CORA Cork Open Research Archive
Cartlann Taighde Oscailte Chorcaí

Title	Mkem-dem based anonymous certificateless hybrid signcryption for broadcast vanet communication
Authors	Umrani, Alia;Vangujar, Apurva K.;Palmieri, Paolo
Publication date	2025-07-22
Original Citation	Umrani, A., Vangujar, A.K. and Palmieri, P. (2026) 'Mkem-dem based anonymous certificateless hybrid signcryption for broadcast vanet communication', in: Lenzini, G., Mori, P. and Furnell S. (eds.) Information Systems Security and Privacy, vol 2459, pp. 246–273. Springer, Cham. https://doi.org/10.1007/978-3-031-89518-0_12
Type of publication	Conference item
Link to publisher's version	https://link.springer.com/book/10.1007/978-3-031-89518-0 - 10.1007/978-3-031-89518-0_12
Rights	© 2025, the Authors, under exclusive license to Springer Nature Switzerland AG. This version of the paper has been accepted for publication, after peer review (when applicable) and is subject to Springer Nature's AM terms of use, but is not the Version of Record and does not reflect post-acceptance improvements, or any corrections. The Version of Record is available online at: https://doi.org/10.1007/978-3-031-89518-0_12
Download date	2026-11-01 19:38:17
Item downloaded from	https://hdl.handle.net/10468/18287



University College Cork, Ireland
Coláiste na hOllscoile Corcaigh



mKEM-DEM Based Anonymous Certificateless Hybrid Signcryption for Broadcast VANET Communication

Alia Umrani(✉), Apurva K. Vangujar, and Paolo Palmieri

School of Computer Science and IT, University College Cork,
Western Road, Cork T12 XF62, Ireland
{a.umrani,a.vangujar,p.palmieri}@cs.ucc.ie

Abstract. Vehicular Ad hoc Networks are open network environments that enable the exchange of safety messages among vehicles. However, due to the critical nature of wireless networks, communication is susceptible to various attacks, including eavesdropping, impersonation, and privacy violations. Therefore, confidentiality, authentication, and anonymity are essential in broadcast communication, achievable by digital signature, encryption, and Pseudo-Identity ($\mathcal{P}_{\mathsf{ID}}$) techniques. We design and introduce a multi-receiver Key Encapsulation Mechanism (mKEM)-Data Encapsulation Mechanism (DEM) based Anonymous Certificateless Hybrid Signcryption (AMCLHS) scheme that combines symmetric and asymmetric key cryptography. We show a straightforward and cost-effective construction of a mKEM-DEM that creates a symmetric session key and signcrypt the message to achieve confidentiality and authentication. The scheme maintains anonymity by assigning a $\mathcal{P}_{\mathsf{ID}}$ to each vehicle user. We design security notions and prove security against chosen-ciphertext attack using the elliptic curve computational Diffie-Hellman assumption under random oracle model and chosen message attack by utilizing the elliptic curve discrete logarithmic assumption. The AMCLHS scheme works in a certificateless environment, exempting the key escrow. We implement our scheme and statistically compare the performance with existing schemes for multiple receivers. Comparative analysis demonstrates that the proposed scheme achieves optimal communication cost, is computationally efficient, and ensures the security properties of confidentiality, authentication, anonymity, non-repudiation, and forward security.

Keywords: Key Encapsulation · Hybrid Signcryption · VANET · Broadcast · Confidentiality · Authentication

1 Introduction

Significant developments in technology have created an extensive framework of billions of interconnected devices into intelligent and complex infrastructures in sectors such as healthcare, industry, education, and transportation [9]. Vehicular Ad hoc Network (VANET) is a special application of the Intelligent Transportation System (ITS) where each vehicle acts as a node.

© The Author(s), under exclusive license to Springer Nature Switzerland AG 2026
G. Lenzini et al. (Eds.): ICISSP 2023/ICISSP 2024, CCIS 2459, pp. 246–273, 2026.
https://doi.org/10.1007/978-3-031-89518-0_12

VANET brings an easy and convenient driving experience for drivers and passengers through cooperative management [8,18]. Each vehicle is equipped with Wireless Sensor Networks (WSNs), embedded devices, and Onboard Units (OBUs), which enable the acquisition of real-time traffic and vehicle information. Each vehicle broadcasts the acquired traffic information, such as traffic jams, early warning, and accidents, to other vehicles, called Vehicle-to-Vehicle (V2V) communication, and to Road-Side Units (RSUs), called Vehicle-to-Infrastructure (V2I) communication. V2V and V2I utilize the Dedicated Short Range Communication (DSRC) protocol stack to communicate with each other. However, due to the high dependability of the wireless network, VANET is susceptible to a range of potential security risks, including eavesdropping, impersonation, privacy violations, unauthorized access, or message modification. Therefore, ensuring confidentiality, authentication, and anonymity becomes significant for secure broadcast communication in VANETs [24]. To achieve these security requirements, cryptographic primitives play a significant role, such as encryption and Digital Signatures (DS), also known as 'sign-then-encrypt' schemes. However, the traditional sign-then-encrypt approach incurs significant computational overhead in a dynamic VANET environment with limited resources, where storage constraints and the demand for high-speed communication introduce additional challenges. Alternatively, Signcryption is a cryptographic mechanism that enables simultaneous encryption and DS cryptographic operations while providing more effective authentication as well as confidentiality. Zheng et al. [40] introduced the first signcryption scheme, which is considered a unique cryptographic concept.

Since then, several signcryption schemes have been introduced in different application scenarios, including but not limited to VANETs. The first Identity (ID)-based signcryption scheme was proposed by Malone-Lee [20]. Following that, Singh et al. [28], in 2024, proposed an ID-based signcryption for secure communication that simultaneously provides confidentiality and authentication. However, in ID-based schemes, the Public Key Generator is responsible for generating the private key of the user, resulting in the potential private key escrow problem. To address this issue, Al-Riyami et al. [1] introduced the concept of certificateless public key cryptography, in which the Key Generation Center (KGC) only computes a part of the user private key called the partial private key (ppk). To generate the actual private key, the user first computes a secret value and combines it with previously computed ppk. Consequently, the KGC does not have knowledge of the entire private key of the user. Subsequently, the first certificateless-based signcryption was proposed by Barbosa et al. [2].

The aforementioned schemes are designed for single-receiver scenarios, making them inefficient for broadcast multi-receiver communication environments. For instance, to send a similar message to multi-receivers, the user has to encrypt it for all recipients individually, which results in higher computational demands and increased communication delays. To address this, Yu et al. [38] proposed an ID-based multi-receiver signcryption scheme that encrypts a message for n designated recipients. The security of this scheme has been proven in the ROM. Later on, several ID-based multi-receiver signcryption schemes were proposed. However, since ID-based PKC has a core challenge of key escrow, Selvi et al. [27] proposed the first multi-receiver certificateless signcryption scheme that works in broadcast settings while eliminating key escrow.

Broadly, two primary approaches are considered to construct the signcryption: (i) PKC signcryption: In this method, both message encryption and signature occur within the public key framework [27]. (ii) Hybrid signcryption: This approach offers the benefits of integrating symmetric encryption with an asymmetric key signature to ensure additional security including integrity, authentication, and non-repudiation [26]. Hybrid signcryption tends to be more efficient in resource-constrained environments compared to PKC signcryption. In the latter, large messages are transmitted along with large public key values, leading to inefficiency. For further understanding, we recommend reviewing Dent's research [6,7] on hybrid signcryption. In the proposed paper, we extend our work on anonymous multi-receiver hybrid signcryption scheme [31] which was presented in ICISSP 2024. Building on this work, we introduce a mKEM-DEM based anonymous certificateless hybrid signcryption scheme for broadcast VANET communication, serving extension of the previous research. In the previous work, we focused on designing a multi-receiver certificateless hybrid signcryption for a generalised broadcast scenario and demonstrated the formal security proofs using the Elliptic-Curve Computational Diffie-Hellman (EC-CDH) and Elliptic-Curve Discrete Logarithm (ECDL) assumptions. In our current work, we present an efficient and secure anonymous certificateless multi-receiver hybrid signcryption technique, particularly for broadcast VANET communication. We provide a comprehensive explanation of the VANET environment, security requirements, possible attacks, and prevention techniques. Additionally, we extend the scope of the literature review and provide a detailed critical analysis of existing signcryption techniques for multi-receivers and their comparison with our proposed work on AMCLHS. We further implement our scheme and compare the statistical results with the existing work.

Our scheme uses mKEM-DEM that first generates a symmetric key and then signcrypts the message using the symmetric key and the receiver's public key, respectively. To ensure confidentiality, indistinguishability is demonstrated against Chosen-Ciphertext Attack *(ind-cca2-I, ind-cca2-II)* which is based on the EC-CDH, and to prove unforgeability, we define Existential Unforgeability which ensures security against Chosen Message Attack (*euf-cma-I, euf-cma-II)* for Type-I ($\mathcal{A}_{\mathrm{I}}$) and Type-II ($\mathcal{A}_{\mathrm{II}}$) adversaries, respectively using the ECDL assumptions. Furthermore, each user, including the sender and receiver vehicles, is granted a Pseudo-Identity ($\mathcal{P}_{\mathsf{ID}}$) to ensure anonymity. Additionally, we demonstrate security for non-repudiation and forward security. Our technique works in a broadcast vehicular environment within certificateless settings, avoiding key escrow concerns. Finally, we implement the scheme and provide a detailed statistical comparison analysis with existing multi-receiver certificateless hybrid signcryption schemes. The efficiency is demonstrated with respect to communication and computation costs and the required security achieved by the existing schemes. The comparison shows that our technique depicts enhanced efficiency, where the cost of unsigncryption is constant as the number of receivers increases and the cost of signcryption linearly increases. Our technique simultaneously fulfills all the security requirements compared to other schemes, including confidentiality by utilizing encryption, unforgeability from DS, anonymity through the use of $\mathcal{P}_{\mathsf{ID}}$, non-repudiation, and forward security due to hybrid settings. The main contributions of our technique are outlined below:

1. We design and extend the mKEM-DEM based Anonymous Certificateless Hybrid Signcryption technique (AMCLHS) for the multi-receiver broadcast VANET scenario. Our scheme integrates asymmetric and symmetric key cryptography to broadcast signcrypted messages.
2. Our scheme achieves user vehicle anonymity by allocating $\mathcal{P}_{\mathsf{ID}}$ to each user. Additionally, it allows the sender vehicle to signcrypt an identical message for multiple recipients while preserving the anonymity of vehicle identities.
3. The technique works within certificateless settings, effectively mitigating the issue of key escrow while ensuring confidentiality and unforgeability.
4. To ensure confidentiality, we define *ind-cca2-I* and *ind-cca2-II*, while unforgeability is proven by presenting security in terms of *euf-cma-I* and *euf-cma-II*, respectively. The security is elaborated under the ROM using the hard assumptions of EC-CDH and ECDL.
5. We computed and statistically compared the communication and computational efficiency of our technique with the existing signcryption schemes with respect to ciphertext length and the total count of computational operations. The results show that the proposed AMCLHS scheme achieves higher computational efficiency and optimal communication cost.

The organization of our paper is defined as follows: In Sect. 2, we provide the literature review. The definitions and mathematical assumptions are provided in Sect. 3. Section 4 defines our AMCLHS framework, and Sect. 5 introduces our technique's security model. In Sect. 6, we describe our proposed AMCLHS technique, while Sect. 7 provides the security analysis using the assumptions. In Sect. 8, we present the performance evaluation and comparison analysis of the proposed technique. Lastly, Sect. 9 concludes and summarizes the future scope.

2 Related Work

2.1 Certificateless Signcryption

Zheng et al. [40] proposed the initial idea of signcryption to achieve confidentiality and authentication more efficiently than sign-then-encrypt by integrating together the signature and encryption operations. Since then, numerous ID-based signcryption techniques were introduced. However, the main challenge remains the existence of a key escrow.

To prevent the challenge of key escrow, the first signcryption scheme based on certificateless settings was introduced by Barbosa et al. [2], which achieves confidentiality as well as authentication and demonstrates security in ROM. Furthermore, a certificateless-based signcryption was introduced for the Internet of Vehicles (IoVs) by Cui et al. [5] and the Internet of Medical Things without utilizing bilinear pairing (BP) operation by Chen et al. [3] under the ECDL and Computational Diffie-Hellman (CDH) assumptions, respectively. Similarly, Zhou et al. [41] introduced a signcryption technique in certificateless settings, ensuring unforgeability and confidentiality in the standard model. However, their technique lacks the property of anonymity. Kasyoka et al. [13] introduced a certificateless-based signcryption technique for WSN. Furthermore, a signcryption technique designed specifically for smart terminals deployed in

resource-constrained environments in cyber-physical power systems was proposed by Li et al. [17].

Nonetheless, the techniques discussed above are constructed particularly for single-recipient scenarios, making them inefficient and unsuitable for broadcast communication settings. The initial ID-based PKC multi-receiver signcryption technique was introduced by Yu et al. [38] which enables message signcryption for n targeted receivers, and the security is proven under the ROM using CDH assumptions. Subsequently, numerous multi-receiver certificateless signcryption schemes were introduced. In 2024, Zhu et al. [43] proposed an online/offline signcryption technique for heterogeneous wireless body area networks. While this technique achieves the required security, it relies on computationally expensive bilinear operations. In 2022, a signcryption technique for fog-based smart grids providing privacy-preserving aggregation was proposed by Li et al. [16]. The scheme ensures confidentiality (*ind-cca*) to protect electricity consumption data from eavesdropping and establishes security against *euf-cma* for unforgeability. However, the scheme lacks the security of the cloud, which offers storage and query services to the control center. Niu et al. [22] introduced a signcryption technique for scenarios based on 5G network slicing that ensures privacy and works in mutual heterogeneous settings. In their technique, the sender operates within a Public Key Infrastructure (PKI) setting, while the receiving entity exists within the certificateless setting. Their technique ensures *euf-cma* as well as *ind-cca2* security and relies on Discrete Logarithm (DL) and CDH hard assumptions. Similarly, Ullah et al., Guo et al., and Elkhalil et al. [9,11,30] presented certificateless signcryption schemes for IoVs and VANETs respectively. However, these schemes either lack anonymity and/or non-repudiation security requirements. In 2023, Luo et al. [19] proposed an anonymous cross-domain smart grid communication scheme using ring signcryption that ensures confidentiality, unforgeability, forward security, and anonymity by demonstrating *ind-cca* and *euf-cma*. In this scheme, each domain works in different PKC settings, such as certificateless, PKI, and ID-based public key settings.

While achieving each security goal, the paper lacks explanations on dynamic ring changes and practical deployment considerations, including the reliability of KGCs and interoperability with existing smart grid protocols. Furthermore, several multi-receiver signcryption techniques based on certificateless settings were proposed tailored for IoT-based maritime transportation systems, edge computing, the Internet of Things (IoT) applications [24,25,32,33,35,37]. While these techniques, designed for extensive and resource-consuming environments, were ensured to be secure in public key scenarios, they likely will incur higher computational costs when handling large messages compared to hybrid settings. In contrast, hybrid signcryption generally offers more efficiency than solely relying on public key signcryption schemes by integrating asymmetric and symmetric key cryptography. Particularly, the message is encrypted with a more efficient and faster symmetric key algorithm, while the symmetric key itself is made secure with PKC [6,7].

