A Discussion on Elliptic Curve Cryptography and Its Applications

Sharad Kumar Verma[1] and Dr. D.B. Ojha[2]

[1]Research Scholar, Mewar University, Rajasthan, India

[2]Professor, Department of Mathematics, Mewar Institute of Technology, Ghaziabad, UP, India

Abstract

Elliptic curve cryptography (ECC) is a kind of public key cryptosystem like RSA. But it differs from RSA in its quicker evolving capacity and by providing attractive and alternative way to researchers of cryptographic algorithm. The security level which is given by RSA can be provided even by smaller keys of ECC (for example, a 160 bit ECC has roughly the same security strength as 1024 bit RSA). In this paper, we will present some ECC algorithms and also gives mathematical explanations on the working of these algorithms.

Keywords: *Elliptic Curve, cryptography, cryptosystem, RSA.*

1. Introduction

Elliptic curve cryptography was introduced in the mid-1980s independently by Koblitz and Miller [3] as a promising alternative for cryptographic protocols based on the discrete logarithm problem in the multiplicative group of a finite field (e.g., Diffie-Hellman key exchange [5] or ElGamal encryption/signature [8]). ECC is a kind of public key cryptosystem like RSA. But it differs from RSA in its quicker evolving capacity and by providing attractive and alternative way to researchers of cryptographic algorithm. The security level which is given by RSA, can be provided even by smaller keys of ECC. For example, the 1024 bit security strength of a RSA could be offered by 163 bit security strength of ECC.

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, and was recently licensed by Hifn, a manufacturer of integrated circuitry (IC) and network security products. 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.

An elliptic curve is not an *ellipse* (oval shape), but is represented as a looping line intersecting two axes (lines on a graph used to indicate the position of a point). ECC is based on properties of a particular type of equation created from the mathematical group (a set of values for which operations can be performed on any two members of the group to produce a third member) derived from points where the line intersects the axes. Multiplying a point on the curve by a number will produce another point on the curve, but it is very difficult to find what number was used, even if you know the original point and the result. Equations based on elliptic curves have a characteristic that is very valuable for cryptography purposes: they are relatively easy to perform, and extremely difficult to reverse.

For current cryptographic purposes, an *elliptic curve* is a plane curve which consists of the points satisfying the equation 1 along with a distinguished point at infinity (∞). This set together with the group operation of the elliptic group theory form an Abelian group, with the point at infinity as identity element. The structure of the group is inherited from the divisor group of the underlying algebraic variety.

The entire security of ECC depends on the ability to compute a point multiplication and the inability to compute the multiplicand given the original and product points [4]. The ECC standards are specified in SEC, Standards for Efficient Cryptography [7].

1.1 Domain parameters

There are certain public constants that are shared between parties involved in secured and trusted ECC communication. This includes curve parameter **a**, **b**, a generator point **G** in the chosen curve, the modulus **p**, order of the curve **n** and the cofactor **h**. There are several standard domain parameters defined by SEC, Standards for Efficient Cryptography [6].

1.2 Point multiplication

Point multiplication is the central operation in ECC. In point multiplication a point P on the elliptic curve is multiplied with a scalar k using elliptic curve equation to obtain another point Q on the same elliptic curve. i.e. k*P=Q

Point multiplication is achieved by two basic elliptic curve operations

- Point addition, adding two points J and K using elliptic curve equation to obtain another point L i.e., L = J + K.
- Point doubling, adding a point J to itself using elliptic curve equation to obtain another point L i.e. L = 2J.

Here is a simple example of point multiplication.

Let P be a point on an elliptic curve. Let k be a scalar that is multiplied with the point P to obtain another point Q on the curve. i.e. to find Q = k*P.

If k = 23 then k*P = 23*P = 2(2(2(2P) + P) + P) + P.

In the ECC explanations given below upper case letter indicates a point in the elliptic curve and the lower case letter indicates a scalar.

1.3 One Way function in ECC

The security of ECC depends on the difficulty of Elliptic Curve Discrete Logarithm Problem. Let P and Q be two points on an elliptic curve such that k*P = Q, where k is a scalar. Q can be easily obtained from P and k but given P and Q, it is computationally infeasible to obtain k, if k is sufficiently large. k is the discrete logarithm of Q to the base P.

2. Elliptic Curve cryptography

An overview of EC cryptographic algorithms for key agreement and digital signature are explained below.

2.1 ECDH – Elliptic curve Diffie Hellman

ECDH, a variant of DH, is a key agreement algorithm. For generating a shared secret between A and B using ECDH, both have to agree up on Elliptic Curve domain parameters. An overview of ECDH is given below.

2.1.1 Key Agreement Algorithm

For establishing shared secret between two device A and B

Step1. Let d_A and d_B be the private key of device A and B respectively, Private keys are random number less than n, where n is a domain parameter.

Step2. Let $Q_A = d_A*G$ and $Q_B = d_B*G$ be the public key of device A and B respectively, G is a domain parameter
Step3. A and B exchanged their public keys
Step4. The end A computes $K = (x_K, y_K) = d_A*Q_B$
Step5. The end B computes $L = (x_L, y_L) = d_B*Q_A$
Step6. Since K=L, shared secret is chosen as x_K

2.1.2 ECDH - Mathematical Explanation

To prove the agreed shared secret K and L at both devices A and B are the same
From Step2, Step4 and Step5

$K = d_A*Q_B = d_A*(d_B*G) = (d_B*d_A)*G = d_B*(d_A*G) = d_B*Q_A = L$

Hence K = L, therefore $x_K = x_L$

Since it is practically impossible to find the private key dA or dB from the public key Q_A or Q_B, its not possible to obtain the shared secret for a third party.

