IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received January 4, 2022, accepted January 26, 2022, date of publication January 28, 2022, date of current version February 11, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3147595

A Conditional Privacy-Preserving Certificateless Aggregate Signature Scheme in the Standard Model for VANETs

HUIWEN WANG[1], LIANGLIANG WANG[1], KAI ZHANG[1], JINGUO LI[1], AND YIYUAN LUO[2,3]

[1]College of Computer Science and Technology, Shanghai University of Electric Power, Shanghai 201306, China
[2]School of Computer Science and Engineering, Huizhou University, Huizhou 516007, China
[3]Henan Key Laboratory of Network Cryptography Technology, Zhengzhou 450001, China

Corresponding author: Liangliang Wang (llwang@shiep.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61802249 and Grant 62072207, and in part by the Henan Key Laboratory of Network Cryptography Technology under Grant LNCT2020-A05.

ABSTRACT Vehicular ad hoc networks (VANETs) are the communication foundation for future intelligent transportation systems and guarantee safe driving of intelligent networked vehicles. Moreover, VANETs face a series of security challenges related to the protection of vehicle privacy, authenticity of transmitted information, and bandwidth limitations. To resolve these contradictions, many certificateless aggregate signature (CLAS) schemes have been proposed. However, the majority of them can neither resist malicious-but-passive key generation center attacks, replay attacks, and link attacks, nor track the actual identities of malicious vehicles. Meanwhile, the security of previous CLAS schemes in VANETs is only formally provided in the random oracle model (ROM), which might be insecure in actual implementation. In addition, most CLAS schemes still have problems of large verification delays and high communication overhead. To address the above-mentioned problems, a new conditional privacy-preserving CLAS scheme in VANETs is proposed, which adopts full aggregation technology to reduce computation and bandwidth resources. According to the formal security proofs under the computational Diffie-Hellman problem (CDHP) given in the standard model (SM), the security level of this scheme is higher than that of other CLAS schemes under ROM in practical applications. Additionally, the use of pseudonym mechanism realizes conditional privacy protection in VANETs. The performance analysis shows that this scheme has a higher efficiency in terms of computation and communication overhead compared with several previous CLAS schemes.

INDEX TERMS Vehicular ad hoc networks (VANETs), conditional privacy preserving, certificateless aggregate signature (CLAS), full aggregation, standard model (SM).

I. INTRODUCTION

As novel mobile wireless self-organizing networks applied on the road, VANETs rely on dedicated short range communication (DSRC) [1] to achieve vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications. The on board units (OBUs) installed on vehicles collect driving information to implement V2V and V2I communications through communication protocols such as 802.11p [2] and 3G/4G. OBUs and roadside units (RSUs) must share all collected vehicle information to improve travel efficiency and reduce traffic accidents.

The open wireless network of VANETs makes it easy for attackers to monitor, delete, and edit communication data [3]. For example, malicious vehicles in VANETs might spread fraud information to obtain benefits or broadcast incorrect information to mislead the decision of the traffic management center (TMC). As common technology to solve these problems, digital signatures can provide various security attributes, such as non-repudiation, authentication, and integrity. Simultaneously, the sensitive personal information of drivers, such as their true identities and driving routes, should be protected to prevent illegal infringement. This problem can be solved by using anonymity. Vehicles can use pseudo-identities when communicating with other vehicles or RSUs. However, if some vehicles maliciously disrupt traffic order, TMC should be capable of tracking their corresponding

The associate editor coordinating the review of this manuscript and approving it for publication was Chuan Heng Foh.

real identities. Therefore, the privacy protection in VANETs must be conditional. In addition to the bandwidth of VANETs, the storage capacity of RSUs and OBUs is limited. In 2003, Boneh *et al.* illustrated the cryptographic primitive of the aggregate signature [4]. After receiving n different signatures from users, the aggregator can aggregate them into one aggregated signature, so that the verifier can judge the validity of n signatures only once, thus greatly reducing the computational cost.

In 2003, the concept of certificateless public key cryptography (CL-PKC) is put forward by Al-Riyami and Paterson, which eliminates the key escrow problem as well as the certificate management problem [5]. In CL-PKC, the user's complete private key is generated by the collaboration between the user and the Key generation center (KGC). The KGC is responsible for generating partial private keys and sending them to the users. The user combines his own secret value to generate a complete user's private key. Subsequently, several certificateless aggregate signature schemes (CLAS) for VANETs [6]–[15] have been proposed. In 1993, Bellare and Rogaway [16] formally put forward the random oracle model (ROM) methodology, which significantly promoted the development of the provable security of public key cryptography. In more detail, random oracle is a strong assumption that is usually an idealized replacement for cryptographic hash functions. However, in reality, the hash function is not completely random or collision-resistant. Cramer and Shoup [17] put forward the first public key encryption subject that proved secure and effective in the standard model (SM) in 1998. In the SM, adversaries can directly compute the value of the hash function rather than query challengers, which is constrained only by time and computing power. Moreover, the security proof in SM is usually harder than that in ROM.

To enhance privacy protection and communication security in VANETs, a lightweight and conditional privacy-protecting CLAS scheme with verifiable security in the SM is proposed for establishing V2I communication.

This article mainly contributes as below:

- An innovative and effective certificateless aggregate signature scheme with conditional privacy protection for VANETs is recommended with some security attributes, including integrity, non-repudiation, anonymity, unlinkability, traceability, and resistance to replay attacks.
- Assuming the computational Diffie-Hellman problem against type I and type II adversaries in the SM, this CLAS scheme presents practically unforgeability under adaptive chosen-message attacks (EUF-CMA). To the best of our knowledge, this is the first conditional privacy-preserving CLAS scheme with verifiable security in the SM for VANETs.
- Independent of map-to-point operations, the proposed CLAS scheme only requires two bilinear pair operations for aggregate verification, which has a lower computational overhead than previous related CLAS schemes [9], [15], [18]–[22]. Moreover, this scheme can achieve a rapid verification under an increase in the signatures, and can effectively reduce the communication overhead.

The context of this study is as follows. Section II describes related work. Our motivation is given in Section III. Section IV offers some preliminaries, involving bilinear pairing, CDHP, system model, scheme framework as well as security models. Section V proposes a new conditional privacy-preserving CLAS scheme. Section VI gives the security proofs and requirements analysis. Section VII conducts the performance analysis, including security attributes, computation overhead, communication overhead and practicability evaluation. At last, Section VIII draws conclusions for the whole paper.

II. RELATED WORK

To protect vehicle privacy and further develop a trusting relationship between vehicles and RSUs, diverse digital signature schemes have been proposed. The traditional public key infrastructure (PKI)-based public key cryptosystem [23] suffers from diverse certificate management problems such as certificate storage, allocation, and revocation. Although the ID-based public key cryptosystem (ID-PKC) [24] can eradicate the certificate management problem in PKI, it is stuck with another key escrow problem. In 2007, Castro *et al.* [25] proposed the first CLAS scheme to resolve the problems in the above-mentioned cryptosystems, but the number of required map-to-point operations increased linearly as the number of signers increased. As the security model in Section IV reveals, the security proofs of CL-PKC typically consider two types of adversaries. Zhang *et al.* [26] proposed a CLAS scheme in 2009, but was insecurity against type I and II adversaries [27]. Xiong *et al.* [28] presented a CLAS scheme with invariable pairing operations in 2013, but it proved to be insecure against some concrete attacks such as type II adversaries, collision-resistant attacks, and insider attacks [29].

Horng *et al.* [30] proposed a conditional privacy-protecting CLAS scheme for vehicle sensor networks in 2015; however, it cannot reduce the bandwidth efficiently without adopting full aggregation. In addition, Li *et al.* verified the insecurity of their scheme against type II adversaries [30]. Azees *et al.* [31] developed an anonymous authentication scheme based on certificates and bilinear pairs for VANETs in 2017; however, it faced certificate management problems. In 2018, Gayathri *et al.* [32] constructed a certificateless signature (CLS) scheme with batch verification for VANETs; however, it suffered a high verification delay. By adopting full aggregation to construct a CLAS authentication scheme in VANETs, Zhong *et al.* [33] considered it semantically secure in ROM in 2019. Unfortunately, it was verified to be insecure against type II adversaries by Kamil and Ogundoyin [20] who further improved a novel CLAS scheme on the basis of the hypothesis that elliptic curve discrete logarithm problem (ECDLP) is difficult. In 2020, Cui *et al.* [34] proposed a privacy-preserving cooperative downloading scheme based on edge computing in VANETs

whose security proof is given in the random oracle model. Zhao *et al.* [12] offered a novel CLAS scheme for the Internet of Vehicles (IoV) in 2020, but the construction of CLS is incorrect [13]. In 2021, Ye *et al.* [14] proposed a superior CLAS scheme with many security attributes for VANETs. In 2021, Zhang *et al.* [35] proposed a Chinese remainder theorem based conditional privacy-preserving authentication protocol to achieve secure V2V communications in VANETs. Full aggregation was employed in this scheme to reduce computation and bandwidth resources. In 2021, Altaf *et al.* [22] proposed a privacy-preserving localized hybrid authentication protocol for large-scale VANETs based on CL-PKC and PKI and showed that it is provably secure against type I and II adversaries in the SM. Ren *et al.* [36] proposed a certificateless batch verification signature scheme based on blockchain for privacy protection of VANETs in 2021. In 2021, Kamil *et al.* [37] developed a group key distribution process and put forward a certificateless authentication scheme with batch verification in long term evolution-vehicle system. The security proofs of all CLS schemes mentioned above were only formally provided in ROM.

III. MOTIVATION

The overall design rationale of this scheme was mainly inspired by Mei *et al.* [15] and Deng *et al.* [38]. In 2021, Mei *et al.* [15] presented a conditional privacy preservation CLAS scheme with multiple security requirements in IoV based on bilinear pairings. However, it requires four bilinear pair operations in the aggregation verification phase, resulting in a higher verification overhead. In addition, the security proof is provided in the random oracle model. In 2020, Deng *et al.* [38] put forward the first provable secure CLAS scheme in SM, but it cannot be applied to VANETs because of its inability to resist replay attacks and lack of anonymity. To combine the advantages and solve the problems of the schemes proposed by Mei *et al.* [15] and Deng *et al.* [38], we propose a certificateless aggregate signature scheme with low verification delay, multiple security attributes and conditional privacy protection for VANETs in the standard model.

