ISSN: 2277-9655
Impact Factor: 1.852

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

A Detailed Study of Elliptic Curve Cryptography Algorithm and Its Performance

Dr.K.Ravikumar[*1], A.Udhayakumar[2]
[*1] Assistant Professor & UGC-NET Coordinator, Department of Computer Science
Tamil University (Established by Government of Tamil Nadu), Thanjavur-613010, India
[2] Assistant Professor, A.M.JAIN College Meenambakkam-114, Research Scholar
Karpagam University, Coimbatore-641021, India
ravikasi2001@yahoo.com

Abstract

In this paper, we propose a detailed study of Elliptic Curve Cryptography Algorithm and its performance..ECC can be used with fewer keys to give more security, high speed in a less bandwidth. While these advantages make ECC propose for mobile devices, they can provide computational burden on secure web servers. In resource constrained system, Elliptic Curve Cryptography is a promising alternative for public algorithms, because it provides similar level of security with proposed shorter keys than conventional integer based public key algorithm. ECC over binary field is taken up with special interest because the operation in binary filed operation, are thought to be more in space and efficient in time. However, the software implementation of ECC over binary field are still slow, especially on low end processors, which are used in small computing devices such as sensors node, mobile phone, etc. This proposed paper, studied the Cryptography algorithms and software implementation of ECC. Firstly, while implementing ECC with software, the choice of some architectural parameters like word size may affect the choice of algorithms or not, has been examined. Also, identification of software for low-end processors has been done. In addition, this paper has examined several implements to the instruction that architecture of an 8 bit processor and studied their impact on the performance of ECC with other algorithms. ECC is well is well suited for high speeds, lower power consumption, bandwidth savings, storage efficiencies, smaller certificates and it reduces computational time and also the amount of data transmitted and stored, and strong security for low-power devices in wireless networks.

Keywords: Cryptography, Elliptic Curve, high speed, security, less bandwidth, Digital Signature.

Introduction

Elliptic curve Cryptography(ECC) Algorithm are most suitable for implementation on Small memory devices such as Mobile , smart cards etc., Neal Koblitz and Victor Miller had proposed Elliptic Curve Cryptography in the year 1985, independently in order to use it for various security purposes such as key exchange and digital signature. When compared to traditional integer based public key algorithm, ECC algorithm can achieve the same level of security with much short keys. For instance, 160 –bit Elliptic Curve Digital Signature Algorithm (ECDSA) has a security level equivalent to 1024-bit Digital Signature Algorithm (DSA). It is due to this fact, ECC algorithm runs faster, requires less space, and consumes less energy. These advantages make ECC a better choice of public–key algorithms, especially in resource constrained systems like sensor nodes, mobile devices etc.

ECC's focus mainly lives on mathematical methods and algorithms, hardware implementations and extensions of instructions set architecture. Its software implementation has been discussed by Hankerson and many, in which they focused on 32-bit processors. Its performance has seen compared with RSA on 8-bit processors by Gura and many. Whereas the studied Elliptic Curves by them are in GF $_{(p)}$. Malan security investigated the feasibility of implementing ECC in sensor nodes. The implementations proposed by them were not optimized well. Given that the performance of ECC on low-end processors is far from being satisfactory, many protocols designed for wireless sensor networks tend to use symmetric-key algorithm alone.

This paper tries to identify the problems in the software implementations of ECC and explorer techniques that can accelerate the software implementations. The focus is on ECC over GF (2^m). The first investigation is about whether processor word size may affect our choice of algorithms or not. Since each operation in ECC has many different ways of implementations.. This paper also attempted to study the comparison of cryptographic algorithms.

Elliptic Curve Cryptosystem

Definition: Some comments are in proper order, before the derivation of the running time of the ECC. By selecting the random integers a and b and modulo n, a random curve E is chosen. It turns out that taking a as single-precision integers and b=1 works quite well in practice.

