# Performance Analysis of Biometric Image Encryption in Transformed Formats using Public Key Cryptography

Dr.M.Gobi, Mrs.R.Sridevi

**Abstract**— Due to the prompted growth of multimedia applications, the preservation of multimedia data becomes very important. The security of multimedia images can be achieved through Cryptography. This type of image encryption requires more attention towards its implementation. There are many different cryptographic techniques used for image encryption. Elliptic Curves Cryptography is believed to provide high security with smaller key sizes, which is very useful in many applications. This paper proposes the transformation of images into Base64 and Binary formats before image encryption. The transformed image is encrypted using Elliptic Curve Cryptography algorithm and encrypted image is stored in the database in a safe manner for future references. Based on the experiments done, Base64 transformation performs better than Binary transformation.

**Index Terms**— Image encryption, Base64 Transformation, Binary Transformation, Authentication.

## 1 Introduction

CURRENTLY, information security is becoming more more significant in data storage and transmission. Images Images are broadly used for several processes. With the the increasing progression of multimedia data based applications, security is important in storing and communicating images, and Encryption is a prevalent technique to support image security [1]. Traditional image encryption techniques convert the original image into another image that is rigid to understand; to keep the image private between users, it is crucial that no third party could try to know the image content without knowing a key for decryption. Hence, to preserve image data from unauthorized access is much required. Image encryption is very much important in the field of data hiding. Image hiding or encryption algorithms start from spatial domain methods to more intricate and consistent frequency domain methods. Nowadays, plenty of color image encryption methods have been suggested. Various data encryption algorithms have been anticipated and widely used, like AES, RSA, or IDEA most of which are used in text or binary data [2] [3]. It is challenging to use them directly for multimedia data and incompetent for encrypting color images because of the high correlation among pixels. For multimedia data are often of high redundancy, of large volumes and require real-time interactions.

- *Dr.M.Gobi is currently working as an Assistant Professor in the department of Computer Science, Chikkanna Government Arts College, Tirupur, Tamilnadu, India. E-mail:mgobimail@gmail.com*
- *Mrs.R.Sridevi(Corresponding author) is currently pursuring Ph Din Computer Science. She is also working as an Assistant Professor in Computer Science Department, PSG College of Arts & Science, Coimbatore, Tamilnadu, India. E-mail:srinashok@gmail.com*

## 2 Image Transformations

Image encryption [4 - 6], as discussed is a tedious task to encrypt the image directly and convert it into cipher text. It includes the manipulation of extremely correlated pixels, which involves a higher level of duplication and the larger sizes. Furthermore, the encryption algorithms, both symmetric and asymmetric algorithms are widely meant for text or data. This nature of cryptographic algorithms is unsuitable for multimedia images and will be improved accordingly. Hence, the image is transformed into other formats using the available encoding schemes. This paper discusses two different types of encoding of images, Base64 encoding and Binary encoding.

### 2.1 Base64 Encoding

Base64 is a common term for a number of encoding schemes that encode binary data into a base64 representation. The Base64 encoding is originated from Multipurpose Internet Mail Extension encoding. This type of encoding schemes is popularly used whenever there is a requirement to encode data that needs be stored and transferred over the media. This is to make sure that the data remain as such without modification during transmission. Base64 is used in so many applications including email through MIME, and XML data storage. The encoding scheme is briefed here:

1. Store the image (.png format in our scenario) in a buffer.
2. Convert the image into byte array.
3. Write the buffered image into byte output array.
4. Encode the byte array obtained in Base 64 encoding.

Then we store the byte array for image as a text file.

### 2.2 Binary Encoding

Binary files are usually recognized as a sequence of bytes, in which binary digits (bits) are grouped in eights. Binary files usually contain bytes that are anticipated as something other

than text characters. Some of the examples of binary files include compiled computer programs and compiles applications. But binary files may contain not only text data, but images, sounds, compressed portions of other files, etc., in other words, any type of file content whatsoever. Here, the binary digits are read and written into a text file for ECC encryption of images.