IV. PRELIMINARIES

We list all the notations used in this article in Table 1. In addition, we present the assumptions, system model, scheme framework, and security models.

A. BILINEAR PAIRING AND CDHP

Suppose that G_1, G_2 denote an additive cyclic group and a multiplicative cyclic group with the same prime order q, respectively. Let P be a generator of G_1, and map $e : G_1 \times G_1 \rightarrow G_2$ be a bilinear pairing [39] with the following three characteristics:

1) Bilinearity: $\forall P_1, P_2 \in G_1$ and $\forall a, b \in Z_q^*$, we have $e(aP_1, bP_2) = e(P_1, P_2)^{ab}$.

2) Non-degeneracy: $e(P_1, P_2) \neq 1_{G_2}$.

3) Computability: $\forall P_1, P_2 \in G_1$, $e(P_1, P_2)$ can be calculated efficiently.

TABLE 1. List of notations.

Notation	Meaning
ν	Security parameter
q	A secure prime number $q > 2^{\nu}$
G_1	An additive cyclic group
G_2	A multiplicative cyclic group
e	A bilinear map
P, Q	Two generators of the group G_1
$params$	System public parameters
P_{pub}	The public key of system
s	The master secret key of system
H_1, H_2, H_3	One way hash functions
k	Identity tracking key
ID_i	The real identity of Vh_i
PID_i	A set of pseudonyms of Vh_i
$PID_{i,j}$	The j^{th} pseudonym of Vh_i
d_i	Partial private Key of Vh_i
x_i	Secret value of Vh_i
$PK_i = (X_i, R_i)$	Public Key of Vh_i
$SK_i = (d_i, x_i)$	Private Key of Vh_i
m_i	Traffic-related message
TS_i	Current timestamp chosen by Vh_i
$\sigma_i = (U_i, V_i, W_i)$	Signature on a message m_i
$\sigma = (U, V, W)$	An aggregate signature

Definition 1: The computational Diffie-Hellman problem (CDHP) [39]:

For two unknown random numbers $a, b \in Z_q^*$, given a tuple (P, aP, bP), the goal of CDHP is to calculate the value of abP where $P \in G_1$.

CDH assumption: No adversary can solve the CDHP in probabilistic polynomial time with a non-negligible probability.

B. SYSTEM MODEL

It can be seen from Fig. 1 that our system model involves the following five entities: 1) Key Generation Center (KGC); 2) Trace Authority (TRA); 3) On-board Units (OBUs); 4) Roadside Units (RSUs); 5) Traffic Management Center (TMC). The detailed functions of the five entities are as below.

1) Key Generation Center (KGC): KGC works with TRA to generate public parameters for VANETs, which is always trusted for strong security. And KGC can generate vehicles partial private keys.
2) Trace Authority (TRA): TRA is in charge of system initialization and vehicles registration. TRA assigns it a pseudo-identity first once a new vehicle joins the VANETs. Only TRA knows the true identities of vehicles. When a malicious traffic situation occurs, TRA reveals the true identities of malicious vehicles.
3) On board Units (OBUs): All vehicles on the road are equipped with OBUs. Vehicles can use OBUs to communicate with other vehicles and RSUs through V2V and V2I communications, respectively. Traffic-related messages and signatures can be transmitted from vehicles to the neighboring RSU using each pseudo identity once.

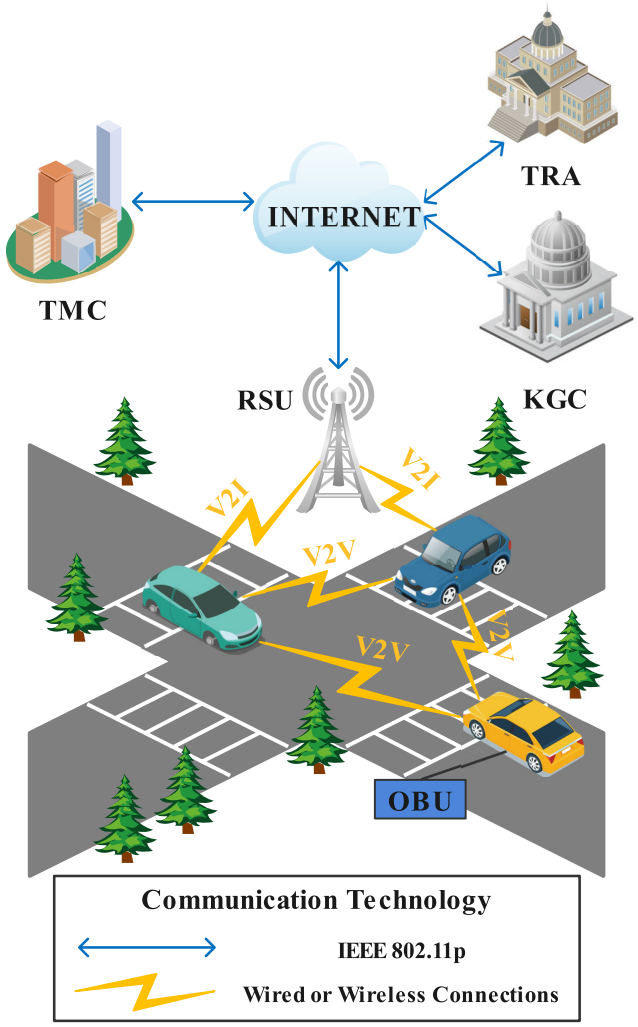

FIGURE 1. System model of VANETs.

4) Roadside Units (RSUs): RSUs are wireless communication devices along the road that use the DSRC protocol to realize V2I communication within its coverage. RSUs can verify the validity of the single traffic-related message from the OBUs. The RSUs then generate one aggregate signature and send it to the TMC.
5) Traffic Management Center (TMC): The TMC verifies the validity of the aggregate signature to define whether messages are accepted or not and analyzes messages to obtain information about traffic conditions. Hence, it can manage and regulate traffic flexibility.

C. SECURITY REQUIREMENTS

This CLAS scheme in V2I communication is ought to fulfill these six security requirements.

1) Authentication and integrity: As the receiver, the RSU must confirm the source of the message and ensure that the message has not been tampered with during transmission.

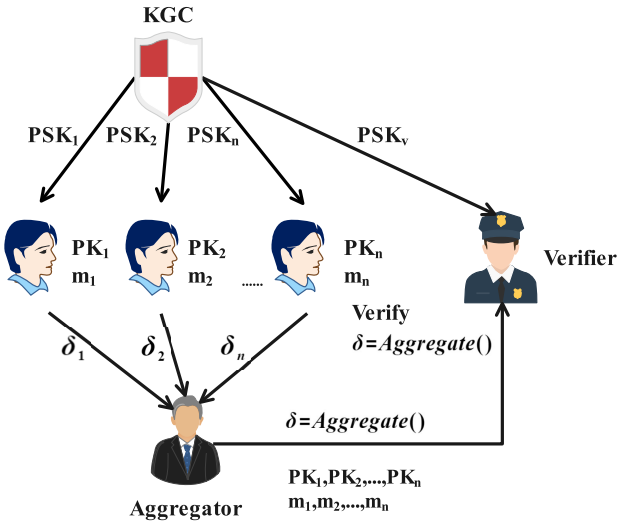

FIGURE 2. Architecture for CLAS scheme.

2) Nonrepudiation: The vehicle cannot refuse the message and corresponding signature generated by it.
3) Anonymity: Vehicles use pseudo-identities in V2I and V2V communications. In addition to the TRA, other vehicles and RSUs that interact with the vehicle do not know the true identity of the vehicle.
4) Unlinkability: A malicious attacker cannot use two or more signatures to determine whether the messages come from the same vehicle.
5) Traceability: When malicious messages appear, only TRA can track the true identity of the vehicle through its pseudo-identity.
6) Replay attack resistance: When a malicious attacker replays the legal signature sent before and sends it to the RSU, the RSU can identify and reject it.

D. SCHEME FRAMEWORK

The architecture of the CLAS scheme is illustrated in Fig. 2. The proposed CLAS scheme involves eight polynomial-time solvable algorithms as follows:

1) *Setup*: KGC and TRA perform this algorithm by taking the security parameter ν as input to output the system master secret key s, identity tracking key k, and system public parameters *params*. Finally, they publish the public parameters *params* while retaining the master secret key s and the identity tracking key k secret.
2) *Pseudonym Generation*: This polynomial time algorithm uses the true identity ID_i of vehicle Vh_i as the input. The TRA then intends to output the pseudonym $PID_{i,j}$. Subsequently, TRA sends the pseudonym $PID_{i,j}$ to vehicle Vh_i.
3) *Partial Private Key Generation*: KGC also performs this algorithm by taking system public parameters *params* and master secret key s to generate the vehicle partial private key d_i for the pseudo identity $PID_{i,j}$.

Subsequently, the partial private key d_i is transmitted from KGC to Vh_i via a secret channel.

4) *Public/Private Key Generation*: It is executed by vehicle Vh_i by taking the system public parameters *params* and the partial private key d_i as input. Vehicle Vh_i outputs vehicle public key PK_i and private key SK_i.
5) *Signature Generation*: Upon inputting the system public parameters *params*, traffic-related message m_i, pseudonym $PID_{i,j}$ and vehicle key pair (PK_i, SK_i), vehicle Vh_i outputs a single signature σ_i on the message m_i. Finally, it cancels $PID_{i,j}$ from the pseudonym set PID_i.
6) *Single Signature Verification*: The RSU performs this algorithm by using a single signature σ_i as input. First, the corresponding RSU determines the freshness of the timestamp TS_i. If timestamp TS_i is in the valid period, then the RSU verifies the single signature σ_i. If the single signature σ_i is valid, the RSU accepts it; otherwise, it is rejected.
7) *Aggregate*: The algorithm is also run by the RSU, which aggregates n single signatures σ_i of n traffic-related messages into one aggregate signature σ and outputs it.
8) *Aggregate Verification*: TMC runs this algorithm by taking the system public parameters *params*, vehicle pseudonyms $(PID_{1,j}, \ldots, PID_{n,j})$, timestamps $(TS_1, \ldots, TS_n)$, their public keys $(PK_1, \ldots, PK_n)$ and an aggregate signature σ as input. TMC accepts or rejects the aggregate signature σ depending on whether it is valid.

