J Inf Process Syst, Vol.13, No.4, pp.970~986, August 2017
https://doi.org/10.3745/JIPS.03.0029

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

CLB-ECC: Certificateless Blind Signature Using ECC

Sanjeet Kumar Nayak*, Sujata Mohanty*, and Banshidhar Majhi*

Abstract

Certificateless public key cryptography (CL-PKC) is a new benchmark in modern cryptography. It not only simplifies the certificate management problem of PKC, but also avoids the key escrow problem of the identity based cryptosystem (ID-PKC). In this article, we propose a certificateless blind signature protocol which is based on elliptic curve cryptography (CLB-ECC). The scheme is suitable for the wireless communication environment because of smaller parameter size. The proposed scheme is proven to be secure against attacks by two different kinds of adversaries. CLB-ECC is efficient in terms of computation compared to the other existing conventional schemes. CLB-ECC can withstand forgery attack, key only attack, and known message attack. An e-cash framework, which is based on CLB-ECC, has also been proposed. As a result, the proposed CLB-ECC scheme seems to be more effective for applying to real life applications like e-shopping, e-voting, etc., in handheld devices.

Keywords

Authentication, Blind Signature, Certificateless Cryptography, Elliptic Curve Cryptography, E-cash

1. Introduction

Due to the huge demands of e-commerce, digital signatures play a vital role in providing authentication to the documents used in electronic media. Digital signatures are based on public key cryptography, which is used to provide authenticity, integrity, and non-repudiation to the messages that are sent over unsecured channels. In the case of the traditional public key cryptosystem, the identity of the user present in the literature is not considered in the process of generating the public key of the user. Normally, a particular user is associated with a public key that is issued and maintained by a trusted third party, known as the certificating authority (CA) [1]. The CA authenticates the public key of the user with its owner. This system suffers from disadvantage of storing and managing certificates for users. Furthermore, these cryptosystems need to execute a lot of computations and the public key of the sender has to be verified by the receiver before using it. To solve this PKC overhead, Shamir [2] introduced the concept of an identity based public key cryptosystem (ID-PKC), in which the public key of the user is derived directly from the identity of the user by a trusted third party, that is called the public key generator (PKG). In such a cryptosystem, the generation of public key for a user is done by

Manuscript received January 27, 2014; first revision May 13, 2014; accepted July 7, 2014; onlinefirst May 21, 2015.

Corresponding Author: Sanjeet Kumar Nayak (sanjeetnayak1@gmail.com)

* Dept. of Computer Science and Engineering, National Institute of Technology, Rourkela, Odisha, India (sanjeetnayak1@gmail.com, sujata.nitr@gmail.com, bmajhi@nitrkl.ac.in)

the PKG, whereas, in the case of a traditional public key cryptosystem, it is the responsibility of the user [3].

Even though ID-PKC reduces the storage and management of public keys, it is still not devoid of using certificates. One of the biggest shortcomings of this system is the key escrow problem (i.e., the private key of the user is known to the PKG). Thus, some malicious PKGs can forge the signature of some participants and decrypt the messages that are destined for users [4,5]. So, if the master key of the PKG is generated in a distributed manner, then the key escrow problem can be resolved. To manage the problem of ID-PKC, Alriyami and Paterson have suggested a certificateless public key cryptosystem (CL-PKC), which provides all the features of ID-PKC without the key escrow problem [4]. This cryptosystem does not require the certificates to guarantee the authenticity of the public key of a user. The private key of the user is generated in collaboration with both the user and The PKG. In order to forge a signature, one has to know both the identity provided by the user and the partial private key of the PKG. Moreover, the key does not need to be certified by any trusted authority [6].

Preserving the privacy of the user is important in some applications like online voting, and e-commerce. In 1982, David Chaum [7] proposed a blind signature scheme based on the RSA cryptosystem for providing anonymity to these applications. In a blind signature, a user obtains the signature of the signer in a message without revealing any information about the content of the message. This is achieved by blinding the message before sending it to the signer for a signature. The resulting signature in the blinded message behaves as if the signature is present in the original message, and it can be publicly verifiable. Subsequently, several blind signature protocols have been proposed based on the various computational hard problems of cryptography, such as the integer factorization problem (IFP) and the discrete logarithm problem (DLP). Due to the blind signature's important role in applications like online voting, electronic cash, and many others, numerous certificateless and ID based blind signature schemes are present in the literature. Some of these schemes have been proven to be secure against various attacks, while some are suitable for key management applications [8-11]. Generally, a blind signature scheme consists of the following four phases: blinding, signing, unblinding, and verification [12].

Furthermore, any blind signature scheme needs to satisfy properties like blindness, correctness, authenticity, unforgeability, non-repudiation, integrity, non-reusability, and untraceability [13,14].

a) Blindness: at the time of signing, the signer is unaware of the content of the message signed by him/her.
b) Correctness: without using the signer's public key the blind signature cannot be verified.
c) Authenticity: a valid signature signifies that the right person has generated the message.
d) Unforgeability: a valid signer can only generate a unique valid signature for a particular message.
e) Non-repudiation: after giving his/her signature in the message, the signer cannot deny having signed the message.
f) Integrity: the content of the message should not be modified throughout the transmission.
g) Non-reusability: the signature used in one message cannot be used for signing other messages.
h) Untraceability: after the message-signature pair has been published to the public, even the signer can't link between the messages with their signature pair.

