Hindawi
Security and Communication Networks
Volume 2017, Article ID 8981606, 14 pages
https://doi.org/10.1155/2017/8981606

*Research Article*

# An ECC-Based Blind Signcryption Scheme for Multiple Digital Documents

**Chien-Hua Tsai[1] and Pin-Chang Su[2]**

[1]*Department of Accounting Information, Chihlee University of Technology, New Taipei City, Taiwan*
[2]*Department of Information Management, National Defense University, Taipei, Taiwan*

Correspondence should be addressed to Chien-Hua Tsai; chienhua@mail.chihlee.edu.tw

Received 20 October 2016; Revised 12 January 2017; Accepted 26 January 2017; Published 22 February 2017

Academic Editor: Angelos Antonopoulos



The popularity of the Internet has comprehensively altered the traditional way of communication and interaction patterns, such as e-contract negotiations, e-payment services, or digital credential processes. In the field of e-form systems, a number of studies investigate the ability of the blind signature to fulfill the basic properties of blindness and untraceability. However, most literatures exploring the blind signature mechanisms only address research and technology pertaining to single blind signature issues. Further, most of the topics only deal with signing rather than encryption. Thus, we propose a new blind signature scheme for multiple digital documents based on elliptic curve cryptography (ECC). Our scheme incorporates the design of signcryption paradigm into the blind signature scheme to strengthen high levels of security. This innovative method also enhances computational efficiency during processing multiple electronic documents since the ECC provides a shorter key length and higher processing speed than other public-key cryptosystems on equivalent secrecy. The analysis results show that the present scheme achieves better performance at low communication overheads as well as with higher level of security. By helping the design of the intrinsic properties, the proposed cryptosystem can be applied to many areas to protect sensitive data in ubiquitous computing environments.

## 1. Introduction

Information and communications technology has strongly influenced the way of people's daily lives, particularly the channel we use. The Internet appears to be benefitting the development of data transmission or exchange in areas of delivering valuable information such as e-contract negotiations, e-voting or e-payment systems, and information market applications. As people's activities on the unfettered network and communication increase, digital information distribution applications become more available. This also means that by manipulating certain data in the query it could possibly lead to unauthorized access and usage of private information to the open Internet connection and potentially cause data leakage and access issues. For example, if an agreement's contents cannot be appropriately maintained secrecy in an electronic contract, some crucial information and sensitive business rules can easily be intercepted by unauthorized people, and the incidents involving the compromise of such data may result in a loss of individuals or business partners and revenue. To ensure that digital communication containing sensitive information remains relatively accessible and secure to the masses, the need of proper security measures should be applied for such data access. A number of techniques can be adopted when implementing privacy and unlinkability mechanisms within such an electronic document application, each with its own contributions to the protection of proprietary information, such as public-key cryptography [1], digital signatures [2, 3], and blind signatures [4–9].

Blind signature which was first described by Chaum [10, 11] has been extensively employed for protecting digital information privacy, and the mechanism makes sensitive data contents anonymous, resistant to forgery, and indisputable. Any blind signature scheme must satisfy two core properties, that is, blindness and untraceability [10, 12, 13]. Blindness property in an interactive signature protocol allows that the signed messages are transmitted between a user and a signer, and

the message contents are unknown from the signer. Untraceability or unlinkability property ensures that the signer cannot link back any message-signature pair later even if the signature is revealed to the public. Chaum's blind signature scheme is based on the integer factorization problem (IFP) and the security relies on the hardness of RSA assumption. This scheme can be considered secure if the underlying hash function is chosen appropriately. In order to enhance the security and efficiency of blind signatures, there have been several constructions of various schemes since the appearance of the blind signature [12–20]. Some works suggest that technical security requirements are based on the discrete logarithm problem (DLP) other than the IFP. There are also combination strategies [7, 9] which simultaneously involve both solving the DLP and tackling the IFP for attaining a high security level. As we probably know by now, the DLP hardness assumption is that, if a hash function is collision-resistant, then it is hard. In this setting, the associated security parameters must be chosen carefully so that the DLP remains hard in certain groups. It is interesting to note that either the DLP or the IFP over a prime field appears to be of roughly the same degree of difficulty [21, 22]. As the computing power increases and the algorithmic skills are constantly improved, there is also the chance that the DLP and IFP for the underlying combinatorial hard problem could be solved deterministically in subexponential time.

More recently, Vanstone [23] has proved that elliptic curve cryptosystems based on the elliptic curve discrete logarithm problem (ECDLP) provide greater efficiency than those cryptographic algorithms for the IFP and DLP. This fact makes solving the ECDLP in subexponential time impracticable. Due to the strength of smaller key size operations for the ECDLP, it is expected that a secure and efficient solution can be achieved under the algorithmic technique assumption. Subsequently, several variations of ECC-based blind signatures [4, 8, 24–27] are consequently proposed, and they have shown their schemes to be remarkably useful in practical applications between the security and the performance. It is worth pointing out that a newly unveiled blind signcryption concept (by combining blind signature and signcryption algorithm) is obtained from Shamsherullah et al. [28] and Sadat et al. [29]; their research is focused on customized designs on electronic payment systems and a proxy approach, respectively, offering strong security requirements to facilitate the progress of communicating and accessing information in complex networks.

From the above literature review, these existing cryptographic protocols are mainly interesting to construct stronger models for blind digital signatures to satisfy basic security guarantees, and their algorithms focus on disguising a message then followed by a digital signature and maintaining a verification along with a blinding factor to the resulting message. Although the strategy can guarantee the blindness and untraceability properties for the message, specific authorized subjects (e.g., project participants) are not assigned to verify this signature correctly since anyone can use signer's public key to verify the signature without identity authentication during the verification phase. In the meantime, different participants interact with each other in establishing communication sessions and the session data can leave an identifier stolen more vulnerable to identity theft or protocol attacks (e.g., the man-in-the-middle attack). Moreover, most of the current studies deal with blind signatures in a single message at a time or a batch of multiple signatures on multiple documents [2, 30] instead of managing large number of digital documents by making a single signature just once [31]. In the case of handling voluminous amount of documents, gradually performing blind signature processes on multiple electronic documents takes more time than going through the same steps in a single digital document. Another concern is that in batch processing the messages are signed to completion in consequence and the rest of these messages will not be affected by the tampering attempt if some of message contents have been compromised. This situation will raise security risks about information disclosure.

Unlike the approaches of one signature at a time or a batch mode, our proposal handles multiple digital documents by creating one-time signature that links all chunked messages to form the avalanche effect in cryptography to protect data from unauthorized access or alteration. In this paper, we also study methods to extend the functionality of member signatures while distinguishing the involvement of designated representatives from unauthorized persons, where the proposed scheme can enable a verifiable action for particular authorized people and authenticates the information correctly at a later time to verify the identities of given users to prevent identity breaches from the verification stage. This is particularly useful in an off-line scenario, where the signatures are able to be self-certified without needing an Internet connection once participants have been registered in the system as existing legitimate entities. In addition, we introduce the concept of ECC-based signcryption technique instead of using a more general class blind signature scheme, which along with its form can greatly minimize the computational load and communication overheads for tackling multiple electronic messages. The signcryption scheme which was first introduced by Zheng [32] is a new cryptographic paradigm that fulfills the integration of encryption and digital signature synchronously at a low-overhead functionality providing program. Research has proven its benefit in improving efficiency in several applications such as three-party communication environments [33], key management for wireless sensor networks [34], and multiple receivers for firewalls [35]. Yu and He [36] suggest a new efficient DLP-based blind signcryption protocol to enhance security goals such as anonymity, untraceability, and unlinkability. Ullah et al. [37] also present an ECC-based blind signcryption scheme that is capable of supplying the properties of confidentiality, integrity, unforgeability, and nonrepudiation for low-power or resource-constrained devices. On top of that, instead of using elliptic curve cryptography, Ch et al. [38, 39] and Nizamuddin et al. [40] introduce an alternative paradigm for signcryption measurements that are based on the notion of hyperelliptic curve cryptography, and their papers propose a more lightweight signcryption model having public verifiability and forward secrecy to reduce the number of bits and obtain better performance than the existing ECC-based schemes. However, all three of the methods do not use a

blinding technique but a nonblind cryptographic primitive to offer the support of public verifiability, and hyperelliptic curve cryptography in genus 2 that requires many more field operations in each group operation has the potential to be competitive with its genus 1 elliptic curve cryptography counterpart [41, 42]. As for multidocument cryptographic processing using signcryption technique, Tsai and Su [31] present a variant of a threshold signcryption protocol by assigning a group of signatures to share a secret link for multiple documents. Their work handles large number of digital documents via a group of participants splitting a secret and each of the members is allocated a share of the secret, whereas the proposed scheme manages multiple documents by one single person employing a blind signcryption technique along with these messages to enable effective protection measures, for example, the anonymity and untraceability properties. It is only natural to consider the signcryption technique from digital information perspective—thus, by combining a signcryption approach with a blinding procedure to carry out the blind digital signature protocol, this type of scheme not only essentially yields strong security requirements of a blind signature manner to detect dishonest adversaries, but also efficiently improves the computation and transmission costs of blind signature processing.

