Efficient Certificateless Multi-Signcryption Scheme using Elliptic curves

G. Swapna[1*], G. Naga Malleswari[2], Gouri Thumbur[3], T. Kusuma[4]
[1*,2,4] VNR Vignana Jyothi Institute of Engineering and Technology,
[1*]swapnacrypto@gmail.com, [2]malleswari.gn@gmail.com, kusuma_g@vnrvjiet.in
[3]Department of ECE, Geetham University, gthumbur@githam.edu

Abstract: Providing together confidentiality and unforgeability with public verifiability are essential components of secure communication in multi-user settings in real-world applications., especially in scenarios where the generation of digital signatures of multiple users on a single message to the single receiver such as blockchain applications like decentralized finance, transaction signing, and private transactions to ensure secure e-commerce transactions and also provide security in cloud, web, and healthcare applications. The cost of unsigncryption is very high if everyone in the group sends individual signcryptions. Multi-signcryption is an effective alternative solution for this. This article newly introduced a novel multi-signcryption approach in a certificate-less environment to meet the requirements of these applications while avoiding the issues of key-escrow and certificate management in ID-based multi-signcryption and Traditional signcryption respectively. Bilinear pairings in elliptic curves are used to design the proposed scheme and verification of unsigncryption is independent of number of users. The security of the proposed scheme is based on the infeasibility of hard problems, CDHP and CBDHP. In this paper, we intend to guarantee public verifiability while lowering the computational cost of schemes existing in an ID-based setting.

Keywords: *Multi-Signcryption, Certificateless Signcryption, Public Verifiability, Bilinear Pairings, Elliptic Curve, Computational Diffie-Helmann Problem, Computational Bilinear Diffie-Helmann Problem.*

1. Introduction

Public key cryptography is based on two essential requirements such as confidentiality and authentication. Encryption technique ensures confidentiality, while authenticity is provided by digital signatures. Encrypting a message and then signing the ciphertext or signing a message and then encrypting the message are the two conventional methods to offer these security features. Zheng [1] created a new cryptographic primitive called signcryption to decrease the overall computing and cost of communication for performing both compositions simultaneously. Signcryption aims to satisfy all security criteria and offer the functionality of public encryption and signature in a single logical step. The core idea of signcryption is to simultaneously sign and encrypt data, instead of following the traditional sequential approach to accomplish the $\mathrm{Cost\ (Signcryption)} \ll \mathrm{Cost(Encryption)} + \mathrm{Cost(Signature)}$. Beak et al., [2] proved that the original scheme [1] is secure and also provided the security model in traditional cryptography. But it suffers from certificate management in multi-user environments. In 1984, Shamir [3] introduced the new concept of identity-based cryptography (IDBC) to overcome certificate management in traditional PKI settings. In IDBC, every user creates their public keys using a unique identity like IP address, mail ID together with their name. After this many signature schemes [4,5,6] are proposed in IDBC. By combining these concepts, Malone Lee [7] introduced Identity Based signcryption (ID-BSC). Because of the advantages of ID-BSC, numerous different signcryption schemes such as hybrid, aggregate, ring, proxy, and multi-signcryption, as well as other variant schemes are proposed in the literature[8].

Multi-signcryption involves the combinations of multiple digital signatures on a single encryption to form a single signcryption text. This enables simultaneous signcryption of multiple parties. Major applications Include secure multi-party communication in fields of banking, medical, and academia; blockchain applications like decentralized finance (DeFi), transaction signing, and private transactions (Etherium) to ensure secure e-commerce transactions and also provide security in cloud and web applications. The e-voting mechanism implemented using multi-signcryption techniques ensures the integrity of the voter by encryption methods, while tampering can be avoided using multiple signatures for accessing the ballot box. Furthermore, military and high-priority confidential government concerns guarantee national security by encasing transferring of classified information between various parties with multi-signcryption techniques.

