Comparison of Elliptic Curve Key Size in Proposed Blind Signature Scheme

Aye Aye Thu, KhinThanMya
University of Computer Studies (Yangon)
suchiq13@gmail.com, khinthanmya@gmail.com

Abstract

Privacy is one of the basic rights for individuals and institutions that need to preserve their confidentiality. Blind signature is widely used for various applications of e-commerce like digital payment system and electronic voting system where to maintain the privacy of customer is necessary. Most of the blind signature schemes are very high at computational overhead and memory usage problem. And then they need to maintain the security of blind signature scheme to protect the data. In this paper, four scalar multiplication methods are run in efficient blind signature scheme by varying the NIST key size. By saving random factors and key size with suitable bits the system solved the memory usage problem. This paper showed that the comparison of key size with other blind signatures schemes and analyzed the security of proposed scheme. Compare with previous schemes, proposed blind signature with 160 bit ECC is more efficient and low computation and it can be applied in E-voting system.

Keywords—BDS, DLP, ECC, ECDLP, NIST, ECM, PKC, E-voting, E-cash

1. Introduction

Cryptography is one of the most important fields in computer information security. It is a method of transferring private information and data through open network communication. People need information in day to day activities. Now days, online communication is at its hike, many a times data travelling over the communication links is secret and the entire users ought to be authenticated for many of application they use. This is best served by implementing blind digital signature scheme. The blind signature scheme is best implemented in the application where secrecy of the user's data is to be converted.

Elliptic curve cryptosystem (ECC) is accepted to be a secure and efficient public key cryptosystem. Elliptic curve method (ECM) was applied on cryptography known as ECC was discovered in 1985 by victor Miller and Neil Koblitz as an alternative mechanism for implementing public key cryptography (PKC). Elliptic curve cryptography has the special characteristics that to date, the best known algorithm that solves it runs in full exponential time. It security comes from the elliptic curve logarithm, which is the discrete logarithm (DLP) in a group defined by points on an elliptic curve over a finite field.

The objective of this paper is to analyze ECC key size in proposed blind signature scheme. It is also solved the time complexity problem with efficient scalar multiplication methods by analyzing key size. It is intend to improve the performance of blind signature scheme. The scheme reduced memory usage problem and computational overhead.

The structure of the paper is as follows: Section 2 discusses the concept of elliptic curve cryptosystem, blind Signature Scheme and point multiplication method. Section 3 explains an overview of previous approaches on blind signature scheme. In Section 4, propose blind signature is presented. Section 5 provides security analysis of the system. The performance of this scheme is examined in Section 6. Finally,

conclusions and future work are presented in Section 7.

2. Elliptic Curve Cryptosystem (ECC)

Elliptic curve cryptography (ECC) was proposed in 1985 by Neal Koblitz and Victor Miller. Elliptic curve cryptographic scheme can provide the same functionality as RSA schemes which are public key mechanisms. The security is based on the difficulty of a different problem, which is called the Elliptic curve discrete logarithm problem. ECC offered by other traditional public key cryptography schemes used nowadays, with smaller key sizes and memory requirements as shown in Table 1 [3]. For example, it is generally accepted that a 1024 bit RSA key provides the same level of security as a 160 bit elliptic curve key.

The use of shorter keys means lower space requirements for key storage and quicker arithmetic operations. These advantages are essential when public key cryptography is applied in constrained devices, such as in mobile devices or RFID. These advantages are the reason behind choosing ECC as the cryptography system in this paper.