Our research contributions aim to improve adoption of the security requirements and to increase the speed of information transmission for multiple blind signcrypted messages. To achieve these objectives, we design a secure and efficient blind signcryption scheme based on elliptic curve cryptography that empowers the combination strategy to verify the authenticity of legitimate entities in the network without disclosing the contents of the signcrypted messages. The proposed scheme has the security attributes for multiple messages, namely, blindness, untraceability, authenticity, confidentiality, correctness, integrity, nonrepudiation, unforgeability, and the avalanche effect of encrypted messages. The comparative evaluation of the study has better performance in terms of computational cost and communication overheads. Additionally, this innovative method offers the useful property of a self-certified identity in off-line scenarios. It can be adapted to mobile computing environments for efficient and secure data transmission. The paper is organized as follows. In the next section, we briefly introduce the RSA-based blind signature form, ECC-based blind signature protocol, and signcryption manner, respectively. In Section 3, we propose an original essay to construct a signcryption-combined scheme for blind digital signatures. In Section 4, we evaluate the performance of the proposed solution and prove its security features. Finally, Section 5 concludes the paper.

## 2. Conceptual Basis

This section first gives a brief introduction to a RSA-based blind signature algorithm. We also sketch an ECC-based blind signature technique and the signcryption mechanism from their respective backgrounds, which will be recommended to our proposed scheme in Section 3.

*2.1. Blind Signature Based on RSA.* The concept of blind signature, first devised by Chaum [10] in 1983, is based on RSA algorithm and the hardness of IFP. According to Chaum's concepts, there are two participants, namely, the signer $S$ and the requester $R$, involved in the signature scheme. Given a message $m$ to be signed, let $(n, e)$ be the signer's public key and the corresponding private key is $d$. The blind signature scheme consists of the following five phases:

(i) *Initializing Phase.* $S$ chooses two distinct primes $p$, $q$ and computes $n = p \cdot q$, $\varphi(n) = (p-1) \cdot (q-1)$. Next, $S$ selects two random numbers $e$ and $d$ such that $1 < e < \varphi(n)$ and $\gcd(\varphi(n), e) = 1$, to determine $d$ as $d \equiv e^{-1} (\mathrm{mod}\, \varphi(n))$. $S$ then publishes $(n, e)$ and keeps $(p, q, d)$ secret.

(ii) *Blinding Phase.* $R$ takes an arbitrary number $r \in [0, n]$ and calculates $m' \equiv mr^e (\mathrm{mod}\, n)$. Then, $R$ sends $m'$ to $S$. In this phase, $R$ blinds the message and $S$ does not know the contents of the message.

(iii) *Signing Phase.* $S$ uses the private key $d$ to compute $s' \equiv (m')^d (\mathrm{mod}\, n)$ and sends it back to $R$.

(iv) *Unblinding Phase.* $R$ acquires the signature $s \equiv s' \cdot r^{-1} (\mathrm{mod}\, n)$.

(v) *Verifying Phase.* Anyone can verify the validity of message-signature pair $(m, s)$ by checking that $s^e \equiv m (\mathrm{mod}\, n)$.

*2.2. Blind Signature Based on ECC.* In 2010, Jeng et al. [4] proposed a fast blind signature scheme, based on the ECDLP. This scheme does not compute modular exponentiation consecutively. Instead, a user can obtain a signature and verify it only through scalar multiplication of points on elliptic curves, for example, point addition and point doubling. ECC requires much lesser numbers for its operations; hence the scheme is very efficient. Let an elliptic group $E_p(a, b)$ be formed as $y^2 = x^3 + ax + b (\mathrm{mod}\, p)$, where $4a^3 + 27b^2 \neq 0 \bmod p$ such that $E_p(a, b)$ is appropriate for cryptography. And then a base point $G$ on $E_p$ is determined whose order is a very large value $u$ such that $u \cdot G = O$. The protocol is described below.

(i) *Initialization.* $R$ randomly selects a secret key $n_i$ and generates the corresponding public key $P_i$ as $P_i \equiv n_i \cdot G (\mathrm{mod} p)$. Likewise, $S$ chooses a random number $n_j$ as the secret key, and the corresponding public key is $P_j \equiv n_j \cdot G (\mathrm{mod}\, p)$.

(ii) *Blinding.* $R$ retains a message $m$, sets $\alpha \equiv m \cdot (n_i \cdot P_i) \cdot (\mathrm{mod}\, p)$, and sends the blinded message $\alpha$ to $S$.

(iii) *Signing.* $S$ arbitrarily chooses another blinding factor $n_v$ and creates a pair of blind signatures $(r, s)$, where $r \equiv n_v \cdot \alpha (\mathrm{mod}\, p)$ and $s \equiv (n_v + n_s) \cdot \alpha \cdot (\mathrm{mod}\, p)$. Then $S$ forwards the message-signature pair $(\alpha, (r, s))$ to $R$ and keeps $(\alpha, n_v)$ in private.

(iv) *Unblinding.* $R$ removes the blind signature $(r, s)$ by applying the secret key $n_i$, along with $S$'s public key $P_j$ to yield $s' \equiv s - m \cdot n_i P_j (\mathrm{mod}\, p)$. And then $R$ calculates $m' = n_i \cdot (n_i - 1) m$.

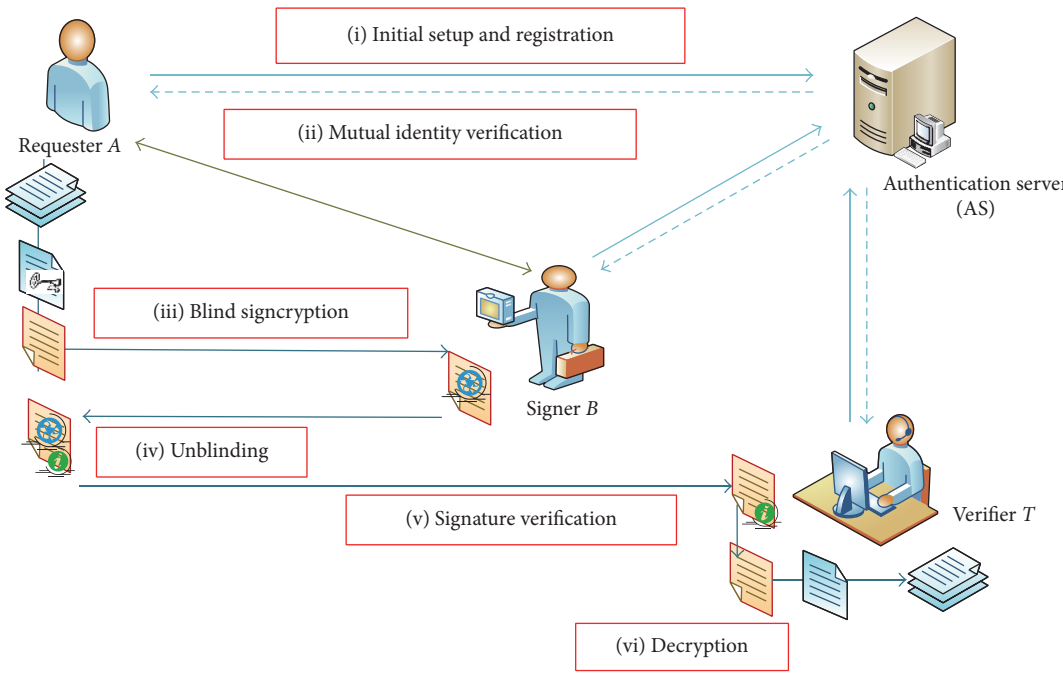


Figure 1: The proposed operational context diagram.

(v) *Verification.* Anyone can use $S$'s public key $P_j$ to verify the authentication of the signature $(m', s', r)$ by checking whether the given formula $r \equiv s' - m' \cdot P_j \pmod{p}$ has been satisfied.