E. SECURITY MODELS

In a certificateless signature cryptosystem, two types of adversaries are always taken into consideration, type I adversary and type II adversary. Detailed introductions of these two adversaries are as follows.

- Type I adversary: This is usually called adversary A_I who has the ability to replace the public key PK_i of vehicle Vh_i, but cannot access the master secret key s. Additionally, A_I is regarded as an outside adversary.
- Type II adversary: This is usually called adversary A_{II}, which is known as a malicious-but-passive KGC. It is not allowed to replace the public key PK_i, but can obtain the master secret key s. Additionally, A_{II} is regarded as an inner adversary.

Definition 2: In the standard model, a CLAS scheme is existential unforgeability under adaptive chosen-message attacks (EUF-CMA) if there is no polynomial time adversary A_I succeeds in the Game I with a non-negligible probability.

Game I: This Game involves two entities: challenger C and adversary A_I.

Initialization. The *Setup* algorithm is executed by challenger C to generate system public parameters *params*. Then, the system public parameters *params* are sent to the A_I, and challenger C secretly retains the master secret key s.

Queries: A_I makes the following queries, which are answered by challenger C. A_I makes User public key queries for each $PID_{i,j}$ prior to other queries.

- User public key queries: When A_I submits a query on a pseudonym $PID_{i,j}$, the public key PK_i is sent from challenger C to it.
- User public key replacement queries: When A_I submits a query on a tuple of $(PID_{i,j}, PK_i')$, challenger C replaces $PK_i = PK_i'$.
- Partial private key extraction queries: When A_I requests a pseudonym $PID_{i,j}$, the corresponding partial private key d_i is sent from challenger C.
- Secret value queries: A_I submits a request on a pseudonym $PID_{i,j}$, and challenger C responds with secret value x_i. If challenger C receives User public key replacement queries on PK_i, A_I cannot query the secret value of the pseudonym $PID_{i,j}$.
- Signature queries: When A_I makes a query on a tuple of $(PID_{i,j}, m_i||TS_i)$, C returns the corresponding signature σ_i to A_I.

Forgery. A_I outputs a valid forged aggregate signature σ^*, which is composed of n signatures σ_i^* on messages $m_i^*(i = 1, .., n)$ from n vehicles, with their pseudonyms $(PID_{1,j}^*, \ldots, PID_{n,j}^*)$ and relevant public keys $(PK_1^*, \ldots, PK_n^*)$.

If the following conditions are fulfilled, A_I is said to succeed in Game I.

1) σ^* is a valid forged aggregate signature of messages $m_i^*(i = 1, .., n)$ with pseudonyms $(PID_{1,j}^*, \ldots, PID_{n,j}^*)$ and the relevant public keys.
2) There is at least a pseudonym $PID_{k,j}^*$, A_I did not make Partial private key extraction queries.
3) A_I did not make Signature queries for $(PID_{k,j}^*, m_k^*||TS_k^*)$.

Definition 3: In the standard model, a CLAS scheme is existential unforgeability under adaptive chosen-message attacks (EUF-CMA) if there is no polynomial time adversary A_{II} succeeds in the Game II with a non-negligible probability.

Game II: This Game involves two entities: challenger C and adversary A_{II}.

Initialization. The *Setup* algorithm is executed by challenger C to generate system public parameters *params*. Then, system public parameters *params* and master secret key s are sent to A_{II}.

Queries: A_{II} makes the following queries, which are answered by challenger C. A_{II} makes User public key queries for each $PID_{i,j}$ prior to other queries.

- User public key queries: When A_{II} submits a query on a pseudonym $PID_{i,j}$, challenger C returns the public key PK_i.
- Secret value queries: A_{II} submits a request on a pseudonym $PID_{i,j}$, and challenger C responds with the secret value x_i.
- Signature queries: When A_{II} makes a query on a tuple of $(PID_{i,j}, m_i||TS_i)$, C returns the corresponding signature σ_i to A_{II}.

Forgery. A_{II} outputs a valid forged aggregate signature σ^*, which is composed of n signatures σ_i^* on

messages $m_i^*(i = 1, .., n)$ from n vehicles, with their pseudonyms $(PID_{1,j}^*, \ldots, PID_{n,j}^*)$ and relevant public keys $(PK_1^*, \ldots, PK_n^*)$.

If the following conditions are fulfilled, A_{II} is said to succeed in Game II.

1) σ^* is a valid forged aggregate signature of messages $m_i^*(i = 1, .., n)$ with pseudonyms $(PID_{1,j}^*, \ldots, PID_{n,j}^*)$ and the relevant public keys.
2) There is at least a pseudonym $PID_{k,j}^*$, A_{II} did not make Secret value queries.
3) A_{II} did not make Signature queries for $(PID_{k,j}^*, m_k^* || TS_k^*)$.

V. THE PROPOSED CLAS SCHEME

We propose an efficient CLAS scheme and give the correctness analysis in this section, which mainly includes the following eight algorithms.

A. SETUP

TRA and KGC enter a security parameter ν into the parameter generator to generate a prime order $q > 2^{\nu}$. They generated two cyclic groups G_1, G_2 with prime order $q > 2^{\nu}$. Suppose e is a bilinear map $e : G_1 \times G_1 \rightarrow G_2$. KGC randomly selects $P, Q \in G_1, s \in Z_q^*$ and computes $P_{pub} = sP$. TRA randomly selects k and computes $K = kP$. Then, the TRA and KGC choose three hash functions $H_1 : G_1 \rightarrow Z_q^*, H_2 : \{0, 1\}^* \times G_1 \times G_1 \rightarrow Z_q^*, H_3 : \{0, 1\}^* \times \{0, 1\}^* \times G_1 \times G_1 \times G_1 \times G_1 \times G_1 \times G_1 \rightarrow Z_q^*$. Finally, they publish the system public parameters $params = \{G_1, G_2, q, e, P, Q, P_{pub}, H_1, H_2, H_3\}$, whereas contains the master secret key s and the identity tracking key k secret.

B. PSEUDONYM GENERATION

For the security of users, vehicles should first register with the TRA before they conduct information exchange in VANETs. In order to achieve anonymity, the true identity of vehicle ID_i cannot be used in the communication processes. TRA will assign it a pseudonym $PID_{i,j}$ once a new vehicle joins the VANETs, where $PID_{i,j}$ represents the j-th pseudonym of vehicle Vh_i. Vehicle Vh_i randomly selects $t_{i,j} \in Z_q^*$, and computes $T_{i,j} = t_{i,j}P$. Then, vehicle Vh_i secretly sends $(ID_i, T_{i,j})$ to the TRA. After verifying the validity of ID_i, TRA computes $PID_{i,1,j} = ID_i \oplus H_1(kP + T_{i,j})$ and $PID_{i,j} = \{PID_{i,1,j}, T_{i,j}\}$. Subsequently, TRA sends $PID_{i,j}$ to vehicle Vh_i. If vehicle Vh_i causes a malicious accident, TRA can track its true identity ID_i by tracking key k. Through the pseudonym $PID_{i,j} = \{PID_{i,1,j}, T_{i,j}\}$, the true identity of the vehicle can be tracked by TRA with calculating $ID_i = PID_{i,1,j} \oplus H_1(kP + T_{i,j})$.

C. PARTIAL PRIVATE KEY GENERATION

By taking the system public parameters $params$ and the master secret key s, KGC generates the vehicle Vh_i partial private key d_i as follows. The KGC randomly chooses $r_i \in Z_q^*$ and computes $R_i = r_iP$. Also compute $k_i = H_2(PID_{i,j}, R_i)$ and $d_i = r_i + k_i s \bmod q$. KGC sets d_i as the partial private key of vehicle Vh_i. After that, KGC sends the partial private key d_i to vehicle Vh_i via a secure socket layer (SSL) protocol.

D. PUBLIC/PRIVATE KEY GENERATION

After vehicle Vh_i sends a message, it re-selects a new secret value $x_i \in Z_q^*$ to update the public and private keys. Then it sets $X_i = x_iP$. Its public key is $PK_i = (X_i, R_i)$ and its private key is (d_i, x_i).

E. SIGNATURE GENERATION

The signature generation for one traffic-related message $m_i \in Z_q^*$ is explained as follows. First, the OBU chooses the current timestamp TS_i. Then, the OBU chooses random $u_i \in Z_q^*$ and computes $U_i = u_iP$ and $V_i = u_iQ$. Subsequently, vehicle Vh_i computes $h_i = H_3(m_i||TS_i, PID_{i,j}, U_i, V_i, W_i, PK_i)$, $W_i = (d_i + h_ix_i)Q + V_i$. Furthermore, it outputs signature $\sigma_i = (U_i, V_i, W_i)$ on $m_i||TS_i$, and sends $(m_i, TS_i, PK_i, PID_{i,j}, U_i, V_i, W_i)$ to the RSU. Each time vehicle Vh_i sends the signature, TRA regenerates a new pseudonym $PID_{i,j}$ and sends it to Vh_i, and vehicle Vh_i replaces the previous pseudonym with the new pseudonym. Thus, each pseudonym only can be used once.

F. SINGLE SIGNATURE VERIFICATION

After receiving signature σ_i on $m_i||TS_i$, the first step of the corresponding RSU is checking the freshness of timestamp TS_i. If TS_i is in the valid period, then the RSU verifies the validity of the signature, as shown below. The RSU calculates $k_i = H_2(PID_{i,j}, R_i)$ and $h_i = H_3(m_i||TS_i, PID_{i,j}, U_i, V_i, W_i, PK_i)$ and checks whether (1) holds.

$$e(W_i, P) = e(R_i + k_iP_{pub} + h_iX_i + U_i, Q) \tag{1}$$

If (1) is established, the RSU will accept single signature σ_i on $m_i||TS_i$; otherwise, it will reject it.

G. AGGREGATE

After receiving a series of n distinct signatures σ_i on different messages $m_i||TS_i$ from different vehicles Vh_i. The RSU computes $U = \sum_{i=1}^{n} U_i$, $V = \sum_{i=1}^{n} V_i$, and $W = \sum_{i=1}^{n} W_i$. Finally, it sends the aggregate signature $\sigma = (U, V, W)$ to TMC.

