Available online at http://scik.org
J. Math. Comput. Sci. 11 (2021), No. 1, 24-43
https://doi.org/10.28919/jmcs/4950
ISSN: 1927-5307

AN EFFICIENT PAIRING-FREE CERTIFICATELESS SIGNCRYPTION SCHEME WITH PUBLIC VERIFIABILITY

G. SWAPNA[1], K.A. AJMATH[2], GOWRI THUMBUR[3,*]

[1]Department of Mathematics, GRIET, Hyderabad, T.S, India

[2]Department of BS&H, Sree Vidyanikethan Engineering College, Tirupati, A.P, India

[3]Department of ECE, GIT, GITAM University, Visakhapatnam, A.P, India

Abstract: Signcryption is a cryptographic technique that provides both confidentiality and authenticity of data during public transmission in many modern applications like Internet -of -Things (IoT), Wireless Sensor Networks (WSNs) etc. The functionalities of encryption and Signature can achieve simultaneously through the signcryption with lower computational cost and communication overheads than those of traditional sign-then-encrypt approach. Many signcryption schemes have been constructed by various researchers in different cryptographic frameworks. Certificateless cryptography is one of the recent public key cryptography which eliminates the key escrow problem and complex certificate management problemsin identity based cryptography and traditional public key cryptography respectively. Providing the security and efficiency in many modern applications including IoT and WSNs is a crucial task. In this paper, we proposed new signcryption scheme in certificateless cryptography. This scheme supports the property of public verifiability and is secure against various types of adversaries in the random oracle model with the assumption that the Computational Diffie-Hellman Problem (CDHP) and the Elliptic Curve Discrete Logarithmic Problems (ECDLP) are hard. Due to pairing-free environment the proposed scheme is computationally more efficient than the existing public verifiable signcryption schemes.

Key words: certificateless signcryption; public verifiability; CDHP; ECDLP.

2010 AMS Subject Classification: 68P25.

---

[*]Corresponding author

E-mail address: gowri3478@ieee.org

Received August 16, 2020

1. INTRODUCTION

Nowadays, secure communication between smart devices or entities is most important in manymodern applications such as Internet of Things (IoT), Wireless medical sensor networks etc. From the security point of view, the authenticity and confidentiality of data are the two critical security requirements [1] in many of these applications. Cryptography provides a solution to theses security requirements and there are several research works been carried out[2,3,4]. Signcryption is a cryptographic technique that enables both confidentiality and authenticity of data during public transmission. Signcryption can provide both the functions of public key encryption and digital signature in a logical single step at a significantly lower cost compared to traditional signature then-encryption methods. Therefore, the design of efficient and secure signcryption schemes is necessary in many applications.

Many signcryption schemes have been proposed in PKI and ID-based settings [5, 6]. However, traditional public key cryptosystems based on the public key infrastructure (PKI) are not suitable for many applications because of the computing loads in authenticating long random public keys. While PKI has widely been accepted in e-commerce applications over the Internet in recent years, a trustworthy certificate authority (CA) is required to issue a certificate for a public key and its holder's identity such that this relation is guaranteed with CA's digital signature. However, for many application devices with low computing capacity and limited storage, the computation and storage costs incurred by PKI is unfavorable. To eliminate the problems in traditional PKC, Shamir [7]introduced the concept of Identity based cryptography (IBC). In IBC, the public key of a user is simply his/her identity information and the corresponding private key is generated by a trusted private key generator (PKG) that holds the system's master secret key. Many cryptographic schemes have been proposed in ID-based frame works to protect data transmitted over network systems. However, the key escrow problem is the inherent problem in IBC schemes. To overcome this problem Al-Riyami and Paterson [8] introduced new paradigm of Certificate Less Public key Cryptography (CL-PKC) in 2003. In this new paradigm, the private key generator (PKG) generates partial public and private keys for usersby using their unique identities and users set their secret value by randomly choosing and also set their full private key and public key. Therefore without involvement of PKG, user generates full private key and public key from their identities and random values. Thus, CL-PKC can successfully resolve the key escrow problem and also eliminate the certificate management problem in traditional PKI.

In 1997, Zhang [9] proposed the first signcryption scheme by combining the digital signature and encryption functionalities simultaneously in a single logical step. Since then many signcryption schemes have been proposed in PKI [5], ID-based [6] and CL-based settings [7, 10]. But many of these schemes are designed using bilinear pairings over elliptic curve cryptography. The computational cost of bilinear pairings is approximately 20 times higher than that of scalar multiplication [11, 12]. Hence the design of signcryption schemes without using pairings is desirable. In this direction, Barreto et al. [13] proposed the first CLSC scheme from(without) bilinear pairingsin 2008.Later on many works came on security and efficiency of CLSC schemes without bilinear pairings. In 2010, Selvi et al. [14] proved that the Barreto et al.[13] scheme is not secure against type-I adversary and proposed a signcryption scheme without pairings. Later Xie etal. [15] proposed a certificate less signcryption scheme. To improve the performance of Xie etal. [15], Li etal. [16],Liu etal. [17] and Jing etal. [18] proposed CLSC schemes without pairings. ButHe etal.[19], Shi etal. [20] pointed the insecurity ofLiu et al. [17], Jing etal. [18] against type-I adversary. But Shi et al. [20] scheme also not secure and it is proved by Liling Cao et al. [21]. Huang etal. [22] described that Zhu etal. [23] scheme is not secure against CCA attack and key replacementattack. Xi-Jun etal. [24] and Zhang etal. [25] proved that Yu etal. [26] CLSC scheme doesn’t provides authenticity and confidentiality. Later F. Li et al. [3], Lui Cui et al. [27] and Gao etal. [28] are also proposed CLSC schemes without pairings.Thus many existing signcryption schemes are not efficient and also not secure and most of these schemes are not public verifiable.

The public verifiability property allows any third party to verify the validity of cipher text without knowing the message and receivers private key. This property plays a vital role in practical applications like access control systems, Mobile ad hoc network etc. However, F. Li et al. [3], Lui Cui et al. [27] are the only two signcryption schemes, which satisfies the public verifiable property, appeared in the literature. But the schemes F. Li et al. [3], Lui Cui et al. [27] are not computationally efficient and hence are not suitable for practical applications where the computational power is limited such as WSNs, Mobile computing etc.

