International Journal of Multidisciplinary Research and Development
Online ISSN: 2349-4182, Print ISSN: 2349-5979; Impact Factor: RJIF 5.72
Received: 19-06-2019; Accepted: 21-07-2019
www.allsubjectjournal.com
Volume 6; Issue 9; September 2019; Page No. 167-170

Elliptic curve based schemes

Sanjay Kumar
Department of Mathematics, Kalindi College, University of Delhi, Delhi, India

Abstract
This paper introduces the concepts and arithmetic of elliptic curve cryptography. Elliptic curve cryptosystems require shorter key length then RSA cryptosystem, but provided equivalent security levels. We present different operations on EC over real and prime fields. We also report standard schemes for ECC. Comparison of ECC with RSA and its benefits for cryptography, application are also presented.

Keywords: finite field, elliptic curve, cryptography, key exchange

1. Introduction

The use of elliptic curves in public key cryptography was independently proposed by Neal Koblitz [1] and Victor Miller [2] in 1985 and since then, an enormous amount of work has been done on elliptic curve cryptography. The attractiveness of using elliptic curves arises from the fact that similar level of security can be achieved with considerably shorter keys than in methods based on the difficulties of solving discrete logarithms over integers or integer factorizations.

Elliptical curve cryptography (ECC) is a public key encryption technique based on *elliptic curve theory* that can be used to create faster, smaller, and more efficient cryptographic keys. 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. The technology can be used in conjunction with most public key encryption methods, such as RSA, and Diffie-Hellman. According to some researchers, ECC can yield a level of security with a 164-bit key that other systems require a 1,024-bit key to achieve. Because ECC helps to establish equivalent security with lower computing power and battery resource usage, it is becoming widely used for mobile applications. ECC was developed by Certicom, a mobile e-business security provider. RSA has been developing its own version of ECC. Many manufacturers, including 3COM, Cylink, Motorola, Pitney Bowes, Siemens, TRW, and VeriFone have included support for ECC in their products.

2. Elliptic curves arithmetic

An elliptic curve is defined by the *normal Weierstrass equation*

$$E: y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \quad (1)$$

Where $a_1, a_2, a_3, a_4, a_6 \in K$.

The elliptic curve is the set of points $(x, y) \in K \times K$ that satisfy equation (1) together with the extra point at infinity, O. We denote the elliptic curve by $E(K)$, the set of K-rational points together with O. Observe that by definition, we can write $E = E(K)$. Two elliptic curves E_1 and E_2 are isomorphic over the field K, denoted $E_1 \cong E_2$, if there exist $u, r, s, t \in K$, $u \neq 0$, such that the admissible change of variables

$$(x, y) \rightarrow (u^2x + r, u^3y + u^2sx + t)$$

transforms E_1 into the E_2. If the characteristic p of $K \neq 2, 3$. Then any curve defined over K is isomorphic with a curve of particularly simple form, namely:

$$E : y^2 = x^3 + ax + b; a, b \in K \quad (2)$$

Similarly, one can simplify the Weierstrass equation for curves over finite fields of characteristic $p = 2$ and 3,we will not deal with these cases here. There exists a natural operation that makes the set of points on an elliptic curve into a group. This operation, known as the *tangent-and-chord-method*, is written additively and has the point at infinity O as zero elements.

2.1. Elliptic Curveover Reals

An elliptic curve over real numbers may be defined as the set of points (x, y) which satisfy an elliptic curve equation of the form $y^2 = x^3 + ax + b$, where x, y, a and b are real numbers. Each choice of the numbers a and b yields a different elliptic curve. Suppose a = - 4 and b = 0.67 gives the elliptic curve with equation $y^2 = x^3 - 4x + 0.67$. If $x^3 + ax + b$ contains no repeated factors, or equivalently if $4a^3 + 27b^2$ is not 0, then the elliptic curve $y^2 = x^3 + ax + b$ can be used to form a group. An elliptic curve group over real numbers consists of the points on the corresponding elliptic curve, together with a special point O called the point at infinity.

