Elliptic curve cryptography using Koblitz encoding method

Kanaklata verma
E&Tc , ME(VLSI)
SSCET, Bhilai, India

Himani Agrawal
Asso prof in E&Tc Deptt.
SSCET, Bhilai, India

ABSTRACT

Elliptic curve cryptography (ECC) is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields .The principal attraction of ECC compared to RSA is that it offers equal security for a smaller bit size, thereby reducing processing overhead. The Encoding(converting message to a point) and Decoding (converting a point to a message) are important functions in Encryption and Decryption in ECC. The paper discusses Koblitz's method to represent a message to a point and vice-versa. The paper also describes implementation results of Koblitz's Encoding methods.

Keywords: Elliptic curve cryptography (ECC), public key , private key.

INTRODUCTION

Elliptic curve cryptography (ECC) is an approach to public-key cryptography based on the algebraic structure of elliptic curves overfinite fields. The use of elliptic curves in cryptography was suggested independently by Neal Koblitz and Victor S. Miller in 1985. Elliptic Curve Cryptography (ECC) is a public key cryptography. In public key

cryptography each user or the device taking part in the communication generally have a pair of keys, a public key and a private key, and a set of operations associated with the keys to do the cryptographic operations.

The mathematical operations of ECC is defined over the elliptic curve $Y^2=x^3+ax+b$

Each value of the 'a' and 'b' gives a different elliptic curve. All points (x, y) which satisfies the above equation plus a point at infinity lies on the elliptic curve. The public key is a point in the curve and the private key is a random number. The public key is obtained by multiplying the private key with the generator point G in the curve. The generator point G, the curve parameters 'a' and 'b', together with few more constants constitutes the domain parameter of ECC. One main advantage of ECC is its small key size. A 160-bit key in ECC is considered to be
as secured as 1024-bit key in RSA.

RSA IMPLEMENTATION:

Proposed in 1977 and named after its inventors, Ron Rivest, Adi Shamir and Leonard Adleman. RSA is a asymmetric key cryptography, it requires larger key length and gives more security. It is not used for encryption of plaintext message, used only for exchanging secret keys because of its power

consumption. RSA encryption is based on the difficulty of the integer factorisation problem, and transforms the message M into the number C.

$$C = M^e \bmod N$$

The numbers e and N are the two public numbers created and published. They are public keys. Message M can be simply the digital value of a block of ASCII characters. The formula states: multiply the Message M by itself e times, then divide the result by the number N and save only the remainder, called C is the encrypted representation of the message.

ENCRYPTION/ DECRYPTION:

Bob sending message to Alice. Bob encrypts the message using Alice public key.

Cipher Text C ← M^e (mod N)

Alice decrypts the message using, private key (d), which is not made public.

Message text ← C^d (mod N)

Disadvantage:

RSA is a highly secure algorithm, provided the keys are generated properly, the only way to attack is to perform a brute- force attack on the modulus. This attack can be simply defeated by increasing the key size. Problems in RSA algorithm are

- Increased processing time – Decryption time increases approximately eight fold times as key sizes double.
- Increased key storage requirement – Key storage requires significant amount of memory for storage.
- Key generation is complex and time consuming. Time increases considerably as the key size increases.
- More computation overhead, exponentiation operation takes place, (it requires more multiplications and modular operations), requires more power.

ECC PUBLIC KEY CRYPTOSYSTEM

In the public key elliptic curve cryptosystems, we assume that entity A wants to send a message m to entity B securely. Order of a point on the curve can be defined as a value n such that
nP = P+P+..+P.. n times = O (infinity)

KEY GENERATION:

Both the entities in the cryptosystem agree upon a,b,p,G,n which are called 'Domain Parameters' of ECC. G is called generator point and n is the order of G. Now A generates a random number nA < n as his private Key and calculates his public key Set PA = G+G+G…+nA times. B generates a random number nB < n as his private Key and calculates his public key, set PB = G+G+G…+nB times..

