Comprehensive Security System for Mobile Network Using Elliptic Curve Cryptography over GF (p)

Lokesh Giripunje  
Student (ME), Department of Computer Science and Engineering  
G.H.Raisoni College of Engineering  
Nagpur, India

Sonali Nimbhorkar  
Assistant Professor, Department of Computer Science and Engineering  
G.H.Raisoni College of Engineering  
Nagpur, India

Abstract—Mobile devices have many differences in their capabilities, computational powers and security requirements. Mobile devices can be used as the enabling technology for accessing Internet based services, as well as for personal communication needs in networking environments. Mobile services are spread throughout the wireless network and are one of the crucial components needed for various applications and services. However, the security of mobile communication has topped the list of concerns for mobile phone users. Confidentiality, Authentication, Integrity and Non-repudiation are required security services for mobile communication. Currently available network security mechanisms are inadequate, hence there is a greater demand to provide a more flexible, reconfigurable, and scalable security mechanism. This project provides effective security solution using Public key cryptography. The implementation of this project is divided into two parts first, design of API for ECC (Elliptic Curve Cryptography) which generates shared secret key required for secure communication and secondly, a web service is created which distributes this key to validate mobile user.

Keywords—Comprehensive Security, Mobile communication security, Public key cryptography, ECC (Elliptic Curve Cryptography), confidentiality, authentication, integrity and non-repudiation.

I. INTRODUCTION

Mobile phones are most common way of communication and accessing Internet based services. Currently, mobile phones are not only used for formal communication but also, sending and receiving sensitive data. However, the security of mobile communication has topped the list of concerns for mobile phone users. Secure communication is in demand for any kind of communication network. Public key cryptography is effective security solution to provide secure mobile communications [1][2][3][4][5]. To provide end to end security we are using Public key cryptography i.e., Elliptic curve cryptography. Intractable problems are the center of public key cryptography and bring computationally demanding operations into a cryptosystem. Elliptic curve cryptography (ECC) is based upon the algebraic structure of elliptic curves over finite field. Elliptic curve cryptography (ECC) is the most efficient public key encryption scheme based on elliptic curve concepts that can be used to create faster, smaller, and efficient cryptographic keys. ECC generates keys through the properties of the elliptic curve equation instead of the conventional method of key generation. This scheme can be used with public key encryption methods, such as RSA, and Diffie-Hellman key exchange. Most attractive feature of ECC is its relatively short operand length compared to that of RSA and also it is based on discrete logarithm in finite fields. ECC can provide various security services in the form of key exchange, communication privacy through encryption, authentication of sender and digital signatures to ensure message integrity[1][2][3][4][5]. ECC helps to establish equivalent security with lower computing power and battery resource usage. The ECC covers all primitives of public key cryptography like digital signature, key exchange, key transport, key management. Presently ECC has been commercially adopted by many standardize organization such as NIST (national Institute of standard and technology), ISO (International Organization for Standardization), and ANSI (American national standard institute). ECC covers the discipline of mathematics and computer science and engineering. Due to extremely high demand of handheld devices (i.e., cellular phones and PDAs) among people, over the years there is gradual increment in demand of various supportive applications and security services. Cryptographic algorithms used by Mobile Subscribers to protect the privacy of their cellular voice and data communication. Public key cryptography algorithms provide the way to achieve security requirements viz; confidentiality and authentication [7][9].

A5/x are the encryption algorithms are used to ensure privacy of communication (voice and data) on mobile phones over radio channels. A5/3 encryption algorithm used for 3G and GEA3 encryption algorithm used for GPRS. F8 is confidentiality algorithms developed by 3GPP used in UMTS System. But this schemes does not provide complete security solution [7][9].
The rest of the paper is structured as follows. Section II gives detailed description of commonly employed security concepts and terminology. In Section III we present basic idea of public key Cryptosystems (PKC), followed by detailed description of Elliptic Curve Cryptography (ECC) in Section IV. Security analysis of ECC is given in Section V. Section VII concludes the paper.

II. CRYPTOGRAPHIC TERMINOLOGY

William Stallings [9] provide a detailed description of commonly employed security concepts and terminology. The concern for security in practice is addressed by choosing a security protocol, which achieves all the required security objectives. Security protocols realize the security objectives through the use of appropriate cryptographic algorithms.

Basic Security Terminologies used in cryptography are:

- A message present in a clear form, which can be understood by any casual observer, is known as the plaintext. The encryption process converts the plaintext to a form that hides the meaning of the message from everyone except the valid communicating parties, and the result is known as the cipher text. Decryption is the inverse of encryption. The processes of encryption and decryption are controlled on a quantity known as the key, which is ideally known only to the valid users. Strength of a security scheme depends on the secrecy of the keys used [9].

