PBAG: A Privacy-Preserving Blockchain-Based Authentication Protocol With Global-Updated Commitment in IoVs

Xia Feng, Kaiping Cui, Liangmin Wang, Member, IEEE, Zhiquan Liu, and Jianfeng Ma, Member, IEEE

Abstract—Internet of Vehicles (IoVs) is increasingly used as a medium to propagate critical information via establishing connections between entities such as vehicles and infrastructures. During message transmission, privacy-preserving authentication is considered the first line of defence against attackers and malicious information. To achieve a more secure and stable communication environment, ever-increasing numbers of blockchain-based authentication schemes are proposed. At first glance, existing approaches provide robust architectures and achieve transparent authentication. However, in these schemes, verifiers need to conduct real-time operations in the blockchain (e.g., querying certificates). To remedy this limit, we propose a privacy-preserving blockchain-based authentication protocol with global-updated commitment (PBAG). In PBAG, based on the identity-based cryptosystem, a global commitment is computed, and a unique evaluation proof is generated for each authorized vehicle. Instead of querying the blockchain in real-time, verifiers can independently authenticate vehicles using the global commitment that is pre-updated with the assistance of the blockchain. Moreover, our scheme proposes a dynamic update mechanism to ensure the freshness of the global commitment and evaluation proofs. Benefiting from the update mechanism, there will be an authentication failure for vehicles holding invalid certificates when using the latest global commitment, thus avoiding the time-consuming of checking the Certificate Revocation List (CRL). In terms of privacy protection, our scheme provides privacy properties such as anonymity and unlinkability. It allows anonymous authentication based on evaluation proofs and achieves traceability of identity in the event of a dispute. The simulation demonstrates that the average computation cost of verifying per message is 0.36ms under the batch-enabled mechanism, reducing by more than 63.7% compared with existing schemes.

Index Terms—Blockchain, authentication, privacy-preserving, commitment.

I. INTRODUCTION

The Internet of Vehicles (IoVs) intends to predominantly facilitate the dissemination of critical information in real-time in the Intelligent Transportation System (ITS). It has been well recognized as a promising technology to enhance traffic safety and driving experience [1], especially in various applications such as congestion control, traffic management, and collision avoidance. Technically, operations in IoVs enable vehicles to exchange information such as speed, location, and heading through the Dedicated Short-Range Communication (DSRC) protocol standard [2]. The primary forms of information exchange mainly include Vehicle-to-Vehicle (V2V), Vehicle-to-Infrastructure (V2I), etc. Besides, cooperative safety applications are a significant branch of IoVs. The emergency message dissemination enables each vehicle to intelligently perceive surrounding conditions and make timely decisions about potential dangers. Unfortunately, due to the openness of the wireless channel, malicious users or adversaries are inevitable in IoVs, which could launch arbitrary attacks on the message communication process, such as replay attacks, forgery attacks, etc. [3]. These security threats can cause traffic congestion and disrupt traffic order. In addition, adversaries can also eavesdrop and collect beacon messages broadcast by the vehicle to infer the vehicle’s driving route and track the vehicle’s trajectory [4]. Thus, security and privacy have become crucial and indispensable issues for IoVs. In this connection, reliable authentication schemes are required to effectively support information dissemination, particularly amid the transformation from assisted driving systems to self-driving vehicles.

Numerous researchers at home and abroad propose various authentication schemes, which mainly fall into three categories, such as public key cryptosystem [5], identity-based cryptosystem [6, 7] and group-based protocols [8, 9]. Public key cryptosystem always needs an online Certificate Authority (CA), which often brings communication latency and involves certificate management problems (e.g., maintaining a cumbersome Certificate Revocation List (CRL)). Although identity-based cryptosystem schemes can alleviate these problems, they must process the escrow issues. There is a risk of secret key leakage once Key Generation Center (KGC) is compromised. Group signature-based schemes can protect users’ true identity, but they may incur additional computation and communication overheads. Moreover, the ability of privacy protection depends on the number of attended vehicles. However, the optimum number of vehicles joining the group...
can only be guaranteed sometimes, particularly in rural areas with sparse vehicles. In addition, most of the above three types of authentication schemes are constructed based on a central architecture, which may cause some problems, such as excessive computing burden and a single point of failure of the central server.