In order to improve the computational efficiency, in this paper, we propose a new certificateless based signcryption scheme without using bilinear pairings(PF-PVCLSC). This scheme provides confidentiality, authenticity and public verifiability and also secure against Type-I and Type-II adversary with less computation cost. The proposed signcryption scheme greatly improve the computation cost than the existing schemes.

The rest of this paper is organized as follows. Preliminaries, syntax and security model of our proposed scheme is presented in section 2 and section 3 respectively. The proposed PF-PVCLSC scheme is described in section 4. We discussed analysis and Conclusion in section 5 and 6 respectively.

2. PRELIMINARIES

In this section we present preliminaries related to proposed scheme such as elliptic curve cryptography and computational problems.

2.1. Elliptic Curve Cryptography

Elliptic curve cryptography (ECC) plays a major role in the modern public key cryptography with respect to computation, communication overheads 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: $y^2 \bmod q = (x^3 + ax + b) \bmod p$ with $a, b \in F_q$ and $(4a^3 + 27b^2) \bmod q \neq 0$. The additive elliptic curve group G is defined as $G = \{(x, y) : x, y \in F_q\}$ 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 $1 - 2\sqrt{q} \le O(E(F_q)) \le q + 1$. The scalar multiplication on the cyclic group G_q defined as $k.P = P + P + P + ----(k\ times)$. Here $P \in G$ is the generator of order n.

2.2.Computational Problems

Definition 1: Computational Diffie-Hellman Problem (CDHP): For a given (P, aP, bP) the CDHP is to compute abP where $a, b \in Z_q^*$and P be the generator of anadditive cyclic group G.

Definition 2: Elliptic Curve Discrete Logarithm Problem (ECDLP):For a given $(P, \alpha P)$, the ECDLP is to computeαP, where $\alpha \in Z_q^*$and P be the generator of the additivecyclic group G.

3. SYNTAX AND SECURITY MODEL OF THE PROPOSED PF-PVCLSC SCHEME

In this section, we present the syntax and security model for our proposed scheme.

3.1.Syntax

The proposed PF-PVCLSC scheme consists of the following six polynomial time algorithms i.e.Setup, Partial Key Generator, Set Private and Public Keys, Signcryption and Unsigncryption and Public Verifiability. The algorithms are described as follows.

- **Setup:**Taking the security parameter k as input, this algorithm is executed by the KGC to generate the system parameters *params* and master key.
- **Partial Key Generator:** This algorithm is performed by the KGC to create the partial private key and partial public key of the user by taking user's identity and master key as inputs.
- **Set Private and Public Keys:** This algorithm is performed by the user. User creates his own secret key by randomly choosing value and sets his full private key and public key.
- **Signcryption:** This algorithm is implemented by user to create signcryption text by taking message, sender's private key(r_s, d_s), public key(X_s, R_s), receiver's public keys (X_R, R_R) and *params* as inputs.
- **Unsigncryption:** This algorithm is run by receiver to recover the message by taking*params*, receiver's private key (r_R, d_R) and public key(X_R, R_R), sender's public key (X_s, R_s)as inputs.
- **Public Verifiability:**This algorithm is run by any third party to verify the validity of signcryption text by taking *params*, signcryption text, and public keys of sender and receiver as inputs.

3.2.Security Model

In this section, we present the security model of the proposed PF-PVCLSC scheme, namely the confidentiality and unforgeability against the following two types of adversaries [2]. The capabilities of adversaries are mentioned as follows:

Type-I Adversary (A_1): The adversary A_1 is not accessible the master key, but he can replace the public keys at his will. The adversary is also called malicious user.

Type-II Adversary (A_2): The adversary A_2 is accessible to the master key, but he can't replace user's public keys. It represents a malicious KGC who generates partial private keys.

The formal security model of CLSC scheme is defined by Barbosa et al.[5]. The adversary $A(A \in \{A_1, A_2\})$ could make the following queries.

Game-I: This is a game between the challenger $\mathcal{C}$ and the adversary A_1.

Setup: Given a security parameter k, the challenger $\mathcal{C}$ runs this algorithm and outputs system parameters *params* and master key s, and $\mathcal{C}$ gives the *params* to A_1 while keeping s secret.

Query phase: In this face, adversary A_1make the following bounded number of queries.

Partial private key query: A_1 gives an ID. $\mathcal{C}$ computes partial private key and gives it toA_1.

Private Keyquery: A_1 supplies an identity ID. Then $\mathcal{C}$ computes corresponding full private key $(r_i, d_i)$and send it back toA_1. But A_1 isnot allowed to query this oracle if the $ID's$ public key has been replaced because $\mathcal{C}$ does not know the secret value x and can't provide a full private key for the user.

Request public key query: A_1 supplies an identity ID. $\mathcal{C}$ computes corresponding public key (R, X) and send it back to A_1.

Replace public key query: A_1 supplies an identity ID and a new public key (R', X'),$\mathcal{C}$ replaces the public key (R, X) with new public key (R', X'), and A_1 does not need to supply the corresponding secret value (r', x')

Signcryption query: A_1 supplies two identities(ID_s, ID_R) and a message *m*. $\mathcal{C}$ computes signcryptionσ and send it back to A_1.

Unsigncryptionquery: A_1 supplies two identities(ID_s, ID_R) and a signcryption σ. $\mathcal{C}$ computes unsigncryption, and returns m or invalid toA_1. IfID_R's public key has been replaced, we require$A_1$to supplyID_R's secret valuex_R to find unsigncryption text σ.

Challenge phase: A_1 makes two messages with equal length $\{m_0, m_1\}$ and two challenge identities $\{ID_{S^*}, ID_{R^*}\}$. $\mathcal{C}$ randomly selects $b \in \{0,1\}$, compute σ^* and returns σ^* to A_1.

Guess Stage: A_1make a polynomial bounded number of queries in find stage. At last, A_1outputs his guessb'. If$b' = b$ then A_1 wins the game. The restriction of A_1 are as follows:

- A_1 can't extract the private key for any identity if his public key has been replaced.
- A_1 can't extract the private key for ID_{R^*} at any point.
- A_1 can't extract the partial private key of ID_{R^*} if his public key has been replaced before the challenge stage.
- In the guess stage, A_1 can't make unsigncryption query on the challenge signcryption text σ^* under ID_{R^*} and ID_{S^*} unless the public key of ID_{R^*} or ID_{S^*} has been replaced after the challenge stage.