Miller [15] and Koblitz [16] independently proposed an efficient public key cryptosystem, in which the group Z_p^* is replaced by the group of points on an elliptic curve defined over a finite field. The major advantage of an elliptic curve cryptosystem (ECC) over DLP is that the best-known algorithm for solving the underlying hard mathematical problem in ECC (i.e., the elliptic curve discrete logarithm problem [ECDLP]) takes an exponential amount of time. However, in case of DLP the best-known algorithm for solving the underlying mathematical problem takes a sub-exponential amount of time. Hence, significantly smaller parameters can be used in ECC than in other systems, such as DLP or RSA, but with an equivalent level of security [17]. It has been established that a 256-bit ECC key can achieve a similar level of security than can be provided by a 3,072-bit key in DLP. Hence, ECC based systems are considered more secure and computationally efficient [18]. Some benefits of having a smaller key size are faster computations, reduction in processing power, and reduction in storage space. This makes ECC ideal for a resource constrained environment such as cellular phones, smartcards, etc. The first blind signature based on ECDLP shows that it saves 34% of the space, as compared to a blind signature based on DLP [19,20].

A certificateless blind signature is the integration of the concept of the blind signature scheme and certificateless signature scheme. This means that a requester will get a certificateless signature without revealing the content of the message to the signer. Zhang and Zhang [21] first proposed the certificateless blind signature (CLB) scheme. They introduced the notion of a blind signature into certificateless public key cryptography. The scheme is based on pairing based cryptography. In 2009, Yang et al. [11] proposed a provably secure certificateless blind signature scheme. This scheme has proven to be secure against attacks by two different kinds of adversaries. Their scheme is efficient in terms of functionality, against some previously known ID-based blind signature schemes. They use very few pairing operations. As a result, their scheme requires less computational costs. Different certificateless blind signature schemes are available in [6,22-24]. It is observed from the literature that most of the recent certificateless blind signature schemes are based on a discrete logarithm problem and integer factorization problem. But, a certificateless signature protocol using ECC can reduce the computational cost, as well as the computation time. This is because ECC can provide the same level of security as compared to that of DLP with sufficiently less key size. Hence, in this article, we propose a certificateless blind signature scheme using ECC (CLB-ECC). CLB-ECC is shown to be resistant to various attacks such as forgery attack, key only attack, and known message attack. In addition to this, we have demonstrated an offline electronic cash (e-cash) framework based on the proposed CLB-ECC.

The rest of the article is organized as follows: in Section 2, an introduction to elliptic curve cryptography is described. The proposed certificateless blind signature scheme is presented in Section 3. Security analysis is discussed in Section 4. The offline electronic cash framework and its security analysis are given in Section 5 and Section 6, respectively. The concluding remarks are given in Section 7.

2. Background of Elliptic Curve Cryptography

This section briefly explains some preliminaries required in this paper, such as the elliptic curve and the computational hard problem in the elliptic curve, all of which help in understanding the proposed scheme.

Let $E(F_P)$ denote an elliptic curve E over a finite field F_P, which is given by the equation $y^2 = (x^3 + ax + b) \ mod\ p$ where $a, b \in F_P$, such that $4a^3 + 27b^2 = 0\ (mod\ p)$. This curve consists of points that satisfy the following equation with the extra point O, which is called the point at infinity and that forms an abelian group [17]. Mathematically, it can be defined as:

$$\{(x, y)|x, y \in F_P \ \&\ E(x, y)= 0\} \cup \{O\} \tag{1}$$

Let $P\ (x_1 , y_1)$ and $Q(x_2 , y_2)$ be two points on the elliptic curve and then the addition of these two points is represented by a third point (x_3, y_3) on the curve where:

$$x_3 = \lambda^2 - x_1 - x_2 \quad \text{and} \quad y_3 = \lambda(x_1 - x_3) - y_1 \tag{2}$$

where

$$\lambda = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1} & \text{if P} \neq \text{Q} \\ \dfrac{3x_1^2 + a}{2y_1^2} & \text{if P} = \text{Q} \end{cases}$$

If $P = Q$, then the point addition operation in the case of ECC is known as the point doubling operation. In ECC, scalar multiplication is realized as a repeated addition of points by the amount of scalar time s, over the finite field F_P. This is expressed as:

$$sP = P + P + \ldots + P\ (s\ times) \tag{3}$$

The ECDLP is defined as given $Q = kP$, where $P, Q \in E(F_P)$ and p is assumed to be of the order n, and to find an integer k, where $0 \leq k \leq n-1$.

Due to its smaller key size, ECC finds applications in smart cards and wireless communication, where the devices have less memory, bandwidth, and computational power.

3. Proposed Certificateless Blind Signature

The scheme consists of three parties, namely, PKG, signer, and requester. All of the three parties agree on an elliptic curve $E_P(a, b)$ of order p. The proposed scheme consists of seven phases, such as the *setup*, *partial private key extraction*, *secret value setup*, *private key setup*, *public key setup*, *certificateless blind signing*, and *certificateless verification*. In the protocol, convention followed to denote the elements in E_P as lower case and the points in E_P as uppercase letters. The symbols used in the scheme are given as:

G	:	a generator point whose order is large (i.e., $nG = O$)
n	:	number of points in E
O	:	point at infinity
x_A, k	:	randomly chosen number by the PKG
l, x_B	:	randomly chosen parameters by the signer
α, β	:	randomly chosen parameters by the requester
m	:	message
$h(.)$	:	a collision-free hash function

The operations on various phases of the proposed CLB-ECC scheme, which are polynomial time algorithms, are given below.

3.1 Setup

This phase consists of the following steps.

Step 1: The PKG chooses the generator point G in the elliptic curve $E_P(a, b)$ that is a large number.

