The ISC Int'l Journal of Information Security

January 2021, Volume 13, Number 1 (pp. 59–72)

http://www.isecure-journal.org

An Efficient Pairing-Free Identity Based Proxy Blind Signature Scheme with Message Recovery

Salome James [1], Gowri Thumbur [2], and P. Vasudeva Reddy [1,*]

[1] *Department of Engineeering Mathematics, Andhra University, Visakhapatnam.*

[2] *Department of Electronics and Communication Engineering, GITAM University, Visakhapatnam.*

ARTICLE INFO.

Article history:
Received: November 12, 2019
Revised: April 10, 2020
Accepted: October 1, 2020
Published Online: October 6, 2020

Keywords:
ID-based Framework, Proxy Signature, Blind Signature, ECDLP, Message Recovery

Type: Research Article

doi: 10.22042/isecure.2020.208473.495

Abstract

In recent years, due to their potential applications, proxy blind signatures became an active research topic and are an extension of the basic proxy signature. A proxy blind signature scheme enables a proxy signer to produce a blind signature on behalf of an original signer. Such schemes are useful in many practical applications such as e-commerce, e-voting, e-tendering systems. Many proxy blind signature schemes have been proposed in the literature. In order to improve the efficiency and to adopt resource constrained devices, in this paper, we propose a pairing free ID-based proxy blind signature scheme with message recovery. The proposed scheme is proven secure against the random oracle model under the hardness assumption of the elliptic curve discrete logarithm problem. We compare our scheme with the other proxy blind signature schemes. The efficiency analysis shows that our scheme is more efficient in terms of computational and communicational point of view. Also due to the message recovery property, our scheme can be deployed easily in low band width devices.

1 Introduction

In today's commercial environment, internet transactions must be used in conjunction with the security services of authentication and non-repudiation of origin of requests sent from a requester, in order to avoid fraudulent action by the signer. The above mentioned security services can be accomplished by the cryptography protocol called digital signature. A digital signature is one of the most important and useful primitive in the field of cryptography, which ensures authentication, data integrity and non-repudiation for electronic transactions in the digital world. In a traditional public key cryptography (PKC), each user has two keys, a private key and a public key. The binding between the public key and the identity of a user is acquired through a digital certificate. Before using the public key of a user, the participant must first verify the certificate of the user. Consequently, this system needs a larger amount of computing time and storage, when the numbers of users increase rapidly. In 1984, Shamir [1] introduced the notion of ID-based cryptography in order to simplify the key management and to eliminate the need for public key certificates. The property of an ID-based cryptography is that a user's identity is his public key and the private key can be obtained by a trusted authority called private key generator (PKG). Since the inception of digital signature schemes, the cryptographer community has always been attempting to design small size signatures to apply for low bandwidth applications. A digital signature scheme with message recovery enables bandwidth to be conserved,

* Corresponding author.
Email addresses: salomecrypto@gmail.com, gowri3478@ieee.org, vasucrypto@andhrauniversity.edu.in
ISSN: 2008-2045

when transmitting a signed message compared to a signature scheme with appendix. The concept of digital signature scheme with message recovery was first introduced by Nyberg and Rueppel [2]. In such a scheme, it is not necessary to transmit the original message along with the signature, because it can be recovered from the signature during the process of verification/message recovery.

Since the growth of electronic commerce, the preservation of the anonymity of users has been an essential necessity. The blind signature is one of the significant cryptographic tools which provide such anonymity for users. It is an interactive signature scheme between a user and a signer. The blind signature enables a user to acquire the signature of a message so that the signer knows neither the message nor the resulting signature. Chaum [3], introduced the notion of blind signature in 1982. The advantages of blind signatures have discovered their way into many privacy-oriented applications such as, anonymous electronic voting [4] and untraceable electronic cash systems [5].

In today's modern society, there is always a necessity to assign the signing capability to a trusted proxy who can sign on a message in place of the original user. The proxy signature scheme is a significant cryptographic technique, which enables an original signer to delegate his signing authority to another (proxy) signer, so that the proxy signer can sign any message on behalf of the original signer and the verifier can verify and distinguish between the original signature and the proxy signature. Proxy signatures have discovered several practical applications, including distributed systems, grid computing, mobile agent systems and mobile communications [6,7]. The concept of proxy signature was first introduced by Mambo *et al.* in 1996 [8]. After the implementation of Mambo *et al.* first scheme, many proxy signature schemes have been proposed [9,10,11]. Moreover, depending on the type of delegation, the proxy signature schemes can be categorized into three types: full delegation [10], partial delegation [11] and delegation by warrant [9].A significant extension of the basic proxy signature is the proxy blind signature, which can be extensively utilized in several practical applications. A proxy blind signature enables the proxy signer to produce a blind signature on behalf of the original signer. Proxy blind signature scheme integrates the properties of proxy signature and blind signature schemes. It is therefore, highly appropriate and can be extensively used for applications involving mobile agents, distributed systems and electronic transactions.

1.1 Related Work

The first proxy blind signature scheme was introduced by Lin and Jan [12] in 2000. Since then, many proxy blind signature schemes have been proposed in PKI-based settings [13-22] and ID-based settings [23-37]. In 2002, Tan *et al.* [13] proposed a proxy blind signature scheme based on Schnorr blind signature scheme. In 2003, Lal and Awasthi [14] pointed out that Tan *et al.*'s scheme [13] was insecure and proposed a new proxy blind signature scheme based on Mambo *et al.*'s scheme [8]. In 2007, Li and Wang [15] proposed a proxy blind signature scheme using verifiable self-certified public keys. In 2009, Qi and Wang [16] proposed an improved proxy blind signature scheme based on multiple hard problems such as factoring and elliptic curve discrete logarithm problems (ECDLP). In 2003, Zhang *et al.* [23] first proposed the ID-based proxy blind signature scheme from bilinear pairings. Thereafter, various ID-based proxy blind signature schemes [24-29] have been developed by the researchers. In 2009, Zhang [30] proposed two proxy blind signature schemes. In 2011, Pradhan and Mohapatra [31] proposed a proxy blind signature scheme based on ECDLP. In 2013, Tan [32] proposed an efficient pairing-free provably secure ID-based proxy blind signature scheme. In the same year 2013, Prabhadevi and Natarajan [33] proposed an ID-based proxy blind signature based on ECDLP for secure vehicular communications to improve the network performance. Furthermore, in 2013, Chen *et al.* [34] proposed an untraceability analysis of two ID-based proxy blind signature from bilinear pairings, in which they pointed out that Zhang's schemes [30] are not correct. In 2014, Chande [35] proposed an improved proxy blind signature scheme based on ECDLP. In their paper, they mounted a linkability attack on Pradhan and Mohapatra's [31] proxy blind signature scheme. In 2016, Padhye and Tiwari [36] proposed an efficient ID-based proxy blind signature with pairing-free realization which reduces the running time. Recently in 2017, a secure ID-based blind and proxy blind signature scheme from bilinear pairings was proposed by Sarde and Banerjee [37], and the security of their scheme is based on CDH problem. Also, in 2013, Diao *et al.* [38] proposed a new proxy blind signature scheme with message recovery property based on Tan *et al.* [13] and Abe-Okamoto's [39] signature schemes. However their scheme is not much efficient due to the large signature size. There are no efficient proxy blind signature schemes with message recovery in the literature. Hence, in order to achieve the efficiency and advantage of message recovery property, it is necessary to design proxy blind signature schemes with message recovery in ID-based setting.

1.2 Motivation

Most of these existing ID-based proxy blind signature schemes are designed using bilinear pairings over elliptic curves. But, the computation of bilinear pairings over elliptic curves is very expensive and time consuming which results low processing in protocols. For example, the computation cost of a pairing operation is approximately twenty times higher than that of an elliptic curve scalar multiplication. Hence the pairing based schemes are not much efficient for practical application where the computation and communication powers are limited.

Recently, Elliptic Curve Cryptography (ECC) based schemes have become more popular, as they provide greater security with smaller keys in size. Furthermore, these schemes require low computation and communication cost and therefore the time management, storage space and consumption of bandwidth become very less with these small keys. Hence, inorder to improve the computation, communication, storage efficiency; it is required to construct a new and secure Identity based proxy blind signature scheme in a pairing free environment.

1.3 Our Contributions

To improve the computational efficiency and communication overhead, in this paper, we present a Pairing Free ID-based Proxy Blind Signature Scheme with Message Recovery (PF-IDBPBS-MR). The main contributions of this paper are summarized as follows.

