EXTENDED GENERALIZED ELGAMAL CRYPTOSYSTEM FOR SECURE M2M COMMUNICATION

Abstract

Elliptical Curve Cryptography (ECC) is one of the most prominent public-key cryptosystem based on the elliptical curve discrete logarithm structure over finite fields. The ECC elliptical curve is better-known than other public encryption systems for its smaller keys sizes, quicker encryption, better security and better safety (like RSA). ECC is suitable to secure data transfer (e.g. ElGamal), secure key sharing and authentication and verification of the digital signature (ECC Diffie-Hellman). The protection capability of the ECC is depends up on a trapping function, provided that the discrete logarithm of a random elliptic curve element cannot be found with regard to a publicly known base point. This is known as Elliptic Curve Discrete Logarithm Problem (ECDLP) it is treated to be mathematically difficult to solve.

The proposed scheme performs better for low cost devices such as nodes in M2M communication by means of computation complexity and storage space. In future the scheme can be extend for M2M communication in Ultra dense networks where heterogeneous nodes may authenticate and perform secure communication. The proposed scheme is also verified formally by using verifier tool ProVerif and the performance is also compared with contemporary security algorithms.

Keywords: Elliptic Curve Cryptography; M2M Communications; ECDLP.

Authors

Dr. B. Satyanarayana Murthy
Professor of CSE
BVC Engineering College
Andhra Pradesh, India
bsnmurthy2012@gmail.com

Dr. Tumma Srinivasarao
Associate Professor of CSE
SR Gudlavalleru Engineering College
Andhra Pradesh, India

Dr. P. B. V Raja Rao
Assistant Professor of CSE, Sri Vishnu Engineering College for Women
Bhimavaram, Andhra Pradesh, India

Dr. K. Rajasekhar
Professor of ECE
BVC Engineering College
Andhra Pradesh, India

I. INTRODUCTION

The safety of the ElGamal data encryption algorithm depends on a discrete log problem over a finite field. Today, ElGamal Scheme or variant is used by a lot of cryptography applications. In the IND-CPA, security was demonstrated by the ElGamal system, but vulnerability was shown by the CCA2 attack in the early 1995. Therefore, cryptography community is very much interested in improving ElGamal encryption [18]. Due to its safe, efficient and low complexity characteristics, the cryptographic community used it especially for mobile communication security. The elliptical curve cryptosystem has certain advantages compared to the public-key cryptosystem, including lower computational capability, small storage, and narrow bandwidth. Therefore it results in good efficiency by combining ElGamal encryption algorithm and elliptic curves. The reliable transmission of data in communications is an important challenge [1].

In addition to RSA and Rabin cryptosystems, ElGamal, named past Taher ElGamal, is another public key cryptosystem. ElGamal's logarithm problem is discrete. If p is an important prime number, e1 is primitive root in the group. G=<Zp,*, X> and r is an integer, then e2=e1r mod p is easy to compute using the fast exponential algorithm, but given e1, e2, and p, it is infeasible to calculate the discrete logarithm problem i.e, $r = \log_{e1} e2 \bmod p$. Figure 1 shows the parameters computation, of Gamal

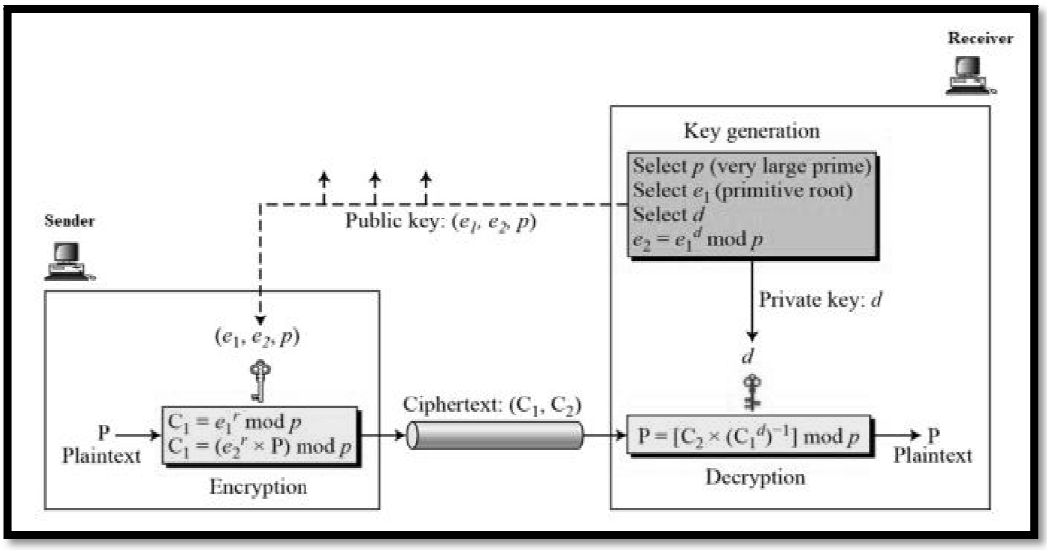

Figure 1: ElGamal parameter computation.

1. **Elliptic Curve Cryptosystem:** While RSA is a secure asymmetrical cryptosystem, it large key size and is costly. Researchers have sought alternatives that give smaller key sizes that offer the same level of security. The Elliptical Curve Cryptosystem (ECC) is one of the famous algorithms. The system relies on Elliptic Curve theory. A brief introduction about elliptical curves is:

    - **Elliptic Curves using real numbers:** This type of elliptic curves consists of two variable cubic curves, same as the variables used in the elliptical circumference for the calculation of the curve lengths, which are not directly connected with ellipses. The general form of the elliptical curve equation is shown in equation 1.

$$y^2 + b_1xy + b_2y = x^3 + a_1x^2 + a_2x + a_3 \ldots\ldots (1)$$

Elliptic curves based on real numbers utilize a particular category of elliptic curves as shown in equation 2:

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

The above equation is a nonsingular curve if $4a^3 + 27b^2 \neq 0$, otherwise the above formula represents a singular elliptical curve. The formula $x^3 + ax + b = 0$ has three separate roots (real or complex) in a nonsingular elliptic curve, and there are no three distinct roots in the equation $x^3 + ax+ b = 0$ in a particular elliptic curve. Looking at the formula, it is observed that there is a degree of 2 on the left, whereas 3 on the right. This means that if all the roots in a curve are real, a straight line can pass through the curve in 3 points. At most, in two points, a vertical line can cross the curve.