- A security protocol formally specifies a set of steps to be followed by communicating parties, so that the mutually desired security objectives are satisfied. The four main security objectives include:
  - **Confidentiality:** This means that the secrecy of the data being exchanged by the communicating parties is maintained, i.e., no one other than the legitimate parties should know the content of the data being exchanged.
  - **Authentication:** It should be possible for the receiver to ensure that the sender of the message is who he claims to be, and the message was sent by him.
  - **Integrity:** It provides a means for the receiver of a message to verify that the message was not altered in transit. It checks originality of message.
  - **Non-repudiation:** The sender of a message should not be able to falsely deny later that he sent the message, and this fact should be verifiable independently by an independent third-party without knowing too much about the content of the disputed message(s).

Security objectives thus provide trust on the Web. They are realized through the use of cryptographic algorithms which are divided into two categories depending on their characteristics.

These categories are:

- **A. Symmetric algorithms:** These algorithms use the same key for encryption and decryption. They rely on the concepts of "confusion and diffusion" to realize their cryptographic properties and are used mainly for confidentiality purposes. Also known as secret key cryptosystems.

- **B. Asymmetric algorithms:** These algorithms use different keys, known as the public key and the private key, for encryption and decryption, respectively. They are constructed from the mathematical abstractions which are based on computationally intractable number-theoretic problems like integer factorization, discrete logarithm, etc. They are primarily used for authentication and non-repudiation [9]. Also known as public key Cryptosystems (PKC).

III. PRINCIPLES OF PUBLIC KEY CRYPTOGEY SYSTEMS

Pair of keys for an encryption and decryption is the basic reason that differentiates public key Cryptosystems (PKC) from secret key cryptosystems. In Public key Cryptosystems (PKC), public key is known to all, however private key is kept confidential, and because it is impossible to calculate the private key even if public key is known but vice versa is possible. Public key algorithms use different keys for signing and decryption, and for encryption and signature verification. The private key may only be known to its owner and must be kept in secret. It may be used for generation of digital signatures or for decrypting private information encrypted with the public key. The public key may be used for verifying digital signatures or for encrypting information. Public key algorithms have a big advantage when used for ensuring privacy of communication. If Sender A want to send plaintext message m to Receiver B, he calculates cipher text c by performing encryption on message (m) to obtain cipher text (c), then transmits cipher text (c) to Receiver B. When Receiver B gets message (c), he calculates original plaintext message m by performing decryption on message (c). [3][9]

![Fig. 1. Encryption/Decryption with public key Cryptosystems (PKC)](image-url)
Some of public key algorithms are ECC, RSA, Diffie-Hellman key exchange and DSA. In the later section we mainly focus on Elliptic Curve Cryptography (ECC).

### Table 1. Applications of public key cryptosystem [9]

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Encryption/decryption</th>
<th>Digital Signature</th>
<th>Key exchange</th>
</tr>
</thead>
<tbody>
<tr>
<td>RSA</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Elliptic curve</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Diffie-Hellman</td>
<td>No</td>
<td>NO</td>
<td>Yes</td>
</tr>
<tr>
<td>DSS</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

### IV. ELLIPTIC CURVE CRYPTOGRAPHY

Elliptic Curve Cryptography (ECC) was first introduced by Victor Miller, who was then at IBM, and Neil Koblitz from the university of Washington independently in 1985. The principal attraction of ECC compared to RSA is that it offers equal security for a far smaller key size, thereby reducing processing overhead. The advantage of ECC over other public key cryptography techniques such as RSA is that the best known algorithm for solving ECDLP the underlying hard mathematical problem in ECC takes the fully exponential time and so far there is a lack of sub exponential attack on ECC. ECC is based on the Discrete Logarithmic problem over the points on an elliptic curve.[1][2][5][7][9][19]

An elliptic curve is the set of Weierstrass equations of the form

\[ y^2 = x^3 + ax + b \]  

or

\[ y^2 + xy = x^3 + ax^2 + b \]  

or

\[ y^2 + y = x^3 + ax + b \]

where \( x \) and \( y \) are variables, \( a \) and \( b \) are constants. For cryptography purposes we always use a finite field.