## 3 Elliptic Curve Cryptography Algorithm

Elliptical curve cryptography (ECC) is an asymmetric cryptographic technique, which is based on elliptic curves which can be used to create smaller and more efficient cryptographic keys in a shorter period [7 - 9]. ECC generates public and private keys through the elliptic curve properties and its equation in place of the traditional method wherein the keys are generated from the product of very large prime numbers. With reference to previous studies [10 – 13], ECC supports the same level of security with a 164-bit key where the other systems normally use a 1,024-bit key. This property of ECC helps achieve the same level of security with low computing power and resource usage. It is now widely used in various real time applications14]. Several discrete logarithm-based protocols have been improved to elliptic curves, which replaces the group with an elliptic curve: the elliptic curve Diffie–Hellman (ECDH) key agreement scheme that is based on the Diffie–Hellman algorithm, the Elliptic Curve Integrated Encryption Scheme (ECIES), also known as Elliptic Curve Augmented Encryption Scheme or simply the Elliptic Curve Encryption Scheme, the Elliptic Curve Digital Signature Algorithm (ECDSA) is based on the Digital Signature Algorithm. The ECIES Scheme for encryption is explained here:

### 3.1 Key Generation

Key generation is an important part where we have to generate a public and private key for each user. Here, Elliptic curve cryptography algorithm is used to generate both of these keys. The encryption of the fused image is done using a public key and the same will be decrypted using the corresponding private key. For this key pair generation, we will select a number d, within the series of 'n', which is the random number representing the maximum limit. The public key, Q can be generated using the following formula.

$$Q = d * P$$

d = the random number that within the range of (1 to n-1).
P = the point on the curve.
Q = public key
D = private key.

### 3.2 Encryption & Decryption

The Elliptic Curve Integrated Encryption Scheme is as follows [12].

**To encrypt,**
**Step 1:** Select a random integer r in [1, n– 1]
**Step 2:** Compute R = rG
**Step 3:** Compute K = $hrQ_B = (K_x, K_y)$,
checks that K ≠ 0
**Step 4:** Compute keys k1 | |k2 = KDF($K_x$),
where KDF is a key derivation function.
**Step 5:** Compute c = $ENC_{k1}(m)$,
where m is the text file of fused image
**Step 6:** Compute t = $MAC_{k2}(c)$,
where MAC is a message authentication code
**Step 7:** Store (R, c, t) in the database for decryption.
**To decrypt,**
**Step 1:** Perform a partial key validation on R
(Check if R ≠ 0, check if the coordinates of R are properly represented elements in $F_q$ and check if R lies on the elliptic curve defined by a and b)
**Step 2:** Compute $K_B = h.d_B.R = (K_x, K_y)$,
check K ≠ 0
**Step 3:** Compute $k_1, k_2$ = KDF ($K_x$)
**Step 4:** Verify that t = $MAC_{k2}(c)$
**Step 5:** Compute m = $ENC_{K1}$-1(c); K = KB,
since K = h.r.QB = h.r.dB.G = h.dB.r.G = h.dB.R = KB

## 4 Proposed System

Most of the cryptographic algorithms, both symmetric and asymmetric are widely used in text or binary data [15]. As all the cryptography algorithms are meant for text data, it may be difficult to use them directly for encrypting the images and is not efficient enough in multimedia data or color images encryption. The proposed system suggests a new approach wherein, an image without being encrypted directly, is transformed or encoded into the Base64 encoding and Binary encoding. These two different transformations give two different text files of different sizes for the same image. This transformed file is encrypted using ECC Encryption algorithm to convert it into a cipher text file using a public key. The cipher text file is stored in a database for future manipulation. If and when required, the encrypted file is decdrypted using its corresponding private key and used to get back its original image used for its original purposes.

