Traceable and Privacy-Preserving Authentication Scheme for Energy Trading in V2G Networks

Shen, Gang; Xia, Chengliangyi; Li, Yumei; Shen, Hua; Meng, Weizhi; Zhang, Mingwu

Published in:
Ieee Internet of Things Journal

Link to article, DOI:
10.1109/JIOT.2023.3311800

Publication date:
2024

Document Version
Peer reviewed version

Traceable and Privacy-Preserving Authentication Scheme for Energy Trading in V2G Networks

Gang Shen, Chengliangyi Xia, Yumei Li, Hua Shen, Weizhi Meng, and Mingwu Zhang*

***Abstract*—With the rapid popularization of electric vehicles (EVs) in modern society, vehicle-to-grid (V2G) has been widely concerned as an emerging technology. However, various privacy and security issues arise frequently in the energy interaction between EVs and the smart grid (SG), such as the lack of secure authentication and disclosure of EVs' identity. Although many crypto-based schemes are proposed to achieve secure authentication of V2G networks, they rely on certification authority (CA) or private key generator (PKG). In response to this problem, some certificateless signature-based schemes have been proposed. Nevertheless, most of them are not suitable for V2G networks due to the high computational cost and communication overhead, and they do not consider the problem of tracking illegal signatures. Therefore, we propose a traceable and privacy-preserving authentication scheme with supporting batch verification for energy trading in V2G networks. We use the method of binary tree level traversal to quickly track EVs with illegal signatures, which can reduce computational resources. Besides, the proposed scheme is easier to be deployed in real world because of avoiding the problems of key escrow and certificate management. Finally, we conduct a comprehensive security analysis and performance evaluation regarding our scheme. We prove that our proposed scheme is secure under the random oracle model (ROM), and the experimental results illustrate that the proposed scheme has less computational cost and communication overhead as compared to the existing schemes.**

***Index Terms*—Vehicle-to-grid (V2G), authentication, certificateless signature, batch verification, random oracle model (ROM).**

I. Introduction

With the rapid development of new energy vehicles and smart grid (SG) technology, the storage, transfer and sharing of energy by private electric vehicles (EVs) has become a reality. Vehicle-to-grid (V2G) is regarded as a key technology that enables EVs to act as loads as well as distributed storage devices connected to the grid [1]. Therefore, V2G network is a specific network, in which the energy transaction between EVs and SG can balance the requirements of SGs. As shown in Fig. 1, entities in a typical V2G network usually include a trusted authority (TA), a key generation centre (KGC), a power plant (PP), an electricity market (EM), some local aggregators (LA), EVs and charging stations (CSs). The EM is responsible for the trading and storage of electrical energy. Only registered EVs in TA can enjoy services in SG. The bidirectional communication between EVs and SG is accomplished through the CSs. The partial private key of EV can be generated by KGC. Also, all interaction information (e.g., service request messages and service response messages) can be transmitted to EM through the LAs.

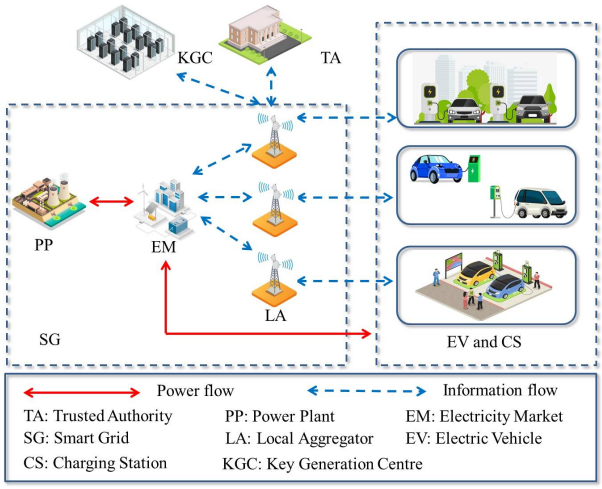


Fig. 1. V2G architecture.

According to statistics, most EVs are parked with an average of 95% of daily time, and their surplus energy is usually sold to the grid during peak periods [2]. As a result, the batteries of EVs are used as the buffer for peak regulation and reactive power compensation, which plays a vital role in the reliable and stable operation of the grid. Obviously, V2G network not only brings us with some economic compensation, but also ensures the stable operation of SG. However, with the large-scale deployment of network communications in V2G networks, traditional V2G networks are facing cyber-security risks and challenges such as man-in-the-middle (MITM) attacks, impersonation attacks, spoofing attacks, replay attacks, etc. In order to prevent these attacks from breaking in, an anonymous mutual authentication between EVs and SG is an essential step before EVs obtain the service from SG. If there is no anonymous mutual authentication stage, the adversary can impersonate the legal CS to eavesdrop the EV's private information from the interactive messages, such as the owner's identity, transaction record, current location and battery status. With this information, malicious adversary may easily infer

Manuscript received xx xxx 2023; revised xx xxx 2023 and xx xxx 2023; accepted xx xxx 2023. This work was supported in part by the Major Research Plan of Hubei Province under Grant/Award No. 2023BAA027, and the National Natural Science Foundation of China under grants 62072134, U2001205, and the key Research and Development Program of Hubei Province under Grant 2021BEA163. *(Corresponding author: Mingwu Zhang)*

Gang Shen, Chenliangyi Xia, Yumei Li, Hua Shen, and Mingwu Zhang are with the School of Computer Science, Hubei University of Technology, Wuhan 430068, China (e-mail: shengang@hbut.edu.cn; 102111093@hbut.edu.cn; leamergo@gmail.com; nancy78733@126.com; csmwzhang@gmail.com).

Weizhi Meng is with the Department of Applied Mathematics and Computer Science, Technology University of Denmark (DTU), Denmark (e-mail: weme@dtu.dk).

the EV owner's work and home address, frequented places and living habits, which may cause a lot of trouble to EV owner [3]. In addition, it is also worth considering that EV may become an internal attacker in V2G networks. For example, the adversary personates an unregistered EV to conduct spoofing attacks and obtain illegal benefits.

To solve this problem, many authentication schemes for V2G networks have been proposed in recent years [4]–[8]. However, these schemes are based on public key infrastructure (PKI), in which the participants should rely on the certificate authority (CA). CA is a certificate issuing authority, which is mainly responsible for issuing certificates, authenticating certificates, and managing issued certificates. If CA is compromised, the security of public keys cannot be guaranteed. Besides, the issuance, renewal and revocation of certificates may cause significant computational cost and communication overhead. Currently, some studies proposed the signature technologies with the identity-based scheme [9], [10], which can solve the problem of certificate management. Unfortunately, the disadvantages of these schemes are also obvious, for example, if private key generation (PKG) is weak, the private key of user's signature will be compromised. To overcome the above problems, a large number of certificateless-based signature schemes for authentication have been proposed [11]–[20]. However, none of these certificateless-based signature schemes have shown to be effective in V2G networks.

In order to promote the security and efficiency of V2G networks, in this work, we propose a traceable and privacy-preserving authentication scheme, which can achieve better performance and feasibility in energy trading V2G networks. Specifically, the main contributions can be summarized into the following three aspects:

1) First, we present a traceable and privacy-preserving authentication scheme for V2G networks by using a designed lightweight certificateless signature (CLS) scheme, in which the privacy of EVs can be ensured and EVs with illegal signatures can be traced back. Meanwhile, the proposed scheme can reduce computational resources through avoiding the burden caused by the issuance, renewal and revocation of certificates.
2) Second, we convert the message signatures of multiple EVs into an aggregate signature and verify them in batches to save computation and communication resources. Additionally, a binary tree hierarchical traversal method is adopted to quickly track EVs with illegal signatures.
3) Finally, we prove the security of the proposed scheme in the random oracle model (ROM). The analysis results demonstrate that the proposed scheme can prevent the leakage of the signature private key, and ensure the legitimacy of EVs' identity and service request message. The evaluation results show that the proposed scheme is more efficient than existing schemes in terms of computational cost and communication overhead.

The remaining parts of this paper are organized as follows. In Section II, we discuss the related work. To facilitate the understanding of our scheme, we present the preliminaries in Section III and system model and adversary model in Section IV. Then, we describe the construction of our scheme in Section V. The security analysis and experimental results are presented in Sections VI and VII, respectively. Section VIII concludes this paper.

II. Related Work

In recent years, CLS schemes have been widely used. In this section, we overview the works relevant to the development of CLS and its application in VANETs.

A. Development of CLS

Much attention has been paid to the research on V2G networks security [21]. However, the existing schemes hardly consider the integrity and authenticity of V2G networks. In order to realize the integrity, authenticity and non-repudiation of the transmitted information, the majority of signature schemes have been proposed. In a conventional public key cryptosystem (PKC), user transfers the public key in obtained key pairs to a fully trusted certificate authority (CA) to receive his/her public key certificates. However, the management of certificates increases the maintenance cost of the system. Soon after, Shamir [22] proposes identity-based encryption (IBE), which uses user's identity as his/her public key to realize the correlation between identity and public key. Although it eliminates certificate management, it is easily to leak users' privacy due to key escrow. To resolve this issue, Paterson and Al-Riyami [23] propose the first certificateless public key cryptosystem (CL-PKC) in 2003. In this scheme, user's key consists of a secret value of his choosing and a portion of a third-party-generated private key.

Obviously, CL-PKC combines the advantages of traditional PKC and identity-based PKC, which not only has no key escrow problem but also eliminates the complexity of certificate management. Based on the advantages of CL-PKC, some specific schemes on CLS are gradually proposed. For example, Yum and Lee [24] present a general construction for CLS scheme in 2004. Huang *et al*. [25] propose a CLS scheme which is proven secure in the ROM. In [26], Zhang *et al*. construct a CLS scheme in the ROM based on scheme [27]. Liu *et al*. [28] propose the first CLS scheme in the standard model, but the scheme was proved to be not secure. Later, a more secure CLS scheme in the standard model was designed by Yuan *et al*. [29]. In 2019, Shim also proposes a new CLS scheme provably secure in the standard model [30]. Yan *et al*. [31] propose a remote data possession checking protocol (namely RDPC) to ensure the security of shared data. In this scheme, an operation record table is introduced to track the operation of file blocks in order to support data dynamics. Subsequently, Li *et al*. [32] present an identity-based remote data integrity checking scheme to check the correctness of the data without downloading them. This scheme avoids the complicated certificate management problems caused by PKI. However, key escrow in identity-based cryptography has the drawback of key disclosure. Therefore, on the basis of scheme [32], Li *et al*. [33] propose a new RDPC scheme using CLS technology, which overcomes the burden caused by certificate management and the insecurity caused by key escrow.