Algorithm.1.Elliptic curve pseudorandom generator

```
Given: A composite integer's n∊N (with no small prime factors).
Output:A non-trial divisor d of n
Procedure:
While (1) {
Select a random curve E: $Y2=x^3+aX+b$ modulo n.
Choose a point p≠Q in E (Zn).
Try to compute mP./*where m is as defined in the text */
If (the computation of mP fails) {
/*we have found a divisor d>1 of n*/
If (d≠n) {Return d}}}
```

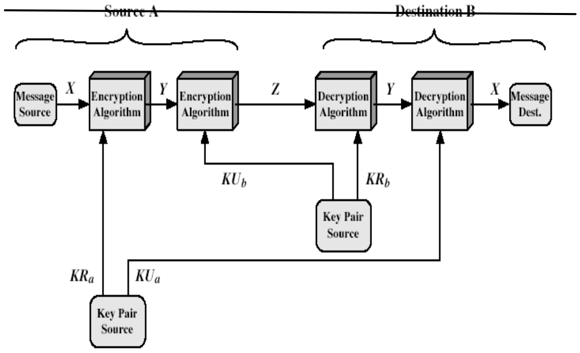


Figure 9.4 Public-Key Cryptosystem: Secrecy and Authentication

Figure 1 Algorithm 1: Elliptic Curve Cryptography with public key Cryptosystem

- The ECC can be effectively parallelized, since different processors can carry out the trials, that is, computation of mP together with the second stage with different sets of random elliptic curves Elliptic curve cryptography [ECC] is a **public-key** cryptosystem just like RSA, Rabin, and El Gamal.
- Every user has a **public** and a **private** key.
  - Public key is used for encryption/signature verification.
  - Private key is used for decryption/signature generation.
- Elliptic curves are used as an extension to other current cryptosystems. (Figure 1).

Mathematical Function

Elliptic curves are not ellipses (the name comes from elliptic integrals). An elliptic curve over real is the set of points (x, y) which satisfy the equation $y^2 = x^3 + a\cdot x + b$, where x, y, a, and b are real numbers. If $4\cdot a^3 + 27\cdot b^2$ is not 0 (i.e. $x^3 + a\cdot x + b$ contains no repeated factors), then the elliptic curve can be used to form a group. An elliptic curve group consists of the points on the curve and a special point O. Elliptic curves additive groups, addition defined geometrically or algebraically (Figure2).

As we have added new instructions, the performance ratio of inversion and multiplication has changed. As a result, adopting projective coordinates may become preferable. Using the new instructions, the inversion is 3.4 times faster than in the baseline architecture, in spite of that multiplication operation has been accelerated even more, when compared with the baseline architecture.As a result,the new instructions is a speed up 9.6 for multiplications.Consequently,the performance ratio of inversion and multiplication has been increased significantly from 5.6 to 15.9.

Adopting projective coordinates to represent the points on the curve is desirable, because the inversion is now much more expensive than the multiplication. This paper adopted the projective co-ordinates discussed Figure 2 and implemented the ECC algorithm with the point addition in mixed coordinates. The performance comparison between affine and projective co-ordinates is shown in the Figure 2.As expected; the projective coordinates have better performance. They are more than twice faster than affine coordinates and achieve a speedup of 8.23 over the base line architecture. Assuming a clock rate of 16MHz, a 163-bit ECC can be performed in about 0.85 seconds

Algorithm .2: Implementing Digital Signature using ECC

Step 1: Encrypt the message using a symmetric key

Step 2: Concatenate the symmetric key + Hash of symmetric key + Hash of message.
Step 3: Encrypt the concatenated string using the receives public key.
Step 4: Sign the data to be transmitted(Encrypted symmetric + Hash of the key + Hash of message).
Step 5: Validate the Signature.
Step 6: Decrypt the message using Receiver private key to get the symmetric key.
Step 7: Validate the integrity of the key using the Hash of the Key.
Step 8: Decrypt the actual message using the symmetric key which has been decrypted and parsed and checked for integrity.
Step 9:Compute Message Digest of data.
Step 10: Validate if the Message Digest of the decrypted text matches the Message Digest of the Original Message.

Figure 3 Algorithm 2: Implementing Digital Signature using ECC

There is no obvious geometric interpretation of elliptic curve arithmetic over finite fields. The algebraic interpretation used for elliptic curve arithmetic over real numbers does readily carry over, and this approach we take (Figure 3).

For set $E_{17}(3,5)$,we are only interested in the nonnegative integers in the quadrant from (0,0) through (p-1,p-1) that satisfy the equation mod p.Table 1 list the points other than O that are part of $E_{23}(1,1)$.

