Research Issues on Elliptic Curve Cryptography and Its applications

[1]Dr.R.Shanmugalakshmi and [2]M.Prabu
[1]Assistant Professor, Department of CSE,
Government College of Technology, Coimbatore, India

[2]Research Scholar, Anna University- Coimbatore, Tamil Nadu, India.

Abstract
Developing technologies in the field of network security. The main motive of this paper to instigate the fast developing cryptography researchers and to increase the security development in the field of information security. In this article, a serious discussion about the comparison between ECC and other cryptography algorithms is attempted to shaped this article. More over, this article explains the role in the network security. ECC's uses with smaller keys to provide high security, high speed in a low bandwidth.

Key words:
ECC, cryptographic algorithms, high security, high speed, low bandwidth

1. Introduction

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. Foe example, the 1024 bit security strength of a RSA could be offered by 163 bit security strength of ECC. Other than this, ECC is particularly well suited for wireless communications, like mobile phones, PDAs, smart cards and sensor networks.EC point of multiplication operation is found to be computationally more efficient than RSA exponentiation

2. Mathematical functions

The elliptic curve certainly is not the ellipse shape, they are so named because they are described by cubic equations, similar to those used for calculating the circumferences of an ellipse. In general and cubic equations for elliptic curves take the form [9] [12]

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

Where a, b, c, d and e are real numbers and X and y take on values in the real numbers

An Elliptic curve E is often expressed as the weierstrass equation:

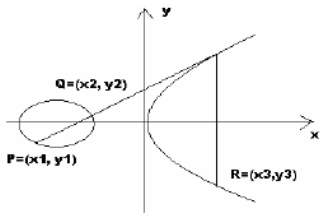
$$Y^2+xy=x^3+ax^2+b$$

Where x, y, a, b ϵF_{2^m}, $b\neq 0$.

There are two operations to describe the abelian group

1. Point Addition

Point Addition: If P(X1,Yl) and Q(X2,Y2) are points on the elliptic curve and if -Xi ≠ X2 (equally P ≠-Q),then, R(X3,Y3)=P+Q can be defined geometrically In the case P ≠Q,a line intersecting the curve at the points P and Q must also intersect the curve at a third point -R, and R(X3, Y3) is the answer, if P=Q (point doubling),the tangent line is used[9][3][4]



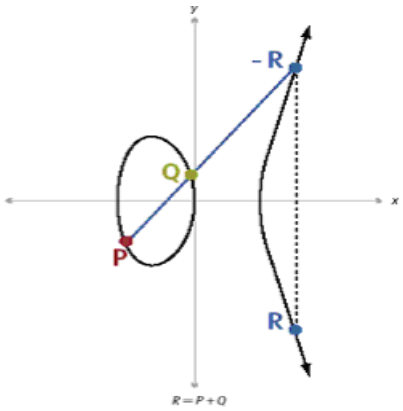
2. Scalar Multiplication (Point Multiplication)

Point Multiplication: Multiplication (also called scalar multiplication) is defined by repeated addition.

$$Q=kP=P+P+....+P.(k \text{ times addition})$$

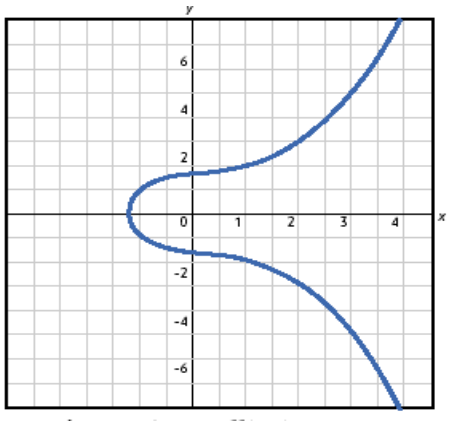
Elliptic curve discrete logarithm problem (ECDLP), which is the ECC's security based on is described as follows. Given an elliptic and a point on it, to determine k from the Q=kP, where Q and P are points on the curve and kP means P added itself k times. It is easy to get Q from k and P, especially for the big numbers. [6][7]

Manuscript received June 5, 2009
Manuscript revised June 20, 2009



3. Elliptic Curve cryptography

Elliptic Curve Cryptography (ECC), which was initially proposed by Victor Miller and Neal Koblitz in 1985, is becoming widely known and accepted. The way that the elliptic curve operations are defined is what gives ECC its higher security at smaller key sizes. [11][10]
The graph turns out to be gently looping lines of various forms.