KEY EXCHANGE:

Entity A computes his Shared Key by Computing K = PA + PA +..+… nB times
Entity A computes his Shared Key by Computing K = PB + PB +..+… nA times
The two above keys have same value because: nA*PB = nA* (nB*G)= nB*(nA *G)= nB * PA

ENCRYPTION:
A sends Cm = 2 ciphertext points those are { kG, Pm + k PB) }.
Where G - generator Point
Pm - plaintext point on the curve
k - a random number chosen by A
PB - public key of B

DECRYPTION:
Pm + kPB - nB(kG) = Pm + k(nB)G - nB(kG) = Pm

ENCODING AND DECODING A MESSAGE IN THE IMPLEMENTATION OF ECC
ECC Encryption and Decryption methods can only encrypt and decrypt a point on the curve not messages. Unfortunately, there are no known polynomial time algorithms for finding a large number of points on an arbitrary curve. We are not simply looking for random points on E, here. We want a systematic way of finding points on Ep(a,b) relating somehow to the plain text message. Therefore, we are forced to use probabilistic algorithms to do this, where the chance of failure is acceptably small. Thus Encoding(message to a point) and Decoding (point to a message) methods are important while Encryption and Decryption.

MESSAGE ENCODING AND DECODING
Let us suppose a text file has to be encrypted, a user can encrypt the ASCII code of each and every printable character on the keyboard , let us say he has to encrypt an 8-bit number , can represent 128 characters on the keyboard. Fig shows the sequence of steps to be followed when a message to be encrypted and decrypted using elliptic Curve Cryptography.
All the points on the elliptic curve can be directly mapped to an ASCII value, select a curve on which we will get a minimum of 128 points, so that we fix each point on the curve to an ASCII value. For example, 'ENCRYPT' can be written as sequence of ASCII characters that is ' 69' '78' '67' '82' '89' '80' '84' we can map these values to fixed points on the curve. This is easiest method for embedding a message but less efficient in terms of security. The steps to be followed during encoding and decoding.

ELLIPTICAL CURVE CRYPTOGRAPHY
Background

Public key cryptosystem gain more popularity since it was proposed by W. Diffie and M. Hellman in 1976, foundation of every cryptosystem is a hard mathematical problem that seems infeasible to solve. The techniques of the public key cryptosystems are classified into three categories,

1. Based on integer factorisation problem, such as RSA [3].
2. Based on discrete log, such as Digital Signature Algorithm (DSA) [4].
3. Based on Elliptic curve, such as Elliptic curve Diffie Hellman (ECDH) [5].

Security degree of all the techniques depends on the hardness of mathematical problem. Elliptic curve is harder to solve, i.e. it takes full exponential time compare to other techniques. ElGamal proposes use of discrete log problem in asymmetric key cryptography in 1985. Elliptic curves are used in mathematics many years before but it can be used in the implementation of asymmetric key cryptography as suggested by Neal Koblitz and Miller independently in 1985.

Elliptic curve group is defined over non- homogeneous (affine) weierstrass equation,

$$y^2 + axy + by = x^3 + cx^2 + dx + e$$

Where a, b, c, d, e are real numbers. Elliptic curve cryptography is defined over special case of equation is

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

Where a and b are real numbers. Necessary condition implement elliptic curve in cryptography, curve should be non-singular, condition for non-singular curve is $4a^3+27b^2 \neq 0$. $4a^3+27b^2$ represents as Δ. All points (x, y) satisfying the equation (2) together point at infinity (O) lies on the elliptic curve. Elliptic curve group can be obtained by varying different a and b values.

ELLIPTIC CURVE CRYPTOSYSTEMS

Discrete log cryptosystems are typically described in the setting of the multiplicative group of the integers modulo a prime p. Such systems can be modified to work in the group of points on an elliptic curve. For instance, the Diffie–Hellman key agreement protocol can be adapted for elliptic curves as follows. First note that a "random" point on an elliptic curve E can serve as a key, since Alice and Bob can agree in advance on a method to convert itto an integer (for example, they can take the image of its x-coordinate under some agreed upon simple map from Fq to the natural numbers).So suppose that E is an elliptic curve over Fq , and Q is an agreed upon (and publicly known) point on the curve. Alice secretly chooses a random integer kA and computes the point kA Q, which she sends to Bob. Likewise, Bob secretly chooses a random kB,

computes kB Q, and sends it to Alice. The common key is P D kAkB Q. Alice computes P by multiplying the point she received from Bob by her secret kA; Bob computes P by multiplying the point he received from Alice by his secret kB. An eavesdropper who wanted to spy on Alice and Bob would have to determine P D kAkB Q knowing Q, kA Q, and kB Q, but not kA or kB. The eavesdropper's task is called the "Diffie–Hellman problem for elliptic curves."