Currently, emerging blockchain technology has received more attention in IoVs. Blockchain can provide a robust and stable communication environment for authentication, effectively avoiding problems such as the single point of failure. Therefore, ever-increasing numbers of blockchain-based authentication schemes are proposed [10], [11], [12], [13], [14], [15], [16]. Blockchain is mainly utilized to store certificates or status change records in such approaches, and vehicles/infrastructures perform the authentication process with smart contracts. However, existing approaches face two major challenges in the authentication process over the IoVs scenario. Firstly, vehicles/infrastructures must establish a connection with the blockchain network in the authentication process, which incurs additional latency. Secondly, the execution of smart contracts is based on the consensus mechanism, which causes additional time overhead in the authentication process.

In this paper, to address the aforementioned efficiency and privacy problems, we propose a novel blockchain-based authentication protocol with global-updated commitment (PBAG). The main contributions are listed below:

- **Our proposal enables authentication without interacting with the blockchain in real-time.** With the assistance of global commitment, the whole authentication process only needs one End-to-End (E2E) wireless transmission between verifiers and vehicles, and verifiers do not need to interact with the blockchain in real-time. More concretely, Root Authority (RA) computes a unique evaluation proof corresponding to the issued certificate for each authorized vehicle and broadcasts a public global commitment based on all valid certificates. The verifiers can directly verify the vehicles’ authorization status by using the global commitment to check the validity of evaluation proofs.

- **Our proposal avoids the time-consuming of checking CRL in the authentication process.** Considering changes of certificate status, our scheme proposes a dynamic update mechanism with the assistance of blockchain to ensure the freshness of the global commitment and evaluation proofs. Benefiting from the update mechanism, there will be an authentication failure for vehicles holding invalid certificates when using the latest global commitment, thus avoiding the time-consuming of checking the CRL. Specifically, the evaluation proof is computed based on the vehicle’s certificate to represent the authorization status. Public global commitment is computed based on all valid certificates in the IoVs. Thus, by utilizing the latest global commitment to check the validity of evaluation proofs directly, vehicles holding invalid certificates cannot pass the authentication.

- **Our proposal achieves privacy-preserving properties.** In PBAG, the zero-knowledge proof enables authentication between vehicles to be processed anonymously, and it can achieve traceability of identity in the event of a dispute. Besides, in terms of linkability, the verifier cannot link messages with a particular user during the authentication. Finally, we conduct the security analysis and demonstrate that our scheme can resist typical attacks.

- **Our proposal conducts theoretical analysis and simulation.** Considering the real-time requirements in the IoVs scenario, we conduct the extensive simulation in Hyperledger Fabric v2.0.0 and Network Simulator v2 to evaluate our proposal in terms of authentication latency and information transmission efficiency. The performance evaluation indicates that the PBAG can significantly reduce the authentication latency and message loss rate. With batch authentication, the average computation cost of verifying per message is 0.36ms, reducing by more than 63.7% compared with [17], [18], [19]. Besides, the message loss rate of our scheme is controlled within 9.6%, outperforming the common schemes and satisfying the communication requirements in IoVs.

The remainder of this paper is organized as follows: Section II reviews some related works. Section III introduces several preliminaries and mathematical assumptions. The framework and threat models are illustrated in Section IV. Section V presents the proposed scheme. The analysis of the security and privacy of our model is in Section VI. In Section VII, we construct the simulation and compare our proposal with existing schemes. Finally, we conclude our works and illustrate the future works in Section VIII.

II. RELATED WORK

This section provides a review of existing work relating to authentication. II-A provides an overview of the commonly used authentication methods constructed with various technologies in IoVs. II-B emphatically presents an overview of blockchain-based authentication protocols in IoVs.

A. An Overview of Authentication in IoVs

Along with the gradual popularization of IoVs, vehicles need to communicate a large amount of data with other entities in high-density and high-traffic environments. In this process, identity authentication provides a countermeasure to resist malicious user attacks while achieving the privacy protection of vehicles. The following content will review recent authentication schemes proposed by researchers at home and abroad.