B. Application of CLS in VANETs

Combining the merits of aggregate signatures and certificateless encryption schemes, Castro and Dahab [34] propose the certificateless aggregate signature (CLAS) scheme for the first time in 2007. Recently, many CLAS schemes have been applied to protect privacy in vehicular ad-hoc networks (VANETs). Yang *et al*. [13] propose an improved CLAS scheme for VANETs, which can withstand the internal attack (comes from internal signers) and coalition attack (comes from the collusion between KGC and roadside unit). Next year, Kumar *et al*. [14] present an efficient and conditional privacy CLAS scheme for VANETs and demonstrate that it is proved in the ROM against adaptive chosen-message attacks. In long term evolution-vehicle system, Kamil *et al*. [15] design a certificateless authentication scheme with batch verification, which can perform the distribution of a group key. In the same year, Xu *et al*. [16] present a new CLAS scheme to solve the problem of routing authentication in VANETs. Also, Mei *et al*. [17] propose a conditional privacy-preserving CLAS scheme to resolve the issues of location privacy and message authenticity, and schemes [15]–[17] are proved to be secure under the ROM. In 2022, Wang *et al*. [18] propose a conditional privacy-preserving CLAS scheme in VANETs, which can reduce bandwidth and computation resources by using full aggregation technology. Later, Liang and Liu [19] point out that the signature in Mei *et al*.'s scheme [17] can be forged by anyone, and improve the scheme. In addition, the Type I adversary can replace the signer's public key to forge his/her legitimate signature. To solve this problem, Ma *et al*. [20] propose an efficient and provably secure CLS scheme in the application scenario of VANETs. However, there are no CLS schemes for V2G networks at present. And the computational costs of the existing signature schemes are relatively large, which are not suitable for the fast execution of authentication in V2G networks. Therefore, our work is dedicated to designing a signature schemes of certificateless PKC for V2G networks.

Recently, the application of privacy-preserving machine learning attracts increasing attention. Zhang *et al*. [35] propose a privacy-preserving decision tree evaluation scheme for e-healthcare system, which uses improved KNN and elementary matrix permutation to provide an effective medical diagnosis for e-healthcare system without revealing medical data and patient privacy. Similarly, an efficient and privacy-preserving online diagnosis scheme for e-healthcare system is proposed by Shen *et al*. [36]. To protect the data owned by multiple data providers, scheme [37] presents a privacy-preserving neural network prediction model, which can implement the prediction task safely in multiple-client model. Meanwhile, a machine learning method for training word vectors with security and privacy protection is proposed by Zhang *et al*. [38]. It is worth mentioning that both schemes [37] and [38] use the inner-product functional encryption algorithm to train the datasets provided by multiple participants, while ensuring the security of the datasets. In addition, Kang *et al*. [39] propose a traceable and forward-secure attribute-based signature scheme with constant-size. The highlight of this scheme is that it not only supports flexible threshold predicates, but also can track the authentic identity of signers with abusive signature behavior. The technologies and methods involved in these schemes provide many wonderful ideas for us to study the security of V2G networks.

III. Preliminaries

In this section, we present the relevant knowledge used in the proposed scheme, including basic CLS algorithm, bilinear pairing and full binary tree.

A. General CLS Scheme

A general CLS scheme includes six algorithms, namely, Setup, Partial Private-Key-Extract, Set-Secret-Value, Set-Public-Key, Sign and Verify. The specific description is as follows.

Setup: By inputting a security parameter κ in this algorithm, a master public/secret key pair (mpk, msk) and system parameter $params$ can be returned.

Partial Private-Key-Extract: By inputting msk, mpk, $params$ and identity ID in this algorithm, a partial private key D_{ID} can be returned.

Set-Secret-Value: By inputting mpk and $params$ in this algorithm, a secret value x_{ID} can be returned.

Set-Public-Key: By inputting mpk, $params$, ID and ID's secret value x_{ID} in this algorithm, a public key PK_{ID} can be returned.

Sign: By inputting mpk, $params$, ID, x_{ID}, D_{ID} and a message m in this algorithm, a certificateless signature σ can be returned.

Verify: After inputting mpk, $params$, ID, PK_{ID} and a message/signature pair (m, σ), if the signature is verified to be correct, the result of this algorithm is true, otherwise it is false.

B. Bilinear Pairing

Let $\mathbb{G}_1, \mathbb{G}_2$ be two cyclic groups with the same prime order q, and P, $Q \in \mathbb{G}_1$ are generators of $\mathbb{G}_1$. The bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ meets the following three properties:

Bilinearity: For any $a, b \in \mathbb{Z}_q^*$ and any $P, Q \in \mathbb{G}_1$, there is $e(aP, bQ) = e(P, Q)^{ab} \in \mathbb{G}_2$.

Non-degeneracy: $e(P, P) \neq 1_{\mathbb{G}_2}$, where $1_{\mathbb{G}_2}$ denotes the unit of $\mathbb{G}_2$.

Computability: For $P, Q \in \mathbb{G}_1$, $e(P, Q)$ can be computed by an efficient algorithm.

We can obtain a description of the Computational Diffie-Hellman (CDH) problem from [40].

Definition 1: (CDH Problem) The CDH problem is defined as: For unknown $a, b \in \mathbb{Z}_q^*$, the value of $abP \in \mathbb{G}_1$ is calculated after a tuple $(P, aP, bP) \in \mathbb{G}_1$ is given.

The CDH problem is difficult in $\mathbb{G}_1$ if no algorithms can solve the CDH problem with a non-negligible advantage ϵ in probabilistic polynomial time.

C. Full Binary Tree

A binary tree is called full if every non-leaf node in the tree has two children and all leaf nodes are at the same depth [41]. Inspired by the location operations that binary trees can quickly implement for data retrieval, we model the signature relationship as a hierarchical structure of full binary tree. Specifically, a full binary tree is constructed with the aggregated signatures of all registered EVs as the root node, in which the leaf nodes store the signatures of each registered EV in turn, and the signatures of any two child nodes will be aggregated and stored in their parent nodes. We give some possible signature binary tree structures as shown in Fig.2. In order to ensure the structure of full binary tree, we populate the free leaf nodes (no signatures are arranged) with "0". Where "+" represents an aggregate operation.

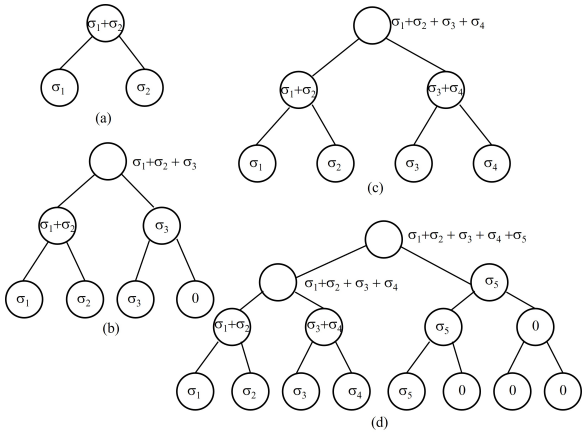


Fig. 2. Full binary tree structures with signatures.

IV. System Model and Adversary Model

In this section, we describe the system model and adversary model in detail.

A. System Model

As shown in Fig. 3, there are five parties in the system model, i.e., trusted authority (TA), key generation centre (KGC), local aggregator (LA), electric vehicle (EV) and charging station (CS) [2]. The functional descriptions of these entities are as follows:

Trusted Authority (TA): TA is a fully trusted third party, which is mainly in charge of system initialization, vehicle registration, signature verification and tracking of vehicles with illegal signatures.

Key Generation Centre (KGC): KGC is a partially trusted entity responsible for producing partial private keys for each EV.

Local Aggregator (LA): LA is equivalent to a local gateway for transmitting information. In addition, LA can aggregate a large number of messages from different CSs and send them to TA.

Charging Station (CS): CS is usually installed in public buildings, shopping malls and public parking lots. As a power purchase terminal for users, it can realize timing, watt-hour and charging, and is a bridge for the exchange of power and information between EVs and SG.

Electric Vehicle (EV): Only EV registered with TA can obtain services from SG by interacting with CS.

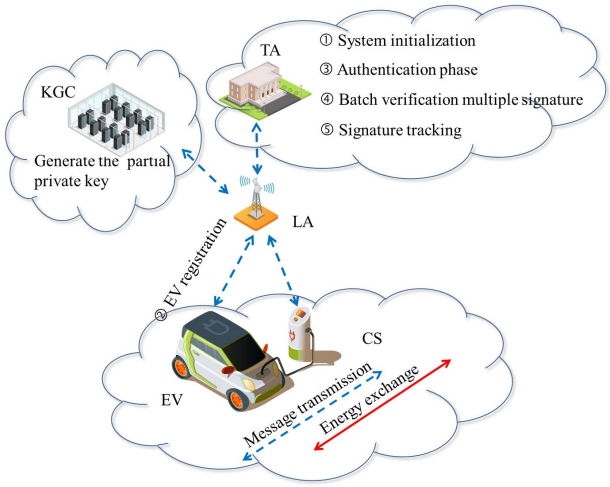


Fig. 3. System model.

B. Adversary Model

In our proposed scheme, there are two types of adversary with distinct capabilities are considered.

External Adversary ($\mathcal{A}_I$): $\mathcal{A}_I$ can launch a public key replacement attacks, that is, he/she can choose a value to replace any EV's public key even if he/she does not have the system master secret key and the partial private key for each EV.

Internal Adversary ($\mathcal{A}_{II}$): $\mathcal{A}_{II}$ is usually an internal entity of the system, who has knowledge of the master secret key but cannot replace the public key of any EV.

For $\mathcal{A}_I$ and $\mathcal{A}_{II}$, we use $Game\ I$ and $Game\ II$ to define the unforgeability of signatures in the proposed scheme, respectively.

The $Game\ I$ is played between a polynomial-time algorithm $\mathcal{B}$ and the adversary $\mathcal{A}_I$. $\mathcal{B}$ is a challenger algorithm that can simulate $\mathcal{A}_I$'s environment. The interaction between $\mathcal{B}$ and $\mathcal{A}_I$ is as follows:

Initialization: Challenger $\mathcal{B}$ runs the Setup algorithm to produce the master secret key and system public parameters $params$. Then, $\mathcal{B}$ keeps the master key secretly, and the $params$ are sent to $\mathcal{A}_I$.

Oracle Simulation: $\mathcal{A}_I$ adaptively issues the following queries to $\mathcal{B}$ and can obtain $\mathcal{B}$'s replies.

1) **Partial-Private-Key-Extract Query**. If $\mathcal{A}_I$ has issued a query on ID, $\mathcal{B}$ returns the partial private key D_{ID} to $\mathcal{A}_I$.
2) **Secret-Value Query**. If $\mathcal{A}_I$ has issued a query on the secret value of ID, $\mathcal{B}$ returns the secret value x_{ID} to $\mathcal{A}_I$.
3) **Key-Replace Query**. If $\mathcal{A}_I$ has submitted a query about (ID, x'_{ID}, U'_{ID}) and $x'_{ID}P = U'_{ID}$, x'_{ID} is a secret value and $\mathcal{B}$ updates (ID, x'_{ID}, U'_{ID}).