A_1's advantage is defined as $ADV_{A_1}^{IND-CLSC-CCA1} = 2\Pr[b' = b] - 1$.

Game-II: This is a game between the challenger $\mathcal{C}$ and the adversary A_2.

Setup: Given a security parameter k, the Adversary A_2 runs the setup algorithm to produce the system parameters *params* and master key s, and he gives the *Params* and master key s to challenger $\mathcal{C}$.

Find Stage: A_2 can make a polynomial bounded number of queries like in definition 3 except the partial private key extraction oracle and public key replacement oracle, because these two oracles are not needed to A_2.

Challenge Stage: A_2 makes two messages with equal length $\{m_0, m_1\}$ and two challenge identities $\{ID_{S^*}, ID_{R^*}\}$. $\mathcal{C}$ randomly selects $b \in \{0,1\}$, compute σ^* and returns σ^* to A_2.

Guess Stage: A_2 make a polynomial bounded number of queries like in find stage with the following conditions:

- A_2 can't extract the private key for ID_{R^*} at any point.
- In the guess stage, A_2 can't make unsigncryption query on the challenge signcryption text σ^* under ID_{R^*} and ID_{S^*}.

At last, A_2 outputs his guess b'. If $b' = b$ then A_2 wins the game. The advantage of A_2 in winning the game is defined as $ADV_{A_1}^{IND-CLSC-CCA2} = 2\Pr[b' = b] - 1$.

Definition3 (*Confidentiality*)**:** A certificateless signcryption scheme provides indistinguishability against adaptive chosen ciphertext attack (IND-CCA2) if polynomially bounded adversaries A_1 and A_2 have negligible advantage in winning the above Game-I and Game-II respectively.

Game-III: This is a game between the challenger $\mathcal{C}$ and the adversary A_1**.**

Setup: Given a security parameter *k*, the challenger $\mathcal{C}$ runs the setup algorithm and outputs the system parameters *params* and master key *s*. $\mathcal{C}$ gives the *params* to A_1 while keeping *s* secret.

Queries: A_1 can make a polynomial bounded number of queries like in definition 3.

Forgery: Eventually,A_1 outputs the signcryption text σ^* on message m^* with ID_{S^*} as the sender and ID_{R^*} as the receiver. A_1wins the game if unsigncryption σ^* is not valid.The restrictions of A_1 are as follows:

- A_1can't extract the private key for any identity if his public key has been replaced.
- A_1can't extract the private key for ID_{S^*} at any point.
- A_1can't extract the partial private key of ID_{S^*} if his public key has been replaced before the challenge stage.
- $\sigma^*$is not the output of a signcryption query on a message m^* with ID_{S^*}as the sender and ID_{R^*} as the receiver.

Game-IV: This is a game between the challenger $\mathcal{C}$ and the adversary A_2.

Setup: Given a security parameter *k*, the Adversary $\mathcal{C}$ runs the setup algorithm to output the system parameters as *params* and master key *s*, $\mathcal{C}$ gives the *params* and master key s to A_2.

Queries: A_2 can make a polynomial bounded number of queries like in definition 4.

Forgery: Eventually, A_2 outputs the signcryption text σ^* on message m^* with ID_{S^*} as the sender and ID_{R^*} as the receiver. A_2wins the game if unsigncryption σ^* is not valid under the restriction of A_2 are as follows:

- A_2can't extract the private key for ID_{S^*} at any point.
- $\sigma^*$is not the output of a signcryption query on a message m^* with ID_{S^*}as the sender and ID_{R^*} as the receiver.

Definition3 (Unforgeability)**:** A CL signcryption scheme is secure against an existential forgery for adaptive message attacks (EUF-ACMA) if a polynomially bounded adversaries $A_1$and A_2 with negligible advantage in winning the above Game-III and Game-IV respectively.

4. PROPOSED SCHEME (PF-PVCLSC)

As discussed in section 3, the proposed PF-PVCLSC scheme consists of the following six algorithms. The detailed functionalities of these algorithms are given below.

Setup

Given a security parameter *k*, KGC selects an additive cyclic group G of large prime order q, a generator P and three hash functions $H_1: \{0,1\}^* \times G \rightarrow Z_q^*$, $H_2: \{0,1\}^* \rightarrow Z_q^*, H_3: G \rightarrow \{0,1\}^n$, where *n* is the number of bits of the message. Then KGC selects the system's master key $s \in Z_q^*$ and computes $P_{pub} = sP$ as system public key. Publish the system parameters $(G, P, P_{pub,} H_1, H_2, H_3)$ and keeps the master key *s* secretly.

Partial Key Generation

KGC runs this algorithm with the user's identity $ID_i$for generation partial private key generation.

1. Choose $x \in Z_q^*$ and compute $X_i = x_i P$ and gives as user's Partial public key.
2. Compute $d_i = x_i + sH_{1i}(X_i, ID_i, P_{pub})$ and gives as user's Partial private key.

Set Private Key and Public key

The user randomly selects $r_i \in Z_q^*$ and compute$R_i = r_i P$. User sets his full public key as (X_i, R_i) and set his full private key as (r_i, d_i).

Signcryption

The sender runs this algorithm with the input parameters as sender's public key(X_s, R_s), sender's private key (r_s, d_s) and receiver's public key(X_R, R_R). The sender does the following for signcryption.

1. Choose $\alpha \in Z_q^*$and compute $U = \alpha P$.
2. Compute$V = \alpha\left(X_R + R_R + H_{1R} P_{pub}\right)$, and $C = m \oplus h_3(V)$.
3. Compute $h_2 = H_2(ID_s \parallel ID_R \parallel C \parallel U \parallel R_s \parallel R_R \parallel X_s \parallel X_R)$.
4. Compute $t = \frac{\alpha}{r_s h_2 + d_s}$.

The signcryption text on the message *m* is$\sigma = (U, C, t)$.