Figure 2 shows two elliptical curves, $y^2 = x^3 – 4x$ and $y^2 = x^3 – 1$.
They are non singular.
The first has one consists of three real roots (x = −2, x = 0, and x = 2), however; the other only has two imaginary one real root (x = 1).

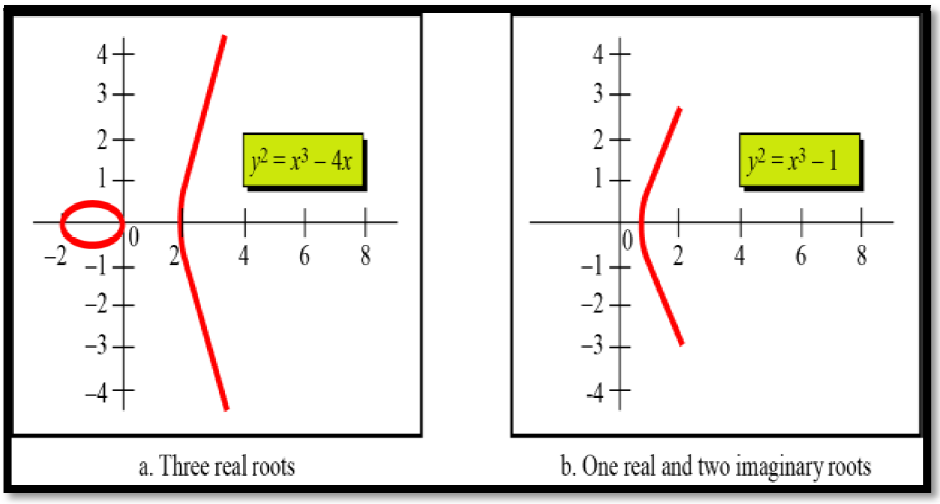

Figure: 2 Two curves across an actual field

The specific characteristics of an elliptical curve of non singular enable to identify an additional action at curve points. However, it is to be remembered that the supplementary procedure adopted here differs from the operation defined for the integers. The function adds two curve points on a given curve to obtain a new set of point on the curve as follows:

R = P + Q, where P = (x1,y1), and Q = (x2,y2), and R = (x3,y3)
Take three different cases to find R as shown below in the figure 3

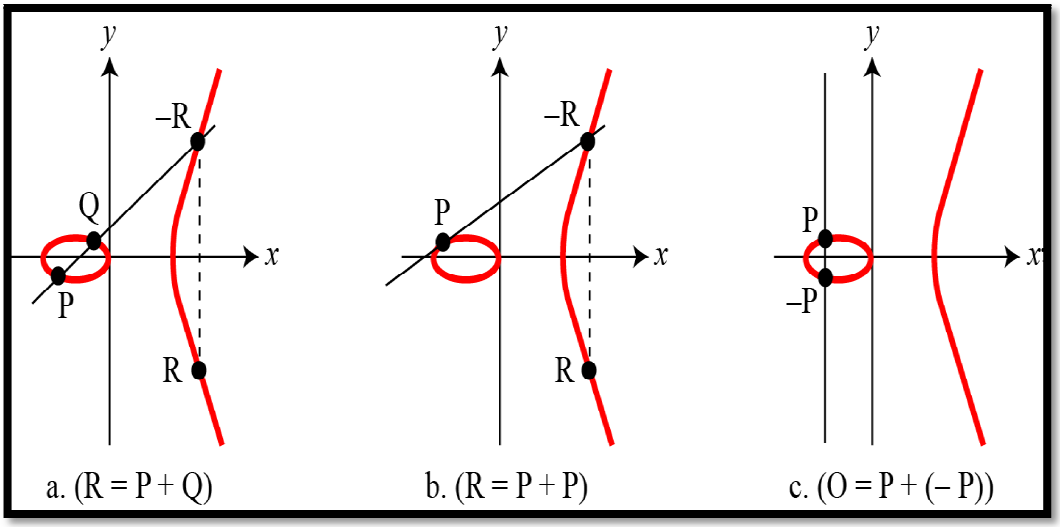

Figure 3: The 3 cases in an elliptical curve

- In the first case, there are different x-coordinates and y-coordinates in P = (x1, y1) and Q = (x2, y2). The P and Q line intercepts the curve at a location known as (–R). R reflects –R as far as the x-axis is concerned. Points of R also x3 and y3 are determined by foremost finding of the pitch of the λ then finding x3 and y3 values as shown below: The coordinates are computed using the equation 3:

$$\lambda = (y_2 - y_1) / (x_2 - x_1)$$
$$x_3 = \lambda^2 - x_1 - x_2 \quad \text{and} \quad y_3 = \lambda(x_1 - x_3) - y_1 \qquad \ldots\ldots (3)$$

- The two points overlap in the second case (R = P + P). In this case, as shown below, the line slope and the point R coordinates can be found using equation 4:

$$\lambda = (3x_1^2 + a) / (2y_1)$$
$$x_3 = \lambda^2 - x_1 - x_2 \quad \text{and} \quad y_3 = \lambda(x_1 - x_3) - y_1 \qquad \ldots\ldots (4)$$

- In case 3, the two are mutually additive inverses. The second point is Q = (x1, -y1) if the first point is P = (x1, y1). The third point does not obtained by intercepting the two points of a line. The mathematicians say that interception is not finite (O) or also known as point at infinity or zero point.