H. AGGREGATE VERIFICATION

Upon receiving the aggregate signature σ and tuples $(m_i, TS_i, PID_{i,j}, PK_i)$, the TMC checks the freshness of timestamp $TS_i(i = 1, 2, \ldots, n)$ first. If so, it calculates $k_i = H_2(PID_{i,j}, R_i)$ and $h_i = H_3(m_i||TS_i, PID_{i,j}, U_i, V_i, W_i, PK_i)$. Finally, it checks whether (2) holds.

$$e(W, P) = e(\sum_{i=1}^{n} R_i + \sum_{i=1}^{n} k_iP_{pub} + \sum_{i=1}^{n} h_iX_i + U, Q) \tag{2}$$

If (2) is established, the TMC accepts the aggregate signature σ on $m_i||TS_i(i = 1, 2, \ldots, n)$; otherwise, it will reject it.

I. CORRECTNESS

The following describes the correctness of the algorithm *single signature verification*.

$$\begin{aligned} e(W_i, P) &= e((d_i + h_i x_i)Q + V_i, P) \\ &= e((d_i + h_i x_i)Q + u_i Q, P) \\ &= e((d_i + h_i x_i)P + u_i P, Q) \\ &= e((r_i + k_i s + h_i x_i)P + U_i, Q) \\ &= e(R_i + k_i P_{pub} + h_i X_i + U_i, Q) \end{aligned}$$

The following describes the correctness of the algorithm *aggregate signature verification*.

$$\begin{aligned} e(W, P) &= e(\sum_{i=1}^{n} W_i, P) \\ &= e(\sum_{i=1}^{n}((d_i + h_i x_i)Q + V_i), P) \\ &= e(\sum_{i=1}^{n}(d_i + h_i x_i)Q + \sum_{i=1}^{n} u_i Q, P) \\ &= e(\sum_{i=1}^{n}(d_i + h_i x_i)P + \sum_{i=1}^{n} u_i P, Q) \\ &= e(\sum_{i=1}^{n}(r_i + k_i s + h_i x_i)P + \sum_{i=1}^{n} U_i, Q) \\ &= e(\sum_{i=1}^{n} R_i + \sum_{i=1}^{n} k_i P_{pub} + \sum_{i=1}^{n} h_i X_i + U, Q) \end{aligned}$$

VI. SECURITY PROOF AND ANALYSIS

The formal security proofs of our CLAS scheme in SM are presented in this section. In addition, we demonstrate that it meets many security demands for establishing V2I communication in VANETs. The ability of adversaries in the ROM is weakened, that is, only by querying challengers can adversaries obtain the hash function values. However, the adversaries in SM can directly calculate hash functions.

A. SECURITY PROOF

Theorem 1: If the CDHP assumption holds, then the proposed CLAS scheme is existentially unforgeable under adaptive chosen-message attacks (EUF-CMA) against adversary A_I in the standard model.

Lemma 1: In the standard model, if an adversary A_I is able to output a valid forged signature of a CLAS scheme by playing Game I, then the CDHP must be solved by a challenger C.

Proof: For a random tuple (P, aP, bP) of CDHP, challenger C is supposed to output the value of abP. A_I is a subroutine of challenger C in Game I. The security proof framework of Lemma 1 is shown in Fig. 3.

Initialization. Given a security parameter ν, challenger C sets $Q = bP$ and executes the algorithm *Setup* to generate the system public parameters $params = \{G_1, G_2, q, e, P, Q, P_{pub}, H_1, H_2, H_3\}$. Then, the system public parameters *params* are sent to A_I, and the master secret key s is kept confidential.

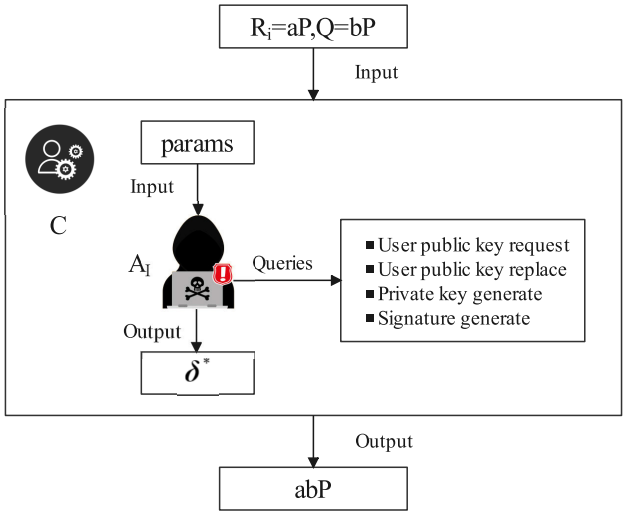

FIGURE 3. Framework for security proof of Lemma 1.

Queries: A_I makes the following queries, which are submitted to challenger C. Challenger C initially has empty lists L_U and L_R. A_I makes User public key queries for each $PID_{i,j}$ prior to other queries.

- User public key queries: Challenger C holds the list $L_U = (PID_{i,j}, r_i, x_i)$. A_I submits a request on a pseudonym $PID_{i,j}$, and challenger C will look for $(PID_{i,j}, r_i, x_i)$ in the L_U. If there exists such a tuple in L_U, then C outputs $(r_i P, x_i P)$. Otherwise, C dose the following.
  1) If $PID_{i,j} = PID^*$, C chooses x_i randomly and sets $R_i = aP$, then adds $(PID_{i,j}, *, x_i)$ to L_U, where $*$ denotes the null value. C sends $PK_i = (R_i, x_i P)$ to A_I.
  2) If $PID_{i,j} \neq PID^*$, C chooses r_i, x_i randomly and sets $R_i = r_i P, X_i = x_i P$, then adds $(PID_{i,j}, r_i, x_i)$ to L_U. C sends $PK_i = (R_i, X_i)$ to A_I.
- User public key replacement queries: A list $L_R = (PID_{i,j}, PK_i, PK_i')$ is maintained by challenger C. When A_I submits a query on a tuple of $(PID_{i,j}, PK_i')$, the challenger C replaces $PK_i = PK_i'$ and appends $(PID_{i,j}, PK_i, PK_i')$ to L_R.
- Partial private key extraction queries: When A_I submits a request on a pseudonym $PID_{i,j}$, the challenger C will look for $(PID_{i,j}, d_i)$ in the L_P. If it is found in L_P, then C outputs d_i. Otherwise, C executes the following.
  1) If $PID_{i,j} = PID^*$, C fails and stops.
  2) If $PID_{i,j} \neq PID^*$, C searches the list L_U to find r_i and computes $d_i = r_i + k_i smodq$. Then, C sends d_i to A_I.
- Secret value queries: When A_I makes a request for a pseudonym $PID_{i,j}$, challenger C looks for x_i in L_U and responds with x_i.
- Signature queries: Upon receiving a query on a tuple of $(PID_{i,j}, m_i||TS_i)$ from A_I, C first runs User public key queries, Partial private key extraction queries, and

Secret value queries to obtain the values of R_i, d_i and x_i. Subsequently, C calculates $k_i = H_2(PID_{i,j}, R_i)$ and $h_i = H_3(m_i||TS_i, PID_{i,j}, U_i, V_i, W_i, PK_i)$. C chooses u_i randomly, then computes $U_i = u_iP$, $V_i = u_iQ$, and $W_i = (d_i + h_ix_i)Q + V_i$. Finally, C returns $\sigma_i = (U_i, V_i, W_i)$ to A_I as the signature of $(PID_{i,j}, m_i||TS_i)$. The signature σ_i generated in this way is valid.

Forgery. A_I outputs a forged aggregate signature $\sigma_i^* = (U_i^*, V_i^*, W_i^*)$ on message $m_i^*||TS_i^*$, and fulfills the conditions as defined in Section IV.

Solve CDHP. If $PID_{i,j} \neq PID^*$, then the game is terminated. Otherwise, $PID_{i,j} = PID^*$, so $PK_i^* = (aP, x_i^*P)$ and $Q = bP$. C searches the list L_U to get x_i^*, and computes $k_i^* = H_2(PID^*, aP)$ and $h_i^* = H_3(m_i^*||TS_i^*, PID^*, U_i^*, V_i^*, W_i^*, PK_i^*)$. Since σ_i^* is a valid signature, $U_i^* = u_i^*P$, $V_i^* = u_i^*Q$, and $W_i^* = (d_i^* + h_i^*x_i^*)Q + V_i^* = (a + k_i^*s + h_i^*x_i^*)bP + V_i^*$. Thus, C can solve CDHP by calculating $abP = W_i^* - V_i^* - k_i^*sbP - h_i^*x_i^*bP$.

Similarly, A_I can return a forged aggregate signature $\sigma^* = (U^*, V^*, W^*)$ on message $m_i^*||TS_i^*$ for $i = 1, 2, .., n$, where $U^* = \sum_{i=1}^n U_i^*$, $V^* = \sum_{i=1}^n V_i^*$, $W^* = \sum_{i=1}^n W_i^*$. There exists at least one user $PID_{k,j}^*$, A_I did not perform Partial private key extraction queries, which represents that σ_k^* is a forged signature on $m_k^*||TS_k^*$. If $PID_{k,j}^* = PID^*$, $PK_k^* = (aP, x_k^*P)$, and $Q = bP$, C can solve CDHP by executing the following procedures.

- Calculate $h_i^* = H_3(m_i^*||TS_i^*, PID_{i,j}^*, U_i^*, V_i^*, W_i^*, PK_i^*)$ for $i = 1, 2, \ldots, n$.
- Search L_U to find r_i^*, and compute $k_i^* = H_2(PID_{i,j}^*, r_i^*P)$, $d_i^* = r_i^* + k_i^*s$ for $i \neq k$.
- Compute $W_i^* = (d_i^* + h_i^*x_i^*)Q + V_i^*$ for $i \neq k$.
- Compute $W_k^* = W^* - \sum_{i=1, i\neq k}^n W_i^*$, where $W_k^* = (a + k_k^*s + h_k^*x_k^*)bP + V_k^*$.
- Search L_U to find x_k^*, and compute $k_k^* = H_2(PID_{k,j}^*, aP)$.
- Thus, C can solve CDHP by calculating $abP = W_k^* - V_k^* - k_k^*sbP - h_k^*x_k^*bP$.

Theorem 2: If the CDHP assumption holds, the proposed CLAS scheme is existentially unforgeable under adaptive chosen-message attacks (EUF-CMA) against adversary A_{II} in the standard model.