Unsigncryption

The receiver runs this algorithm with the input parameters as sender's public key(X_s, R_s), receiver's private key (x_R, d_R) and receiver's public key(X_R, R_R). The receiver does the following.

1. Compute $h_2 = H_2(ID_s \parallel ID_R \parallel C \parallel U \parallel R_s \parallel R_R \parallel X_s \parallel X_R)$.
2. Compute $t(h_2 R_s + X_s + H_{1s} P_{pub}) = U'$.
3. Compute $h'_2 = H_2(ID_s \parallel ID_R \parallel C \parallel U' \parallel R_s \parallel R_R \parallel X_s \parallel X_R)$.
4. If $h_2 = h'_2$, then accepts the message and retrieves the message m as $m = C \oplus h_3(V')$, where $V' = (r_R + d_R)U$.

Public Verifiability

In case of necessary, any third party can verify the signcryption text without having any information about original message and receiver's private key. In our PF-PVCLSC, any third party can verify that $h'_2 = h_2$, where $h'_2 = H_2(ID_s \parallel ID_R \parallel C \parallel U' \parallel R_s \parallel R_R \parallel X_s \parallel X_R)$ and $h_2 = H_2(ID_s \parallel ID_R \parallel C \parallel U \parallel R_s \parallel R_R \parallel X_s \parallel X_R)$.

5. Analysis of the Proposed PF-PVCLSC Scheme

In this section, we present proof of correctness of the proposed scheme and theoretically prove that our scheme is secure based on CDHP and ECDLP in security analysis. Finally, we compare our scheme with existing schemes in performance analysis.

5.1. Proof of correction

Here we verify some mathematical correctness of the elements, which is used in our scheme.

$$V' = (r_R + d_R)U = \left(r_R + x_R + sH_{1R}\left(X_R, ID_R, P_{pub}\right)\right)\alpha P$$

$$= \left(r_R P + x_R P + sPH_{1R}\left(X_R, ID_R, P_{pub}\right)\right)\alpha$$

$$= \left(R_R + X_R + P_{pub} H_{1R}\left(X_R, ID_R, P_{pub}\right)\right)\alpha = V.$$

$$U' = t(h_2 R_s + X_s + H_{1s} P_{pub}) = \frac{\alpha}{r_s h_2 + d_s}(h_2 r_s + x_s + sH_{1s})P$$

$$= \frac{\alpha}{r_s h_2 + d_s}(h_2 x_s + d_s)P = \alpha P = U.$$

$h'_2 = h_2$ if and only if $U' = U$.

5.2. Security Analysis

In this section, we present the security analysis of the proposed scheme in the random oracle model based on CDHP and ECDLP are hard.

Theorem 1 (Confidentiality against adversary A_1): In the random oracle model, the proposed PF-PVCLSC scheme is secure against the adversary A_1 with the assumption that the CDHP is hard.

Proof: let us consider an adversary A_1 who wants to break our PF-PVCLSC scheme. Here we construct an algorithm $\mathcal{C}$ that A_1 uses to solve CDH problem. The algorithm $\mathcal{C}$ wants to compute abP as the solution of CDH problem from the instance (P, aP, bP). To track the oracle models $H_1, H_2, H_3,$ partial key generation, private key generation, public key generation, signcryption and unsigncryption, $\mathcal{C}$ maintains hash lists $L_1, L_2, L_3, L_d, L_{pk}, L_{pub}, L_{sc}, L_{usc}$ respectively. Additionally, $\mathcal{C}$ maintains one more hash list L_{rec} to store the parameters of challenging users. At the beginning stage, each list is empty.

Setup: Using the input parameters k, $\mathcal{C}$ complete the setup algorithm and publish the parameters as $(G, q, P, P_{Pub}, H_1, H_2, H_3)$ to A_1. After this, $\mathcal{C}$ performs all the algorithms which are mentioned in the original scheme and furnish responses to the adversary $A_1's$ queries.

The adversary A_1 do the following queries.

$\boldsymbol{H_1 - query}$: When $\mathcal{C}$ obtain the query $H_1(X, ID, P_{pub})$ from A_1, if (X, ID, P_{pub}, h_1) exists in the list L_1, $\mathcal{C}$ returns, h_1 to A_1. Otherwise picks a random $h_1 \in Z_q^*$ and then send to A_1. Also store this new h_1 to the list L_1.

$\boldsymbol{H_2 - query}$: When A_1 makes a query on $H_2(ID_s, ID_s C, U, R_s, R_R, X_R, X_s)$, If L_2 list contains $H_2(C, U, R_s, R_R, X_R, X_s, h_2)$, $\mathcal{C}$ returns h_2 to A_1. Otherwise $\mathcal{C}$ randomlyselects $h_2 \epsilon Z_q^*$ and then send to A_1. Also store this new h_2 to the list L_2.

$\boldsymbol{H_3 - query}$: When A_1 makes a query on $H_3(V_R)$, If L_3 list contains $H_3(V_R)$, $\mathcal{C}$ returns h_3 to A_1. Otherwise $\mathcal{C}$ picks a random $h_3 \in Z_q^*$ and send to A_1. Also store this new h_3 to the list L_3.

Partial Private Key query: When $\mathcal{C}$ receives a query on (x, d, ID), first check that whether the tuple (x, d, ID) already exists in the list L_d. If it exist, $\mathcal{C}$ replies with (x, d, ID) to A_1. Otherwise,

$\mathcal{C}$ randomly choose $x, d \in Z_q^*$ and compute partial private key as $d = x + sH_1(X, ID, P_{pub})$ and send toA_1. Also add this new tuple to the list L_d.

Private Keyquery: When $\mathcal{C}$ obtain a request for (r_{ID}, ID), $\mathcal{C}$ give the response for the query(r_{ID}, ID)as (r, d, ID)if the tuple exists in the list L_{pk} to A_1. Otherwise $\mathcal{C}$ randomly choose $r \in Z_q^*$ and getting d from partial private key query, then submit (r, d, ID) to A_1 and insert the new values (r, d, ID) to the list L_{pk}.

Public key query(L_{pub}): A_1 send a request for (ID, R, X). $\mathcal{C}$ gives the reply as follows.