2.2 Certificateless Hybrid Signcryption

The hybrid signcryption technique was first introduced by Dent et al. [6,7], which is secure against insider and outsider adversaries. Later, Li et al. [15] initially pre-

sented the certificateless-based hybrid signcryption. Within the framework of IoT, a certificateless-based hybrid signcryption technique was introduced by Wu et al. [34] that uses the PKC to compute a symmetric signcryption key which is subsequently used to signcrypt the message. Although their technique ensures potential security requirements such as confidentiality, unforgeability, session security, and public verifiability under the hard assumptions of CDH and Decisional Bilinear Diffie-Hellman (DBDH), it utilises higher computations by using BP operations.

In the same manner, a certificateless-based hybrid signcryption technique for WSNs was presented by Yin et al. [36]. Similarly, a certificateless-based hybrid signcryption designed for IoT applications was introduced by Zhang et al. [39]. Their approach aims to ensure confidentiality as well as unforgeability under the hard assumptions of DL, CDH, DBDH, and BDH. Furthermore, Gong et al. [10] introduced a hybrid signcryption technique in certificateless settings designed for a lightweight and secure IoT environment that provides data integrity, confidentiality, and unforgeability. While this technique proves security under CDH and DBDH assumptions, it uses expensive BP operations for key setup and generation. In the context of VANETs, Hongzhen et al. [12] introduced a certificateless-based signcryption technique and proved security under the ROM using the hard assumptions of ECDL and CDH. Furthermore, Niu et al. [21] presented a hybrid signcryption in 2017 for multi-messages and multi-recipients simultaneously in a heterogeneous environment. Their technique demonstrates *euf-cma* and *ind-cca* security and relies upon the DBDH and variants of DBDH and Computational BDH hard assumptions. In 2020, Khan et al. [14] proposed a certificateless key-encapsulated signcryption technique for flying ad-hoc networks. This technique deploys a key-agreement protocol between the sender-receiver communication while ensuring confidentiality and unforgeability by evaluating security against *ind-cca* and *euf-cma* using ROM. Despite proving security and efficiency using Hyper-ECDL and Hyper-ECDH assumptions, the scheme lacks anonymity, and the computational efficiency is evaluated only for signcrypted text.

In 2021, Zhou et al. [42] introduced anonymous signcryption techniques for multiple-message and multiple receivers in certificateless settings. The proposed scheme encrypts multiple messages by first computing a secret key and then signcrypting each message. Each receiver unsigncrypts the message, achieving confidentiality, unforgeability, anonymity, forward security, and prevention against replay attacks. The security is proven under the ROM using DL and CDH assumptions. However, for n receivers, the scheme exhibits higher computational costs. Additionally, as the scheme utilizes symmetric key encryption, it does not demonstrate how non-repudiation can be achieved. Niu et al. [23], in 2022, introduced a broadcast WSN signcryption technique based on certificateless PKC. The main objectives of their scheme are to ensure confidentiality and data integrity during transmission while ensuring the receiver's anonymity under the ECDH and ECDL. Their approach relies on a trusted third authority to perform computations for encryption, assuming the constant availability of this entity. However, this assumption may not hold in certain scenarios where the trusted third party becomes offline, making the scheme unrealistic. Furthermore, their technique has a higher computational overhead than our proposed technique. Table 1 provides a comprehensive overview of the schemes along with their limitations.

Table 1. Existing Schemes and Limitations.

Schemes	Limitations
Gong et al. [10]	The scheme utilises BP for initialisation phase and key generation. It does not provide anonymity or evaluate the non-repudiation security requirement
Hongzhen et al. [12]	While the scheme addresses the required security, it does so at the expense of increased computational complexity compared to our AMCLHS technique
Khan et al. [14]	The scheme proves security and efficiency using Hyper-ECDL (HECDL) and Hyper-ECDH (HECDH) assumptions. However, it lacks anonymity, and the computational efficiency is evaluated only for signcrypted text while unsigncryption is not evaluated
Niu et al. [21]	The scheme has a higher computational cost. For instance, with the increase in the number of receivers, the computational cost of signcryption grows quadratically, while that of unsigncryption increases linearly. Additionally, it does not evaluate forward security and non-repudiation security requirements
Niu et al. [23]	Their approach relies on a trusted third authority to perform computations for encryption, assuming the constant availability of this entity. However, this assumption may not hold in certain scenarios where the trusted third party becomes offline, making the scheme unrealistic. Additionally, it does not provide forward security or non-repudiation security requirements or their evaluation
Wu et al. [34]	The scheme has a significant computational overhead due to BP operations. Additionally, non-repudiation and anonymity security requirements are not assessed
Yin et al. [36]	The scheme uses expensive BP operations, resulting in high computational cost. Additionally, it does not provide anonymity, evaluation of non-repudiation, or forward security
Zhang et al. [39]	The scheme has a higher computational cost, with unsigncryption cost increasing linearly with the number of receivers, and it does not address non-repudiation
Zhou et al. [42]	For n receivers, the scheme exhibits higher computational costs. Additionally, as the scheme utilises symmetric key encryption, it does not demonstrate and evaluate how non-repudiation can be achieved hence lacking this security requirement

3 Preliminaries and Assumptions

1. **Elliptic-Curve Computational Diffie-Hellman (EC-CDH):** This assumption is based on [4,31].

Definition 1. *Given* $(P, xP, yP) \in \mathbb{G}$*, for* $x, y \in \mathbb{Z}_q^*$*, the EC-CDH assumption remain intact if it is computationally impractical for any Probabilistic Polynomial-Time (PPT) adversary to find a solution to compute* xyP*.*

2. **Elliptic-Curve Discrete Logarithm (ECDL):** The ECDL assumption is taken from [4,31].

Definition 2. *This assumption holds, if given* P *and* $Q \in \mathbb{G}$*, it is hard for any PPT adversary to find* $x \in \mathbb{Z}_q^*$ *with non-negligible probability such that* $Q = xP$*.*

3. **The multi-recipient Key Encapsulation Mechanism** (mKEM) **and Data Encapsulation Mechanism** (DEM)**:** The concept of mKEM initially was presented by N.P Smart [29] and operates similarly to the KEM. It chooses as input the public keys $(\mathsf{pk}_{\mathsf{r}_i})$ of multiple receivers for $1 \leq \mathsf{i} \leq t$ where $t < n$. It outputs a symmetric session key $(\mathcal{K}_{\mathsf{sy}})$ and a corresponding encapsulation (C) of that key $\mathcal{K}_{\mathsf{sy}}$.

Definition 3. *The structure of* mKEM *is based on [29,31] and is defined as follows:*

(a) mKEM*: It comprises four algorithms, which are outlined below:* (Setup, KeyGen, mKEM.Encaps, mKEM.Decaps)*:*
   - Setup*: Taking* 1^λ *as security primitive, the algorithm generates* PP*.*
   - KeyGen*: On input* PP*, it generates each user's key pair* $(\mathsf{pk}, \mathsf{sk})$*.*
   - mKEM.Encaps*: Taking the* PP *and each receiver's public keys* $(\mathsf{pk}_{\mathsf{r}_i})$ *as input, where* $1 \leq \mathsf{i} \leq t$*, it generates a symmetric session key* $(\mathcal{K}_{\mathsf{sy}})$ *and a corresponding encapsulation* C_1 *of* $\mathcal{K}_{\mathsf{sy}}$*. The* $\mathcal{K}_{\mathsf{sy}}$ *will be later utilized in* DEM*.*
   - mKEM.Decaps*: On input the* PP*, an encapsulation* C_1*, and private key* $(\mathsf{sk}_{\mathsf{r}_i})$ *of receiver, it generates* $\mathcal{K}_{\mathsf{sy}}$*. The correctness of this algorithm hold if* $\mathcal{K}_{\mathsf{sy}} = \mathsf{mKEM.Decaps}(\mathsf{PP}, \mathsf{sk}_{\mathsf{r}_i}, \mathsf{C}_1)$*.*

(b) DEM*: This comprises two algorithms* $(\mathsf{Enc}_{\mathcal{K}_{\mathsf{sy}}}, \mathsf{Dec}_{\mathcal{K}_{\mathsf{sy}}})$ *[21,31] which are defined as follows:*
   - $\mathsf{Enc}_{\mathcal{K}_{\mathsf{sy}}}$*: Taking* $(\mathcal{K}_{\mathsf{sy}}, \mathsf{msg})$ *as input, it generates a ciphertext* C_2*.*
   - $\mathsf{Dec}_{\mathcal{K}_{\mathsf{sy}}}$*: On input* $(\mathcal{K}_{\mathsf{sy}}, \mathsf{C}_2)$*, it computes* msg'*. The* DEM *algorithm is correct if* $\mathsf{msg}' = \mathsf{msg}$*.*

4. KEM-DEM **Hybrid Signcryption:**

Definition 4. *The* KEM-DEM *hybrid signcryption technique's structure is based on [6,31]. It comprises four algorithms:* $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Gen} - \mathsf{Enc}, \mathsf{Dec} - \mathsf{Ver})$*:*

(a) Setup*: Taking* 1^λ *as a security primitive, it generates* PP*.*
(b) KeyGen*: On input* PP*, it generates sender's private and public key pair* $(\mathsf{pk}_\mathsf{s}, \mathsf{sk}_\mathsf{s})$ *and receiver's key pair* $(\mathsf{pk}_\mathsf{r}, \mathsf{sk}_\mathsf{r})$*.*
(c) Gen − Enc*:* Gen − Enc *consists of two following sub-algorithms:*

- Encaps*: Taking* $(\mathsf{PP}, \mathsf{sk_s}, \mathsf{pk_r}, \mathsf{msg})$ *as input, it generates a symmetric session key* $\mathcal{K}_{\mathsf{sy}}$ *and its corresponding encapsulation* $\mathsf{C_1}$.
- $\mathsf{Enc}_{\mathcal{K}_{\mathsf{sy}}}$*: This algorithm takes as input* $\mathcal{K}_{\mathsf{sy}}$ *to computes* $\mathsf{C_2}$ *and outputs a ciphertext* $\mathcal{CT} = (\mathsf{C_1}, \mathsf{C_2})$.

(d) Dec − Ver*: This phase consists of following three algorithms run by the sender:*

- Decaps*: Taking* $(\mathsf{sk_r}, \mathsf{C_1})$ *as input, this algorithm generates* $\mathcal{K}_{\mathsf{sy}}$. *The sender terminates the process if* $\mathcal{K}_{\mathsf{sy}} = \perp$. *Otherwise, the receiver proceeds to the next algorithmic step.*
- $\mathsf{Dec}_{\mathcal{K}_{\mathsf{sy}}}$*: Taking* $(\mathsf{C_2}, \mathcal{K}_{\mathsf{sy}})$ *as input, it outputs* msg. *The receiver terminates if* $\mathsf{msg} = \perp$. *Otherwise, the receiver proceeds to run the next step.*
- Ver*: On input* $(\mathsf{pk_s}, \mathsf{msg}, \mathsf{C_1})$*, this algorithm evaluates if the result is valid or not. If yes, it outputs* msg*, else* $\perp$.

4 AMCLHS Framework

4.1 Framework

The AMCLHS technique follows the framework of [31]. It comprises of following participants: Registration Authority (RA), KGC, and n number of users, including a sender vehicle and multiple receiver vehicles such as $n = (\mathcal{P}_{\mathsf{ID_s}}, \{\mathcal{P}_{\mathsf{ID_1}}, \ldots, \mathcal{P}_{\mathsf{ID_{r_i}}}, \ldots, \mathcal{P}_{\mathsf{ID_{r_t}}}\})$. Every user vehicle is integrated with secure onboard units (OBUs) along with Tamper-Proof Devices (TPDs) that provide communication and secure storage, respectively. Furthermore, each vehicle generates its secret value (sv) and a corresponding public key (pk). Assuming a sender vehicle having a pseudo-identity $\mathcal{P}_{\mathsf{ID_s}}$ sends a msg of variable-length to t targeted receiver vehicles with a pseudo-identity $\mathcal{P}_{\mathsf{ID_{r_i}}}$ where $1 \leq \mathsf{i} \leq t$. The framework is shown in Fig. 1. Each participant's responsibility is as follows:

- **KGC:** This entity is a trusted authority within the system architecture. Its responsibilities include computing public parameters (PP), its own master public key and master private key pair $(\mathsf{mpk}, \mathsf{msk})$, and each vehicle user's partial private key (ppk) that is participating in a broadcast vehicular communication.
- **RA:** It is a partially-trusted entity. It is responsible for generating its private and public key pairs, $(\mathsf{sk_{RA}}, \mathsf{pk_{RA}})$. RA also plays a role in user vehicle registration, ID verification, and $\mathcal{P}_{\mathsf{ID}}$ assigning. We assume there is mutual trust between KGC and RA.
- **Sender Vehicle:** Let a sending vehicle with $\mathcal{P}_{\mathsf{ID_s}}$ intend to send an emergency msg. It takes public key set $(\mathsf{pk_{r_i}})$ of designated receiver vehicle to encrypts a msg, and utilizes its private key $(\mathsf{sk_s})$ to sign the msg. The sender vehicle broadcasts the ciphertext $(\mathcal{CT})$ to t targeted receiving vehicles.
- **Receiver Vehicle:** The intended receiving vehicle containing $(\mathcal{P}_{\mathsf{ID_{r_i}}}, \mathsf{sk_{r_i}})$, performs the $\mathcal{CT}$ decryption to obtain an emergency msg, and uses public key of the sending vehicle $(\mathsf{pk_s})$ to verify the signature on msg.

4.2 Definition of AMCLHS

Our technique employs a hybrid methodology that integrates together mKEM and DEM techniques. Prior to performing a signcryption operation on a msg, RA assesses the real-identity ($\mathcal{R}_{\mathsf{ID}}$) of each vehicle. If it is valid, the RA registers and issues a $\mathcal{P}_{\mathsf{ID}}$ to each vehicle user. To implement the signcryption algorithm, our technique first employs mKEM, which takes receiving vehicle's public key sets as input to output a symmetric session key $\mathcal{K}_{\mathsf{sy}}$ along with its encapsulation C_1. Further, to compute the signature (σ), the sending vehicle chooses its private key, which is contained within C_1. The σ is then verified in unsigncryption phase as described in Definition 5. Subsequently, the combination of $\mathcal{K}_{\mathsf{sy}}$ and DEM is utilized to perform encryption on msg symmetrically, resulting in the C_2. This C_2 is further combined with C_1 to form the signcrypted ciphertext pair $\mathcal{CT} = (\mathsf{C}_1, \mathsf{C}_2)$. To decrypt the $\mathcal{CT}$, the receiving vehicle first de-capsules the C_1 using its private key, obtaining $\mathcal{K}_{\mathsf{sy}}$. Next, C_2 is decrypted using $\mathcal{K}_{\mathsf{sy}}$ to obtain the msg. Finally, the receiving vehicle performs the σ verification by applying the algorithm Ver. In this algorithm, the sending vehicle takes its public key and C_1 as input to verify the signature and outputs if it is valid or not. Thus, our technique presents a cost-efficient and reliable framework to signcrypt and unsigncrypt a broadcast msg by integrating a unique hybrid model that integrates both symmetric and asymmetric key strategies.