- We proposed a new PF-IDBPBS-MR scheme which integrates the concepts of proxy and blind signature with message recovery property in identity based framework.
- Our PF-IDBPBS-MR scheme is proven secure against the existential forgery on adaptive chosen message and identity attacks under the hardness assumption of Elliptic Curve Discrete Logarithm Problem (ECDLP) in the Random Oracle Model (ROM).
- The proposed scheme does not use any expensive bilinear pairing operations, which improves the computational efficiency of the proposed scheme.
- Due to the message recovery property and ECC based cryptography without pairings, the proposed scheme improves the communication efficiency.
- Finally, we compare our scheme with the existing related schemes in terms of computation and communication cost point of view.

1.4 Organisation of the Paper

The paper is organized as follows. Section 2, briefly provides some preliminaries. Section 3, provides the syntax and security models for our Pairing Free ID-based Proxy Blind Signature Scheme with Message Recovery (PF-IDBPBS-MR). Section 4 presents the proposed PF-IDBPBS-MR scheme. Section 5, presents the security analysis of our scheme against various types of adversaries. This section also provides a detailed efficiency comparison of our scheme. Finally, Section 6 concludes the paper.

2 Preliminaries

This section briefly describes the fundamental concepts of the elliptic curve and the complexity assumption, on which the proposed scheme is designed and attains the desired security.

2.1 Elliptic Curve Cryptography

Elliptic curve cryptography (ECC) plays a major role in the modern PKC [40,41], because of the computation, communication and security strengths.

Let $E_q(a,b)$ be a set of elliptic curve points over the prime field F_q , defined by the non-singular elliptic curve equation $y^2 \mod q = (x^3 + ax + b) \mod p$ with $a, b \in F_q$ and $(4a^3 + 27b^2) \mod q \neq 0$. The additive elliptic curve group is defined as $G_q\{(x,y) : x, y \in F_q\}$ and $(x,y) \in E_q(a,b) \cup \{O\}$, where the point O is known as *point at infinity*. The order of the elliptic curve over F_q is O(E(F_q)) satisfies the relation $1 - 2\sqrt{q} \leq O(E(F_q)) \leq q + 1$. The scalar multiplication on the cyclic group G_q defined as $kP = P+P+\cdots+P$ (k times). Here $P \in G_q$ is the generator of order n.

2.2 Elliptic Curve Discrete Logarithm Problem

Given a random instance p the generator of G and $Q = aP$ where $a \in Z_q^*$ the Elliptic Curve Discrete Logarithm Problem (ECDLP) is to find a from P and Q.

- Given a tuple (P, Q), it is computationally hard for any Probabilistic Polynomial Time (PPT) algorithm ADV to determine a, where $Q = aP$ and $a \in Z_q^*$.
- The probability that any polynomial-time bounded algorithm AdV can solve the ECDLP is defined as $Advg_{Adv,Gq}^{ECDLP}$ =Prob $\{Adv(P,Q) = a \ni P, Q \in G_q \text{ and } Q = aP, a \in Z_q^*\}$.

2.3 Notations

Table-1 presents the symbols and their descriptions used in this paper.

Table 1. Notations and Meanings

Notation	Meaning
$E(F_q)$	Group of elliptic curve points over F_q
k	Security parameter
G_1	An additive group which is generated P by with the order $\hat{q}$ on the super singular elliptic curve
G	An additive cyclic group generated by a point P on a non-singular elliptic curve
H_i, F_i	Cryptographic hash functions
$a \parallel b$	Concatenation of two strings a and b
$\oplus$	X-OR computation in the binary system
$[x]_{10}$	Decimal representation of $x \in\in 0, 1^*$
$[y]_2$	Binary representation of $y \in Z$
$l2\|\beta\|$	The first I_2 bits of β from the left side
$\|\beta\|_{l_1}$	The first I_1 bits of β from the right side
Ω	Signature on the message

2.4 Acronyms

Table-2 presents the acronyms used in this paper.

Table 2. Acronyms and explanation

Acronyms	Explanation
ECDLP	Elliptic Curve Discrete Logarithm Problem
PKC	Public Key Cryptography
ID-based	Identity-based
PF-IDBPBS-MR	Pairing Free Identity-based Proxy Blind Signature Scheme with Message Recovery
ECC	Elliptic Curve Cryptography
PPT	Probabilistic Polynomial Time
PKG	Private Key Generator
ROM	Random Oracle Model
EF-ACMA	Existential Forgery under the Adaptive Chosen Message Attack

3 Syntax and Security Model

3.1 Syntax

Let A denote the original signer with identity ID_A and private key d_A The original signer A delegates his signing rights to a proxy signer B with identity ID_B and private key d_B. A warrant is used to delegate the signing rights. We now present a formal model for our PF-IDBPBS-MR scheme. A formal model of the proposed scheme includes the following polynomial time algorithms: System Setup, Key Extract, Delegation Generation, Delegation Verification, Proxy Key Generation, Proxy Blind Signature Generation, Message Recovery and Proxy Blind Signature Verification. The following is a detailed description of these algorithms.

(1) **System Setup:** For a given security parameter $k \in Z^+$ the Private Key Generator (PKG) runs this algorithm and generates the system parameters Params and also the master key s. Params are made public and s is kept secret. Params are implicit input to all the algorithms below.
(2) **Key Extract:** The PKG runs this algorithm to generate the public key and private key for a given user's identity ID. PKG sends the private key to the corresponding user over a secure channel.
(3) **Delegation Generation:** This algorithm takes the private key d_A of the original signer and a warrant m_w as input and outputs the delegation W.
(4) **Delegation Verification:** This algorithm takes the identity d_A of the original signer and the delegation W as input and checks if it is a valid delegation from the original signer A.
(5) **Proxy Key Generation:** This algorithm takes the delegation W and some other secret information as input and outputs a signing key D_{psk} for the proxy signer.
(6) **Proxy Blind Signature Generation:** This is an interactive probabilistic polynomial time algorithm executed between proxy signer and user. In this algorithm, a user can obtain a proxy blind signature from a proxy signer on a message of his choice without linking the view of protocol with signature. During execution, the user sends a blinded message to proxy signer and the proxy signer generates a signature on it. The user, then receives this signature and unblinds it and displays Ω as the proxy blind signature.
(7) **Message Recovery and Proxy Blind Signature Verification:** This is a deterministic polynomial time algorithm, in which the verifier receives the signature Ω and takes the original signer's identity ID_A and the proxy signer's identity ID_B as input and after that recovers the message and displays acceptance or rejection.

3.2 Security Model of the Proposed PF-IDBPBS-MR Scheme

Here, we provide the security model of the proposed PF-IDBPBS-MR scheme. A proxy blind signature scheme should satisfy the unforgeability security feature and some additional security requirements. We

present all these security requirements in the following section.

3.2.1 Unforgeability

The unforgeability of our PF-IDBPBS-MR scheme guarantees that only the delegated proxy signer can generate a valid proxy blind signature and not even the original signer can generate a valid proxy blind signature on behalf of a proxy signer. In the following we consider three types of potential adversaries as defined in [42].

Type 1 Adversary: The adversary A_1 only contains the public keys of the original signer and the proxy signer.
Type 2 Adversary: The adversary A_2 contains the public keys of the original signer and the proxy signer. Additionally, the adversary A_2 contains the private key of the proxy signer.
Type 3 Adversary: The adversary A_3 contains the public keys of the original signer and the proxy signer. The adversary A_3 also contains the private key of the original signer.

From the capabilities of the above adversaries, it is evident that unforgeability with respect to type 2 and 3 adversaries implies unforgeability with respect to type 1 adversary. In the following, we only consider adversaries of type 2 and type 3 to prove the security through following games (Game-I and Game-II).

3.2.1.1 Game-I: Existential Unforgeability Against the Adversary A_2

The existential unforgeability of the proxy blind signature scheme is determined by considering the following security game between a challenger ξ and an adversary A_2.