*2.3. Signcryption Mechanism.* Signcryption, first presented by Zheng [32] in 1997, is a new cryptographic technique that fulfills digital signature and public-key encryption simultaneously in a single step at lower computational costs and communication overheads than signing and encrypting separately. Due to its advantages, both confidentiality and authenticity are seamlessly accomplished, and it is widely used for email transmission, files delivery, and data communication. A generic signcryption scheme $\Sigma = (\text{Gen}, \text{SC}, \text{USC})$ typically consists of the following three phases: key generation (Gen), signcryption (SC), and unsigncryption (USC). *Gen* generates a pair of keys for any user $U$: $(\text{SDK}_U, \text{VEK}_U) \leftarrow \text{Gen}(U, \lambda)$, where $\lambda$ is the security parameter, $\text{SDK}_U$ is the private signing/decryption key of user $U$, and $\text{VEK}_U$ is his/her public verification/encryption key. For any message $m \in M$, the signcrypted text $\sigma$ is obtained as $\sigma \leftarrow \text{SC}(m, \text{SDK}_S, \text{VEK}_R)$, where $S$ denotes the sender and $R$ is the receiver. SC is generally a probabilistic algorithm while USC is most likely to be deterministic where $m \cup \{\perp\} \leftarrow \text{USC}(\sigma, \text{SDK}_R, \text{VEK}_S)$ in which $\perp$ denotes the invalid result of unsigncryption.

Signcryption schemes can be trusted by providing two different mathematical functions as mentioned above; one is the signature and the other is the encryption. The choice of confidentiality and authenticity would be made based on the level of security desired by any digital signature scheme in conjunction with a public-key encryption scheme.

## 3. The Proposed Scheme

In this section, we introduce a secure and efficient blind signature scheme, which embeds the signcryption technique in the mutual authentication procedure for singular or multiple electronic message contents based on the ECDLP. Solving the ECDLP circumstance becomes computationally infeasible if any antagonist attempts to gather some secret information from captured participants to perform a specific action (e.g., counterfeit identity). In addition, our study uses interleaving structural features, that is, the ECC-based hard problem and the shift permutation problem, to raise the levels of security for the transmission of such information. Particularly, owing to the difficulty of solving the ECDLP and the small key lengths in ECC, the security strength and efficiency of the proposed solution will certainly lead to very promising results.

Our scheme comprises the following six phases: initial setup and registration phase, mutual identity verification phase, blind signcryption phase, unblinding phase, signature verification phase, and decryption phase. The operational context diagram of the proposed scheme is shown in Figure 1, and "Abbreviations" section summarizes the notations and the denotations thereof about the mechanism used. There are three participants in our blind signature protocol, namely, a requester $A$, a signer $B$, and a verifier $T$, respectively. Then, an authentication server AS is responsible for generating the system parameters and issuing secure electronic identities to users.

*3.1. Initial Setup and Registration Phase.* During the initial and registration stage, we first specify the domain's parameters to set up the system configuration. The default arguments that are made up of several key fields are as follows.

(i) A secure elliptic curve $E(F_q)$ is defined over a finite field $F_q$, where $q$ is a large prime number such that the number is greater than 283 bits; that is, a 283-bit key in ECC is considered to be as secured as 3072-bit key in RSA [43, 44]. Next, an order $d$ will be selected, together with the base point $G$ on the elliptic curve $E(F_q)$, and the proper choice satisfies $d \cdot G = O$, where $O$ is the point at infinity.

(ii) To generate a public-private key pair, the AS randomly chooses a secret value of $n_{\mathrm{AS}}$, from $[2, d-2]$ as the private key, and the associated public key can be derived from (1).

$$\mathrm{PK}_{\mathrm{AS}} = n_{\mathrm{AS}} \cdot G. \tag{1}$$

(iii) Then, the AS publishes $\mathrm{PK}_{\mathrm{AS}}$ to all users as well as the system parameters, ($E(F_q)$, $G$, and $d$), and keeps $n_{\mathrm{AS}}$ as a secret.

(iv) Each user, that is, $A$, $B$, and $T$, must register on the dedicated server (AS) as a legitimate participant before proceeding to related services.

(v) Next, all the users select random values $n_A$, $n_B$, $n_T$ as their private keys in the same way. Accordingly, the paired public keys of all users are generated with (2).

$$\begin{aligned} \mathrm{PK}_A &= n_A \cdot G, \\ \mathrm{PK}_B &= n_B \cdot G, \\ \mathrm{PK}_T &= n_T \cdot G. \end{aligned} \tag{2}$$

(vi) After creation, all participants have their own unique pair of keys. The message of private keys with identifies $\mathrm{id}_A$, $\mathrm{id}_B$, and $\mathrm{id}_T$ will be transmitted to the AS via a secure channel. In addition, the AS will apply the hash function, $h_1(\cdot)$, to produce a random nonsecret salt value, $e$, for verifying the identity of a user thereafter. The hash value can be used to determine the critical issue of identity assurance in an off-line status as a self-certification approach, and the associated hash values are obtained from (3).

$$\begin{aligned} e_A &= h_1\left(\mathrm{id}_A, \mathrm{PK}_A\right), \\ e_B &= h_1\left(\mathrm{id}_B, \mathrm{PK}_B\right), \\ e_T &= h_1\left(\mathrm{id}_T, \mathrm{PK}_T\right). \end{aligned} \tag{3}$$

(vii) In the meantime, the AS still needs the corresponding data points $Z_A$, $Z_B$, $Z_T$ on the elliptic curve to generate the relative certificates. Each data point containing a random numerical value, $l$, is calculated according to (4).

$$\begin{aligned} Z_A &= l_A \cdot G = \left(x_{Z_A}, y_{Z_A}\right), \\ Z_B &= l_B \cdot G = \left(x_{Z_B}, y_{Z_B}\right), \\ Z_T &= l_T \cdot G = \left(x_{Z_T}, y_{Z_T}\right). \end{aligned} \tag{4}$$

(viii) The certificates associated with each participant are therefore computed by (5).

$$\begin{aligned} \mathrm{ca}_A &= {l_A}^{-1}\left(e_A + x_{Z_A} \cdot n_{\mathrm{AS}}\right), \\ \mathrm{ca}_B &= {l_B}^{-1}\left(e_B + x_{Z_B} \cdot n_{\mathrm{AS}}\right), \\ \mathrm{ca}_T &= {l_T}^{-1}\left(e_T + x_{Z_T} \cdot n_{\mathrm{AS}}\right). \end{aligned} \tag{5}$$

(ix) When the setup process prepares all the appropriate parameters for the actions that were run, the AS securely sends the messages, ($e_{\mathrm{user}}$, $Z_{\mathrm{user}}$, and $\mathrm{ca}_{\mathrm{user}}$), to each user and also makes the global system parameters publicly known including $\mathrm{PK}_A$, $\mathrm{PK}_B$, $\mathrm{PK}_T$, $h_1(\cdot)$, and $h_2(\cdot)$.

*3.2. Mutual Identity Verification Phase.* When finishing the registration process, each entity is able to effectively communicate with the related parties. The user authentication agreement between the requester $A$ and the signer $B$ operates as below.

(i) In the request, the message ($\mathrm{ca}_A$, $e_A$, $Z_A$, $\mathrm{PK}_A$, $\mathrm{PK}_{\mathrm{AS}}$) is sent from $A$ to $B$, and vice versa (i.e., the message ($\mathrm{ca}_B$, $e_B$, $Z_B$, $\mathrm{PK}_B$, $\mathrm{PK}_{\mathrm{AS}}$) also reaches the targeted recipient from $B$ to $A$). According to the message from the requester $A$, the signer $B$ first checks whether the received message is original or not. If the message digest has not been altered, the signer $B$ goes on the identity verification process. Otherwise, the signer $B$ rejects the requester $A$'s authentication request. The authenticity of the received message must satisfy the constraint equation (6).

$$\begin{aligned} u_1 &= {\mathrm{ca}_A}^{-1} \bmod d, \\ u_2 &= e_A \cdot u_1 \bmod d, \\ u_3 &= x_{Z_A} \cdot u_1 \bmod d. \end{aligned} \tag{6}$$

(ii) If the message is genuine, the requester $A$ is a valid user and the signer $B$ continues the mutual verification context, or else the signer $B$ revokes the procedure. Next, the signer $B$ applies the public key from the AS to the message so as to authenticate the requester $A$'s identity. The discriminant validity is constructed as (7), and the authenticity of $A$ is verified by (8).

$$Q_A = u_2 \cdot G + u_3 \cdot \mathrm{PK}_{\mathrm{AS}} = \left(x_{Q_A}, y_{Q_A}\right), \tag{7}$$

$$x_{Z_A} \stackrel{?}{=} x_{Q_A}. \tag{8}$$

(iii) The signer $B$ compares $x_{Z_A}$ with $x_{Q_A}$. If $x_{Z_A} = x_{Q_A}$ which implies the identity verification is valid, the signer $B$ is then convinced that the requester $A$ is a legal entity. The requester $A$ can also verify the signer $B$'s identity, and it works in much the same way as the signer $B$ does. That is, the requester $A$ verifies whether $x_{Z_B}$ is identical to $x_{Q_B}$ or not.