## 5 Experimental Results and Discussion

The random color images with the same dimension have been taken for sampling. The size of the sampling is 20 and they are random in nature. The samples are taken in .png format. For the convenience of transformation of images into other file formats, the images are taken in same format. The image file after transformed into other encoded formats has been analyzed for comparing the performance of each encoding scheme in combination with Elliptic Curve Cryptography. The various color images are taken into consideration. The .png image file formats with 256 x 256 pixel dimensions are invariably considered here in comparing Base64 encoding with Binary encoding before and after ECC Encryption. The parameters which are considered for this comparative study includes the image size, time taken for encoding, encoded file size, time taken for encryption, encrypted file size, time taken for decryption, decrypted file size etc., The size of the original image after encoding is considerably increased in both the schemes and is found to be comparatively smaller in Base64 encoding

than Binary Scheme as shown in Figure 1.

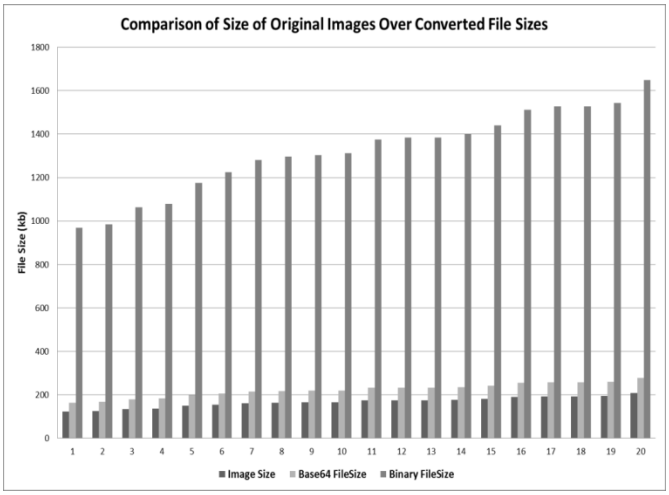


Fig 1. Comparison of size of original images over Transformed Files.

The other important parameters considered in this comparative study are the encryption time and decryption time for the images after transformed using these encoding schemes. As already discussed, the transformed images, while storing as a text file is a large string file and is very large is size.

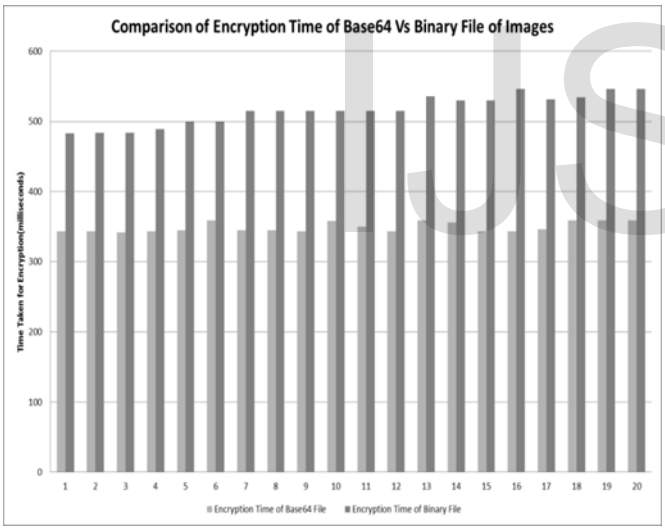


Fig 2. Comparison of Encryption Time of Base64 Vs Binary Files.

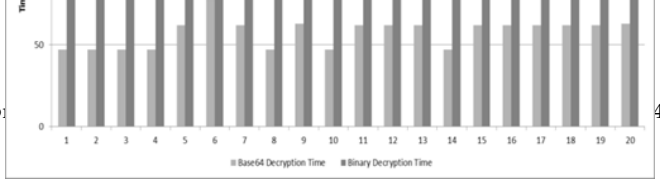


Fig 3. Comparison of Decryption Time of Base64 Vs Binary Files.