An elliptic curve

In elliptic curve cryptosystems, the elliptic curve is used to define the members of the set over which the group is calculated, as well as the operations between them which define how math works in the group.

Pros of Elliptic curve cryptography (ECC)

- ECC offers considerably greater security for a given key size. The smaller key size also makes possible much more compact implementations for a given level of security, which means faster cryptographic operations, running on smaller chips or more compact software. This means less heat production and less power consumption.

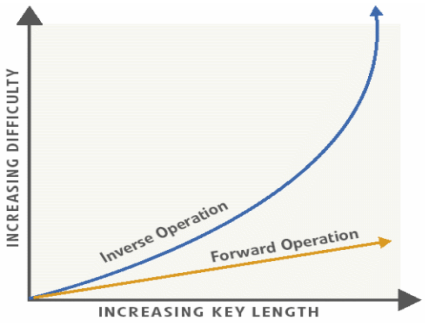
- There are extremely efficient, compact hardware implementations available for ECC exponentiation operations, offering potential reductions in implementation footprint even beyond those due to the smaller key length alone.

Performance of ECC

Its inverse operation gets harder, faster, against increasing key length than do the inverse operations in Diffie Hellman and RSA.
As security requirements become more stringent, and as processing power gets cheaper and more available, ECC becomes the more practical system for use. And as security requirements become more demanding, and processors become more powerful.
This keeps ECC implementations smaller and more efficient than other implementations. ECC can use a considerably shorter key and offer the same level of security as other asymmetric algorithms using much larger ones. Moreover, the gulf between ECC and its competitors in terms of key size required for a given level of security becomes dramatically more pronounced, at higher levels of security.

Security and Future enhancement of ECC

First, the fact that the security and practicality of a given asymmetric relies upon the difference in difficulty between cryptosystems



Second, the fact that the difference in difficulty between the forward and the inverse operation in a given system is a function of the key length in use, due to the fact that the difficulty of the forward and the inverse operations increase as very different functions of the key length, the inverse operations get harder faster.
Third, the fact that as you are forced to use longer key lengths to adjust to the greater processing power now

available to attack the cryptosystem, even the 'legitimate' forward operations get harder.

4. Comparison

Key sizes

Comparison between the two asymmetric cryptographic algorithms such as RSA and ECC, Same level of security, data sizes, encrypted message sizes and computational power. But ECC have smaller keys than other cryptographic algorithms (RSA) [2] [4]

ECC Key Size (bits)	RSA Key Size (bits)	Key Size ratio
160	1024	1:6
224	2048	1:9
256	3072	1:12
384	7680	1:20
512	15360	1:30

Table 1. Equivalent key sizes for ECC and RSA

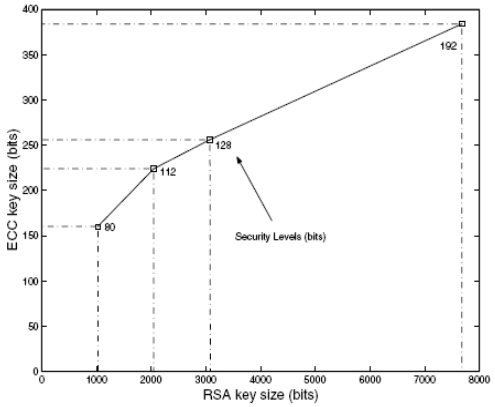


Figure 1. RSA vs. ECC key sizes

ECC offers equal security for a far smaller key size, there by reducing processing overhead the best known algorithm for solving hard the elliptic curve discrete logarithm problem (ECDLP).It takes full exponential time. This means that significantly smaller parameters can be used in ECC than in other systems such as RSA and DSA, but with equivalent levels of security. ECC takes full-exponential time and RSA takes sub-exponential time. For example, RSA with key size of n,1024 bit takes 3*10n MIPS years with best known attack ECC with 160 bit key size takes 9.6*10n MIPS years. ECC offers same level of security with smaller keys. [9]

5. Application of ECC

Elliptic Curve Digital Signature Algorithm (ECDSA)

In the context of this paper, the Elliptic Curve Cryptography is a means for generating signatures. As a consequence, the Elliptic Curve Digital Signature Algorithm (ECDSA)will be used. First an elliptic curve E is defined over $GF(p)$ or $GF(2k)$ with large group of order n and a point P of large order is selected and made public to all users. Then, the following key generation primitive is used by each party to generate the individual public and private key pairs.
Furthermore, for each transaction the signature and verification primitives are used. We briefly outline the Elliptic Curve Digital Signature Algorithm (ECDSA) below, details of which can be found in [11].

