network in secure and efficient way. This paper demonstrates the way of implementing Blowfish cryptography algorithm on GPU for improving performance. This implementation uses GPGPU and CUDA. CUDA is used as a programming model for implementing on the GPU. The experiment shows multifold difference in performance of CPU and GPU in encryption-decryption of large files.

Introduction of dynamic elements such as P-arrays and S-boxes makes the modified blowfish algorithm and that is explained in the paper.

IV. METHODOLOGY

Initially an image is taken as a cover image to embed the secret message. The selected image is stream ciphered to get most significant bits (MSB) and compress those MSB bits to make space for embedding the encrypted secret message, the

sender reads the secret message from the user, encrypt that message and embed the message into the image in the space where compression of bits are done and finally encrypt the image that was embedded. Secret message is encrypted using Elliptic curve cryptography (ECC) algorithm and image is encrypted using Blowfish algorithm.

In ECC, each characters of the secret message are represented as ASCII values and those ASCII values are plotted on the ECC curve using equation 1.

$$Y^2 = x^3 + ax + b \quad \text{..... (1)}$$

Sender encrypts the message using public key generated by the elliptic curve cryptography using the formula as shown in equation 2.

$$Q = d*P \quad \text{..... (2)}$$

Let _m‘ plain text that needs to be encrypted. The message is represented on the curve. Consider *‘m’* has the point *‘M’* on the curve *‘E’* as shown in equation 3 and 4.

$$C1 = k*P \quad \text{..... (3)}$$

$$C2 = M + k*Q \quad \text{..... (4)}$$

At the receiver side the message is decrypted using the formula as shown in equation 5 to retrieve the originally sent secret message without any loss in the message.

$$M = C2 - d * C1 \quad \text{..... (5)}$$

The process of getting back the original message is as shown below,

$$M = C2 - d * C1$$

‘M’ can be represented as ‘C2 – d * C1’

Message encryption: Once user‘s secret message M is read, it is encrypted using ECC encryption

C2 – d * C1 = (M + k * Q) – d * (P * k) (C2 = M + k * Q and C1 = k * P)

= M + k * d * P – d * k *P (canceling out k * d * P)

= M (Original Message)

The flow for ECC encryption and decryption is as shown in the figure 2.

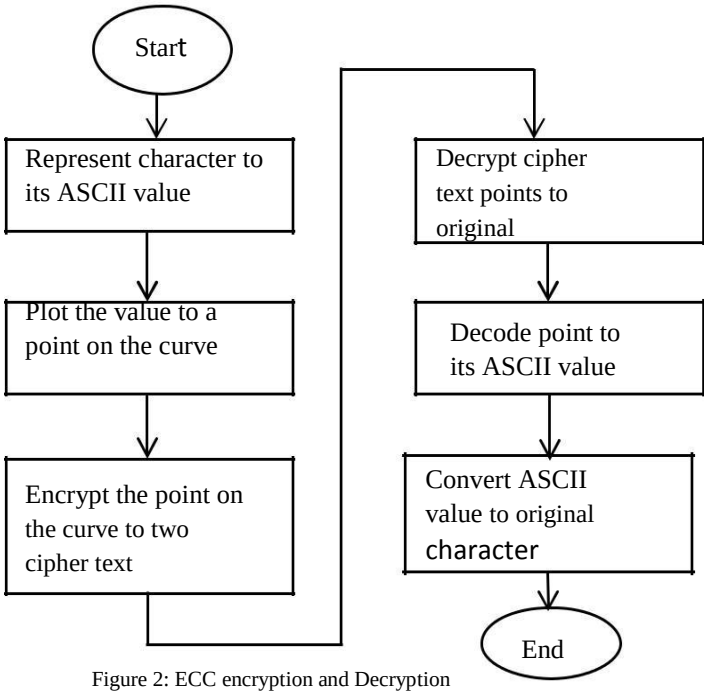

Figure 2: ECC encryption and Decryption

Table 1Definition of parameters

Symbol	Description
x&y	Elements of finite field
a&b	Each value of _a‘ and _b‘ gives a different elliptic curve
D	The random number that we have selected within the range of (1 to n_1)
P	P is point on the curve
Q	Q‘ is public key to be generated.
C_1&C_2	Two cipher texts
K	Randomly select _k‘ from [1 – (n_1)]
M	M is original message that was send.

algorithm using Q as the public key to get cipher text C_1 and C_2.