4) **Sign Query**. If $\mathcal{A}_I$ has issued a query for the signature on message m, $\mathcal{B}$ returns the signature σ to $\mathcal{A}_I$.

Forgery: $\mathcal{A}_I$ outputs a tuple $(ID^*, m^*, t^*, \sigma^*, PK_{ID^*})$, which includes identity, message, timestamp, signature and public key. If the following three conditions are satisfied, $\mathcal{A}_I$ wins the game.

1) $\mathcal{A}_I$ has never issued a Partial-Private-Key-Extract Query on ID^*.
2) The signature on (ID^*, m^*, t^*) has never been queried by $\mathcal{A}_I$.
3) Verify $(params, ID^*, m^*, U_{ID^*}, \sigma^*) = 1$.

The $Game\ II$ is played between a polynomial-time algorithm $\mathcal{B}$ and the adversary $\mathcal{A}_{II}$, and the steps are as follows:

Initialization: Challenger $\mathcal{B}$ runs the Setup algorithm to produce the master secret key and system public parameters $params$. Then, $\mathcal{B}$ keeps the master key secretly, and the $params$ are sent to $\mathcal{A}_{II}$.

Oracle Simulation: $\mathcal{A}_{II}$ adaptively issues the following queries to $\mathcal{B}$ and can obtain $\mathcal{B}$'s replies.

1) **Partial-Private-Key-Extract Query**. If $\mathcal{A}_{II}$ has issued a query on ID, $\mathcal{B}$ returns the partial private key D_{ID} to $\mathcal{A}_{II}$.
2) **Secret-Value-Extract Query**. If $\mathcal{A}_{II}$ has issued a query on the secret value of ID, $\mathcal{B}$ returns the secret value x_{ID} to $\mathcal{A}_{II}$.
3) **Sign Query**. If $\mathcal{A}_{II}$ has issued a query for the signature, $\mathcal{B}$ returns the signature σ to $\mathcal{A}_{II}$.

Forgery: $\mathcal{A}_{II}$ outputs a tuple $(ID^*, m^*, t^*, \sigma^*, PK_{ID^*})$, which includes identity, message, timestamp, signature and public key. If the following three conditions are satisfied, $\mathcal{A}_{II}$ wins the game.

1) $\mathcal{A}_{II}$ has never made the secret value extraction on ID^*.
2) $\mathcal{A}_{II}$ has never replaced public key query on ID^*.
3) The signature on (ID^*, m^*, t^*) has never been queried by $\mathcal{A}_{II}$.
4) Verify $(params, ID^*, m^*, U_{ID^*}, \sigma^*) = 1$.

C. Security Requirements

The proposed scheme should meet the following security requirements.

Message unforgeability: Messages transmitted between system entities cannot be forged by malicious third parties.

Anonymity: EV's real identity will not be disclosed to CS or other EVs when it interacts with SG.

Authentication and integrity: EV or CS can determine whether the transmission message comes from a valid entity by verifying the signature, thus determining the integrity of the message.

Unlinkability: CS cannot determine whether two messages are sent by the same EV.

Traceability: If an EV has malicious behavior, TA can trace its real identity from the pseudonym of the EV.

Replay attack resistance: A malicious EV cannot repeatedly send previously sent messages to deceive the CS.

Signature traceability: TA has the ability to quickly track EVs with illegal signatures when verifying signatures in batches.

V. Proposed Scheme

In this section, we present a new lightweight CLS scheme and use it to construct a traceable and privacy-preserving authentication scheme for energy trading in V2G networks. For ease of understanding, we give the main notations in the proposed scheme and their descriptions, as shown in Table I.

TABLE I
Main Notations and Descriptions

Notation	Description
κ	Security parameter
$\mathbb{G}_1$, $\mathbb{G}_2$	Two cyclic groups of the same prime order q
P	The generator of $\mathbb{G}_1$
s	Master secret key
P_{pub}	Master public key
EV_i	The ith electric vehicle
ID_{EV_i}	Real identity of EV_i
PID_{EV_i}	Pseudonym identity of EV_i
$D_{PID_{EV_i}}$	Partial private key of EV_i
$PK_{PID_{EV_i}}$	Public key of EV_i
x_{ID_i}	Secret value of EV_i
m_i	Message of EV_i
σ_i	Signature of m_i
σ	An aggregate signature
t_i, t_i'	Current timestamp
h_1^i	The hash value H_1 in σ_i
h_2^i	The hash value H_2 in σ_i

A. The Proposed CLS Scheme

First, we propose a lightweight CLS scheme for V2G networks, which consists of the following six algorithms.

Algorithm 1: Setup

Input: A security parameter κ.
Output:
1: P is the generator of $\mathbb{G}_1$, the bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$;
2: Randomly selects $s \in \mathbb{Z}_q$, computes $P_{pub} = sP$;
3: There are three hash functions $H_0, H_1, H_2 : \{0,1\} \to \mathbb{G}_1$;
4: **return** the master secret key s and the parameters $\mathsf{params} = \{q, \mathbb{G}_1, \mathbb{G}_2, P, P_{pub}, e, H_0, H_1, H_2\}$.

Algorithm 2: Partial-Private-Key-Extract

Input: The user's $ID \in \{0,1\}^*$.
Output:
1: Computes $Q_{ID} = H_0(ID)$ and $D_{ID} = sQ_{ID}$;
2: **return** the user's partial private key D_{ID}.

Algorithm 3: Set-Secret-Value

Input: A random number $x_{ID} \in \mathbb{Z}_q$.
Output:
1: **return** the secret value x_{ID}.

Algorithm 4: Set-Public-Key

Input: x_{ID}.
Output:
1: Computes $U_{ID} = x_{ID}P$;
2: **return** the public key $PK_{ID} = (U_{ID}, Q_{ID})$.

Algorithm 5: Sign

Input: The message m and the random number x_{ID}.
Output:
1: Randomly chooses $r \in \mathbb{Z}_q$, computes $R = rP$;
2: Computes $h_1 = H_1(ID, m, PK_{ID}, t), h_2 = H_2(ID, m, R, t)$;
3: Computes $\tau = D_{ID} + x_{ID}h_1 + rh_2$;
4: **return** the signature $\sigma = (\tau, R)$.

Algorithm 6: Verify

Input: The message m and the identity ID.
Output:
1: Computes $h_1 = H_1(ID, m, PK_{ID}, t), h_2 = H_2(ID, m, R, t)$;
2: Checks the equation: $e(\tau, P) = e(Q_{ID}, P_{pub})e(h_1, U_{ID})e(h_2, R)$;
3: **return** 1 if the equation holds, otherwise outputs 0.

If the signature is valid, then we have

$$\begin{aligned} e(\tau, P) &= e(D_{ID} + x_{ID}h_1 + rh_2, P) \\ &= e(D_{ID}, P)e(x_{ID}h_1, P)e(rh_2, P) \\ &= e(Q_{ID}, P_{pub})e(h_1, U_{ID})e(h_2, R) \end{aligned}$$

Therefore, the proposed CLS scheme satisfies the correctness.

B. Specific Scheme

Based on the proposed CLS scheme, we construct a traceable and privacy-preserving authentication scheme for energy trading in V2G networks. The specific scheme includes five parts: system initialization, EV registration, authentication phase, batch verification of multiple signatures and signature tracking.

1) System Initialization: In the proposed scheme, TA initials the system. Given a security parameter κ, TA runs **Algorithm** 1 to generate the master secret key s, the master public key P_{pub} and chooses four hash functions $H_0, H_1, H_2 : \{0,1\} \rightarrow \mathbb{G}_1$, $H_3^{\Lambda} : \{0,1\}^* \rightarrow \{0,1\}^l$, where H_3^{Λ} is a key hash with the key space of $\{0,1\}^*$. Then TA makes $params = \{q, \mathbb{G}_1, \mathbb{G}_2, P, P_{pub}, e, H_0, H_1, H_2, H_3^{\Lambda}\}$ public.

2) EV Registration: EVs must be registered in TA before they can get the services provided by the SG. Meanwhile, TA computes the pseudonym identity for each registered EV_i to protect their real identity, where EV_i represents the ith EV.

- EV_i submits the real identity ID_{EV_i}, including licence plate number, user name, address, telephone number, etc., to TA through the secure channel.
- TA chooses $\lambda_i \in \Lambda$ as a hash key, and computes a pseudonym identity $PID_{EV_i} = H_3^{\lambda_i}(ID_{EV_i}||VT_i)$ for EV_i [2], where VT_i is a valid time interval of pseudonym identity.
- Then, $\{PID_{EV_i}, ID_{EV_i}\}$ is stored in a tamper-proof device (TPD) of EV_i and TA's member list [2].
- According to **Algorithm** 3, EV_i selects a random number $x_{ID_i} \in \mathbb{Z}_q$ as his/her secret value.

3) Authentication Phase: Once participating in the charging and discharging service, EV_i first generates a signature for the sent message $m_i \in \{0,1\}^*$ before sending to CS, which can ensure message integrity and authentication.

- KGC generates the partial private key $D_{PID_{EV_i}} = sH_0(PID_{EV_i}) = sQ_{PID_{EV_i}}$ by using the **Algorithm** 2 and EV_i's pseudonym identity PID_{EV_i}, then it sends the partial private key to the corresponding EV_i.
- Using **Algorithm** 4, EV_i calculates its public key as $PK_{PID_{EV_i}} = (x_{ID_i}P, H_0(PID_{EV_i})) = (U_{PID_{EV_i}}, Q_{PID_{EV_i}})$. Then EV_i chooses random number $r_i \in \mathbb{Z}_q$ and computes $R_i = r_iP$.
- EV_i calls the **Algorithm** 5 to generate a signature σ_i (τ_i, R_i) of message m_i, where $\tau_i = D_{PID_{EV_i}} + x_{ID_i}h_1^i + r_ih_2^i$, $h_1^i = H_1(PID_{EV_i}, m_i, PK_{PID_{EV_i}}, t_i)$ and $h_2^i = H_2(PID_{EV_i}, m_i, R_i, t_i)$, t_i is the current timestamp. Next, EV_i sends the message m_i together with the signature σ_i to TA through CS.
- Upon receiving the information from EV_i, TA first verifies the freshness of t_i and discards m_i if it expires. Then, TA invokes **Algorithm** 6 to verify the validity of the signature. If the signature is invalid, TA rejects this signature σ_i. Otherwise, it accepts σ_i.

If the equation $e(\tau_i, P) = e(Q_{PID_{EV_i}}, P_{pub})e(h_1^i, U_{PID_{EV_i}})e(h_2^i, R_i)$ holds, the EV_i complete the authentication.