Table 1.Example Points over the elliptic curve E_{23} (1,1)

(0,1)	(6,4)	(12,19)	(0,22)	(6,19)	(13,7)
(1,7)	(7,11)	(13,16)	(1,16)	(7,12)	(17,3)
(3,10)	(9,7)	(17,20)	(3,13)	(9,16)	(18,3)
(4,0)	(11,3)	(18,20)	(5,4)	(11,20)	(19,5)
(5,19)	(12,4)	(19,18)			

Figure 4 Graphical representation of elliptic curve $y^{2}{}_{=x}{}^{3}{}_{+x+1}$

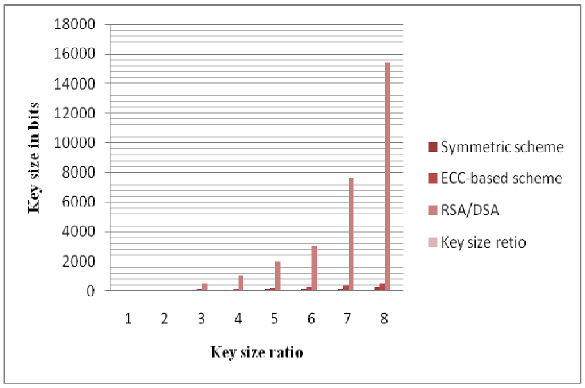
Since that the number of points in E_p (a, b) is approximately equal to the number of elements in Z_p, namely p elements. (Figure 4).

Digital Signature

A Digital Signature is a construct which helps achieve non-repudiation of Origin of data .By digitally signing the document, the person who signs it assures that he is the author of the document or the message that was signed. A digital signature is the electronic equivalent of a handwritten signature. When attaché to an electronic document, it provides authentications of the signer, date and time of signature and contents of the signed document. Digital signatures are much more secure than hand-written signatures. There is no known way to forge a digital signature. Another advantage that digital signatures have over hand-written signatures is applied to the entire message. The only signature scheme are supported at this time is the Elliptic Curve Digital Signature Algorithm(ECDSA). The security of ECC depends on how difficult it is to determine k given kP and P.This is 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. Table 2 compares various algorithms by showing comparable key sizes in terms of computational effort for cryptanalysis. As can be seen, a considerably smaller key size can be used for ECC compared to RSA. 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 short key length than a comparably secure RSA.

Table.2.Comparable Key Sizes for Equivalent Security

Symmetric scheme (key size in bits)	ECC-based scheme (size of *n* in bits)	RSA/DSA (modulus size in bits)	Key size retio
56	112	512	5:1
80	160	1024	6:1
112	224	2048	9:1
128	256	3072	12:1
192	384	7680	20:1
256	512	15360	30:1



The above mentioned table shows that the ECC-based key sizes is much lesser and productive, when compared with RSA/DSA key sizes. The different in the key size goes in an ascending order in 1:4 ratios.

Figure 5 Comparable Key Sizes for Equivalent Security

The above mentioned graph shows that ECC based keys can be used with lesser amount of key size when compared with RSA/DSA keys. In the x- axis the key size is mentioned and in the y- axis key storage with scale of 2000 ms is mentioned.

Performance Evaluation And Analysis

There has been a lot of discussion in the crypto community, especially those interested in the mobile space, about the implementation of ECC algorithm.

Table 3: Performance Evaluation of cryptography algorithm

Curves	Algorithm	Key generation	**Signature**	**Verification**	**Total Time**
Random ($GF(2^{191})$)	**ECDSA-F^{192}**	**11.7**	**11.3**	**60**	**83**
	ECDSA-$F^{P=192}$	**5.5**	**6.3**	**26**	**37.8**
	RSA-1024	**1(SEC)**	**43.3**	**0.65**	**1,043.05**
	DSA	**22.7**	**23.6**	**28.3**	**74.6**
RSA	**RAS-1024**	**2740.87**	**66.56**	**3.86**	**2811.29**
	RSA-2048	**26,442.04**	**440.69**	**13.45**	**26896.18**
DSA	**DSA-768**	**14,735**	**15.55**	**26.13**	**1,4776.48**
	DSA-1024	**54,674**	**24.28**	**47.23**	**5.9421.28**

In the Table 3, Performance evaluation of cryptography algorithm is explained in a detailed manner. In those, key sizes of various cryptography algorithms like DES, RSA, ECC, is mentioned. In addition, their performance evaluation is shown on the basis of their key generation, signature, Verification. . As a result of this evaluation, it is very clear that the key size and key generation of ECC algorithm is much better than the other algorithms. Also, it is understandable that ECC's 160 bits is equal to RSA's 1024 bits.