In many scenarios in real-world applications, it is necessary to send multiple digital signatures on a single message to a single receiver in an authenticated way. If the group of senders performed the signcryptions individually, it leads to high computational costs and a lot of communications.

To avoid all these Zhang et al., [9] introduced a multi-signcryption scheme in 2009. In addition to providing secure encryption for multiple senders and the functionality of multiple signatures on a single message, multi-signcryption methods also allow for unsigncryption at a cost equivalent to that of a single unsigncryption, independent of the number of senders. Later, in [10] Selvi et al., proved that the scheme in [9] is unforgeable and it is not secure and designed a new algorithm in the multi-user setting to fix the problem in [9]. However, the schemes in[9, 10] don’t provide public verifiability. Afterward, Swapna et al. [11] proposed another approach to adding this attribute. However, all these schemes are in an ID-based setting. It suffers from a key escrow problem. In IDBC, the trusted third party(TTP) will create the user’s private key. So, TTP can generate signatures on different messages, which will address the key escrow problem.

As per our knowledge, there is no multi-signcryption scheme to avoid key escrow problems. So, this paper introduces the multi-signcryption in a certificate-less setting, to prevent the key escrow problem. None of these approaches [9,10,11] fix the key escrow problem. Thus, developing a new certificate-less signcryption in multi-user settings is crucial and beneficial for use in financial transactions, smart homes, health care applications, blockchain applications, e-voting, and sensitive areas.

Contributions:

As far as we are aware, there is no multi-signcryption mechanism in a certificateless setting. To accomplish this goal, we present the idea of multi-signcryption in a certificate-less setting and provide the public verifiability. We propose a certificateless multi-signcryption (CL-MSC) technique and formally defined in this article. We provide a concrete prototype of a CL-MSC scheme and demonstrate its security through the Computational Diffie-Hellman Problem (CDHP), and Bilinear Computational Diffie-Hellman Problem (BCDHP) hardness.

Additionally, we show that our scheme is more efficient than the multi-signcryption schemes that are currently present and also provides public verifiability.

Outline of the Paper:

The remaining paper is structured as follows. Section 2 presents a few associated mathematical definitions, The formal model of a CL-MSC scheme and it’s security requirements. In Section 3, we introduce the new primitive of the CL-MSC scheme. We demonstrate the security of the scheme and analyse it in Section 4. We compare our scheme's efficiency to that of the other available schemes in Section 5, proving the dominance of our scheme over the current schemes. Lastly, a brief summary is provided in Section 7.

2. Preliminaries

In this section, we define computational hard problems[12], the formal model of the CL-MSC scheme, and its security requirements.

2.1. Computational hard problems

Definition 1: Let P be the generator of the additive cyclic group G_1 from the points on elliptic curve. The Computational Diffie-Hellman Problem (CDHP) is to determine $abP \in G_1$, from the given instance (P, aP, bP) with a known parameter P.