For instance, with Elliptic Curve Cryptography (ECC), Alshudukhi et al. [20] constructed a lightweight authentication scheme for secure IoVs communications while achieving conditional privacy protection. It was worth mentioning that, compared to traditional protocols that stored initial public parameters and keys of the system in the On Board Unit (OBU), the scheme preloaded these into the Tamper-Proof Device (TPD) of each anti-tamper device of the RSU. Similarly, the method that relied on the TPD was also used in scheme [21] to achieve conditional privacy protection. To protect the security of the intelligent terminal in IoVs,
Wei et al. [22] proposed two intelligent terminal-based privacy-preserving multi-modal implicit authentication protocols, where the password and the vehicle owner’s behaviour features are utilized as the authentication factors. Meanwhile, the scheme can guarantee that sensitive and private information is not leaked, such as vehicle owner’s behavior features. Security analysis and experiments demonstrate that this scheme can yield better security and efficiency compared to other related schemes. Zhang et al. [23] proposed a novel privacy-preserving authentication protocol using bilinear pairings. This protocol employed the hash function to compute the vehicle’s true ID, and the Trusted Authority (TA) was responsible for verifying the correctness of the timestamp and message digest, which only required low communication overhead and computational cost. Besides, to improve efficiency, the scheme supports the vehicle in verifying multiple messages with a batch authentication mechanism. In terms of security, the scheme can provide various security and privacy features, such as traceability, revocability, unlinkability, and resist common attacks. Theodore et al. [24] proposed a new lightweight authentication and privacy-preserving protocol, which contains five main phases: system initialization, entity registration, mutual authentication, broadcast and verification, and vehicle revocation phases. With the assistance of an improved real-time efficient stream loss tolerant authentication (TESLA) containing a Cuckoo Filter (CF), the scheme effectively preserved vehicles’ real information within the Road Side Unit (RSU) range. Moreover, the scheme also constructed lightweight two-way authentication while achieving reliable privacy protection and resistance to common attacks.

Based on Public Key Infrastructure (PKI) and Certificate-Less Public Key Cryptography (CL-PKC), Altaf and Maity [25] proposed an efficient localized hybrid authentication protocol with privacy protection. In order to minimize centralized dependency, the protocol assigned the key management task to multiple PKI-authenticated local semi-trusted RSUs as well as delegated the identity management to the Transport Root Authority (TRA). In terms of efficiency, the protocol supports vehicles in authenticating outgoing messages with a new and efficient certificate-less signature scheme. Moreover, the formal security analysis indicated that this scheme can defend against various types of forgery and impersonation attacks. Furthermore, Ali et al. [26] constructed a certificateless cryptosystem system and proposed a PKI-based hybrid signcryption scheme, which achieves reliable conditional privacy protection. In terms of efficiency, the scheme provided a batch unsigncryption method, which supported the vehicle in unsigncrying multiple messages simultaneously. Nevertheless, such schemes inevitably consume hosts of storage to store certificates. Besides, the management of all the certificates would incur extra computational and communication costs [27]. Wang and Liu [28] proposed a secure and effective message authentication protocol, which achieves mutual authentication between vehicles and RSUs by combining the advantages of pseudonym-based and group signature-based methods. In order to tackle the security issues in V2V communication, the scheme constructed an efficient and secure group-based message authentication algorithm, especially making the vehicle not be framed by other compromised vehicles or RSUs. However, this scheme completes the authentication process through bilinear operations, which results in more computational overhead. Bansal et al. [29] utilized identity-based cryptography and ECC to construct an identity-based computationally efficient privacy-preserving authentication scheme for Vehicle-to-Vehicle (V2V) communication in IoVs. During authentication, the scheme can support batch signature verification and guarantee many features, such as message integrity, non-repudiation, and anonymity of the vehicle’s identity. Regrettably, there is no outstanding performance, especially in the message verification phase. Moreover, the scheme is weak in terms of attack resistance, e.g., not resistant to the Denial-of-Service (DoS) attack. Similarly, there are also hosts of works aiming to realize efficient batch authentication to prevent valuable messages from being discarded due to authentication delays in IoVs [30], [31], [32], [33], [34], [35].