*3.3. Blind Signcryption Phase.* The blind signcryption phase is a single continuous action rather than a three-stage process. In order to facilitate a more overt understanding of the context and later comparison with other existing methods between the operational baseline conditions, we logically divide the implementation into three substeps and this progress can be considered as the core part of the proposed scheme. Each one of these operations is closely aligned to an integration activity.

*3.3.1. Encryption Substep.* The purpose of the encryption stage is to avoid suffering the leak of sensitive information against the wishes of those who intend to snoop. We follow additional steps to increase operational security, and especially of that data is traveling across networks.

(i) To ensure the safe and secure delivery of digital information to the signer $B$ through the Internet, the requester $A$ first partitions a data message into a sequence, $\overline{v}$, of different plaintext blocks $v_i$ ($\geq 1$), and the separate blocks in each data segment can be expressed as (9).

$$\overline{v} = \{v_1, v_2, \ldots, v_i\}. \tag{9}$$

(ii) Secondly, the requester $A$ uses the $h_1(\cdot)$ hash function to produce a specific hash value $t$ known as a message digest for the sequence $\overline{v}$ of $v_i$, and the operation can be uniformly implemented by (10). At the same time, the one-way function $f_{m2p}(\cdot)$ that takes the sequence of data blocks as inputs is applied to transform the plaintext messages into a series, $\overline{V}$, of elliptic curve points $V_i$ ($\geq 1$). The data transformation can be done with (11).

$$h_1(\overline{v}) = t, \tag{10}$$

$$f_{m2p}(\overline{v}) = \{V_1, V_2, \ldots, V_i\} = \overline{V}. \tag{11}$$

(iii) Thirdly, in order to make the relationship between the plaintext messages and the representative points on the elliptic curve as complex as possible, the requester $A$ defines a set, $\overline{p}$, of binary sequences $p_i$ by (12), that is, the sequences whose terms are either 0 or 1. Also, each entry $p_i$ in the binary will match exactly the number of the aforementioned data points $V_i$.

$$\overline{p} = \{(p_1, p_2, \ldots, p_i) : p_i = 0, 1\}. \tag{12}$$

(iv) Fourthly, the requester $A$ generates a random number as a permutation value, and the given decimal integer $w$ which will be converted into its binary form and can be mapped onto $\overline{p}$ is organized by (13). The permutations which are controlled by the encoded binary sequence $w$ start with the most significant bit of $w_1$ first toward the least significant bit of $w_i$ end and do the following operations. When the current binary digit is 1 and the right side digit is 0, the corresponding data points are shifted to the right by one position. The operation shifts the place of relative point right by three bits if the two consecutive bits are equal to 1. In contrary, when the upper bit of the matching data is 0 and the lower bit is either 1 or 0, the left operations shift bits in transition, marching them to the left one bit or the left three bits, respectively. The sequence of left ($\ll$) or right ($\gg$) shifts corresponds to the function as (14).

$$w = \{f: w_1 \longrightarrow p_1, w_2 \longrightarrow p_2, \ldots, w_i \longrightarrow p_i : w_i = p_i = f(i)\}, \tag{13}$$

$$f(i) = \begin{cases} \gg 1 & \text{if } i\text{th bit is 1, } (i+1)\text{th is 0,} \\ \gg 3 & \text{if } i\text{th bit is 1, } (i+1)\text{th is 1,} \\ \ll 1 & \text{if } i\text{th bit is 0, } (i+1)\text{th is 1,} \\ \ll 3 & \text{if } i\text{th bit is 0, } (i+1)\text{th is 0,} \end{cases} \quad \text{such that } f(i) \in \overline{p}. \tag{14}$$

(v) After that, the requester $A$ needs the essential arguments including the arbitrary integer $w$, the hash value $t$, a randomly chosen number $k$, and a public key $\mathrm{PK}_T$ from the verifier $T$, to systematically transform the foregoing plaintext messages to corresponding ciphertext points. Equations (15) through (18) summarize the encryption operations. There is a specified point $K$ which is calculated from the product of $k$ and the base point $G$, and it serves to detect that the received ciphertext has not been tampered with while in transit. In such a way, each ciphertext block $C_0, C_1, C_2, \ldots, C_i$ ($\geq 1$) is combined with the previous ciphertext block before being computed. Note that the starting point $C_0$ included in the ciphertext data segments contains two secret parameters $w$ and $t$ representing a permutation value and an integrity check value, respectively, and the two significant factors will exhibit the avalanche effect, which causes a drastic variation in the ciphertext if either the plaintext, for example, $V_i$, or the value of characteristics, for example, $k$, $p_i$, $\mathrm{PK}_T$, is changed slightly.

$$K = k \cdot G, \tag{15}$$

$$C_0 = \left[f_{m2p}(w, t) + k \cdot \mathrm{PK}_T\right], \tag{16}$$

$$C_i = \left[V_i + p_i \cdot C_{i-1}\right], \quad i \geq 1, \tag{17}$$

$$\overline{C} = \{C_0, C_1, C_2, \ldots, C_i\}. \tag{18}$$

(vi) Lastly, the requester $A$ applies a publicly known hash function $h_2(\cdot)$ as (19) to the encrypted message $\overline{C}$ to create a unique message digest $m$ after obtaining the sequence of ciphertext blocks $C_i$.

$$h_2\left(\overline{C}\right) = m. \tag{19}$$

*3.3.2. Blinding Substep.* The core goal of blindness is to protect the messages from the signer without knowing its contents. For the blindness property, the requester $A$ uses the public and private key pair as a blinding factor $(n_A \cdot \mathrm{PK}_A)$ with the message digest $m$ to blind the message, and the blinding operation is computed by (20). Then the blinded message $\alpha$ is passed to the signer $B$.

$$\alpha = m \cdot n_A \cdot \mathrm{PK}_A. \tag{20}$$

*3.3.3. Signing Substep.* Upon receipt of the resulting message $\alpha$, the signer $B$ haphazardly selects an integer $\beta \in [2, d-2]$ to determine a secret element $R$ as (21) and combines the private key $n_B$ with $\beta$ to obtain the blind signature $S$ using (22). The message-signature pair $(\alpha, (R, S))$ is then forwarded back to the requester $A$. Since $\beta$ is a random number and a pair consisting of a secret value and a signature $(R, S)$ is arbitrary too, this implies that each individual construction yields a completely different signature and it is not possible to forge any valid signature on messages.

$$R = \beta \cdot \alpha, \tag{21}$$

$$S = (n_B + \beta) \cdot \alpha. \tag{22}$$

*3.4. Unblinding Phase.* To unblind the received signature of the message-signature pair, the requester $A$ first takes the blind signature $S$, the previously generated message digest $m$, the private key $n_A$, and the public key $\mathrm{PK}_B$ of the signer to extract the blinded signature $S'$ as expressed by (23). Also, the requester $A$ computes the nonce message digest value $m'$ and the unblind operation is governed by (24). Then both $S'$ and $m'$ along with the triple $(R, \overline{C}, K)$ are sent to the verifier $T$ to testify that its blinded allegation-signature-request message is authentic.

$$S' = S - m \cdot n_A \cdot \mathrm{PK}_B, \tag{23}$$

$$m' = n_A \cdot (n_A - 1) \cdot m + m. \tag{24}$$

*3.5. Signature Verification Phase.* After receiving the message-signature tuple $(S', m', R, \overline{C}, K)$, the verifier $T$ uses the signer's public key $\mathrm{PK}_B$ to verify the authentication of the alleged signature and the passing message digest by checking whether (25) holds. If the resulting message-signature pair $(S', m')$ is accepted as valid, the verifier $T$ then can proceed to decrypt the sequence $\overline{C}$ of ciphertext blocks.

$$R - h_2\left(\overline{C}\right) \cdot \mathrm{PK}_B \stackrel{?}{=} S' - m' \cdot \mathrm{PK}_B. \tag{25}$$

*3.6. Decryption Phase.* Decryption is the reverse process, converting the ciphertext message back into its original form. In this case, the encrypted messages contain the transformed data points $V_i$ and the related sequence entries $p_i$ thereof and the random generated permutation value $w$ along with the message digest $t$. Besides, the number of data segments is repeatedly carried over from previous data blocks. Thus the verifier $T$ needs these things to get the original messages back.