Step 2: Then it selects $x_A \in Z_P^*$ randomly and computes the public key as:

$$Y_A = x_A G \tag{4}$$

3.2 Partial Private Key Extraction

For a signer with the identity ID_B, the PKG randomly chooses $k \in Z_P^*$ and computes the following equations:

$$R=kG \tag{5}$$

$$s=k+x_A[R]_x \tag{6}$$

$$P=sG \tag{7}$$

The PKG sends (s, R) to the signer with the identity ID_B. The signer (B) can check the authenticity of the received (s, R) by verifying the following condition:

$$sG=R+[R]_x Y_A \tag{8}$$

Then, the PKG publishes the system parameters as $< E, G, P, Y_A >$.

3.3 Secret Value Setup

The signer (B) with identity ID_B chooses $l \in Z_P^*$ as his/her secret information.

3.4 Private Key Setup

The signer (B) sets $x_B \in Z_P^*$ as his/her signing private key and computes the following equations:

$$U = lG \tag{9}$$

3.5 Public Key Setup

The signer (B) publishes Y_B as given by:

$$Y_B = x_B G \tag{10}$$

3.6 Certificateless Blind Signing

The signer (B) and the requester execute the following steps for creating a blind signature in message m.

Step 1: The signer computes:

$$t_1 = x_B l^{-1} \tag{11}$$

$$t_2 = sx_B^{-1} \tag{12}$$

and sends (U, t_1, t_2, s) to the requester.

Step 2: The requester chooses (α, β) $\in Z_P^*$ randomly and computes t' and t as:

$$t' = h(m \,||\, t_1U \,||\, t_2Y_B + U - \alpha G - s\beta G) \tag{13}$$

$$t = t' + \beta \tag{14}$$

It sends t to the signer (B).

Step 3: The signer computes the partial signature s' as:

$$s' = l - ts \tag{15}$$

and sends s' to the requester.

Step 4: The requester computes s'' as:

$$s'' = s' - \alpha \tag{16}$$

$\sigma = (s'', t')$ is the certificateless blind signature in m.

3.7 Certificateless Verification

To check the authenticity of signature σ in message m, the verifier performs the following steps:

Step 1: Computes t'' as:

$$t'' = h(m \,||\, Y_B \,||\, s''G + (1+t')P) \tag{17}$$

Step 2: Accepts the signature as a valid one if $t'' = t$. The proof of correctness is given below.

The subsequent steps in the proof given below, are formed using the Eqs. (7), (9), (10), (11), (12), (14), (15), and (16).

$$\begin{aligned}
& t' \\
& = h(m \,||\, t_1U \,||\, t_2Y_B + U - \alpha G - s\beta G) \\
& = h(m \,||\, x_B l^{-1} lG \,||\, sx_B^{-1} x_B G + lG - \alpha G - s\beta G) \\
& = \mathrm{h(m} \,||\, \mathrm{Y}_B \,||\, \mathrm{sG} + (\mathrm{s'} + \mathrm{ts})\mathrm{G} - \alpha\,\mathrm{G} - \mathrm{s}\,\beta\,\mathrm{G}) \\
& = \mathrm{h(m} \,||\, \mathrm{Y}_B \,||\, \mathrm{sG} - \alpha\,\mathrm{G} + \mathrm{s'G} + \mathrm{stG} - \mathrm{s}\,\beta\,\mathrm{G}) \\
& = \mathrm{h(m} \,||\, \mathrm{Y}_B \,||\, \mathrm{sG} - \alpha\,\mathrm{G} + \mathrm{s'G} + \mathrm{s(t'} + \beta)\,\mathrm{G} - \mathrm{s}\,\beta\,\mathrm{G}) \\
& = \mathrm{h(m} \,||\, \mathrm{Y}_B \,||\, \mathrm{sG} - \alpha\,\mathrm{G} + \mathrm{s'G} + \mathrm{st'G} + s\beta\,\mathrm{G} - \mathrm{s}\,\beta\,\mathrm{G}) \\
& = \mathrm{h(m} \,||\, \mathrm{Y}_B \,||\, \mathrm{s'G} - \alpha\,\mathrm{G} + \mathrm{sG} + \mathrm{st'G}) \\
& = \mathrm{h(m} \,||\, \mathrm{Y}_B \,||\, (\mathrm{s'} - \alpha)\,\mathrm{G} + \mathrm{sG} + \mathrm{st'G}) \\
& = \mathrm{h(m} \,||\, \mathrm{Y}_B \,||\, (\mathrm{s''G} + \mathrm{sG} + \mathrm{st'G}) \\
& = \mathrm{h(m} \,||\, \mathrm{Y}_B \,||\, (\mathrm{s''G} + \mathrm{s}(1 + \mathrm{t'})\,\mathrm{G}) \\
& = \mathrm{h(m} \,||\, \mathrm{Y}_B \,||\, (\mathrm{s''G} + (1 + \mathrm{t'})\,\mathrm{P}) \\
& = \mathrm{t''}
\end{aligned}$$

The overall flow of the proposed CLB-ECC is given in Fig. 1.