2.2. Elliptic Curves over Fields F_p

The equation of the elliptic curve on a prime field F_p is $y^2 \bmod p = x^3 + ax + b \bmod p$, where $4a^3 + 27b^2 \bmod p \neq 0$. Here the elements of the finite field are integers between 0 and p – 1. All the operations such as addition, substation, division, multiplication involves integers between 0 and p – 1. The prime number p is chosen such that there is finitely large number of points on the elliptic curve to make the

cryptosystem secure. The graph for this elliptic curve equation is not a smooth curve. Hence the geometrical explanation of point addition and doubling as in real numbers will not work here. However, the algebraic rules for point addition and point doubling can be adapted for elliptic curves over F_p.
Let E be the curve $y^2 = x^3 + x + 1$ over F_5 [14]. To find solutions of this equation in F_5, just consider x = 0, 1, 2, 3, 4 and take square roots to find the corresponding y's. We get
E = {(0, 1), (1, 4), (2, 1), (2, 4), (3, 1), (3, 4), (4, 2), (4, 3), *O*}.

Example (Point Addition)
$P(x_1,y_1)$ = (2, 1) and $Q(x_2,y_2)$ = (3, 4) are distinct points on the elliptic curve $y^2 = x^3 + x + 1$ over F_5. Then we have to find $R(x_3,y_3)$:
$R(x_3,y_3) = P(x_1,y_1) + Q(x_2,y_2)$

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1} \bmod (p) = \frac{4-1}{3-2} = 3$$

$$x_3 = \lambda^2 - x_1 - x_2 = 4$$

$$y_3 = \lambda(x_1 - x_3) - y_1 = 3$$

$\therefore$ $R(x_3, y_3)$ = (4, 3) which is the point of the elliptic curve over F_5.

Example (Point Addition)
In previous example if we take point $p(x_1,y_1)$ = (2,1) and -$P(x_1,-y_1)$ = (2,-1)

$\therefore$ P+ (-P) = O.

Example (Point Doubling)
Suppose point $P(x_1,y_1)$ =(2,1). Then find P+P = R
$2P(x_1,y_1) = R(x,y)$.
Where

$$\lambda = \frac{3x_1^2 + 1}{2y_1} \bmod(p) = \frac{3*2^2+1}{2*1} \bmod(5) = 4$$

$$x = \lambda^2 - 2x_1 \bmod(p) = 4^2 - 2*2 \bmod(5) = 2$$

$$y = -y_1 + \lambda(x_1 - x) mod(p) = -1 + 4(2-2) mod(5) = 4$$

$\therefore$ R(x,y) = (2,4)

3. Elliptic curve cryptosystems
Elliptic curve cryptosystems (ECCs) include key distribution, encryption algorithms. The key distribution algorithm is used to share a secret key and the encryption algorithm enables confidential communication. ECCs are based on the addition of rational points on a chosen elliptic curve. An elliptic curve E over the finite field GF(p) where p is a prime, is the set of points (x,y) satisfying the following equation:

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

Where a, b are integer modulo p, satisfying: $4a^3 + 27b^2 \neq 0 \pmod p$, and include an point O called point at infinity.

3.1. An Elliptic Curve ElGamal Cryptosystem [3]
Bob chooses an elliptic curve E (mod p), where p is a large prime. He chooses a point on E and a secret integer a. He computes $\beta = a\alpha = (\alpha + \alpha + \alpha + ...$a times). The points α and β are made public, while a is kept secret. Alice expresses her message as a point x on E. She chooses a random integer k, computes $y_1 = k\alpha$ and $y_2 = x + k\beta$, and sends the pair y_1, y_2 to Bob. Bob decrypts by calculating $x = y_2 - ay_1$.
Hence, the decryption yields the correct plaintext. There are some practical difficulties in implementing an ElGamal Cryptosystem on an elliptic curve. This system, when implemented in Z_p (or in GF (p^n) with n > 1) has a message expansion factor of two, An elliptic curve implementation has a message expansion factor of (about) four. This happens since there are approximately p plaintexts, but each ciphertext consists of four field elements. A more serious problem is that the plaintext space consists of the points on the curve E, and there is no convenient method known of deterministically generating points on E.

3.2. Elliptic curve Menezes-vanstone cryptosystem
Menezes-Vanstone Elliptic Curve Cryptosystem [4] is a solution to the problem of encoding a message in a point. It uses a point on an elliptic curve to mask a point in the plane. It is fast and simple. Let H be a cyclic subgroup of Ep (a, b) with the generator G. Bob has a private key n_B, and a public key n_BG. The message M is converted into a point P_M = (x, y) in GF (p).

Encryption algorithm
- Alice select a random number r < |H|, and calculates r $n_BG = (x_k, y_k)$.
- Alice sends (rG, x_kx mod p, y_ky mod p) to Bob.