Definition 5. *In AMCLHS technique, the sending vehicle having $\mathcal{P}_{\mathsf{ID}_s}$ as identity, broadcasts a* msg *of flexible size to t-targeted receiving vehicles shown as $\mathcal{P}_{\mathsf{ID}_{r_i}}$ where $1 \leq \mathsf{i} \leq t$. Our scheme is based on Definitions 3 and 4 and follows the structure of [31]. Our technique is composed of eight polynomial time algorithms which are defined below:*

1. **Setup:** Taking security parameter (1^{λ}) as input, the KGC outputs $(\mathsf{PP}, \mathsf{msk}, \mathsf{mpk})$. Additionally, RA computes the $\mathsf{sk}_{\mathsf{RA}}, \mathsf{pk}_{\mathsf{RA}}$ pair.
2. **Pseudo-Identity:** On input, Real Identity $(\mathcal{R}_{\mathsf{ID}}, \mathcal{P}_{\mathsf{ID}_1})$ and $\mathsf{pk}_{\mathsf{RA}}$, in this algorithm, RA outputs a $\mathcal{P}_{\mathsf{ID}}$.
3. **Partial Private Key:** Taking $(\mathsf{mpk}, \mathsf{msk})$ as input in this algorithm, KGC computes the partial private key (ppk) for each $\mathcal{P}_{\mathsf{ID}}$.
4. **Secret Value:** Taking $\mathcal{P}_{\mathsf{ID}}$ as input, each participating vehicle outputs a secret value (sv).
5. **Private Key:** In this algorithm, each user vehicle takes as input the $(\mathsf{ppk}, \mathsf{sv})$, and generates the sk.
6. **Public Key:** Each participating vehicle runs this algorithm to generate pk by taking sv as input.
7. **Signcryption:** To perform signcryption operation on msg and produce a $\mathcal{CT}$, the sending vehicle executes two steps in this algorithm. Phase-1 runs mKEM.Encaps algorithm and Phase-2 executes $\mathsf{Enc}_{\mathcal{K}_{\mathsf{sy}}}$ algorithm which is based on Definition 3. Each phase is described below:
   - Phase-1 (mKEM.Encaps): In this algorithm, the sender takes $(\mathsf{PP}, \mathsf{sk}_s, \mathsf{msg}, \mathsf{pk}_{r_i})$ where $1 \leq \mathsf{i} \leq t$ as input and generates $(\mathsf{C}_1, \mathcal{K}_{\mathsf{sy}})$.
   - Phase-2 ($\mathsf{Enc}_{\mathcal{K}_{\mathsf{sy}}}$): Taking as input $(\mathcal{K}_{\mathsf{sy}}, \mathsf{msg})$, it outputs C_2 and sets $\mathcal{CT} = (\mathsf{C}_1, \mathsf{C}_2)$.

8. **Unsigncryption**: This algorithm consists of three steps executed by a designated receiving vehicle for unsigncrypting and retrieving the msg. Phase-1 is consisted of mKEM.Decaps; in Phase-2, $\mathsf{Dec}_{\mathcal{K}_{\mathsf{sy}}}$ is executed; and Phase-3 involves the Ver algorithm as defined in Definition 3.
   - Phase-1 (mKEM.Decaps): On input $(\mathsf{sk}_{\mathsf{r}_i}, \mathsf{C}_1)$, it outputs $\mathcal{K}_{\mathsf{sy}}$.
   - Phase-2 ($\mathsf{Dec}_{\mathcal{K}_{\mathsf{sy}}}$): Taking $(\mathcal{K}_{\mathsf{sy}}, \mathsf{C}_2)$ as input, this algorithm generates msg'. The msg is rejected by receiving vehicle if $\mathsf{msg}' \neq \mathsf{msg}$. Otherwise, it proceeds to the next step.
   - Phase-3 (Ver): On input $(\mathsf{msg}', \mathsf{C}_1, \mathsf{pk}_{\mathsf{s}})$, the receiving vehicle verifies the σ. If valid, it accepts the msg, or else returns $\perp$.

5 Security Model

To achieve confidentiality and unforgeability, we describe the definitions of security in terms of *ind-cca2* and *euf-cma*, as per [31]. We particularly design Game-I to prove *ind-cca2-I* security against Type-I adversary ($\mathcal{A}_{\mathrm{I}}$) and *ind-cca2-II* against Type-II ($\mathcal{A}_{\mathrm{II}}$) adversary, respectively. Furthermore, Game-II, we evaluate *euf-cma-I* security against $\mathcal{A}_{\mathrm{I}}$ and *euf-cma-II* security against $\mathcal{A}_{\mathrm{II}}$. Each game in the following sub-sections.

1. $\mathcal{A}_{\mathrm{I}}$: This adversary is considered an honest-but-curious entity who does not have the knowledge of msk but is able to replace the pk of a an honest ID with the key of her/his own preference. However, $\mathcal{A}_{\mathrm{I}}$ is prohibited from making a ppk query (q_{ppk}) for any of the honest and targeted ID.
2. $\mathcal{A}_{\mathrm{II}}$: $\mathcal{A}_{\mathrm{II}}$, also identified as malicious KGC, is not allowed to execute any public key replace (q_{pr}) query for any target identity. Moreover, it is prohibited to run a secret value (sv) extract (q_{sv}) queries. Further, if the q_{pr} is placed for the same ID, then the q_{sv} for the same ID is not permitted.

5.1 Game-I

This game is composed of three steps where a Challenger ($\mathcal{C}$) interacts with $\mathcal{A}$. The $\mathcal{A}$ consistently sends a number of queries (polynomially bounded), including hash queries, secret key queries, and public key queries. In the end, the $\mathcal{C}$ receives a pair of targeted messages $(\mathsf{msg}_0, \mathsf{msg}_1)$ from $\mathcal{A}$. $\mathcal{C}$ responds by randomly choosing $\beta \in \{0,1\}^*$ and generating a challenge ($\mathcal{CT}^*$). The $\mathcal{C}$ sends the $\mathcal{CT}^*$ to $\mathcal{A}$. In response, $\mathcal{A}$ sends $\beta' \in \{0,1\}^*$. In the end, $\mathcal{A}$ wins Game-I, if $\beta = \beta'$. The security model of our technique is further described in Definition 6.

Definition 6. *To prove ind-cca2, the algorithm requires that any PPT $\mathcal{A}$ cannot distinguish between ciphertexts. Therefore, our security games achieve confidentiality by defining ciphertext indistinguishability. The probability that the $\mathcal{A}$ will win the games is defined as the advantage of $\mathcal{A}$ [31].*

1. ***Phase-1***: In this phase, $\mathcal{A}$ submits a number of hash q_{H_l} queries for $l = \{1,2,3\}$. The responses are recorded by $\mathcal{C}$ by keeping a list L_l of q_{H_l}.

- **Setup**: $\mathcal{C}$ computes $(\mathsf{PP}, \mathsf{msk}, \mathsf{mpk}, \mathsf{sk_{RA}}, \mathsf{pk_{RA}})$ and sends it to $\mathcal{A}$. The, $\mathcal{A}$ chooses t target $\mathcal{P}_{\mathsf{ID}_{r_i}}$ where $1 \leq \mathsf{i} \leq t$.

2. ***Phase-2***: In the Phase-2, the $\mathcal{A}$ asks further queries under the conditions described in Sect. 5. These queries are: public key retrieve (q_{pk}), partial private key (q_{ppk}), secret value extract (q_{sv}), public key replace (q_{pr}), signcryption (q_{sc}), and unsigncryption (q_{usc}). The $\mathcal{C}$ keeps void list L_{pk} to save the secret and public key values received to use later. The responses by $\mathcal{C}$ are described below:
   - q_{pk}: When $\mathcal{C}$ receives such a query for $\mathcal{P}_{\mathsf{ID}}$, it searches L_{pk} for corresponding pk. If not found, $\mathcal{C}$ executes the algorithm of secret value, as in the main scheme, to output a sv for corresponding $\mathcal{P}_{\mathsf{ID}}$ and proceeds to compute pk and provide it to $\mathcal{A}$.
   - q_{ppk}: On such a query, the $\mathcal{C}$ verifies if $\mathcal{P}_{\mathsf{ID}} = \mathcal{P}^*_{\mathsf{ID}}$. If yes, it aborts. Otherwise, it fetches the L_{pk} to retrieve ppk. If not found, $\mathcal{C}$ executes the ppk algorithm and sends it back. $\mathcal{C}$ upgrades L_{pk}.
   - q_{sv}: Upon receiving q_{sv}, the $\mathcal{C}$ fetches L_{pk} to check the corresponding sv. If not found, $\mathcal{C}$ executes q_{pk} and sends back sv.
   - q_{pr}: On such a query for $\mathcal{P}_{\mathsf{ID}}$, pk is substituted by pk'. The $\mathcal{C}$ upgrades the L_{pk}.
   - q_{sc}: Upon receiving such a query with input $(\mathsf{msg}, \mathcal{P}_{\mathsf{ID}_s}, \mathcal{P}_{\mathsf{ID}_{r_i}})$ by $\mathcal{C}$, it evaluates if $\mathcal{P}_{\mathsf{ID}_{r_i}} = \mathcal{P}^*_{\mathsf{ID}}$. If yes, it executes the signcryption to output $\mathcal{CT}$. Else, $\mathcal{C}$ runs the usual signcryption operation by fetching values from L_{pk}.
   - q_{usc}: On such a query with an input $(\mathcal{CT}, \mathcal{P}_{\mathsf{ID}_s}, \mathcal{P}_{\mathsf{ID}_{r_i}})$, $\mathcal{C}$ examines if $\mathcal{P}_{\mathsf{ID}_{r_i}} = \mathcal{P}^*_{\mathsf{ID}}$. If yes, $\mathcal{C}$ runs the unsigncryption to return msg. Else, $\mathcal{C}$ executes the usual unsigncryption by fetching values from L_{pk}.
3. ***Challenge***: The $\mathcal{A}$ sends a target $(\mathsf{msg}_0, \mathsf{msg}_1)$. The $\mathcal{C}$ selects $\beta \in \{0,1\}^*$ at random, computes $\mathcal{CT}^*$, and returns the $\mathcal{CT}^*$.
4. ***Phase-3***: In this phase, $\mathcal{A}$ proceeds with more queries with the restriction that $\mathcal{CT}^*$ is prohibited to be presented in q_{usc}.
5. ***Guess***: Lastly, $\mathcal{A}$ replies with guess bit $\beta' \in \{0,1\}^*$. In the end, $\mathcal{A}$ wins the Game-I if $\beta = \beta'$. The $\mathcal{A}_{\mathrm{I}}$'s advantage of winning is denoted as:

$$Adv^{ind-cca2}_{\mathcal{A}_{\mathrm{I}}} = \mid \Pr[\beta = \beta'] - 1/2 \mid \tag{1}$$

   The $\mathcal{A}_{\mathrm{II}}$'s winning advantage is shown as:

$$Adv^{ind-cca2}_{\mathcal{A}_{\mathrm{II}}} = \mid \Pr[\beta = \beta'] - 1/2 \mid \tag{2}$$

5.2 Game-II

In Game-II, the $\mathcal{C}$ engages with $\mathcal{A}$ in two stages. Firstly, in each step, The $\mathcal{A}$ consistently sends a number of queries, including hash queries, secret key queries, and public key queries. Finally, $\mathcal{A}$ generates and returns the forged $\mathcal{CT}$. If unsigncryption executes and does not abort $(\perp)$, $\mathcal{A}$ wins Game-II. Further security description is shown in Definition 7.

Definition 7. *An AMCLHS technique proves euf-cma security against Type-I and Type-II adversaries. This attack states that if unsigncryption does not $\perp$ when given forged $\mathcal{CT}$ by an $\mathcal{A}$, the $\mathcal{A}$ wins the game. The advantage of winning Game-II for a PPT $\mathcal{A}$ is negligible [31].*

1. ***Phase-1***: In this phase, a number of hash q_{H_l} queries are submitted by the $\mathcal{A}$ where $\{l = 1, 2, 3\}$. To keep track of the responses, the $\mathcal{C}$ stores a list L_l of q_{H_l}.
   - **Setup**: $\mathcal{C}$ computes $(\mathsf{PP}, \mathsf{msk}, \mathsf{mpk}, \mathsf{sk}_{\mathsf{RA}}, \mathsf{pk}_{\mathsf{RA}})$ and responds with PP to $\mathcal{A}$. Next, $\mathcal{A}$ chooses a $\mathcal{P}^*_{\mathsf{ID}_s}$ as a target identity.
2. ***Phase-2***: First, $\mathcal{A}$ submits some queries according to the predefined rules in Sect. 5. These queries are q_{pk}, q_{ppk}, q_{pr}, q_{sv}, q_{sc}, and q_{usc} (description of these queries is given in Phase-2 of Game-I in Definition 5.1). To keep the pk and sv information, $\mathcal{C}$ maintains an empty list L_{pk} in the initial step.
3. ***Forgery***: In this step, $\mathcal{A}$ generates a forged $\mathcal{CT}$ against the $\mathcal{P}^*_{\mathsf{ID}_s}$. If unsigncryption does not abort and does not return $\perp$, the $\mathcal{A}$ wins the Game-II.

6 Anonymous Multi-receiver Certificateless Hybrid Signcryption (AMCLHS)

The AMCLHS technique follows the framework described in Definition 5 and follows the extension of [31]. Here, we present the construction of the technique, and the model of our scheme is explained in Fig. 1.

1. **Setup**: Taking 1^λ as a security parameter, KGC begins the system operation. The KGC selects a group $(\mathbb{G})$ with large prime order (q), established from an elliptic curve (E) and defined over a finite field $\mathbb{F}_q$. It chooses a $P \in \mathbb{G}$ as a generator point and creates hash functions. The hash function are defined as: $H_0 : \{0,1\}^\ell \times \mathbb{G} \rightarrow \mathbb{G}$, where ℓ denotes a positive integer, $H_1 : \{0,1\}^* \times \mathbb{G} \rightarrow \mathbb{G}$, $H_2 : \mathbb{G} \rightarrow \{0,1\}^k$, where k indicates the size of the plaintext, and $H_3 : \{0,1\}^* \times \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}$.
   Next, the KGC sets $\mathsf{PP} = \{\mathbb{G}, E, P, q, H_0, H_1, H_2, H_3\}$. It generates $\mathsf{msk} = x_0$ where $x_0 \in \mathbb{Z}^*_q$ and correspondingly computes $\mathsf{mpk} = x_0 P$. The, it publishes $(\mathsf{PP}, \mathsf{mpk})$ and keeps msk secure. Similarly, RA chooses $v \in \mathbb{Z}^*_q$ randomly and sets $\mathsf{sk}_{\mathsf{RA}} = v$ and outputs corresponding $\mathsf{pk}_{\mathsf{RA}} = vP$. It also makes $\mathsf{pk}_{\mathsf{RA}}$ public and keeps $\mathsf{sk}_{\mathsf{RA}}$ as a secret.
2. **Pseudo-Identity**: Each vehicle user and RA execute this algorithm as follows:
   - User: It selects a real-identity $\mathcal{R}_{\mathsf{ID}}$ and $\alpha \in \mathbb{Z}^*_q$ at random to compute $R = \alpha P$. On input $(\mathcal{R}_{\mathsf{ID}}, \alpha)$, the vehicle user calculates $\mathcal{P}_{\mathsf{ID}_1} = \mathcal{R}_{\mathsf{ID}} \oplus H_0(\alpha \mathsf{pk}_{\mathsf{RA}})$ and forwards $(R, \mathcal{P}_{\mathsf{ID}_1})$ to RA.
   - RA: It takes as input $(R, \mathcal{P}_{\mathsf{ID}_1})$, evaluates if $\mathcal{R}_{\mathsf{ID}} = \mathcal{P}_{\mathsf{ID}_1} \oplus H_0(Rv)$. If the verification of $\mathcal{R}_{\mathsf{ID}}$ is successful, RA validates the user vehicles' request for registration, calculates $\mathcal{P}_{\mathsf{ID}} = \mathcal{R}_{\mathsf{ID}} \oplus H_0(\alpha \mathsf{pk}_{\mathsf{RA}})$, and assigns $\mathcal{P}_{\mathsf{ID}}$ to each user vehicle. Otherwise, RA declines the registration request.
3. **Partial Private Key**: KGC takes as input $(\mathcal{P}_{\mathsf{ID}}, \mathsf{mpk}, \mathsf{msk})$ and calculates $Q_{\mathcal{P}_{\mathsf{ID}}} = H_1(\mathcal{P}_{\mathsf{ID}} \| \mathsf{mpk})$. Then it takes $Q_{\mathcal{P}_{\mathsf{ID}}}$ as input, to calculate ppk as $\mathsf{d} = x_0 Q_{\mathcal{P}_{\mathsf{ID}}}$.
4. **Secret Value**: The participating vehicles with $\mathcal{P}_{\mathsf{ID}}$ randomly selects $x \in \mathbb{Z}^*_q$ and computes $\mathsf{sv} = x$.