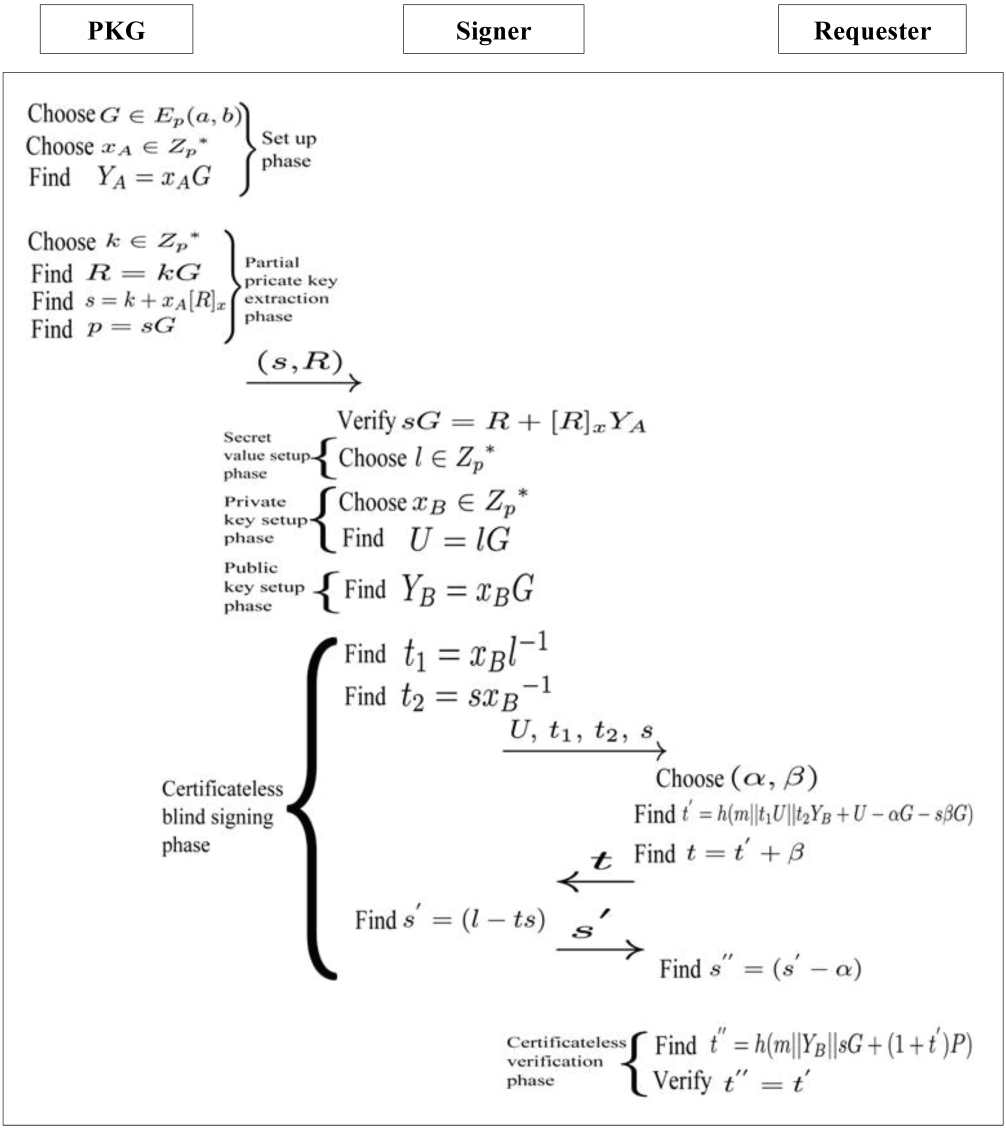

Fig. 1. Layout of the proposed certificateless blind signature scheme. PKG=public key generator.

4. Security Analysis

In this section, we have addressed the security issues of the proposed certificateless blind signature scheme in terms of two different types of adversaries in which one has no access to the master key (adversary-I) and the other has access to the master key (adversary-II) [21]. Adversary-II represents a malicious PKG that generates the partial private keys of users. Since there is no certificate that is used to provide the authenticity of the public keys, we must assume that an adversary can replace a user's public key that it knows the corresponding secret value for with a duplicate or false key. For adversary-I, it is permitted to extract a partial private key, replace public keys, and use the new values in a forgery [23]. For adversary-II, various adversarial activities, such as signature queries and secret value extraction requests, are permissible, but it is disallowed from replacing user's public keys [25]. We used two different games against both the forgers, adversary-I and adversary-II, as explained below. At the end of this section we have provided the efficiency analysis of the proposed scheme.

4.1 Game-I

This game is performed between a challenger and adversary-I for obtaining a certificateless blind signature. The game consists of the following three phases: *initialization*, *queries*, and *output*. Each phase is described below.

- Initialization: The challenger runs the setup algorithm and generates a master key and public system parameters. It keeps the master key secret and delivers the public system parameters to adversary-I. Here, adversary-I does not have any knowledge of the master key.
- Queries: Adversary-I may issue the following queries to the challenger.

1. Partial private key: When adversary-I makes a request for the partial private key of a user with identity ID_B, the challenger responds to the user's partial private key ID_B', which is an equal length of ID_B, by running the partial private key extraction algorithm.
2. Secret key: When adversary-I makes a request for the full private key of a user with identity ID_B, the challenger responds to the user's full private key x_B by running the secret key setup and the public key setup algorithms.
3. Request public key: When adversary-I makes a request for the public key of a user having identity ID_B, the challenger responds the user's public key Y_B by running the set public key algorithm.
4. Replace public key: Adversary-I can replace the original public key Y_B', which is chosen by him/her.
5. Blind signature (m, ID_B): When adversary-I makes a request for a blind signature or for a message m from a user with identity ID_B and public key Y_B, it may be noted that Y_B might be a replaced public key. To obtain the blind signature on message m, adversary-I engages the challenger with the certificateless blind signature algorithm. Finally, the challenger returns a blind signature σ' of message m under identity ID_B and public key Y_B.

- Output: Adversary-I delivers (ID_t, m_t, σ'), where ID_t is the identity of the target user, m_t is the message, and σ' is the blind signature for message m_t. Adversary-I wins the game if:

1. The partial private key (ID_t) and blind sign (m_t, ID_t) queries have never been queried.
2. The verify algorithm returns 1 (i.e., the signature σ' for a message m_t is valid under Y_B', which is replaced by adversary-I).