B. Blockchain-Based Authentication in IoVs

Although various technologies involved in II-A play significant roles in facilitating research on identity authentication, considering the dynamics and complexity of IoVs topology, many defects are still unavoidable, such as single-point failure problems. Recently, a wide range of research works related to blockchain-based authentication have been proposed at home and abroad, which provides a promising approach to constructing distributed authentication architecture. This subsection provides an overview of existing works relating to blockchain-based authentication protocols and highlights the motivation for considering the utilization of the commitment mechanism to address potential challenges.

Blockchain technology [36] emerges as a promising approach for decentralized authentication for IoVs. For instance, Li et al. [12] proposed a double-layer blockchain and decentralized identifiers-assisted secure registration and authentication scheme for IoVs, namely BDRA. The scheme employs the decentralized identifiers technology in IoVs to reduce the dependence on the trusted third party, as well as further enhancing the decentralized nature of the network. BDRA changed the traditional vehicle networking architecture, designed a double-layer blockchain structure for IoVs, and added vehicles as nodes to the blockchain. However, due to the dynamic topology of IoVs in practice, it is difficult to construct a blockchain network using dynamic vehicles as nodes in terms of practical feasibility.

In addition, a host of researchers utilize encryption to construct secure, reliable, and efficient identity authentication protocols. To tackle the challenges associated with revoking private keys and frequent interactions, Lin et al. [13] proposed a blockchain-based conditional privacy-preserving authentication protocol (BCPPA), which integrates blockchain technology and key derivation algorithms. This protocol alleviates the need for vehicles’ OBU to store a large number of private keys and fulfills the requirement for accelerated...
authentication by employing batch authentication processes. However, the scheme fails to achieve efficient traceability, and the efficiency of message authentication needs further improvement. Therefore, the author team further proposed a more efficient authentication protocol [14], which achieves a traceable one-time anonymous public key generation process and encompasses a more effective message authentication and tracking mechanism. Additionally, in order to achieve tamper-resistant and traceability in the authentication process, Wang et al. [37] proposed a paring-assisted V2I authentication scheme with trustworthiness scalable computation. This scheme encrypts the combination of trustworthiness and attributes of vehicles, then stores ciphertext into the blockchain. However, bilinear pairing operation incurs high computation costs during the authentication process in this scheme.

From the above analyses, authentication efficiency is a crucial factor when constructing the authentication protocol based on blockchain technology. Numerous researchers at home and abroad have conducted extensive research on this issue. For example, He et al. [15] proposed a hierarchical blockchain-assisted conditional privacy-preserving authentication scheme, which guarantees multiple security properties, such as unlinkability, traceability, and resistance to various attacks. Still, it also has the flaw of being ineffective. In order to break through this dilemma, numerous researchers at home and abroad have conducted a host of studies on lightweight authentication. For instance, Son et al. [38] proposed a blockchain-based lightweight V2I handover authentication scheme. The scheme enables vehicles to perform lightweight computations in handover situations. Additionally, benefiting from the mechanism of multiple institutions jointly maintaining blockchain databases, the scheme supports that the vehicle can authenticate to RSUs using only hash and XOR operations as long as the vehicle has completed initial authentication with an RSU before. In terms of security, the scheme can defend against various attacks and can provide many security features, such as anonymity and untraceability. Zhang et al. [39] proposed a lightweight privacy-preserving traffic route management scheme that combines blockchain technology and the fog computing model. This scheme allowed vehicles to utilize homomorphic encryption to encrypt driving routes and send them to a fog node. The fog node is responsible for aggregating the received ciphertexts, and the Traffic Management Center (TMC) can perform traffic management by decrypting aggregation results. In this process, TMC only learns the total number of vehicles on each road segment rather than the individual routes of the vehicles. Moreover, in order to achieve efficient vehicular communication, fog computing is introduced to support mobility, low latency, and location awareness, thereby reducing traffic congestion in VANETs and ensuring communication security. Blockchain is used to conduct public key management of vehicles. The performance analysis indicates that the scheme outperforms other related representative schemes in terms of security and efficiency.