Lemma 2: In the standard model, if an adversary A_{II} is able to output a valid forged signature of a CLAS scheme by playing Game II, then the CDHP must be solved by a challenger C.

Proof: For a random tuple (P, aP, bP) of CDHP, challenger C is supposed to output the value of abP. A_{II} is a subroutine of challenger C in Game II. The security proof framework of Lemma 2 is shown in Fig. 4.

Initialization. Given a security parameter ν, challenger C sets $Q = bP$ and executes the algorithm *Setup* to generate the system public parameters $params = \{G_1, G_2, q, e, P, Q, P_{pub}, H_1, H_2, H_3\}$. Then, the system public parameters *params* and the master secret key s are sent to A_{II}.

Queries: A_{II} makes the following queries, which are submitted to challenger C. Challenger C initially has empty list L_U. A_{II} makes User public key queries for each $PID_{i,j}$ prior to other queries.

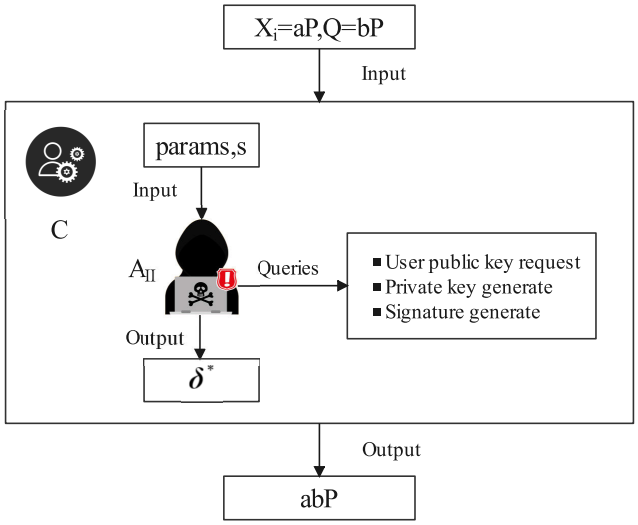

FIGURE 4. Framework for security proof of Lemma 2.

- User public key queries: Challenger C holds the list $L_U = (PID_{i,j}, r_i, x_i)$. Upon receiving a query on a pseudonym $PID_{i,j}$, the challenger C will look for $(PID_{i,j}, r_i, x_i)$ in L_U. If it is found in L_U, then C outputs (r_iP, x_iP). Otherwise, C dose the following.
  1) If $PID_{i,j} = PID^*$, C chooses r_i randomly and sets $X_i = aP$, then adds $(PID_{i,j}, r_i, *)$ to L_U. C sends $PK_i = (r_iP, X_i)$ to A_{II}.
  2) If $PID_{i,j} \neq PID^*$, C chooses r_i, x_i randomly and sets $R_i = r_iP$, $X_i = x_iP$, then adds $(PID_{i,j}, r_i, x_i)$ to L_U. C sends $PK_i = (R_i, X_i)$ to A_{II}.
- Secret value queries: A_{II} makes a query on a pseudonym $PID_{i,j}$. If $PID_{i,j} = PID^*$, challenger C terminates and fails. Otherwise, C looks up L_U for x_i and responds with x_i.
- Signature queries: When A_{II} submits a query on a tuple of $(PID_{i,j}, m_i||TS_i)$, C first runs User public key queries, Secret value queries, and searches the list L_U to obtain the values of R_i, x_i, and r_i. Subsequently, C computes $k_i = H_2(PID_{i,j}, R_i)$, $d_i = r_i + k_ismodq$, and $h_i = H_3(m_i||TS_i, PID_{i,j}, U_i, V_i, W_i, PK_i)$. C chooses u_i randomly, then computes $U_i = u_iP$, $V_i = u_iQ$, and $W_i = (d_i + h_ix_i)Q + V_i$. Finally, C returns $\sigma_i = (U_i, V_i, W_i)$ to A_{II} as the signature of $(PID_{i,j}, m_i||TS_i)$. The signature σ_i generated in this way is valid.

Forgery. A_{II} outputs a forged aggregate signature $\sigma_i^* = (U_i^*, V_i^*, W_i^*)$ on message $m_i^*||TS_i^*$, and fulfills the conditions as defined in Section IV.

Solve CDHP. If $PID_{i,j} \neq PID^*$, then the game is terminated. Otherwise, $PID_{i,j} = PID^*$, so $PK_i^* = (r_i^*P, aP)$ and $Q = bP$. C searches the list L_U to get r_i^*, and computes $k_i^* = H_2(PID^*, R_i^*)$ and $h_i^* = H_3(m_i^*||TS_i^*, PID^*, U_i^*, V_i^*, W_i^*, PK_i^*)$. Since σ_i^* is a valid

signature, $U_i^* = u_i^* P$, $V_i^* = u_i^* Q$, and $W_i^* = (d_i^* + h_i^* x_i^*)Q + V_i^* = (r_i^* + k_i^* s + h_i^* a)bP + V_i^*$. Thus, C can solve CDHP by calculating $abP = h_i^{*-1}(W_i^* - V_i^* - (r_i^* + k_i^* s)bP)$.

Similarly, A_{II} can return a forged aggregate signature $\sigma^* = (U^*, V^*, W^*)$ on message $m_i^* || TS_i^*$ for $i = 1, 2, .., n$, where $U^* = \sum_{i=1}^{n} U_i^*$, $V^* = \sum_{i=1}^{n} V_i^*$, $W^* = \sum_{i=1}^{n} W_i^*$. There exists at least one user $PID_{k,j}^*$, A_{II} did not perform Secret value queries, which implies that σ_k^* is a forged signature on $m_k^* || TS_k^*$. If $PID_{k,j}^* = PID^*$, $PK_k^* = (r_k^* P, aP)$, and $Q = bP$. C can solve CDHP by executing the following procedures.

- Calculate $h_i^* = H_3(m_i^* || TS_i^*, PID_{i,j}^*, U_i^*, V_i^*, W_i^*, PK_i^*)$ for $i = 1, 2, \ldots, n$.
- Search L_U to find r_i^*, and compute $k_i^* = H_2(PID_{i,j}^*, R_i^*)$, $d_i^* = r_i^* + k_i^* s$ for $i \neq k$.
- Compute $W_i^* = (d_i^* + h_i^* x_i^*)Q + V_i^*$ for $i \neq k$.
- Compute $W_k^* = W^* - \sum_{i=1, i\neq k}^{n} W_i^*$, where $W_k^* = (r_k^* + k_k^* s + h_k^* a)bP + V_k^*$.
- Search L_U to find x_k^*, and compute $k_k^* = H_2(PID_{k,j}^*, R_k^*)$.
- Thus, C can solve CDHP by calculating $abP = h_k^{*-1}(W_k^* - V_k^* - (r_k^* + k_k^* s)bP)$.

B. ANALYSIS OF SECURITY REQUIREMENTS

1) Authentication and integrity: Authentication and integrity are implemented by EUF-CMA proofs of *Theorem 1* and *Theorem 2*. The validity of the signatures is verified by executing the algorithm *Aggregate Verification*, which ensures that the proposed CLAS scheme satisfies these two security requirements.
2) Nonrepudiation: Because the TRA can link to the true identity ID_i based on the pseudonym $PID_{i,j}$, vehicles cannot deny the corresponding signature σ_i generated by themselves. Hence, the proposed CLAS scheme satisfies nonrepudiation.
3) Anonymity: In our CLAS scheme, only the pseudonyms PID_i of vehicles are used during the process of communication with RSUs and other vehicles. Except for the TRA, neither the vehicles nor the RSUs know the true identity ID_i of the vehicle. When a new vehicle Vh_i joins the VANETs, TRA generates pseudonym by running the *Pseudonym Generation* algorithm for the vehicle Vh_i: $PID_{i,1,j} = ID_i \oplus H_1(kP + T_{i,j})$, $PID_{i,j} = \{PID_{i,1,j}, T_{i,j}\}$.
4) Unlinkability: Vehicle Vh_i sends $(m_i, TS_i, PK_i, PID_{i,j}, U_i, V_i, W_i)$ to the nearby RSU. In this scheme, the pseudonym is generated by computing $T_{i,j} = t_{i,j}P$, $PID_{i,1,j} = ID_i \oplus H_1(kP + T_{i,j})$, and $PID_{i,j} = \{PID_{i,1,j}, T_{i,j}\}$. Random number $t_{i,j}$ generates different pseudonyms for different messages on the vehicles. Each signature σ_i has a different pseudonym $PID_{i,j}$, so vehicle can not be linked to any two signatures. Therefore, the proposed scheme supports unlinkability.
5) Traceability: Only the TRA can extract the true identities ID_i of vehicles. Using the pseudonym $PID_{i,j} = \{PID_{i,1,j}, T_{i,j}\}$, the true identity of vehicle can be tracked by TRA by calculating $ID_i = PID_{i,1,j} \oplus H_1(kP + T_{i,j})$. The tracking key k is kept secure by the TRA. Therefore, only TRA can track the true identities of malicious vehicles.
6) Replay attack resistance: Timestamp TS_i is used to ensure the freshness of signature σ_i in the *Signature Generation* algorithm. Verifiers can check whether message m_i is replayed by the validity of timestamp TS_i. Hence, a malicious vehicle is unable to replay a signed message.

VII. PERFORMANCE ANALYSIS

We compare this CLAS scheme and several previous CLAS schemes in terms of security attributes, computation overhead, and communication overhead in this section.

A. SECURITY ATTRIBUTES

Table 2 shows the comparison results, including resist A_I, resist A_{II}, authentication, anonymity, unlinkability, traceability, RAT, and security model. Here, the symbol $\surd$ stands for satisfying the requirement, and the symbol $\times$ stands for not satisfying the requirement. RAT represents replay attack resistance. The results show that this scheme, and schemes [15], [20] satisfy all the security attributes, whereas the other schemes only satisfy some of the attributes. However, ROM has been used to prove security in schemes [9], [15], [18]–[22]. The schemes that are provably secure in ROM may be insecure in actual implementation. The proposed scheme proved secure in SM, which is more practical for the actual application of VANETs.

B. COMPUTATION OVERHEAD

Omnet++, Veins, Sumo and Miracl are carried out in this article for simulation experiments. The method of evaluation developed by Cui *et al.* scheme [40] was applied to the performance analysis. They used a bilinear map cryptographic algorithm $e : G_1 \times G_1 \rightarrow G_2$ to achieve the 80-bit security level. G_1 is an additive cyclic group with the generator P which is realized on a super singular elliptic curve $E : y^2 = x^3 + xmodp$, where p is 512 bits and q is 160 bits. Table 3 shows the executing times of cryptographic operations.