The correctness of the single verification is

$$\begin{aligned} e(\tau_i, P) &= e(D_{PID_{EV_i}} + x_{ID_i}h_1^i + r_ih_2^i, P) \\ &= e(D_{PID_{EV_i}}, P)e(x_{ID_i}h_1^i, P)e(r_ih_2^i, P) \\ &= e(Q_{PID_{EV_i}}, sP)e(h_1^i, x_{ID_i}P)e(h_2^i, r_iP) \\ &= e(Q_{PID_{EV_i}}, P_{pub})e(h_1^i, U_{PID_{EV_i}})e(h_2^i, R_i). \end{aligned}$$

4) Batch Verification of Multiple Signatures: In practice, LA will receive request messages from local multiple EVs during the same period. After receiving message signature pairs (m_i, σ_i) from different EV_i, LA aggregates all signatures into σ. Then LA sends $(\sigma||t_i')$ to TA, where t_i' is the current timestamp. After receiving $(\sigma||t_i')$, TA first verifies the freshness of t_i' and discards $(m_1 \cdots m_n)$ if it expires. TA then checks whether the following authentication equation is true:

$$e(P, \tau) = e(P_{pub}, \sum_{i=1}^{n} Q_{PID_{EV_i}})e(U_{PID_{EV_i}}, \sum_{i=1}^{n} h_1^i)e(R, \sum_{i=1}^{n} h_2^i).$$

If the equation holds, then the batch verification of multiple signatures is passed and these request messages are accepted.

The correctness of the batch verification is

$$
\begin{aligned}
e(P,\tau) &= e(P,\sum_{i=1}^{n}\tau_i) = e(P,\sum_{i=1}^{n}(D_{PID_{EV_i}} + x_{ID_i}h_1^i + r_i h_2^i)) \\
&= e(P,\sum_{i=1}^{n} D_{PID_{EV_i}})e(P,\sum_{i=1}^{n} x_{ID_i}h_1^i)e(P,\sum_{i=1}^{n} r_i h_2^i) \\
&= e(sP,\sum_{i=1}^{n} Q_{PID_{EV_i}})e(x_{ID_i}P,\sum_{i=1}^{n} h_1^i)e(r_i P,\sum_{i=1}^{n} h_2^i) \\
&= e(P_{pub},\sum_{i=1}^{n} Q_{PID_{EV_i}})e(U_{PID_{EV_i}},\sum_{i=1}^{n} h_1^i)e(R,\sum_{i=1}^{n} h_2^i)
\end{aligned}
$$

5) Signature Tracking: Illegal signatures are usually encountered in aggregate signatures, which will hinder the batch verification of multiple signatures, thus reducing the efficiency of V2G networks. In order to trace the illegal signature quickly, we model the signature relationship as a hierarchy to be a full binary tree in the proposed scheme. Specifically, TA first assigns the signatures of each participating EV to the leaf nodes of a full binary tree through **Algorithm** 7, and then uses **Algorithm** 8 to build a full binary tree structure. Finally, TA can quickly find illegal signatures according to **Algorithm** 9.

Algorithm 7: Supplementing Leaf Nodes

Input: The single signature σ_i of each registered EV received by TA
Output: All the leaf nodes of a full binary tree

```
Set the height l = 1 of the full binary tree
for each single signature σ_i do
    while number of single signatures > 2^l do
        if i < num then
            The single signature σ_i is stored in the leaf nodes of the full binary tree;
        else
            Set the number of leaf nodes to 0;
        l ← l + 1
return all the leaf nodes of a full binary tree
```

Algorithm 8: Build Full Binary Tree

Input: All the leaf nodes of a full binary tree
Output: A full binary tree

```
if The number of leaf nodes in this layer is nodes.size() == 1 then
    return nodes;
else
    for (int i = 0; i < nodes.size(); i++) do
        Node parent = Node left + Node right;
    return Node parent;
return the root node σ of a full binary tree
```

Algorithm 9: Signature Tracking

Input: Aggregate signatures to be checked
Output: Illegal signatures

```
for aggregate signature in the root node σ do
    if The children of this signature are NULL and the signature is inconsistent then
        return inconsistent single signatures;
    else
        if (!left.child().equals(left half)) then
            left_error = find_error(left.child);
        if (!right.child().equals(right half)) then
            right_error = find_error (right.child);
return illegal signatures
```

VI. Security Anylasis

In this section, we conduct a security analysis of the proposed scheme. Through two theorems, we prove that our scheme is existential unforgeable under adaptive chosen-message attacks in the ROM. And, we elaborate that our scheme can meet the security requirements of V2G networks.

A. Security Proof

Theorem 1. *In the ROM, if an $\mathcal{A}_I$ can forge a valid signature in probabilistic polynomial time with a non-negligible probability ϵ after q_{H_0} H_0 queries, q_{ppke} **Partial-Private-Key-Extract** queries and q_s **Sign** queries, the CDH problem can be solved by an algorithm $\mathcal{B}$ with non-negligible probability $\epsilon^{'} \geq (1-\frac{1}{q_{H_0}})^{q_{ppke}}(1-\frac{1}{q_s+1})^{q_s}\frac{1}{q_{H_0}(q_s+1)}\epsilon$.*

Proof. Given some public parameters $params = (q, \mathbb{G}_1, \mathbb{G}_2, P, e)$ and an adversary $\mathcal{A}_I$, we build an algorithm $\mathcal{B}$ to solve the CDH problem through interacting with $\mathcal{A}_I$ in polynomial-time algorithm.

Suppose aP, $bP \in \mathbb{G}_1$ are the random inputs of the CDH problem instance. Given aP, $bP \in \mathbb{G}_1$, $\mathcal{B}$ is required to output abP.

Initialization. $\mathcal{B}$ sets $P_{pub} = aP$, and then sends the public parameters $(q, \mathbb{G}_1, \mathbb{G}_2, P, P_{pub}, e, H_0, H_1, H_2)$ to $\mathcal{A}_I$. The three hash functions H_0, H_1, H_2 are viewed as random oracle.

Oracle Simulation. The following queries are issued adaptively by $\mathcal{A}_I$.

1) H_0 **Query**. $\mathcal{B}$ sets ID_I as the challenge identity and represents the identity of the i-th as ID_i, where $I \in \{1, .., q_{H_0}\}$. $\mathcal{B}$ holds a list L_0 consisting of three tuples $(ID_i, H_0(ID_i), c_i)$. If ID_i exists in L_0, $\mathcal{B}$ responds to $\mathcal{A}_I$ with $H_0(ID_i)$. Otherwise, $\mathcal{B}$ sets $H_0(ID_i)$ as

$$
H_0(ID_i) = \begin{cases} c_i P, c_i \in \mathbb{Z}_q, i \neq I \\ bP, i = I \end{cases} \tag{1}
$$

2) **Partial-Private-Key-Extract Query**. $\mathcal{B}$ holds the list L_e consisting of (ID_i, D_{ID_i}). Upon receiving a query about ID_i, $\mathcal{B}$ will check L_e. If the adversary has issued

a query on ID_i, $\mathcal{B}$ returns the partial private key D_{ID_i} to $\mathcal{A}_I$. Otherwise, $\mathcal{B}$ sets the partial private key as

$$D_{ID_i} = \begin{cases} c_i aP, i \neq I \\ \perp, i = I \end{cases} \tag{2}$$

$\mathcal{B}$ returns to this query with D_{ID_i}, and adds (ID_i, D_{ID_i}) into L_e.

3) **Secret-Value Query**. If the adversary $\mathcal{A}_I$ has issued a query on the secret value of ID_i, $\mathcal{B}$ returns the secret value x_{ID_i} to $\mathcal{A}_I$.
4) **Key-Replace Query**. On receiving a new private/public key pair (x'_{ID_i}, U'_{ID_i}) on the identity ID_i, $\mathcal{B}$ checks the equation $x'_{ID_i}P = U'_{ID_i}$. If the equation holds, then $\mathcal{B}$ updates $\left(ID_i, x'_{ID_i}, U'_{ID_i}\right)$.
5) H_1 **Query**. $\mathcal{B}$ holds a list L_1 consisting of tuples $(ID_i, m_i, t_i, PK_{ID_i}, H_1(ID_i, m_i, t_i, PK_{ID_i}), d_i)$. If (ID_i, m_i, PK_{ID_i}) exists in L_1, then $\mathcal{B}$ responds to $\mathcal{A}_I$ with $H_1(ID_i, m_i, t_i, PK_{ID_i}), d_i)$. Otherwise, $\mathcal{B}$ randomly chooses $d_i \in \mathbb{Z}_q$, computes and returns $H_1(ID_i, m_i, t_i, PK_{ID_i}) = d_i P$. $\mathcal{B}$ responses to this query with $H_1(ID_i, m, t, PK_{ID_i})$ and adds $(ID_i, m_i, t_i, PK_{ID_i}, H_1(ID_i, m_i, PK_{ID_i}), d_i)$ into L_1.
6) H_2 **Query**. $\mathcal{B}$ maintains a list L_2 consisting of $(ID_i, m_i, t_i, R_i, H_2(ID_i, m_i, R_i), y_i, c)$. The list is initially empty. If (ID_i, m_i, t_i, R_i) is in the list L_2, $\mathcal{B}$ sends $H_2(ID_i, m_i, t_i, R_i)$ to $\mathcal{A}_I$. Otherwise, $\mathcal{B}$ set $H_2(ID_i, m_i, t_i, R_i)$ as

$$H_2(ID_i, m_i, t_i, R_i) = \begin{cases} y_i P, y_i \in \mathbb{Z}_q, & i \neq I \\ bP, & i = I, c = 0 \\ y_i P, & i = I, c = 1 \end{cases} \tag{3}$$

Let the probability of $c = 1$ is ζ, i.e., the probability of head shows up when $\mathcal{B}$ throws a biased coin with two sides, and the probability of $c = 0$ (the tail shows up) is $1 - \zeta$. $\mathcal{B}$ responses to this query with $H_2(ID_i, m_i, t_i, R_i)$ and adds $(ID_i, m_i, t_i, R_i, H_2(ID_i, m_i, t_i, R_i), y_i, c)$ into L_2.

7) **Sign Query**. For the signature query on (ID_i, m_i, t_i), $\mathcal{B}$ randomly chooses $r_i \in \mathbb{Z}_q$ and lets $R_i = r_i P - aP$. $\mathcal{B}$ computes the signature

$$\tau_i = \begin{cases} c_i aP + d_i U_{ID_i} + y_i r_i P - y_i aP, & i \neq I \\ d_i U_{ID_I} + r_i bP, & i = I, c = 0 \\ \perp, & i = I, c = 1 \end{cases} \tag{4}$$

$\mathcal{B}$ sends the signature $\sigma_i = (\tau_i, R_i)$ to $\mathcal{A}_I$. Note that U_{ID^*} is the current public key.