(i) First, the conversion function $f_{m2p}(\cdot)$ having the random permutation value and the hashed message pair $(w, t)$ can be explicitly specified by assigning the verifier's private key, $n_T$, the verification point $K$, and the initialization block $C_0$ arguments. If (26) can properly express the causal relationship implied by this assignment process, this means that the measurement corresponds accurately to its corresponding latent variables.

$$f_{m2p}(w, t) = C_0 - n_T \cdot K. \tag{26}$$

(ii) Next, the verifier $T$ uses another conversion function $f_{p2m}(\cdot)$ which maps an elliptic curve point to a message block, to acquire the specific pair $(w, t)$. By taking the input arguments, the return operation from (27) yields its untransformed information.

$$(w, t) = f_{p2m}\left[f_{m2p}(w, t)\right]. \tag{27}$$

(iii) Once both the permutation value $w$ and the correct message digest $t$ are collected, this makes the obtained references suitable for decryption of messages. The verifier $T$ applies the permutation sequence $w$ (from (13)) in binary format to the associated message sequence $\overline{p}$ previously defined in (12) and then performs bit shifting operations to find the number of matching permutation values in corresponding bit positions in the two binary sequences. The bit-reverse operation is similar to the forward bit shifting trick (from (14)), but it is intended for operating in the opposite direction on individual bits. Equation (28) indicates that it uses the relevant rules regarding reversals for bit patterns to locate the bit offset in an ordered sequence of bits. While the underlying permutations with respect to the sequence of message blocks are interpreted, the ciphertext blocks can be easily deciphered back into the plaintext messages.

$$f(i) = \begin{cases} \ll 1 & \text{if } i\text{th bit is 1, } (i+1)\text{th is 0,} \\ \ll 3 & \text{if } i\text{th bit is 1, } (i+1)\text{th is 1,} \\ \gg 1 & \text{if } i\text{th bit is 0, } (i+1)\text{th is 1,} \\ \gg 3 & \text{if } i\text{th bit is 0, } (i+1)\text{th is 0,} \end{cases} \quad \text{such that } f(i) \in \overline{p}. \tag{28}$$

(iv) After that, the process of reverting the ciphertext units $C_i$ to the plaintext segments of data points $V_i$ is progressively carried out by (29). And all the corresponding plaintext data sets can be recovered from the relevant ciphertext blocks as an expression of the sequence form $\overline{V} = \{V_1, V_2, \ldots, V_i\}$.

$$V_i = \left[C_i - x_i \cdot C_{i-1}\right], \quad i \geq 1. \tag{29}$$

(v) Finally, the verifier $T$ reuses the conversion function $f_{p2m}(\cdot)$ to convert the data points into the numeric

values, as expressed in (30), and all the separated elements in the sequence $\overline{v}$ are then concatenated to form one continuous text message as the original plaintext.

$$f_{p2m}\left(\overline{V}\right) = \overline{v}. \tag{30}$$

## 4. Security Analysis and Performance Evaluation

In this section, we will first describe the security analysis of the proposed scheme and then show that our solution can reach greater efficiency with respect to the performance assessments.

*4.1. Security Analysis.* The security of our scheme is based upon the difficulty of solving the ECDLP. In the meanwhile, the signature approach has applied the signcryption technique within the functionality of blind signature, which thereby strengthens the overall security of electronic communications. Apart from providing the crux properties of blindness and untraceability, some additional characteristics like authenticity, confidentiality, correctness, integrity, nonrepudiation, and unforgeability as formalized requirements from previous works [5, 6, 16, 18–20] are incorporated in the proposed scheme to make it stronger as well as more useful for various applications. We examine these security requirements of our scheme as follows.

*4.1.1. Blindness.* Blindness means that the signer cannot view the content of the message while he/she signs the message. The blinded message of our scheme is generated as $\alpha = m \cdot n_A \cdot \mathrm{PK}_A$ in (20). The signer $B$ or an opponent is unable to derive the message $\alpha$ without the parameters, namely, the message digest $m$ and the blinding factor $(n_A \cdot \mathrm{PK}_A)$. Since finding the blinding factor in this equation leads to encounter calculating the number of points on the elliptic curve over fields, it becomes extremely difficult to break the value of knowing desired points when tackling the ECDLP. The other parameter value $m$ is not an easy attempt that reverses a hash function. Therefore, the present approach is able to fulfill the blindness property because the signer $B$ signs the blinded message and knows nothing about the content of the message.

*4.1.2. Untraceability.* Untraceability is also an essential security requirement in any blind signature scheme. The signer is unable to link the signature with the message when the message-signature pair has been revealed to the public. In this experiment, the message-signature pair $(\alpha, (R, S))$ is produced from (20), (21), and (22). The signer $B$ only has the information about his or her own private key $n_B$ and a random number $\beta$, for each blind signature requested. Without the knowledge of the secret factors, a unique message digest $m$ and $A$'s private key $n_A$, from the requester $A$, the signer $B$, or the verifier $T$ cannot trace the association between the message and the blind signature. Hence, this scheme can achieve the untraceability or unlinkability property of a blind signature.

*4.1.3. Authenticity.* Authenticity is the property that has two purposes. One ensures that a message received is the exact same message which was sent, and the other verifies that all communication participants are who they really claim to be. With regard to message authentication, the current scheme can provably provide the authenticity ability of electronic documents or data while maintaining the privacy of the signature, and these messages are able to be adequately protected from inappropriate or malicious modifications through a valid corresponding checksum at the verifier side as described in (25). As for identity verification, the identities of all parties can be reliably verified during an interactive communication model using the identity authentication $x_{Z_A} \stackrel{?}{=} x_{Q_A}$ of (8). If a third party impersonates a legitimate user to gain unauthorized access to the message data, it is computationally impractical for solving the ECDLP in elliptic curves (e.g., to obtain $n_{\mathrm{AS}}$ from $\mathrm{PK}_{\mathrm{AS}}$). Surely the proposed model renders the property of authenticity.

*4.1.4. Confidentiality.* Confidentiality specifies that the contents of the message are required to be kept confidential from unauthorized persons, entities, or processes. In this study, all messages first are encrypted and disguised (blinded) by the requester $A$, signcrypted by the signer $B$, and then passed through a permutation process before conveying them to the verifier $T$. If there is an opponent that succeeds in intercepting the messages during transmission, the opponent should be unable to decrypt the transmitted ciphertext in a very strong form of cascaded encryption technique. The message-related attributes, especially a set of messages of different types, cannot easily be derived without reference values for cryptanalysis works. For example, the value of $K$, a verification point, as shown in (15), which depends parametrically on $k$ (a random number) and $G$ (a base point), can be difficult to find by other means. The attacker has to encounter calculating the number of points on the elliptic curve over fields, and it becomes extremely hard to break the value of knowing desired points when tackling the ECDLP. Accordingly, the present method can secure the contents of the message to reach the property of confidentiality.

*4.1.5. Correctness.* Correctness indicates that everyone with the signer's public key can check the correctness of a signature. As we mentioned in Section 1, the signature of the signer is revealed to public leading to an identity leak issue. The public delegate as a verifier will learn the identity of the signer on each session from a unique electronic binding between an identity and a public key via a digital certificate. As a result, the public verifying may put various confidential messages at risk. In our design, the correctness of the signature of a message signed through the signature verification procedure can be checked by the verifier $T$ as a major role using $B$'s public key via an authentication form. To verify the correctness of the signature from the signer $B$, the verifier $T$ has to check whether (25) is valid. If the equation holds, then $(S', m')$ is accepted as a valid signature of the message. During the course of the verification, the verifier $T$ can successfully achieve the identity authentication from the signer $B$ through the

Table 1: Comparison of the proposed scheme and the two existing similar methods.

| Security goals | Algorithm | | |
|---|---|---|---|
| | A new efficient blind signcryption (Yu and He, 2008) [36] | Blind signcryption scheme based on elliptic curves (Ullah et al., 2014) [37] | Our scheme |
| Blindness | × | √ | √ |
| Untraceability | × | √ | √ |
| Authenticity | × | √ | √ |
| Confidentiality | √ | √ | √ |
| Correctness | × | × | √ |
| Integrity | √ | √ | √ |
| Nonrepudiation | √ | √ | √ |
| Unforgeability | √ | √ | √ |

secret value $n_B$ which is $B$'s private key and embedded into (22). Consequently, the proposed design conforms to the correctness property.

*4.1.6. Integrity.* Integrity denotes that the information cannot be altered during the transmission, neither accidentally nor maliciously. If an antagonist attempts to alter a certain piece of data, for example, portions of ciphertext $C_i$, being communicated between the sender and the recipient, it is not easy to tamper with the message segments. Such tampering requires at least two or more secret parameters like a permutation value $w$ and an integrity check value $t$ in (16), and they are barely obtained from a conversion function of elliptic curve points that maps the messages to the curve. Furthermore, each portion of the ciphertext that is given the corresponding coordinate position and is embedded in the encoded text as given in (17) is quite dependent on all message blocks. Once there is an intentional act to make any change to a particular message, it should result in dramatically different consequences with respect to the avalanche effect. Thus, the proposed solution provides the integrity property.