Assuming that the ECDLP problem is untraceable, our scheme is unforgeable against adversary-I. We have shown that if there exists an adversary-I to win the Game-I described above, then one can build an algorithm B that solves the ECDLP assumption.

Adversary-I issues queries as described in Game-I during its attack. The algorithm B lists L_i to keep track of the answer to the queries and a list L_0 to keep track of the identity, public key, and secret value tuples. We assume that adversary-I makes a query requesting identity. The following queries are handled as follows:

1. Query on partial private key: B recovers the user's public key Y_B from list L_0 and picks $x_B \in Z_p^*$ randomly. Then, B inserts the parameters (ID_B, Y_B) in list L_1. B picks k to compute $R = kG$. Then he/she chooses x_A at random to compute s. Then, he/she adds the tuple (s, R) into list L_0. B outputs 'failure' and stops since it is unable to answer the query. In our scheme, B returns false, as it will not pass verification Eq. (8). This is because the difficulty lies in solving ECDLP.
2. Secret key query: Suppose algorithm B picks a random l', as chosen by adversary-I, the private key of user is then set as $U' = l'G$. U' is stored in list L_0.
3. Public key replacement request: The adversary replaces the public key Y_B as Y_B' with U' stored in the list L_0. For message m and the corresponding replaced public key Y_B', the algorithm B returns a response of 'failure', as it will not pass verification Eq. (17).

4.2 Game-II

This game is performed between a challenger and adversary-II for the certificateless blind signature scheme as explained below. This game consists of the following three phases: *initialization*, *queries,* and *output*. Each phase is described below.

- Initialization: The challenger runs the setup algorithm and generates a master key and public system parameters. The challenger sends the master key and public system parameters to adversary-II.
- Queries: Adversary-II may issue the following queries to the challenger.
  1. Secret key: When adversary-II makes a request for the full private key of a user with identity ID_B, the challenger responds to the user's full private key x_B' by running the partial-private key extraction, secret value setup, and private key setup algorithms.
  2. Public key request: When adversary-II makes a request for the public key of a user with identity ID_B, the challenger responds to the user's public key Y_B by running the set public key algorithm.
  3. Blind signature (m, ID_B): When adversary-II makes a request for a blind signature in message m for a user with identity ID_B and the public key Y_B, the certificateless blind signature protocol runs between the challenger and adversary-II returns a blind signature σ' of message m under identity ID_B and public key Y_B.

- Output: Finally, adversary-II generates outputs (ID_t, m_t, σ'), where ID_t is the identity of the target user, m_t is a message, and σ' is a blind signature of message m_t. Adversary-II wins the game if:

1. The secret key and blind sign queries have never been queried.
2. The verification protocol outputs 1 (i.e., the signature σ' of message m_t is valid under Y_B, which

may be replaced by adversary-II).

In the section below, we show that our scheme is unforgeable against an attack by adversary-II.

1. The challenger runs the setup algorithm to generate the system's parameters and master key. Then he/she sends them to adversary-II.
2. Adversary-II handles various the queries of Game-II in the following manner:
   - Adversary-II produces message m_t and identity ID_t. The challenger sends a valid signature σ' on m_t for identity ID_t where the generation of σ' needs ID_t's secret values. In our scheme, the secret values of ID_t are secured under the difficulty of solving ECDLP.
   - Suppose that adversary-II replaces the public key of user (Y_B) by its own value and gives the replaced Y_B to the challenger. The challenger runs the certificateless blind signature algorithm with adversary-II and produces signature σ'. In our scheme, the signature tuple σ' produced by the challenger is not equal to the original signature σ. As for obtaining a valid signature, Eq. (17) must be satisfied. At the end of the game, adversary-II outputs a message-signature-identity tuple (m_t,σ',ID_t) that never matches with the original message-signature-identity tuple (m, σ, ID_B).

With respect to the security point of view, several theorems have been analyzed. We have shown that the proposed CLB-ECC satisfies the blindness property and that it can withstand forgery attack, key-only attack, and known message attack.

THEOREM 1. The proposed certificateless blind signature scheme satisfies the blindness property.

Proof. Given any valid blind signature $\sigma = (s'', t')$ of message m and any previously stored signature, (i.e., (s^*, t^*)), there always exists a unique pair of blinding factors, $\alpha, \beta \in Z_p^*$ that maps (s'', t') to (s^*, t^*), which leads to the same relations defined in the blind signature issuing protocol. As such, the signer can't establish a link between the valid blind signature and the corresponding previously stored blind signature. Thus, the proposed scheme satisfies the blindness property.

THEOREM 2. The proposed certificateless blind signature scheme can withstand a forgery attack.

Proof. In our proposed scheme, we have used SHA as the hash function. The property of the hash function says that from the message digest (or hash value) it is infeasible to extract the message. Given Y_A and G, it is infeasible to compute x_A from Eq. (4). The difficulty of solving this is based upon the elliptic curve discrete logarithm problem. For passing verification Eq. (17) successfully, an attacker has to randomly choose any two values from α, β, and t and compute the third one. If he/she chooses α, β randomly, then it is infeasible to find t since the hash function is non-invertible. Again, if he/she chooses t and either α or β, then it is also infeasible to find β or α. Given the valid signature (s'', t'), it is infeasible to derive another valid signature (s''^*, t'^*) in such a way that Eq. (17) is satisfied.

THEOREM 3. The proposed certificateless blind signature scheme can withstand a key-only attack.

Proof. A valid signature pair has to be formed by the attacker to make the key-only attack possible.