- **Elliptic curves over GF (p):** Modular arithmetic is necessary for cryptography. An elliptic curve 'E' operated additionally over the GF (p) field with p>3 and a, b € GF (p) is denoted as Ep (a, b) with *O* as extra point.

$$y^2 = x^3 + ax + b$$

- **Inverse Computation:** The contrary of a point (m, n) is (m, -n) where –n is the additive inverse of n. For instance, if q=13, the inverse of (4, 2) is (4, 11).
- **Curve Point Computation:** For computing the points at the Ep (a, b) on the curve the following pseudo code is used.

ellipticCurve_points (p,a,b) // p is the modulus

{

x ← 0

```
        while ( x < p )
        {
        w ← ( x^3 + ax + b ) mod p // w is y^2
        if ( w is a perfect square in Zp) output (x, √w), (x, −√w)
        x ← x + 1
        }
}
```

In other words, for example the equation is $y^2 = x^3 + x + 1$ and the computation will be carried out on modulo 13 and the points of elliptic curve over E_{13} (1,1) as shown in figure 4

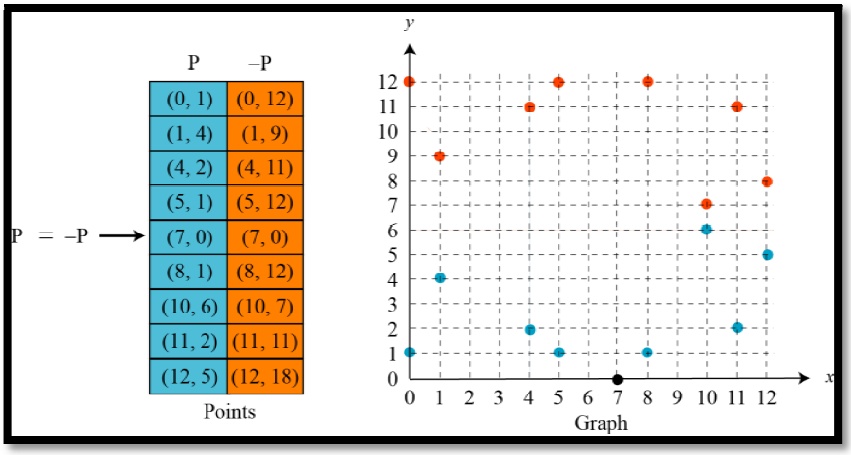

Figure 4: Elliptical curve points across GF (p)

Some y^2 values in arithmetic modulo 13 have no square root. These are not the elliptical points of curve. In the current curve it do not include points x = 2, 3 and 6 or x = 9. For each curve, each point has the inverse. The inverses are represented as pairs. It should be noted that the inverse of (7, 0) is (7, 0) itself. Note that in Zp of a specific a two of a kind of inverse points, the values of y are an additive reverse to each other. For instance, under Z13, 4, 9 are inverses under addition. So we could say y = 4, then –y = 9. On the same vertical lines, both are inverses.

- **Adding two points:** The earlier defined elliptical curve group equation (3) is used with a computation done on GF (p). The additive and multiplicative inverses are used instead of subtraction and division. For instance,

Let, R=P + Q where P = (4, 2) and Q = (10, 6) are the points,

a. $\lambda = (6 \quad 2) \times (10 \quad 4)^{-1} \bmod 13 = 4 \times 6^{-1} \bmod 13 = 5 \bmod 13$

b. $x = (5^2 - 4 - 10) \bmod 13 = 11 \bmod 13$

c. $y = [5(4 - 11) - 2] \bmod 13 = 2 \bmod 13$

d. R = (11, 2), which is a point on the curve

- **Point multiplication with a constant:** To multiply the number with a constant k in arithmetic means to add k times. In order to multiply a point P with the constant k on the elliptical curve, one point P is added in k. For example, in E_{13} (1, 1) this results in the increase of point (1,4) to 4 (5, 1). This leads to a multiplication of point (8,1) by three (10, 7).

- **Elliptic curves over GF (2^n):** The GF (2^n) field can define the calculations of the elliptical curve group. Adding and multiplying numbers are the same as adding and multiplying polynomials. To identify an elliptical curve over GF (2^n), the cubic equation desires to be changed. The commonly described equation is shown in equation 5:

$$y2 + xy = x3 + ax + b \qquad (5)$$

Where b is not equals to 0. Note that x, y values, a, and b polynomials have n-bit words.

- **Inverse Computing:** if P = (x, y), then –P = (x, x+y).
- **Curve Points:** For the irreducible polynomial of $f(x) = x^3 + x + 1$, select GF (2^3) with elements {0, 1, g, g^2, g^3, g^4, g^5}, that is similar to that $g^3 + g + 1 = 0$ or $g^3 = g + 1$. Additional 'g' powers can be accordingly calculated. The values for the g's are shown below.

0	000	$g^3 = g + 1$	011
1	001	$g^4 = g^2 + g$	110
g	010	$g^5 = g^2 + g + 1$	111
g^2	100	$g^6 = g^2 + 1$	101

With $y^2 + xy = x^3 + g^3x^2 + 1$, $a = g^3$, and using the elliptic curve b = 1, the following curve points are obtained.

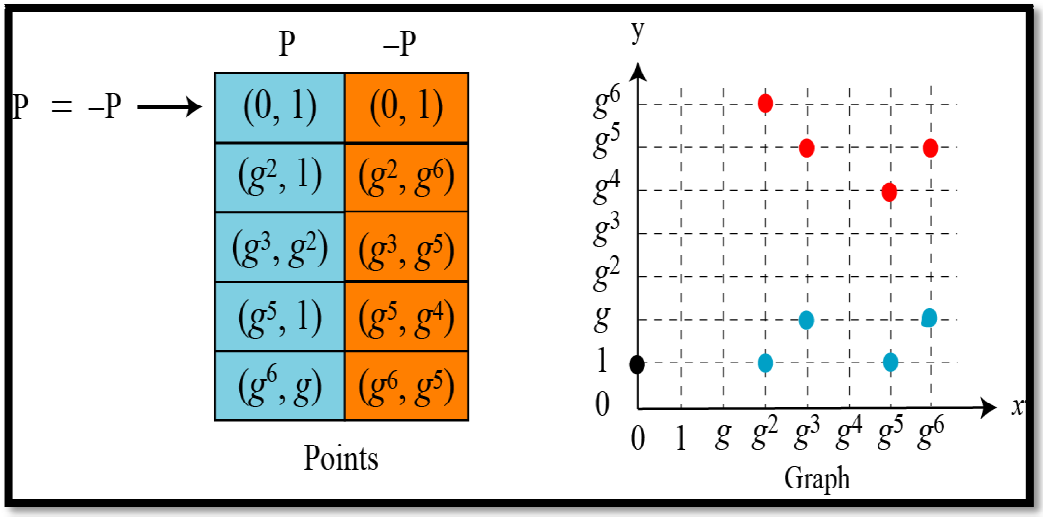

Figure 5: Elliptical curve points across GF (2^n)

Adding Two Points: GF (2n) rules for the addition of points are fairly different from the GF (p) and are shown in equations 6 and 7.