Decryption algorithm
- Bob calculates n_Br G = r $n_BG = (x_k, y_k)$.
- Bob recovers x and y by $x_{k-1}x_k$x mod p and $y_{k-1}y_k$y mod p.
- Bob converts the point(x, y) to get the original plaintext M.

3.3. Elliptic curve diffie-hellman keyexchange [13]
Elliptic curve Deffie-Hellman key exchange was first introduced by Diffie and Hellman in the year 1976. Now we explain the implementation of elliptic curve Deffie-Hellman key exchange. Alice and Bob want to exchange a key. Thus, they agreed on a public point generator or the base point G on an elliptic curve $y^2 \equiv x^3 + ax + b \pmod p$. Now choose p=7211 and a=1, b=7206 and the point G = (3,5). Alice chooses a random integer k_A = 12 and Bob chooses random integer k_B = 23. Alice and Bob keep these private to themselves but publish the k_AG and k_BG.in this case we have
k_AG = (1794,6365) and k_BG = (3861,1242)
Alice now takes k_BG and multiples by k_A to get :
$k_A(k_BG)$ = 12(3861,1242) = (1472,2098).
Similarly, Bob takes k_AG and multiples by k_B to get the key:
$K_B(k_AG)$ = 23(1794,6375) = (1472,2098)
Therefore Alice and Bob have the same key.

3.4 Elliptic Curve Integrated Encryption Schem (ECIES)

The ECIES scheme is an elliptic curve variant of the famous ElGamal public key encryption scheme [8, chap. 6]. It was proposed by Bellare and Rogaway [9], Cramer and Shoup [10] showed the scheme secure against adaptive chosen ciphertext attacks, under the Random Oracle model and the elliptic curve Gap Diffie–Hellman assumption (which is: given an efficient ECDDHP solver, the ECDHP problem remains hard). The ECIES scheme is standardized in ANSI X9.63 [11] and IEEE P1363 [12]. An ECIES encryption is as follows; KDF is a key derivation function, Enc is a symmetric encryption scheme, and MAC a message authentication scheme.

ECIES Encryption

Input: Domain parameters Ψ = (q, FR, S, a, b, P, n, h), a public key Q, and a message m**.**

Output: A ciphertext c = (R,C,t).

1. Choose k ∈ R[1, n−1].
2. Compute R = kP and Z = hkQ.
   a. If Z = ∞, go to step (3).
   b. Else, destroy k.
3. Compute (K_1, K_2) = KDF (xZ,R), where xZ is the x–coordinate of Z.
4. Compute C = Enc_{K_1}(m), and t = MAC_{K_2}(C).
5. Return c =(R, C, t**).**

ECIES Decryption

Input: A domain parameters Ψ = (q, FR, S, a, b, P, n, h), a private key d, and a ciphertext c = (R,C,t).

Output: A plaintext m or "failure" (i.e, ciphertext rejection).

1. Validate the public key R, if the validation fails, return "failure".
2. Compute Z = hdR, if Z = ∞, return "failure".
3. Compute K_1, K_2 = KDF(xZ,R).
4. Verify that t =MAC_{K_2}(C), if not return "failure".
5. Return m = Dec_{K_1}(m).

4. Security of elliptic curve cryptosystems

Because difficulty of the ECDLP, highly secure systems can be designed that require much smaller key sizes than RSA in order to achieve comparable levels of security. ECC demands less resources. On the server, no particular performance need for switching to ECC. In the client, there are good reasons. Table gives approximate parameter sizes for comparable strength elliptic curve systems and RSA. This is based on current best techniques for solving the ECDLP and factorising large integers. Consequently, using elliptic curves, we can define highly secure systems that use much smaller keys compared with equivalent "traditional" systems, such as RSA or DSA. In particular, such systems require relatively modest computing capability and memory - ideal, for example, for a smart card or mobile phone [5, 6]

Table 1: Equivalent key sizes for ECC and RSA

Elliptic curve system	RSA	Key Size Ratio
160 bits	1024 bits	1:6
224 bits	2048 bits	1:9
256 bits	3072 bits	1:12
384 bits	7680 bits	1:20
512 bits	15360 bits	1:30

The security of ECC depends on how difficult it is to determine k given kP and P. This referred to as the elliptic curve logarithm problem. The fastest known technique for taking the elliptic curve logarithm is known as the Pollard rho method. As can be seen, a considerably smaller key size can be used for ECC compared to RSA is. Furthermore, for equal key lengths, the computational effort required for ECC and RSA is comparable. Thus, there is a computational advantage to using ECC with a shorter key length than a comparably secure RSA [5, 6].