Suppose the attacker is able to create the signature pair. Then, the attacker will not be able to unblind the signature pair because he/she needs the parameter x_B and β. But extraction of these two parameters is impossible due to ECDLP.

THEOREM 4. The proposed certificateless blind signature scheme can withstand a known-message attack.

Proof. In the known-message attack, the attacker has access to one or more message-signature pairs and can generate a certificateless blind signature for his/her message. The proposed scheme can withstand the known message attack. Suppose the attacker has a message signature triplet (s'', t', m) and wants to generate the signature for message m^*. First of all, he/she has to generate certificateless blind signature t'^* on m^* and then generate the unblind signature s''^*. The signatures are sent to the verifier for verification. The verifier has the public key Y_A issued by the PKG to the verifier for message m and uses it for his/her own message m instead. But due to the ECDLP problem he/she isn't able to attack.

4.3 Efficiency Analysis

In this section, we compare our scheme with other certificateless blind signature protocols in terms of the computational time required for the signing and verifying phase and the result is shown in Table 1. We have used the following notations in the comparison process: P_m is the time requirement for scalar multiplication to take place on the elliptic curve group, P_{ex} is the time requirement for exponentiation operation on the group G, and P_e is the time requirement for performing a pairing operation on bilinear mapping [6,21]. The time requirement for carrying out a hash operation is negligible. Hence, we have omitted hash operation in the comparison table. We implemented our proposed scheme in a system configured with an Intel Core i3 processor and 4 GB RAM. We used PBC library [26] for performing all the operations listed above, for schemes of Zhang and Gao [6], Zhang and Zhang [21], and proposed CLB-ECC.

The comparison above shows that from amongst the three certificateless blind signature schemes our scheme requires less computational time. Our scheme requires approximately 21 milliseconds (ms) for executing the signing and verifying phase using a 160-bit modular elliptic curve group, which is the least amount out of all three. The major advantage of our scheme is that no pairing operation is used in the signing and verification process, which is the most time consuming operation.

Table 1. Comparison of computational time

Phase	Zhang and Gao [6]	Zhang and Zhang [21]	CLB-ECC
Signing	$P_e + P_{ex} + 3P_m$ (≈43 ms)	$P_e + P_{ex}$ (≈31 ms)	$5P_m$ (≈15 ms)
Verifying	$P_e + P_{ex} + P_m$ (≈34 ms)	$3P_e + 3P_{ex}$ (≈32 ms)	$2P_m$ (≈6 ms)
Total	$2P_e + 2P_{ex} + 4P_m$ (≈77 ms)	$4P_e + 2P_{ex}$ (≈63 ms)	$7P_m$ (≈21 ms)

CLB-ECC=certificateless blind signature scheme using elliptic curve cryptosystem.

5. Proposed E-Cash Framework

To validate our proposed scheme, we have proposed an offline electronic cash system that is based on the proposed CLB-ECC. In 1982, D. Chaum proposed the first electronic cash protocol based on a blind signature protocol. Since then, numerous electronic cash schemes have been proposed [27]. But as far as the use of a certificateless blind signature scheme using ECC is concerned no e-cash scheme has been proposed. We have used the e-cash framework given by Ashrafi et al. [28]. The proposed e-cash system consists of the following three participants: Customer, Merchant, and Bank. This system consists of the following five phases: *setup*, *initialization request*, *initialization response*, *payment request*, and *payment processing*. Details about each phase are given below.

5.1 Setup

Step 1: The bank chooses the elliptic curve $E_P(a, b)$ parameters and p that is a large number. After that, the bank chooses a point on the curve G and then, the bank chooses its secret key x_A from Z_p^* and computes its public key Y_A as:

$$Y_A = x_A G \tag{18}$$

Step 2: The merchant with identity ID_B chooses l as its secret information (Transaction ID) and sets its private key Y_B as:

$$Y_B = x_B G \tag{19}$$

5.2 Initialization Request

When a customer wants to buy goods/services from a merchant, he/she makes a request for the merchant and bank's public keys and system parameters of the merchant and the bank.

5.3 Initialization Response

Step 1: The bank chooses k at random and computes:

$$R = kG \tag{20}$$

$$s = k + x_A\,[R]_x \tag{21}$$

$$P = sG \tag{22}$$

Then, the bank sends (s, R) to the merchant.

Step 2: The bank sends as $< E, G, P, Y_A >$ to the customer.

Step 3: The merchant computes:

$$U = lG \tag{23}$$

where l is the transaction ID.

Step 4: The merchant computes t_1 and t_2 using his/her own secret parameter.

$$t_1 = x_B l^{-1} \tag{24}$$

$$t_2 = s x_B^{-1} \tag{25}$$

The merchant sends (U, t_1, t_2, s) to the customer.

5.4 Payment Request

Step 1: The customer constitutes the e-coin (m) by incorporating card details, validity period, and cost into m.

$$m = card\ detail \oplus validity\ period \oplus cost \tag{26}$$

Step 2: Then, the customer chooses (α, β) as his/her transaction password and computes:

$$t' = h(m \| t_1 U \| t_2 Y_B + U - \alpha G - s\beta G) \tag{27}$$

$$t = t' + \beta \tag{28}$$

and sends t to the merchant.

5.5 Payment Processing

Step 1: Upon receiving the payment response from the customer, the merchant contacts the bank by sending his/her identity ID_B .

Step 2: The merchant checks the authenticity of the received (s, R) by verifying the following condition:

$$sG = R + [R]_x Y_A \tag{29}$$

If the above condition is satisfied, the merchant computes s' as follows:

$$s' = l - ts \tag{30}$$

and sends s' to the customer.