Table 4:Comparison Table

RSA	ECC
1024	160
2048	282
4096	409

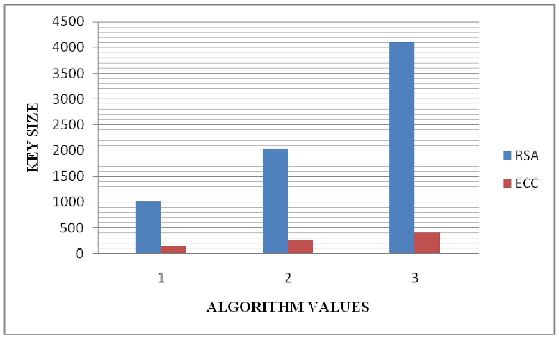


Figure 6. **Performance Evaluation of cryptography algorithm**

The above graph (Figure 6) demonstrates the efficiency of the ECC algorithm when it comes to the minimization of the size of the key generated in the process. Among the various algorithms which are put to use, Elliptic Curve Cryptography generates keys in the smallest possible way when compared with the other cryptographic algorithms, whose keys are the larger among the keys generated by various Cryptography systems. In the X-axis various cryptographic algorithms have been mentioned and in the Y-axis key sizes are mentioned. The graph clearly points that ECC's key consumes lesser amount of keys.

RSA: Uses an public key encrypt and a private key decrypt for key exchange. These scenarios do not perform signing or verification. RSA is modular exponentiation in integer rings and its security stems from the di_culty of factoring large integers. RSA operation are modular exponentiations of large integers with a typical size of 512 to 2048 bits. RSA encryptions generates a cipher text C from a message M based on a modular exponentiation C=Me mod n. Decrypt regenerates the message by computing M=Cd mod n.

ECC: Operates on groups of points over elliptic curves and derives its security from the hardness of the elliptic curve discrete logarithm problem (ECDLP). ECDLP allows ECC to achieve the same level of security with smaller key sizes and higher computational efficiency. ECC -160 provides comparable security to RSA-1024. ECC-224 provides comparable security to RSA-2048.

Conclusions

In this paper, a detailed study of ECC and its performance and it has been examined that how some architectural features, such as key size and ISA, which affect the performance of ECC. Our result show that ECC is faster ,and occupies least memory space than RSA algorithm. The algorithm, for ECC over binary field has been first examined, and after comparing algorithms for the major field operations that are required in ECC, the identification of set of efficient algorithms suitable for resource constrained systems has been done. Besides, the performance of these algorithms for different word sizes has been compared. As a result, the change of word sizes result in different choices of algorithms. The stimulation of the implementation is on an 8-bit micro controller. The proposed implementations are more than twice faster than previous results without instruction set architecture extensions or hardware accelerations.

The performance of ECC application on various mobile devices, we can conclude that ECC is the most suitable PKC scheme for use in a constrained environment. Its efficiency and security makes it an attractive alternative to conventional cryptosystems, like RSA and DSA, not just in constrained devices, but also on powerful computers. It is, without a doubt, fast being recognized as a powerful cryptographic scheme. A detailed study of ECC is done for our verification; implementation of ECC with key exchange is most suitable for Wireless Sensor Network

References

[1] N. gura, a. patel a.wanter “comparing elliptic curve cryptography and RSA on 8-bit cpu, “proceedings of cryptographic hardware and embedded system “2004.

[2] D. Hankerson and A. Menezes “software implementation of elliptic curve cryptographic over binary fields,” proceedings of workshop cryptography hardware embedded system “2000.

[3] D.J. malan , m.welsh “ a public key infrastructure for key distribution in tinyOS based on elliptic curve cryptographic,” 2004.

[4] Atmel corporation, 8-bit microcontroller with 128K bytes in-system programmable flash: AT mega 128, 2004.

[5] N. koblitz, elliptic curve cryptosystem,” mathematics of computation, vol. 48, pp. 203-209, 1987

[6] I. blake, g. seroussi, and n. smart, elliptic curves in cryptography, Cambridge University press, 1999.

[7] L.lopcz and r.dahab, “high speed software multiplication in F(2^m),” proceedings of idocrypto ’00, pp. 203-212, 2000.

[8] V.miller, “uses of elliptic curves in cryptography,” advance in cryptology: proceedings of crypto ’85, pp. 471-426, 1986.

[9] National institute of standards and technology, digital signature standard, FIPS publication 186-2, Feb. 2000

[10] J.solinas “efficient arithmetic on koblitz curves, “designs, codes and crypto graphy, vol. 19, pp.195-249, 2000.