(1) Initialization Phase: The challenger ξ executes the setup algorithm using input security parameter k and produces the system parameters master secret key and master public key and then sends them to the adversary A_2 and keeps s secret with itself.
(2) Queries Phase: The adversary A_2 adaptively makes the following queries to the challenger ξ on the oracles below.
   a. Key Extract Oracle: On receiving a query from the adversary the challenger computes by taking ID_i as input and forwards the output D_i to the adversary A_2.
   b. Delegation Generation Oracle: On receiving a query from the adversary A_2 the challenger ξ computes delegation W by taking the designator's identity ID_i and a warrant m_w.
   c. Delegation Verification Oracle: On receiving the input (ID_i, W, m_w) the challenger ξ verifies the validity of the delegation. If the delegation is valid, it outputs 1, otherwise it returns 0.
(3) Forgery. Finally, the adversary A_2 outputs (ID_i^*, W, m_w) as delegation and wins the game if the following holds.
   i) (ID_i^*, W, m_w) is a valid delegation.
   ii) ID_i^* has never been submitted to the extract oracle and never been queried to the Delegation generation oracle.

Definition 1. A proxy blind signature scheme is unforgeable against the adversary A_2 if the advantage of the above game I is negligible after making atmost q_{dg} delegation generation queries within the running time t.

3.2.1.2 Game-II: Existential Unforgeability Against the Adversary A_3

The existential unforgeability of the proxy blind signature scheme is proved by considering the following game between the challenger ξ and adversary A_3.

(1) Initialization Phase: The initialization phase made by the type 3 adversary A_3 is similar to that of the type 2 adversary A_2 described under the existential unforgeability against the adversary A_2
(2) Queries Phase: The adversary A_3 adaptively makes the following queries to the challenger on the oracles below.
   a. The Key Extract Oracle and Delegation Generation Oracle made by the type 3 adversary A_3 are similar to that of the type 2 A_2 adversary described under the existential unforgeability against the adversary A_2.
   c. Proxy Key Generation Oracle: When adversary A_3 queries a proxy key generation oracle of the proxy signer for W, m_w the challenger ξ computes the proxy signing key D_{psk} and ξ responds to the adversary A_3 with D_{psk}.
   d. Proxy Blind Signature Generation Oracle: This oracle takes the delegation W and message $m \in \{0,1\}^{l_1}$ as input and outputs a proxy blind signature Ω created by the proxy signer.
(3) Forgery: Finally, the adversary A_3 outputs (ID_i^*, m^*, Ω^*) as forgery and wins the game if,

i) $\Omega*$ is a valid signature.

ii) ID_i^* has never requested to the Extraction Oracle and Proxy Key Generation, (ID_i^*, m^*) has never requested to the Proxy Blind Signature Oracle.

Definition 2. A proxy blind signature scheme is existentially unforgeable against the adversary A_3, if the advantage of the above game II is negligible after making at most q_{pbs} proxy blind signature queries within the running time t.

Definition 3. The proposed PF-IDBPBS-MR scheme is said to be existentially unforgeable under the adaptive chosen message and identity attacks, if there exists no probabilistic polynomial time adversaries (Type 2 and Type 3) with non-negligible advantage in the above two games.

3.2.2 Additional Security Requirements

In addition to the above Unforgeability security requirement, a proxy blind signature scheme must also satisfy the following security requirements: (1) Verifiability (2) Identifiability (3)Prevention of misuse and (4) Blindness.

(1) **Verifiability:** From the proxy signature, the verifier can be convinced of the original signer's agreement on the signed message.
(2) **Identifiability:** Anyone can determine the identity of the corresponding proxy signer from the proxy signature.
(3) **Prevention of misuse:** The proxy signer cannot use the proxy key for other purposes than generating a valid proxy signature. That is he/she is unable to sign messages which are not authorized by the original signer.
(4) **Blindness:** A signature is supposed to be blind if a given message-signature pair and the signer's view are statistically independent. While correctly operating one instance of the blind signature scheme, let the output be $(R_A, R_B, Y, \sigma, m_w, v)$ (i.e., message-signature pair) and the view of the protocol V^1 Later, the signer is not able to link V^1 to $(R_A, R_B, Y, \sigma, m_w, v)$ The content of the message is therefore, blind to the signer.

Definition 4. (Blindness) Let Adv be a probabilistic polynomial-time adversary which performs the role of the signer, U_0 and U_1 be two honest users. U_0 and U_1 engage in the blind signature issuing scheme with Adv on messages m_e and m_{1-e} and output signatures σ_e σ_{1-e} respectively, where $b \in \{0,1\}$ is a random bit chosen uniformly. $(m_b, m_{1-b}, \in_b, \in_{1-b})$ are sent to Adv and then Adv outputs $b^1 \in \{0,1\}$ For all such Adv , U_0 and U_1 for any constant c , and for sufficiently large n, $\left|pr\left[b = b^1\right] - 1/2\right| < n^{-c}$.

4 Proposed PF-IDBPBS-MR Scheme

The proposed Pairing Free Identity-based Proxy Blind signature with Message Recovery scheme consists of the following algorithms:

System Setup: For a given security parameter $k \in Z^+$, the PKG runs this algorithm as follows.

(1) Choose a cyclic additive group G of prime order q with the points on an elliptic curve E and P as the generator of G.
(2) Select a random $s \in Z_q^*$ as the master secret key and sets the master public key $P_{pub} = sP$.
(3) Choose $H_1 : \{0,1\}^* \to Z_q^*$, $H_2 : \{0,1\}^* \to Z_q^*$, $H_3 : \{0,1\}^* \to Z_q^*$,$H_4 : \{0,1\}^* \to Z_q^*$ and $F_1 : \{0,1\}^{l_1} \to \{0,1\}^{l_2}$, $F_2 : \{0,1\}^{l_2} \to \{0,1\}^{l_1}$ as hash functions I_1 and I_2 are positive integers such that $|q| = l_1 + l_2$.
(4) PKG publishes the system parameters $P_{arams} = (E, G, q, P, P_{pab}, H_1, H_2, H_3, H_4, F_1, F_2, l_1, l_2)$ as public and keeps the master key $\langle s \rangle$ as secret.

Key Extract: PKG runs this algorithm by taking user's identity ID_i , and system parametersP_{arams} as input. The PKG selects a random number $r_i \in Z_q^*$ and computes $R_i = r_iP$; $h_i = H_1(ID_i, R_i, P_{pub})$; $d_i = (r_i + sh_i) mod q$.

PKG sends $D_i = (d_i, R_i)$ to the user securely. The user keeps d_i as his private key and publishes $.R_i$ The user can validate D_i by checking whether the equation $d_iP = R_i + h_1 pab$ holds or not.

Clearly, $d_iP = (r_i + sh_i)P = r_iP + sh_iP = R_i + h_iP_{pub}$

Delegation Generation: The original signer produces a warrant m_w which keeps the record of proxy information such as the identities of the original signer, proxy signer, proxy validity period etc. This algorithm takes original signer's secret key d_A and a warrant m_w as input and outputs the delegation W The original signer A does the following.

(1) Select random $a \in Z_q^*$ and compute $Y = aP$
(2) Compute $h_{1,A} = H_2(m_w, Y, ID_B)$. $x = h_{1,A}d_A + a mod q$.

The original signer A outputs the delegation $W = (ID_A, R_A, ID_B, m_w, Y, x)$ on the warrant m_w and sends it to the proxy signer B

Delegation Verification: To verify the delegation W for the message m_w the proxy signer B initially computes $h_{1A} = H_2(m_w, Y, ID_B), h_A = H_1(ID_A, R_A, P_{pub})$, and checks whether the equation $xP = h_{1A}(R_A + h_AP_{pab}) + Y$ holds or not.

If it holds, the proxy signer B accepts the delegation (Y, x) corresponding to ID_A, R_A, ID_B on m_w Otherwise, rejects.

Proxy Key Generation: If the proxy signer B accepts the delegation (Y, x) then B computes the proxy signing key by using the original signer $A's$ delegation key and proxy signer $B's$ private key d_B.

Compute $D_{psk} = x + d_B h_{1B} mod q$ where $h_{1B} = H_2(m_w, Y, ID_A)$.

Proxy Blind Signature Generation: In order to sign a message $m \in \{0,1\}^{l_1}$ blindly by the proxy signer, the user C and the proxy signer B performs the following steps.

1. Signer: The proxy signer B chooses a number $y \in Z_q^*$ and computes $U = yP$ and sends (U, R_B, R_A, Y) to the user C as a commitment.

2. Blinding: The user C chooses blinding factors $n_1, n_2 \in Z_q^*$ and computes
$\beta = F_1(m) \parallel F_2((F_1(m) \oplus (m))$
$\sigma = n_1 U + n_2 \beta P$
$z_1 = H_3(ID_B, R_B, \sigma)$ and $z_2 = n_1^{-1} z_1 mod q$ Now the user C sends z_2 to the proxy signer B.

