Abstract

Objectives: Network security has always been an emerging study in communication technology as the hackers are emergent in developing newer mechanisms to crack the encrypted information of almost all existing cryptographic algorithms.

Method: Elliptic Curve Cryptography (ECC) is a novel cryptographic scheme which is proven to be feasible for public key cryptosystems. The advantages of ECC assure secure transmission of data over an insecure medium. The notion of key agreement protocol of standard ECC follows ELGAMAL encryption. In this paper we have proposed an improved ELGAMAL encryption system which adds an increased step of protection with ECC cryptography.

Findings: The proposed algorithm is developed as software tool to evaluate the novelty and the usefulness of the proposed algorithm in networked environment. Moreover, the algorithm has also well compared with the existing encryption methods over time complexity and security. The results show that the proposed scheme outperforms the standard ELGAMAL encryption technique with enhanced security and elapsed time in both encryption and decryption tasks. The results have also elucidated that the size of cipher text size is also extended than the standard ELGAMAL.
Application: This work is suggested to the field of network security for enforcing a secure communications between the nodes.

Key Words and Phrases: ECC, ELGAMAL, Key Agreement, Session Key

1 Introduction

Over the past few decades, the secure communication of computer networks is established through Elliptic Curve Cryptography (ECC), a novel cryptosystem, which entails a hard mathematical formulation for transforming user data into ciphers of unpredictable text and vice versa. The principal attraction of ECC lies on its wider acceptance as an alternate to traditional Public Key Infrastructure (PKI) system as it is a high speed and low power consuming algorithm. ECC offers potential security with smaller and faster keys. The basic operations of ECC in PKI include key and signature generations, signing and verifications.

The base of ECC is said to be the equation of elliptic curve $E: y^2 = x^3 + ax + b$ and are defined over a finite field $\mathbb{F}_p$ where $p$ is a prime and $a, b \in \mathbb{F}_p$. $E \mathbb{P}(a,b)$ are set of integers $x$ and $y$ that satisfies the equation. These are often called as prime curve that reduces modulo $p$ at each stage. To find all possible values of $y^2 = x^3 + ax + b \mod p$, substitute $x$ with all possible values of mod $p$ and extract the values those are matched with $y^2$. Replacing the values into the cubic and reducing $p$ yields the possible values of $y^2$. The possible values of $x$ that could be paired with $y$ are identified to a given point on curve. Given the curve $E$ the cryptographic groups are then employed in large on all points on $E\mathbb{P}$ including with a point at infinity, the neutral element.
1.1 ECC Building Blocks

**Point:** A Point is the x, y co-ordinate on elliptic curve that lies on \( y^2 = x^3 + ax + b \mod p \). For example a point P1 can be denoted as \( P1=(x, y) \).

**Point Addition:** A point P1, or two points P1 and P2 produces another point P3 using point addition which can be denoted as \( P1+P2=P3 \).

**Rule of thumb for point addition**

1. If \( P1 \neq P2 \) and neither \( \infty \), the slope of line for point addition is computed using the formula
   
   \[ \text{a. If}(x_1 \neq x_2) \text{ then } x_3 = m^2 - x_1 - x_2 \text{ and } y_3 = m(x_1 - x_3) - y_1 \]
   
   Where \( m = \frac{y_2 - y_1}{x_2 - x_1} \) . If \( x_1 = x_2 \) but \( y_1 \neq y_2 \) then \( P1+P2 = \infty \)

2. If \( P1=P2 \) and \( y_1 \neq 0 \) then \( x_3 = m^2 - 2x_1 \) and \( y_3 = m(x_1 - x_3) - y_1 \)
   
   Where \( m = \frac{3x_1^2 - a}{2y_1} \)

Fig. 1. Elliptic Curve \( y^2 = x^3 + ax + b \)
3. if $P_1 = P_2$ and $y_1 = 0$, then $P_1 + P_2 = \infty$

4. $P_1 + \infty = P_1$ for all points in $P$ on $E$.