- If (ID, R, X) already exists in the list L_{pub}, $\mathcal{C}$ gives (R, X) to A_1.
- Otherwise $\mathcal{C}$ checks the previous list L_{pk} and L_d. If there exists a tuple in the list regarding to the identity ID,$\mathcal{C}$ can get $(r, x)$from the previous list L_{pk} and $L_d$and then find $R = rP$ and give the response as (ID, R, X) to A_1 and include these new values in the list L_{pub}.

If there is no response related to ID in the list L_{pk} andL_d. If $ID = ID^*$, $\mathcal{C}$ choose $r, x \in Z_q^*$, find $R = rP, X = xP,$ and include the tuple (R, X, ID) in the list L_{pub} and send (R, X, ID) toA_1. Also store this values in the list L_{rec} as (R, X, ID, ID^*).

If $ID \neq ID^*$, obtain (R, X) through the private key query and then send to A_1.

Replace Public Key query: When A_1 furnish the identity ID and a new public key (R', X'), $\mathcal{C}$ replace the old values (R, X) with the new values (R', X').

Signcryption query (L_{sc}): A_1 communicate $\mathcal{C}$ with$ID_s, ID_R$and a message$m$ for signcryption. First $\mathcal{C}$ checks whether (ID_s, R_s) exists in the list L_1 and give the response as follows.

- If $ID \neq ID^*$, then $\mathcal{C}$ obtain (ID_s, d_s, X_s), (ID_R, R_R, X_R) from the list L_{Pk}, L_{Pub} and runs this algorithm for the signcryption and send $\sigma = (U, C, t)$.
- If $ID = ID^*$, Then $\mathcal{C}$ terminates this algorithm.

Unsigncryptionquery (L_{sc}): When $\mathcal{C}$ obtain an unsigncryption query (ID_s, ID_R, σ) from $A_1, \mathcal{C}$ checks (ID_R, R_R) in the list L_1 and respond as follows.

- If $ID \neq ID^*$ then $\mathcal{C}$ get the tuples (ID_s, R_s, X_s),(ID_R, d_R, r_R) related to ID_s, ID_R respectively from the list L_{Pub}, L_{Pk}runs the algorithm and send the message to A_1.
- If $ID = ID^*, \mathcal{C}$ investigate for the tuple $(ID_s, X_s, R_s, ID_R, X_R, R_R, *, C)$ in the signcryption list. If he finds a tuple $(ID_s, X_s, R_s ID_R, X_R, R_R, *, C)$ then $\mathcal{C}$ returns m as the response. Otherwise, $\mathcal{C}$ returns reject as the response.

Finally, A_1 obtains a signcryption text $\sigma = (U, C, t)$ from sender and receiver whose identities are ID_s&ID_R respectively. If $ID \neq ID^*$ then $\mathcal{C}$ returns "abort" and stop the session. If $ID = ID^*, \mathcal{C}$ obtains (V, h_3) from the list L_3 and then compute $m = C \oplus h_3(V)$. At last$\mathcal{C}$ completes unsigncryption. $\mathcal{C}$ selects h_1' from $(ID_s, X_s, P_{pub}, h_1', C)$ from the list L_1, selects R_s, X_s from (ID_s, R_s, X_s) which is in the listL_{pk}, pick h_2 from $(R_s, X_s, R_R, X_R, C, U, h_2)$ the list L_2, then $\mathcal{C}$ verifies whether the equation $t\left(h_2 R_S + X_s + h_1 P_{pub}\right) = U$ is valid or not. If the equation holds, then $\mathcal{C}$ outputs m, otherwise $\mathcal{C}$ returns reject as the response.

Challenge stage: A_1 can adaptively make two different messages $m_0, m_1$with the same length and two challenge identitiesID_s&ID_R. $\mathcal{C}$ first check (ID_R, X_R) in the list L_1.

If $ID \neq ID^*$ then $\mathcal{C}$ stops the algorithm.

Otherwise, $\mathcal{C}$ makes a public query to ensure that (X_R, R_R) already exist in the list L_{Rec}. Then the algorithm $\mathcal{C}$ selects $t^*, C^* \in Z_q^*$at random and sets $U^* = sP$.$\mathcal{C}$ sends the challenge signcryption text $\sigma^* = (U^*, C^*, t^*)$ to A_1.

Guess stage: A_1 can make a polynomial bounded number of queries in the find stage. Finally, $\mathcal{C}$ outputs the guess C^*. If $C = C^*, A_1$ makes a query in $h_3$with $V' = \alpha\left(X_R + R_R + h_1 P_{pub}\right)$. In this case the applicant answer of the CDH problem is stored in the list L_3. $\mathcal{C}$ ignores the guessed value of A_1, selects V' randomly from L_3 and outputs $[V' - (x_R + r_R)U^*]/h_1 = \alpha sP$ as the answer to

CDH Problem, where x_R, r_R, U^*, V' are known to the algorithm $\mathcal{C}$. Thus, $\mathcal{C}$ solves the CDHP as$[V' - (x_R + r_R)U^*]/h_1 = \alpha sP$ for the CDHP problem.

Theorem 2 (Confidentialityagainst adversaryA_2): The proposed scheme isIND-PF-PVCLSC-CCA2 secure against the adversary A_2 in the random oracle model with the assumption that the CDHP is hard.

Proof: The proof of this theorem is same as the previous theorem 1 except the following steps.

1. In this game adversaryA_2 have a knowledge on master key s.
2. In the public key query L_{pub}, we set $R = sP$ rather than $R = rP$, and insert $(ID, -, x)$ into L_{rec} other than (ID, r, x).
3. In the guess stage, $\mathcal{C}$ finds$V' - (x_R + h_1 s)U = \alpha sP$ as the answer to the CDH problem.

Theorem 3 (Unforgeabilityagainst adversariesA_1&A_2): The proposed PF-PVCLSC scheme is secure and unforgeable against the adversaries $A_{i\ (i=1,2)}$in the random oracle model with the assumption that the ECDLP is hard.

Proof: Suppose that there is an adversary $A_{i\ (i=1,2)}$ who can break our PF-PVCLSC scheme. We want to build an algorithm $\mathcal{C}$ which uses $A_{i\ (i=1,2)}$ to solve ECDL problem. The algorithm receives an instance adversary $(P, \gamma P)$ of the DL problem and his goal is to compute γ.