Some cryptographic notations are defined as followings:

- T_{bp}: The time for executing a bilinear pairing operation $e(P_1, P_2)$, where $P_1, P_2 \in G_1$.
- T_{mtp}: The time for executing a map-to-point hash function of the bilinear pairing.
- T_{mul}: The time for executing a scalar multiplication operation aP of the bilinear pairing ($P \in G_1, a \in Z_q^*$).
- T_{pa}: The time for executing a point addition operation $P_1 + P_2$ of the bilinear pairing ($P_1, P_2 \in G_1$).
- T_h: The time for executing a one-way hash function.

Assume that RSU requires to aggregate n signatures from vechiles. In Kumar *et al.* [9], the full computational cost in signature generation is $T_{mtp} + 4T_{mul} + 2T_{pa} + 2T_h \approx 10.149ms$. The full computation overhead in single signature verification is $4T_{bp} + 2T_{mtp} + 3T_{mul} + 2T_h \approx 70.8071ms$. The full computational cost in aggregate verification is $4T_{bp} + (n+1)T_{mtp} + 3nT_{mul} + 3(n-1)T_{pa} + 2nT_h \approx 8.2174n + 62.5898ms$.

TABLE 2. Security attributes comparison.

Scheme	Resist A_I	Resist A_{II}	Authentication	Anonymity	Unlinkability	Traceability	RAT	Model
Kumar et al. [9]	√	√	√	√	×	×	×	ROM
Mei et al. [15]	√	√	√	√	√	√	√	ROM
Wang et al. [18]	√	×	√	√	×	√	×	ROM
Zhao et al. [19]	√	√	√	√	√	√	×	ROM
Kamil et al. [20]	√	√	√	√	√	√	√	ROM
Xu et al. [21]	√	×	√	×	×	×	×	ROM
Altaf *et al.* [22]	√	√	√	√	×	√	√	ROM
Our proposed scheme	√	√	√	√	√	√	√	SM

TABLE 3. Running time of cryptographic operations.

Cryptographic Operations	T_{bp}	T_{mtp}	T_{mul}	T_{pa}	T_h
Running Time(ms)	15.0738	2.3366	1.9456	0.014	0.001

In Mei *et al.* [15], the full computational cost in signature generation is $2T_{mtp} + 4T_{mul} + 2T_{pa} + T_h \approx 12.4846ms$. The full computation overhead in single signature verification is $4T_{bp} + 2T_{mtp} + 2T_{mul} + T_h \approx 68.8606ms$. The full computational cost in aggregate verification is $4T_{bp}+2T_{mtp}+2nT_{mul} + (2n-2)T_{pa} + nT_h \approx 3.9202n + 69.9404ms$. In Wang *et al.* [18], the full computational cost in signature generation is $4T_{mul} + 2T_{pa} + T_h \approx 7.8114ms$. The full computation overhead in single signature verification is $3T_{bp} + T_{mtp} + 3T_{mul} + T_{pa} + T_h \approx 53.4098ms$. The total computational cost in aggregate verification is $3T_{bp}+nT_{mtp}+3nT_{mul} + (3n-2)T_{pa} + nT_h \approx 8.2254n + 45.1934ms$. In Zhao *et al.* [19], the full computational cost in signature generation is $2T_{mtp}+4T_{mul}+2T_{pa}+2T_h \approx 12.4856ms$. The full computation overhead in single signature verification is $4T_{bp} + 2T_{mtp} + 2T_{mul} + T_{pa} + 2T_h \approx 68.8756ms$. The full computational cost in aggregate verification is $4T_{bp}+2T_{mtp}+2nT_{mul} + (4n-3)T_{pa} + 2nT_h \approx 3.9492n + 64.9264ms$. In Kamil *et al.* [20], the total computational cost in signature generation is $T_{mtp} + 4T_{mul} + 2T_{pa} + 2T_h \approx 10.149ms$. The full computation overhead in single signature verification is $3T_{bp} + 2T_{mtp} + 2T_{mul} + T_{pa} + T_h \approx 53.8008ms$. The full computational cost in aggregate verification is $3T_{bp}+nT_{mtp}+2nT_{mul} + (2n-1)T_{pa} + nT_h \approx 6.2568n + 45.2074ms$. In Xu *et al.* [21], the full computational cost in signature generation is $T_{mtp} + 3T_{mul} + T_{pa} + 2T_h \approx 8.1894ms$. The full computation overhead in single signature verification is $3T_{bp} + 2T_{mtp} + 2T_{mul} + T_{pa} + 2T_h \approx 53.8018ms$. The full computational cost in aggregate verification is $3T_{bp} + (n+1)T_{mtp}+2nT_{mul}+(3n-2)T_{pa}+2nT_h \approx 6.2718n+47.544ms$. In Altaf *et al.* [22], the full computational cost in signature generation is $T_{mtp} + 2T_{mul} + T_{pa} + T_h \approx 6.2428ms$. The full computation overhead in single signature verification is $3T_{bp} + 2T_{mtp} + T_{mul} + T_{pa} + T_h \approx 51.8552ms$. The full computational cost in aggregate verification is $3T_{bp} + (n+1)T_{mtp} + nT_{mul} + (4n-3)T_{pa} + nT_h \approx 4.3392n + 47.516ms$. In our proposed scheme, the computation of signature generation requires no map-to-point operations. The full computational overhead of this step is $3T_{mul} + T_{pa} + T_h \approx 5.8518ms$. The computation of single signature verification needs only two bilinear pairing operations and no map-to-point hash functions. The full computational cost of this step is $2T_{bp} + 2T_{mul} + 3T_{pa} + 2T_h \approx 34.0828ms$. The computation of aggregate verification also requires only two bilinear pairing operations and no Map-To-Point hash functions. Consequently, the full computation overhead of this step is $2T_{bp} + 2nT_{mul} + 3nT_{pa} + 2nT_h \approx 3.9352n + 30.1476ms$.

We estimate the computation overhead of this scheme and several related CLAS schemes for VANETs [9], [15], [18]–[22] in Table 4 using a simple and visual method. The computation overhead of signing and verifying one message of the seven CLAS schemes is expressed in Fig. 5. As illustrated in Table 4 and Fig. 5, the proposed CLAS scheme exhibits a lower computation overhead between signature generation and single signature verification than the other seven CLAS schemes for VANETs. The aggregate verification delays of the seven schemes involved in this paper are shown in Fig. 6. Apparently, the aggregate verification delay increases linearly with the increase of signatures. From the results shown in Fig. 6, the aggregate verification cost in this scheme is lower than that in [9], [15], [18]–[22].

C. COMMUNICATION OVERHEAD

We carry out the communication overhead analysis with the main consideration of signature and timestamp. According to the aforementioned, the length of p is 64 bytes, so the length of a unit in G_1 is 128 bytes. Furthermore, the length of hash function and timestamp is 20 bytes and 4 bytes, respectively. We estimate the communication overhead, as shown in Table 5 and Fig. 7. We assume that RSU receives n signatures and sends an aggregate signature. In order to facilitate calculation and comparison, we set $n = 100$.

In Kumar *et al.* [9], the length of single signature is $2 \times 128 = 256$ bytes, and the length of aggregate signature is $(n+1) \times 128 = (100+1) \times 128 = 12928$ bytes. In Mei *et al.* [15], the length of single signature is $2 \times 128 + 4 = 260$ bytes, and the length of aggregate signature is $2 \times 128 + 4n = 2 \times 128 + 4 \times 100 = 656$ bytes.

TABLE 4. Computation overhead comparison.

Scheme	Signature Generation Cost (ms)	Single Signature Verification Cost (ms)	Aggregate Verification Cost (ms)
Kumar et al. [9]	$T_{mtp}+4T_{mul}+2T_{pa}$ $+2T_h \approx 10.149$	$4T_{bp}+2T_{mtp}+3T_{mul}$ $+2T_h \approx 70.8072$	$4T_{bp}+(n+1)T_{mtp}+3nT_{mul}$ $+3(n-1)T_{pa}+2nT_h$ $\approx 8.2174n+62.5898$
Mei et al. [15]	$2T_{mtp}+4T_{mul}+2T_{pa}$ $+T_h \approx 12.4846$	$4T_{bp}+2T_{mtp}+2T_{mul}$ $+T_h \approx 68.8606$	$4T_{bp}+2T_{mtp}+2nT_{mul}$ $+(2n-2)T_{pa}+nT_h$ $\approx 3.9202n+69.9404$
Wang et al. [18]	$4T_{mul}+2T_{pa}+T_h$ ≈ 7.8114	$3T_{bp}+T_{mtp}+3T_{mul}$ $+T_{pa}+T_h \approx 53.4098$	$3T_{bp}+nT_{mtp}+3nT_{mul}$ $+(3n-2)T_{pa}+nT_h$ $\approx 8.2254n+45.1934$
Zhao et al. [19]	$2T_{mtp}+4T_{mul}+2T_{pa}$ $+2T_h \approx 12.4856$	$4T_{bp}+2T_{mtp}+2T_{mul}$ $+T_{pa}+2T_h \approx 68.8756$	$4T_{bp}+2T_{mtp}+2nT_{mul}$ $+(4n-3)T_{pa}+2nT_h$ $\approx 3.9492n+64.9264$
Kamil et al. [20]	$T_{mtp}+4T_{mul}+2T_{pa}$ $+2T_h \approx 10.149$	$3T_{bp}+2T_{mtp}+2T_{mul}$ $+T_{pa}+T_h \approx 53.8008$	$3T_{bp}+(n+1)T_{mtp}+2nT_{mul}$ $+(2n-1)T_{pa}+nT_h$ $\approx 6.2568n+47.544$
Xu et al. [21]	$T_{mtp}+3T_{mul}+T_{pa}$ $+2T_h \approx 8.1894$	$3T_{bp}+2T_{mtp}+2T_{mul}$ $+T_{pa}+2T_h \approx 53.8018$	$3T_{bp}+nT_{mtp}+2nT_{mul}$ $+(3n-2)T_{pa}+2nT_h$ $\approx 6.2718n+45.1934$
Altaf *et al.* [22]	$T_{mtp}+2T_{mul}+T_{pa}$ $+T_h \approx 6.2428$	$3T_{bp}+2T_{mtp}+T_{mul}$ $+T_{pa}+T_h \approx 51.8552$	$3T_{bp}+(n+1)T_{mtp}+nT_{mul}$ $+(4n-3)T_{pa}+nT_h$ $\approx 4.3392n+47.516$
Our proposed scheme	$3T_{mul}+T_{pa}+T_h$ ≈ 5.8518	$2T_{bp}+2T_{mul}+3T_{pa}$ $+2T_h \approx 34.0828$	$2T_{bp}+2nT_{mul}+3nT_{pa}$ $+2nT_h \approx 3.9352n+30.1476$

TABLE 5. Communication overhead comparison.