The encryption time reflects the size of the input text file which may either be in Base64 string or Binary String. The implementation shows that the encryption time shown in Figure 2 and the decryption time shown in Figure 3, Base64 takes very lesser than the time taken for Binary string.

Figure 4 and Figure 5 explains the sizes of files before and after the ECC Encryption scheme. The size of transforming images highly affects the size of its corresponding cipher text file. The input to the ECC encryption algorithm is the transformed text file (Base64 or Binary) and the sizes of both the transformations are analyzed.

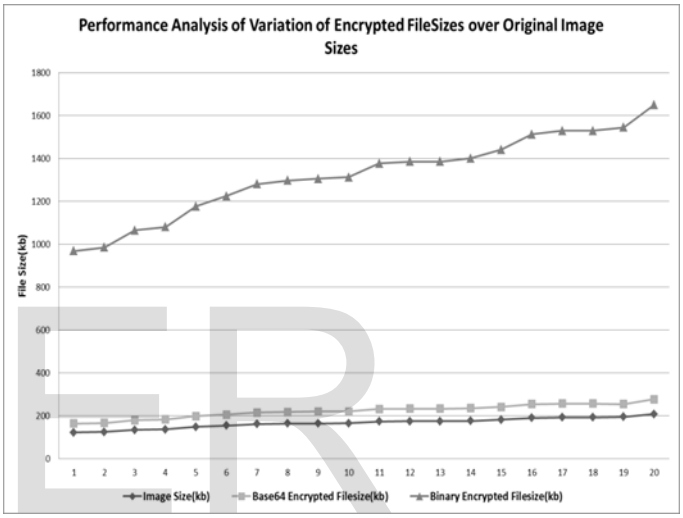


Fig 4. Performance Analysis of Variation in encypted FileSizes over Original Image Size.

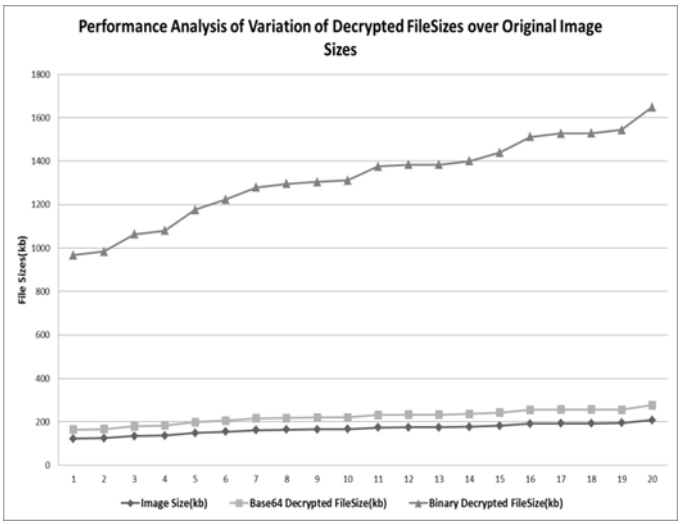


Fig 5. Performance Analysis of Variation in Decrypted FileSizes over Original Image Size.

## 6 CONCLUSION

Image encryption is a very encouraging field to work to find a more cost effective method to perform encryption for multi-media data. Elliptic Curve Cryptography has proven for its smaller key sizes with higher performance when compared with other asymmetric algorithms. This paper discusses an approach where the images are transformed into two different formats Base64 and Binary. The transformed images are encrypted using Elliptic Curve Cryptography algorithm and are stored in the database. The encrypted images are retrieved from the database when required and decrypted by the authenticated user. The performance of the approach is analyzed taking size and time as two critical factors. By the experiments done, it is shown that, in the Base64 transformation of images, the image before and after transformation has a very small increase in its size, whereas it is significantly larger in binary transformation. The time taken for the transformation also supports Base64 because binary transformation takes much higher time than Base64. When encrypted with ECC, the size of the encrypted image is very large in Binary transformation. Image encryption using Base64 has resulted in comparatively lower size files which facilitates lower memory occupation and faster execution. These comparisons demonstrate the appeal of Image encryption using ECC with Base64 transformation especially for applications that require high security for images.