Setup: The algorithm $\mathcal{C}$ sets $P_{pub} = \gamma P$and $P_{pub} = sP$ for an adversaries A_1 and A_2 respectively. The remaining procedure is same as theorem 1 and 2 for the adversaries $A_1$and A_2 respectively.

Queries:$A_1$and A_2 are two adversaries can adaptively make a polynomial bounded number of queries like theorem 1 and 2 respectively.

Forgery: An adversary $A_{i\ (i=1,2)}$ out puts signcryption text $\sigma^* = (t^*, c^*, U^*)$ on m^* after receiving a polynomial bounded number of queries with the sender's identity ID_S and the receiver'sidentityID_R.

The algorithm $\mathcal{C}$ first checks the list L_1. If $ID \neq ID^*$, then $\mathcal{C}$ aborts. Otherwise $\mathcal{C}$ can get the private key ofID_R, find $V_R^* = (x_R + d_R)U^*$ and get h_3^* from H_3 queries withV_R^*. $\mathcal{C}$ retrieves m^* and verifies σ^*.

If $A_{i\,(i=1,2)}$ has successfully forged a signature of a user, $\mathcal{C}$ can get two legal signcryptions$(m^*, ID_S, ID_R, U^*, h_2, t_1)$ and $(m^*, ID_S, ID_R, U^*, h_2', t_2)$ where$h_2 \neq h_2'$. Thus we can get$U^* = \alpha P = t_1(r_s h_2 + d_s) = t_2(r_s h_2' + d_s)$.

A_1 chooses $t_1(r_s h_2 + d_s) = t_2(r_s h_2' + d_s) \Rightarrow t_1(r_s h_2 + x_s + \gamma h_1) = t_2(r_s h_2' + x_s + \gamma h_1)$ and he computes γ, science only γ is unknown in the above equation.

A_2 chooses $t_1(r_s h_2 + d_s) = t_2(r_s h_2' + d_s) \Rightarrow t_1(r_s h_2 + x_s + s h_1) = t_2(r_s h_2' + x_s + s h_1)$ and he computes s, science only s is unknown in the above equation. We set $R = rP = \gamma P$ in the public key query, so γ can be computed, which is the solution of the ECDLP instance. Hence ECDLP can be solved.

5.3. Performance Analysis of the Scheme

In this section, we presents the performance of our PF-PVCLSC scheme with respect to computational and communication point of view. For the evaluation of computation and communication costs, we consider the experimental results from the works [29, 30, 31] where various cryptographic operations are evaluated using MIRACL software on Pentium IV and are listed in Table-1. The operations and their conversions presented in Table-1 are achieved by considering the points on elliptic curve group G over the Koblitz curve$E/F_P :\ y^2 = x^3 + ax + b\ mod p$ on a finite field Z_q^*, where the length of the elements of the elliptic curve group G is about 320 bits; a, b in Z_q^*, and the size of q is about 160 bits. Table-2 presents the comparison of our scheme with existing public verifiable signcryption schemes [3, 27] in terms of computation cost.

Table 1: Notations and their Conversions

Notations	Descriptions
T_{ML}	Time needed to execute modular multiplication operation
T_M	Time needed to execute elliptic curve scalar multiplication: $T_M \approx 29T_{ML}$
T_{ME}	Time needed to execute modular exponentiation: $T_{ME} \approx 240T_{ML}$
T_{PE}	Time needed to execute pairing based exponentiation: $T_{PE} \approx 43.5T_{ML}$
T_{PA}	Time needed to execute addition of 2 elliptic curve points: $T_A \approx 0.12T_{ML}$
T_{MTP}	Time needed to execute a map to point(hash function): $T_{MTP} \approx T_M \approx 29T_{ML}$

Computation Cost: In the following, we present the computational complexity of our scheme and other existing secure CLAS schemes namely, Li Cui et al. [27] and F. Li et al. [3] schemes. To evaluate the computational complexity, we consider the signcryption cost, Unsigncryption cost and total cost. Since the Li Cui et al. scheme [27] requires $4T_M + 2T_{PA} + 3T_{MTP}$operationsfor signcryption and $5T_M + 3T_{PA} + 3T_{MTP}$operations for unsigncryption. Hence the total computational cost for Li Cui et al. scheme [27] is$9T_M + 5T_{PA} + 6T_{MTP} = 435.6T_{ML}$. Similarly, the total computation cost for F. Li et al.scheme [3] is $6T_{PE} + T_M = 290T_{ML}$.

From the above Table 2, we can observe that the proposed PF-PVCLSC scheme improves the computational efficiency 53.287% than that Li Cui [27] scheme and 29.834% than the F. Li et al. [3] scheme. The comparison of computation cost and communication cost of our PF-PVCLSC scheme and Li Cui et al. [27] and F. Li et al. [3] schemes is presented graphically in Fig. 1 and Fig. 2 respectively. From Table 2, Fig. 1 and Fig. 2, we conclude that the proposed scheme is computationally efficient than Li Cui et al. [27] and F. Li et al. [3] schemes.

Table 2. Comparison of Computation Cost

Scheme	Computation Cost		Total time	Improvement (in %)
	Signcryption	Unsigncryption		
Li Cui [27]	$4T_M + 2T_{PA} + 3T_{MTP}$	$5T_M + 3T_{PA} + 3T_{MTP}$	$9T_M + 5T_{PA} + 6T_{MTP} = 435.6T_{ML}$	53.287
F. Li [3]	$3T_{PE}$	$3T_{PE} + T_M$	$6T_{PE} + T_M = 290T_{ML}$	29.834
Ours	$3T_M + 2T_{PA}$	$4T_M + 2T_{PA}$	$7T_M + 4T_{PA} = 203.48T_{ML}$	

Communication Cost: In the following, we present the communication cost of our scheme and other existing secure CLSC schemes namely, Li Cui et al. [27] and F. Li et al. [3] schemes. To evaluate the communication cost, we consider the length of the ciphertext.From the experimental results [29, 30, 31], we consider the results to evaluate the communication cost of the schemes in Table 3. Since the ciphertext of the proposed scheme $\sigma = (U, C, t)$has three elements in G. Suppose if the size the of the cipher text C is 100 bits then the communication cost of our scheme is $|G| + |m| + |Z_q^*|$= 360+100+160=580 bits. Similarly, the communication cost of theLi Cui et al. [27]scheme is $|G| + |m| + |Z_q^*|$= 580 bits and the F. Li et al. [3] et al. scheme is $|G_1| + |m| + |Z_q^*|$=1284 bits. From Table 3, we can observe that our scheme has equal communication cost with Li Cui et al. [27]scheme and more efficient than Li Cui et al. [27] scheme.