1. If $P = (x_1, y_1)$, $Q = (x_2, y_2)$, $Q \neq -P$ and $Q \neq P$, then $R = (x_3, y_3) = P + Q$ can be found as

$$\lambda = (y_2 + y_1) / (x_2 + x_1)$$
$$x_3 = \lambda^2 + \lambda + x_1 + x_2 + a \quad \text{and} \quad y_3 = \lambda(x_1 + x_3) + x_3 + y_1 \quad \ldots\ldots \quad (6)$$

2. If $Q = P$, then $R = P + P$ (or $R = 2P$) can be found as

$$\lambda = (x_1 + y_1) / x_1$$
$$x_3 = \lambda^2 + \lambda + a \quad \text{and} \quad y_3 = x_1^2 + (\lambda + 1) x_3 \quad \ldots\ldots \quad (7)$$

II. LITERATURE REVIEW

There are different approaches available for secure communication in constrained environments. Most widely used conventional and modern block cipher approaches for constrained environments are Advanced Encryption Standard (AES), Rivest–Shamir–Adleman (RSA). Apart from this, the other classical approaches also proposed by authors. The following are some of the existing approaches.

1. **ElGamal Encryption using Elliptic Curve Cryptography:** In future years computer systems are expected to be depends on Elliptical Curve Cryptography (ECC). Increased RSA key length may not help, because the process of encryption and decryption will also be slowed down. The ECC 256-bit equivalent to the RSA 3072-bit is considered. ECC is known to provide much more secure than RSA, while ECC encryption ensures the same security as RSA. It is nevertheless slow to encrypt the symmetric key (e.g. AES) and is therefore rare to be used in real message encryption. The ECC encoding of ElGamal may be described as an ElGamal cryptosystem using the Elliptical Curve arithmetic over a finite field. The authors researched on certain very significant feature of ECC for its application in cryptography and investigated the use of elliptical curves in ElGamal encryption and understand its challenges in data encryption. It is also a fast encryption approach and compared with other popular symmetrical and public key cryptosystems [2].

   Limitations to this approach are that the elliptic curve when visualized only a few points fall on the curve, i.e, all the Elliptic Curve points over the finite field cannot be implemented and visualized [3]. Hence, the normal elliptical curve and the finite field curve have to be referenced side by side for further simplicity of the basic arithmetic operations in an ECC. Other limitation is that it is not suitable for smaller devices, which means that the methodology needs to be generalized.

2. **CLEFIA:** CLEFIA is a proprietary block cipher algorithm devised by Sony. Its name comes from the French word clef, that means “key”. The size of the block is 128 bits, the key 128 bits, 192 bit or 256 bits. The method includes a new, compatible AES 128-bit CLEFIA block cipher supporting 128, 192 and 256-bit key lengths. CLEFIA [4] is able to achieve sufficient immunity from known attacks and flexibility by adopting a number of new and cutting-edge design techniques to achieve effective software and hardware implementation. In hardware and software, CLEFIA achieves good performances profile which is about 1.60 Gbps with less than 6K gates, about 13 cycles/byte of software with a CMOS ASIC library of 0,09 μm, 1.48 Gbps, and AMD Athlon 64 of 2,4 GHz. CLEFIA is a highly efficient block cipher, especially hardware.

The limitation of this approach is, as it is symmetric key cryptosystem, it is necessary to exchange the encryption keys before starting actual communication. Hence additional key exchange algorithms are needed.

3. **Lightweight authentication scheme using Elliptic curve cryptography for the communication:** Intelligent grid is a developed grid system for customer energy monitoring to adjust the amount of electricity generated. This advanced grid system can contribute to the promotion of cultural heritage by ensuring a continuous and intelligent supply of power. Intelligent grid is one of the main functionalities for smart cities, and every city with smarter facilities attracts visitors eventually to visit the rich heritage. Smart networks are in charge of safeguarding the basic communication between stations and the relevant control centre, by monitoring and acquisition of data using Supervisory Control and Data Acquisition (SCADA). While customer sub-communications require further improvements, existing protocols fail to satisfy full intelligent grid security requirements [5]. Due to the complex nature of the intelligent grid and various security requirements, the development of an appropriate authentication system is a vital challenge. An ideal authentication scheme should handle all known lightweight security attacks involving delay-sensitive networks like smart grid calculations. The ECC[6] offers the same level of safety with much smaller sizes compared to other safety techniques such as RSA, DSA and DH. Given the complex, time-consuming nature of the smart grid, the author proposes an ECC-based system of lightweight authentication. In addition to providing low calculation and communication costs for mutual authentication, the proposed system is tolerant to all known security attacks.

III. METHODOLOGY

In future years computer systems are expected to be based on Elliptical Curve Cryptosystem (ECC). RSA key length increase may not help, because the process of encryption and decryption will also be slowed down. The ECC 256-bit equivalent to the RSA 3072-bit is considered. ECC is known to provide much more efficient implementation security than RSA, while ECC encryption ensures the same security as RSA. It is slow to encrypt larger messages than symmetrical key (e.g. AES), and thus rarely used. The ECC-based encryption of Elgamal can be described as the Elgamal cryptographic system and is applied over a finite field to elliptic curve [7].

The security of the ECC depends on a trapdoor function that is with a publicly known base point the discrete logarithm of an elliptical curve element is hard to find. It is known as an elliptical curve that is not used to solve the problem of discrete logarithms known as **E**lliptic **C**urve **D**iscrete **L**ogarithm **P**roblem (ECDLP). Moreover, the following sub sections provide more information.

It is known that the ElGamal cryptosystem works over **Zp**. But, the generalized ElGamal cryptosystem can be on any group **G**. Not only restrict our self on **Zp**. So it is the generalized version of ElGamal cryptosystem. For this, it needs a discussion about the Discrete Log Problem (DLP) in that group.

1. **Visualization of Elliptic curve:** For demonstrate basic mathematical functions in elliptic curve, the JavaPlot library and the python ECC packages are used. The library allows us

to visualize various mathematical functions by defining a format for the graph with basic charts. This feature is used to draw our ellipse with a modified substantial part of the source code in the library to meet the visualization needs. This tool enables to visualize the two point addition and to double a curve point. The curve taken is $y^2 = x^3 + x + 6$ and set for the display.