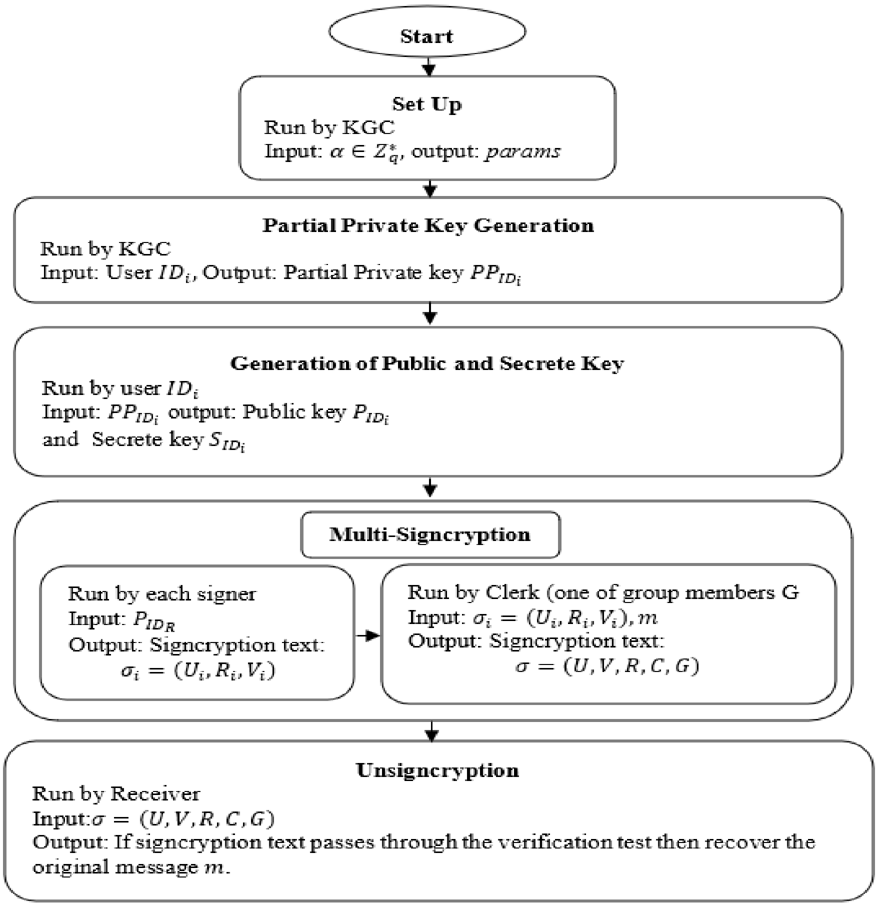
Definition 2: Let P be the generator of the additive cyclic group G_1 from the points on elliptic curve. According to CDHP assumption on G_1, there is no polynomial-time algorithm that can be able to solve the CDHP in G_1 with a non-negligible advantage.

Definition 3: Let P be the generator of the additive cyclic group G_1 from the points on elliptic curve, G_2 be a multiplicative cyclic group, and $\hat{e}: G_1 \times G_1 \rightarrow G_2$ is a bilinear pairing. Let $a, b \in Z_q^*$, be selected at random and kept confidential. The Computational Bilinear Diffie-Hellman problem (CBDHP) is to compute $\hat{e}(P, P)^{ab} \in G_2$, from $P, aP, bP \in G_1$.

Definition 4: Let P be the generator of the additive cyclic group G_1 from the points on elliptic curve. According to CBDHP assumption on G_2, there is no polynomial-time algorithm that can be able to solve the CBDHP in G_2 with a non-negligible advantage.

2.2. Formal Model for CL-APSC Scheme

Here, we define the CL-MSC scheme model through a flow chart algorithm. It contains five algorithms: set up, Partial Private Key Generation, Generation of Public and secret Key, Multi-Signcryption, and Unsigncryption.


Start
Set Up
Run by KGC
Input: $\alpha \in Z_q^*$, output: $params$
Partial Private Key Generation
Run by KGC
Input: User ID_i, Output: Partial Private key PP_{ID_i}
Generation of Public and Secrete Key
Run by user ID_i
Input: PP_{ID_i} output: Public key P_{ID_i}
and Secrete key S_{ID_i}
Multi-Signcryption
Run by each signer
Input: P_{ID_R}
Output: Signcryption text:
$\sigma_i = (U_i, R_i, V_i)$
Run by Clerk (one of group members G
Input: $\sigma_i = (U_i, R_i, V_i), m$
Output: Signcryption text:
$\sigma = (U, V, R, C, G)$
Unsigncryption
Run by Receiver
Input:$\sigma = (U, V, R, C, G)$
Output: If signcryption text passes through the verification test then recover the original message m.