Table 1. NIST Recommended Key Size

Symmetric-Key	ECC	RSA	Comment
64 bit	128 bit	700 bit	Short Period Security
80 bit	160bit	1024 bit	Medium Period Security
128 bit	256 bit	2048	Long Period Security

The elliptic curve consists of a discrete set of points which satisfy the following equation over finite field F_p.

$$y^2 = (x^3 + ax + b) \bmod p$$

where $a, b \in F_p$ and

$$4a^3 + 27\,b^2 \neq 0 \pmod p$$

Point multiplication uses two basic elliptic curve operations:

- Point addition (add two point to find another point)
- Point doubling (adding point p to itself to find another point)

For example to calculate kP = Q if 'K' is 23 then kP = 23P = 2(2(2(2P) +P) +P) +P so to get the result point addition and point doubling is used repeatedly [7]. The elements of the finite field are integers between 0 and p-1 thus all the operations such as addition, subtraction, multiplication, division include integers between 0 and p-1. The prime number 'p' is the finitely large number of points on the EC in order to make the cryptosystem secure.

2.1 Point Addition

Let 'J' and 'K' be the two points on the curve which $J = (x_J, y_J)$ and $K = (x_K, y_K)$ and $L = K + J$ where $L = (x_L, y_L)$ and 's' is the incline of the line through 'J' and 'K' then:

$$s = \frac{y_J - y_K}{x_J - x_K} \bmod p$$

$$x_L = s^2 - x_J - x_K \bmod p$$

$$y_L = -y_J + s(x_J - x_L) \bmod p$$

So if K = -J for example $K = (x_J, -y_J \bmod p)$ then J + K = O, where 'O' is the point at infinity. If K = J, then J + K = 2J needs to use point doubling equation too J + K = K + J.

2.2 Point Subtraction

If 'J' and 'K' are two points on the curve which $J = (x_J, y_J)$ and $K = (x_K, y_K)$ Then:

$$J - K = J + (-K)$$

where $-K = (x_k, -y_k \bmod p)$

In certain implementation of point multiplication such as NAF point subtraction is used [6].

2.3 Point Doubling

Assume J = (x_J, y_J) and $y_J \neq 0$ so the calculating of L = 2J where L = (x_L, $-y_L$) is ;

$$S = \frac{3x_J^2 + a}{2y_J} \mod p$$

('S' is the tangent at point 'J' and 'a' is one of the parameters selected with the ECC)

$$x_L = s^2 - 2x_J \mod p$$

$$y_L = -y_J + s(x_J - x_L) \mod p$$

And O is the point at infinity if $y_J = o$ then 2J= O.

2.4 Elliptic Curve Discrete Logarithm Problem (ECDLP)

The classical or general DLP (discrete logarithm problem) is the following:

If $b=a^k \pmod p$, where p is prime and k is any random integer. DLP is the problem to find k. Similarly, ECDLP is the discrete log problem for elliptic curves.

2.5 Blind Signature

Blind signature scheme is a method of signing that the signer does not see the content of the document [1]. Moreover, if the signer sees the document/sign pairs, he cannot decide for whom or when the document has been signed. So it is similar to signing a document blindly. The blind signature schemes must meet following requirements such as correctness, blindness, unforgeability and untraceability. Most of the schemes cannot satisfy the blind signature scheme's requirements. The proposed scheme can solve these requirements [8] and it is also maintain the time complexity problem.

In order to protect the confidentiality of the message, requester uses a blinding factor to blind the message and sends it to the singer to get the signature. Signer puts his signature on the blinded message and returns the blinded message signature pair. Requester unblinds the signature to get a valid signature on the original signature which can be publicly verified. Blind signature is a variation to the digital signature where the signer is unaware of the content of the message to be signed by him.

2.6 Point Multiplication Algorithms

Scalar multiplication is the principal back-bone procedure in elliptic curve cryptosystems. It is the most frequent method used in scalar multiplication. Elliptic curves have some properties that allow optimization of scalar multiplications. This paper used the binary Double-Add, Ternary expansion, Montgomery ladder, Addition-subtraction algorithm to compute scalar multiplication as described in previous paper [9].

3. Related Works

In [10], Vanstone had concluded that ECC provided roughly 10 times greater efficiency than either integer factorization systems or discrete logarithm systems, in terms of computational overheads, key sizes and bandwidth. This paper focused on the security of elliptic curve cryptography, relying upon the difficulty of solving the elliptic curve discrete logarithm problem.