Forgery. $\mathcal{A}_I$ forges a tuple $(ID_i{}^*, m_i{}^*, t_i{}^*, \sigma_i^*, R_i^*)$. If the following three conditions are satisfied, $\mathcal{A}_I$ wins the game.

1) $\mathcal{A}_I$ has never issued a Partial-Private-Key-Extract Query on $ID_i{}^*$.
2) The signature on (ID_i^*, m_i^*, t_i^*) has never queried by $\mathcal{A}_I$.
3) Verify $(params, ID_i^*, m_i^*, U_{ID_i^*}, \sigma_i^*) = 1$.

If $ID^* \neq ID_I$, abort. Otherwise, $\mathcal{B}$ iterates over L_2, if $c = 0$, abort. Otherwise, $c = 1$, we have

$$\begin{aligned} \tau_i{}^* &= sQ_{ID_i^*} + x_{ID_i} H_1(ID_i^*, m_i^*, t_i^*, PK_{ID_i^*}) \\ &\quad + r_i^* H_2(ID_i^*, m_i^*, t_i^*, R_i^*) \\ &= abP + d_i^* x_{ID_i^*} P + y_i^* r_i^* P \\ &= abP + d_i^* U_{ID^*} + y_i^* R_i^* \\ &\Longrightarrow abP = \tau_i^* - d_i^* U_{ID_i^*} - y_i^* R_i^* \end{aligned} \tag{5}$$

$\mathcal{B}$ will output the solution of the CDH problem, i.e., $\tau_i * -d_i U_{ID^*} - y_i^* R_i^*$, where $U_{ID_i^*}$ is the current public key. □

Probability analysis: If $\mathcal{B}$ can solve the CDH problem, then the following three conditions must be met.

1) C_1: $\mathcal{B}$ never stops the game.
2) C_2: A_I forges a valid signature.
3) C_3: $ID_i^* = ID_i, c = 1$.

Therefore, the probability of $\mathcal{B}$ successfully solving difficult problem is $\epsilon^{'}$ = Pr$[C_1 \wedge C_2 \wedge C_3]$ = Pr$[C_1]$ · Pr$[C_2|C_1]$ · Pr$[C_3|C_1 \wedge C_2]$. It is easy to see that the probabilities of non-termination of the game in H_0 **query** and **Sign query** are $(1-\frac{1}{q_{H_0}})^{q_{ppke}}$ and $(1-\frac{1}{q_{H_0}}\zeta)^{q_s}$, respectively. The probabilities of Pr$[C_2|C_1]$ and Pr$[C_3|C_1 \wedge C_2]$ are ϵ and $\frac{1}{q_{H_0}}\zeta$, respectively. To sum up, $\epsilon^{'} \geq (1 - \frac{1}{q_{H_0}})^{q_{ppke}}(1-\zeta)^{q_s}\frac{1}{q_{H_0}}\zeta\epsilon$. Because when $\zeta = \frac{1}{q_s+1}$, $(1-\zeta)^{q_s}\zeta$ can obtain the maximum value. Therefore, $\epsilon^{'} \geq (1 - \frac{1}{q_{H_0}})^{q_{ppke}}(1 - \frac{1}{q_s+1})^{q_s}\frac{1}{q_{H_0}(q_s+1)}\epsilon$.

Theorem 2. *In the ROM, if an $\mathcal{A}_{II}$ can forge a valid signature in probabilistic polynomial time with a non-negligible probability ϵ after q_{H_0} H_0 queries, q_{sv} **Secret-Value** queries, q_s **Sign** queries, the CDH problem can be solved by an algorithm $\mathcal{B}$ with non-negligible probability $\epsilon^{'} \geq (1 - \frac{1}{q_{H_0}})^{q_{sv}}(1 - \frac{1}{q_s+1})^{q_s}\frac{1}{q_{H_0}(q_s+1)}\epsilon$.*

Proof. Given some public parameters $pp = (q, \mathbb{G}_1, \mathbb{G}_2, P, e)$ and an adversary $\mathcal{A}_{II}$, we build an algorithm $\mathcal{B}$ to solve the CDH problem through interacting with $\mathcal{A}_{II}$ in polynomial-time algorithm.

Suppose aP, $bP \in \mathbb{G}_1$ are the random inputs of the CDH problem instance. Given aP, $bP \in \mathbb{G}_1$, $\mathcal{B}$ is required to output abP.

Initialization. $\mathcal{B}$ randomly choose $s \in \mathbb{Z}_q$ and computes $P_{pub} = sP$. Then the public parameters $(q, \mathbb{G}_1, \mathbb{G}_2, P, P_{pub}, e, H_0, H_1, H_2)$ are sent to $\mathcal{A}_{II}$. The three hash functions H_0, H_1, H_2 are viewed as random oracle.

Oracle Simulation. The following queries are issued adaptively by $\mathcal{A}_{II}$.

1) H_0 **Query**. $\mathcal{B}$ represents the i-th identity as ID_i, and sets ID_I as the challenge identity, where $I \in \{1, .., q_{H_0}\}$. $\mathcal{B}$ holds a list L_0 consisting of three tuples $(ID_i, H_0(ID_i), d_i)$. If ID_i exists in L_0, $\mathcal{B}$ responds to $\mathcal{A}_{II}$ with $H_0(ID_i)$. Otherwise, $\mathcal{B}$ randomly chooses $c_i \in \mathbb{Z}_q$ and computes $H_0(ID_i) = c_i P$. $\mathcal{B}$ adds $(ID_i, H_0(ID_i), c_i)$ into L_0.
2) **Partial-Private-Key-Extract Query**. $\mathcal{B}$ holds list L_e consisting of (ID_i, D_{ID_i}). Upon receiving a query about ID_i, $\mathcal{B}$ will check L_e. If the adversary has issued

a query on ID_i, $\mathcal{B}$ returns the partial private key. Otherwise, $\mathcal{B}$ computes the partial private key $D_{ID_i} = sc_iP$. $\mathcal{B}$ returns to this query with D_{ID_i} and adds (ID_i, D_{ID_i}) into L_e.

3) **Secret-Value Query**. If $\mathcal{A}_I$ makes a query on the secret value of ID_i, $\mathcal{B}$ returns the secret value x_{ID_i} when $i \neq I$, otherwise $\perp$.
4) H_1 **Query**. $\mathcal{B}$ holds a list L_1 consisting of tuples $(ID_i, m_i, t_i, PK_{ID_i}, H_1(ID_i, m_i, t_i, PK_{ID_i}), d_i)$. If $(ID_i, m_i, t_i, PK_{ID_i})$ exists in L_1, then $\mathcal{B}$ responds to $\mathcal{A}_{II}$ with $H_1(ID_i, m_i, t_i, PK_{ID_i}))$.

$$H_1(ID_i, m_i, t_i, V) = \begin{cases} d_iP, d_i \in \mathbb{Z}_q, & i \neq I \\ d_ibP, & i = I \end{cases} \quad (6)$$

$\mathcal{B}$ responses to this query with $H_1(ID_i, m_i, t_i, PK_{ID_i})$ and adds $(ID_i, m_i, PK_{ID_i}, H_1(ID_i, m_i, t_i, PK_{ID_i}), d_i)$ into L_1 list.

5) H_2 **Query**. $\mathcal{B}$ maintains a list L_2 consisting of $(ID_i, m_i, t_i, R_i, H_2(ID_i, m_i, t_i, R_i), y_i, c)$. The list is initially empty. If (ID_i, m_i, t_i, R_i) is in the list L_2, $\mathcal{B}$ sends $H_2(ID_i, m_i, t_i, R_i)$ to $\mathcal{A}_{II}$. Otherwise, $\mathcal{B}$ sets $H_2(ID_i, m_i, t_i, R_i)$ as

$$H_2(ID_i, m_i, t_i, R_i) = \begin{cases} y_iP, y_i \in \mathbb{Z}_q, & i \neq I \\ d_iaP, & i = I, c = 0 \\ y_iP, & i = I, c = 1 \end{cases} \quad (7)$$

Let the probability of $c = 1$ is ζ, i.e., the probability of head shows up when $\mathcal{B}$ throws a biased coin, and the probability of $c = 0$ (the tail shows up) is $1 - \zeta$. $\mathcal{B}$ responses to this query with $H_2(ID_i, m_i, t_i, R_i)$ and adds $(ID_i, m_i, t_i, R_i, H_2(ID_i, m_i, t_i, R_i), y_i, c)$ into L_2.

6) **Sign Query**. For a signature query on (ID_i, m_i, t_i), $\mathcal{B}$ randomly chooses $r_i \in \mathbb{Z}_q$ and lets $R_i = r_iP - bP$. $\mathcal{B}$ computes the signature

$$\tau_i = \begin{cases} sc_iP + x_{ID_i}d_iP + y_ir_iP - y_ibP, & i \neq I \\ sc_iP + d_ir_iaP, & i = I, c = 0 \\ \perp, & i = I, c = 1 \end{cases} \quad (8)$$

$\mathcal{B}$ responds to $\mathcal{A}_{II}$ with σ_i= (τ_i, R_i).

Forgery. $\mathcal{A}_{II}$ outputs a tuple $(ID_i{}^*, m_i{}^*, t_i{}^*, \sigma_i^*, R_i^*)$. If the following two conditions are satisfied, $\mathcal{A}_{II}$ wins the game.

1) $\mathcal{A}_{II}$ has never made the secret value extraction on ID_i^*.
2) $\mathcal{A}_{II}$ has never replaced public key query on ID_i^*.
3) The signature on (ID_i^*, m_i^*, t_i^*) has never queried by $\mathcal{A}_{II}$.
4) Verify $(params, ID_i^*, m_i^*, t_i^*, U_{ID_i^*}, \sigma_i^*)$ = 1.
If $ID^* \neq ID_I$, abort. Otherwise, $\mathcal{B}$ iterates over L_2 list, if $c = 0$, abort. Otherwise, $c = 1$, we have

$$\begin{aligned} \tau_i{}^* &= sQ_{ID_i^*} + x_{ID_i}H_1(ID_i^*, m_i^*, t_i^*, PK_{ID_i^*}) \\ &\quad + r_i^*H_2(ID_i^*, m_i^*, t_i^*, R_i^*) \\ &= sc_i^*P + d^*abP + y_i^*r_i^*P \\ &= sc_i^*P + d^*abP + y_i^*R_i^* \\ &\Longrightarrow abP = (d^*)^{-1}(\tau_i^* - sc_i^*P - y_i^*R_i^*) \end{aligned} \quad (9)$$

$\mathcal{B}$ will output the solution of the CDH problem, i.e., $(d^*)^{-1}(\tau_i^* - sc^*P - y_i^*R_i^*)$. □

Probability analysis: If $\mathcal{B}$ can solve the CDH problem, then the following three conditions must be met.