2.3. Security requirements for CL-MSC Scheme

The combined requisites are unforgeability and confidentiality for the signcryption as its nature. These security requirements are complicated and quite severe. We suppose that R is the receiver and, S_i, $i = 1,2,3 \ldots n$ are the signcrypters. The proposed CL-MSC scheme should satisfy security requirements such as confidentiality, unforgeability, and public verifiability.

- Confidentiality: It is infeasible for any intruder to retrieve the plain-text m or to generate the private key of the receiver R from σ, the signcryption text .
- Unforgeability: Any intruder can’t forge the signature of anyone in the signcrypter group.
- Public Verifiability: Any third party can authenticate the validity of our CL-MSC scheme without the knowledge of the original message and private key of the receiver

3. Design of CL-MSC Scheme

In this section, the authors introduce the new multi-signcryption scheme in certificate-less settings in elliptic curves using bilinear pairings. The scheme has five algorithms: Set-Up, Partial Private Key (PPK) Generation, Generation of Public and Private keys, Multi-Signcryption, and Unsigncryption. The algorithms are described as follows.

Set-Up: KGC selects the security parameter λ and chooses additive cyclic groups G_1 and G_2 of order q and P be the generator of G_1. Defines a bilinear mapping as $\hat{e}$: $G_1 \times G_1 \to G_2$. Also defines hash functions $H_1: \{0,1\}^* \to G_1, H_2: G_2 \to \{0,1\}^*, H_3: \{0,1\}^* \to Z_q^*$. KGC randomly chooses the secrete key $\alpha \in Z_q^*$ and computes $P_{Pub} = \alpha P$. Finally, KGC publishes the parameters as $params = \{P, q, G_1, G_2, P_{Public}, \hat{e}, H_1, H_2, H_3\}$.

Partial Private Key Generation: This algorithm is run by KGC with the sender’s identity $IDs_i(i = 1, 2, 3, \ldots n)$, receiver's identity ID_R to create PPK. KGC generates the PPK as $PP_{ID_i} = \alpha Q_{ID_i}$ where $Q_{ID_i} = H_1(ID_i)$. Finally, send the PPK to the respective user.

Generation of Public and Secrete Keys: All the users, run this algorithm to generate their public and secret keys. Every user randomly selects the secret value $\mu_i \in Z_q^*$ and computes the private key as $S_{ID_i} = \mu_i PP_{ID_i}$ also computes their public key as $P_{ID_i} = \hat{e}(P, S_{ID_i})$.

Multi-Signcryption Scheme: Each sender with an identity IDs_i in the group G of n members execute this algorithm with the receiver identity ID_R, the public key of the receiver P_{ID_R}, and the message m to generate signcryption text σ_i. Each sender performs the following steps.

1. Each sender randomly chooses $x_i \in Z_q^*$ and finds $U_i = x_i P$ and $R_i = \mu_i P$.
2. Computes $\mathfrak{D}_i = P_{ID_R}{}^{x_i}$.
3. Send $(U_i, \mathfrak{D}_i)$ to all other senders in the group using the secure channel.
4. After receiving $(U_i, \mathfrak{D}_i)$ from the other senders, each sender computes $\mathfrak{D} = \prod_{i=1}^{n} \mathfrak{D}_i, \beta = H_2(\mathfrak{D})$ and encrypts the message m by $E_\beta(m)$ and then

computes $U = \sum_{i=1}^{n} U_i$ and $R = \sum_{i=1}^{n} R_i$.

5. Computes $h = H_3(C, U, R, P_{ID_R}, G)$.
6. Each sender generates the signature by computing $V_i = S_{ID_{S_i}} + hx_iP_{Pub}$ and then sends it to the clerk along with the values $U, R,$ and C. Once receiving (V_i, U, R, C) from all the senders, the clerk verifies whether $U, R,$ and C values are the same, if so, then the clerk computes $V = \sum_{i=1}^{n} V_i$. Finally, output the resultant signcryption text $\sigma = (C, U, R, V, G)$ and sends it to the receiver R.

