INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 5, Issue 2)
Available online at: www.ijariit.com

Comparative analysis of ECC and Hessian Elliptic curve based message cryptosystem

Palaka Venkata Gangadhar Naveen
pvg413@gmail.com
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

Nayana Rao
nnayanarao@gmail.com
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

Dr. R. Sivaranjani
sivaranjani.cse@anits.edu.in
Anil Neerukonda Institute of Technology and Sciences, Visakhapatnam, Andhra Pradesh

ABSTRACT

Security is a major concern is nowadays, in many applications many algorithms were proposed majorly categorized into symmetric and asymmetric key cryptographic algorithms. The limitations with symmetric key cryptography are the scope of knowing the shared key between the communicators to intruders. Hence, asymmetric key cryptography algorithms were preferred, among those ECC is one best mechanism. This paper is aiming at the analysis of time complexity, efficiency and security levels of Elliptic Curve Cryptography (ECC) and based upon the analysis, further extension to a strong encryption and decryption algorithm. In addition to that ECC will be extended to other ECC based derived algorithms include a Hessian algorithm. Due to the many important properties making the model attractive in cryptograph, the paper code is implemented in the Magma software to ensure the correctness of formulas in this work. A comparative analysis will be done among the ECC and Hessian algorithm.

Keywords— *ECC, Hessian curve, Encryption, Decryption, Security analysis*

1. INTRODUCTION

The cryptographic algorithms is an important role to ensure confidentiality, integrity, authentication, and non- repudiation of the transmitted data. ECC has a number of advantages over other public-key cryptosystems, such as RSA, which make it an attractive and alternative to security. In particular, for a given level of security, the size of the cryptographic keys and operands involved in the computation of EC cryptosystems is normally much shorter than other cryptosystems.

Cryptography algorithms suggested the best solution for many communication network systems by providing by offering security features such as confidentiality, data integrity, authenticity and non-repudiation. Elliptic curve cryptography (ECC)[4] is one of the best public key cryptography technique which offers optimal solution to many constrained environment such as power, bandwidth, speed, small device etc., Public Key Cryptography consists of a pair of keys, named a public key and a private key which would be used to operate encryption and decryption of data. Each end user generates these key pairs where private key knows only by a particular user and whereas the public key is distributed to all users taking part in the communication. ECC based cryptographic algorithm requires many predefined constant parameters to be known by the entire user in the communication system. These parameters are used to perform encryption and decryption operation during transmission of data. ECC require lesser key size than another cryptosystem. Unlike symmetric key cryptosystem, it does not share a secret key between the end users which will tend to insecure communication. But symmetric key cryptosystem [2,3] is much faster than public key cryptosystem due to the generation of key pairs. ECC generates keys through the properties of the elliptic curve equation instead of the traditional method of generation as the product of very large prime numbers (RSA). ECC [1] can yield a level of security with a 164-bit key that other systems require a 1024-bit key to achieve.

2. ELLIPTIC CURVE CRYPTOGRAPHY

Elliptic Curve Cryptography [5] is one of the famous public key cryptographic technique was independently proposed by Miller and Koblitz in 1985. It uses elliptic curve where variables and coefficient are bounded to elements of a finite field. These researchers put an enormous amount of work to offer the same level of security with lesser key size compared with existing methods which are based on difficulties of solving discrete logarithm problem over integers or integer factorization.
A polynomial equation of elliptic curve:

$$y^2 = x^3 + ax + b \quad (1)$$

2.1 Weierstrass formulation of elliptic curves

From the mathematical standpoint, an elliptic curve is a solution set to the bivariate polynomial equation $f(x,y)=0$, Where $f(x,y)$ is of total degree 3, and $f(x,y)$ is irreducible.

In cryptography consider particular equations and particular (finite) fields over which curves are defined. The points on the curve form a commutative group. What makes elliptic curves, particularly attractive for cryptographic applications is that the discrete logarithm problem in elliptic curve groups is computationally hard. Moreover, it is "harder" than in groups, previously considered, thereby allowing shorter key lengths.