#### A. Elliptic curve over a Galois field

Using the real numbers for cryptography have a lot of problems as it is very difficult to store them precisely in computer memory and predict how much storage will be needed for them. Abstractly, a finite field consists of a finite set of objects called field elements together with the description of two operations addition and multiplication that can be performed on pairs of field elements. These operations must possess certain properties. It turns out that there is a finite field containing \( p \) field elements if and only if \( n \) is a power of a prime number, and furthermore that for each such there is precisely one finite field. The finite field containing elements is denoted by \( GF(p) \).

The difficulty can be solved by using Galois fields. In a Galois field, the number of elements is finite. Since the number of elements if finite, we can find a unique representation for each of them, which allows us to store and handle the elements in an efficient way. ECC deal with various properties of points on curve and functions. Galois showed that the number of elements in a Galois field is always a positive prime power, \( p^n \) and is denoted by \( GF(p^n) \).

Two special Galois fields are standard for use in Elliptic Curve cryptography. They are \( GF(p) \) when \( n=1 \) and \( GF(2^n) \) when \( p=2 \).[1][5][9]

Remaining paper deals with Elliptic Curve cryptography over \( GF(p) \). To illustrate the ECC, let us consider the following elliptic curve:

\[ y^2 = x^3 + 1x + 1 \mod 751 \]

The elliptic curve group generated by the above elliptic curve is \( E \mod 751(1,1) \). Let the generator point \( e1 = (5,120) \). Then multiples of \( dx \) of the generator point \( e1 \) are (for \( 1 \leq d \leq 751 \)).

---

**Galois fields GF \( (p^n) \)**

- (Prime curve) \( GF (p) \) when \( n=1 \)
- (Binary curve) \( GF (2^n) \) when \( p=2 \)

**Fig. 2. Classification of Galois fields**
**B. Elliptic curve over a GF (p)**

Elliptic curves over GF(p) are of the form $E_p(a,b): y^2 \equiv x^3 + ax + b \pmod{p}$ where $a, b \in F_p$ and $(4a^3 + 27b^2) \equiv -1 \pmod{p}$.

After generating Elliptic curve, any number is entered and checked for prime number. If it is not prime number then lower number which is prime is selected as prime number.

Here, the selected number 751 is prime number.

**C. Finding points on the curve**

The following algorithm gives the points on the curve $E_p(a,b)$ [1].

```plaintext
Algorithm elliptic_points (p, a, b)
{
    x=0
    while(x<p)
    {
        w=(x^3+ax+b) \pmod{p}
        if( w is a perfect square in Zp )
        output ((x,√w),(x,-√w))
        x=x+1
    }
}
```
The number of points on an elliptic curve over a finite field must satisfy Hasse’s theorem. Given a field GF(p), the order of the curve N will satisfy the following equation. [1][5][9]:

\[ p + 1 - 2\sqrt{p} \leq N \leq p + 1 + 2\sqrt{p} \]

Point additions (PA) and point doublings (PD) can be implemented using coordinate systems [18] like affine, coordinate system, Standard projective, Standard projective and affine, Jacobian projective, Jacobian projective and affine, Lopez–Dahab.

The most popular coordinate representation is affine representation which is based on two coordinates (x, y) and other representations such as projective, Jacobian, Lopez–Dahab uses three coordinates. Transforming affine coordinates into one of the other representations is almost simple but not vice versa, since transformation requires costlier field inversion [10][3].

Cryptography needs modular arithmetic for addition operation, algebraic structure like group and field. The group defines the set of the points on the elliptic curves and the addition operation on the points. The field defines the addition, subtraction, multiplication, and division that are required for determining the addition of the points in the group. [5]

Figure 5 shows the addition of two points on an elliptic curve. Elliptic curves have the interesting property that adding two points on the elliptic curve results a third point on the curve. Therefore, adding two points, P1 and P2, gets us to point P3, also on the curve. Small changes in P1 or P2 can cause a large change in the position of P3. Point addition is the addition of two points J and K on an elliptic curve to obtain another point L on the same elliptic curve as shown in figure 4 and point doubling. Point doubling is the addition of a point J on the elliptic curve to itself to obtain another point L as shown in figure 5.

![Fig.5. Group laws of Elliptic Curve (point addition)](image)

![Fig.6. Group laws of Elliptic Curve (point doubling)](image)

![Fig.7. Showing Elliptic Curve Points](image)
With 751 as prime number, total numbers of Curve point generated are 726. All these curve points satisfy the curve equation. The number of curve points depends on value of prime number, larger the prime number larger the number of curve points.

Then, the order of point is selected (lower than the number of curve points). Here 700 is selected as order of point. Finally, G(5,120) is selected as generating point and O(19,470) is taken as point of infinity.