Scheme	Single signature length	Size (bytes)	Aggregate signature length	Size (bytes, $n=100$)
Kumar et al. [9]	$2\|G_1\|$	256	$(n+1)\|G_1\|$	12928
Mei et al. [15]	$2\|G_1\|+\|timestamp\|$	260	$2\|G_1\|+n\|timestamp\|$	656
Wang et al. [18]	$2\|G_1\|$	256	$(n+1)\|G_1\|$	12928
Zhao et al. [19]	$2\|G_1\|$	256	$(n+1)\|G_1\|$	12928
Kamil et al. [20]	$2\|G_1\|+\|timestamp\|$	260	$2\|G_1\|+n\|timestamp\|$	656
Xu et al. [21]	$2\|G_1\|$	256	$(n+1)\|G_1\|$	12928
Altaf *et al.* [22]	$2\|G_1\|+\|timestamp\|$	260	$(n+1)\|G_1\|+n\|timestamp\|$	13328
Our proposed scheme	$3\|G_1\|+\|timestamp\|$	388	$3\|G_1\|+n\|timestamp\|$	784

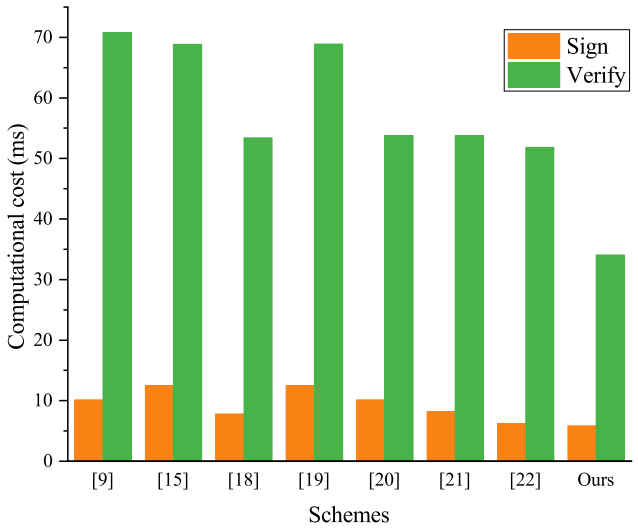

FIGURE 5. Computation overhead of signing and verifying one message.

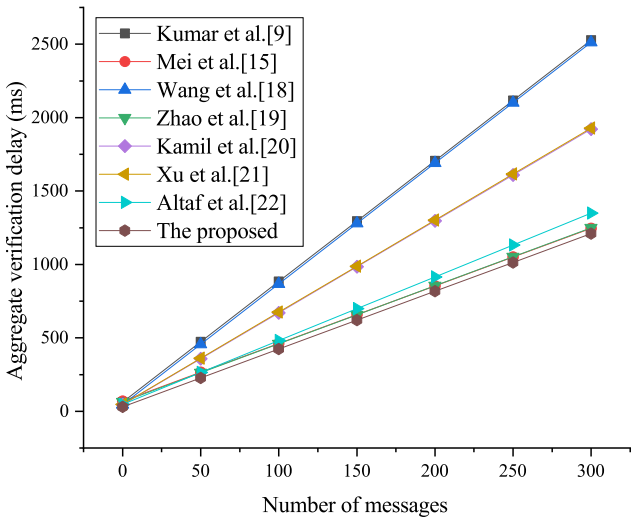

FIGURE 6. Aggregate verification delay under the influence of the number of messages.

In Wang *et al.* [18], the length of single signature is $2\times 128 = 256$ bytes, and the length of aggregate signature is $(n+1)\times 128 = (100+1)\times 128 = 12928$ bytes. In Zhao *et al.* [19], the length of single signature is $2 \times 128 = 256$ bytes, and the length of aggregate signature is $(n+1)\times 128 = (100+1)\times 128 = 12928$ bytes. In Kamil *et al.* [20], the

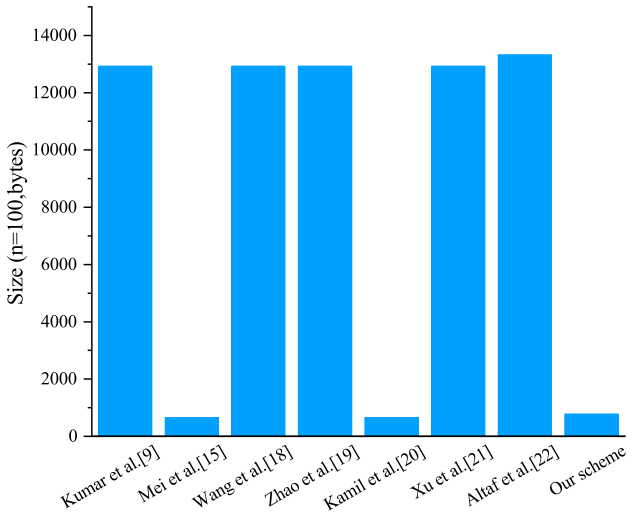

FIGURE 7. Aggregate signature size.

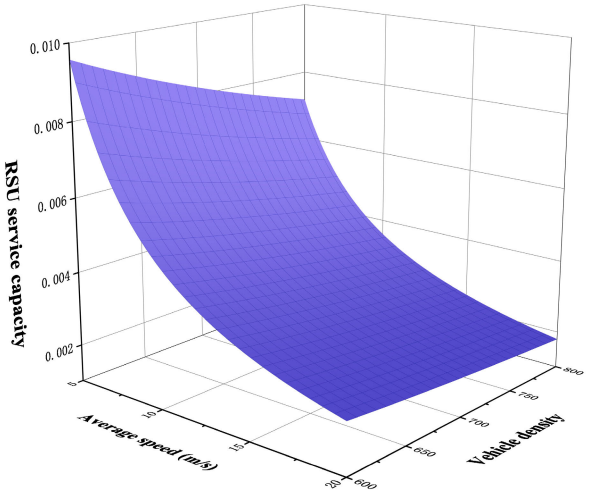

FIGURE 8. *Rsc* with various vehicle speed and density.

length of single signature is $2 \times 128 + 4 = 260$ bytes, and the length of aggregate signature is $2 \times 128 + 4n = 2 \times 128 + 4 \times 100 = 656$ bytes. In Xu *et al.* [21], the length of single signature is $2 \times 128 = 256$ bytes, and the length of aggregate signature is $(n + 1) \times 128 = (100 + 1) \times 128 = 12928$ bytes. In Altaf *et al.* [22], the length of single signature is $2 \times 128 + 4 = 260$ bytes, and the length of aggregate signature is $(n + 1) \times 128 + n \times 4 = (100 + 1) \times 128 + 100 \times 4 = 13328$ bytes. In this CLAS scheme, the length of single signature is $3 \times 128 + 4 = 388$ bytes, which contains three units in G_1 and one timestamp. The length of aggregate signature is $3 \times 128 + 4n = 3 \times 128 + 4 \times 100 = 784$ bytes, which contains three units in G_1 and n timestamps.

Fig. 7 shows that the communication overhead generated from this scheme is lower than those of schemes [9], [18], [19], [21], [22], and slightly larger than those of schemes [15], [20]. However, compared with schemes [15], [20], our scheme characterizes a lower computation overhead, and the security proofs of our CLAS scheme are formally given in the SM.

D. PRACTICABILITY EVALUATION

For evaluating the processing capacity of RSU, the RSU service capacity *Rsc* is introduced, which is calculated as follows [41]:

$$Rsc = \frac{p \cdot d}{T_{ver} \cdot N \cdot v}. \tag{3}$$

T_{ver} indicates the time for verifying a single signature. Here T_{ver} is $34.0828ms$. Let N indicate the density of vehicles in the area covered by the RSU that varies from 600 to 800 m, v indicates the vehicle average speed that varies from 5 m/s to 20 m/s, p indicates the probability that the signature is valid and d indicates the communication range of the RSU coverage area which is considered to be 1000 m. It is evident from Fig. 8 that the *Rsc* gradually decreases with increasing vehicle density and speed. In addition, RSU can verify eight signatures every 300 ms. Thus, it can be concluded that the vehicle density is supposed to be decreased to get higher RSU service capacity *Rsc* in this scheme.

VIII. CONCLUSION

To realize privacy protection in V2I communication for VANETs, a wide range of provably secure CLAS schemes in ROM have been proposed. However, the schemes that proved secure in ROM may not be secure in the actual implementation. A conditional CLAS scheme with privacy protection was recommended in this paper, which satisfies the security requirements of VANETs, and demonstrates that it is EUF-CMA against type I and II adversaries in the SM under the CDHP. Lastly, the performance evaluation indicates that this scheme is more effective considering security attributes, computation and communication cost than other relevant schemes. Therefore, this CLAS scheme is high availability in VANETs with lower verification delay and more security properties.

REFERENCES

[1] C. Cseh, "Architecture of the dedicated short-range communications (DSRC) protocol," in *Proc. 48th IEEE VTC*, vol. 3, May 1998, pp. 2095–2099.
[2] Z. H. Mir and F. Filali, "LTE and IEEE 802.11 P for vehicular networking: A performance evaluation," *EURASIP J. Wireless Commun. Netw.*, vol. 2014, no. 1, pp. 1–15, 2014.
[3] M. S. Sheikh and J. Liang, "A comprehensive survey on VANET security services in traffic management system," *Wireless Commun. Mobile Comput.*, vol. 2019, pp. 1–23, Sep. 2019.
[4] D. Boneh, C. Gentry, B. Lynn, and H. Shacham, "Aggregate and verifiably encrypted signatures from bilinear maps," in *Proc. Int. Conf. theory Appl. Cryptograph. Techn.* Springer, 2003, pp. 416–432.
[5] S. S. Al-Riyami and K. G. Paterson, "Certificateless public key cryptography," in *Proc. Int. Conf. theory Appl. Cryptol. Inf. Secur.* Springer, 2003, pp. 452–473.