It is not hard to modify the Diffie–Hellman protocol for the purpose of message transmission, using an idea of ElGamal . Suppose that the set of message units has been embedded in E in some agreed upon way, and Bob wants to send Alice a message M 2 E. Alice and Bob have already exchanged kA Q and kB Q as in Diffie–Hellman. Bob now chooses another secret random integer l, and sends Alice the pair of points .l Q; M C l.kA Q//. To decipher the message, Alice multiplies the first point in the pair by her secret kA and then subtracts the result from the second point in the pair.We next describe the elliptic curve analogue (ECDSA) of the U.S. government digital signature algorithm (DSA). The ECDSA is an ANSI standard and is also being considered by the ANSI X9F1 and IEEE P1363 standards committees as a digital signature standard.

An elliptic curve point can be represented as (x, y), where x is the x-coordinate and y is the y-coordinate. It is desirable to represent a point using a compact representation in order to reduce the required bandwidth/memory. One trivial way to do this is to represent a point by its x-coordinate and an additional bit. Elliptic curve is quadratic equation, each x- coordinate there exists two possible y-points on the curve. So, 1-bit is sufficient for to differentiate y-coordinate. Additional bit is used to represent actual y-coordinate. Solving quadratic operation requires square root operation. So, this technique is not practical because square root operation is computation overhead. Majid Khabbazian introduces a novel approach for double point compression technique and extended it to triple point compression . The compression process of the proposed schemes requires almost no computational effort and decompression involves no square root operations. Using compression techniques save memory up to 25 percent.

ECC commercially accepted and adopted by many standardising bodies such as ANSI, IEEE, ISO and NIST . Many products are using ECC due to same level of security with smaller key size than conventional public key cryptosystems. Elliptic curve cryptography has moved from being an interesting theoretical alternative to being a cutting edge technology adopted by an increasing number of companies. ECC has withstood a generation of attacks; and in the growing wireless industry, its advantages over RSA have made it an attractive security alternative . Wireless Internet mail industry leaders such as Qualcomm have embraced ECC, as well as other major companies in the wireless industry such as Motorola, Docomo, and RIM. Major computer companies such as IBM, Sun Microsystems, Microsoft, and Hewlett-Packard are all investing in ECC . Smartcard companies such as Gemplus are also using ECC to improve their products' security. Wireless devices are rapidly becoming more dependent on security features such as the ability to do secure email, secure Web browsing, and virtual private networking to corporate networks, and ECC allows more efficient implementation of all of these features.

CONCLUSION

Elliptic curve cryptography has been emerged as a vast field of interest for application specific security requirements. It has its roots into the number theory which was already used for cryptographic applications before ECC. The elliptic curve discrete logarithm problem makes ECC most efficient with smaller key size compared to earlier RSA algorithm. It is mostly considered for resource constrained devices. Research in the field of Elliptic Curve Cryptography has emerged in various directions to analyze its proper implementation on hardware as well as software platforms.

REFERENCE

1. Neal Koblitz Et Al. "The State Of Elliptic Curve Cryptography" In 2000.
2. Levent Ertaul† And Nitu J. Chavan Et Al "Elliptic Curve Cryptography Based Threshold Cryptography (Ecc-Tc) Implementation For Manets" In 2007.
3. Tarun Narayan Shankar Et Al. "Cryptography With Elliptic Curves" In 2009.
4. V. Gayoso Martínez Et Al. "A Survey Of The Elliptic Curve Integrated Encryption Scheme" In 2010.
5. Rahat Afreen Et Al . "Review On Elliptic Curve Cryptography For Embedded Systems" In 2011.
6. D. Hankerson, A. Menezes, S. Vanstone, "Guide To Elliptic Curve Cryptography".In 2004.
7. S. C. Shantz, "From Euclid's Gcd To Montgomery Multiplication To The Great Divide,"Sun Microsystems Technical Report, June 2001.
8. M. W Paryasto, Kuspriyanto, S. Sutikno, A. Sasongko, "Issues In Elliptic Curve Cryptographic Implementation", Internetworking Indonesial Journal, 2009.