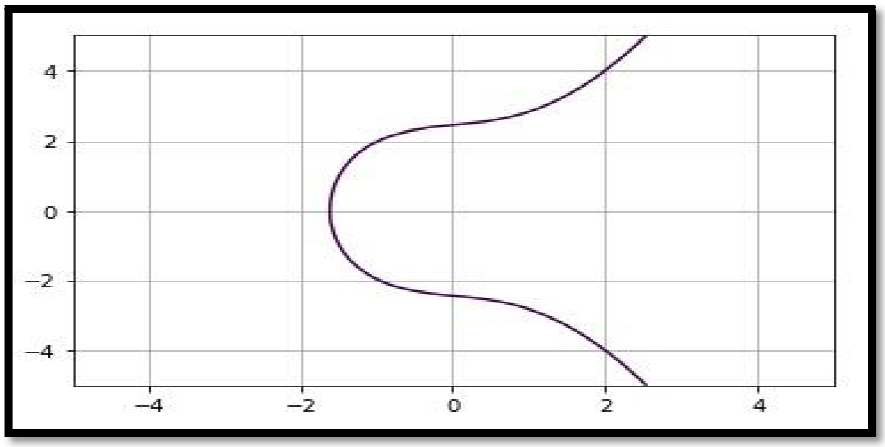

Figure 6: Elliptic Curve Visualization of the form $y^2 = x^3 + x + 6$

2. **Discrete Log Problem (DLP):** Let group, (G, *), be an abelian group. Now, in this group on we define the discrete log issue (G,*). Given (G, α, β), where G is a finite group, α belongs to G, β belongs to H where H is a sub group generated by α.
   $H = \{ \alpha^i \mid i >= 0\}$ i.e, $\alpha^i = \alpha$ x α x α …….. x α. H is acyclic group.
   Choose a, such that β = αa and it is hard to find 'a'. That is, if G is group, α is a generator. Choose 'a', $0 <= i < |G| - 2$, $\beta = \alpha^a = \alpha$ x α x α …… x α. The Discrete Log Problem is, given (G, α, β), finding 'a' is hard. Then it is a discrete log problem. It is the generalized discrete log problem.

3. **Generalized ElGamal cryptosystem based on G:** If an M2M node wishes to transmit a message to another node, the relevant public and private key pairs has to be created. If the group G exists and it is hard to find 'a' in the discrete log problem with a generalized ElGamal cryptosystem. The proposed scheme is shown in figure 7.

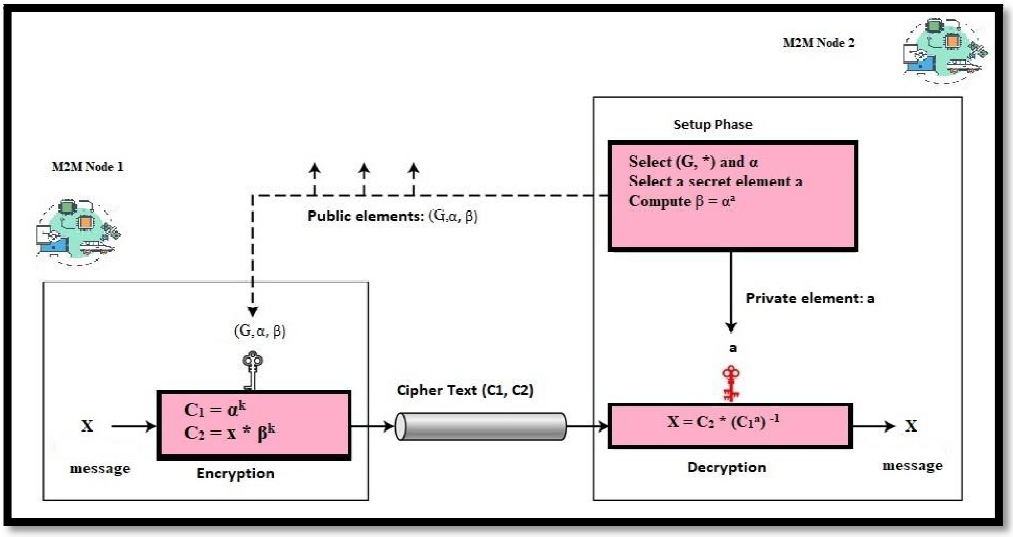

Figure 7: Architecture of Generalized ElGamal Cryptosystem over Elliptic Curves

The setup phase of an M2M node follows the steps below to compute the corresponding keys.

- The node choose a group G over the operation *, (G, *).
- The node also selects, α, called primitive element or generator.
- The node chooses a secret element 'a'.
- Now, the node computes $\beta = \alpha^a$, where $\alpha^a = \alpha \times \alpha \times \alpha \ldots\ldots \times \alpha$. ('a' times α)
- The node makes β is public.

The plain text space here is P = G and the cipher test space is C = G * G and the key space is k = { (G, α, a, β) | $\beta = \alpha^a$ }, where 'a' is kept private element and (G, α, β) are public elements of an M2M node. Now, the M2M node sends the public elements to the other end, say the receiver. The encryption elements send to the node are e2 = (G, α, β). If the node sends an initiating node message, then it has to select a random integer 'k'. And then encrypts the selected message, say 'x' with the elements (α, β) , that is, $E_{e2}(x, k)$ = (C1, C2).

Where $C1 = \alpha^k$; $C_2 = x * \beta^k$; $0 <= k <= |G| - 2$

Then, node sends the cipher text blocks C = (C1, C2) to the receiver. Now the receiver performs the decryption process as described below:

$$X = C_2 * (C_1^a)^{-1}$$

Like this, all the cipher text blocks are decrypted by the receiving node. This is possible with a group, (G, *).

4. **Generalized elgamal cryptosystem over Elliptic Curves (GE3C2):** Suppose the elliptic curve over Zp, E = {(x, y)} where x, y ∈ Zp * Zp and $Y^2 = x^3 + ax + b \bmod p$ and a, b ∈ Zp and $4a^3 + 27b^2 \neq 0 \pmod p$. This will form an abelian group, (E, '+'). The addition is defined as follows. Take any two points (P, Q) where P = (x1, y1) and Q = (x2, y2) then P + Q = R and P + (- P) = *O* (point at infinity). The formulas for addition of two points are same as in the case of elliptic curve arithmetic. Under this addition a group is formed and uses this group for ElGamal cryptosystem [8]. The generalized ElGamal cryptosystem over elliptic curves is illustrated below.