Step 3: After receiving s'', the customer computes s' using his/her password (α) as follows:

$$s'' = s - \alpha \tag{31}$$

Here, (s'' , t') is the certificateless blind signature on the e-cash (m). Then, the customer sends the e-coin along with its signature (s'', t') to the bank.

Step 4: For an e-coin (m) and corresponding signature (s'', t'), the bank checks authenticity of the coin by computing:

$$t'' = h(m \| Y_B \| sG + (1+t')P) \tag{32}$$

The bank verifies whether $t'' = t$. If so, bank accepts the coin, otherwise it is rejected. Then, the bank deducts the required amount from the customer's account.

The overall block diagram of the proposed e-cash scheme is given in Fig. 2.

6. Security Analysis of the Proposed E-Cash Scheme

In this section, we analyze our protocol in different adversarial situations that may arise in e-cash transactions.

THEOREM 1. The proposed e-cash scheme can withstand a replay attack.

Proof. For instance, a dishonest merchant replays (re-transmit) the same payment information of an honest customer. When a customer sends his/her payment details to a merchant, a dishonest merchant will change its transaction ID l and replays it as l', but when the customer constitutes the e-coin using the replayed l, it will not pass verification Eq. (32).

THEOREM 2. A dishonest customer cannot double spend an e-coin.

Proof. In the proposed scheme, each time the merchant receives a refund request from a customer, he/she uses the authorization status to reveal the actual amount debited earlier. By providing the transaction ID (l) to the bank, it can reveal the actual amount of purchase from the e-coin (m). As each participant uses his/her own secret key to generate the signature, it is not possible for a customer to claim more money than the actual amount. If so, the identity of the double spender can be revealed by the bank.

7. Conclusion

The notion and security models of the proposed ECDLP based certificateless blind signature scheme are formalized. The proposed scheme is proven to be secure against type-I and type-II adversary attacks. The proposed scheme is computationally more efficient than traditional pairing based schemes as we used an elliptic curve cryptosystem with 160-bit point representation. Moreover, we also proposed an e-cash scheme that is based upon the proposed certificateless blind signature scheme, which ensures both participants' rights in the electronic transaction process. The proposed e-cash scheme is different from other existing schemes, as it does not require a strong trust relationship between the customer and merchant to exist. The proposed scheme meets the basic security requirements in electronic commerce and is simple and computationally efficient.

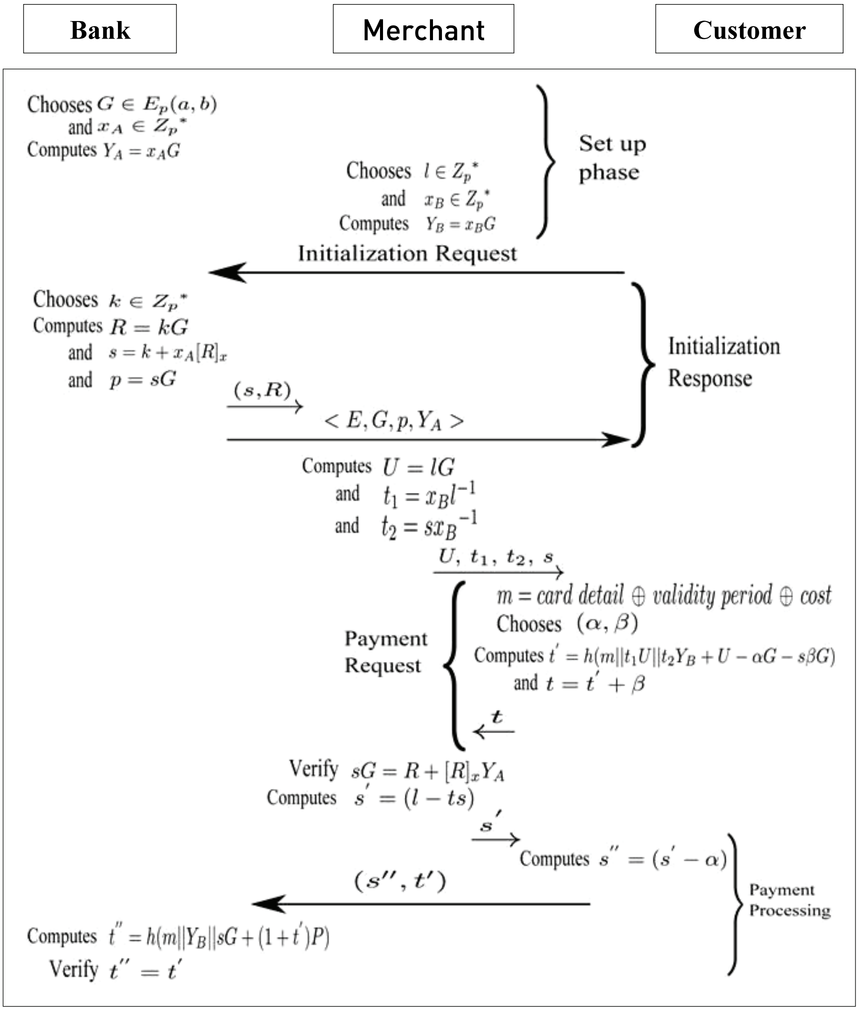

Fig. 2. Layout of the proposed e-cash scheme.

References

[1] Z. Zhang, D. S. Wong, J. Xu, and D. Feng, "Certificateless public-key signature: security model and efficient construction," in *Applied Cryptography and Network Security.* Heidelberg: Springer, 2006, pp. 293-308.

[2] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in Cryptology.* Heidelberg: Springer, 1985, pp. 47-53.

[3] D. H. Yum and P. J. Lee, "Generic construction of certificateless signature," in *Information Security and Privacy.* Heidelberg: Springer, 2004, pp. 200-211.