5. **Private Key**: Taking (d, x) as input, each participating vehicle computes as $\mathsf{sk} = (\mathsf{d}, x)$.
6. **Public Key**: Taking x as input, the user vehicles calculate $\mathsf{pk} = xP$.
7. **Signcryption**: The sending vehicle having $(\mathcal{P}_{\mathsf{ID}_s}, \mathsf{sk}_s)$ executes the phases below to signcrypt a msg and broadcasts $\mathcal{CT}$ to receiving vehicles with $(\mathcal{P}_{\mathsf{ID}_{r_i}}, \mathsf{pk}_{r_i})$ where $1 \leq \mathsf{i} \leq t$:
   - Phase-1 (mKEM-Encaps):
     (a) Selects $r \in \mathbb{Z}_q^*$ at random to calculates $U = rP$.
     (b) Taking its partial private key (d_s), secret value (x_s), pk_{r_i}, and $Q_{\mathcal{P}_{\mathsf{ID}_{r_i}}}$, the sending vehicle computes $Y_{1_i} = \mathsf{d}_s Q_{\mathcal{P}_{\mathsf{ID}_{r_i}}}$ and $Y_{2_i} = x_s \mathsf{pk}_{r_i}$.
     (c) Calculates $\delta_i = (Y_{1_i} Y_{2_i})$ and $\mathcal{K}_{sy} = H_2(\delta_i)$.
     (d) Calculates $\rho_i = H_3(\mathsf{msg}, \delta_i, \mathsf{pk}_s, \mathsf{pk}_{r_i})$ and computes $\sigma_i = r^{-1}(\rho_i \| w \mathsf{d}_s x_s)$ where $w = \mathrm{x}_U \mod q$ represents U's x-coordinate. Then it defines $\mathsf{C}_{1_i} = (\rho_i, \sigma_i)$ and results $(\mathsf{C}_{1_i}, \mathcal{K}_{sy})$.
   - Phase-2 ($\mathsf{Enc}_{\mathcal{K}_{sy}}$):
     (a) In this step, the sending vehicle calculates $\mathsf{C}_{2_i} = \mathsf{Enc}_{\mathcal{K}_{sy}}(\mathsf{msg})$. Sets $\mathcal{CT}_i = (\mathsf{C}_{1_i}, \mathsf{C}_{2_i})$ and broadcasts it to t receivers.
8. **Unsigncryption**: The intended receiving vehicle with $\mathcal{P}_{\mathsf{ID}_{r_i}}$ takes as input $(\mathsf{sk}_{r_i}, \mathsf{pk}_s)$ for i^{th} receiver and runs the following algorithms in each phase to unsigncrypt $\mathcal{CT}_i$ and retrieve msg:
   - Phase-1 (mKEM-Decaps):
     (a) On input $(x_{r_i}, \mathsf{d}_{r_i})$, it computes $Y'_{1_i} = \mathsf{d}_{r_i} Q_{\mathcal{P}_{\mathsf{ID}_s}}$ and $Y'_{2_i} = \mathsf{pk}_s x_{r_i}$.
     (b) Calculates $\delta'_i = (Y'_{1_i} Y'_{2_i})$ and $\mathcal{K}_{sy} = H_2(\delta'_i)$. It aborts if $\mathcal{K}_{sy} = \perp$; otherwise, it decrypts msg as follows:
   - Phase-2 ($\mathsf{Dec}_{\mathcal{K}_{sy}}$):
     (a) Computes $\mathsf{msg}' = \mathsf{Dec}_{\mathcal{K}_{sy}}(\mathsf{C}_{2_i})$. The receiver verifies σ_i, if $\mathsf{msg}' = \mathsf{msg}$, else returns $\perp$.
   - Phase-3 (Ver):
     (a) This steps takes $(\mathsf{C}_{1_i}, \mathsf{pk}_s)$ as input and generates $\rho'_i = H_3(\mathsf{msg}', \delta'_i, \mathsf{pk}_s, \mathsf{pk}_{r_i})$. and evaluates if $\rho'_i = \rho_i$. If yes, it verifies σ_i by assessing if $U = rP$ and $w' = x_U \mathrm{mod}\ q$. If $w' = w$, it will accept the signcryption on msg otherwise, returns $\perp$.

Correctness

1. $\mathcal{R}_{\mathsf{ID}} = \mathcal{P}_{\mathsf{ID}_1} \oplus H_0(Rv) = \mathcal{R}_{\mathsf{ID}} \oplus H_0(\alpha \mathsf{pk}_{\mathsf{RA}}) \oplus H_0(Rv) = \mathcal{R}_{\mathsf{ID}} \oplus H_0(Rv) \oplus H_0(Rv) = \mathcal{R}_{\mathsf{ID}}$
2. $Y_{1_i} = \mathsf{d}_s Q_{\mathcal{P}_{\mathsf{ID}_{r_i}}} = x_0 Q_{\mathcal{P}_{\mathsf{ID}_s}} Q_{\mathcal{P}_{\mathsf{ID}_{r_i}}} = \mathsf{d}_{r_i} Q_{\mathcal{P}_{\mathsf{ID}_s}}$
3. $Y_{2_i} = x_s \mathsf{pk}_{r_i} = x_s \mathsf{pk}_{r_i} = x_s x_{r_i} P = \mathsf{pk}_s x_{r_i} = \mathsf{pk}_s x_{r_i}$
4. Let $u_1 = \rho P$ and $u_2 = w \mathsf{pk}_s Z_{1_i} Q^{-1}_{\mathcal{P}_{\mathsf{ID}_{r_i}}}$
   $U = \sigma_i^{-1}(u_1 \| u_2) = \sigma_i^{-1}(\rho_i P \| w \mathsf{pk}_s Y_{1_i} Q^{-1}_{\mathcal{P}_{\mathsf{ID}_{r_i}}}) = \sigma_i^{-1}(\rho_i P \| w \mathsf{pk}_s \mathsf{d}_s Q_{\mathcal{P}_{\mathsf{ID}_{r_i}}} Q^{-1}_{\mathcal{P}_{\mathsf{ID}_{r_i}}}) = \sigma_i^{-1}(\rho_i P \| w x_s P \mathsf{d}_s) = \frac{(\rho_i P \| w x_s P \mathsf{d}_s)}{\sigma_i} = \frac{P}{r^{-1}} = rP.$

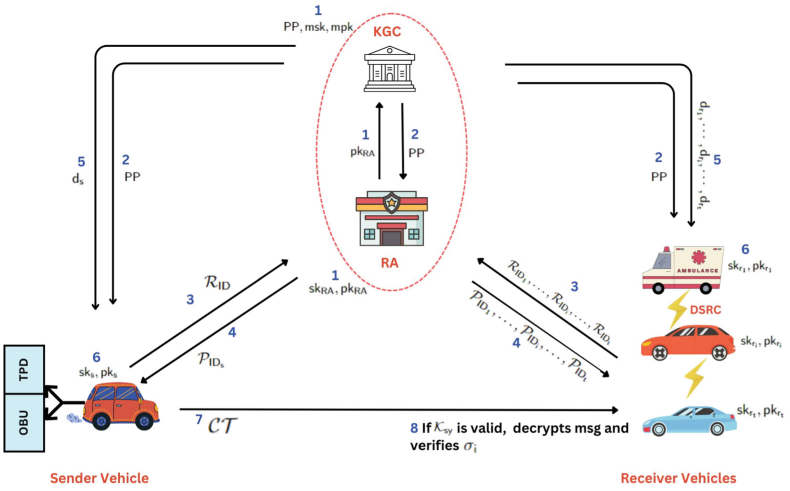


Fig. 1. Our Proposed AMCLHS Scheme.

7 Security Analysis

This section represents the security analysis of our AMCLHS technique which is according to the security model described in Sect. 5 and follows the criteria defined in [31]. Theorem 1 and Theorem 2 proves *ind-cca2-I* and *ind-cca2-II* security to ensure confidentiality, respectively and is based on Game-I in Definition 5.1. Subsequently, Theorem 3 and Theorem 4 defines *euf-cma-I* and *euf-cma-II* security, respectively, to prove unforgeability and they follow Game-II in Definition 5.2.

Confidentiality

Theorem 1. *The AMCLHS technique proves ind-cca2-I security under the hard assumptions of EC-CDH in ROM. If there is* $\mathcal{A}_{\mathrm{I}}$ *having a non-negligible advantage* (ϵ) *in winning the game, then there exists a* $\mathcal{C}$ *that can provide a solution to the EC-CDH with the non-negligible advantage* (ϵ') *[31].*

Proof. On input $(P, xP, yP) \in \mathbb{G}$ as an EC-CDH random parameter, the $\mathcal{C}$ must solve xyP as defined in Definition 1 after interaction with $\mathcal{A}_{\mathrm{I}}$ as shown below:

1. ***Phase-1***: $\mathcal{A}_{\mathrm{I}}$ submits a number of queries q (all queries are polynomially bounded). To keep a track of q_{H_l} responses, the $\mathcal{C}$ maintains a list L_l.
   - **Setup**: The $\mathcal{C}$ derives $\mathsf{PP} = \{\mathbb{G}, E, P, q, H_0, H_1, H_2, H_3\}$, defines $\mathsf{mpk} = \theta P$ and forwards $(\mathsf{PP}, \mathsf{mpk})$ to $\mathcal{A}_{\mathrm{I}}$. Then, $\mathcal{A}_{\mathrm{I}}$ chooses t- targeted receiving vehicles denoted by $\mathcal{P}^*_{\mathsf{ID}_i}$ where $1 \leq \mathsf{i} \leq t$.
   - H_1 **Query** (q_{H_1}): As a response to q_{H_1}, $\mathcal{C}$ assesses if $(Q_{\mathcal{P}_{\mathsf{ID}_i}}, \mathsf{mpk}, \mathcal{P}_{\mathsf{ID}_i})$ exists in L_1. If it does, it sends $Q_{\mathcal{P}_{\mathsf{ID}_i}}$ to $\mathcal{A}_{\mathrm{I}}$. Otherwise, if $\mathcal{P}_{\mathsf{ID}_i} \neq \mathcal{P}^*_{\mathsf{ID}_i}$, fixes $Q_{\mathcal{P}_{\mathsf{ID}_i}} = H_1(\mathcal{P}_{\mathsf{ID}_i} \| \mathsf{mpk})$ and if $\mathcal{P}_{\mathsf{ID}_i} = \mathcal{P}^*_{\mathsf{ID}_i}$, $\mathcal{C}$ selects $\mu \in \mathbb{Z}^*_q$, calculates $Q_{\mathcal{P}_{\mathsf{ID}_i}} = \mu P$, revises the tuple $(Q_{\mathcal{P}_{\mathsf{ID}_i}}, \mathsf{mpk}, \mathcal{P}_{\mathsf{ID}_i})$ in L_1 and forwards $Q_{\mathcal{P}_{\mathsf{ID}_i}}$ to $\mathcal{A}_{\mathrm{I}}$.

- H_2 **Query** (q_{H_2}): On q_{H_2}, $\mathcal{C}$ reviews if $(\mathcal{K}_{\mathsf{sy}}, \delta_{\mathsf{i}}, Y_{1_{\mathsf{i}}}, Y_{2_{\mathsf{i}}})$ is present in L_2. If yes, it replies with $\mathcal{K}_{\mathsf{sy}}$. Else, $\mathcal{C}$ selects $\mathcal{K}_{\mathsf{sy}} \in \{0,1\}^k$ at random, sets $(\mathcal{K}_{\mathsf{sy}}, \delta_{\mathsf{i}}, Y_{1_{\mathsf{i}}}, Y_{2_{\mathsf{i}}})$, and returns $\mathcal{K}_{\mathsf{sy}}$ to $\mathcal{A}_{\mathrm{I}}$.
- H_3 **Query** (q_{H_3}): In response to q_{H_3}, $\mathcal{C}$ assesses if $H_3(\mathsf{msg}, \delta_{\mathsf{i}}, \rho_{\mathsf{i}})$ is present in L_3. If it does, $\mathcal{C}$ sends ρ_{i} to $\mathcal{A}_{\mathrm{I}}$. Else, it randomly selects $\rho_{\mathsf{i}} \in \mathbb{Z}_q^*$, updates $H_3(\mathsf{msg}, \delta_{\mathsf{i}}, \rho_{\mathsf{i}})$ and replies to $\mathcal{A}_{\mathrm{I}}$ with ρ_{i}.