5. Some problems and issues with elliptic curve systems

When we discuss the difficulty of solving hard problems, we normally do so in terms of the size of the problem facing the cryptanalyst. For RSA, the size of the problem is the length of the modulus that must be factored. For elliptic curve cryptosystems the size of the problem is the number of points N in the group we are working with. The elliptic curve discrete logarithm problem seems to be particularly hard to solve. Several algorithms might be used that have a running time that depends on the square root of N where N is the number of points in the group in which operations are performed.

It is interesting to note that such algorithms were among those used for factoring or solving the discrete logarithm problem when RSA was first proposed. The introduction of cryptosystems based on factoring and the discrete logarithm problem prompted developments in finding solutions to both problems.

It appears that an elliptic curve cryptosystem implemented over the 160-bit field GF(2^{160}) offers roughly the same resistance to attack as would a 1024-bit RSA. This currently offers the opportunity to use shorter keys than with RSA which might lead to better storage requirements and improved performance.

5.1. Elliptic Curve Generation and Security

The main issue is that the true difficulty of the ECDLP is not fully understood. Recent research has shown that some elliptic curves that were believed suitable for elliptic curve cryptography are in fact not appropriate. For example, if the order of the base point P is equal to the prime p then it turns out that the ECDLP can be solved efficiently[5,6].

When defining an elliptic curve system, a curve and a base point (P) are required. Note that these elements are not secret. For a given curve and base point, it is trivial to generate public and private keys for users. The difficulty of the ECDLP means that it is infeasible to deduce the private key from the public key. However, it is an extremely difficult problem to generate a suitable curve and base point in the first place. The main problem is how to count the number of points on the curve. Having done this, it is then necessary to select a suitable base point P, which must have a large order to ensure the difficulty of the ECDLP. But the order of P must divide the number of points on the curve. So, having found the number of points on the curve, it is quite likely that a suitable base point cannot be found. There are a variety of other restrictions that must be satisfied when generating curves [7].

6. Conclusions

We have briefly described operations on elliptic curves, elliptic curve based schemes and its security issues. Elliptic Curve Cryptography provides greater security and more

efficient performance than the first generation public key techniques (RSA and Diffie-Hellman) now in use.ECC is a stronger option than the RSA and is the discrete logarithm systems for the future. ECC is an excellent choice for doing asymmetric cryptography in mobile, portable, and necessarily constrained devices.

7. References

1. Koblitz N. Elliptic curve cryptosystems, Mathematics of Computation. 1987; 48:203209.
2. Miller V. Use of elliptic curves in cryptography, Advances in Cryptology CRYPTO'85 (LNCS 218). 1986; 483:417–426.
3. Ali Aydin M, Zeynep Aydin G. A survey of elliptic curve cryptography journal of electrical & electronics engineering. 2006; 6:2.
4. Menezes A, Vanstone SA. Elliptic Curve Cryptosystems and Their Implementation, Journal of Cryptology. 1993; 6:209-224.
5. Aydos M, Savaş E, Koç ÇK. Implementing network security protocols based on elliptic curve cryptography", Proceedings of the Fourth Symposium on Computer Networks, Istanbul, Turkey, 1999, 20-21, 130-139.
6. Çetin Kaya KOÇ. Cryptography: State of the Art and Current Trends", Istanbul, Turkey, SACIS, 2003.
7. http://www.certicom.com/resources/ecc_tutorial/ecc_tutorial.html
8. Stinson DR. Cryptography: Theory and Practice. CRC Press, 1995.
9. Bellare M, Rogaway P. Minimizing the Use of Random Oracles in Authenticated Encryption Schemes. In Proc. of the First International Conference on Information and Communication Security, Lecture Notes In Computer Science, Springer Verlag. 1997; 1334:1-16.
10. Cramer R, Shoup V. Design and Analysis of Practical Public Key Encryption Schemes Secure against Adaptive Chosen Ciphertext Attack, SIAM Journal on Computing. 2004; 33(1):167-226.
11. ANSI X9.63: Public Key Cryptography for the Financial Services Industry: Key Agreement and Key Transport using Elliptic Curve Cryptography, ANSI, 2001.
12. IEEE 1363: Standard Specifications for Public Key Cryptography, IEEE, 2000.
13. HailizaKamarulhaili and LiewKhangJie: "Elliptic Curve Cryptography and Point Counting Algorithms, Accessed on 2013 from www.intechopen.com.
14. Martin Leslie. Elliptic Curve Cryptography", Advanced Combinatorics, 2006.