*4.1.7. Nonrepudiation.* Nonrepudiation denotes that the signer cannot deny having signed a message that has a valid signature. In our case, the blinded message $\alpha$ has been electronically signed by the signer $B$ that purported to sign the document, and the signature containing specific values usually accompanies the document to send back the requester $A$. $B$ cannot repudiate having signed $\alpha$ since the signature was created with $B$'s private key $n_B$ and a randomly selected number $\beta$. In addition, through the signature validation process as represented by (25), the verifier $T$ can later confirm that the signature of the message has been entitled by the designated signer $B$ because $T$ has to use the corresponding public key as $B$'s $\mathrm{PK}_B$ during the verification. So, the proposed method offers the nonrepudiation property.

*4.1.8. Unforgeability.* Unforgeability refers that only the signer can give a valid signature for the associated message, and he/she should not be able to generate more signatures than the number of valid signing executions (a.k.a. nonreusability) in an interactive signature agreement. If an adversary impersonates the signer $B$ to forge a legally blind signature, he/she can intercept or eavesdrop the blinded message $\alpha$ but is unable to obtain a valid pair $(\alpha, (R, S))$ to execute the signature generation process without a designated signer, $B$, holding private key $n_B$. Similarly, if the signer $B$ attempts to willfully create two more valid signatures after interacting with the requester $A$ once, it is practically impossible for $B$ to guess a random signature $(R, S)$. Besides, the verifier $T$ can use the signature verification procedure $R - h_2(\overline{C}) \cdot \mathrm{PK}_B \stackrel{?}{=} S' - m' \cdot \mathrm{PK}_B$ as defined in (25), to determine a received message tuple, $(S', m', R, \overline{C}, K)$, corresponding to that signature against the forgery. For these parameters, the adversary or the dishonest signer then has to encounter the hardness of solving the ECDLP and the difficulty of inverting the one-way hash function. The proposed scheme indeed satisfies the property of unforgeability.

We have described the multifaceted characteristics of the proposed scheme in terms of security requirements; it has been pointed out that distinguishing attributes do fit well within blind signatures. In Table 1, we present a comparison of the above-mentioned two latest schemes in Section 1, based on security properties for blind signcryption techniques. The symbol "√" on a security requirement means that it is satisfied with the feature, while the symbol "×" indicates that it does not provide satisfaction in a specified manner. As seen from Table 1, due to the eight essential properties, the present method offers enhanced security functions in related applications of blind signcryption whereas the existing successful schemes suffer from some weaknesses including blindness, untraceability, and correctness.

*4.2. Performance Evaluation.* The subsection following the next investigates a detailed quantitative measure comparing the performance of our proposed algorithm with the two aforesaid algorithms in blind signcryption systems. We will examine theoretical results of the three different strategies for solving the cryptological operations involved with respect to the costs of computation and communication incurred by each task according to the concept of modular arithmetic operations [31, 45]. The notations including scalar multiplication, point addition, hash construction, and modular

Table 2: The computational complexity symbols and the meanings.

| Symbol | Description | Operation cost |
|---|---|---|
| $T_{\text{MUL}}$ | The execution time of a multiplication operation | $= 1T_{\text{MUL}}$ |
| $T_{\text{ADD}}$ | The execution time of an addition operation | Negligible |
| $T_{\text{EXP}}$ | The execution time of an exponentiation operation | $\approx 240T_{\text{MUL}}$ |
| $T_{\text{INVS}}$ | The execution time of a modular multiplicative inverse | $\approx 240T_{\text{MUL}}$ |
| $T_{\text{ECMUL}}$ | The execution time of an ECC point multiplication | $\approx 29T_{\text{MUL}}$ |
| $T_{\text{ECADD}}$ | The execution time of an ECC point addition | $\approx 5T_{\text{MUL}}$ |
| $T_h$ | The execution time of an ECC point hash operation | $\approx 23T_{\text{MUL}}$ |
| $t_h$ | The execution time of a basic hash function operation | $\approx 0.4T_{\text{MUL}}$ |

arithmetic that we used to evaluate the performance are shown in Table 2.

Table 3 summarizes the comparison results between our scheme and the existing similar blind signcryption schemes in terms of computational costs. Compared to the three related algorithms by evaluating one single electronic document processing, the proposed scheme requires two public-key encryption and decryption operations for each task, which lead to a performance penalty. This is more time-consuming work regarding the computational complexity of dealing with both the ECDLP computation and the permutation procedure simultaneously. As we can see, if we compare the outcomes with the same baseline measures as shadow areas in Table 3, the proposed scheme has much lower computational complexity, even with encryption and decryption latency-time tradeoffs, than the other two blind signcryption approaches. In spite of imposing more sophisticated manipulation techniques, this nature makes the proposed solution bear strongly secure structure and effectively prevent unwanted network intrusions.

As the number of electronic documents is gradually increased, maintaining the efficiency and security of blind signcryption protocols becomes critical to the continuity of the related operations. To estimate different performance levels for these blind signcryption schemes in the context of multiple documents (e.g., a multipage document), we repeatedly conduct the required steps to complete each blind signcryption process. Table 4 yields the performance comparison for the proposed signcryption-combined blind signature scheme against the two exemplary blind signcryption protocols in terms of number of documents. As shown in Table 4, Yu et al.'s DLP-based method causes the substantial increase in computational cost on each associative multiplication operation. Although our scheme reaches a slightly higher computational complexity for dealing with one single digital document about $121T_{\text{MUL}}$ in the total cost than Ullah et al.'s approach due to the mutual authentication operation (i.e., $2T_{\text{ECMUL}} + 1T_{\text{ECADD}} + 2T_{\text{MUL}} + 1T_{\text{INVS}} \approx 305T_{\text{MUL}}$), the computational costs of the two existing methods potentially take more time to execute cryptographic-related operations with a dramatic increase in managing vast numbers of documents from 2 to 10. The performance penalty associated with the relative inefficiency of these blind signcryption based algorithms is closely correlated if every single digital document has to go through all of the time-consuming steps involved. Unlike the classic approaches that handle a single electronic document each task, our solution consumes lower costs to perform the security-related operations for processing relatively large amounts of digital documents and always runs in weakly polynomial time. Put another way, the proposed scheme requires only one-time operation to blind signcryption, unblinding, signature verification, and decrypt processes for multiple document messages whereas the existing mechanisms need to keep reiterating the procedure several times to manipulate large quantities of data in a paginated form for blinding, signing, unblinding, and signature verification actions. Through the contiguously tabular analysis, we believe that our proposed signcryption-embedded approach significantly outperforms the other existing methods in carrying out several levels of cryptographic operations on large numbers of documents. This much efficient cryptosystem is good to use in various kinds of blind signature applications.

## 5. Conclusions

This paper presents a new alternative scheme of blind signatures for electronic messages and documents processing based on both the ECDLP and the bit-level permutation problem difficulties. To make the relationship between the content of the messages and the message-signature pair thereof as perplexed as possible, we embed the signcryption technique into the functions of blind signature besides the cryptographic primitives and explore the constructive solution to tackle the tricky challenges such as identity, privacy, anonymity, and security.

We have seen how the concept of aggregate signcryption like blind signature and encryption can be used to build a signcryption-combined blind signature scheme and also indicated that the proposed scheme is capable of being more beneficial and requires less number of multiplication operations compared to the two existing solutions in physically secure and efficient implementations for digital information protection. At the security analysis, the work investigates the related security requirements from a blind signature design methodology and these strong security properties are fully satisfied with the relevant parameters. In addition, the study evaluates the performance effects of different levels in carrying out large numbers of digital messages, and the experimental results give lower computational costs and communication overheads.