2. ***Phase-2***: In this phase, public and private key queries are submitted by $\mathcal{A}_{\mathrm{I}}$ including q_{pk}, q_{ppk}, q_{pr}, q_{sv}, and q_{usc}. The $\mathcal{C}$ keeps a list L_{pk} to record the responses and responds to these queries as shown below:
   - q_{pk}: In response to this query, $\mathcal{C}$ determines if pk_{i} is present in L_{pk} If yes, it forwards the same to $\mathcal{A}_{\mathrm{I}}$. Else, selects $x_{\mathsf{i}} \in \mathbb{Z}_q^*$, calculates $\mathsf{pk}_{\mathsf{i}} = x_{\mathsf{i}}P$, upgrades L_{pk} as $(\mathcal{P}_{\mathsf{ID}_{\mathsf{i}}}, -, \mathsf{pk}_{\mathsf{i}}, x_{\mathsf{i}})$ and responds to $\mathcal{A}_{\mathrm{I}}$ with pk_{i}.
   - q_{ppk}: Here, $\mathcal{C}$ will abort if $\mathcal{P}_{\mathsf{ID}_{\mathsf{i}}} = \mathcal{P}^*_{\mathsf{ID}_{\mathsf{i}}}$. Else, it checks L_{pk} for d_{i}. If it is present, returns it to $\mathcal{A}_{\mathrm{I}}$; if not, selects $Q_{\mathcal{P}_{\mathsf{ID}_{\mathsf{i}}}} = \mu P$ from L_1 and responds to $\mathcal{A}_{\mathrm{I}}$ with $\mathsf{d}_{\mathsf{i}} = \mathsf{mpk} Q_{\mathcal{P}_{\mathsf{ID}_{\mathsf{i}}}}$. Next, it upgrades $(\mathcal{P}_{\mathsf{ID}_{\mathsf{i}}}, \mathsf{d}_{\mathsf{i}}, \mathsf{pk}_{\mathsf{i}}, x_{\mathsf{i}})$ in L_{pk}.
   - q_{sv}: $\mathcal{C}$ determines if sv is present in L_{pk}; if yes, returns x_{i} to $\mathcal{A}_{\mathrm{I}}$. Otherwise, runs q_{pk} and sends x_{i} to $\mathcal{A}_{\mathrm{I}}$.
   - q_{pr}: $\mathcal{C}$ replaces the pk_{i} with $\mathsf{pk}'_{\mathsf{i}}$ for $\mathcal{P}_{\mathsf{ID}_{\mathsf{i}}}$ and upgrades L_{pk} as $(\mathcal{P}_{\mathsf{ID}_{\mathsf{i}}}, \mathsf{d}_{\mathsf{i}}, \mathsf{pk}'_{\mathsf{i}}, -)$.
   - q_{sc}: On q_{sc}, $\mathcal{C}$ determines if $\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}} = \mathcal{P}^*_{\mathsf{ID}_{\mathsf{i}}}$ or not. If not, it executes the usual signcryption operation by extracting values from L_{pk}. Else, it runs the signcryption as follows:
     - First, if pk_{i} was replaced previously, $\mathcal{A}_{\mathrm{I}}$ will give any other value of his/her choice.
     - Takes $r \in \mathbb{Z}_q^*$ at random to calculate $U = rP$.
     - Extracts $Q_{\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}}}$ from L_1 and calculates $Y_{1_{\mathsf{i}}} = \mathsf{d_s} Q_{\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}}}$, $Y_{2_{\mathsf{i}}} = x_{\mathsf{s}} \mathsf{pk}_{\mathsf{r_i}}$, $\delta_{\mathsf{i}} = (Y_{1_{\mathsf{i}}} Y_{2_{\mathsf{i}}})$, $\mathcal{K}_{\mathsf{sy}} = H_2(\delta_{\mathsf{i}})$, and upgrades L_2.
     - Calculates $\rho_{\mathsf{i}} = H_3(\mathsf{msg}, \delta_{\mathsf{i}}, \mathsf{pk_s}, \mathsf{pk_{r_i}})$ and upgrades L_3.
     - It calculates σ_{i}, $\mathsf{C}_{1_{\mathsf{i}}} = (\rho_{\mathsf{i}}, \sigma_{\mathsf{i}})$, $\mathsf{C}_{2_{\mathsf{i}}} = \mathsf{Enc}_{\mathcal{K}_{\mathsf{sy}}}(\mathsf{msg})$ and then sends $\mathcal{CT}_{\mathsf{i}} = (\mathsf{C}_{1_{\mathsf{i}}}, \mathsf{C}_{2_{\mathsf{i}}})$ to $\mathcal{A}_{\mathrm{I}}$.
   - q_{usc}: $\mathcal{C}$ determines if $\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}} = \mathcal{P}^*_{\mathsf{ID}_{\mathsf{i}}}$ or not. If not, it runs the usual unsigncryption. Else, it executes the unsigncryption as defined below:
     - Fetches L_2 and L_3 for $(\mathcal{K}_{\mathsf{sy}}, \delta'_{\mathsf{i}}, Y'_{1_{\mathsf{i}}}, Y'_{2_{\mathsf{i}}})$ and $H_3(\mathsf{msg}, \delta'_{\mathsf{i}}, \rho'_{\mathsf{i}})$.
     - If not found, it responds with "failure." Else, computes $\mathcal{K}_{\mathsf{sy}} \neq \perp$ and $\mathsf{msg}' = \mathsf{Dec}_{\mathcal{K}_{\mathsf{sy}}}(\mathsf{C}_{2_{\mathsf{i}}})$.
     - Evaluates if $\rho'_{\mathsf{i}} = \rho_{\mathsf{i}}$; if it satisfies, then assesses if $U = rP$ and $w' = \mathrm{x}_U \bmod q$ hold. If yes, it responds with msg; else, returns $\perp$.

3. ***Challenge***: $\mathcal{A}_{\mathrm{I}}$ selects a target message pair $(\mathsf{msg}_0, \mathsf{msg}_1)$ of equal length and sends it to $\mathcal{C}$ with $\mathcal{P}_{\mathsf{ID_s}}$ and a target $\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}}$. Moreover, the $\mathcal{A}_{\mathrm{I}}$ cannot ask q_{ppk} for the target $\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}}$. $\mathcal{C}$ aborts and returns $\perp$, if $\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}} \neq \mathcal{P}^*_{\mathsf{ID}_{\mathsf{i}}}$. Else, it selects $\beta \in \{0,1\}^*$ and proceeds as follows to compute a challenge $\mathcal{CT}^*_{\mathsf{i}}$:
   - Calculates $U^* = r^*P$ by taking $r^* \in \mathbb{Z}_q^*$ at random.
   - Calculates $Y^*_{1_{\mathsf{i}}} = \mathsf{d_s} Q_{\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}}}$, $Y^*_{2_{\mathsf{i}}} = x_{\mathsf{s}} \mathsf{pk}_{\mathsf{r_i}}$, $\delta^*_{\mathsf{i}} = (Y^*_{1_{\mathsf{i}}} Y^*_{2_{\mathsf{i}}})$ to generate $\mathcal{K}^*_{\mathsf{sy}} = H_2(\delta^*_{\mathsf{i}})$
   - First computes $\rho^*_{\mathsf{i}} = H_3(\mathsf{msg}, \delta^*_{\mathsf{i}}, \mathsf{pk_s}, \mathsf{pk_{r_i}})$, $\sigma^*_{\mathsf{i}} = r^{*-1}(\rho^*_{\mathsf{i}} \| w \mathsf{d_s} x_{\mathsf{s}})$ and generates $\mathsf{C}^*_{1_{\mathsf{i}}} = (\rho^*_{\mathsf{i}}, \sigma^*_{\mathsf{i}})$, $\mathsf{C}^*_{2_{\mathsf{i}}} = \mathsf{Enc}_{\mathcal{K}^*_{\mathsf{sy}}}(\mathsf{msg})$. Finally, it calculates $\mathcal{CT}^*_{\mathsf{i}} = (\mathsf{C}^*_{1_{\mathsf{i}}}, \mathsf{C}^*_{2_{\mathsf{i}}})$.

4. ***Phase-3***: More queries are issued by $\mathcal{A}_{\mathrm{I}}$ as per *Phase-1*; however, it is prohibited to ask the q_{ppk} for targeted $\mathcal{P}_{\mathsf{ID}_{\mathsf{r}_i}}$ and the q_{usc} for targeted $\mathcal{CT}^*_{\mathsf{i}}$.
5. ***Guess***: $\mathcal{A}_{\mathrm{I}}$ returns a guess bit $\beta' \in \{0,1\}^*$. In response, if $\beta' = \beta$, $\mathcal{C}$ outputs $\frac{\theta Y_{1_i} - \mathsf{d_i} r}{(\mathsf{d_s} - U)} = \theta\mu P$ using $\mathsf{mpk} = \theta P$ and $Q_{\mathcal{P}_{\mathsf{ID}_i}} = \mu P$ as a valid EC-CDH solution.

Finally, the $\mathcal{C}$ solves EC-CDH problem as $\theta\mu P$. The $\mathcal{C}'s$ advantage of winning *(ind-cca2-I)* game is evaluated by computing the probability of aborting the game in the following events:

1. In q_{ppk}, it aborts if $\mathcal{P}_{\mathsf{ID}_i} = \mathcal{P}^*_{\mathsf{ID}_i}$. The probability of abort is $\Pr(E_{q_{\mathsf{ppk}}}) = 1/q_{\mathsf{ppk}}$.
2. In q_{usc}, it ends due to invalid msg. The abort probability is $\Pr(E_{q_{\mathsf{usc}}}) = q_{\mathsf{usc}}/2^k$.
3. In the challenge phase, upon the adversary's query for $\mathcal{P}_{\mathsf{ID}_{\mathsf{r}_i}} \neq \mathcal{P}^*_{\mathsf{ID}_i}$, $\mathcal{C}$ ends the game. The abort probability is $\Pr(E_{q_{H_1}}) = (1 - 1/q_{H_1})$.

Moreover, the $\mathcal{C}$ fetches L_1 to retrieve $Q_{\mathcal{P}_{\mathsf{ID}_i}}$ and L_2 to retrieve Y_{1_i} and evaluates $\frac{\theta Y_{1_i} - \mathsf{d_i} r}{(\mathsf{d_s} - U)} = \theta\mu P$. The probability is $(1/q_{H_1} + 1/q_{H_2})$. Therefore, $\mathcal{C}'s$ advantage ϵ' of winning the game with probability is given as:

$$\epsilon' \geq \epsilon \left(\frac{1}{q_{H_1}} + \frac{1}{q_{H_2}}\right)\left(\frac{1}{q_{H_1}}\right)\left(1 - \frac{1}{q_{\mathsf{ppk}}}\right)\left(1 - \frac{q_{\mathsf{usc}}}{2^k}\right) \tag{3}$$

Theorem 2. *The AMCLHS technique proves ind-cca2-II security under the hard assumptions of EC-CDH in ROM. If there is $\mathcal{A}_{\mathrm{II}}$ having a non-negligible advantage (ϵ) in winning the game, then there exists a $\mathcal{C}$ that can provide a solution to the EC-CDH with the non-negligible advantage (ϵ') [31].*

The interaction of $\mathcal{A}_{\mathrm{II}}$ is similar to the game outlined in Theorem 1. However, it is prohibited from asking q_{pr} and q_{sv}. To obtain $\beta' = \beta$, $\mathcal{A}_{\mathrm{II}}$ have to solve EC-CDH. The $\mathcal{C}$ solves the EC-CDH as defined in the proof below.

Proof. On input $(P, xP, yP) \in \mathbb{G}$ as an EC-CDH random parameter, the $\mathcal{C}$ must solve xyP as defined in Definition 1 after interaction with $\mathcal{A}_{\mathrm{II}}$ as shown below:

1. ***Phase-1***: $\mathcal{A}_{\mathrm{II}}$ submits a number of queries q (all queries are polynomially bounded). To keep a track of q_{H_l} responses, the $\mathcal{C}$ maintains a list L_l.
   - **Setup**: The $\mathcal{C}$ derives $\mathsf{PP} = \{\mathbb{G}, E, P, q, H_0, H_1, H_2, H_3\}$, defines $\mathsf{mpk} = \theta P$ and forwards $(\mathsf{PP}, \mathsf{mpk})$ to $\mathcal{A}_{\mathrm{II}}$. Then, $\mathcal{A}_{\mathrm{II}}$ chooses t- targeted receiving vehicles denoted by $\mathcal{P}^*_{\mathsf{ID}_i}$ where $1 \leq \mathsf{i} \leq t$.
   - H_1 **Query** (q_{H_1}): As a response to q_{H_1}, $\mathcal{C}$ assesses if $(Q_{\mathcal{P}_{\mathsf{ID}_i}}, \mathsf{mpk}, \mathcal{P}_{\mathsf{ID}_i})$ exists in L_1. If it does, it sends $Q_{\mathcal{P}_{\mathsf{ID}_i}}$ to $\mathcal{A}_{\mathrm{II}}$. Otherwise, if $\mathcal{P}_{\mathsf{ID}_i} \neq \mathcal{P}^*_{\mathsf{ID}_i}$, fixes $Q_{\mathcal{P}_{\mathsf{ID}_i}} = H_1(\mathcal{P}_{\mathsf{ID}_i} \| \mathsf{mpk})$ and if $\mathcal{P}_{\mathsf{ID}_i} = \mathcal{P}^*_{\mathsf{ID}_i}$, $\mathcal{C}$ selects $\mu \in \mathbb{Z}^*_q$, calculates $Q_{\mathcal{P}_{\mathsf{ID}_i}} = \mu P$, revises the tuple $(Q_{\mathcal{P}_{\mathsf{ID}_i}}, \mathsf{mpk}, \mathcal{P}_{\mathsf{ID}_i})$ in L_1 and forwards $Q_{\mathcal{P}_{\mathsf{ID}_i}}$ to $\mathcal{A}_{\mathrm{II}}$.
   - $H_2(q_{H_2}), H_3(q_{H_3})$ **Query**: When $\mathcal{C}$ receives H_2 and H_3 queries, it examines if $(\mathcal{K}_{\mathsf{sy}}, \delta_{\mathsf{i}}, Y_{1_i}, Y_{2_i})$ and $H_3(\mathsf{msg}, \delta_{\mathsf{i}}, \rho_{\mathsf{i}})$ are present in L_2 and L_3. If yes, it responds to $\mathcal{A}_{\mathrm{II}}$ with $\mathcal{K}_{\mathsf{sy}}$ and ρ_{i}. Else, it randomly selects $\mathcal{K}_{\mathsf{sy}} \in \{0,1\}^k$ and $\rho_{\mathsf{i}} \in \mathbb{Z}^*_q$, adds in the lists $(\mathcal{K}_{\mathsf{sy}}, \delta_{\mathsf{i}}, Y_{1_i}, Y_{2_i})$, and $H_3(\mathsf{msg}, \delta_{\mathsf{i}}, \rho_{\mathsf{i}})$ and returns the values to $\mathcal{A}_{\mathrm{II}}$.
2. ***Phase-2***: $\mathcal{A}_{\mathrm{II}}$ submits further including q_{pk}, q_{sv}, and q_{usc}. The $\mathcal{C}$ keeps a list L_{pk} to record the responses and responds to these queries as shown below:

- q_{pk}: In response to this query, $\mathcal{C}$ determines if $\mathsf{pk_i}$ is present in L_{pk} If yes, it forwards the same to $\mathcal{A}_{\mathrm{II}}$. Else, selects $x_{\mathsf{i}} \in \mathbb{Z}_q^*$, calculates $\mathsf{pk_i} = x_{\mathsf{i}}P$, upgrades L_{pk} as $(\mathcal{P}_{\mathsf{ID_i}}, -, \mathsf{pk_i}, x_{\mathsf{i}})$ and responds to $\mathcal{A}_{\mathrm{II}}$ with $\mathsf{pk_i}$.
- q_{sv}: $\mathcal{C}$ determines if $\mathcal{P}_{\mathsf{ID_i}} = \mathcal{P}^*_{\mathsf{ID_i}}$. If yes, it aborts because $\mathcal{P}_{\mathsf{ID_i}}$ is a target identity. Else, it assesses if x_{i} is already present in L_{pk}. If yes, it returns x_{i} to $\mathcal{A}_{\mathrm{II}}$. Else, $\mathcal{C}$ runs q_{pk}, calculates $\mathsf{pk_i} = x_{\mathsf{i}}P$, upgrades L_{pk} as $(\mathcal{P}_{\mathsf{ID_i}}, \mathsf{d_i}, \mathsf{pk_i}, x_{\mathsf{i}})$ and sends x_{i} to $\mathcal{A}_{\mathrm{II}}$.
- q_{sc}: In q_{sc}, $\mathcal{C}$ responds to $\mathcal{A}_{\mathrm{II}}$ by taking values from L_{pk} as in Theorem 1.
- q_{usc}: In q_{usc}, $\mathcal{C}$ responds to $\mathcal{A}_{\mathrm{II}}$ by taking values from L_{pk} as in Theorem 1.