If a node desires to transfer a message to other node, then it has to generate the corresponding public and private key pairs. The process of generating public and private elements is shown in the figure 8. The setup phase of an M2M node follows the steps below to compute the corresponding keys.

- The node choose E
- The node also selects, α, called primitive element or generator.
- The node chooses a secret element 'a'.
- Now, the node computes $\beta = a^{\alpha}$, where $\alpha^{a} = \alpha \times \alpha \times \alpha \ldots\ldots \times \alpha$. ('a' times α)
- The node makes the public elements (E, α, β) and 'a' is private.

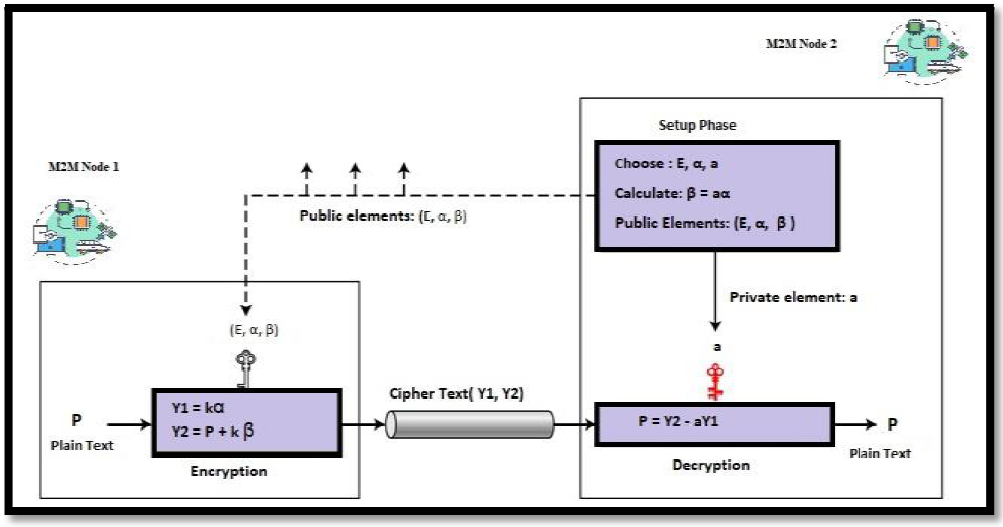

Figure 8: Architecture of Generalized ElGamal Cryptosystem over Elliptic Curves

Now, suppose a node wants to send the message (x) coming from the elliptic curve points x ∈ E. Table 3.1 Various GECC parameters used by our cryptosystem

Parameter	**Value**
Prime p	0xfffffffffffffffffffffffffffffffffffffffffffffffffffffffefffffc2f
a	0x0000000000000000000000000000000000000000000000000000000000000000
b	0x0000000000000000000000000000000000000000000000000000000000000111
α	(0x79be667ef9dcbbac55a06295ce870b07029bfcdb2dce28d959f2815b16f81798,

5. **Implementation of generalized elgamal ECC cryptosystem (GE^3C^2):** The proposed method used both JECC [1] library and python ecc-pycrypto package to apply the Generalized Elgamal encryption using Elliptic curves. The ecc-pycrypto is a package that offers APIs for producing keys by means of presented ECC curves, developing new ECC curves, performing ECC arithmetic (point addition and multiplication) and ciphering and deciphering integers using the public/private keys [9]. The variety of curve parameters

and P(x,y) selected are illustrated in table 1. Encryption: An input text P of variable length is crack into equal length blocks (p_0, p_1….. p_n). If the input message is ASCII characters and every one letter of significance can be encoded with a 8-bit integer (0,255) at the sender and likewise decoded at the receiver. The encoded message is then entrenched into the x,y - coordinates such that (x,y) lies on the agreed elliptic curve. The sending node chooses a random number k and then uses the receiver‘s public elements (E, α, β). The cipher text is a combination of two elements namely, Y1, Y2 where Y1 = k α and Y2 = x + k β.

Decryption: The node recovers the message by x = Y2 – aY1 where Y1 and Y2 are the received cipher text blocks.

IV. RESULTS

The proposed scheme is tested with ASCII input file nature of diverse sizes and also deliberate the time essential for encryption and decryption. The decryption procedure takes fewer time when compared to encryption [10].

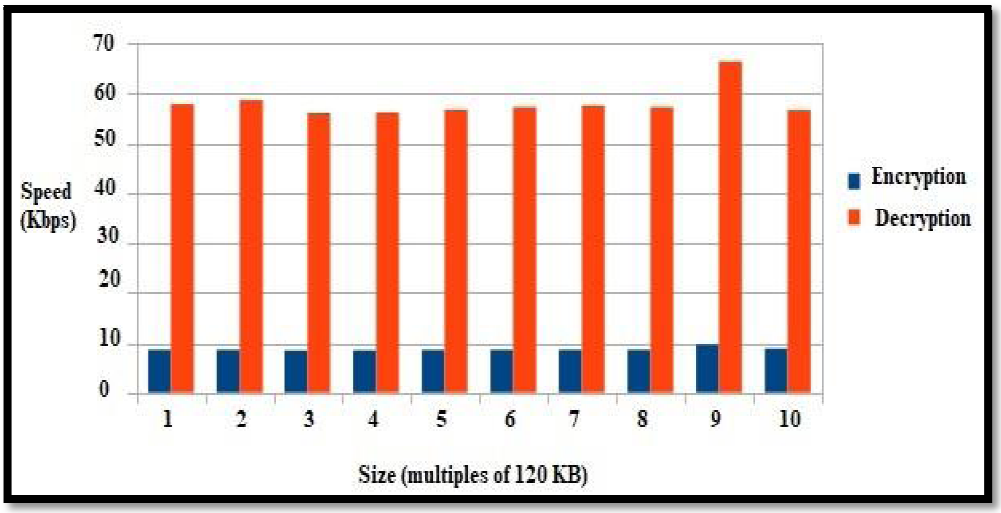

Figure 9: Encryption Vs Decryption on various input sizes using GE^3C^2.