1) C_1: $\mathcal{B}$ never stops the game.
2) C_2: A_I forges a valid signature.
3) C_3: $ID_i^* = ID_i, c = 1$.

Therefore, the probability of $\mathcal{B}$ successfully solving difficult problem is $\epsilon^{'}$ = Pr$[C_1 \wedge C_2 \wedge C_3]$ = Pr$[C_1]$ · Pr$[C_2|C_1]$ · Pr$[C_3|C_1 \wedge C_2]$. It is easy to see that the probabilities of non-termination of the game in **Secret-Value query** and **Sign query** are $(1 - \frac{1}{q_{H_0}})^{q_{sv}}$ and $(1 - \frac{1}{q_{H_0}}\zeta)^{q_s}$, respectively. The probabilities of Pr$[C_2|C_1]$ and Pr$[C_3|C_1 \wedge C_2]$ are ϵ and $\frac{1}{q_{H_0}}\zeta$, respectively. To sum up, $\epsilon^{'} \geq (1 - \frac{1}{q_{H_0}})^{q_{sv}}(1 - \zeta)^{q_s}\frac{1}{q_{H_0}}\zeta\epsilon$. Because when $\zeta = \frac{1}{q_s+1}$, $(1-\zeta)^{q_s}\zeta$ can obtain the maximum value. Therefore, $\epsilon^{'} \geq (1 - \frac{1}{q_{H_0}})^{q_{sv}}(1 - \frac{1}{q_s+1})^{q_s}\frac{1}{q_{H_0}(q_s+1)}\epsilon$.

Based on the results obtained, our scheme is existential unforgeable under adaptive chosen-message attacks in the ROM.

B. Analysis of Other Security Requirements

Next, we analyze how the proposed scheme meets the following security requirements.

Anonymity: When an EV is registered, TA calculates a pseudonym identity PID_{EV_i} for this EV. Only TA can know the real identity in the proposed scheme. During the interaction between EV and SG, the pseudonym identity PID_{EV_i} is used to protect EV's real identity information ID_{EV_i}.

Authentication and integrity: To ensure message authentication and integrity, each message is signed by the registered EV before sending to CS, and the receiver can check the validity of the message signature. The formal proof that the signature can not be maliciously forged or modified has been given in Section VI-A.

Unlinkability: In our scheme, the pseudonym identity PID_{EV_i} of EV is generated by calculating $PID_{EV_i} = H_3^{\Lambda}(ID_{EV_i}||VT_i)$, where H_3^{Λ} is a keyed hash. Since H_3^{Λ} is computationally indistinguishable and one-way, no one can connect any two pseudonym identity to the identity of EV.

Traceability: Because the TA has the hash key $\lambda_i \in \Lambda$, only TA can acquire an corresponding EV's real identity according to its pseudonym identity $PID_{EV_i} = H_3^{\lambda_i}(ID_{EV_i}||VT_i)$.

Replay attack resistance: We consider the timestamp t_i in the signature generation algorithm. Therefore, the receiver can verify the freshness of message according to the timestamp t_i, which effectively resists the attack of malicious EVs replaying a signed message.

Signature traceability: We map the signatures of all registered EVs to a full binary tree. Hence, TA has the ability to quickly track the EVs with illegal signatures when verifying signatures in batches.

VII. Experimental Results

In this section, we compare the proposed scheme with the state-of-the art schemes in terms of security features,

computational cost and communication overhead. We use relic cryptographic meta-toolkit [42] to implement our proposed scheme on a personal computer (with an i7 12700 2.3 GHz core, 16GB RAM, Ubuntu 18.04 operating system), the curve we used is BN curve, which can achieve 80 bits security. The various operation time is averaged after multiple tests, and the results are shown in Table II. Because the operation time of point addition and hash function is relatively small from T_{bp}, T_{mtp} and T_{mul}, we ignore them in the whole performance evaluation.

TABLE II
NOTATION, DESCRIPTION AND OPERATION TIME

Notation	Description	Operation time
T_{bp}	a bilinear pairing operation time	3.016 ms
T_{mtp}	a map-to-point hash operation time	1.607 ms
T_{mul}	a point multiplication operation time	0.286 ms

A. Comparison of Security Features

The comparison results of security features are listed in Table III. Security features mainly include anonymity, authentication, unlinkability, traceability, the ability to resist $\mathcal{A}_I$ (Resist $\mathcal{A}_I$), the ability to resist $\mathcal{A}_{II}$ (Resist $\mathcal{A}_{II}$), replay attack resistance (RAT) and signature tracking (ST). Here, "$\surd$" indicates that a scheme meets the corresponding security feature, and "$\times$" indicates that a scheme does not meet the corresponding security feature. From Table III, [18], [17] and [15] can satisfy the security features except ST. [13] and [14] satisfy the same security features. [16] only satisfies the security features of authentication and resist $\mathcal{A}_I$. However, compared with the above schemes, our scheme can meet all the security features.

B. Comparison of Computational Cost

In this section, we compare the computational cost of our scheme with the existing bilinear pairing-based schemes [18], [17], [13], [14] and [15]. For consistency, a bilinear pairing e: $\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ is used to achieve the 80 bits security level. $\mathbb{G}_1$ is the additive cyclic group with prime order $q > 2^{\kappa}$ realized on a supersingular elliptic curve equation $y^2 = (x^3 + x) mod p$, where the values of prime p, q are 512 bits and 160 bits, respectively. Therefore, according to the method of scheme [2], the length of a element in $\mathbb{G}_1$ is 512 bits $\div$ 8 $\times$ 2 = 128 bytes. Table IV shows the theoretical analysis results of the computational cost. According to the results in Table IV, we draw the computational cost figure of signature generation and single signature verification as shown in Fig.4, and the computational cost figure of aggregate verification as shown in Fig. 5.

From Fig. 4, it is not difficult to see that our scheme does not have the best computational cost in signature generation and single signature verification among these comparison schemes. However, the aggregate verification efficiency of our scheme increases with the increase of the number of EVs, as shown in Fig. 5. From Fig. 5, we see that when the number of participating EVs n is greater than 20, the aggregate verification cost of our proposed scheme is always the lowest compared with all the related schemes. When $n = 80$, the cost of aggregate verification of our proposed scheme is about 38 ms, while that of [18], [17], [15] is almost close to 60 ms, that of [13] and [14] is about 80 ms, and that of [16] exceeds 180 ms.

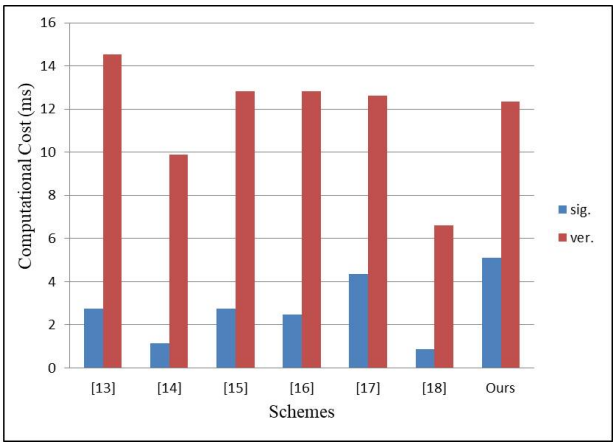


Fig. 4. Computational cost of signature and verification

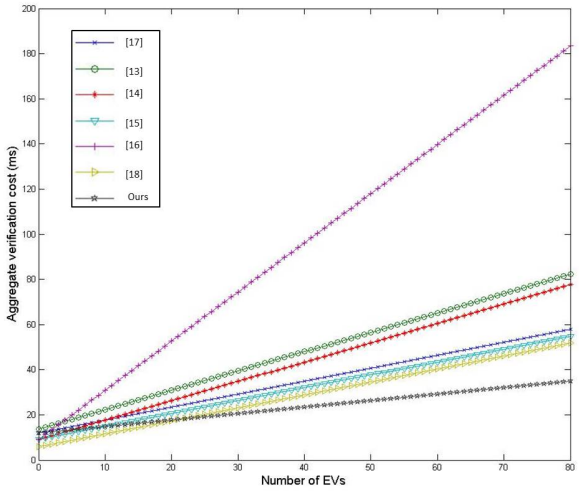


Fig. 5. Computational cost of aggregate verification

C. Comparison of Communication Overhead

In this section, we analyze and compare our scheme with the schemes mentioned above in terms of communication overhead. As mentioned in the previous section, the length of $\mathbb{G}_1$ is 128 bytes. In general, the length of a hash function is assumed to be 20 bytes and the length of a timestamp is assumed to be 4 bytes. For convenience, we only consider the length of signature and timestamp in the transmitted message. For example, the message signature generated by EV in our scheme is (σ_i, m_i), where $m_i \in \mathbb{G}_1$. Therefore, the length of a message sent from one EV to LA is $1|\mathbb{G}_1| + |t_i| = 1 \times 128 + 4 = 132$ bytes, and the length of the message sent simultaneously from

TABLE III
COMPARISON OF SECURITY FEATURES

Schemes	Anonymity	Authentication	Unlinkability	Traceability	Resist $\mathcal{A}_I$	Resist $\mathcal{A}_{II}$	RAT	ST
Yang *et al.*'s [13]	√	√	×	×	√	√	×	×
Kumar *et al.*'s [14]	√	√	×	×	√	√	×	×
Kamil *et al.*'s [15]	√	√	√	√	√	√	√	×
Xu *et al.*'s [16]	×	√	×	×	√	×	×	×
Mei *et al.*'s [17]	√	√	√	√	√	√	√	×
Wang *et al.*'s [18]	√	√	√	√	√	√	√	×
Our Scheme	√	√	√	√	√	√	√	√

TABLE IV
COMPARISON OF COMPUTATIONAL COST

Schemes	Signature Generation Cost (ms)	Single Signature Verification Cost (ms)	Aggregate Verification (ms)
Yang *et al.*'s [13]	$1T_{mtp} + 4T_{mul} \approx 2.751$	$4T_{bp} + 1T_{mtp} + 3T_{mul} \approx 14.529$	$4T_{bp} + 1T_{mtp} + 3nT_{mul}$
Kumar *et al.*'s [14]	$4T_{mul} \approx 1.144$	$3T_{bp} + 3T_{mul} \approx 9.906$	$3T_{bp} + 3nT_{mul}$
Kamil *et al.*'s [15]	$1T_{mtp} + 4T_{mul} \approx 2.751$	$3T_{bp} + 2T_{mtp} + 2T_{mul} \approx 12.834$	$3T_{bp} + 2nT_{mul}$
Xu *et al.*'s [16]	$1T_{mtp} + 3T_{mul} \approx 2.465$	$3T_{bp} + 2T_{mtp} + 2T_{mul} \approx 12.834$	$3T_{bp} + nT_{mtp} + 2nT_{mul}$
Mei *et al.*'s [17]	$2T_{mtp} + 4T_{mul} \approx 4.358$	$4T_{bp} + 2T_{mul} \approx 12.636$	$4T_{bp} + 2nT_{mul}$
Wang *et al.*'s [18]	$3T_{mul} \approx 0.858$	$2T_{bp} + 2T_{mul} \approx 6.604$	$2T_{bp} + 2nT_{mul}$
Our Scheme	$3T_{mtp} + 1T_{mul} \approx 5.107$	$4T_{bp} + 1T_{mul} \approx 12.35$	$4T_{bp} + nT_{mul}$