Jena et.al and group [2] proposed blind signature scheme based on elliptic curve discrete logarithm. This scheme achieves the same security with fewer bit key as compared to RSA. In addition, it has low computation requirement.

In 2009, M. Nikoogadam & A. Zakerolhosseini [4] presented a novel intraceable blind signature scheme. It is also based on discrete logarithm problem and is more efficient than other schemes presented based on discrete logarithm problem. A.S.Samal and chohotaray proposed blind signature [5] based upon elliptic curve cryptography. This scheme suggests a novel blind signature based on the elliptic curve discrete logarithm problem. However, they

needed to solve the computational overhead and consider security problem.

4. The Proposed Blind Signature Scheme

The proposed BDS system contains five phases. They are

1. Initialization
2. Blinding
3. Signing
4. Unblinding and
5. Verifying

In the initialization phase, the signer produced the random number k and run the scalar multiplication steps. Then signer generated the x coordinate as a factor $r' = (x_1 \bmod n)$ and sent R' the requester as a key agreement.

$$R' = kG = (x_1, y_1)$$

In a blinding phase, the requester converted the message *m* into point by using message to point conversion method. Then the requester changed the point into hash code with *SHA-1* algorithm and generated random number v to blind the message. The requester did scalar multiplication and generated the blind factor to produce blind message $m' = H(m)\, r^{-1} r' v \pmod n$ with inversion method. Finally the requester sent back the blind message m' to the singer.

At the signing phase, signer generated the public G and private key d. After that signer signed the blind message $s = dm' + kr' \pmod n$ with his private key *d* and send back blind signature *s* to the requester. In the unblinding phase, requester extracted the signer's digital signature by deducting the blind factor with multiplication and inversion techniques. At the end of the transaction, the requester obtains the singer's signature $s' = sv^{-1} r'^{-1} r \pmod n$ in the original message without revealing the original message to the signer. In the verification phase, the verifier can verify the digital signature by doing inversion, scalar multiplication, conversion and hashing. Later, anyone who has the public key of the signer can verify the signature on message m by using $s' G \stackrel{?}{=} QH(m) + Rr$. Then the signature is verified as valid; otherwise, it is considered invalid. Detail procedure of the proposed blind signature scheme is described and explained in previous paper [8]. Table 2 shows the notation and system parameter of the proposed scheme.

5. Security Analysis

This section examines the attack analysis of the blind signature to fulfill security requirements. The security of the proposed method is based on the difficulty of the ECDLP.

Theorem 1: *The proposed ECDLP based signature scheme can withstand a known message attack.*

Proof: In a known message attack, the attacker has access to some message signature pair. The message signature pairs have been collected earlier. The attacker uses the relationship between the previous pair to analyze the current signature. For example, if the requester has not changed the v, r^{-1} and r′, the attacker can have more information to analyze the system. However, he is not able to verifying stage because the voter will be having public key corresponding to the private key used by the signer for blind message m′. Therefore, the proposed blind signature scheme satisfies the known message attack.

Theorem 2: *The proposed ECDLP based signature scheme can withstand a key only attack.*

Proof: The attacker has to create a valid signature pair. Let the attacker be able to create the signature pair. This implies the signer is compromised, which is quite rare. But, this will not help as in the unblinding stage, the attacker will not be able to unblind the signature pair as it

will not be having the required parameters (v, r, r′). The equation is $s' = sv^{-1}r'^{-1}r \pmod n$ and the extraction of v, r and r′ are impossible due to ECDLP. Therefore, the proposed blind signature scheme achieves the key only attack.

Theorem 3: *The proposed ECDLP based blind signature scheme can withstand chosen message attack.*

Proof : The chosen plaintext attack is similar to the known message attack, the blind message/ signature pairs have been chosen by the attacker herself. For example, if the attacker has access to the requester's computer. He can choose some message and intercepted blind message. However, He cannot have random integers because the integer is normally embedded in the message used by the requester. The proposed signature scheme therefore withstands the chosen message attack.