2.2 ECDSA - Elliptic curve Digital Signature Algorithm

Digital Signature Standard (DSA) is a public key algorithm that is used for Digital Signature. The DSA standard is specified FIPS 186-2, Digital Signature Standard [2]. ECDSA is a variant of the Digital Signature Algorithm (DSA). For sending a signed message from A to B, both have to agree up on Elliptic Curve domain parameters. Sender A have a key pair consisting of a private key d_A (a randomly selected integer less than n, where n is the order of the curve, an elliptic curve domain parameter) and a public key $Q_A = d_A*G$ (G is the generator point, an elliptic curve domain parameter). An overview of ECDSA process is defined below.

2.2.1 Signing

Consider the device A that signs the data M that it sends to B.

Step7. Let d_A be A's private key
Step8. Calculate m = HASH (M), where HASH is a hash function, such as SHA-1
Step9. Select a random integer k such that $0<k<n$
Step10. Calculate $r = x_1 \bmod n$, where $(x_1, y_1) = k*G$
Step11. Calculate $s = k^{-1}(m + d_A*r) \bmod n$
Step12. The signature is the pair (r, s)

2.2.2 Verification

Step13. Let M be the message and (r, s) be the signature received from A.
Step14. Let Q_A be A's public key. Since Q_A is public, B has access to it.
Step15. Calculate m = HASH (M)
Step16. Calculate $w = s^{-1} \bmod n$
Step17. Calculate $u_1= m*w \bmod n$ and $u_2 = r*w \bmod n$
Step18. Calculate $(x_1, y_1) = u_1*G + u_2*Q_A$
Step19. The signature is valid if $x_1 = r \bmod n$, invalid otherwise.

2.2.3 ECDSA - Mathematical Explanation

From the verification step19, the signature is valid if $x_1=r \bmod n$ ---- [EX1]
But from Step18, x_1 is the x-coordinate of equation $u_1*G+u_2*Q_A$
From Step10 r is the x-coordinate of equation k*G
Thus to prove equation EX1, It has to prove that $u_1*G+u_2*Q_A = k*G$
Substituting the value of u1 and u2 from Step17 the first part of the above equation
$u_1*G+u_2*Q_A = (m*w \bmod n)*G + (r*w \bmod n)*Q_A$
But $Q_A=d_A*G$, therefore
$u_1*G+u_2*Q_A = (m*w \bmod n)*G + (r*w*d_A \bmod n)*G$, i.e.
$u_1*G+u_2*Q_A = (w*(m+r*d_A) \bmod n)*G$
But from Step16, $w=s^{-1} \bmod n$, i.e $s^{-1} \equiv w \bmod n$ therefore
$u_1*G+u_2*Q_A = (s^{-1}*(m+r*d_A) \bmod n)*G$ ----- [EX2]
but from equation Step11 $s=k^{-1}(m+d_A*r) \bmod n$, i.e. $k = s- 1(m+d_A*r) \bmod n$
Substituting k in EX2
$u_1*G + u_2*Q_A = k*G$
Therefore $x_1=r \bmod n$

3. Equations

$$y^2 = x^3 + ax + b \tag{1}$$

4. Conclusion

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. This is paper gives a crystal clear picture of a comparative study between ECC and RSA, ECC's advantages and some application of ECC like ECDSA. The demonstration included some of the theoretical and practical aspects of ECC.

References

[1] Neal Koblitz. Elliptic curve cryptosystems. Mathematics of Computation, 48:203-209, 1987.

[2] FIPS PUB 186-2, Digital Signature Standard (DSS), January 2000.

[3] Victor S. Miller. Use of elliptic curves in cryptography. In H.C. Williams, editor, *Advances in Cryptology CRYPTO'85*, vol. 218 of *Lecture Notes in Computer Science*, pp. 417-426. Springer-Verlag, 1986.

[4] http://en.wikipedia.org/wiki/Elliptic_curve_cryptography.

[5] Whitfield Diffie and Martin E. Hellman. New directions in cryptography, *IEEE Transactions on Information Theory*, 22(6):644-654, 1976.

[6] Certicom, Standards for Efficient Cryptography, SEC 2: Recommended Elliptic Curve Domain Parameters, Version 1.0, September 2000.

[7] Certicom, Standards for Efficient Cryptography, SEC 1: Elliptic Curve Cryptography, Version 1.0, September 2000.

[8] Taher ElGamal. A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Transactions on Information Theory*, 31(4):469- 472, 1985.

Sharad Kumar Verma, received his Bachelor of computer application (BCA) degree from MCRPV, Bhopal(MP), INDIA in 2004, Master of computer application (MCA) degree from UPTU Lucknow(UP), INDIA in 2007, and currently pursuing Ph.D in computer science (Network Security) from MEWAR University, Rajasthan, INDIA. He has more than three years of teaching experience in Meerut Institute of Engineering & Technology, Meerut (UP) INDIA. He is the author/co-author of more than 5 publications in reputed journals. The research fields of interest are Coding Theory and Time Synchronization in Wireless network.

Dr. Deo Brat Ojha, Birth Place & date –Bokaro Steel City, (Jharkhand), INDIA on 05/07/1975. Ph.D from Department of Applied Mathematics, Institute of Technology, Banaras Hindu University, Varanasi (U.P.), INDIA in 2004. The degree field is Optimization Techniques In Mathematical Programming. The major field of study is Functional Analysis. He has more than seven year teaching experience as PROFESSOR & more than eight year research experience. . He is working at MEWAR Institute of Technology, Ghaziabad (U.P.), INDIA. He is the author/co-author of more than 50 publications in technical journals and conferences.