TABLE V
COMPARISON OF COMMUNICATION OVERHEAD

Schemes	Single signature transmit (bytes)	n signature transmit (bytes, n=100)
Yang *et al.*'s [13]	$2\|\mathbb{G}_1\| = 256$	$(n+1)\|\mathbb{G}_1\| = 128 + 128n = 12928$
Kumar *et al.*'s [14]	$2\|\mathbb{G}_1\| = 256$	$(n+1)\|\mathbb{G}_1\| = 128 + 128n = 12928$
Kamil *et al.*'s [15]	$2\|\mathbb{G}_1\| + \|t_i\| = 260$	$2\|\mathbb{G}_1\| + n\|t_i\| = 256 + 4n = 656$
Xu *et al.*'s [16]	$2\|\mathbb{G}_1\| + \|t_i\| = 260$	$(n+1)\|\mathbb{G}_1\| + n\|t_i\| = 128 + 132n = 13328$
Mei *et al.*'s [17]	$2\|\mathbb{G}_1\| + \|t_i\| = 260$	$2\|\mathbb{G}_1\| + n\|t_i\| = 256 + 4n = 656$
Wang *et al.*'s [18]	$3\|\mathbb{G}_1\| + \|t_i\| = 388$	$3\|\mathbb{G}_1\| + n\|t_i\| = 384 + 4n = 784$
Our Scheme	$1\|\mathbb{G}_1\| + \|t_i\| = 132$	$1\|\mathbb{G}_1\| + n\|t_i\| = 128 + 4n = 528$

multiple EVs to LA is $1|\mathbb{G}_1| + n|t_i| = 128 + 4 \times n = 528$ bytes, where $n = 100$. Similarly, we calculate the communication overhead of other schemes [13]–[18], and list them in Table V. In order to see the comparison differences more clearly, we draw the communication overhead of signature transmit figure according to the results of Table V, as shown in Fig. 6. From Fig. 6, we can see that a single signature transmission cost of our proposed scheme is 132 bytes, which is about 50% of that of other comparison schemes. The cost of multi-signatures transmission is 528 bytes (set n = 100), which is about 80% of [17] and [15], 67.3% of scheme [18], and about 4% of [13], [14] and [16]. Compared with the schemes mentioned above, our scheme has the lowest communication overhead. Therefore, our scheme is easier to satisfy the communication requirements of V2G networks due to the low communication overhead.

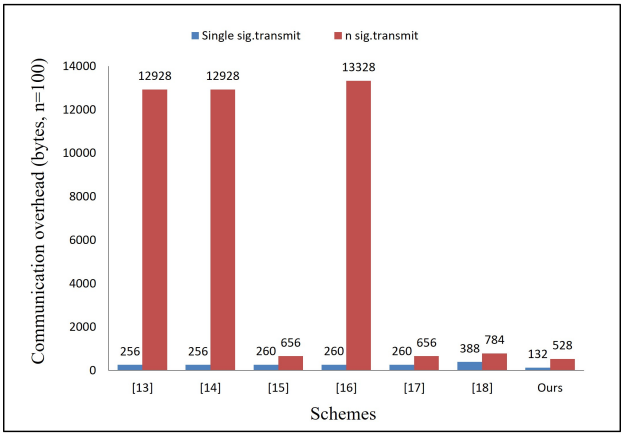


Fig. 6. Communication overhead of signature transmit

D. Efficiency Evaluation of Signature Tracking

In this section, we evaluate the performance of tracking illegal signatures in the proposed scheme. Specifically, we obtain the computational cost of tracking illegal signatures according to the increase of number of signatures and the increase of binary tree layers, respectively. From Fig. 7 (a), we set the number of signatures from 1000 to 10000, and assume that the number of illegal signatures is 1, 10, 20, 30, 40, respectively. From Fig. 7 (b), we set the the number of layers of binary tree from 11 to 15. To ensure the accuracy of results, we perform five tests for each case and calculate the average cost to eliminate the relative error. Fig. 7 (a) shows the computational cost of signature tracking under different cases. It is not difficult to see that the signature tracking cost increases with the number of signatures. The reason is that the more signatures, the more layers of constructed full binary tree. Meanwhile, the signature tracking cost is also affected by the number of illegal signatures, that is, the more illegal signatures, the greater the tracking cost. However, even if the number of signatures increases to 10000 and the number of

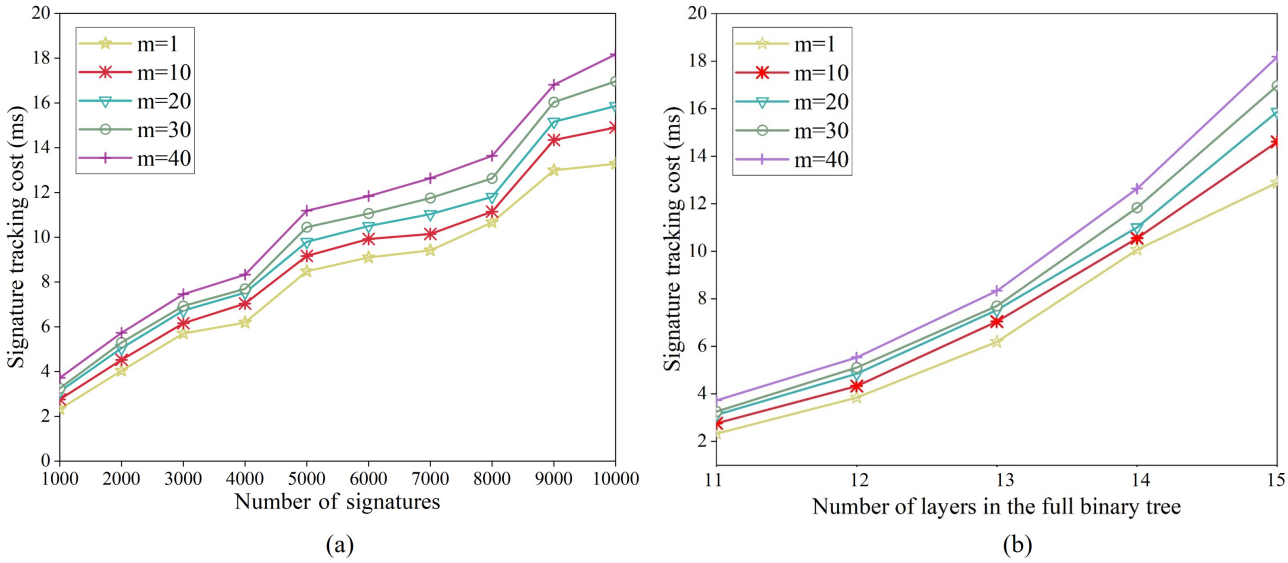


Fig. 7. Computational cost of signature tracking

illegal signatures reaches 40, the tracking cost is within 20 ms. Fig. 7 (b) shows that the signature tracking cost increases significantly with the increase of the number of full binary tree layers, because each increase in the number of full binary tree layers doubles the number of leaf nodes, that is, the number of signatures is doubled. However, even if the number of layers increases to 15 (the number of signatures is $2^{14-1} = 16384$) and the number of illegal signatures reaches 40, the signature tracking cost is still less than 20 ms. Therefore, it can be seen that the efficiency of the proposed scheme in tracking illegal signatures is excellent.

VIII. Conclusion

To realize the privacy protection of V2G networks, we proposed a traceable and privacy-preserving authentication scheme for energy trading in V2G networks. First, we designed a secure and efficient CLS scheme, which can not only resist internal and external adversaries, but also meet other security requirements in V2G networks. Then, we used a method of binary tree level traversal to quickly track EVs with illegal signatures. Finally, we conducted a comprehensive security analysis in the ROM and gave a formal security proof under the CDH assumption. Furthermore, we implemented our scheme by using Relic Library, the experimental results show that our scheme has less computational cost and communication overhead as compared to the existing schemes. Therefore, our scheme has high security and efficiency, and is suitable for in the V2G network environment with more participants or low bandwidth.

References

[1] M. H. Rehmani, M. Reisslein, A. Rachedi, M. Erol-Kantarci, and M. Radenkovic, "Integrating renewable energy resources into the smart grid: Recent developments in information and communication technologies," *IEEE Trans. Ind. Informat.*, vol. 14, no. 7, pp. 2814–2825, Jul. 2018.

[2] Y. Su, G. Shen, and M. Zhang, "A novel privacy-preserving authentication scheme for V2G networks," *IEEE Systems J.*, vol. 14, no. 2, pp. 1963–1971, Jun. 2020.

[3] A. Abdallah and X. Shen, "Lightweight authentication and privacy-preserving scheme for V2G connections," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2615–2629, Mar. 2017.

[4] K. Kaur, S. Garg, G. Kaddoum, F. Gagnon, S. H. Ahmed, and M. Guizani, "A secure, lightweight, and privacy-preserving authentication scheme for V2G connections in smart grid," in *Proc. IEEE INFOCOM Workshops*, Paris, France, 2019, pp. 541–546.

[5] P. Gope and B. Sikdar, "An efficient privacy-preserving authentication scheme for energy internet-based vehicle-to-grid communication," *IEEE Trans. Smart Grid.*, vol. 10, no. 6, pp. 6607–6618, Jun. 2019.

[6] Y. Zhang, J. Zhou, and R. Guo, "Efficient privacy-preserving authentication for V2G networks," *Peer-to-Peer Netw. Appl.*, vol. 14, no. 3, pp. 1366–1378, Oct. 2021.

[7] S. Aggarwal, N. Kumar, and P. Gope, "An efficient blockchain-based authentication scheme for energy-trading in V2G networks," *IEEE Trans. Ind. Informatics*, vol. 17, no. 10, pp. 6971–6980, Oct. 2021.

[8] M. Zhang, B. Zhu, Y. Li, and Y. Wang,"TPM-based conditional privacy-preserving authentication protocol in VANETs," *Symmetry*, vol. 14, no. 6, pp. 1123, May. 2022.

[9] A. Karati, S. H. Islam, G. P. Biswas, M. Z. A. Bhuiyan, P. Vijayakumar, and M. Karuppiah, "Provably secure identity-based signcryption scheme for crowdsourced industrial Internet of Things environments," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2904–2914, Aug. 2018.