Table 3: Comparison between the proposed scheme and the two existing blind signcryption schemes based on a task in one electronic document.

| Item | Method | | | | | |
|---|---|---|---|---|---|---|
| | A new efficient blind signcryption (Yu and He, 2008) [36] | | Blind signcryption scheme based on elliptic curves (Ullah et al., 2014) [37] | | Our signcryption-combined scheme | |
| | Cost | | | | | |
| | Computational cost | Rough estimation | Computational cost | Rough estimation | Computational cost | Rough estimation |
| Signcryption | | | | | | |
| Encryption | Not specified | Not specified | Not specified | Not specified | $2T_{\mathrm{ECMUL}} + 1T_h + 1t_h + 2T_{\mathrm{MUL}} + 1T_{\mathrm{ADD}}$ | $83T_{\mathrm{MUL}}$ |
| Blinding | $5T_{\mathrm{MUL}} + 8T_{\mathrm{EXP}} + 1T_{\mathrm{INVS}} + 5T_{\mathrm{ADD}} + 6t_h$ | $2167T_{\mathrm{MUL}}$ | $3T_{\mathrm{ECMUL}} + 3T_{\mathrm{MUL}} + 1T_{\mathrm{ECADD}} + 5T_{\mathrm{ADD}} + 1T_{\mathrm{INVS}} + 2t_h$ | $336T_{\mathrm{MUL}}$ | $1T_{\mathrm{ECMUL}} + 1T_{\mathrm{MUL}}$ | $30T_{\mathrm{MUL}}$ |
| Signing | | | | | $2T_{\mathrm{ECMUL}} + 1T_{\mathrm{ADD}}$ | $58T_{\mathrm{MUL}}$ |
| Unsigncryption | | | | | | |
| Unblinding | $2T_{\mathrm{MUL}} + 4T_{\mathrm{EXP}} + 4t_h$ | $964T_{\mathrm{MUL}}$ | $1T_{\mathrm{ECMUL}} + 2T_{\mathrm{MUL}} + 2T_{\mathrm{ECADD}} + 1T_h + 1t_h$ | $64T_{\mathrm{MUL}}$ | $1T_{\mathrm{ECMUL}} + 1T_{\mathrm{ECADD}} + 3T_{\mathrm{MUL}} + 1T_{\mathrm{ADD}}$ | $37T_{\mathrm{MUL}}$ |
| Signature verification | | | | | $2T_{\mathrm{ECMUL}} + 2T_{\mathrm{ECADD}} + 1T_h$ | $91T_{\mathrm{MUL}}$ |
| Decryption | Not specified | Not specified | Not specified | Not specified | $1T_{\mathrm{ECMUL}} + 1T_{\mathrm{ECADD}} + 1t_h$ | $34T_{\mathrm{MUL}}$ |
| Total cost without encryption and decryption | $7T_{\mathrm{MUL}} + 12T_{\mathrm{EXP}} + 1T_{\mathrm{INVS}} + 5T_{\mathrm{ADD}} + 10t_h$ | $3131T_{\mathrm{MUL}}$ | $4T_{\mathrm{ECMUL}} + 5T_{\mathrm{MUL}} + 3T_{\mathrm{ECADD}} + 5T_{\mathrm{ADD}} + 1T_{\mathrm{INVS}} + 1T_h + 3t_h$ | $400T_{\mathrm{MUL}}$ | $6T_{\mathrm{ECMUL}} + 3T_{\mathrm{ECADD}} + 4T_{\mathrm{MUL}} + 2T_{\mathrm{ADD}} + 1T_h$ | $216T_{\mathrm{MUL}}$ |

Table 4: Performance comparison between the proposed scheme and the other two schemes across multiple documents.

| Number of documents | Method | | |
|---|---|---|---|
| | A new efficient blind signcryption (Yu and He, 2008) [36] | Blind signcryption scheme based on elliptic curves (Ullah et al., 2014) [37] | The proposed scheme |
| 1 | $3131T_{\mathrm{MUL}}$ | $400T_{\mathrm{MUL}}$ | $521T_{\mathrm{MUL}}$ |
| 2 | $6262T_{\mathrm{MUL}}$ | $800T_{\mathrm{MUL}}$ | $521T_{\mathrm{MUL}}$ |
| 3 | $9393T_{\mathrm{MUL}}$ | $1200T_{\mathrm{MUL}}$ | $521T_{\mathrm{MUL}}$ |
| 4 | $12524T_{\mathrm{MUL}}$ | $1600T_{\mathrm{MUL}}$ | $521T_{\mathrm{MUL}}$ |
| 5 | $15655T_{\mathrm{MUL}}$ | $2000T_{\mathrm{MUL}}$ | $521T_{\mathrm{MUL}}$ |
| 6 | $18786T_{\mathrm{MUL}}$ | $2400T_{\mathrm{MUL}}$ | $521T_{\mathrm{MUL}}$ |
| 7 | $21917T_{\mathrm{MUL}}$ | $2800T_{\mathrm{MUL}}$ | $521T_{\mathrm{MUL}}$ |
| 8 | $25048T_{\mathrm{MUL}}$ | $3200T_{\mathrm{MUL}}$ | $521T_{\mathrm{MUL}}$ |
| 9 | $28179T_{\mathrm{MUL}}$ | $3600T_{\mathrm{MUL}}$ | $521T_{\mathrm{MUL}}$ |
| 10 | $31310T_{\mathrm{MUL}}$ | $4000T_{\mathrm{MUL}}$ | $521T_{\mathrm{MUL}}$ |

Annotation: to strengthen the security protection mechanisms, the mutual identity verification phase to authenticate the communicating parties to each other is required to prevent the identity forgery or fraud, and the cost of each authentication thus takes $305T_{\mathrm{MUL}}$ time to calculate the complexity (i.e., $2T_{\mathrm{ECMUL}} + 1T_{\mathrm{ECADD}} + 2T_{\mathrm{MUL}} + 1T_{\mathrm{INVS}}$).

By providing the above-mentioned abilities of the security structure and the computation efficiency, the proposed scheme not only speeds up current blind signature techniques and digital information application programs, but also extends the field for a new protocol method using these secure yet efficient structure primitives. This facilitates much faster blind signatures and electronic messages processing as with many distributions that take place at scale, combining high performance with robust security for constructing various anonymous applications including electronic payment systems, voting services, credential-based access control processes, and digital content protection platforms.

## Abbreviations

$E(F_q)$: An elliptical curve $E$ over a finite field $F_q$
$G$: A base point of an elliptical curve
$d$: A prime order of $G$
$q$: A prime number such that $q > 2^{283}$
$\mathrm{id}_A$, $\mathrm{id}_B$, $\mathrm{id}_T$: User's identity information such as requester $A$, signer $B$ and verifier $T$
$\mathrm{PK}_{\mathrm{AS}}$, $n_{\mathrm{AS}}$: A public and private key pair from AS
$\mathrm{PK}_A$, $\mathrm{PK}_B$, $\mathrm{PK}_T$: Public keys of all the users as requester $A$, signer $B$ and verifier $T$
$n_A$, $n_B$, $n_T$: Private keys of all the users as requester $A$, signer $B$ and verifier $T$
$\mathrm{ca}_A$, $\mathrm{ca}_B$, $\mathrm{ca}_T$: The users' certificates for requester $A$, signer $B$ and verifier $T$
$Z_A$, $Z_B$, $Z_T$: Representative points on an elliptic curve $E$ defined over $F_q$
$e_A$, $e_B$, $e_T$: An identity value selected for requester $A$, signer $B$ and verifier $T$
$l_A$, $l_B$, $l_T$: A random number selected from AS for requester $A$, signer $B$ and verifier $T$
$u_1$, $u_2$, $u_3$: Nonce values
$Q_A$, $Q_B$, $Q_T$: Intermediate points on an elliptic curve $E$ defined over $F_q$
$h_1(\cdot)$: A hash function to be used for public key, identity, and plaintext messages
$h_2(\cdot)$: A hash function to be used for ciphertext messages
$f_{m2p}(\cdot)$: A conversion function from a message to an elliptic curve point
$f_{p2m}(\cdot)$: A conversion function from an elliptic curve point to a message
V: A plaintext segment
$C$: A ciphertext stream
$w$: A permutation value in bit shift operations
$t$: A hash value derived from a plaintext sequence
$m$: A hash value derived from a ciphertext sequence
$\alpha$: A blinded message
$\beta$: A random integer number
$k$: An arbitrary integer number
$K$: A verification point
$R$: A secret element
$S$: A blind signature
$\|$: The concatenation operation.

## Competing Interests

The authors declare that they have no competing interests.

## References

[1] C. Brzuska, M. Fischlin, A. Lehmann, and D. Schröder, "Unlinkability of sanitizable signatures," in *Proceedings of the 13th International Conference on Practice and Theory in Public Key Cryptography (PKC '10)*, vol. 6056, pp. 444–461, Springer, Paris, France, May 2010.

[2] C.-F. Chou, W. C. Cheng, and L. Golubchik, "Performance study of online batch-based digital signature schemes," *Journal of Network and Computer Applications*, vol. 33, no. 2, pp. 98–114, 2010.

[3] A. C. Enache, "About group digital signatures," *Journal of Mobile, Embedded and Distributed Systems*, vol. 4, no. 3, pp. 193–202, 2012.

[4] F.-G. Jeng, T.-L. Chen, and T.-S. Chen, "An ECC-based blind signature scheme," *Journal of Networks*, vol. 5, no. 8, pp. 921–928, 2010.