ECDSA key generation

The user A follows three steps
1. Select a random integer $d \in [2, n-2]$
2. Compute $Q = d.P$
3. The public and private keys of the user A are (E, P, n, Q) and d, respectively

ECDSA signature generation

The user A signs the message m using three steps:
1. Select a random integer $k \in [2, n-2]$
2. Compute $k.d = (x1, y1)$ and $r = x1 \bmod n$.
If $x1 \in GF(2k)$, it is assumed that $x1$ is represented as a binary number.
If $r = 0$ then go to step 1
3. Compute $k-1 \bmod n$
4. Compute $s = k-1(H(m) - dr) \bmod n$.

Here H is the secure hash algorithm SHA.
If $s = 0$, go to step 1.

5. The signature for the message m is the pair of integers (r, s)

ECDSA signature verification

The user B verifies A's signature (r, s) on the message m by applying the following steps:
1. Compute $c = s-1 \bmod n$ and $H(m)$
2. Compute $u1 = H(m)c \bmod n$ and $u2 = rc \bmod n$
3. Compute $u1.P + u2.Q = (x0, y0)$ and $v = x0 \bmod n$
4. Accept the signature if $v = r$

6. Conclusion

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.
A detailed study of ECDSA is done for our verification.
The performance, security and future enhancement of ECC is discussed.

References

[1] Abhishek Parakh, *"Oblivious Transfer using Elliptic Curves"*Department of Electrical and Computer Engineering",2006,Proceedings of the 15th International Conferences on Computing.

[2] Eugen Petac *"About a method for Distribution keys of a computer netwrok using elliptic curves"*Department of mathematics and Computer Science,1997

[3] Guicheng shen,Xuefeng Zheng, *"Research on Implementation of Elliptic Curve Cryptosystem in E-Commerce"*, International Symposium on Electronic Commerce and Security,2008

[4] G.V.S.Raju and Rehan Akbani,2003, *"Elliptic Curve Cryptosystem and its Applications",*2003,The University of Texas at san Atonio.

[5] Hans Eberle,Nils Gura,Scheduling Chang-Shantz *"A Cryptographic Processor for Arbitary Elliptic Curves Over GF(2^m)",*2003 proceedings of the Aplication-Secific System,Architectures,and Processors(ASAP'03).

[6] jia Xiangya, Wang Chao. "The *Application of Elliptic Curve Cryptosystem in Wireless Communication",* 2005 IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communication,2005

[7] Kristin Lauter,Microsoft Corporation,2004, *"The Advantage of Elliptic Curve Cryptography For Wireless Security"*,IEEE Wireless Communications.

[8] A.Nithin V.S,Deepthi P.P, Dhanaraj K.J,Sathidevi P.S *"Stream ciphers Based on the Elliptic Curves"* national Institute of Technolgy,Calicut,Internatioanl Conference on Computational Intelligence and Multimedia Applications 2007

[9] Sarwono Sutikno,Andy Surya,Ronny Effendi, *"An Implementation of ElGamal Elliptic Curves Cryptosystems",*1998.Integrated System Laboratory,Bandung Institute of Technology.

[10] Qizhi Qiu and Qianxing Xiong,2003, *"Research On Elliptic Curve cryptography"* Wuhan University of Technology

[11] w.stallings, "networks security and cryptography,fourth edition,2001

[12] Yacine Rebahi,Jprdi Jaen Pallares,Gergely Kovacs,Dorgham Sisalem *"Performance Analysis of Identity management in the session Initiation Protocol"*IEEE Journal.

Dr. R.Shanmugalakshmi is working as an Assistant Professor in the Department of Computer Science and Engineering in Government College of Technology, Coimbatore, India. She has published more than 40 International/National journals. Her research area includes Image Processing, Neural Networks, Information Security and Cryptography. She has received Vijya Ratna Award from India International Friendship Society in the year of 1996, she has received Mahila Jyothi Award from Integrated Council for Socio-Economic Progress in the year of 2001 and she has received Eminent Educationalist Award from International Institute of Management, New Delhi in the year of 2008.
She is member of Computer Society of India, ISTE and FIE.

M.Prabu is working as a Lecturer in the Department of Computer Science and Engineering in Adhiyamaan college of Engineering, Housr, Tamil Nadu, India. He is presently doing his Ph.D in Anna University, Coimbatore, India. His area of interest are computer Networks, Information Security and Cryptography. He is life member of ISTE.