Message Embedding: After encryption the message is embedded into an image at the sender side and send to the authenticated receiver.

Message Decryption: At the receiver‘s side the message is extracted and decrypted using ECC private key that was transmitted using a secure network.

Blowfish is a symmetric block algorithm. The key length is variable ranges from 32 to 448 bits, default 128 bits key length. Blowfish uses 64-bit block size; 16 rounds of encryption is used in blowfish implementation, and are not susceptible to attack.

Image encryption: After the secret data is embedded, the cover image is encrypted using Blowfish algorithm to provide double security.

Image decryption: At the receiver side the image is decrypted and then the secret data is extracted.

V. RESULTS

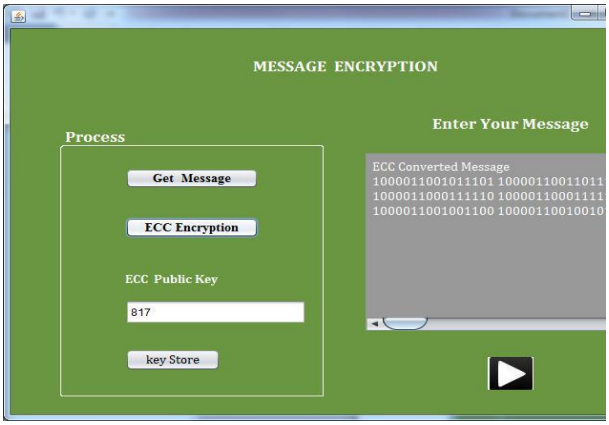

Figure 3: Message encryption using ECC

The secret message s read from the user and encrypted using ECC encryption public key to generate private key for the decryption.

The secret message s read from the user and encrypted using ECC encryption public key to generate private key for the decryption. Figure 4 shows the graph for the secret message that is encrypted using the graph.

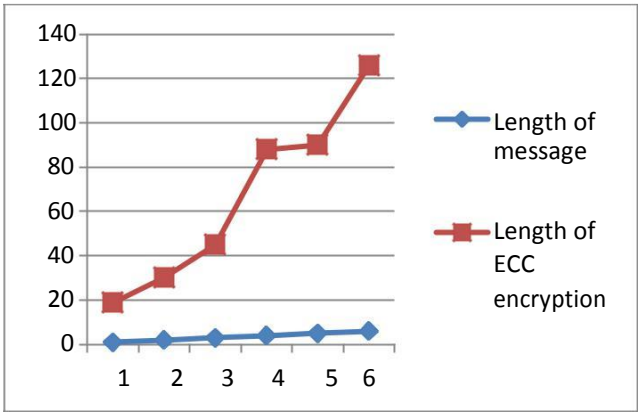

Figure 4: ECC encryption

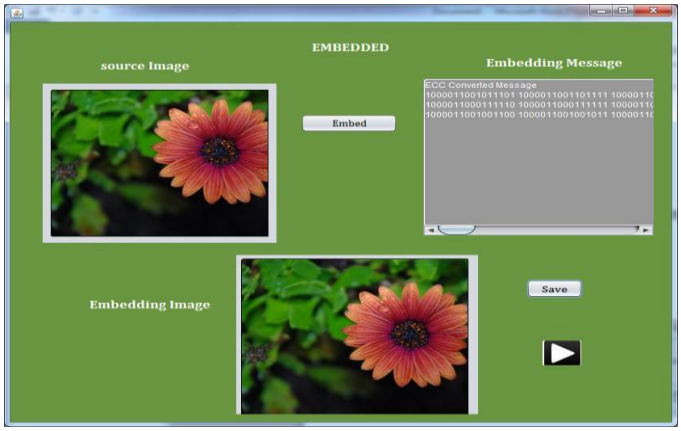

Figure 5: Message embedding into an image

Encrypted message is embedded into a selected image to get an embedded image and that image is shared with the authenticated receiver.