Unsigncryption: The receiver with an identity ID_R executes this algorithm with the sender's identity ID_{S_i}, the public key of the sender's $P_{ID_{S_i}}$, and the signcryption text σ_i. To verify the signcryption text and decrypt the message. The receiver performs the following steps.

1. Compute $\mathfrak{D}' = \hat{e}(U, S_{ID_R})$
2. Compute $\beta' = H_2(\mathfrak{D}')$
3. Compute $m' = D_{\beta'}(C)$
4. Compute $h = H_3(C, U, R, P_{ID_R}, G)$.
5. Accept the message m' iff $\hat{e}(P, V) = \hat{e}(P_{Pub}, R + hU)$

4. Analysis of the CL-MSC Scheme

In this section, we provide the proof of correctness, security analysis, and efficiency analysis of our Cl-MSC scheme.

4.1. Correctness of CL-MSC Scheme

The acceptability or correctness of the proposed scheme is proved by the following equations.

$$\hat{e}(P,V) = \hat{e}\,(P, \sum_{i=1}^{n} V_i) = \hat{e}\,(P, \sum_{i=1}^{n} S_{ID_S} + hx_iP_{Pub})$$

$$= \hat{e}\,(P, \sum_{i=1}^{n} \mu_i P P_{ID_i} + hx_iP_{Pub}) = \hat{e}\,(P, \sum_{i=1}^{n} \mu_i \alpha Q_{ID_i} + hx_i\alpha P)$$

$$= \hat{e}\,(\alpha P, \sum_{i=1}^{n} \mu_i Q_{ID_i} + hx_iP) = \hat{e}(P_{Pub}, R + hU)$$

4.2. Security Analysis of CL-MSC Scheme

In this section, we discuss the security parameters of the CL-MSC Scheme like confidentiality, unforgeability, and public verifiability.

- *Confidentiality*: without knowledge of $\mathfrak{D}$, No one can decrypt the message, since it needs the receiver's private key.

$$\mathfrak{D}' = \hat{e}(U, S_{ID_R}) = \hat{e}\,(\sum_{i=1}^{n} U_i\,, S_{ID_R}) = \hat{e}\,(\sum_{i=1}^{n} x_iP\,, S_{ID_R})$$

$$= \hat{e}\,(\sum_{i=1}^{n} P\,, S_{ID_R}) = \mathop{\mathbf{G}}_{i=1}^{n} \hat{e}(P, S_{ID_R})^{x_i} = \mathop{\mathbf{G}}_{i=1}^{n} P_{ID_R}{}^{x_i} = \mathop{\mathbf{G}}_{i=1}^{n} \mathfrak{D}_i = \mathfrak{D}.$$

To obtain the receiver’s private key, an intruder must solve CBDHP. But it is infeasible in terms of security parameters.

- *Unforgeability*: Any sender of the group G or any outsider who is not involved in the whole protocol can forge the proposed CL-MSC scheme. Initially, since the CL-MSC technique uses the private keys of the other signers, so no one in the signcrypter group is able to produce a legitimate one. Furthermore, the clerk in signcrypter group G, who combines all the signatures, can select his x_i and $\mathfrak{D}_i$ values before endeavoring to compute V, such that $\hat{e}(P,\ V) = \hat{e}(P_{Pub},\ R + hU)$ holds if he wishes to sign on the false message m. However, this is the same as solving the bilinear inversion problem, which is reducible to CDHP in G_1 and ECDLP in G_2. So that, the clerk can’t forge the proposed CL-MSC scheme. Since the other signers in group G have less privilege to forge the signature than the clerk, so, the signcryptors are unable to forge the CL-MSC scheme. Finally, any intruder who is not involved in the CL-MSC protocol is unable to forge CL-MSC scheme even though the intruder obtained the signatures of all the signers, because he needed the private keys of all users to forge the CL-MSC scheme. Obtaining each signer's private key at these stages is the same as forging the signature [13], which is proven secure. Therefore, our CL-MSC scheme is unforgeable.
- *Public Verifiability*: Validation of the authenticity of the signcrypted message without knowing the original message is termed public verifiability. any third party can verify the authenticity of our CL-MSC scheme if any disputes occur between sender and receiver.