[4] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *Advances in Cryptology-ASIACRYPT 2003.* Heidelberg: Springer, 2003, pp. 452-473.

[5] J. K. Liu, M. H. Au, and W. Susilo, "Self-generated-certificate public key cryptography and certificateless signature/encryption scheme in the standard model," in *Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security (ASIACCS 2007),* Singapore, 2007, pp. 273-283.

[6] J. Zhang and S. Gao, "Efficient provable certificateless blind signature scheme," in *Proceedings of 2010 International Conference on Networking, Sensing and Control (ICNSC)*, Chicago, IL, 2010, pp. 292-297.

[7] D. Chaum, "Blind signatures for untraceable payments," in *Advances in Cryptology.* New York: Springer, 1983, pp. 199-203.

[8] F. Zhang and K. Kim, "Efficient ID-based blind signature and proxy signature from bilinear pairings," in *Proceedings of the 8th Australasian Conference (ACISP2003)*, Wollongong, Australia, 2003, pp. 312-323.

[9] R. Li, J. Yu, G. Li, and D. Li, "A new identity-based blind signature scheme with batch verifications," in *Proceedings of International Conference on Multimedia and Ubiquitous Engineering (MUE'07)*, Seoul, Korea, 2007, pp. 1051-1056.

[10] D. He, J. Chen, and R. Zhang, "An efficient identity-based blind signature scheme without bilinear pairings," *Computers & Electrical Engineering*, vol. 37, no. 4, pp. 444-450, 2011.

[11] X. Yang, Z. Liang, P. Wei, and J. Shen, "A provably secure certificateless blind signature scheme," in *Proceedings of 5th International Conference on Information Assurance and Security (IAS'09)*, Xi'an, China, 2009, pp. 643-646.

[12] C. I. Fan, W. K. Chen, and Y. S. Yeh, "Randomization enhanced Chaum's blind signature scheme," *Computer Communications*, vol. 23, no. 17, pp. 1677-1680, 2000.

[13] Z. Shao, "Improved user efficient blind signatures," *Electronics Letters*, vol. 36, no. 16, pp. 1372-1374, 2000.

[14] S. K. Nayak, B. Majhi, and S. Mohanty, "An ECDLP based untraceable blind signature scheme," in *Proceedings of 2013 International Conference on Circuits, Power and Computing Technologies (ICCPCT),* Nagercoil, India, 2013, pp. 829-834.

[15] V. Miller, "Use of elliptic curves in cryptography," in *Advances in Cryptology: CRYPTO'85 Proceedings.* Heidelberg: Springer, 1986, pp. 417-426.

[16] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203-209, 1987.

[17] J. Lopez and R. Dahab, "An overview of elliptic curve cryptography," Technical Report, 2000.

[18] 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.

[19] M. H. Chang, I. T. Chen, I. C. Wu, and Y. S. Yeh, "Schnorr blind signature based on elliptic curves," *Asian Journal of Information Technology*, vol. 2, no. 3, pp. 130-134, 2003.

[20] C. Popescu, "Blind signature schemes based on the elliptic curve discrete logarithm problem," *Studies in Informatics and Control*, vol. 19, no. 4, pp. 397-402, 2010.

[21] L. Zhang and F. Zhang, "Certificateless signature and blind signature," *Journal of Electronics (China)*, vol. 25, no. 5, pp. 629-635, 2008.

[22] S. Sun and Q. Wen, "Novel efficient certificateless blind signature schemes," in *Proceedings of International Symposium on Computer Network and Multimedia Technology (CNMT 2009),* Wuhan, China, 2009, pp. 1-5.

[23] A. W. Dent, "A survey of certificateless encryption schemes and security models," *International Journal of Information Security*, vol. 7, no. 5, pp. 349-377, 2008.

[24] S. Jose, A. Gautam, and C. Pandurangan, "A new certificateless blind signature scheme," *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, vol. 5, no. 1, pp. 122-141, 2014.

[25] Z. Wan, J. Weng, and J. Li, "Security mediated certificateless signatures without pairing," *Journal of Computers*, vol. 5, no. 12, pp. 1862-1869, 2010.

[26] B. Lynn, "The pairing-based cryptography library," 2006; http://crypto.stanford.edu/pbc.

[27] J. Wang, “Realization of non-track electronic cash,” *Procedia Engineering*, vol. 15, pp. 3265-3269, 2011.

[28] M. Z. Ashrafi and S. K. Ng, “Privacy-preserving e-payments using one-time payment details,” *Computer Standards & Interfaces*, vol. 31, no. 2, pp. 321-328, 2009.

Sanjeet Kumar Nayak http://orcid.org/0000-0003-4290-0632

He received his B.Tech. degree in Computer Science & Engineering from Biju Patnaik University of Technology, Odisha in 2010, and M.Tech. degree in Computer Science & Engineering from National Institute of Technology, Rourkela in 2013. His research interest includes cryptography, network security, and web security.

Sujata Mohanty http://orcid.org/0000-0002-3502-8201

She received her Ph.D. degree in Computer Science from National Institute of Technology, Rourkela, India in 2013. Presently, she is working as assistant professor in department of Computer Science at National Institute of Technology, Rourkela. Her recent research interest includes information security and cryptography.

Banshidhar Majhi http://orcid.org/0000-0002-2843-1908

He is presently working as a professor in the Department of Computer Science and Engineering, NIT Rourkela. He has 24 years of teaching and research experience and 3 years of industry experience. His research interests include image processing, computer vision, cryptographic protocols and iris biometrics. He has guided 10 doctoral works and published 45 articled in referred journals.