**Point Doubling:** Point doubling is the addition of a point $P_1$ to itself to obtain another point $P_2$, on the elliptic curve. Supposing if $P_1$ and $k$ is a positive integer $kP_1$ denotes the successive doubling of $P_1+P_1+\ldots+P_1$ (with $k$ summands). For example the computation of $18P_1$ can be simplified as follows:

$$
P_1+P_1=2P_1;2P_1+2P_1=4P_1;4P_1+4P_1=8P_1;8P_1+8P_1=16P_1;16P_1+2P_1=18P_1$$

Rule of thumb for point doubling:

1. if $y_1 \neq 0$ then $x_3 = m^2 - 2x_1 y_3 = m(x_1 - x_3) - y_1$
where $m = \frac{3x_1^2 - A}{2y_1}$

2. if $y_1 = 0$, then $P_1 + P_1 \neq \infty$

**Point Multiplication:** Performing repetitive addition is one way of computing point multiplication and simpler method is the double and add method. $k \times P_1$ denotes the point multiplication of an elliptic curve point $P_1$ to itself by $k$ times which results another point on the curve[1].

## 2 Review of Literature

The existing literature on ECC consists of an authentication protocol for the establishment of secure communication between base station and nodes in mobile networks[2]. The protocol is employed using Elliptic Curve Cryptography, a public key cryptographic scheme for authenticating the base stations of mobile networks. The process of authentication is determined in three steps of message transformation such as the request from the base station, the nodes response and verification of base station. The core process of the protocol is to generate public and private keys for secure transmission of data throughout the authentication process of base station. The author has claimed that their work has achieved a secure communication in mobile networks with simple steps with reduced complexity.
The comparison of different methods of point multiplication and established that the inclusion of NAF (Non-adjacent Form) in addition and subtraction method decreases the number of point additions and speeds up the computation of scalar multiplication. The authors have also suggested that the implementation of projective coordinate than affine coordinate system improves the efficiency of the system.

A security system for mobile network for validating the mobile user is presented. The author has divided the system into two parts, where the first part consists of an API for ECC for generating shared secret key for secure communication and created a web service for distributing the keys to validate the mobile user.

Another proposal is a modified ECC that transforms a message into an affine point on the EC, and applied the knapsack algorithm for encrypting the message over a finite field. Every single digit ASCII integer of the character is converted to a set of coordinates that fit the EC and the transformation introduces a non-linearity in the character to hide its identity completely. The transformed character is then encrypted with ECC. The application of knapsack algorithm turns the message into confusing integers that does not allow the cryptanalyst attempting brute force attack. The authors have claimed that the application of knapsack algorithm is a new contribution that increases the security of user message.

A modified cryptosystem that divides the message into blocks of single characters to be converted to an hexadecimal values is proposed. As the hexadecimal value of single character has two digits it could be expressed as a point on the curve. The encryption and decryption of the proposed algorithm uses the point as hexadecimal values of each character on elliptic curve. The author has claimed that the hardware implementation of their algorithm requires less storage, memory, and bandwidth.

3 Methodology

ECC is not a standalone algorithm which solely takes care of the end-end protection of user data; rather it is a part of PKI for generating keys. In Asymmetric PKI cryptography, a user or system participates in the communication have a pair of keys called pub-


lic and private keys. Private Key is a secret key known only to the particular user, and public key is distributed to one or more users who are agreed upon a communication. Those keys are further manipulated with key agreement protocols for encryption and decryption of data. The proposed work concerns with the way the data are encrypted. Domain Parameters of the proposed algorithm

- the prime $p$ to define the size of the field
- the values $a$ and $b$ to define the coefficients of elliptic curve equation
- the base point $G$ for generating sub groups
- private key $k$

Proposed Algorithm:

1. determine prime $p$ and coefficients $a$ and $b$ of elliptic curve equation $y^2 = x^3 + ax + b$
2. extract all possible $x$ and $y$ coordinates that originates the curve
3. choose a random base point $G$ for generating subgroups
4. Let $k_i$ be a random private key (SPRK) of the sender with $1 < k_i < p - 1$
   i. Compute senders public key $SPUK = k_i \ast G$
   ii. Forward SPUK to the receiver
5. Let $k_j$ be the random private key (UPRK) of the receiver with $1 < k_j < p - 1$
   i. Compute receivers public key $RPUK = k_j \ast G$
   ii. Forward RPUK to the sender
6. Determine the session key $K$
   i. Sender: $k_i \ast (k_j \ast G)$
   ii. Receiver: $k_j \ast (k_i \ast G)$
7. Let sender encrypt message $M$ with the modulo inverse of $M$ with key $K$
8. Let receiver decrypt message $M$ with modulo inverse of $K$
4 Illustration

In this paper, a modified ELGAMAL key agreement protocol is proposed to enhance the security of the data by adding an increased step of mod inverse of message M with key K. For the illustration let us consider an Elliptic curve E11(1,6) that originates the equation (1)

\[(1)\]

To find the x and y coordinates of elliptic curve, replace the values of x from 0 to 10 on to equation (1) to extract \(y^2\) and identify the possible values of y that could be paired with x. Table 1 denotes the generation of curve points.

<table>
<thead>
<tr>
<th>y</th>
<th>(y^2)</th>
<th>((y^2)\mod 11)</th>
<th>x (x^3+x+6)</th>
<th>((x^3+x+6)\mod 11)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>6</td>
<td>6</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>8</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>4</td>
<td>16</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>9</td>
<td>9</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>16</td>
<td>5</td>
<td>74</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>25</td>
<td>3</td>
<td>136</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>36</td>
<td>3</td>
<td>228</td>
<td>8</td>
</tr>
<tr>
<td>7</td>
<td>49</td>
<td>5</td>
<td>356</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>64</td>
<td>9</td>
<td>526</td>
<td>9</td>
</tr>
<tr>
<td>9</td>
<td>81</td>
<td>4</td>
<td>744</td>
<td>7</td>
</tr>
<tr>
<td>10</td>
<td>100</td>
<td>1</td>
<td>1016</td>
<td>4</td>
</tr>
</tbody>
</table>

From table 1, the values of the attribute \((y^2)\mod 11\) is matched with the attribute of \((x^3+x+6)\mod 11\) for the generating the curve points. For the value y=0, \((y^2)\mod 11\) is 0 which doesn't have any match with \((x^3+x+6)\mod 11\). When y=2, \((y^2)\mod 11\) is 4, which has four matches on \((x^3+x+6)\mod 11\) for the corresponding x values 5, 7, and 10. The curve points for the equation E11(1,6) = (2, 4), (2, 7), (3, 5), (3, 6), (5, 2), (5, 9), (7, 2), (7, 9), (8, 3), (8, 8), (10, 2), (10, 9), and \(\infty\). Fig.2 denotes the point generation of the curve E11(1,6).

An addition table is constructed using the curve points in the next step, for the purpose of simplifying the task of point multipli-
cation. As it is discussed, point multiplication is achieved through repetitive addition of points.

Point Addition

$P_1= (2, 4)$ and $P_2= (3, 5)$. The addition of $P_1 + P_2= Q$ is derived as follows:

$x_1=2, y_1=4$ and $x_2=3, y_2=5$

$m = \frac{y_2-y_1}{x_2-x_1} = m = \frac{5-4}{3-2} = 1$

$x_3 = m^2 - x_1 - x_2 = 12 - 2 - 3 = 7 - 5 = -4 \% 11 = 7$

$y_3 = m(x_1 - x_3) - y_1 = 1(2 - 7) - 4 = -5 - 4 = -9 \% 11 = 2$

Hence, in $E_{11}(1,6)$ the addition of points $P_1 + P_2$, $(2,4) + (3,5) = (7,2)$. The addition table is derived by adding all points with all other points on the curve, including the point at infinity. The table of addition is depicted in Fig.3.