4.3. Efficiency Analysis of CL-MSC Scheme

Here the authors compare our CL-MSC scheme with the related schemes existing in the other paradigm, in an identity-based setting, since there is no other Multi-Signcryption scheme in the certificate-less setting. The efficiency of our scheme is compared with the schemes like [9,10,11]. The computational and communication costs are assessed by referring to the experimental findings from the works [14,15,16], whereby a range of cryptographic operations are assessed on a Pentium IV computer using MIRACL software. The results are presented in Table 1. The methods used to accomplish the operations and their conversions presented in Table 1 involve taking into account the points on elliptic curve group G over the Koblitz curve E/F_P: $x^3 + ax + b\ modp$ on a field of Z_q^*, where the length of the elements in the elliptic curve group G is approximately 320 bits; $a,\ b\ \epsilon\ Z_q^*$, and the size of q is approximately 160. Comparison of our scheme with existing schemes in table-2 in computation point of view. As we compare with the existing schemes our scheme has less computational cost and verification of unsigncryption scheme is irrespective of the senders. Fig 1. shows that the running time for the signcryption algorithm of our CL-MSC scheme is comparatively very low, with the other existing schemes and it is $9135T_{ML}$ for 70 users . Fig 2. shows that the verification of the unsigncryption algorithm is irrespective of signers and the running time is $290T_{ML}$. Fig 3. Shows that the total running time of our scheme is very low and it is $9525T_{ML}$ for 70 users. As the number of users increased in the network then our scheme is efficient in computational point of view.

Table 1: Notations and running time of numerous cryptographic operations

Notations	Descriptions and running time
$\boldsymbol{T_P}$	Running time for computation of one pairing 87 T_{ML}
$\boldsymbol{T_{PE}}$	Running time for computation of on Pairing based exponentiation $\approx 43.5T_{ML}$
$\boldsymbol{T_{SM}}$	Running time for computation of one scalar multiplication $\approx 29T_{ML}$

Table-2: Performance comparison

Scheme	Multi-Signcryption	Unsigncryption	Total time	Total cost in $\boldsymbol{T_{ML}}$
[9]	$3nT_{SM}+nT_{PE}+nT_P$	$T_{PE}+4T_P$	$3nT_{SM}+(n+1)T_{PE}+(n+4)T_P$	217.5n+391.5
[10]	$3nT_{SM}+nT_{PE}+nT_P$	$T_{PE}+4T_P$	$3nT_{SM}+(n+1)T_{PE}+(n+4)T_P$	217.5n+391.5
[11]	$2nT_{SM}+nT_{PE}+nT_P$	$T_{SM}+3T_P$	$(2n+1)T_{SM}+nT_{PE}+(n+3)T_P$	188.5n+290
[ours]	$3nT_{SM}+nT_{PE}$	$3T_P+T_{SM}$	$(3n+1)T_{SM}+nT_{PE}+3T_P$	130.5n+290