[10] G. Wu, Z. Zhao, F. Guo, W. Susilo, and F. Zhang, "On the general construction of tightly secure identity-based signature schemes," *Comput. J.*, vol. 63, no. 12, pp. 1835–1848, Oct. 2021.

[11] J. Li, Y. Ji, K. Choo, and D. Hogrefe, "CL-CPPA: Certificate-less conditional privacy-preserving authentication protocol for the Internet of Vehicles," *IEEE Internet Things J.*, vol. 6, no. 6, pp. 49–51, May. 2019.

[12] Gowri Thumbur, G. Srinivasa Rao, P. Vasudeva Reddy, N. B. Gayathri, D. V. R. Koti 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, Aug. 2020.

[13] 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.*, 2018, pp. 2334–2338.

[14] 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, Apr. 2019.

[15] 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.

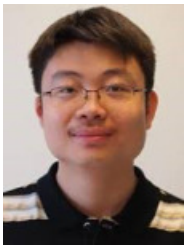
[16] Z. Xu. D. He, N. Kumar, and K.-K.-R. Choo, "Efficient certificateless aggregate signature scheme for performing secure routing in VANETs," *Secur. Commun. Netw.*, vol. 2020, pp. 1–12, Feb. 2020.
[17] Q. Mei, H. Xiong, J. Chen, M. Yang, S. Kumari, and M. K. Khan, "Efficient certificateless aggregate signature with conditional privacy preservation in IoV," *IEEE Syst. J.*, vol. 15, no. 1, pp. 245–256, Mar. 2021.
[18] H. Wang, L. Wang, K. Zhang, J. Li, and Y. Luo, "A conditional privacy-preserving certificateless aggregate signature scheme in the standard model for VANETs," in *IEEE Access*, vol. 10, pp. 15605–15618, Feb. 2022.
[19] Y. Liang and Y. Liu, "Analysis and improvement of an efficient certificateless aggregate signature with conditional privacy preservation in VANETs," *IEEE Syst. J.*, vol. 17, no. 1, pp. 664–672, Mar. 2023.
[20] Kui. Ma, Y. Zhou, Y. Wang, C. Dong, Z. Xia, B. Yang, and M. Zhang, "An efficient certificateless signature scheme with provably secrity and its applications," *IEEE Syst. J.*, pp. 1–12, 2023, doi: 10.1109/JSYST.2023.3269597.
[21] V. Hassija, V. Chamola, S. Garg, D. Krishna, G. Kaddoum, and D. Jayakody, "A blockchain-based framework for lightweight data sharing and energy trading in V2G network," *IEEE Trans. Veh. Technol.*, vol. 69, no. 6, pp. 5799–5812, Jun. 2020.
[22] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proc. Advances in Cryptology*, Santa Barbara, California, USA, Aug. 1984, pp. 47–53.
[23] S. Al-Riyami and K.G. Paterson, "Certificateless public key cryptography," in *Proc. 9th International Conference on the Theory and Application of Cryptology and Information Security*, Taipei, Taiwan, Dec. 2003, pp. 452–473.
[24] D. H. Yum and P. J. Lee, "Generic construction of certificateless signature," in *Proc. 9th Australasian Conference on Information Security and Privacy*, Sydney, Australia, Jul. 2004, pp. 200–211.
[25] X. Huang, W. Susilo, Y. Mu, and F. Zhang, "On the security of certificateless signature schemes from asiacrypt 2003," in *Proc. 4th International Conference on Cryptology and Network Security*, Xiamen, China, Dec. 2005, pp. 13–25.
[26] Z. Zhang, D.S. Wong, J. Xu, and D. Feng, "Certificateless public-key signature: Security model and efficient construction," in *Proc. 4th International Conference on Applied Cryptography and Network Security*, Singapore, Jun. 2006, pp. 293–308.
[27] J. H. Park and B.G. Kang, "Security analysis of the certificateless signature scheme proposed at SecUbiq," in *Proc. Emerging Directions in Embedded and Ubiquitous Computing Workshops*, Taipei, Taiwan, Dec. 2006, pp. 686-691.
[28] J. K. Liu, M. H. Au, and W. Susilo, "Self-generated-certificate public key cryptography and certificateless signature/encryption scheme in the standard model," in *Proc. ACM Symposium on information, Computer and Communication Security*, 2007, pp. 273–283.
[29] Y. Yuan and C. Wang, "Certificateless signature scheme with security enhanced in the standard model," *Inf. Process. Lett.*, vol. 114, no. 9, pp. 492-499, Sept. 2014.
[30] K. A. Shim, "A new certificateless signature scheme provably secure in the standard model," *IEEE Syst. J.*, vol. 13, no. 2, pp. 1421–1430, Jun. 2019.
[31] H. Yan, J. Li, J. Han, and Y. Zhang, "A novel efficient remote data possession checking protocol in cloud storage," *IEEE Trans. Inf. Forensics. Secur.*, vol. 12, no. 1, pp. 78–88, Jan. 2017.
[32] J. Li, H. Yan, and Y. Zhang,"Identity-based privacy preserving remote data integrity checking for cloud storage," *IEEE Syst. J.*, vol. 15, no. 1, pp. 577–585, Mar. 2021.
[33] J. Li, H. Yan, and Y. Zhang, "Certificateless public integrity checking of group shared data on cloud storage," *IEEE Trans. Serv. Comput.*, vol. 14, no. 1, pp. 71–81, Jan. 2021.
[34] R. Castro and R. Dahab, "Efficient certificateless signatures suitable for aggregation," in *Cryptol.ePrint Archive*, Berlin, Germany: Springer, 2007, p. 454.
[35] M. Zhang, Y. Chen, and W. Susilo, "Decision tree evaluation on sensitive datasets for secure e-healthcare systems," *IEEE Trans. Dependable Secur.*, pp. 1–14, 2022. doi: 10.1109/TDSC.2022.3219849.
[36] G. Shen, Z. Fu, Y. Gui, W. Susilo, and M. Zhang, "Efficient and privacy-preserving online diagnosis scheme based on federated learning in e-healthcare system," *Inf. Sci.*, vol. 647, pp. 1–16, Jun. 2023.
[37] M. Zhang, S. Huang, G. Shen, and Y. Wang, "PPNNP: A privacy-preserving neural network prediction with separated data providers using multi-client inner-product encryption," *Comput. Stand. Interfaces*, vol. 84, pp. 103678, Jan. 2023.
[38] M. Zhang, Z. Li, and P. Zhang, "A secure and privacy-preserving word vector training scheme based on functional encryption with inner-product presicates," *Comput. Stand. Interfaces*, vol. 86, pp. 103734, Aug. 2023.
[39] Z. Kang, J. Li, J. Shen, J. Han, Y. Zuo, and Y. Zhang, "TFS-ABS: Traceable and forward-secure attribute-based signature scheme with constant-size," *IEEE Tran. Knowl. Data Eng.*, vol. 35, no. 9, pp. 9514–9530, Sep. 2023.
[40] D. Boneh and M. K. Franklin, "Identity-based encryption from the weil pairing," in *CRYPTO*, 2001, pp. 213–229.
[41] L. Liu, G. Han, Z. Xu, J. Jiang, L. Shu, and M. Martinez-Garcia, "Boundary tracking of continuous objects based on binary tree structured svm for industrial wireless sensor networks," *IEEE Trans. Mob. Comput.*, vol. 21, no. 3, pp. 849–861, Mar. 2022.
[42] D. F. Aranha, C. P. L. GouvÃła, T. Markmann, R. S. Wahby, and K. Liao, RELIC is an Efficient Library for Cryptography, https://github.com/relic-toolkit/relic.

Gang Shen received the B.S. degree in electrical automation from Wuhan Institute of Technology, Wuhan, China, in 2002, and the M.S. degree in control theory and control engineering from Huazhong University of Science and Technology, Wuhan, China, in 2009, and the Ph.D. degree in traffic information engineering and control from Wuhan University of Technology, Wuhan, China, in 2019. He is currently a Lecturer with the School of Computer Science, Hubei University of Technology, Wuhan, China. His current research interests include cryptography, network security, and privacy preservation.

Chengliangyi Xia is currently pursuing the M.S. degree with the School of Computer Science, Hubei University of Technology, Wuhan, China. His current research interests include Blockchain security and privacy preservation.

Yumei Li received her Ph.D. degree in statistics from Nanjing Normal University, China. She is currently a lecturer at the Hubei University of Technology. Her main research interests include the linear homomorphic signature and its applications in network coding, cloud computing, and blockchain.

Hua Shen received her Ph.D. degree in computer science from the School of Computer Science, Wuhan University (China) in 2014. She is currently a Professor in the School of Computer Science at Hubei University of Technology (HBUT), China. During 2019 to 2020, she has visited University of Wollongong (Australia) as a visiting Research Fellow, supervised by Prof. Willy Susilo. Her research interests include privacy computing and information security.



Weizhi Meng is currently an Associate Professor in the Department of Applied Mathematics and Computer Science, Technical University of Denmark (DTU), Denmark. He won the Outstanding Academic Performance Award during his doctoral study, and is a recipient of the Hong Kong Institution of Engineers (HKIE) Outstanding Paper Award for Young Engineers/Researchers in both 2014 and 2017. He also received the IEEE ComSoc Best Young Researcher Award for Europe, Middle East, & Africa Region (EMEA) in 2020. His primary research interests are intersections among cyber security, artificial intelligence and blockchain technology, such as intrusion detection, IoT security, biometric authentication, and blockchain. He is an ACM Distinguished Speaker, and is directing the SPTAGE Lab at DTU.

Mingwu Zhang is a Professor with School of Computer Science, Hubei University of Technology, Wuhan, China, and also the Director of the Hubei Engineering Research Centre for Industrial Big Data. He received his M.S. degree in Computer Science and Engineering from Hubei Polytechnic University in 2000 and the Ph.D. degree in South China Agric University in 2009, respectively. From August 2010 to August 2012, he was a JSPS Postdoctoral Fellow with the Japan Society of Promotion Sciences, Institute of Mathematics for Industry, Kyushu University, Fukuoka, Japan. From 2015 to 2016, he was a Senior Visiting Scholar with the School of Computing and Information Technology, University of Wollongong, Wollongong, NSW, Australia. His research interests include cryptography technology for networks and data security, secure computation and privacy preservation in big-data and clouds.

Prof. Zhang received Five Best Paper Awards in the international conference, such as ACISP'18 and Inscrypt'18. He has served as a Program Committee Member of several international conferences and published over 100 articles in international conferences and journals, such as ASIACRYPT, ACISP, ProvSec, ISPEC, Inscrypt, the IEEE Transactions on Information Forensics and Security, the Theoretical Computer Science, and the IEEE Transactions on Dependable and Secure Computing.