**D. Generating Public and Private Keys using Elliptic Curve Diffie-Hellman Scheme (ECDH)**

The original Diffie-Hellman algorithm requires 1024 bits to achieve sufficient security but Diffie Hellman based on elliptic Curve can achieve the same security level with 160 bit [6].

1. User A chooses \( E(a,b) \) with an elliptic curve over GF(p).
2. User A chooses a generator point, \( e_1(x_1,y_1) \) on the curve.
3. User A chooses an integer \( K_a \).
4. User A calculates \( e_2a(x_2,y_2)=K_a*e_1(x_1,y_1) \). Multiplication here means multiple additions of points.
5. User A announces \( e_2a(x_2,y_2) \) as his public key; he keeps \( K_a \) as his private key.
6. Similar process is carried out for User B.
7. Finally, the session/secret key is generated with the help of Diffie-Hellman key exchange as \( R=K_a*K_b*e_1 \)

Where \( K_a=\)Private Key of User A  
\( K_b=\)Private Key of User B

To calculate public and shared secret keys involve most famous scalar multiplication operation which is nothing but point doubling and point addition of point P as shown in the figure 4 and figure 5.

The overall process of generating Public, Private Key and Shared Secret Key using Elliptic Curve Diffie-Hellman Scheme (ECDH) is shown in figure 6.

![Fig.8. Obtaining Generating Point and Point of Infinity](image1)

![Fig.9. Diffie-Hellman protocol based on ECC](image2)
In given example, Public and Private Keys are generated using Elliptic Curve Diffie-Hellman Scheme (ECDH). For 650 as private key of user A, (670,448) is generated public key of user A and with 600 as private key of user B, (302,293) is generated public key of user B.

![Figure 10: Generating Public and Private Keys using Elliptic Curve Diffie-Hellman Scheme (ECDH)](image)

Generated public key in Figure 11 is used for generation of shared secret key. For obtained public key the shared secret key generated at User A and User B is same (545,389).

![Figure 11: Generating Shared Secret Key using Elliptic Curve Diffie-Hellman Scheme (ECDH)](image)

The major cryptographic function in Elliptic Curve Cryptography is scalar point multiplication which computes \( Q = kP \), a point \( P \) is multiplied by an integer \( k \) resulting in another point \( Q \) on the curve. Scalar multiplication is performed through a combination of point additions and point doublings, e.g. \( 11P = 2(2(2P)) + P + P*3 \).

Discrete methods to represent scalars are as follows:
Single Scalar Multiplication: Let $E$ be an elliptic curve over a field $K$, $P$ a point in the group $E(K)$, a positive integer $k \in [1, n - 1]$, where $n$ is the order of $E(K)$. Then the computation of $[k]P$ is called single scalar multiplication.

Double Scalar Multiplication: Let $E$ be an elliptic curve over a field $K$, $P$ and $Q$ two distinct points in the group $E(K)$, $k, l$ two distinct positive integers in the interval $[1, n - 1]$ where $n$ is the group order of $E(K)$. Then the computation of $[k]P + [l]Q$ is called double scalar multiplication.

Scalar multiplication is the computationally heaviest operation in signature verification in elliptic curve based cryptosystem. The most important objective of scalar multiplication is to improve the speed of both types of scalar multiplication. In general, there are several approaches to accomplish the purpose selection is discussed [4][9] that focuses on:

- Proper usage of coordinate systems.
- Selecting arithmetic efficient curves.
- Combination of operation, sometimes point addition and point multiplication performed together to reduce the number of field operation.

For the implementation of scalar multiplication following forms are used such as Right-to-left binary method, Left-to-right binary method, Non Adjacent Form, Width -w Nonadjacent Form Joint Sparse Form, Double and add form, Addition chains, Fibonacci and add, Montgomery method.

Implementation of point multiplication can be separated into three distinct layers like Finite field arithmetic, Elliptic curve point addition and doubling. Point multiplication scheme makes secure against attacks, various methods have been suggested using special point representations for specifically chosen elliptic curves recommended by NIST. Here we use Double and Add form to implement scalar Multiplication [1][3][5][6].

After implementing ECC, a web service is created using existing code. Finally connectivity is established between mobile device and web server. It helps to carried out Secure Access Authentication to protect information of the subscribers and avoid fraud.