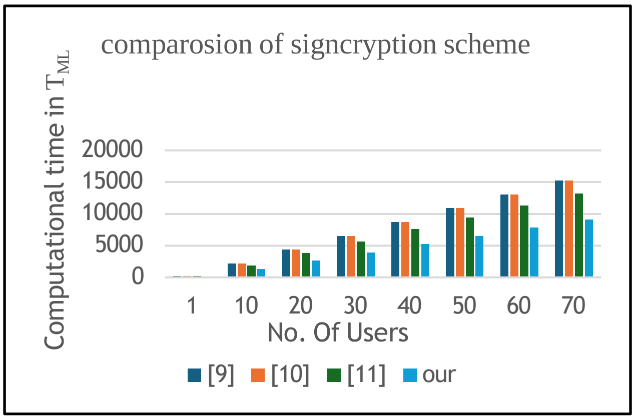


Fig 1. Running time for Signcryption Algorithm

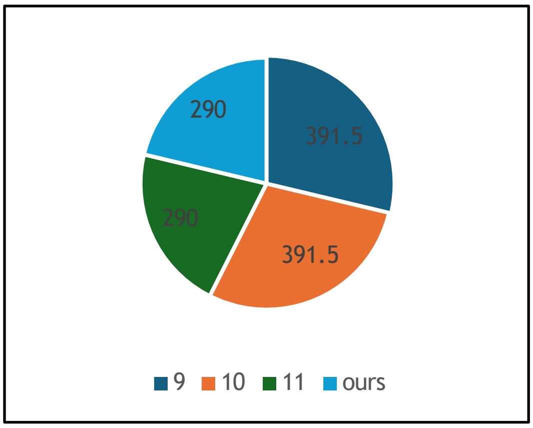


Fig 2. Running time of Unsigncryption time

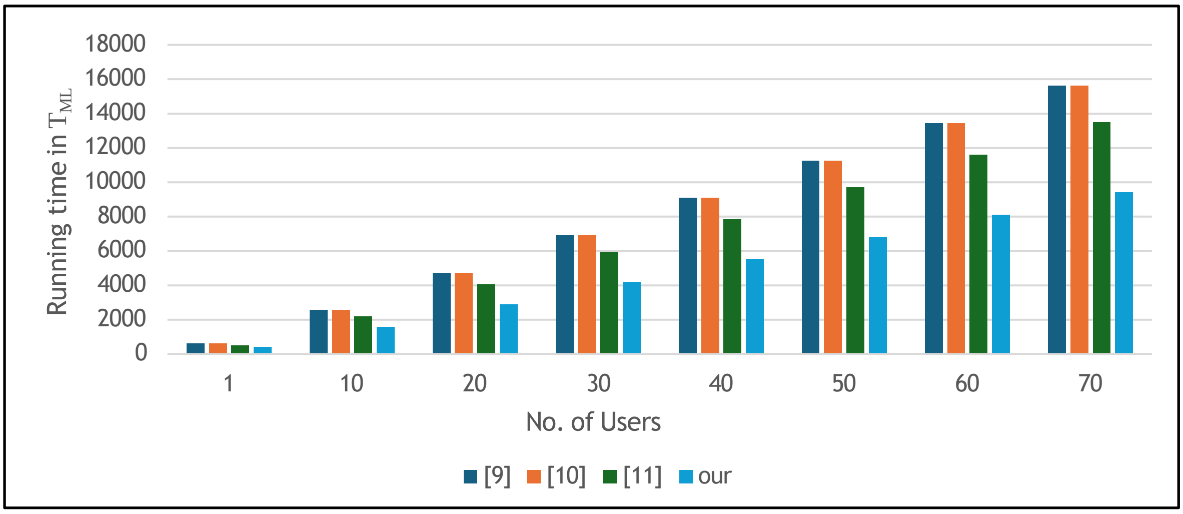


Fig 3. Total Running time in T_{ML}

Table 3:Security Notions

	Confidentiality	**Unforgeable**	**Public Verifiable**	**No Key Escrow Problem**
[9]	√	√	×	×
[10]	√	√	×	×
[11]	√	√	√	×
[Ours]	√	√	√	√

5. Conclusion

This paper introduces a novel certificateless multi-signcryption scheme, to address the inherent challenges of ensuring unforgeability, confidentiality, public verifiability and key escrow problem in communication protocols. We introduced multi-signcryption scheme using bilinear pairings in elliptic curve. We proved that our scheme is confidential and unforgeable and public verifiable in a certificateless setting. By employing certificateless cryptography, the scheme eliminates the need for certificates, thereby mitigating the risks associated with key escrow. Furthermore, performance evaluations indicate that the scheme achieves commendable efficiency in terms of computational overhead.