3. ***Challenge***: $\mathcal{A}_{\mathrm{II}}$ selects a target message pair $(\mathsf{msg}_0, \mathsf{msg}_1)$ of equal length and sends it to $\mathcal{C}$ with $\mathcal{P}_{\mathsf{ID_s}}$ and a target $\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}}$. Moreover, the $\mathcal{A}_{\mathrm{II}}$ cannot ask q_{sv} for the target $\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}}$. $\mathcal{C}$ aborts and returns $\perp$, if $\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}} \neq \mathcal{P}^*_{\mathsf{ID_i}}$. Else, it selects $\beta \in \{0,1\}^*$ and proceeds as follows to compute a challenge $\mathcal{CT}^*_{\mathsf{i}}$:
   - Calculates $U^* = r^*P$ by taking $r^* \in \mathbb{Z}_q^*$ at random.
   - Calculates $Y^*_{1_{\mathsf{i}}} = \mathsf{d_s} Q_{\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}}}$, $Y^*_{2_{\mathsf{i}}} = x_{\mathsf{s}} \mathsf{pk}_{\mathsf{r_i}}$, $\delta^*_{\mathsf{i}} = (Y^*_{1_{\mathsf{i}}} Y^*_{2_{\mathsf{i}}})$ to generate $\mathcal{K}^*_{\mathsf{sy}} = H_2(\delta^*_{\mathsf{i}})$
   - First computes $\rho^*_{\mathsf{i}} = H_3(\mathsf{msg}, \delta^*_{\mathsf{i}}, \mathsf{pk_s}, \mathsf{pk}_{\mathsf{r_i}})$, $\sigma^*_{\mathsf{i}} = r^{*-1}(\rho^*_{\mathsf{i}} \| w \mathsf{d_s} x_{\mathsf{s}})$ and generates $\mathsf{C}^*_{1_{\mathsf{i}}} = (\rho^*_{\mathsf{i}}, \sigma^*_{\mathsf{i}})$, $\mathsf{C}^*_{2_{\mathsf{i}}} = \mathsf{Enc}_{\mathcal{K}^*_{\mathsf{sy}}}(\mathsf{msg})$. Finally, it calculates $\mathcal{CT}^*_{\mathsf{i}} = (\mathsf{C}^*_{1_{\mathsf{i}}}, \mathsf{C}^*_{2_{\mathsf{i}}})$.
4. ***Phase-3***: More queries are issued by $\mathcal{A}_{\mathrm{II}}$ as per *Phase-1*; however, it is prohibited to ask the q_{sv} for targeted $\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}}$ and the q_{usc} for targeted $\mathcal{CT}^*_{\mathsf{i}}$.
5. ***Guess***: $\mathcal{A}_{\mathrm{II}}$ returns a guess bit $\beta' \in \{0,1\}^*$. In response, if $\beta' = \beta$, $\mathcal{C}$ outputs $\frac{\theta Y_{1_{\mathsf{i}}} - \mathsf{d_i} r}{(\mathsf{d_s} - U)} = \theta \mu P$ using $\mathsf{mpk} = \theta P$ and $Q_{\mathcal{P}_{\mathsf{ID_i}}} = \mu P$ as a valid EC-CDH solution.

Finally, the $\mathcal{C}$ solves EC-CDH problem as $\theta \mu P$. The $\mathcal{C}'s$ advantage of winning *(ind-cca2-II)* game is evaluated by computing the probability of aborting the game in the following events:

- In q_{sv}, it aborts if $\mathcal{P}_{\mathsf{ID_i}} = \mathcal{P}^*_{\mathsf{ID_i}}$. The probability of abort is $\Pr(E_{q_{\mathsf{sv}}}) = 1/q_{\mathsf{sv}}$.
- In q_{usc}, it ends due to invalid msg. The abort probability is $\Pr(E_{q_{\mathsf{usc}}}) = q_{\mathsf{usc}}/2^k$.
- In the challenge phase, upon adversary's query for $\mathcal{P}_{\mathsf{ID}_{\mathsf{r_i}}} \neq \mathcal{P}^*_{\mathsf{ID_i}}$, $\mathcal{C}$ ends the game. The abort probability is $\Pr(E_{q_{H_1}}) = (1 - 1/q_{H_1})$.

Moreover, the $\mathcal{C}$ fetches L_1 to retrieve $Q_{\mathcal{P}_{\mathsf{ID_i}}}$ and L_2 to retrieve $Y_{1_{\mathsf{i}}}$ and evaluates $\frac{\theta Y_{1_{\mathsf{i}}} - \mathsf{d_i} r}{(\mathsf{d_s} - U)} = \theta \mu P$. The probability is $(1/q_{H_1} + 1/q_{H_2})$. Therefore, $\mathcal{C}'s$ advantage ϵ' of winning the game with probability is given as:

$$\epsilon' \geq \epsilon \left(\frac{1}{q_{H_1}} + \frac{1}{q_{H_2}}\right)\left(\frac{1}{q_{H_1}}\right)\left(1 - \frac{1}{q_{\mathsf{sv}}}\right)\left(1 - \frac{q_{\mathsf{usc}}}{2^k}\right) \tag{4}$$

Unforgeability

Theorem 3. *The AMCLHS technique proves euf-cma-I security under the hard assumptions of ECDL in ROM. If there is $\mathcal{A}_{\mathrm{I}}$ having a non-negligible advantage (ϵ) in winning the game, then there exists a $\mathcal{C}$ that can provide a solution to the ECDL with the non-negligible advantage (ϵ') [31].*

Proof. On input $P \in \mathbb{G}$ and a new generator $Q = \phi P$, the $\mathcal{C}$ must determine ϕ as defined in Definition 2 after interaction with $\mathcal{A}_{\mathrm{I}}$ as shown below:

1. ***Phase-1***: $\mathcal{A}_{\mathrm{II}}$ submits a number of queries q To keep a track of q_{H_l} responses, the $\mathcal{C}$ maintains a list L_l.
   - **Setup:** The $\mathcal{C}$ derives $\mathsf{PP} = \{\mathbb{G}, E, P, q, H_0, H_1, H_2, H_3\}$, defines $\mathsf{mpk} = \theta P$ and forwards $(\mathsf{PP}, \mathsf{mpk})$ to $\mathcal{A}_{\mathrm{I}}$. Then, $\mathcal{A}_{\mathrm{I}}$ chooses targeted sending vehicle identity denoted by $\mathcal{P}^*_{\mathsf{ID_s}}$.
2. ***Phase-2***: In this phase, public and private key queries are submitted by $\mathcal{A}_{\mathrm{I}}$ including q_{pk}, q_{ppk}, q_{pr}, q_{sv}, and q_{usc}. The $\mathcal{C}$ keeps a list L_{pk} to record the responses. The responses to these queries are according to *Phase-2* of Theorem 1 except q_{ppk}, which is shown below:
   - q_{ppk}: Here, $\mathcal{C}$ will abort if $\mathcal{P}_{\mathsf{ID}} = \mathcal{P}^*_{\mathsf{ID_s}}$. Else, it checks L_{pk} for $\mathsf{d_i}$. If it is present, returns it to $\mathcal{A}_{\mathrm{I}}$; if not, selects $\phi \in \mathbb{Z}^*_q$ at random and calculates $\mathsf{d_i} = \phi Q_{\mathcal{P}_{\mathsf{ID_i}}}$ and responds to $\mathcal{A}_{\mathrm{I}}$ with $\mathsf{d_i} = \phi Q_{\mathcal{P}_{\mathsf{ID_i}}}$. Next, it upgrades $(\mathcal{P}_{\mathsf{ID_i}}, \mathsf{d_i}, \mathsf{pk_i}, x_{\mathsf{i}})$ in L_{pk}.
3. ***Forgery***: $\mathcal{A}_{\mathrm{I}}$ takes $(\mathcal{P}^*_{\mathsf{ID_s}}, \mathcal{P}_{\mathsf{ID_{r_i}}})$, and generates a forged $\mathcal{CT}^*_{\mathsf{i}} = (\mathsf{C}^*_{1_{\mathsf{i}}}, \mathsf{C}^*_{2_{\mathsf{i}}})$ on msg^* where $\mathsf{C}^*_{1_{\mathsf{i}}} = (\rho^*_{\mathsf{i}}, \sigma^*_{\mathsf{i}})$ and $\mathsf{C}^*_{2_{\mathsf{i}}} = \mathsf{Enc}_{\mathcal{K}^*_{\mathsf{sy}}}(\mathsf{msg})$. In this case, $\mathcal{CT}^*_{\mathsf{i}}$ is legitimate and not generated by q_{sc} oracle.
   - Case-1 ($\mathcal{P}_{\mathsf{ID}} \neq \mathcal{P}^*_{\mathsf{ID_s}}$): In this case, The $\mathcal{C}$ responds with $\perp$.
   - Case-2 ($\mathcal{P}_{\mathsf{ID}} = \mathcal{P}^*_{\mathsf{ID_s}}$): Here, $\mathcal{C}$ fetches L_{pk} to retrieve $(\mathcal{P}^*_{\mathsf{ID_i}}, \mathsf{d}^*_{\mathsf{i}}, \mathsf{pk}^*_{\mathsf{i}}, x^*_{\mathsf{i}})$ and L_3 to retrieve $(\mathsf{msg}^*, \delta^*_{\mathsf{i}}, \rho^*_{\mathsf{i}})$.

   Following the Forking Lemma, $\mathcal{C}$ reruns $\mathcal{A}_{\mathrm{I}}$ using the same random input tape but provides different parameters from H_1 and H_3. This demonstrates that, $h^*_1 = H_1(\mathsf{mpk}, \mathcal{P}^*_{\mathsf{ID_i}})$ and $h'^*_1 = H_1(\mathsf{mpk}, \mathcal{P}^*_{\mathsf{ID_i}})$, where $h^*_1 \neq h'^*_1$ i.e. $Q^*_{\mathcal{P}_{\mathsf{ID_s}}} \neq Q'^*_{\mathcal{P}_{\mathsf{ID_s}}}$. In the same way, $h^*_3 = H_3(\mathsf{msg}^*, \delta^*_{\mathsf{i}}, \mathsf{pk}^*_{\mathsf{s}}, \mathsf{pk}^*_{\mathsf{r_i}})$, $h'^*_3 = H_3(\mathsf{msg}^*, \delta^*_{\mathsf{i}}, \mathsf{pk}^*_{\mathsf{s}}, \mathsf{pk}^*_{\mathsf{r_i}})$ where $h^*_3 \neq h'^*_3$ i.e. $\rho^*_{\mathsf{i}} \neq \rho'^*_{\mathsf{i}}$. Ultimately, the $\mathcal{A}_{\mathrm{I}}$ generates second forged ciphertext, $\mathcal{CT}'^*_{\mathsf{i}} = (\mathsf{C}^*_{1_{\mathsf{i}}}, \mathsf{C}^*_{2_{\mathsf{i}}})$ for the same msg^*, where $\mathsf{C}^*_{1_{\mathsf{i}}} = (\rho'^*_{\mathsf{i}}, \sigma'^*_{\mathsf{i}})$ and $\mathsf{C}^*_{2_{\mathsf{i}}} = \mathsf{Enc}_{\mathcal{K}^*_{\mathsf{sy}}}(\mathsf{msg})$. Hence, two valid σ^*_{i} are obtained by $\mathcal{C}$:

$$\sigma^*_{\mathsf{i}} = r^{*-1}(\rho^*_{\mathsf{i}} \| w \mathsf{d}^*_{\mathsf{s}} x_{\mathsf{s}}) \tag{5}$$

$$\sigma'^*_{\mathsf{i}} = r'^{*-1}(\rho'^*_{\mathsf{i}} \| w \mathsf{d}'^*_{\mathsf{s}} x_{\mathsf{s}}) \tag{6}$$

In this case, $r^* = r'^*$ and $\mathsf{d}^*_{\mathsf{s}} = \mathsf{d}'^*_{\mathsf{s}}$. Based on the Eqs. 8 and 9, $\mathcal{C}$ is able to retrieve ϕ in the following manner:

$$\phi = r^{*-1}(\rho'^*_{\mathsf{i}} - \rho^*_{\mathsf{i}}) + (\sigma^*_{\mathsf{i}} - \sigma'^*_{\mathsf{i}})(r^{*-1}(w x_{\mathsf{s}}(Q^*_{\mathcal{P}_{\mathsf{ID_s}}} - Q'^*_{\mathcal{P}_{\mathsf{ID_s}}}))^{-1}$$

Therefore, the $\mathcal{C}$ successfully provides a solution to ECDL hard assumption $Q = \phi P$ with an advantage ϵ':

$$\epsilon' \geq \epsilon \left(\frac{1}{q_{H_1}} + \frac{1}{q_{H_2}}\right)\left(\frac{1}{q_{H_1}}\right)\left(1 - \frac{1}{q_{\mathsf{ppk}}}\right)\left(1 - \frac{q_{\mathsf{usc}}}{2^k}\right) \tag{7}$$

Theorem 4. *The AMCLHS technique proves euf-cma-II security under the hard assumptions of ECDL in ROM. If there is $\mathcal{A}_{\mathrm{II}}$ having a non-negligible advantage (ϵ) in winning the game, then there exists a $\mathcal{C}$ that can provide a solution to the ECDL with the non-negligible advantage (ϵ') [31].*

Proof. On input $P \in \mathbb{G}$ and a new generator $Q = \pi P$, the $\mathcal{C}$ must determine π as defined in Definition 2 after interaction with $\mathcal{A}_{\mathrm{I}}$ as shown below:

1. ***Phase-1***: $\mathcal{A}_{\mathrm{II}}$ submits a number of queries q as defined in Theorem 2. To keep track of q_{H_l} responses, the $\mathcal{C}$ maintains a list L_l.
   - **Setup**: The $\mathcal{C}$ derives $\mathsf{PP} = \{\mathbb{G}, E, P, q, H_0, H_1, H_2, H_3\}$, defines $\mathsf{mpk} = \theta P$ and forwards $(\mathsf{PP}, \mathsf{mpk})$ to $\mathcal{A}_{\mathrm{II}}$. Then, $\mathcal{A}_{\mathrm{II}}$ chooses targeted sending vehicle identity denoted by $\mathcal{P}^*_{\mathsf{ID_s}}$.
2. ***Phase-2***: In this phase, public and private key queries are submitted by $\mathcal{A}_{\mathrm{II}}$ including q_{pk}, q_{ppk}, q_{pr}, q_{sv}, and q_{usc}. The $\mathcal{C}$ keeps a list L_{pk} to record the responses. The responses to these queries are according to *Phase-2* of Theorem 2 except q_{sv}, which is shown below:
   - q_{sv}: Here, $\mathcal{C}$ will abort if $\mathcal{P}_{\mathsf{ID}} = \mathcal{P}^*_{\mathsf{ID_s}}$. Else, it checks L_{pk} for x_{i}. If it is present, returns it to $\mathcal{A}_{\mathrm{II}}$; if not, selects $\pi \in \mathbb{Z}^*_q$ at random and calculates $\mathsf{pk_i} = \pi P$ and responds to $\mathcal{A}_{\mathrm{II}}$ with $\mathsf{pk_i} = \pi P$. Next, it upgrades $(\mathcal{P}_{\mathsf{ID_i}}, \mathsf{d_i}, \mathsf{pk_i}, x_{\mathsf{i}})$ in L_{pk}.
3. ***Forgery***: Using $(\mathcal{P}^*_{\mathsf{ID_s}}, \mathcal{P}_{\mathsf{ID_{r_i}}})$, $\mathcal{A}_{\mathrm{II}}$ creates a forged $\mathcal{CT}^*_{\mathsf{i}} = (\mathsf{C}^*_{1_{\mathsf{i}}}, \mathsf{C}^*_{2_{\mathsf{i}}})$ on msg^* where $\mathsf{C}^*_{1_{\mathsf{i}}} = (\rho^*_{\mathsf{i}}, \sigma^*_{\mathsf{i}})$ and $\mathsf{C}^*_{2_{\mathsf{i}}} = \mathsf{Enc}_{\mathcal{K}^*_{\mathsf{sy}}}(\mathsf{msg})$. In this case, $\mathcal{CT}^*_{\mathsf{i}}$ is legitimate and does not originate from q_{sc} oracle.
   - Case-1 ($\mathcal{P}_{\mathsf{ID}} \neq \mathcal{P}^*_{\mathsf{ID_s}}$): In this case, The $\mathcal{C}$ responds with $\perp$.
   - Case-2 ($\mathcal{P}_{\mathsf{ID}} = \mathcal{P}^*_{\mathsf{ID_s}}$): Here, $\mathcal{C}$ fetches L_{pk} to retrieve $(\mathcal{P}^*_{\mathsf{ID_i}}, \mathsf{d}^*_{\mathsf{i}}, \mathsf{pk}^*_{\mathsf{i}}, x^*_{\mathsf{i}})$ and L_3 to retrieve $(\mathsf{msg}^*, \delta^*_{\mathsf{i}}, \rho^*_{\mathsf{i}})$.