Table 3. Comparison of Communication Cost

Scheme	Cipher text size /Communication cost
Li Cui et al. [27]	$\lvert G\rvert + \lvert m\rvert + \lvert Z_q^*\rvert$= 580 bits
F. Li et al. [3]	$\lvert G_1\rvert + \lvert m\rvert + \lvert Z_q^*\rvert$=1284 bits
Ours	$\lvert G\rvert + \lvert m\rvert + \lvert Z_q^*\rvert$= 580 bits

The comparison of our PF-PVCLSC scheme with other CLSC schemes in terms of frame work, security such as unforgeability, confidentiality and public verifiability properties are shown in Table 4.

Figure 1. Performance Evaluation for Computational Cost

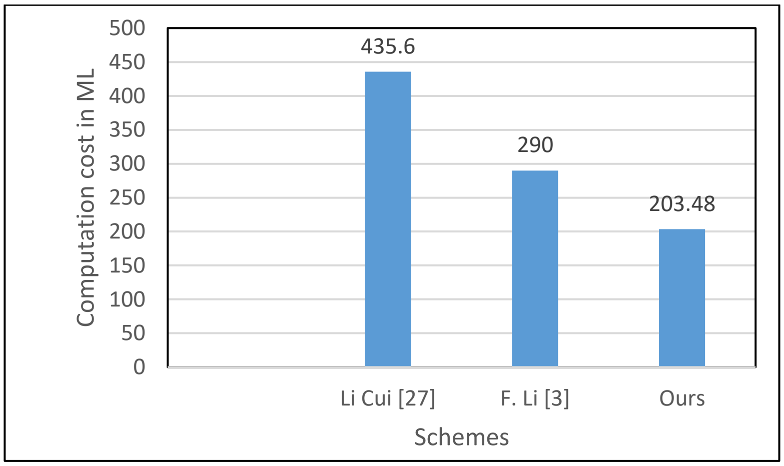

Figure 2. Performance Evaluation for Communication Cost

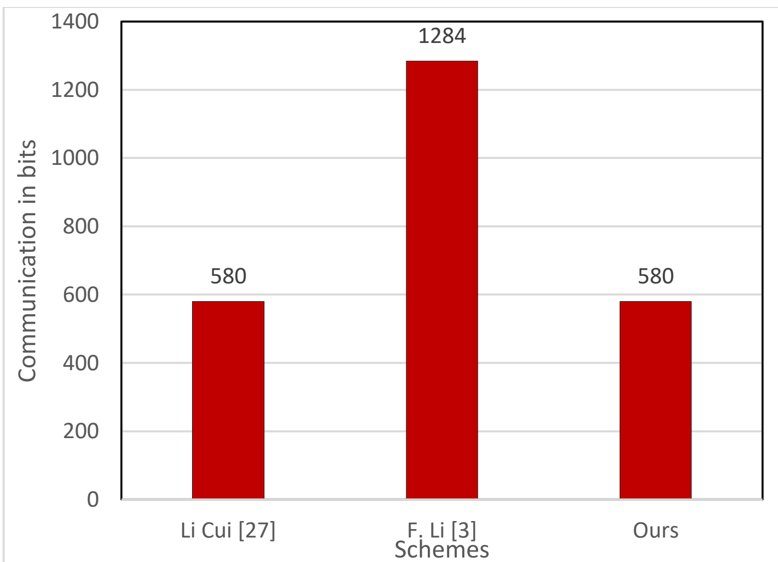

Table 4. Comparison with Supported Features

Scheme	Framework	Unforgeability	Confidentiality	Public Verifiability
Li Cui et al.[27]	Certificateless and Pairing-Free	✓	✓	✓
F.Liet al. [3]	Certificateless and Pairing based	✓	✓	✓
OursPF-CLPVSC	Certificateless and Pairing-Free	✓	✓	✓

6. CONCLUSION

In this paper, we proposed a new signcryption scheme in Certificateless based cryptography. This scheme does not use the expensive bilinear pairings. This scheme supports the property of public verifiability and is secure against various types of adversaries in the random oracle model with the assumption that the Computational Diffie-Hellman Problem (CDHP) and the Elliptic Curve Discrete Logarithmic Problems (ECDLP) are hard. Due to pairing-free environment the proposed scheme is computationally more efficient than the existing public verifiable signcryption schemes. The efficiency analysis shows that the proposed scheme improves the computational efficiency from 29.834% to53.287% than the existing schemes. Also, the proposed scheme has better communication efficiency than the existing schemes. Hence, the proposed PF-PVSC scheme is a good candidate for deployment on resource constrained devices where the devices have limited computing power, storage space and communication bandwidth such as WSNs, VANETs, IoT, sensor devices etc.

CONFLICT OF INTERESTS

The authors declare that there is no conflict of interests.

REFERENCES

[1] M. Malik, Kamaldeep, M. Dutta, On the Applicability of Certificateless Public Key Cryptography (CL-PKC) for Securing the Internet of Things (IoT), in: M. Dutta, C.R. Krishna, R. Kumar, M. Kalra (Eds.), Proceedings of International Conference on IoT Inclusive Life (ICIIL 2019), NITTTR Chandigarh, India, Springer Singapore, Singapore, 2020: pp. 43–50.

[2] A. Shah, M. Engineer, A Survey of Lightweight Cryptographic Algorithms for IoT-Based Applications, in: S. Tiwari, M.C. Trivedi, K.K. Mishra, A.K. Misra, K.K. Kumar (Eds.), Smart Innovations in Communication and Computational Sciences, Springer Singapore, Singapore, 2019: pp. 283–293.

[3] F. Li, J. Hong, A.A. Omala, Efficient certificateless access control for industrial Internet of Things, Future Gener. Comput. Syst. 76 (2017), 285–292.