An elliptic curve E_{ab} defined over a prime field F_p (with p>3) can be written in the simplified Weierstrass form as:

$$\boldsymbol{E_{ab}(F_p): y^2 = x^3 + ax + b} \tag{2}$$

Where $a,b \in F_p$

2.2 Point addition

Consider two distinct points J and K such that $J = (X_J, Y_J)$ and $K = (X_K, Y_K)$. Let $L = J + K$ where $L = (X_L, Y_L)$, then

$$X_L = S^2 - X_J - X_K \; mod \; p \tag{3}$$
$$Y_L = -Y_J + S(X_J - X_L) \; mod \; p \tag{4}$$
$$S = (Y_J - Y_K)/(X_J - X_K) \; mod \; p \tag{5}$$

Where S is the slope of the line through J and K. If $K = -J$ that is, $K = (X_J, -Y_J \; mod \; p)$ then $J + K = O$. where O is the point at infinity. If $K = J$ then $J + K = 2J$, then point doubling equations are used. Also $J + K = K + J$

2.3 Generation of elliptic curve points

The equation of Elliptic curve prime field is used to generate the elliptic curve points which help to generate private and public key pairs. If $x=0, y=1\ldots\ldots.p.$

To Substitute $x=1, y=0; a=1; b=1; p=23$ in equation 1.

$$y^2 mod \; p = (x^3 + ax + b) mod \; p$$
$$0 \; mod \; 23 = \big(13 + 1(1) + (1)\big) mod 23$$
$$0 \quad = \quad 3 \; mod \; 23$$
$$0 \quad = \quad 3$$

Likewise to find the points when both sides will be equal then it should be a point. Like $x=1; y=7; a=1; b=1; p=23$ in the equation.

$$7^2 mod \; 23 = 3 \; mod \; 23$$
$$3 \quad = \quad 3$$

Then (1, 7) is one of the elliptic curve points in the table. This process continued until to generate p points by varying x and y values as shown in table 1. These points are used to draw the elliptic curve as shown in figure 1. In the above-mentioned curve, an elliptic curve over prime field equations are chosen, to perform point scalar point multiplication which requires both point addition and doubling operation.

Table 1: Points on the elliptic curve E_{23} (1,1)

(0, 1)	(6, 4)	(12, 9)
(0, 22)	(6, 19)	(13, 7)
(1, 7)	(7, 11)	(13, 16)
(1, 16)	(7, 12)	(17, 3)
(3, 10)	(9, 7)	(17, 20)
(3, 13)	(9, 16)	(18, 3)
(4, 0)	(11, 3)	(18, 20)
(5, 4)	(11, 20)	(19, 5)
(5, 19)	(12, 4)	(19, 18)

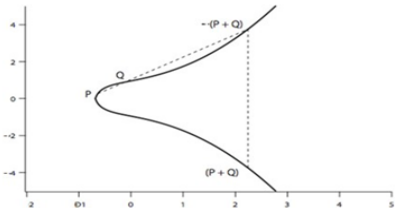

Fig. 1: Elliptic curve equation $y^2 = x^3 + x + 1$

3. HESSIAN CURVE CRYPTOGRAPHY

The hessian curve is a plane curve similar to folium of Descartes, named by Otto Hesse. This curve was suggested for application in elliptic curve cryptography because arithmetic in this curve representation is faster and needs less memory than arithmetic in standard Weierstrass form. The hessian curve proposes the family of generalized Hessian curves with efficient, unified addition formulas for the purpose of safety transmission of information. A Hessian curve H_d defined over a field F is given by the cubic equation:

$$H_d: x^3 + y^3 + 1 = dxy$$

Where $d \in F$.