3. Signing: The proxy signer B computes $t_1 = (y + z_2 D_p sk)$ and sends back t_1 to the user C.

4. Unblinding: The user C computes the following.
$t_2 = ((n_1 t_1 + n_2 \beta) mod q$
$\alpha = H_4(ID_B, t_2 P)$ and $v = [\alpha \oplus \beta]_{10}$.

The user C outputs $(m, R_A, R_B, Y, \sigma, m_w, v)$ and $\Omega = (R_A, R_B, Y, \sigma, m_w, v)$ is the valid proxy blind signature on the message m. The proxy blind signature generation process can be shown in fig-1.

Proxy Blind Signature Verification: Given the identities ID_A and ID_B to verify the blind signature for the message m the verifier executes the following.

Compute $h_A = H_1(ID_A, R_A, P_{pub})$;
$h_B = H_1(ID_B, R_B, P_{pab})$;
$h_{1A} = H_2(m_w, Y, ID_B)$,
$h_{1B} = H_2(m_w, Y, ID_A)$;
$z_1 = H_3(ID_B, R_B, \sigma)$;
$\tilde{\alpha} = H_4(ID_B, \sigma + z_1[Y + h_{1A}(R_A + h_A P_{pub}) + h_{1B}(R_B + h_B P_{pub})]$;
$\tilde{\beta} = [v]_2 \oplus \tilde{\alpha}$.

The verifier recovers the message $\tilde{m} = |\tilde{\beta}|_{L_1} \oplus F_2\left[l2|\tilde{\beta}|\right]$.

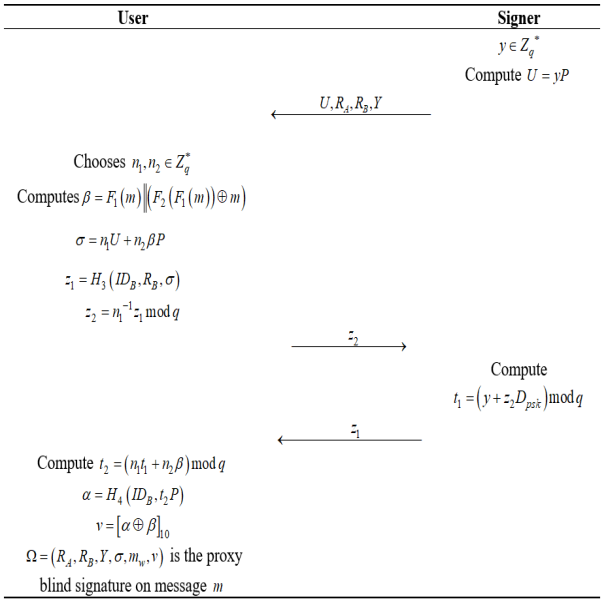

Figure 1. The proxy blind signature issuing protocol

Accept the signature Ω as valid signature on the message $\tilde{m} = m$, if and only if $l2|\tilde{\beta}| = F_1(\tilde{m})$.
Proof of Correctness: The correctness of the scheme can be verified as follows. Consider $t_2 P = (n_1 t_1 + n_2 \beta) P$
$= n_1(y + z_2 D_{psk}) + n_2 \beta) P$
$= n_1(y + z_2 D_{psk}) P + n_2 \beta P$
$= n_1 yP + n_1(n_1^{-1} z_1) D_{psk} P + n_2 \beta P$
$= n_1 yP + z_1 D_{psk} P + n_2 \beta P$
$= n_1 yP + z_1(x + d_B h_{1B}) P + n_2 \beta P$
$= n_1 yP + z_1(h_{1A} d_A + a + d_B h_{1B}) P + n_2 \beta P$
$= n_1 yP + z_1[h_{1A}(r_A + s h_A) + a + h_{1B}(r_B + s h_B)] P + n_2 \beta P$
$= \sigma + z_1[Y + h_{1A}(R_A + h_A P_{pub}) + h_{1B}(R_B + h_B P_{pub})]$

5 Analysis of the Proposed PF-IDBPBS-MR Scheme

The security analysis and efficiency analysis of the proposed PF-IDBPBS-MR scheme are provided in this section.

5.1 Security Analysis of the Proposed PF-IDBPBS-MR Scheme

In the following, we will analyse the security of our PF-IDBPBS-MR scheme. We prove the unforgeability and the additional security requirements of our PF-IDBPBS-MR scheme.

5.1.1 Unforgeability Against Type 2 and Type 3 Adversaries

In this section, we discuss the unforgeability of the proposed PF-IDBPBS-MR scheme against Type 2

and Type 3 adversaries through the following theorems.

Theorem 1. *The proposed PF-IDBPBS-MR scheme is existentially unforgeable under the adaptive chosen message and identity attacks against the type 2 adversary A_2 in the random oracle model provided the ECDLP is intractable by any polynomial time-bounded algorithm in the elliptic curve group G.*

Proof. Suppose an adaptive chosen message and identity type 2 adversary A_2 wants to break the security of the proposed PF-IDBPBS-MR scheme, then we prove that a polynomial time-bounded adversary A_2 can solve the ECDLP with the help of the algorithm ξ The type 2 adversary A_2 contains the public keys of the original signer and the proxy signer and also contains the private key of the proxy signer. The unforgeability of the proposed PF-IDBPBS-MR scheme against type 2 adversary A_2 requires that it is difficult to produce a valid delegation without the private key of the original signer. We prove that, if there exists an adversary who can forge a valid delegation of our scheme, then there exists an algorithm ξ to solve an instance of ECDLP. Therefore, ξ outputs s for a given random instance $(P, sP) \in G$ where $s \in Z_q^*$ In order to solve the ECDLP, ξ sets the master secret key as s and master public key $P_{pab} = sP$ as where s is unknown. The adversary A_2 and challenger ξ interacts as defined in Game I. The simulation process is considered to be in the random oracle model.

(1) **Initialization Phase:** ξ executes the setup algorithm and produces the system parameters $Params$ and sends $Params$ to the adversary A_2. ξ responds to the adversary A_2 queries as follows.

(2) **Queries Phase:** Adversary A_2 performs the oracle simulation and the algorithm ξ responds to these oracles as follows.

a. Queries on Oracle $H_1(H_1(ID_i, R_i, P_{pab}))$: ξ maintains an initial-empty H_1- oracle list L_1 which contains the tuples of the form $(ID_i, R_i, P_{pub}, l_i)$ When the adversary A_2 asks a H_1 query with the input (ID_i, R_i, P_{pub}) then ξ returns l_i if there is a tuple $(ID_i, R_i, P_{pub}, l_i)$ in L_1. Otherwise ξ chooses a random $l_i \in Z_q^*$ and adds $(ID_i, R_i, P_{pub}, l_i)$ to the list L_1. Finally ξ returns l_i to the adversary A_2.

b. Queries on Oracle $H_2(H_2(m_w, Y_i, ID_i))$: ξ maintains an initial-empty H_2 - oracle list L_2, which contains the tuples of the form $(m_w, Y_i, ID_i, l_1 i)$. When the adversary A_2 asks a H_2 query with the input (m_w, Y_i, ID_i) then ξ returns $l_1 i$ if there is a tuple $(m_w, Y_i, ID_i, l_1 i)$ in L_2. Otherwise ξ chooses a random $l_1 i \in Z_q^*$ and adds $(m_w, Y_i, ID_i, l_1 i)$ to the list L_2. Finally ξ returns $l_1 i$ to the adversary A_2

c. Key Extraction Queries: When adversary A_2 makes this query on ID_i, ξ first makes queries on H_1 and recovers l_i from L_1 list. Then ξ replies to the adversary A_2 as follows.
   i. If $i =$ original signer, ξ aborts.
   ii. If $i \neq$ original signer, ξ chooses $r_i \in z_q^*$ sets $R_i = r_i P - l_i P_{pub}$ and $d_i = r_i$

d. Delegation Generation Queries: On receiving a Delegation Generation query on the warrant m_w with the original signer's identity ID_i, ξ first recovers the values l_i, l_{1i} from L_1 and L_2 respectively and then performs the following.
   i. If $ID_i = ID_A$ then ξ choses $a_i \in Z_q^*$ and sets $Y_i = a_i P$ and computes $x_i = l_{1i} d_i + a_i mod q$
   ii. If $ID_i \neq ID_A$ quit the protocol.
   Finally ξ returns $(ID_i, R_i, m_w, Y_i, x_i)$ as the delegation on m_w with original signer's identity ID_i

(e) Delegation Verification: On receiving (Y_i, x_i) on m_w with original signer's identity ID_i, ξ recovers $(ID_i, R_i, P_{pub}, l_i)$, (m_w, Y_i, ID_i, l_{1i}) from L_1 and L_2 respectively and then performs the following.
   i. If $ID_i = ID_A$ then ξ aborts.
   ii. Otherwise ξ verifies the correctness of the delegation.