It is observed that decryption is eight times faster than encryption because it involves less number of multiplications than encryption. Figure 9 shows the strength of the algorithm comparison between encryption and decryption times for a block size of120KB. The proposed schemed GE^3C^2 realization is analyzed through further algorithms (AES, DES, RSA and E^3C^2) by the different measures as described in table 2.

Algorithm	Properties	Key Size
AES	ECB/PKCS5 Padding	128 bit
DES	ECB/PKCS5 Padding	56 bit
RSA	ECB/PKCS1 Padding	1024 bit
E^3C^2	ECC_SHA1	256 bit
GE^3C^2	GECC	256 bit

Table 2: Comparison of various Protocols

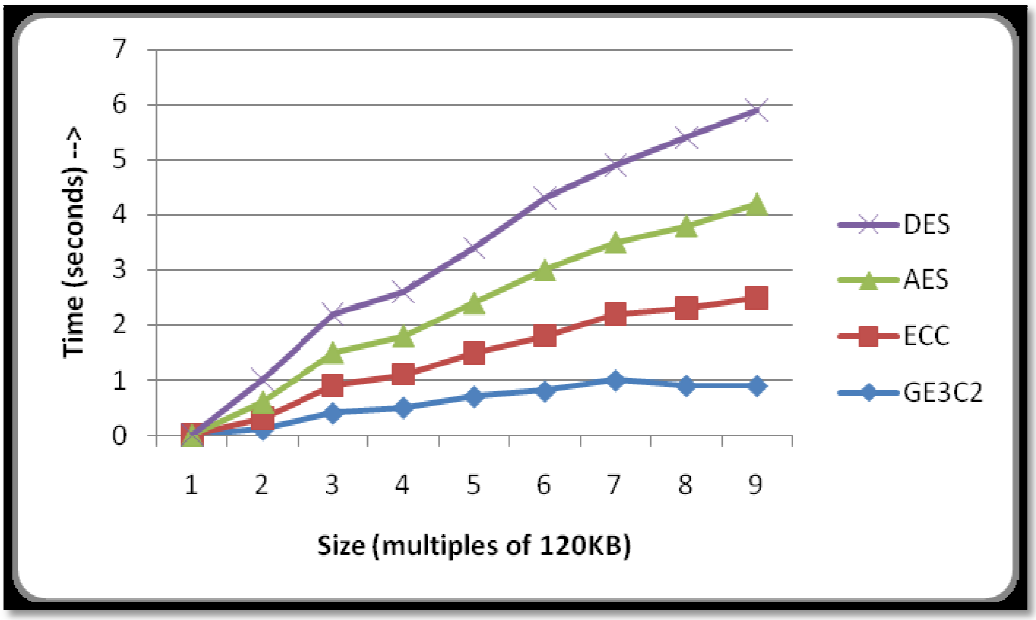

Figure 10: Performance of Encryption times of various algorithms on variable input sizes

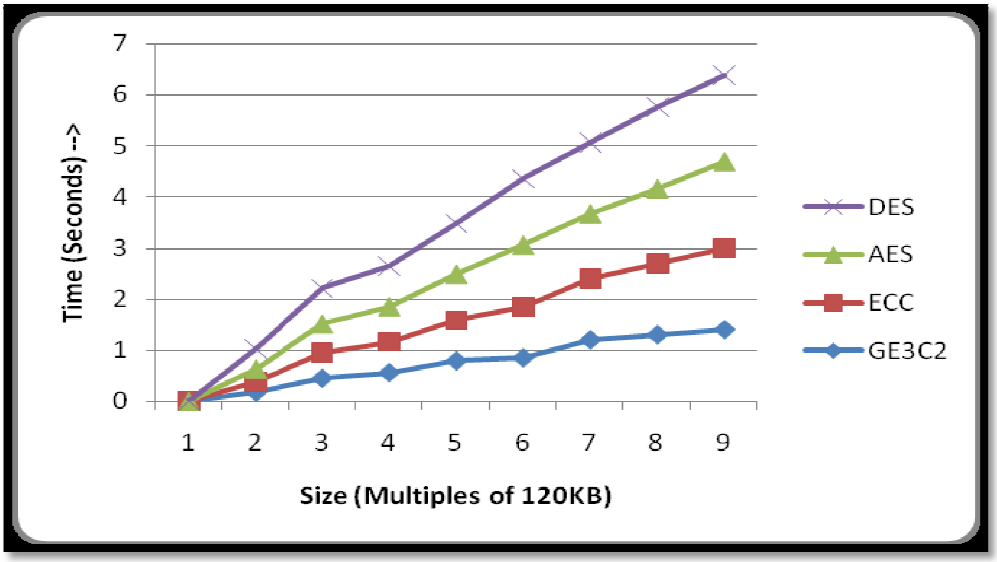

Figure 11: Evaluation of Decryption times of various algorithms on variable input sizes

Encryption Performance: Encryption speed is a primary deciding factor of selecting an algorithm for securing data. The proposed scheme performs better for various sizes of inputs. figure 10 illustrates the performance of various approaches. The proposed GE^3C^2 implementation performs better than AES, DES and the scheme E^3C^2

Decryption Speed: The decryption process takes some more time than that of encryption process, but still it performs better than the other approaches.

V. CONCLUSION

The proposed scheme also performs better for low cost devices such as nodes in M2M communication by means of computation complexity and storage space. In future the scheme can be extend for M2M communication in Ultra dense networks where heterogeneous nodes may authenticate and perform secure communication.

The proposed scheme also verified using the Pro Verifier tool to analyze the security capability of the algorithm. The scheme also analyzed informally under a given adversarial model to prove the robustness of the algorithm against known security attacks. Performance study of the proposed method in comparison with modern related authentication methods shows that the projected approach gives less computational and communication overhead.

REFERENCES

[1] Dipanwita Sadhukhan, Sangram Ray, “Cryptanalysis of an Elliptic Curve Cryptography Based Lightweight Authentication Scheme for Smart Grid Communication”, 4th Int‘l Conf. on Recent Advances in Information Technology | RAIT-2018.

[2] Sarwono Sutikno, Andy Surya, Ronny Effendi, “An Implementation of ElGamal Elliptic Curves Cryptosystems”, Asia-Pacific Conference on Circuits and Systems. Microelectronics and Integrating Systems. Proceedings, November 1998, pp 483-486.