Figure 6: Image encryption using Blowfish

Figure 7: Recovery of original image

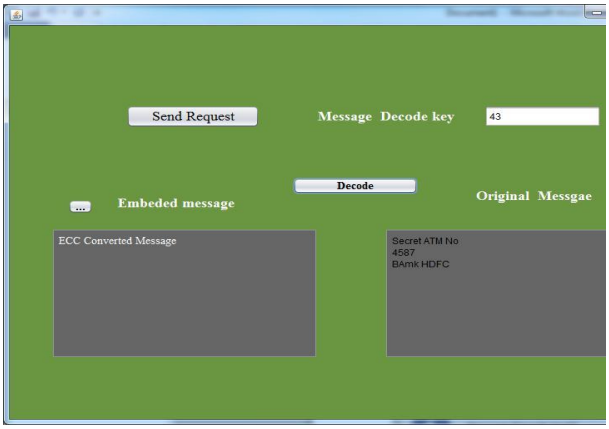

Figure 8: Message decryption at the receiver side

At the receiver side the message is decrypted using private key of ECC algorithm to get the originally sent secret message.

VI. CONCLUSION

This paper proposes a reversible message hiding scheme in an encrypted image which increase the embedding payload in an image and lossless recovery of data at the receiver side. In the proposed method, an image is selected and user‘s secret message is read and that secret message is encrypted using ECC algorithm using public key and also the cover image is encrypted using
Blowfish algorithm at the sender‘s side, during decryption the image is decrypted, the message is extracted from the image and decrypted using ECC generated private key at the receiver‘s side to get the lossless recover of original message.

VII. AUTHORS ACKNOWLEDGEMENT

I am highly thankful to my guide S.J. Prashantha B.E, M.Tech. Assistant Professor Department of CS&E, Adichunchanagiri Institute of Technology, Chikkmagaluru-577102 for his consistent support & guidance all through the paper. I also thank all those who directly or indirectly assisted in the successful completion of the paper.

REFERENCES

[1] zhenxing qian et.al, ―reversible data hiding in encrypted images with distributed source encoding‖, IEEE transactions on circuits and systems for video technology, vol. 26, no. 4, April 2016.
[2] Zhang, G. et.al, ―Scalable coding of encrypted images,‖ IEEE Trans. Image Process., vol. 21, no. 6, pp. 3108–3114, Jun. 2012.
[3] k. Ma, et.al, ―reversible data hiding in encrypted images by reserving room before encryption,‖ IEEE trans. Inf. Forensics security, vol. 8, no. 3, pp. 553–562, mar. 2013.
[4] Hong, et.al, ―An improved reversible data hiding in encrypted images using side match,‖
[5] IEEE Signal Process. Lett, vol. 19, no. 4, pp. 199– 202, Apr. 2012.
[6] Tejal Mahajan et.al,‖ Enhancing Blowfish File
[7] Encryption Algorithm through Parallel Computing on GPU‖, IEEE International Conference on Computer, Communication and Control (IC4-2015).
[8] Fatima Amounas et.al, ―Secure Encryption Scheme of Amazigh Alphabet Based ECC Using
[9] Finite State Machine‖, 978-1-4799-0324-5/13/$31.00_2013 IEEE.
[10] Ni, et.al, ―Reversible data hiding,‖ IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 3, pp. 354–362, Mar. 2006.
[11] X. Zhang, ―Reversible data hiding in encrypted image,‖ IEEE Signal Process. Lett., vol. 18, no. 4, pp. 255–258, Apr. 2011.
[12] W. Zhang, et.al, ―Reversibility improved data hiding in encrypted images,‖ Signal Process., vol. 94, pp. 118–127, Jan. 2014.
[13] [10]https://bithin.wordpress.com/2012/02/22/simpl e explanation-for-elliptic-curve-cryptography-ecc/
[11]https://en.wikipedia.org/wiki/Elliptic_curve_cr yptography
[12]https://en.wikipedia.org/wiki/Blowfish(cipher)