ξ verifies the correctness of the delegation (Y_i, x_i) with the equation $x_i P = h_{1,i}(R_i + h_i Ppub) + Y_i$ and outputs the result. The delegation (Y_i, x_i) is valid if ID_i, m_w have never been queried during the Extraction and Delegation Generation oracles respectively.

(3) **Forgery:** Finally, the adversary A_2 outputs (Y_i^*, x_i^*) with h_{1i}^* on m_w^* as a valid delegation with the original signer's identity ID_i From Forking lemma [43], ξ recovers another (m_w^*, R_i^*, h_{1i}^*) from L_2 - list and then replays the random oracle with the same random tape but different choice of hash value of H_2 i.e., on the same warrant m_w^*, ξ. obtains another forged delegation $(Y_{,i}, x_i)$ with h_{1i} such that $h_{1i}^* \neq h_{1i}$ and $x_i^* \neq x_i$ Therefore, (Y_i, x_i) and (Y_i^*, x_i^*) are two valid delegations on the same warrant m_w^* Therefore, the following equations hold. $x_i P = h_{1i}(Ri, h_i Ppub) + Y_i$ and $x_i^* P = h_{1i}^*(R_i + h_i^* P_{pub}) + Y_i$.

By r_i,s we now denote discrete logarithms of R_i,

P_{pab} respectively. i.e., R_i=r_iP, P_{pub}=sP, ξ solves the unknown values r_i,s from the above equations and outputs s as the solution of ECDLP. □

Theorem 2. *The proposed PF-IDBPBS-MR scheme is existentially unforgeable under the adaptive chosen message and identity attacks against the type 3 adversary A_3 in the random oracle model provided the ECDLP is intractable by any polynomial time-bounded algorithm in the elliptic curve group G.*

Proof. Suppose an adaptive chosen message and identity type 3 adversary A_3 wants to break the security of the proposed PF-IDBPBS-MR scheme, then we prove that a polynomial time-bounded adversary A_3 can solve the *ECDLP* with the help of the algorithm ξ The type 3 adversary A_3 contains the public keys of the original signer and the proxy signer and also contains the private key of the original signer. The adversary A_3 can produce a valid delegation but cannot produce a valid proxy signing key because it does not know the private key of the proxy signer. We prove that, if there exists an adversary A_3 who can produce a forged proxy blind signature, then there exists an algorithm ξ to solve an instance of *ECDLP*. Thus ξ outputs s_1 for a given random instance $(P, s_1P) \in G$ where $s_1 \in z_q^*$ To solve the *ECDLP*, ξ sets the master secret key as s_1 and master public key as P_{pu2b}=s_1 P where s_1 is unknown.The adversary A_3 and the challenger ξ interacts as defined in Game-II.

(1) **Initialization Phase:** The initialization phase made by the type 3 adversary A_3 is similar to that of the type 2 adversary A_2 described in the proof under theorem 3.

(2) **Queries Phase:** Adversary A_3 performs the oracle simulation and the algorithm ξ responds to these oracles as follows.

a. The queries on the oracles H_1, H_2 made by the type 3 adversary A_3 are similar to that of the type 2 adversary A_2 described in the proof under theorem 3.

c. Queries on Oracle H_3 $(H_3\ (ID_i, R_i, \sigma_j))$: ξ maintains an initial-empty H_3 - oracle list L_3 which contains the tuples of the form $(ID_i, R_i, \sigma_j, l_2i)$ When the adversary A_3 asks a H_3 query with the input (ID_i, R_i, σ_j) then ξ returns l_2i if there is a tuple $(ID_i, R_i, \sigma_j, l_2i)$ in L_3 Otherwise ξ chooses a random $l_2i \in Z_q^*$ and adds $(ID_i, R_i, \sigma_j, l_2i)$ to the list L_3 Finally ξ returns l_2i to the adversary A_3

d. Queries on Oracle $H_4(H_4(ID_i, t_2jP)) : \xi$ maintains an initial-empty H_4 - oracle list L_4 which contains the tuples of the form (ID_i, t_2jP, l_{3i}) When the adversary A_3 asks a H_4 query with the input (ID_i, t_2jP) then ξ returns l_{3i} if there is a tuple (ID_i, t_2jP, l_{3i}) in L_4. Otherwise ξ chooses a random $l_{3i} \in Z_q^*$ and adds (ID_i, t_2jP, l_{3i}) to the list L_4. Finally ξ returns L_3i to the adversary A_3

e. Queries on F_1, F_2: ξ maintains two separate lists F_1-list, F_2-list which are initially empty. If the queries are made earlier, then it returns the same answer. Otherwise, ξ chooses random numbers from $\{0,1\}^{l_2}$ and $\{0,1\}^{l_1}$ respectively and returns to adversary A_3, ξ stores these values in F_1-list, F_2-list respectively.

f. Key Extraction Queries: The key extraction queries made by the type 3 adversary A_3 is similar to that of the type 2 adversary A_2 described in the proof under theorem 3.

g. Delegation Generation Queries: When the adversary A_3 makes this query to ξ on the warrant m_w with the original signer's identity ID_i then ξ computes the corresponding delegation. ξ knows the private key of the original signer and therefore ξ can execute Delegation Generation queries on (ID_i, m_w) to compute the corresponding delegation (Y_i, x_i).

h. Proxy Key Generation Queries: When the adversary A_3 queries a proxy key generation of the proxy signer for m_w, then ξ computes the proxy signing key $D_{pskj} = x_i + d_j h_{1j}$ and ξ responds to the adversary with D_{pskj} (Here i represents the original signer and j represents the proxy signer).

i. Proxy Blind Signature Generation Queries: When the adversary A_3 makes this query $(ID_j, m_j), \xi$ first makes queries on H_1, H_2, H_3, H_4, F_1, F_2 oracles and recovers the tuples $(ID_i, R_i, P_{pub}, l_i)$, (m_w, Y_i, ID_i, l_{1i}), $(ID_i, R_i, \sigma_j, l_{2i})$ $(ID_i, t_{2j}, P, l_{3i})$ from L_1, L_2, L_3, L_4, F_1, F_2 lists respectively. Then ξ generates the blind signature as follows.
Chooses $y_j \in Z_q^*$ and sets $U_j = y_iP$
$\beta_j = F_1(m_j) \parallel F_2((F_1(m_j)) \oplus m_j$
t_{1j}= $(z_{2j} D_{pskj} + y_j)$
t_{2j}= $(n_1 t_{1j} + n_2 \beta_j)$
α_j= $H(ID_j, t_{2j}P)$ and $v_j = [\alpha_j \oplus \beta_j]_{10}$.

Finally ξ responds to the adversary A_3 with the blind signature $\Omega_i = (R_i, R_j, Y_i, \sigma_j, m_w, v_j)$ Clearly Ω_i is a valid blind signature since it satisfies the verification equation.

(3) Forgery: After forging a valid signature $\Omega^* = (R_i^*, R_j^*, Y^*, \sigma^*, m_w^*, v^*)$ on the message m_i^* under the identities ID_i^*, ID_j^* by the ad-

versary A_3 , ξ recovers the corresponding tuples $(ID_i, R_i, P_{pab}, l_i)$, (m_w, Y_i, ID_i, l_{1i}) , $(ID_i, R_i, \sigma_j, l_{2i})$, $(ID_i, t_{2j}P, l_{3i})$ from L_1, L_2, L_3, L_4 lists.

a. If $ID_i \neq ID^*_{s1}$ then ξ aborts.
b. If $ID_i = ID^*_{s1}$, then ξ computes the value of s_1 as follows.

Let $\Omega^{*(j)} = (R^*_i, R^*_j, Y^*, \sigma^{*(j)}, m^*_w, v^*)$ denote $\Omega^* = (R^*_i, R^*_j, Y^*, \sigma^*, m^*_w, v^*)$.