Table 2.Notation and System parameters

T	Elliptic Curve Domain Parameters
a,b	Coefficients defining the elliptic curve
M	message
G	Base Point
N	Order of G, a Prime Number
h	HashValue
m′	Blinded Message
s	Blind Signature
s′	Signature
r′	X coordinate of R′
r	X coordinate of R
R,R′	Points on Elliptic Curve
D	Private Key of the Signer

6. Performance Evaluation

In order to compare the time consumption of the algorithms, different size of ECC bit run in proposed scheme. The key lengths of the ECC bits are 112, 160, 224, 256, 384 and 512. Firstly, double and add algorithm is run with different key size in ECC as described in Table 3.

Table 3. Comparison of ECC Key Size with Double and Add Algorithm in Proposed Scheme (Milliseconds)

Phases	ECC Key Size (NIST)					
	112	**160**	**224**	**256**	**384**	**512**
Initialization	0.886	2.333	1.89	3.364	2.430	2.548
Blinding	0.386	0.518	0.49	0.483	0.385	0.368
Signing	1.20	1.723	1.99	1.608	2.366	2.623
Unblinding	0.16	0.345	0.50	0.571	0.843	0.874
Verification	15.6	15.58	17.6	16.90	18.58	20.89
Total	**18.232**	**20.499**	**22.47**	**22.926**	**24.604**	**27.303**

In this Table 3, Double and add algorithm with 512 bit is the highest complexity when compared to other bits respectively. In the proposed scheme, 160 bit key size is chosen because the proposed system is need to fast and exact to apply in other applications such as E-cash payment system. So, the system used 160 bits key size with proposed scheme. It is not only maintaining the security but also controlling the performance of proposed scheme.

Table 4. Comparison of ECC Key Size with Ternary Expansion Algorithm in Proposed Scheme (Milliseconds)

Phases	ECC Key Size (NIST)					
	112	**160**	**224**	**256**	**384**	**512**
Initialization	0.864	2.562	2.487	3.211	2.603	2.976
Blinding	0.496	0.705	0.637	0.518	0.625	0.445
Signing	1.333	1.825	1.299	1.504	3.017	3.013
Unblinding	0.189	0.476	0.998	0.582	0.861	0.974
Verification	15.593	15.200	16.409	17.120	32.84	19.97
Total	**18.475**	**20.768**	**21.83**	**22.935**	**39.946**	**27.378**

According to the table, the complexity of the 384 bit ECC is higher than the others and 112 bit ECC is lowest in all phases.

Table 5. Comparison of ECC Key Size with Montgomery Ladder Algorithm in Proposed Scheme (Milliseconds)

Phases	ECC Key Size (NIST)					
	112	**160**	**224**	**256**	**384**	**512**
Initialization	2.192	4.148	3.606	3.871	5.122	5.593
Blinding	0.576	0.775	0.514	0.549	5.122	5.593
Signing	1.893	3.060	1.828	2.974	3.903	4.489
Unblinding	0.136	1.377	0.431	0.628	0.893	1.014
Verification	15.189	21.472	19.915	18.007	21.096	23.048
Total	**19.986**	**30.382**	**26.294**	**26.029**	**36.136**	**39.737**

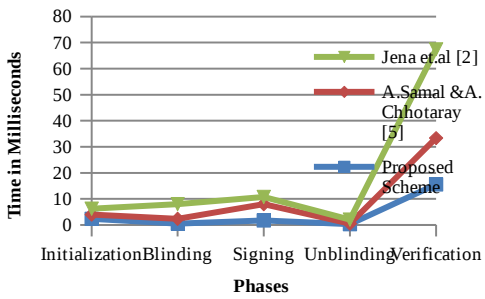
Table 5 provided the comparisons of ECC key size with Montgomery Ladder algorithm in proposed scheme. In this comparison, the average complexity of 512 bit ECC is higher than the others.