Fig.12. Showing Emulator with Connectivity to server applications available for download

Fig.13. Emulator showing no. of applications available for download if user is authenticated

V. SECURITY OF ELLIPTIC CURVE CRYPTOGRAPHY
As RSA depends on the difficulty of large-number factorization for its security, ECC depends on the difficulty of the large number discrete logarithm calculation. This is referred to as the Elliptic Curve Discrete Logarithm Problem (ECDLP). Elliptic curves for which the total number of points on the curve equals the number of essentials in the primary finite field are also considered cryptographically pathetic. Again the security of ECC depends upon how to calculate k when point is given in scalar multiplication [10][4].

<table>
<thead>
<tr>
<th>Symmetric scheme(key size in bits)</th>
<th>Elliptic curve cryptography based schemes (key size in bits)</th>
<th>RSA/DSA(modulus size in bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>56</td>
<td>112</td>
<td>512</td>
</tr>
<tr>
<td>80</td>
<td>160</td>
<td>1024</td>
</tr>
<tr>
<td>112</td>
<td>224</td>
<td>2648</td>
</tr>
<tr>
<td>128</td>
<td>256</td>
<td>3072</td>
</tr>
<tr>
<td>192</td>
<td>384</td>
<td>7680</td>
</tr>
<tr>
<td>256</td>
<td>512</td>
<td>15360</td>
</tr>
</tbody>
</table>

The security levels which is given by RSA can be provided by smaller keys of elliptic curve cryptosystem as compared to RSA, which offers 1024 bit security strength, ECC offers the same in 160 bit key length. [1][2][5][9].

## Table 2
### Comparison of Required Key Size for Various Algorithms Based On Same Level of Security, Data Sizes, Encrypted Message Sizes and Computational Power (By NIST Recommendation)

<table>
<thead>
<tr>
<th>Time to Break in MIPS Years</th>
<th>RSA/DSA Key Size</th>
<th>ECC Key Size</th>
<th>RSA/ECC Key Size Ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^4$</td>
<td>512</td>
<td>106</td>
<td>5:1</td>
</tr>
<tr>
<td>$10^8$</td>
<td>768</td>
<td>32</td>
<td>6:1</td>
</tr>
<tr>
<td>$10^{11}$</td>
<td>1024</td>
<td>163</td>
<td>7:1</td>
</tr>
<tr>
<td>$10^{20}$</td>
<td>2043</td>
<td>210</td>
<td>10:1</td>
</tr>
<tr>
<td>$10^{78}$</td>
<td>21000</td>
<td>600</td>
<td>35:1</td>
</tr>
</tbody>
</table>

Efficiency of ECC is depends upon factors such as computational overhead, key size, bandwidth, ECC provides higher-strength per-bit which include higher speeds, smaller power consumption, bandwidth savings, storage efficiencies, and smaller certificates [12].

For providing security mechanism will require fundamental basic security services such as authentication, confidentiality, non-repudiation and message integrity [5][9]. The implementation ECC shows that it offers complete security solution.

### VI. Performance Parameters For Elliptic Curve Cryptography Implementation

Although RSA, El-GAMAL and Diffie–Hellman are secure asymmetric key cryptosystem, their security comes with a price, their large keys. So researchers have looked for providing substitute that provides the same level of security with smaller keys. For Elliptic Curve Cryptography implementation following consideration should meet [3][15][16] :

- Suitability of methods available for optimizing finite field arithmetic like addition, multiplication, squaring, and inversion.
- Suitability of methods available for optimizing elliptic curve arithmetic like point addition, point doubling, and scalar multiplication.
- Application platform like software, hardware, or firmware.
- Constraints of a particular computing environment e.g., processor speed, storage, code size, gate count, power consumption.
- Constraints of a particular communications environment e.g., bandwidth, response time.

Efficiency of ECC is depends upon factors such as computational overhead, key size, bandwidth, ECC provides higher-strength per-bit which include higher speeds, lower power consumption, bandwidth savings, storage efficiencies, and smaller certificates.

### VI. Conclusion
Secure Access Authentication in mobile communication is very crucial to protect information of the subscribers and avoid fraud. This paper studied security by means of elliptic curve cryptographic technique. Actual implementation of elliptic curve cryptography on GF (P) shows that security of the proposed system is very hard. It has been mentioned in many literatures that a considerably smaller key size can be used for ECC compared to RSA. Also mathematical calculations required by elliptic curve cryptosystem are easier, hence, require a low calculation power. Therefore ECC is a more appropriate cryptosystem to be used on small devices like mobile phones. The ECC has received considerable attention from mathematicians around the world, and no Significant breakthroughs have been made in weaknesses in the algorithm.

REFERENCES

© 2013, IJARCSSE All Rights Reserved