From Forking Lemma [43], if we have a replay of ξ with same random tape but different choice of H_4 the adversary A_3 will output five signatures for $j = 1, 2, 3, 4, 5$. The following equation holds $\sigma^{*(j)} = t^*_{2j}P - z_1^{*(j)}[Y + h^*_{1A}(R^*_A + h^*_A P_{pub}) + h^*_{1B}(R^*_B + h^*_B P_{pub})]$ for $j = 1, 2, 3, 4, 5$. By r_A, r_B, s_1, a, γ we now denote discrete logarithms of R_A, R_B, P_{pub},Y, σ respectively, that is $R_A = r_A P$, $R_B = r_B P$, $P_{pub} = s_1 P$, $Y = aP$,$\sigma = \gamma P$ From the above equation, we get five equations as follows $\gamma^{*(j)} = t^*_{2j} - z_1^{*(j)}[a + h^*_{1A}(r^*_A + h^*_A s_1) + h^*_{1B}(r^*_B + h^*_B s_1)]$ for $j = 1, 2, 3, 4, 5$. ξ solves the unknown values r_A, r_B, S_1, a, γ from the above five linear independent equations and outputs s_1 as the solution of ECDLP. □

5.1.2 Additional Security Requirements of the Proposed PF-IDBPBS-MR Scheme

In addition to the above unforgeability security requirement, a proxy blind signature scheme must also satisfy the following security requirements: (1) Verifiability (2) Identifiability (3)Prevention of misuse and (4) Blindness. Now, we discuss these additional security requirements of our proposed scheme.

Verifiability: From the proxy verification phase, the verifier can be convinced that the proxy signer contains the blind signature of the original signer on the warrant m_w Moreover, the warrant includes the identity information of the original signer, proxy signer and the limit of delegated signing capacity etc. Consequently, the verifier can be convinced of the original signer's agreement on the signed message. Accordingly, the scheme satisfies the security requirement of verifiability.

Identifiability: The verification of a valid proxy blind signature needs the public key of the proxy signer, and successively proves that the blind signature has been produced by the proxy signer. It includes the warrant m_w in a valid proxy blind signature, so that anyone can determine the identity of the corresponding proxy signer from the warrant m_w Accordingly, the scheme satisfies the security requirement of identifiability.

Prevention of Misuse: Using the warrant m_w we have determined the limit of the delegated signing capability on the warrant m_w in our proxy blind signature scheme. Hence, the proxy signer cannot sign any messages which are not authorized by the original signer. Accordingly, the scheme satisfies the security requirement of prevention of misuse.

Blindness: The blindness of the proposed scheme can be proved using the following theorem-3.

Theorem 3. *The proposed scheme satisfies the blindness property.*

Proof. Let $(R_A, R_B, Y, \sigma, m_w, v)$ be one of the two signatures given to adversary Adv. Let (U, z_2, t_1) be the data exchanged during one of the signature issuing schemes in the view ofAdv . It is enough to show that there exists two random factors (n_1, n_2) that map (U, z_2, t_1) to $(R_A, R_B, Y, \sigma, m_w, v)$ From the description of the scheme, we know the following equations must hold .

$$\sigma = n_1 U + n_2 \beta P \tag{1}$$

$$z_2 = n_1^{-1} z_1 mod q \tag{2}$$

$$t_2 = (n_1 t_1 + n_2 \beta) mod q \tag{3}$$

From equations (2), (3), we can get that $n_1 = z_1 z_2^{-1} mod q$ and $n_2\beta = t_2 - n_1 t_1 mod q$. It is obvious that $n_1, n_2 \in Z^*_q$ uniquely exist and next we show that $n_1, n_2 \in Z^*_q$ satisfy equation (1) also. Thus (U, z_2, t_1) and $(R_A, R_B, Y, \sigma, m_w, v)$ have exactly the same relation defined by the signature issuing protocol. Such n_1, n_2 always exist regardless of the values of (U, z_2, t_1) and $(R_A, R_B, Y, \sigma, m_w, v)$ Therefore, even an infinitely powerful Adv outputs a correct value b' with probability exactly $\frac{1}{2}$ So the proposed scheme is unconditionally blind. □

5.2 Efficiency Analysis of the Proposed PF-IDBPBS-MR Scheme

In this section, we analyze the performance of our PF-IDBPBS-MR scheme. We compare our scheme with the relevant schemes [23,28,32,36,37] both in terms of computation and communication (signature length) point of view.

5.2.1 Computational Efficiency

To evaluate the performance of our proposed scheme, different cryptographic operations and their notations, presented in Table 3 are considered. The conversions of these cryptographic operations have been taken from the experimental results [44-47].

Table 3. Notations and descriptions of different cryptographic operations and their conversions

Notation	Deseriptions(Time required to perform)
T_{ML}	Modular multiplication operation
T_{EM}	Elliptic curve point multiplication (Scalar multiplication in G_1) : $T_{EM} = 29T_{ML}$
T_{BP}	Bilinear pairing operation in G_2 : $T_{BP} = 87T_{ML}$
T_{PX}	Pairing-based exponentiation operation in G_2 : $T_{PX} = 43.5T_{ML}$
T_{EX}	Modular exponentiation operation in Z_q^*: $T_{EX} = 240T_{ML}$
T_{IN}	Modular inversion operation in Z_q^*: $T_{IN} = 11.6T_{ML}$
T_{MTP}	Map-to-point (hash function): $T_{MPT} = T_{EM} = 29T_{ML}$
T_{PA}	Addition of two elliptic curve points (point addition in G_1): $T_{PA} = 0.12T_{ML}$

Table-4 provides the comparison of our proposed PF-IDBPBS-MR scheme with the existing proxy blind signature schemes in terms of computational point of view. Since the proposed scheme is pairing free, no pairing operations are involved. To evaluate the computational efficiency, we consider the delegation generation cost, delegation verification cost, proxy signature generation cost, proxy signature verification cost and the total cost. In our PF-IDBPBS-MR scheme, for delegation generation, the original signer needs to compute one scalar multiplication in G_1 For delegation verification, the proxy signer needs to compute three scalar multiplications in G_1 and two point additions in G_1 For proxy signature generation, the proxy signer needs to compute four scalar multiplications in G_1 and one point addition in G_1 For proxy signature verification, the verifier needs to compute five scalar multiplications in G_1 and five point additions in G_1 Hence, the total computation cost of our scheme is $377.96T_{ML}$ Similarly, we computed the delegation generation cost, delegation verification cost, proxy signature generation cost, proxy signature verification cost and the total cost for all the existing proxy blind signature schemes [23,28,32,36,37] and are presented in table-4. Also we present the comparison of the computational cost graphically in Fig-2.

From Table-4 the total computational cost of our proposed scheme is 377.96 T_{ML} which is 34.89]% less than Zhang *et al.* scheme [23], 50.85% less than Pan *et al.* scheme [28], 7.18% less than Tan scheme [32], 27.81% less than Padhye and Tiwari scheme [36] and 52.46% less than Sarde & Banerjee scheme [37]. Obviously, the computation cost of our PF-IDBPBS-MR scheme is much less and hence, the scheme is computationally more efficient compared to the existing proxy blind signature schemes [23,28,32,36,37].

5.2.2 Communicational Efficiency

Table-5 provides the comparison of our proposed PF-IDBPBS-MR scheme with the existing proxy blind signature schemes in terms of communicational point of view. Our proposed scheme is constructed on ECC. In bilinear pairing, to achieve a security level of 80 bits, we consider $\hat{e} : G_1 \times G_2 \rightarrow G_T$ where G_1 is an additive group which is generated by $\hat{P}$ with the order $\hat{q}$ on the super singular elliptic curve $\hat{E} : y^2 = x^3 + x \ mod \ \hat{p}$ with embedding degree 2. Here p consists of 512 bit prime number and q is of 160 bit solinas prime number. In ECC, to achieve the same 80 bit security level, we consider$G$ as an additive cyclic group generated by a point P on a non-singular elliptic curve $E : y^2 = x^3 + ax + b \ mod \ p$ and its order is q where p, q are prime numbers of 160 bit each and $a.b \in Z_q^*$. Hence the size of p is 512 bits (i.e. 64 bytes) and the size of p is 160 bits (i.e. 20 bytes). Therefore, the size of elements in G_1 is $512 \times 2 = 1024$ bits and the size of elements in G is $160 \times 2 = 320$ bits. Also the size of the elements in Z_q^* is 160 bits.

To evaluate the communication cost, we consider the length of the signature. Our proposed PF-IDBPBS-MR scheme has signature length $4\,|G| + |q| = 4(320) + 160 = 1440 \ bits \ = 180 \ bytes$. Similarly, we computed the signature length for all other existing proxy blind signature schemes and presented in Table 5. Also,we present these communication costs graphically in Fig-3 .