[6] H.-R. Tseng, R.-H. Jan, W. Yang, and E. Jou, "A secure aggregated message authentication scheme for vehicular ad-hoc networks," in *Proc. 18th World Congr. Intell. Transp. Syst.*, 2011, pp. 1–14.

[7] P. Kumar and V. Sharma, "On the security of certificateless aggregate signature scheme in vehicular ad hoc networks," in *Soft Computing: Theories and Applications*. Springer, 2018, pp. 715–722.

[8] X. Yang, C. Chen, T. Ma, Y. Li, and C. Wang, "An improved certificateless aggregate signature scheme for vehicular ad-hoc networks," in *Proc. IEEE 3rd Adv. Inf. Technol., Electron. Autom. Control Conf. (IAEAC)*, Oct. 2018, pp. 2334–2338.

[9] P. Kumar, S. Kumari, V. Sharma, X. Li, A. K. Sangaiah, and S. H. Islam, "Secure CLS and CL-AS schemes designed for VANETs," *J. Supercomput.*, vol. 75, no. 6, pp. 3076–3098, Jun. 2019.

[10] Y. Ming and H. Cheng, "Efficient certificateless conditional privacy-preserving authentication scheme in VANETs," *Mobile Inf. Syst.*, vol. 2019, pp. 1–19, Feb. 2019.

[11] X. Hu, W. Tan, C. Ma, and H. Xu, "Certificateless aggregate signature scheme with high efficencicy in vehicular ad-hoc network," in *Proc. 4th Int. Conf. Electron. Inf. Technol. Comput. Eng.*, Nov. 2020, pp. 1008–1012.

[12] Y. Zhao, Y. Hou, L. Wang, S. Kumari, M. K. Khan, and H. Xiong, "An efficient certificateless aggregate signature scheme for the Internet of Vehicles," *Trans. Emerg. Telecommun. Technol.*, vol. 31, no. 5, p. e3708, May 2020.

[13] G. Thumbur, G. S. Rao, P. V. Reddy, N. B. Gayathri, D. V. R. K. Reddy, and M. Padmavathamma, "Efficient and secure certificateless aggregate signature-based authentication scheme for vehicular ad hoc networks," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1908–1920, Feb. 2021.

[14] X. Ye, G. Xu, X. Cheng, Y. Li, and Z. Qin, "Certificateless-based anonymous authentication and aggregate signature scheme for vehicular ad hoc networks," *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–16, Mar. 2021.

[15] Q. Mei, H. Xiong, J. Chen, M. Yang, S. Kumari, and M. K. Khan, "Efficient certificateless aggregate signature with conditional privacy preservation in IoV," *IEEE Syst. J.*, vol. 15, no. 1, pp. 245–256, Mar. 2021.

[16] M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," in *Proc. 1st ACM Conf. Comput. Commun. Secur. (CCS)*, 1993, pp. 62–73.

[17] R. Cramer and V. Shoup, "A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack," in *Proc. Annu. Int. Cryptol. Conf.* Springer, 1998, pp. 13–25.

[18] W. Daxing and T. Jikai, "Probably secure cetificateless aggregate signature algorithm for vehicular ad hoc network," *J. Electron. Inf. Technol.*, vol. 40, no. 1, pp. 11–17, 2018.

[19] N. Zhao and G. Zhang, "Privacy-protected certificateless aggregate signature scheme in VANET," in *Proc. 11th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2019, pp. 1–6.

[20] I. A. Kamil and S. O. Ogundoyin, "On the security of privacy-preserving authentication scheme with full aggregation in vehicular ad hoc network," *Secur. Privacy*, vol. 3, no. 3, p. e104, May 2020.

[21] Z. Xu, D. He, N. Kumar, and K.-K.-R. Choo, "Efficient certificateless aggregate signature scheme for performing secure routing in VANETs," *Secur. Commun. Netw.*, vol. 2020, pp. 1–12, Feb. 2020.

[22] F. Altaf and S. Maity, "PLHAS: Privacy-preserving localized hybrid authentication scheme for large scale vehicular ad hoc networks," *Veh. Commun.*, vol. 30, Aug. 2021, Art. no. 100347. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214209621000164

[23] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *J. Comput. Secur.*, vol. 15, no. 1, pp. 39–68, 2007.

[24] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proc. Workshop Theory Appl. Cryptograph. Techn.* Springer, 1984, pp. 47–53.

[25] R. Castro and R. Dahab, "Efficient certificateless signatures suitable for aggregation," in *Proc. IACR Cryptol. ePrint Arch.*, 2007, p. 454.

[26] L. Zhang and F. Zhang, "A new certificateless aggregate signature scheme," *Comput. Commun.*, vol. 32, no. 6, pp. 1079–1085, 2009.

[27] K.-A. Shim, "On the security of a certificateless aggregate signature scheme," *IEEE Commun. Lett.*, vol. 15, no. 10, pp. 1136–1138, Oct. 2011.

[28] H. Xiong, Z. Guan, Z. Chen, and F. Li, "An efficient certificateless aggregate signature with constant pairing computations," *Inf. Sci.*, vol. 219, pp. 225–235, Jan. 2013.

[29] L. Cheng, Q. Wen, Z. Jin, H. Zhang, and L. Zhou, "Cryptanalysis and improvement of a certificateless aggregate signature scheme," *Inf. Sci.*, vol. 295, pp. 337–346, Feb. 2015.

[30] S.-J. Horng, S.-F. Tzeng, P.-H. Huang, X. Wang, T. Li, and M. K. Khan, "An efficient certificateless aggregate signature with conditional privacy-preserving for vehicular sensor networks," *Inf. Sci.*, vol. 317, pp. 48–66, Oct. 2015.

[31] M. Azees, P. Vijayakumar, and L. J. Deboarh, "EAAP: Efficient anonymous authentication with conditional privacy-preserving scheme for vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2467–2476, Sep. 2017.

[32] N. B. Gayathri, G. Thumbur, P. V. Reddy, and Z. U. R. Muhammad, "Efficient pairing-free certificateless authentication scheme with batch verification for vehicular ad-hoc networks," *IEEE Access*, vol. 6, pp. 31808–31819, 2018.

[33] H. Zhong, S. Han, J. Cui, J. Zhang, and Y. Xu, "Privacy-preserving authentication scheme with full aggregation in VANET," *Inf. Sci.*, vol. 476, pp. 211–221, Feb. 2019.

[34] J. Cui, L. Wei, H. Zhong, J. Zhang, Y. Xu, and L. Liu, "Edge computing in VANETs—An efficient and privacy-preserving cooperative downloading scheme," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 6, pp. 1191–1204, Jun. 2020.

[35] J. Zhang, J. Cui, H. Zhong, Z. Chen, and L. Liu, "PA-CRT: Chinese remainder theorem based conditional privacy-preserving authentication scheme in vehicular ad-hoc networks," *IEEE Trans. Dependable Secure Comput.*, vol. 18, no. 2, pp. 722–735, Mar./Apr. 2021.

[36] Y. Ren, X. Li, S.-F. Sun, X. Yuan, and X. Zhang, "Privacy-preserving batch verification signature scheme based on blockchain for vehicular ad-hoc networks," *J. Inf. Secur. Appl.*, vol. 58, May 2021, Art. no. 102698. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214212620308450

[37] I. A. Kamil and S. O. Ogundoyin, "A lightweight certificateless authentication scheme and group key agreement with dynamic updating mechanism for LTE-V-based Internet of Vehicles in smart cities," *J. Inf. Secur. Appl.*, vol. 63, Dec. 2021, Art. no. 102994. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214212621002027

[38] L. Deng, B. Ning, and Y. Jiang, "A lightweight certificateless aggregation signature scheme with provably security in the standard model," *IEEE Syst. J.*, vol. 14, no. 3, pp. 4242–4251, Sep. 2020.

[39] F. Zhang, R. Safavi-Naini, and W. Susilo, "An efficient signature scheme from bilinear pairings and its applications," in *Proc. Int. Workshop Public Key Cryptogr.* Springer, 2004, pp. 277–290.

[40] J. Cui, D. Wu, J. Zhang, Y. Xu, and H. Zhong, "An efficient authentication scheme based on semi-trusted authority in VANETs," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2972–2986, Mar. 2019.

[41] P. Vijayakumar, V. Chang, L. J. Deborah, B. Balusamy, and P. G. Shynu, "Computationally efficient privacy preserving anonymous mutual and batch authentication schemes for vehicular ad hoc networks," *Future Generat. Comput. Syst.*, vol. 78, pp. 943–955, Jan. 2018.

HUIWEN WANG is currently pursuing the M.S. degree with the Academy of Computer Science and Technology, Shanghai University of Electric Power (SUEP), Shanghai, China. Her research interests include vehicle ad hoc networks and privacy preserving.

LIANGLIANG WANG received the Ph.D. degree from Shanghai Jiao Tong University (SJTU), China, in 2016. He is currently working as an Associate Professor with the Academy of Computer Science and Technology, SUEP, China. His research interests include the information security and smart grid.

KAI ZHANG received the B.S. degree in computer science and technology from Shandong Normal University, China, in 2012, and the Ph.D. degree in computer science and technology from East China Normal University, China, in 2017. He is currently an Associate Professor with SUEP, China. He is mainly engaged in fields of applied cryptography and information security.

JINGUO LI received the B.S. degree in information security and the Ph.D. degree in computer science and technology from Hunan University, China, in 2007 and 2014, respectively. He is currently an Associate Professor with the College of Computer Science and Technology, SUEP. His research interests include information security and applied cryptography.

YIYUAN LUO received the B.S. degree in computer science and engineering and the Ph.D. degree in computer security from Shanghai Jiao Tong University, Shanghai, China, in 2007 and 2013, respectively.

From 2009 to 2010, he was a Research Assistant with the Department of Electrical and Computer Engineering, University of Waterloo. Since 2019, he has been an Associate Professor with Huizhou University, Huizhou, China. He is the author of more than 20 articles. His research interests include security analysis and design of cryptographic schemes.