[4] R. Elhabob, Y. Zhao, I. Sella, An Efficient Certificateless Public key Cryptography with Authorized Equality Test in IIoT, J. Amb. Intell. Human. Comput. 11 (2020), 1065–1083.

[5] A.K. Singh, A Review of Elliptic Curve based Signcryption Schemes, Int. J. Comput. Appl. 102 (2014), 26-30.

[6] M. Mandal, G. Sharma, A.K. Varma, A Computational Review of Identity-based Signcryption Schemes, Int. J. Netw. Secur. 18 (2016), 969-977.

[7] A. Shamir, Identity-Based Cryptosystems and Signature Schemes, in: G.R. Blakley, D. Chaum (Eds.), Advances in Cryptology, Springer Berlin Heidelberg, Berlin, Heidelberg, 1985: pp. 47–53.

[8] S.S. Al-Riyami, K.G. Paterson, Certificateless Public Key Cryptography, in: C.-S. Laih (Ed.), Advances in Cryptology - ASIACRYPT 2003, Springer Berlin Heidelberg, Berlin, Heidelberg, 2003: pp. 452–473.

[9] Y. Zheng, Digital signcryption or how to achieve cost(signature & encryption) ≪ cost(signature) + cost(encryption), in: B.S. Kaliski (Ed.), Advances in Cryptology — CRYPTO ’97, Springer Berlin Heidelberg, Berlin, Heidelberg, 1997: pp. 165–179.

[10] M. Barbosa, P. Farshim, Certificateless signcryption, in: Proceedings of the 2008 ACM Symposium on Information, Computer and Communications Security - ASIACCS ’08, ACM Press, Tokyo, Japan, 2008: p. 369.

[11] X. Cao, W. Kou, A Pairing-free Identity-based Authenticated Key Agreement Scheme with Minimal Message Exchanges, J. Inform. Sci. 180 (2010), 2895-2903.

[12] D. He, J. Chen, J. Hu, An ID-based Proxy Signature Schemes withoutBilinear Pairings, J. Ann. Telecommun. 66 (2011), 657-662.

[13] P. Barreto, A. Deusajute, E. Cruz, C. F. Pereira, and R. Rodrigues. Toward Efficient Certificateless Signcryption from (without) BilinearPairings,http://sbseg2008.inf.ufrgs.br/proceedings/data/pdf/st03 03 artigo.pdf.

[14] S. S.D. Selvi, S.S. Vivek, C.P. Rangan, Cryptanalysis of Certificateless Signcryption Schemes and an Efficient Construction without Pairing, Proceedings of international Conference on information Security and Cryptology, Berlin, LNCS vol. 6151, (2009), 75-92.

[15] W. Xie, Z. Zhang, Certificateless signcryption without pairing. Cryptology ePrint Archive: Report 2010/187, http://eprint.iacr.org/2010/187, (2010).

[16] H. Li, H. Zhu, Y.M. Wang, Certificateless Signcryption Scheme without Pairing, Comput. Res. Develop. 47 (2010), 1587-1594.

[17] W.H. Liu, C. Xu, Certificateless Signcryption Scheme without Bilinear Pairing, J. Softw. 22 (2011), 1918-1926.

[18] X. Jing, Provably secure certificateless signcryption scheme without pairing, in: Proceedings of 2011 International Conference on Electronic & Mechanical Engineering and Information Technology, IEEE, Harbin, Heilongjiang, China, 2011: pp. 4753–4756.

[19] D. He, Security Analysis of a Certificateless Signcryption Scheme, J. Softw. 24 (2013), 618-622.

[20] W.B. Shi, N. Kumar, P. Gong, Z. Zhang, Cryptanalysis and Improvement of a Certificateless Signcryption Scheme without Bilinear Pairing, Front. Comput. Sci. 8 (2014), 656-666.

[21] L. Cao, W.C. Ge, Analysis of Certificateless Signcryption Schemes and Construction of a Secure and Efficient Pairing free one based on ECC, KSII Trans. Internet Inform. Syst. 12 (2018), 4527-4547.

[22] Y. Huang, J. Zhang, H. Chen, On the security of a certificateless signcryption scheme, in: 2014 IEEE Workshop on Electronics, Computer and Applications, IEEE, Ottawa, ON, Canada, 2014: pp. 664–667.

[23] H. Zhu, H. Li, Y. Wang, Certificateless Signcryption Scheme without Pairing, J. Comput. Res. Develop. 47 (2010), 1587–1594.

[24] X.-J. Lin, L. Sun, H. Qu, D. Liu, Cryptanalysis of A Pairing-Free Certificateless Signcryption Scheme, Computer J. 61 (2018), 539–544.

[25] J. Zhang, J. Mao, On the Security of Pairing-Free Certificateless Signcryption Scheme, Security in Computer Systems and networks, Computer J. 61 (2017), 469-471.

[26] H. Yu, B. Yang, Pairing-free and Secure Certificateless Signcryption Scheme, Computer J. 60 (2017), 1187–1196.

[27] L. Cui, B. Yun, S. Lin, B. Wenhua, A New Certificateless Signcryption Scheme Without Bilinear Pairing, in: 2018 13th International Conference on Computer Science & Education (ICCSE), IEEE, Colombo, 2018: pp. 1–5.

[28] G.M. Gao, X.G. Peng, L.Z. Jin, Efficient Access Control Scheme with Certificateless Signcryption for Wireless body Area Networks, Int. J. Netw. Secur. 21 (2019), 428-437.

[29] K. Ren, W. Lou, K. Zeng, P. Moran, On Broadcast Authentication in Wireless Sensor Networks, IEEE Trans. Wireless Commun. 6 (2007) ,4136–4144.

[30] X. Cao, W. Kou, X. Du, A Pairing-free Identity-based Authenticated Key Agreement Protocol with Minimal Message Exchanges, Inform. Sci. 180 (2010), 2895–2903.

[31] S.-Y. Tan, S.-H. Heng, B.-M. Goi, Java Implementation for Pairing-Based Cryptosystems, in: D. Taniar, O. Gervasi, B. Murgante, E. Pardede, B.O. Apduhan (Eds.), Computational Science and Its Applications – ICCSA 2010, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010: pp. 188–198.