From Table-5 and Fig-3 we can observe that the signature length of our PF-IDBPBS-MR scheme is much less and hence, the scheme is more efficient compared to the existing proxy blind signature schemes [23,28,32,36,37] in terms of communication point of view.

From the above discussion, it is clear that the proposed PF-IDBPBS-MR scheme is much more efficient compared to the existing proxy blind signature schemes both in terms of computation and communication.

6 Conclusion

In this article, we have presented a pairing-free Identity-based proxy blind signature scheme with message recovery by integrating the features and advantages of both proxy signature and blind signature. The proxy blind signature scheme permits a proxy signer to produce a blind signature on behalf of an original signer. The proposed scheme is significantly useful, particularly in electronic commerce

Table 4. Comparison of computational efficiency of our scheme with other schemes

Scheme	Delegation Generation cost	Delegation verification cost	Proxy Blind Signature cost	Proxy Blind Verification Cost	Total Cost
Zhang *et al.* (2003) [23]	$1T_{EM}$	$2T_{BP} + 1T_{EM} + 1T_{PA}$	$5T_{EM} + 1T_{PA}$	$2T_{BP} + T_{EM} + 2T_{PA}$	$580.48T_{ML}$
Pan *et al.* (2008) [28]	$2T_{PX} + 1T_{EM}$	$2T_{PX} + 1T_{BP} + 2T_{EM} + 2T_{PA}$	$2T_{PX} + 2T_{EM}$	$3T_{PX} 1T_{BP} + 2T_{EM} + 2T_{PA}$	$768.98T_{ML}$
Tan(2013) [32]	$1T_{EM}$	$3T_{EM} + 2T_{PA}$	$8T_{EM} + 7T_{PA}$	$2T_{EM} + 1T_{PA}$	$407.20T_{ML}$
Padhye and Tiwari(2006)[36]	$1T_{EM}$	$3T_{EM} + 2T_{PA}$	$8T_{EM} + 6T_{PA}$	$6T_{EM} + 5T_{PA}$	$523.56T_{ML}$
Sarde and Banerjee (2017) [37]	$1T_{BP} + 1T_{PX} + 4T_{EM} + 2T_{PA}$	$1T_{BP} + 1T_{PX}$	$3T_{PX} + 2T_{EM} + 2T_{PA}$	$1T_{BP} + 3T_{PX} + 1T_{IN}$	$795.08T_{ML}$
Proposed Scheme	$1T_{EM}$	$3T_{EM} + 2T_{PA}$	$4T_{EM} + 1T_{PA}$	$5T_{EM} + 5T_{PA}$	$377.96T_{ML}$

Table 5. Comparison of communicational efficiency of our scheme with other schemes

Scheme	Message Recovery	Signature length	In bytes
Zhang *et al.* (2003)[23]	×	$2\|G_2\| + \|m\|$	356 bytes
Pan *et al.* (2008)[28]	×	$\|G_1\| + 2\|G_2\| + \|m\|$	484 bytes
Tan(2013) [32]	×	$4\|G\| + 2\|q\| + \|m\|$	300 bytes
Padhye and Tiwari(2006)[36]	×	$4\|G\| + \|q\| + \|m\|$	280 bytes
Sarde and Banerjee (2017) [37]	×	$\|G_1\| + 2\|q\| + \|m\|$	268 bytes
Proposed Scheme	✓	$4\|G\| + \|q\|$	180 bytes

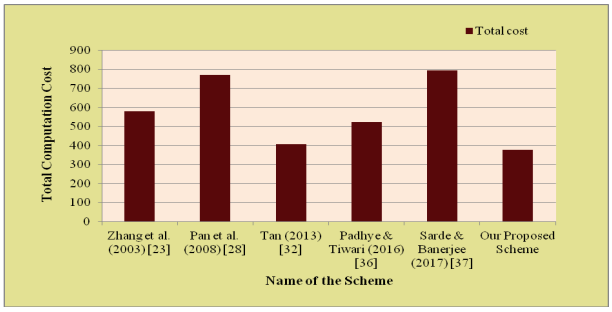

Figure 2. Graphical representation of total computation cost

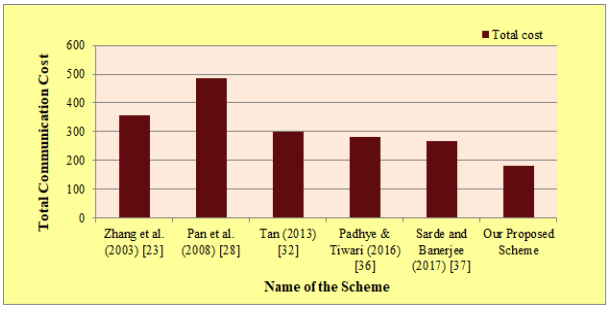

Figure 3. Graphical representation of total communication cost

applications, thereby inspiring the need of delegation of signing capacity along with user anonymity. Moreover, due to the message recovery property, our proposed scheme is designed for low bandwidth communication channels. The proposed scheme is proven secure against different types of adversaries in the random oracle model under the hardness assumption of the elliptic curve discrete logarithm problem. The efficiency analysis indicates that compared to the well-known existing proxy blind signature schemes, our proposed scheme is very efficient regarding computation and communication.

References

[1] A. Shamir,*Identity-based Cryptosystems and Signature Schemes,* Crypto '84, Springer-Verlag, LNCS 196 (1985), 47-53.

[2] K. Nyberg and R. A. Rueppel, *New Signature Scheme based on the DSA giving Message Recovery,*zs qs In Proc. of 1st ACM Conference on Communication and Computer Security, Virginia, USA, (1993), 58-61.

[3] D. Chaum, *Blind Signatures for Untraceable Payments,* In Advances in Cryptology-Proceedings of CRYPTO'82, Springer-Verlag, New York, (1983), 199-203.

[4] A. Fujioka, T. Okamoto and K. Ohta, *A Practical Secret Voting Scheme for Large Scale Elections,* Advances in Cryptology-AUSCRYPT'92, Lecture Notes in Computer Science 718 (1992), 244-251.

[5] D. Chaum, A. Fiat and M. Naor, *Untraceable*

Electronic Cash, In Advances in Cryptology-CRYPTO'88, Santa Barbara, CA,USA, 21-25August, Lecture Notes in Computer Science 403, Springer: Berlin, (1988), 319-327.

[6] B. C. Neuman, *Proxy-based Authorization and Accounting for Distributed System,* In: Proc. of the 13th International Conference on Distributed Computing Systems, (1993), 283-291.

[7] I. Foster, C. Kesselman, G. Tsudik and S. Tuecke, *A Security Architecture for Computational Grids,* Proceedings of the 5th ACM Conference on Computers and Communication Security, (1998), 83-92.

[8] M. Mambo, K. Usuda and E. Okamoto, *Proxy Signatures for Delegating Signing Operation,* In: 3rd ACM Conference on Computer and Communications Security (CCS'96), New York: ACM Press, (1996), 48-57.

[9] S. Kim, S. Park and D.Won, *Proxy Signatures, revisited,* Proc. of ICICS 97, Springer-Verlag, LNCS 1334 (1997), 223-232.

[10] B. Lee, H. Kim and K. Kim, *Secure Mobile Agent using Strong Non-designated Proxy Signature.* Information Security and Privacy (ACISP'01), Springer-Verlag, LNCS 2119 (2001), 474-486.

[11] T. Okamoto, M. Tada and E. Okamoto, *Extended Proxy Signatures for Smart Cards,* Information Security Workshop (ISW'99), Springer-Verlag, LNCS 1729 (1999), 247-258.

[12] W. D. Lin and J. K. Jan, *A Security Personal Learning Tools Using a Proxy Blind Signature Scheme,* Proc. of Int. Conf. on Chinese Language Computing, Illinois, USA, (2000), 273-277.

[13] Z. Tan, Z. Liu and C. Tang, *Digital Proxy Blind Signature Schemes Based on DLP And ECDLP*, MM Research Preprints, No. 21, MMRC, AMSS, Academia, Sinica, Beijing, (2002), 212-217.

[14] S. Lal and A. K. Awasthi, *Proxy Blind Signature Scheme,* Cryptology ePrint Archive, Report 2003/072, available at http://eprint.iacr.org/2003/072/.