Overall, the certificateless multi-signcryption scheme presented in this paper offers a promising solution for achieving confidentiality, unforgeability, and public verifiability in communication protocols, thereby catering to the evolving security requirements of modern information systems.

6. References

1. Y. Zheng, “Digital signcryption or how to achieve cost(signature encryption,” in Proceedings of the cost(signature) + cost(encryption), in Advances in Cryptology - CRYPTO 97, 17th Annual International Cryptology Conference, pp. 165–179, Santa Barbara, California, USA, 1997.
2. J Baek, J Newmarch, R Safavi-Naini, and W. Susilo, A Survey of Identity-Based Cryptography, In Proc. of the 10th Annual Conference for Australian Unix User's Group pp 95-102, 2004.
3. Shamir, A. (1985). Identity-Based Cryptosystems and Signature Schemes. In: Blakley, G.R., Chaum, D. (eds) Advances in Cryptology. CRYPTO 1984. Lecture Notes in Computer Science, vol 196. Springer, Berlin, Heidelberg. https://doi.org/10.1007/3-540-39568-7_5.
4. M. Mambo, K. Usuda and E. Okamoto, Proxy signatures: Delegation of the power to sign messages, IEICE Trans. Fundam. Electron. Comm. Comput. Sci. 79 (1996), no. 9, 1338–1354.
5. C. Gamage, J. Leiwo, and Y. Zheng, ‘‘An efficient scheme for secure mesasage transmission using proxy-signcryption,’’ in Proc. 22nd Australasim Comput. Sci. Conf., Auckland, New Zealand: Springer-Verlag, 1999, pp. 420–431.
6. J Baek, J Newmarch, R Safavi-Naini and W. Susilo, ―A Survey of Identity-Based Cryptography, In Proc. of the 10th Annual Conference for Australian Unix User's Group pp 95-102, 2004.
7. J. Malone Lee, “Identity based Signcryption”, In Cryptology e-Print Archive, Report 2002/098, 2002.
8. Padmalaya Nayak, G Swapna, “Security issues in IoT applications using certificateless aggregate signcryption schemes: An overview”, Internet of Things, Volume 21, 2023, 100641, ISSN 2542-6605, https://doi.org/10.1016/j.iot.2022.100641.
9. J Zhang, J Mao, A novel identity-based multi-signcryption scheme, Computer communications, Vol 32(1), 2009, pp. 14-18. https://doi.org/10.1016/j.comcom.2008.07.004.
10. Selvi, S.S.D., Vivek, S.S., Rangan, C.P. (2009). Breaking and Fixing of an Identity Based Multi-Signcryption Scheme. ProvSec 2009. LNCS, vol 5848. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-04642-1_7
11. Swapna, G., & Vasudeva Reddy, P. (2014). Efficient Identity Based Multi-Signcryption Scheme with Public Verifiability. *Journal of Discrete Mathematical Sciences and Cryptography*, *17*(2), pp. 181–190. https://doi.org/10.1080/09720529.2013.867674
12. Koblitz, A. H., Koblitz, N., & Menezes, A. (2011). Elliptic curve cryptography: The serpentine course of a paradigm shift. *Journal of Number theory*, *131*(5), 781-814.
13. K. A. Shim. “An ID-based aggregate signature scheme with constant pairing computations”, *The Journal of Systems and Software*, Vol. 83, 2010, pp. 1873–1880.
14. K. Ren, W. Lou, K. Zeng, P. Moran, On Broadcast Authentication in Wireless Sensor Networks, IEEE Trans. Wireless Commun. 6 (2007) ,4136–4144.
15. 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.

16. 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.