3.1 Point addition

Let $P_1 = (X_1:Y_1:Z_1)$ and $P_2 = (X_2:Y_2:Z_2)$ be two points distinct. Assuming that $Z_1=Z_2=1$ then the algorithm is given by:

$$A = X_1\ Y_2$$
$$B = Y_1\ X_2$$
$$X_3 = B\ Y_1\text{-}Y_2\ A$$
$$Y_3 = X_1\ A\text{-}\ B\ X_2$$
$$Z_3 = Y_2\ X_2\text{-}\ X_1\ Y_1$$

Example: Given the following points in the curve for $d=-1$ $P_1=(1:0:-1)$ and $P_2=(0:-1:1)$, then if $P_3=P_1+P_2$ we have:

$$X_3 = 0\text{-}1 = -1$$
$$Y_3 = -1\text{-}0 = -1$$
$$Z_3 = 0\text{-}0 = 0$$
Then: $P_3 = (-1:-1:0)$

3.2 Point doubling

Let $P = (X_1 : Y_1 : Z_1)$ be a point, then the doubling formula is given by:

$$A = X_1^2$$
$$B = Y_1^2$$
$$D = A + B$$
$$G = (X_1 + Y_1)^2 - D$$
$$X_3 = (2Y_1 - G) \times (X_1 + A + 1)$$
$$Y_3 = (G - 2X_1) \times (Y_1 + B + 1)$$
$$Z_3 = (X_1 - Y_1) \times (G + 2D)$$

The cost of this algorithm is three multiplications + three squarings + 11 additions + 3×2.

Example: If $\{P=(-1:-1:1)\}$ is a point over the Hessian curve with parameter d=-1, then the coordinates of $\{2P=(X:Y:Z)\}$ are given by:

$$X = (2.(-1)\text{-}2)(-1+1+1) = -4$$
$$Y = (-4\text{-}2.(-1))((-1)+1+1) = -2$$
$$Z = (-1\text{-}(-1))\ ((-4)+2.2) = 0$$
That is, 2P = (-4:-2:0)

4. ECC AND HEC BASED ENCRYPTION AND DECRYPTION

ECC algorithm involves to perform key generation, generation of elliptic curve points, mapping of plaintext into numbers, conversion of number into points, encrypt the plaintext points using the private key and finally decrypt the ciphertext points. These operations are performed by the following sequence of steps as shown below:

4.1 Key generation process

- User A generates a random private key $n_A < n$.
- User A also computes public key $P_A = n_A \times G$;
- $P_A = G+G+G...+ n_A$ times. (Point Addition)
- User B generates a random private key $n_B < n$.
- User B also compute his public key, $P_B = n_B \times G$;
- $P_B = G+G+G...+ n_B$ times. (Point Addition)

4.2 Encryption-Decryption process

The process of converting plaintext P_m into cipher text C_m known as encryption process. Since elliptic curve cryptography deals with elliptic curve point for key generation process. Each character of plaintext should be converted into points using Koblitz method. Then converted plaintext points are encrypted with the help of its generated keys to generate cipher text points (C_m).

$$C_m = \{k.G, P_m + k.P_B\} \quad (6)$$

Where,
G: Generator Point
P_m*:* Plaintext point on the curve
k: Random number was chosen by A
P_B*:* Public key of user B.

Elliptic curve decryption process: The cipher text point's C_m is converted into plaintext point P_m using Koblitz method.