[15] J. Li and S. Wang, *New Efficient Proxy Blind Signature Scheme Using Verifiable Self-certified Public Key,* IJ Network Security 4(2) (2007), 193-200.

[16] C. Qi and Y. Wang, *An Improved Proxy Blind Signature Scheme Based on Factoring and ECDLP,* Int. Conf. on Computational Intelligence and Software Engineering, IEEE (2009), 1-4.

[17] J. Su and J. Liu, *A Proxy Blind Signature Scheme Based on DLP,* Int. Conf. on Internet Technology and Applications, IEEE, (2010), 1-4.

[18] G. K. Verma, *B.B. Singh and Harendra Singh, Provably Secure Certificate-based Proxy Blind Signature Scheme from Pairings,* Information Sciences, vol. 468 (2018), 1-13.

[19] L. He, J. Ma, R. Mo and D. Wei, *Designated Verifier Proxy Blind Signature Scheme for Unmanned Aerial Vehicle Network Based on Mobile Edge Computing,* Security and Communication Networks, vol. 2019, Article ID 8583130, 12 pages (2019) https://doi.org/10.1155/2019/8583130.

[20] H. Zhu, Y. Tan, L. Zhu, Q. Zhang and Y. Li, *An Efficient Identity-Based Proxy Blind Signature for Semioffline Services. Wireless Comm. and Mobile Computing,* (2018), 1-9, 10.1155/2018/5401890.2018.

[21] G. Kumar, B.B. Singh and H. Singh, (2018). *Provably Secure Certificate-Based Proxy Blind Signature Scheme from Pairings.* Information Sciences, (2018), 468. 10.1016/j.ins.2018.08.031.

[22] Z. Tan, *An E-cash Scheme Based on Proxy Blind Signature from Bilinear Pairings,* Journal of Computers 5(11) (2010), 1638-1645.

[23] F. Zhang, R. Safavi-Naini and C. Y. Lin, *New Proxy Signature, Proxy Blind Signature and Proxy Ring Signature Schemes from Bilinear Pairings,* IACR Cryptology ePrint Archive, (2003), 104.

[24] R. A. Sahu and S. Padhye, *ID-based Signature Scheme from Bilinear Pairings: A survey. Front.* Electr. Electron. Eng. 6 (2011), 487-500.

[25] J. He, C. Qi and F. Sun, *A New Identity-based Proxy Blind Signature Scheme,* IEEE, Int. Conf. on Information Science and Technology, (2012).

[26] P. Heng, K. Ke and C. Gu, *Efficient ID-based Proxy Blind Signature Schemes from Pairings,* Int. Conf. on Computational Intelligence and Security, IEEE, (2008), 390-393.

[27] B. Majhi, D.K. Sahu and R.N. Subudhi, *An Efficient ID-based Proxy Signature, Proxy Blind Signature and Proxy Partial Blind Signature,* Int. Conf. on Information Technology ICIT'08 , IEEE, (2008), 19-23.

[28] H. Pan, K. Ke and C. Gu, *Efficient ID-based Proxy Blind Signature Schemes from Pairings,* Int. Conf. on Computational Intelligence and Security, (2008), DOI 10.1109/CIS.2008.101.

[29] S. Rawal and S. Padhye, *Cryptanalysis of ID based Proxy-Blind signature scheme over lattice,* ICT Express (2019), https://doi.org/10.1016/j.icte.2019.05.001.

[30] X. Zhang, *Two Improved ID-based Proxy Blind Signatures,* J. Comp. En. 35(3) (2009), 15-17.

[31] S. Pradhan and R. K. Mohapatra, *Proxy Blind Signature Scheme Based on ECDLP,* Int. J. of Engineering Science & Technology 3(3) (2011), 2244-2248.

[32] Z. Tan, *Efficient Pairing-free Provably Secure Identity-based Proxy Blind Signature Scheme.* Security and Communication Networks 6 (2013),

593-601.
[33] S. Prabhadevi and A.M. Natarajan, *Utilization of ID-based Proxy Blind Signature Based on ECDLP in Secure Vehicular Communications,* Int. J. of Engineering and Innovative Technology (IJEIT) 3(5) (2013).
[34] H. Chen, J. Chen, G. Cai and A. Luo, *Untraceability Analysis of Two ID-based Proxy Blind Signature from Bilinear Pairings,* Res. J. of Applied Sciences, Engineering and Technology 5(3) (2013), 1054-1058.
[35] M. K. Chande, *An Improved Proxy Blind Signature Scheme Based on ECDLP,* Malaya Journal of Matematik 2(3) (2014), 228-235.
[36] S. Padhye and N. Tiwari, *An Efficient ID-based Proxy Blind Signature with Pairing-free Realization,* 3rd Int. Conf. on Innovative Engg.Tech. (ICIET'2016), Bangkok, Thailand, (2016).
[37] P. Sarde and A. Banerjee, *A Secure ID-based Blind and Proxy Blind Signature Scheme from Bilinear Pairings,* Journal of Applied Security Research 12(2) (2017), 276-286.
[38] L. Diao, J. Gu and I. L. Yen, *A New Proxy Blind Signature Scheme with Message Recovery.* Information Technology Journal 12(21) (2013), 6159-6163.
[39] M. Abe and T. Okamoto, *A Signature Scheme with Message Recovery as Secure as Discrete Logarithm,* In Advances in Cryptology-ASIACRYPT99, Springer, LNCS 1716 (1999), 378-389.
[40] N. Koblitz, *Elliptic Curve Cryptosystem,* Journal of Mathematics of Computation 48(177) (1987), 203-209.
[41] V. S. Miller, *Use of elliptic curves in cryptography,* In Proceeding on Advances in cryptology-CRYPTO 85, Springer-Verlag, New York, LNCS 218, (1985), 417-426.
[42] X. Huang, W. Susilo,V Y. Mu and W. Wu, *Proxy Signature without Random Oracles,* Int. Conf. on Mobile Ad Hoc and Sensor Networks, Springer-Verlag, Berlin, Germany, 4325 (2006), 473-484.
[43] D. Pointcheval and J. Stern, *Security Arguments for Digital Signatures and Blind Signatures,* Journal of Cryptology, Springer-Verlag, 13 (3) (2000), 361-396.
[44] P. Barreto, H.Y. Kim, B. Lynn and M. Scott, *Efficient Algorithms for Pairing Based Cryptosystems,* Annual Int. Cryptology Conf. CRYPTO 2002: Advances in Cryptology, Springer, Berlin, Heidelberg, LNCS 2442 (2002), 354-369.
[45] X. Cao, W. Kou and X. Du, *A Pairing-free Identity Based Authenticated Key Agreement Protocol with Minimal Message Exchanges,* Information Sciences, 180 (15) (2010), 2895-2903.
[46] S.Y. Tan, S. H. Heng and B. M. Goi, *Java Implementation for Pairing-based Cryptosystems,* Int. Conf. on Computational Science and Its Applications ICCSA '10, Springer, Berlin, LNCS 6019 (2010), 188-198). Heidelberg.
[47] Shamus Software Ltd. Miracl Library Available: http:certivox.org/display/EXT/MIRACL.

Salome James received her M.Sc. (Mathematics) from Nagarjuna University and M.Phil. (Cryptography) from Andhra University, A.P., India. She is currently pursuing Ph.D. in the area of Cryptography at Andhra University, Visakhapatnam, A.P., India. Her research interests include Number Theory and Elliptic Curve Cryptography.

Gowri Thumbur received the B.Tech. degree in electronics and communication engineering from Nagarjuna University, and M. Tech degree from JNTU Anantapur, A.P, India. She received Ph.D. degree from JNTU-Kakinada, A.P, India. She is Senior Member in IEEE and life member in ISSS. She is currently working in the Department of Electronics and Communication Engineering, GITAM Institute of Technology, GITAM University, Visakhapatnam, A.P, India. Her research interests include Signal Processing, Digital Information Systems and Computer Electronics, Digital Image Processing and Information Security.

P. Vasudeva Reddy received M.Sc. and Ph.D. (Cryptography) degrees from S.V. University and received M.Tech. (CSTNetworks) from Andhra University, India. He is currently working as Professor in the Department of Engineering Mathematics at Andhra University, Visakhapatnam, India. His field of interest includes Algebra & Number theory Applications, Cryptography. He has several publications in reputed national and international journals. He is a reviewer adversary board member for various journals. He is a life member of Indian Mathematical Society (IMS) and Cryptology Research Society of India (CRSI).