![Fig. 2. Point Generation of the Curve E11(1,6)](image-url)
Enhanced ELGAMAL Encryption:

Let \( p = 11 \), base point \( G = (3, 5) \) on curve \( y^2 = x^3 + x + 6 \), \( A \) chooses \( k_1 = 10 \) and publishes \( 10G = (8, 8) \)

Encryption:

B wants to send a message \( M = 6 \) to \( A \):
1. B chooses a random \( k_j = 5 \) and calculates \( 5 \times 10G = (5, 9) \), where he takes the key \( K = 5 \)
2. B sends \( A \) the pair \((C_1, C_2)\), where
   a. \( C_1 = 5G = (2, 7) \)
   b. \( C_2 = K \times M^{-1} \mod p = 5 \times 2 \mod q = 10 \mod 11 = 10 \)

Decryption:

\( A \) calculates \( 10 \times (2, 7) = (5, 9) \) getting \( K = 5 \)
\( A \) calculates the inverse of \( K^{-1} \mod 11 = 9 \)

\( A \) decrypts by inverting \( C_2 \times K^{-1} \mod 11 = 10 \times 9 = 90 \mod 11 = 2^{-1} = 6 \)
5 Experimental study

The enhanced ELGAMAL encryption technique is implemented using java program for investigating its performance with the existing key agreement protocols. The time taken for key generation, encryption and decryption are analyzed to measure the level of protection that is ensured by the proposed system. The more the time takes for encryption and decryption is more the security of the system. As the proposed method implements modulo inverse operation on both stages of encryption and decryption, it offers additional security on user data and protects users from different types of network threats such as man-in-the-middle, eavesdropping and so on. Fig.4. depicts the encryption and decryption of Improved ELGAMAL Scheme.

Fig.4. Improved Elgamal Encryption and Decryption

The cipher text of the improved ELGAMAL encryption doubles the size of the plain text over encryption which initiates a positive sign on the proposed work. In addition, the key generation time of both the improved ELGAMAL and standard ELGAMAL means the same as it is generated from the notion of ECC. But, there is a variation between improved ELGAMAL with standard ELGAMAL over encryption and decryption time because the underlying the mechanism of proposed work adds an addition of step inversing the message takes additional time for both encryption which is shown in Fig. 5.
The variation of improved ELGAMAL with standard ELGAMAL with respect to plain text cipher text, key generation time, encryption and decryption time comparisons are shown in Table. 2

Table 2: Performance analysis of improved ELGAMAL encryption scheme.

<table>
<thead>
<tr>
<th>Measure</th>
<th>Improved ELGAMAL</th>
<th>Standard ELGAMAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plain Text Size</td>
<td>765</td>
<td>765</td>
</tr>
<tr>
<td>Cipher Text Size</td>
<td>1530</td>
<td>1443</td>
</tr>
<tr>
<td>Key Generation Time</td>
<td>484</td>
<td>484</td>
</tr>
<tr>
<td>Encryption Time</td>
<td>79</td>
<td>58</td>
</tr>
<tr>
<td>Decryption Time</td>
<td>47</td>
<td>35</td>
</tr>
</tbody>
</table>

6 Conclusion

ECC plays a vital role in cryptography as it is highly challengeable for the cryptanalyst to crack information. Though, the hard mathematical formulation of ECC leads the users to build highly complicated systems, the modern hacking tools are vibrant enough to break the system, hence, there is always been a need for protection of systems with newer ideas and concepts. In this paper, a
novel concept of improved ELGAMAL encryption scheme is introduced as additional step of increased security using mod inversion on both sides of sender and receiver. The proposed method offers a promising security by inverting the characters of message which would turn the message into even more confusable than the existing ELGAMAL scheme.

References