[3] FuMinfeng, Chen Wei, “Elliptic Curve Cryptosystem EIGamal Encryption and Transmission Scheme”, International Conference on Computer Application and System Modeling (ICCASM 2010), October 2010, pp 51-53.
[4] Yuling Luo, Xue Ouyang, Junxiu Liu, And Lvchen Cao, “An image encryption method based on elliptic curve ElGamal encryption and chaotic systems”, IEEE Access, March 2019, pp 38507-38522.
[5] Meena Singh, Rajan MA, Shivraj VL, and Balamuralidhar P, “Secure MQTT for Internet of Things (IoT)”, Fifth International Conference on Communication Systems and Network Technologies, April 2015, pp 746-751.
[6] Khalid Mahmood, Shehzad Ashraf Chaudhry, “An Elliptic Curve Cryptography based Lightweight Authentication Scheme for Smart Grid Communication”, Future Generation Computer Systems, Volume 81, April 2018, pp 557-565.
[7] Koblitz, Neal. “Elliptic curve cryptosystems." Mathematics of computation 48.177 (1987): 203-209. Bos, Joppe W., et al. “Elliptic curve cryptography in practice.” Financial Cryptography and Data Security. Springer Berlin Heidelberg, 2014, pp 157-175.
[8] Java Elliptic Curve Cryptography(JECC) software : http://jecc.sourceforge.net/JavaPlot Library:-http://vase.essex.ac.uk/soft ware/ JavaPlot/
[9] V. Gupta, S. Gupta, and S. Chang, “Performance Analysis of Elliptic Curve Cryptography for SSL”, CM Wksp. Wireless Security, Mobicom 2002, Atlanta, GA, Sept. 2002,
[10] J. Guajardo and C. Paar, “Efficient Algorithms for Elliptic Curve Cryptosystems,” B. S. Kaliski Jr., Ed., Advances in Cryptology — Crypto, 1997, LNCS, vol. 1294, Springer-Verlag, 1997, pp. 342–56.
[11] M. Ciet et al., “Trading Inversions for Multiplications in Elliptic Curve Cryptography,” preprint, 2003, http://eprint. iacr.org/
[12] K. Eisentraeger et al., “Fast Elliptic Curve Arithmetic and Improved Weil Pairing Evaluation,” M. Joye, Ed., Topics in Cryptology — CT- RSA 2003, LNCS, vol. 2612, Springer-Verlag, 2003, pp. 343–54.
[13] Rongxing Lu, Xu Li, Xiaohui Liang, Xuemin Shen, and Xiaodong Lin “GRS: the green, reliability, and security of emerging machine to machine communications,” IEEE Communications Magazine, Vol. 49, No. 4, April 2011, pp. 28-35.
[14] Sachin Agarwal, Christoph Peylo, Ravishankar Borgaonkar, and Jean- Pierre Seifert, “Operator-based over-the-air M2M wireless sensor network security,” Proceedings of the 14th International Conference on Intelligence in Next Generation Networks (ICIN), October 2010, pp. 1-5.
[15] Tien-Dung Nguyen, Aymen Al-Saffar, and Eui-Nam Huh, “A dynamic ID-based authentication scheme,” Proceedings of the Sixth International Conference on Networked Computing and Advanced Information Management (NCM), August 2010, pp. 248-253.
[16] L. Sha, S. Gopalakrishnan, X. Liu, and Q. Wang, “Cyber-physical systems: a new frontier,” Proceedings of IEEE International Conference on Sensor Networks, Ubiquitous and Trustworthy Computing (SUTU), June 2008, pp. 1-9.
[17] A. A. Cardenas, S. Amin, and S. Sastry, “Secure control: towards survivable cyber-physical systems,” Proceedings of IEEE 28th International Conference on Distributed Computing Systems, June 2008 pp. 495–500.
[18] Min Chen, Jiafu Wan, and Fang Li, “Machine-to-machine communications: architectures, standards, and applications,” KSII Transactions on Internet and Information Systems, Vol. 6, No. 2, February 2012, pp. 480-497.
[19] Chen Hongsong, Fu Zhongchuan, and Zhang Dongyan, “Security and trust research in M2M system,” Proceedings of IEEE International Conference on Vehicular Electronics and Safety (ICVES), July 2011, pp. 286-290.
[20] Shuo Chen and Maode Ma, “A Dynamic-Encryption Authentication Scheme for M2M Security in Cyber-Physical Systems”, Globecom 2013, pp 2897-2901.
[21] Amira Barki, Abdelmadjid Bouabdallah, Said Gharout, “M2M Security: challenges and Solutions‖ in IEEE Communications Surveys & Tutorials, Volume 18, Issue 2, 2016, pp 1241-1254.

[22] Gary Davis, “2020: Life with 50 billion connected devices”, IEEE International Conference on Consumer Electronics (ICCE), January 2018.
[23] D. Evans, “The internet of things, how the next evolution of the internet is changing everything”, in Cisco Internet Business Solutions Group (IBSG) white paper, April 2011.
[24] Y. Zhang, R. Yu, S. Xie, W. Yao, Y. Xiao, and M. Guizani, “Home M2M networks: Architectures, standards, and QoS improvement,” Communications Magazine, IEEE, vol. 49, pp. 44–52, April 2011.
[25] R. Ma, H.-H. Chen, Y.-R. Huang, and W. Meng, “Smart grid communication: Its challenges and opportunities,” Smart Grid, IEEE Transactions on, vol. 4, pp. 36–46, March 2013.
[26] P. McDaniel and S. McLaughlin, “Security and privacy challenges in the smart grid,” Security Privacy, IEEE, vol. 7, pp. 75–77, May 2009.
[27] P. Jokar, N. Arianpoo, and V. C. M. Leung, “A survey on security issues in smart grids,” Security Communication Networks, June 2012.
[28] Shuyi Chen, Ruofei Ma, Hsiao-Hwa Chen, Hong Zhang, Weixia Meng, Jiamin Liu, “Machine-to-Machine Communications in Ultra-Dense Networks – A Survey”, in IEEE Communications Surveys & Tutorials, Volume 19, Issues 3, 2017, pp 1478 – 1503.