   Following the Forking Lemma, $\mathcal{C}$ reruns $\mathcal{A}_{\mathrm{I}}$ using the same random input tape but provides different parameters from H_1 and H_3. This demonstrates that, $h^*_1 = H_1(\mathsf{mpk}, \mathcal{P}^*_{\mathsf{ID_i}})$ and $h'^*_1 = H_1(\mathsf{mpk}, \mathcal{P}^*_{\mathsf{ID_i}})$, where $h^*_1 \neq h'^*_1$ i.e. $Q^*_{\mathcal{P}_{\mathsf{ID_s}}} \neq Q'^*_{\mathcal{P}_{\mathsf{ID_s}}}$. In the same way, $h^*_3 = H_3(\mathsf{msg}^*, \delta^*_{\mathsf{i}}, \mathsf{pk}^*_{\mathsf{s}}, \mathsf{pk}^*_{\mathsf{r_i}})$, $h'^*_3 = H_3(\mathsf{msg}^*, \delta^*_{\mathsf{i}}, \mathsf{pk}^*_{\mathsf{s}}, \mathsf{pk}^*_{\mathsf{r_i}})$ where $h^*_3 \neq h'^*_3$ i.e. $\rho^*_{\mathsf{i}} \neq \rho'^*_{\mathsf{i}}$. Ultimately, the $\mathcal{A}_{\mathrm{I}}$ generates second forged ciphertext, $\mathcal{CT}'^*_{\mathsf{i}} = (\mathsf{C}^*_{1_{\mathsf{i}}}, \mathsf{C}^*_{2_{\mathsf{i}}})$ for the same msg^*, where $\mathsf{C}^*_{1_{\mathsf{i}}} = (\rho'^*_{\mathsf{i}}, \sigma'^*_{\mathsf{i}})$ and $\mathsf{C}^*_{2_{\mathsf{i}}} = \mathsf{Enc}_{\mathcal{K}^*_{\mathsf{sy}}}(\mathsf{msg})$. Hence, two valid σ^*_{i} are obtained by $\mathcal{C}$:

$$\sigma^*_{\mathsf{i}} = r^{*-1}(\rho^*_{\mathsf{i}} \| w\mathsf{d}^*_{\mathsf{s}} x^*_{\mathsf{s}}) \tag{8}$$

$$\sigma'^*_{\mathsf{i}} = r'^{*-1}(\rho'^*_{\mathsf{i}} \| w\mathsf{d}^*_{\mathsf{s}} x'^*_{\mathsf{s}}) \tag{9}$$

In this case, $r^* = r'^*$ and $x^*_{\mathsf{s}} = x'^*_{\mathsf{s}}$. Based on the Eqs. 8 and 9, $\mathcal{C}$ is able to retrieve π in the following manner:

$$\pi = r^{*-1}(\rho'^*_{\mathsf{i}} - \rho^*_{\mathsf{i}}) + (\sigma^*_{\mathsf{i}} - \sigma'^*_{\mathsf{i}})(r^{*-1}(w\mathsf{mpk}(Q^*_{\mathcal{P}_{\mathsf{ID_s}}} - Q^*_{\mathcal{P}'_{\mathsf{ID_s}}}))^{-1}$$

Therefore, the $\mathcal{C}$ successfully provides a solution to ECDL hard assumption $Q = \pi P$ with an advantage ϵ':

$$\epsilon' \geq \epsilon \left(\frac{1}{q_{H_1}} + \frac{1}{q_{H_2}}\right)\left(\frac{1}{q_{H_1}}\right)\left(1 - \frac{1}{q_{\mathsf{sv}}}\right)\left(1 - \frac{q_{\mathsf{usc}}}{2^k}\right) \tag{10}$$

Anonymity: $\mathcal{P}_{\mathsf{ID}}$s are issued to individual vehicle users under our AMCLHS technique. Each vehicle uses $\mathcal{P}_{\mathsf{ID}}$ for communicating with other vehicles instead of the original identity $\mathcal{R}_{\mathsf{ID}}$. Thus, the sending vehicle broadcasts the same msg to receiving vehicles,

and at the same time, the $\mathcal{R}_{\mathsf{ID}}$ of the receiving vehicles remains private. Furthermore, each vehicle's $\mathcal{R}_{\mathsf{ID}}$ is verified by RA using its private key $\mathsf{sk}_{\mathsf{RA}} = v$ as $\mathcal{R}_{\mathsf{ID}} = \mathcal{P}_{\mathsf{ID}_1} \oplus H_0(Rv) = \mathcal{R}_{\mathsf{ID}} \oplus H_0(\alpha \mathsf{pk}_{\mathsf{RA}}) \oplus H_0(Rv) = \mathcal{R}_{\mathsf{ID}} \oplus H_0(Rv) \oplus H_0(Rv) = \mathcal{R}_{\mathsf{ID}}$. If it is valid, only then RA assigns the $\mathcal{P}_{\mathsf{ID}}$ to each vehicle. Similarly, the associated $\mathcal{P}_{\mathsf{ID}}$ will be rejected in the event that $\mathcal{R}_{\mathsf{ID}}$ is invalid. Moreover, no other entity can fraudulently authenticate $\mathcal{R}_{\mathsf{ID}}$ because only RA is aware of its $\mathsf{sk}_{\mathsf{RA}}$. Additionally, it can reveal $\mathcal{R}_{\mathsf{ID}}$ in the event of a malicious activity.

Non-repudiation: Primarily, the DS is utilised to ensure non-repudiation; the sending vehicle signs a msg using its sk_{s} and the receiving vehicle verifies by taking pk_{s}. By signing msg with their sk_{s}, the sender provides proof of having sent the msg and cannot subsequently retract it because the sending vehicle alone has knowledge of its sk_{s}. In the same way, the sending vehicle signs the msg in our proposed model with its sk_{s} as $\sigma_{\mathsf{i}} = r^{-1}(\rho_{\mathsf{i}} \| w \mathsf{d}_{\mathsf{s}} x_{\mathsf{s}})$. The receiving vehicle user verifies msg as $R_{\mathsf{i}} = \sigma_{\mathsf{i}}^{-1}(\rho_{\mathsf{i}} P \| w \mathsf{pk}_{\mathsf{s}} Y_{1_{\mathsf{i}}} Q_{\mathcal{P}_{\mathsf{ID}_{\mathsf{r}_{\mathsf{i}}}}}^{-1})$ by utilizing pk_{s}. In case of an emergency, for instance, if an adversary, such as a malicious vehicle user, sends a bogus message, it cannot be denied later since the sending vehicle signs msg with its sk_{s} and only the sending vehicle has the complete knowledge of its private key.

Forward Security: A key agreement framework that produces a fresh key for every session is implemented to guarantee forward secrecy. $\mathcal{A}$ cannot take advantage of the previous session even in the event that the key for past sessions is exposed. In this technique, a random secret value $x \in \mathbb{Z}_q^*$ and a ppk are used to derive the symmetric session key $\mathcal{K}_{\mathsf{sy}}$ with $(\mathsf{sk}, \mathsf{pk})$. The value of x is, therefore, randomly generated and regularly changed for every session, so regardless of whether sk is exposed, $\mathcal{A}$ will not be able to retrieve the prior sessions.

8 Performance Analysis

Here, the designed AMCLHS techniques' computational costs, communication costs, and security requirements are compared with those of other multi-receiver signcryption techniques.

8.1 Computational Cost

According to Table 2, for the computation cost, τ_{bp} represents the processing time of the BP operation, τ_m indicates the execution duration of the point multiplication operation, τ_e denotes time to execute the operation of exponentiation in $\mathbb{Z}_q^*$, and n shows the count of participating users in other technique. Multi-receiver scheme computational costs are put into comparison with [12,21,23,24]. The signcryption overhead is computed for multiple recipients as defined in our scheme, whereas the unsigncryption overhead is calculated for each recipient individually. Of all the schemes mentioned above, the highest computational overhead is provided by Niu et al. [21], computing total operations $(2n+4)\tau_{bp} + 1\tau_m + (2n+2)\tau_e$, with $(2n+4)\tau_{bp}$ bilinear pairings, being the most constly. Hongzhen et al. [12] employ total $(8n)\tau_m$ operations. Peng et al. [24] utilize total $(2n+5)\tau_m$ operations for both signcryption and unsigncryption phases. Niu et al. [23]

requires $(4n + 6)\tau_m$ total operations. In contrast to the mentioned signcryption techniques, our presented AMCLHS technique offers the highest efficiency with a total of $(n+5)\tau_m$ operations. It provides a unique pairing of a constant unsigncryption cost per receiver and a linear signcryption cost, irrespective of size. This ideal combination creates a dependable and expandable system that sets a new standard for performance. Due to its reliability and scalability, our scheme stands out as an appealing option for larger and more complicated VANET broadcast communication scenarios, offering a substantial improvement over existing schemes.

Table 2. Comparison of Computational Overhead with Existing Multi-receiver Schemes [31].

Schemes	Signcryption	Unsigncryption	Total
Hongzhen et al. [12]	$(4n)\tau_m$	$(4n)\tau_m$	$(8n)\tau_m$
Niu et al. [21]	$(2n)\tau_{bp} + 1\tau_m + (2n)\tau_e$	$4\tau_{bp} + 2\tau_e$	$(2n + 4)\tau_{bp} + 1\tau_m + (2n + 2)\tau_e$
Niu et al. [23]	$(2n + 4)\tau_m$	$(2n + 2)\tau_m$	$(4n + 6)\tau_m$
Peng et al. [24]	$(2n + 1)\tau_m$	$4\tau_m$	$(2n + 5)\tau_m$
Our scheme	$(n + 2)\tau_m$	$3\tau_m$	$(n + 5)\tau_m$

We implemented our scheme and evaluated the performance of our technique with respect to computational and communication costs. We conducted runtime experiments for the signcryption and unsigncryption employing the Python Cryptography Library and the NIST P256 (SECP256R1) elliptic curve with $q = 160$. This approach allowed us to precisely measure the efficiency of the AMCLHS scheme, providing more insights into its real-world application. The experiments were executed on a 64-bit Windows operating system with 16.0 GB of RAM and a microprocessor frequency of 2.80 GHz. The experiment was conducted for 1000 iterations, and the execution time for the multiplication operation on the elliptic curve during the signcryption phase was approximately 0.00316 milliseconds (ms). Figure 2 illustrates the computational cost of the signcryption phase. We do not compare the signcryption cost with that of Niu et al. [21] as our scheme excludes BP and E operations. Given that Niu et al.'s scheme involves a greater number of operations, it can be inferred that their approach is less efficient. It demonstrates how the number of recipients has a linear effect on the cost of signcryption (i.e., 10, 20, 30, 40, 50) and exhibits faster performance and maximum efficiency compared to others. We additionally assess the computational cost of unsigncryption in comparison to existing schemes. The average time taken for the multiplication operation during the unsigncryption phase was 0.00415 milliseconds. Figure 3 illustrates that while the unsigncryption cost for [24] is constant, it increases linearly with the number of receivers for [12] and [23]. In contrast to each of these schemes, the proposed AMCLHS scheme has the lowest and constant unsigncryption cost for each receiver, achieving higher efficiency.

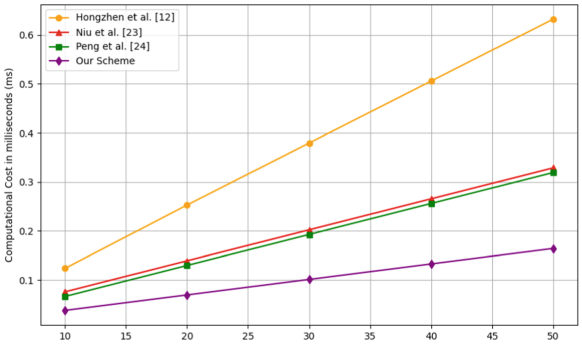


Fig. 2. Comparison of Computation Cost (Signcryption).

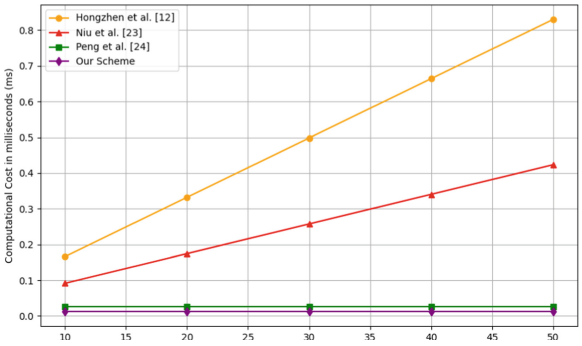


Fig. 3. Comparison of Computation Cost (Unsigncryption).

8.2 Security Requirements

We also demonstrate the comparison of the required security fulfilled by our scheme and existing multi-receiver hybrid signcryption schemes [12,21,23,24].

Table 3 highlights this comparison, covering key security requirements such as confidentiality, unforgeability, anonymity, non-repudiation, and forward security. This table provides a clear overview of how the AMCLHS scheme performs relative to others in meeting security requirements. As depicted in Table 3, [12,23] fulfills each security requirement but entails greater computation and communication overhead. In contrast to other schemes in the comparison, our proposed technique effectively satisfies all security requirements and offers improved efficiency with lower computing costs.

Table 3. Comparison based on Security Requirements with Existing Multi-receiver Schemes [31].

Schemes	Confidentiality	Unforgeability	Anonymity	Non-repudiation	Forward Security
Hongzhen et al. [12]	✓	✓	✓	✓	✓
Niu et al. [21]	✓	✓	✓	✓	✓
Niu et al. [23]	✓	✓	✓	×	×
Peng et al. [24]	✓	✓	✓	×	×
Our scheme	✓	✓	✓	✓	✓

8.3 Communication Cost

The cost of communication is generally the measure of the signcryption ciphertext length. Table 4 presents the cost of communication in terms of the ciphertext size generated by each scheme [12,21,23,24] in signcryption as well as unsigncryption phases. Out of the four schemes, our AMCLHS technique has the minimal cost of communication since it only utilises $n|\mathsf{msg}| + |\mathbb{Z}_q^*| + |\mathbb{G}| + |\mathcal{K}_{\mathsf{sy}}|$ bits to signcrypt a msg. We compare the communication cost with existing schemes for $n = 10$, 20, 30, 40, and 50. For better comparison, each parameter is given the following values measured in bits: $|\mathsf{msg}| = 160$; $|\mathbb{Z}_q^*| = 160$; $|\mathbb{G}| = 256$; and $|\mathcal{K}_{\mathsf{sy}}| = 64$, while, for convenience, the final ciphertext length is computed in bytes.

Table 4. Comparison of Communication Cost with Existing Multi-receiver Schemes.