Table 6. Comparison of ECC Key Size with Addition-Subtraction Algorithm in Proposed Scheme (Milliseconds)

Phases	ECC Key Size (NIST)					
	112	**160**	**224**	**256**	**384**	**512**
Initialization	9.892	3.994	10.197	4.933	10.362	4.138
Blinding	9.771	14.109	9.713	1.391	9.899	0.947
Signing	3.741	4.463	13.300	21.348	14.694	7.228
Unblinding	0.436	0.496	0.573	0.623	0.9450	0.903
Verification	31.542	46.674	42.273	42.319	44.492	44.658
Total	**55.382**	**69.736**	**76.056**	**70.614**	**80.392**	**57.874**

Finally, Table 6 provided the comparison of ECC key size with addition-subtraction algorithms in proposed scheme. Among the algorithms, it has owned high complexity increased in 384 bit ECC. All time required for schemes have been measured on Core i7 CPU and processor speed 2.4GHz in Window 7 platform by java library.

Table 7.Comparison of Blind Signature Schemes



It is important to describe that the key length for considered schemes are selected to provide equal security levels. NIST recommended that 160 bit ECC key size has owned medium period security. The use of shorter keys leading to lower storage space and it can faster arithmetic calculation. Overall, comparisons of ECDLP based scheme with other schemes are analyzed in the Table 7. According to the result, the proposed ECDLP based approach outperforms that in [5] by 70% and that in [2] by 96% in terms of processing time. In conclusion, it is obviously known that the proposed ECDLP based blind signature with double and add method is more efficient than previous ECDLP based schemes.

7. Conclusion and Future Works

In this work, proposed ECDLP based blind signature scheme can provide efficient low computation overhead than previous scheme. The system also analyzed the ECC key size with four scalar multiplication methods. The proposed system gave the efficient time complexity and less memory space in every phase. To improve the performance, more efficient scalar multiplication methods can apply in e-voting system as future work and need to design more effective blind signature schemes which provably secure in the standard model.

9. References

[1] D.L. Chaum, “Blind Signature Systems,” US Patent 4759063, 1988.

[2] D. Jena, S. Kuma, and B. Majhi, “A Novel Untraceable Blind Signature Based on Elliptic Curve Discrete Logarithm Problem,” International Journal of Computer Science and Network Security, IJCSNS, vol.7, no.6. pp. 269-275, 2009.

[3] S.S. Kumar, “Elliptic Curve Cryptography For Constrained Devices,” 2006.

[4] M. Nikoogadam & A. Zakerolhosseini, “An Efficient Blind Signature Scheme Based On the Elliptic Curve Discrete Logarithm Problem,” The ISC Int’l Journal of Information Security, Volume 1, Number 2, July, 2009.

[5] A. Samal and A. Chhotaray, “A novel blind signature based upon ECDLP,” Thesis, Department of

Computer Science and Engineering, Orissia, Japan, 2012.
[6] G. Stoneburner, “Underlying Technical Models for Information Technology Security,” Recommendations of the National Institute of Standards and Technology, NIST Special Publication 800-33, 2001.
[7] E. Tanta, “Elliptic Curve Cryptography,” An Implementation Guide, In Anoop MS. India: 2007.
[8] A.A. Thu and K.T. Mya, “Implementation of an Efficient Blind Signature Scheme,” in the Proceeding of the 5th International Conference on Networking and Information Technology (ICNIT 2014), Singapore, November, 2014, pp. 443-448.
[9] A.A. Thu and K.T. Mya, “Comparisons of Scalar multiplication Methods By using Proposed Blind Signature Scheme,” in the Proceeding of 13th International Conference on Computer Applications (ICCA 2015), Yangon, Myanmar, February, 2015, pp. 200-206.
[10] S.A. Vanstone, “Elliptic Curve Cryptosystem- the answer to strong, fast public key cryptography for Securing Constrained Environments,” Information Security Technical Report, Vol.2, no.2, pp.78-87, 1997.