[5] D. Pointcheval and J. Stern, "Security arguments for digital signatures and blind signatures," *Journal of Cryptology*, vol. 13, no. 3, pp. 361–396, 2000.

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

[7] N. M. F. Tahat, E. S. Ismail, and R. R. Ahmad, "A new blind signature scheme based on factoring and discrete logarithms," *International Journal of Cryptology Research*, vol. 1, no. 1, pp. 1–9, 2009.

[8] A. Thu, "Implementation of an efficient blind signature scheme," *International Journal of Innovation, Management and Technology*, vol. 5, no. 6, pp. 443–448, 2014.

[9] S. Verma and B. Kumar Sharma, "New proxy blind multi signature based on integer-factorization and discrete-logarithm problems," *Bulletin of Electrical Engineering and Informatics*, vol. 1, no. 3, pp. 185–190, 2012.

[10] D. Chaum, "Blind signatures for untraceable payments," in *Advances in Cryptology—CRYPTO '82*, vol. 3 of *Lecture Notes in Computer Science*, pp. 199–203, Springer, 1983.

[11] D. Chaum, "Security without identification: transaction systems to make big brother obsolete," *Communications of the ACM*, vol. 28, no. 10, pp. 1030–1044, 1985.

[12] L. Harn, "Cryptanalysis of the blind signatures based on the discrete logarithm problem," *Electronics Letters*, vol. 31, no. 14, p. 1136, 1995.

[13] C.-C. Lee, M.-S. Hwang, and W.-P. Yang, "A new blind signature based on the discrete logarithm problem for untraceability," *Applied Mathematics and Computation*, vol. 164, no. 3, pp. 837–841, 2005.

[14] J. L. Camenisch, J. M. Piveteau, and M. A. Stadler, "Blind signatures based on discrete logarithm problem," in *Proceedings of the Advances in Cryptology (EUROCRYPT '94)*, vol. 950 of *Lecture Notes in Computer Science*, pp. 428–432, Springer, Perugia, Italy, 1994.

[15] C.-I. Fan and C.-L. Lei, "Efficient blind signature scheme based on quadratic residues," *Electronics Letters*, vol. 32, no. 9, pp. 811–813, 1996.

[16] 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.

[17] C.-I. Fan, C.-I. Wang, and W.-Z. Sun, "Fast randomization schemes for Chaum blind signatures," *International Journal of Innovative Computing, Information and Control*, vol. 5, no. 11, pp. 3887–3900, 2009.

[18] M.-S. Hwang, C.-C. Lee, and Y.-C. Lai, "An untraceable blind signature scheme," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 86, no. 7, pp. 1902–1906, 2003.

[19] W.-S. Juang and C.-L. Lei, "Partially blind threshold signatures based on discrete logarithm," *Computer Communications*, vol. 22, no. 1, pp. 73–86, 1999.

[20] V. R. L. Shen, Y. F. Chung, T. S. Chen, and Y. A. Lin, "A blind signature based on discrete logarithm problem," *International Journal of Innovative Computing, Information and Control*, vol. 7, no. 9, pp. 5403–5416, 2011.

[21] K. Rabah, "Elliptic curve cryptography over binary finite field GF($2^m$)," *Information Technology Journal*, vol. 5, no. 1, pp. 204–229, 2006.

[22] S. Su and S. Lü, "A public key cryptosystem based on three new provable problems," *Theoretical Computer Science*, vol. 426-427, pp. 91–117, 2012.

[23] 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.

[24] I. Bütün and M. Demirer, "A blind digital signature scheme using elliptic curve digital signature algorithm," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 21, no. 4, pp. 945–956, 2013.

[25] K. Chakraborty and J. Mehta, "A stamped blind signature scheme based on elliptic curve discrete logarithm problem," *International Journal of Network Security*, vol. 14, no. 6, pp. 316–319, 2012.

[26] D. Jena, S. K. Jena, and B. Majhi, "A novel untraceable blind signature based on elliptic curve discrete logarithm problem," *International Journal of Computer Science and Network Security*, vol. 7, no. 6, pp. 269–275, 2007.

[27] M. Nikooghadam and A. Zakerolhosseini, "An efficient blind signature scheme based on the elliptic curve discrete logarithm problem," *International Journal of Information Security*, vol. 1, no. 2, pp. 125–131, 2009.

[28] Shamsherullah, Nizamudin, A. I. Umar, Noor-ul-Amin, R. Ullah, and I. Ullah, "Blind signcryption scheme based on hyper elliptic curve for untraceable payment system," in *Proceedings of the 13th International Conference on Statistical Sciences*, vol. 28, pp. 337–344, Peshawar, Pakistan, 2015.

[29] A. Sadat, I. Ullah, H. Khattak, S. Ullah, and Amjadurrehman, "Proxy blind signcrypion based on elliptic curve," *International Journal of Computer Science and Information Security*, vol. 14, no. 3, pp. 257–262, 2016.

[30] C.-H. Lin, R.-H. Hsu, and L. Harn, "Improved DSA variant for batch verification," *Applied Mathematics and Computation*, vol. 169, no. 1, pp. 75–81, 2005.

[31] C.-H. Tsai and P.-C. Su, "Multi-document threshold signcryption scheme," *Security and Communication Networks*, vol. 8, no. 13, pp. 2244–2256, 2015.

[32] Y. Zheng, "Digital signcryption or how to achieve cost(signature & encryption) $\ll$ cost(signature) + cost(encryption)," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1294, pp. 165–179, 1997.

[33] H.-Y. Lin, T.-S. Wu, and S.-K. Huang, "Certificate-based secure three-party signcryption scheme with low costs," *JISE. Journal of Information Science and Engineering*, vol. 28, no. 4, pp. 739–753, 2012.

[34] A. Braeken and P. Porambage, "Efficient generalized signcryption based on ECC," *International Journal on Cryptography and Information Security*, vol. 5, no. 2, pp. 1–13, 2015.

[35] N. Din, A. I. Umar, N. Amin, and Abdul Waheed, "A novel multi receiver signcryption scheme based on elliptic curves for firewalls," *Journal of Applied Environmental and Biological Sciences*, vol. 6, no. 2S, pp. 144–150, 2016.

[36] X. Yu and D. He, "A new efficient blind signcryption," *Wuhan University. Journal of Natural Sciences*, vol. 13, no. 6, pp. 662–664, 2008.

[37] R. Ullah, N. Uddin, A. I. Umar, and N. Amin, "Blind signcryption scheme based on elliptic curves," in *Proceedings of the Conference on Information Assurance and Cyber Security (CIACS '14)*, pp. 51–54, IEEE Xplore Digital Library, Rawalpindi, Pakistan, June 2014.

[38] S. A. Ch, Nizamuddin, and M. Sher, "Public verifiable signcryption schemes with forward secrecy based on hyperelliptic curve cryptosystem," *Communications in Computer and Information Science*, vol. 285, pp. 135–142, 2012.

[39] S. A. Ch, Nizamuddin, M. Sher, A. Ghani, H. Naqvi, and A. Irshad, "An efficient signcryption scheme with forward secrecy and public verifiability based on hyper elliptic curve cryptography," *Multimedia Tools and Applications*, vol. 74, no. 5, pp. 1711–1723, 2015.

[40] Nizamuddin, S. A. Ch, W. Nasar, and Q. Javaid, "Efficient signcryption schemes based on hyperelliptic curve cryptosystem," in *Proceedings of the 7th International Conference on Emerging Technologies (ICET '11)*, September 2011.

[41] D. J. Bernstein and T. Lange, "Hyper-and-elliptic-curve cryptography," *LMS Journal of Computation and Mathematics*, vol. 17, pp. 181–202, 2014.

[42] J. W. Bos, C. Costello, H. Hisil, and K. Lauter, "Fast cryptography in genus 2," in *Advances in Cryptology—EUROCRYPT 2013*, vol. 7881 of *Lecture Notes in Computer Science*, pp. 194–210, Springer, 2013.

[43] M. Gobi, R. Sridevi, and R. Rahini priyadharshini, "A comparative study on the performance and the security of RSA and ECC algorithm," in *Proceedings of the UGC Sponsored National Conference on Advanced Networking and Applications*, pp. 168–171, Jalgaon, India, 2015.

[44] R. Sinha, H. K. Srivastava, and S. Gupta, "Performance based comparison study of rsa and elliptic curve cryptography," *International Journal of Scientific & Engineering Research*, vol. 4, no. 5, pp. 720–725, 2013.

[45] N. Tahat and E. E. Abdallah, "A new signing algorithm based on elliptic curve discrete logarithms and quadratic residue problems," *Italian Journal of Pure and Applied Mathematics*, vol. 32, pp. 125–132, 2014.