## REFERENCES

[1] Hiral Rathod, Mahendra Singh Sisodia, Sanjay Kumar Sharma, "Design and Implementation of Image Encryption Algorithm by using Block Based Symmetric Transformation Algorithm (Hyper Image Encryption Algorithm)", International Journal of Computer Technology and Electronics Engineering (IJCTEE), Volume 1, Issue 3.

[2] M.Gobi and Dr.K.Vivekanandan, "A New Digital Envelope Approach for Secure Electronic Medical Records", IJCSNS International Journal of Computer Science and Network Security, Vol.9 No.1, January 2009.

[3] Ramachandran Ganesan, Mohan Gobi, and Kanniappan Vivekanandan, "A Novel Digital Envelope Approach for A Secure E-Commerce Channel", International Journal of Network Security, Vol.11, No.3, PP.121{127, Nov. 2010.

[4] Komal D Patel, Sonal Belani," Image Encryption Using Different Techniques:A Review", International Journal of Emerging Technology and Advanced Engineering, ISSN 2250-2459, Volume 1, Issue 1, November 2011.

[5] Rajinder Kaur, Er.Kanwalprit Singh," Image Encryption Techniques:A Selected Review", IOSR Journal of Computer Engineering (IOSR-JCE) e-ISSN: 2278-0661, p- ISSN: 2278-8727, Volume 9, Issue 6 (Mar. - Apr. 2013), PP 80-83.

[6] Dr M.Gobi and D.Kannan," A Secured Public Key Cryptosystem for Biometric Encryption", International Journal of Computer Science and Information Technologies, Vol. 5 (1) , 2014, 184-191.

[7] Kamlesh Gupta and Sanjay Silakari," Performance Analysis for image Encryption using ECC", International Conference on Computational Intelligence and Communication Networks, 2010.

[8] Wenjun Lu, Avinash L. Varna, and Min Wu, "Security Analysis for Privacy Preserving Search of Multimedia", Proceedings of IEEE 17th International Conference on Image Processing, September 26-29, 2010, Hong Kong.

[9] Mahalakshmi. U and Shankar Sriram V.S, "An ECC based Multibiometric System for Enhancing Security", Indian Journal of Science and Technology, vol 6, April 2013.

[10] Bibhudendra Acharya, Sarat Kumar Patra, and Ganapati Panda," Image Encryption by Novel Cryptosystem Using Matrix Transformation", First International Conference on Emerging Trends in Engineering and Technology, National Institute of Technology Rourkela, 2008.

[11] R. Ganesan, M. Gobi and V.S. Janakiraman, "Implementation of MD5 Integrity Checking Mechanism for M-Commerce Transactions", International Journal of Computer Science and Applications, Vol.1, No.3, pp.194 -196, 2008.

[12] R. Ganesan, M. Gobi and K. Vivekanandan, "Elliptic and Hyperelliptic Curve Cryptography Over Finite Field Fp", i-Manager's Journal on Software Engineering, Vol. 3, No. 2, pp. 43-48, 2008.

[13] Dr.M.Gobi and Mrs.R.Sridevi, "Reversible Data Embedding Using Asymmetric Cryptosystem", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 8, pp. 623 – 626, August 2013.

[14] Dr.M.Gobi and Mrs.S.Selvi, " A New Design and Implementation of Mixer Hashing Mechanism for Mobile Transactions", International Journal of Advanced Research in Computer Science and Software Engineering, Volume

[15] Dr. M. Gobi and Mrs. R. Sridevi, "An Approach for Secure Data Storage in Cloud Environment", *International Journal of Computer and Communication Engineering*, Vol. 2, No. 2, pp. 206-209, March 2013