Schemes	Ciphertext Length	Ciphertext Length in bytes				
		$n = 10$	$n = 20$	$n = 30$	$n = 40$	$n = 50$
Hongzhen et al. [12]	$n\|\mathsf{msg}\| + n\|\mathbb{G}\| + n\|\mathbb{Z}_q^*\|$	645	1440	2160	2880	3600
Niu et al. [21]	$n\|\mathsf{msg}\| + \|\mathbb{G}\| + 2n\|\mathbb{G}\|$	872	1712	2606	3387	4238
Niu et al. [23]	$n\|(\mathsf{msg} + 2)\| + 2\|\mathbb{G}\| + 2\|\mathbb{Z}_q^*\|$	306	499	712	914	1117
Peng et al. [24]	$n\|\mathsf{msg}\| + (n + 2)\|\mathbb{Z}_q^*\|$	440	840	1240	1640	2040
Our scheme	$n\|\mathsf{msg}\| + \|\mathbb{Z}_q^*\| + \|\mathbb{G}\| + \|\mathcal{K}_{\mathsf{sy}}\|$	260	460	672	860	1060

Legend: n is the number of users, $|\mathsf{msg}|$ is the plaintext Length, $|\mathbb{Z}_q^*|$ is the length of an element in finite field $\mathbb{Z}_q^*$, $|\mathbb{G}|$ is the length of an element in $\mathbb{G}$, and $|\mathcal{K}_{\mathsf{sy}}|$ is the size of the symmetric key

Based on the statistical results depicted, in Table 4 Fig. 4 compares our scheme's communication costs with those of the current schemes. When considering $n = 10$, 20, 30, 40, and 50, our scheme demonstrates the lowest ciphertext length, requiring only 260, 460, 672, 860, and 1060 bytes, respectively, which proves to be the minimum among the other schemes. In conclusion, Table 5 delineates the communication complexity and demonstrates that, in signcryption, our method exhibits a linear communication cost, whereas the unsigncryption cost stays constant.

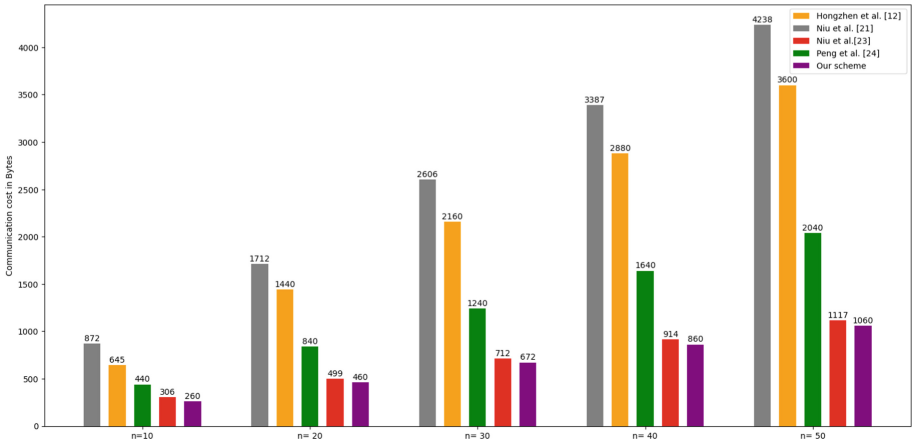


Fig. 4. Comparison of Communication Cost.

Table 5. Comparison of Complexity of Communication [31].

Schemes	Signcryption	Unsigncryption
Hongzhen et al. [12]	$\mathcal{O}(n)$	$\mathcal{O}(n)$
Niu et al. [21]	$\mathcal{O}(n^2)$	$\mathcal{O}(n)$
Niu et al. [23]	$\mathcal{O}(n)$	$\mathcal{O}(n)$
Peng et al. [24]	$\mathcal{O}(n^2)$	$\mathcal{O}(n)$
Our scheme	$\mathcal{O}(n)$	$\mathcal{O}(1)$

9 Conclusion and Future Scope

In this work, an unique AMCLHS scheme based on mKEM-DEM for broadcast VANET communication. The scheme utilizes the $(\mathsf{pk}, \mathsf{sk})$ pair to compute a symmetric session key for vehicle users. Each user then signcrypts the msg using the previously generated symmetric session key and their private key. Employing EC-CDH and ECDL assumptions, we illustrate security in terms of *ind-cca2* and *euf-cma* against $\mathcal{A}_{\mathrm{I}}$ and $\mathcal{A}_{\mathrm{II}}$. Furthermore, to provide anonymity, each participant is issued a $\mathcal{P}_{\mathsf{ID}}$, which is associated with their $\mathcal{R}_{\mathsf{ID}}$. Finally, we implement our technique and assess the outcomes against current single- and multi-receiver certificateless hybrid signcryption techniques. The comparative analysis demonstrates the lower communication costs and computational efficiency of our scheme, which simultaneously achieves confidentiality, unforgeability, anonymity, non-repudiation, and forward security. The signcryption cost is linear with the number of designated recipients, while the unsigncryption cost stays constant.

In the future, we aim to design and implement a post-quantum hybrid signcryption scheme for broadcast VANET environments. For enhanced security, including transparency and authorization, we plan to incorporate zero-knowledge and homomorphic encryption techniques.

Acknowledgments. Alia Umrani and Apurva K Vangujar are supported by PhD scholarships funded by the Science Foundation Ireland ADVANCE Centre for Research Training, under Grant number 18/CRT/6222. This research has also been supported by Science Foundation Ireland under Grant number 13/RC/2077_P2 (CONNECT).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Al-Riyami, S.S., Paterson, K.G.: Certificateless public key cryptography. In: Laih, C.-S. (ed.) ASIACRYPT 2003. LNCS, vol. 2894, pp. 452–473. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-40061-5_29
2. Barbosa, M., Farshim, P.: Certificateless signcryption. In: Abe, M., Gligor, V.D. (eds.) Proceedings of the 2008 ACM Symposium on Information, Computer and Communications Security, ASIACCS 2008, Tokyo, Japan, 18–20 March 2008, pp. 369–372 (2008)
3. Chen, X., He, D., Khan, M.K., Luo, M., Peng, C.: A secure certificateless signcryption scheme without pairing for internet of medical things. IEEE Internet Things J. **10**(10), 9136–9147 (2023)
4. Cohen, H., et al. (eds.): Handbook of Elliptic and Hyperelliptic Curve Cryptography (2005). https://doi.org/10.1201/9781420034981
5. Cui, B., Lu, W., He, W.: A new certificateless signcryption scheme for securing internet of vehicles in the 5G era. Secur. Commun. Netw. **2022** (2022)
6. Dent, A.W.: Hybrid signcryption schemes with insider security. In: Boyd, C., Nieto, J.M.G. (eds.) Information Security and Privacy, 10th Australasian Conference, ACISP 2005, Brisbane, Australia, 4–6 July 2005, Proceedings. Lecture Notes in Computer Science, vol. 3574, pp. 253–266 (2005)
7. Dent, A.W.: Hybrid signcryption schemes with outsider security. In: Zhou, J., Lopez, J., Deng, R.H., Bao, F. (eds.) ISC 2005. LNCS, vol. 3650, pp. 203–217. Springer, Heidelberg (2005). https://doi.org/10.1007/11556992_15
8. Du, H., Wen, Q., Zhang, S., Gao, M.: An improved conditional privacy protection scheme based on ring signcryption for vanets. IEEE Internet Things J. (2023)
9. Elkhalil, A., Zhang, J.: Practical heterogeneous signcryption system for vehicular communication in vanets. Computing **105**(1), 89–113 (2023). https://doi.org/10.1007/S00607-022-01114-0
10. Gong, B., Wu, Y., Wang, Q., Ren, Y., Guo, C.: A secure and lightweight certificateless hybrid signcryption scheme for internet of things. Future Gener. Comput. Syst. **127**, 23–30 (2022)
11. Guo, R., Xu, L., Li, X., Zhang, Y., Li, X.: An efficient certificateless ring signcryption scheme with conditional privacy-preserving in vanets. J. Syst. Archit. **129**, 102633 (2022). https://doi.org/10.1016/J.SYSARC.2022.102633
12. Hongzhen, D., Qiaoyan, W., Shanshan, Z., Mingchu, G.: A pairing-free certificateless signcryption scheme for vehicular ad hoc networks. Chin. J. Electron. **30**(5), 947–955 (2021)
13. Kasyoka, P.N., Kimwele, M.W., Mbandu, A.S.: Efficient certificateless signcryption scheme for wireless sensor networks in ubiquitous healthcare systems. Wirel. Pers. Commun. **118**(4), 3349–3366 (2021)
14. Khan, M.A., et al.: An efficient and provably secure certificateless key-encapsulated signcryption scheme for flying ad-hoc network. IEEE Access **8**, 36807–36828 (2020). https://doi.org/10.1109/ACCESS.2020.2974381

15. Li, F., Shirase, M., Takagi, T.: Certificateless hybrid signcryption. In: Bao, F., Li, H., Wang, G. (eds.) ISPEC 2009. LNCS, vol. 5451, pp. 112–123. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-00843-6_11
16. Li, K., Yang, Y., Wang, S.: Privacy-preserving aggregate signcryption scheme with allowing dynamic updating of pseudonyms for fog-based smart grids. Peer-to-Peer Netw. Appl. **15**(4), 2101–2115 (2022). https://doi.org/10.1007/S12083-022-01343-2
17. Li, X., Jiang, C., Du, D., Wang, S., Fei, M., Wu, L.: A novel efficient signcryption scheme for resource-constrained smart terminals in cyber-physical power systems. CoRR abs/2212.04198 (2022)
18. Liang, Y., Yan, H., Liu, Y.: Unlinkable signcryption scheme for multi-receiver in vanets. IEEE Trans. Intell. Transp. Syst. (2023)
19. Luo, M., Zhou, X., Qiu, M.: A revocable anonymous cross-domain communication scheme for smart grid based on ring signcryption. Peer-to-Peer Netw. Appl. 1–14 (2023)
20. Malone-Lee, J.: Identity-based signcryption. IACR Cryptol. ePrint Arch. 98 (2002)
21. Niu, S., Niu, L., Yang, X., Wang, C., Jia, X.: Heterogeneous hybrid signcryption for multi-message and multi-receiver. PLoS ONE **12**(9), e0184407 (2017)
22. Niu, S., Shao, H., Hu, Y., Zhou, S., Wang, C.: Privacy-preserving mutual heterogeneous signcryption schemes based on 5G network slicing. IEEE Internet Things J. **9**(19), 19086–19100 (2022)
23. Niu, S., Zhou, S., Fang, L., Hu, Y., Wang, C.: Broadcast signcryption scheme based on certificateless in wireless sensor network. Comput. Netw. **211**, 108995 (2022)
24. Peng, C., Chen, J., Obaidat, M.S., Vijayakumar, P., He, D.: Efficient and provably secure multireceiver signcryption scheme for multicast communication in edge computing. IEEE Internet Things J. **7**(7), 6056–6068 (2020)
25. Qiu, J., Fan, K., Zhang, K., Pan, Q., Li, H., Yang, Y.: An efficient multi-message and multi-receiver signcryption scheme for heterogeneous smart mobile IoT. IEEE Access **7**, 180205–180217 (2019)
26. Selvi, S.S.D., Vivek, S.S., Rangan, C.P.: Certificateless KEM and hybrid signcryption schemes revisited. IACR Cryptol. ePrint Arch. 462 (2009)
27. Selvi, S., Vivek, S.S., Shukla, D., Rangan Chandrasekaran, P.: Efficient and provably secure certificateless multi-receiver signcryption. In: Baek, J., Bao, F., Chen, K., Lai, X. (eds.) ProvSec 2008. LNCS, vol. 5324, pp. 52–67. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-88733-1_4
28. Singh, T., Kumar, M., Gelmecha, D.J., Thakur, G.K., Tyagi, V., Sori, W.J.: A lightweight identity based generalized signcryption scheme for secure communication in standard model. Int. J. Inf. Technol. **16**(2), 1221–1229 (2024)
29. Smart, N.P.: Efficient key encapsulation to multiple parties. In: Blundo, C., Cimato, S. (eds.) Security in Communication Networks, 4th International Conference, SCN 2004, Amalfi, Italy, 8–10 September 2004, Revised Selected Papers. Lecture Notes in Computer Science, vol. 3352, pp. 208–219 (2004)
30. Ullah, I., et al.: An efficient and secure multimessage and multireceiver signcryption scheme for edge-enabled internet of vehicles. IEEE Internet Things J. **9**(4), 2688–2697 (2022). https://doi.org/10.1109/JIOT.2021.3093068
31. Umrani, A., Vangujar, A.K., Palmieri, P.: Anonymous multi-receiver certificateless hybrid signcryption for broadcast communication. In: Lenzini, G., Mori, P., Furnell, S. (eds.) Proceedings of the 10th International Conference on Information Systems Security and Privacy, ICISSP 2024, Rome, Italy, 26–28 February 2024, pp. 733–744. SCITEPRESS (2024). https://doi.org/10.5220/0012353400003648
32. Umrani, A., Vangujar, A.K., Palmieri, P.: A multi-receiver certificateless signcryption (MCLS) scheme. In: 8th International Conference on Cryptography, Security and Privacy, CSP 2024, Osaka, Japan, 20–22 April 2024. IEEE (2024)

33. Vangujar, A.K., Umrani, A., Palmieri, P.: Identity-based cluster authentication and key exchange (ID-CAKE) message broadcasting and batch verification in VANETs. In: Andreoni, M. (ed.) Applied Cryptography and Network Security Workshops - ACNS 2024 Satellite Workshops, AIBlock, AIHWS, AIoTS, SCI, AAC, SiMLA, LLE, and CIMSS, Abu Dhabi, United Arab Emirates, 5–8 March 2024, Proceedings, Part II. Lecture Notes in Computer Science, vol. 14587, pp. 162–179. Springer (2024). https://doi.org/10.1007/978-3-031-61489-7_10
34. Wu, Y., Gong, B., Zhang, Y., et al.: An improved efficient certificateless hybrid signcryption scheme for internet of things. Wirel. Commun. Mob. Comput. **2022** (2022)
35. Yang, Y., He, D., Vijayakumar, P., Gupta, B.B., Xie, Q.: An efficient identity-based aggregate signcryption scheme with blockchain for IoT-enabled maritime transportation system. IEEE Trans. Green Commun. Netw. **6**(3), 1520–1531 (2022)
36. Yin, A., Liang, H.: Certificateless hybrid signcryption scheme for secure communication of wireless sensor networks. Wirel. Pers. Commun. **80**(3), 1049–1062 (2015)
37. Yu, X., Zhao, W., Tang, D.: Efficient and provably secure multi-receiver signcryption scheme using implicit certificate in edge computing. J. Syst. Archit. **126**, 102457 (2022)
38. Yu, Y., Yang, B., Huang, X., Zhang, M.: Efficient identity-based signcryption scheme for multiple receivers. In: Xiao, B., Yang, L.T., Ma, J., Muller-Schloer, C., Hua, Yu. (eds.) ATC 2007. LNCS, vol. 4610, pp. 13–21. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-73547-2_4
39. Zhang, W., et al.: Certificateless hybrid signcryption by a novel protocol applied to internet of things. Comput. Intell. Neurosci. **2022** (2022)
40. Zheng, Y.: Digital signcryption or how to achieve cost(signature & encryption) ¡¡ cost(signature) + cost(encryption). In: Kaliski, B.S. (ed.) Advances in Cryptology - CRYPTO 1997, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, 17–21 August 1997, Proceedings. Lecture Notes in Computer Science, vol. 1294, pp. 165–179 (1997)
41. Zhou, C.: Certificateless signcryption scheme without random oracles. Chin. J. Electron. **27**, 1002–1008 (2018)
42. Zhou, Y., Xu, R., Qiao, Z., Yang, B., Xia, Z., Zhang, M.: An anonymous and efficient multi-message and multi-receiver certificateless signcryption scheme for vanet. IEEE Internet Things J. (2023)
43. Zhu, H., Jin, C., Xu, Y., Chen, G., Chen, L.: Efficient and secure heterogeneous online/offline signcryption for wireless body area network. Pervasive Mob. Comput. 101893 (2024)

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (http://creativecommons.org/licenses/by/4.0/), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