$$P_m + k.P_B - n_B(k.G) = P_m + k(n_B)G - n_B(k.G) = P_m \quad (7)$$

5. RESULTS AND COMPARATIVE ANALYSIS

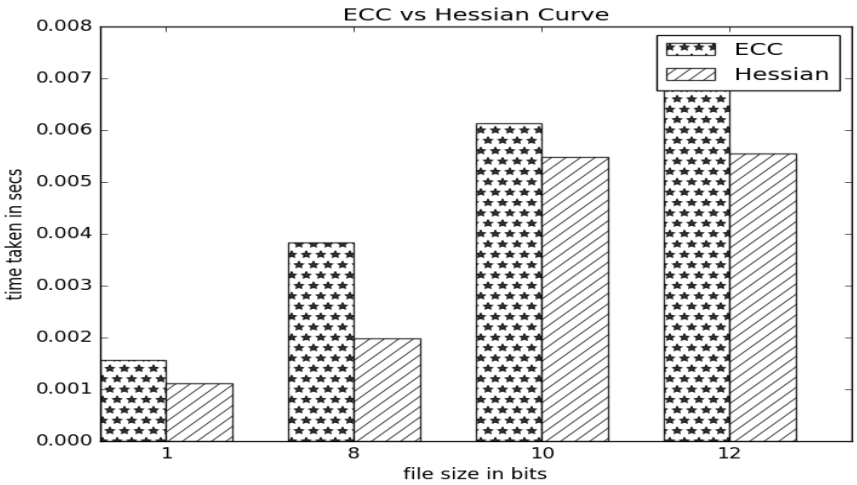

Fig. 2: Bar chart plotted for ECC vs. Hessian

Table 2: Comparison of file size vs. time complexity of ECC and Hessian Curves

ECC				HESSIAN			
Type	**Input**	**Size (bits)**	**Time (sec.)**	**Type**	**Input**	**Size**	**Time**
Char	s	1	0.001567	Char	s	1	0.001109
Password	Root@123	8	0.003833	Password	Root@123	8	0.001981
String	helloworld	10	0.006130	String	helloworld	10	0.000481
Aadhar	486055966880	12	0.0075089	Aadhar	486055966880	12	0.005547

6. HESSIAN CURVE SECURITY ANALYSIS

Side-channel attacks are a recent class of attacks that have been revealed to be very powerful in practice. By measuring some side channel information (running time, power consumption, and an attacker is able to recover some secret data from a carelessly implemented crypto algorithm. This paper investigates the Hessian parameterization of an elliptic curve as a step towards resistance against such attacks in the context of elliptic curve cryptography. The idea is to use the same procedure to compute the addition, the doubling or the subtraction of points. As a result, this gives a 33% performance improvement as compared to the best-reported methods and requires much less memory.

7. CONCLUSION

From this comparative analysis of elliptic curve and hessian curve, based on point generation, point addition and point doubling for both the curves, along with strong encryption and decryption algorithm and compared the results based on time complexities. It clearly shows that the hessian curve is better than an elliptic curve.

8. REFERENCES

[1] Emmanuel Fouosta "*Parallelising pairings on stress hassen elliptic curves",* Journal name: Arab Journal Of Mathematical Sciences, pass from 12 January 2018 to 5 June 2018 (Base Paper)
[2] Pairings for beginners by Craig Costello.
[3] https://www.math.uzh.ch/sepp/magma-2.19.8-cr/Handbook.pdf
[4] Comparitive analysis of Elliptic Curve Cryptosystem and its survey (www.jchps.com).
[5] R. Avanzi, H. Cohen, C. Doche, G. Frey, T. Lange, K. Nguyen, F. Vercauteren, Handbook of elliptic and hyperelliptic curve cryptography, Discrete Math. Appl. (2006).
[6] Amara M and Siad A, Elliptic Curve Cryptography and Its Applications, IEEE proceeding of International workshop on Systems, Signal Processing and their Applications, 2011, 247-250.
[7] Koblitz, Elliptic curve cryptosystems, Mathematics of Computation, 48, 1987, 2003–2009.
[8] Maria Celestin Vigila S and Muneeswaran K, Nonce Based Elliptic Curve Cryptosystem for Text and Image Applications, International Journal of Network Security, 14 (4), 2012, 236-242.
[9] Smart N.P, The Hessian form of an elliptic curve, Springer-Verlag Berlin Heidelberg, 2001.
[10] William Stallings, Cryptography and Network Security, 4th ed, Principles and Practices, Dorling Kindersley (India) Pvt, ltd, 2009.