Leveraging smart contracts to develop identity authentication protocols suitable for the IoVs scenario has also attracted lots of attention. With the smart contract, Xue et al. [40] proposed a blockchain-based tamper-proof roaming authentication protocol for IoVs. The authors leverage smart contracts to achieve the protocol’s different phases, including user/AP registration, authentication, and revocation. Besides, the protocol constructs an unforgeable billing scheme by deploying bloom filters on the blockchain, contributing to security and performance validations. The security and performance analysis are good evidence that the protocol can provide reliable security features and incur acceptable authentication costs. Similarly, Zhang et al. [41] proposed a dual blockchain-assisted conditional privacy-preserving authentication protocol for IoVs, which can ensure that the transmitted information is reliable even without a centralized third party. With the assistance of smart contracts, the scheme achieved conditional tracking and decentralized dynamic revocation for illegal vehicles, rendering the scheme efficient and scalable. Security and performance analyses indicate that the scheme has certain advantages deploying in IoVs scenario compared with related schemes. Zhou et al. [42] proposed a more effective blockchain-based EBCPPA protocol for IoVs. The study analyzed the security requirements of current practical applications and designed two building modules: the SoK scheme and smart contracts. Specifically, the scheme constructed a SoK scheme for vehicles and RSUs to realize efficient authentication in communications. Besides, with the assistance of smart contracts, the scheme provides anonymous public key management and key revocation. The security analysis and performance validations show that EBCPPA can achieve reliable privacy protection and is more efficient than other existing state-of-the-art schemes.

Most blockchain-based authentication approaches face three major challenges. Firstly, such schemes need to involve blockchain interaction during the authentication phase, leading to longer delays and huge computation costs. Secondly, lacking a flawless identity management mechanism. In particular, verifiers need to query a CRL to check the current status of the certificates during the authentication process, which incurs extra communication overhead. Apart from that, plenty of blockchain-based authentication schemes do not support sufficient decentralization, and the single trusted authority has the most permissions. Therefore, the existing works cannot be fully adapted to real-world scenarios.

III. PRELIMINARIES AND MATHEMATICAL ASSUMPTIONS

In this section, we introduce several preliminaries and mathematical assumptions involved in this paper.

A. Bilinear Pairing

We define three basic groups, $G_1$, $G_2$ and $G_T$ are multiplicative cyclic group of prime order $q$. It is denoted as $G_1 = \langle g \rangle$, $G_2 = \langle g \rangle$ and $G_T = \langle g \rangle$, respectively. Thus, $e: G_1 \times G_2 \rightarrow G_T$ is a bilinear mapping [43], [44] with the following properties:

- **Bilinear**: For $\forall g_1 \in G_1$, $\forall g_2 \in G_2$, $a, b \in \mathbb{Z}_q$, it satisfies $e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}$.
- **Non-degeneracy**: $\exists g_1 \in G_1$, $\exists g_2 \in G_2$, which satisfies $e(g_1, g_2) \neq 1_{G_T}$.
where depend on the degree of the committed polynomial. In this

\[ C \] 

Multi-Party Computation (MPC) protocol \[48\] to compute the

\[ \text{commitment scheme can generate an evaluation proof for any} \]

\[ \text{referred to as the KZG Commitment Scheme} \] \[46\]. KZG

\[ n \]

C. KZG Polynomial Commitments

Kate, Zaverucha and Goldberg propose a constant-size com-

\[ \text{mitment scheme for degree} \]

\[ j \]

\[ \text{Global commitment.} \]

\[ \text{we can find the unique polynomial} \]

\[ \Psi(X) \]

\[ \text{for the given} \]

\[ n \]

\[ \text{pairs} \]

\[ (x_i, y_i)_{i \in [0, n)} \]

\[ \text{using Lagrange interpolation} \] \[45\], and it satisfies \[\Psi(x_i) = y_i\]. Specifically, the \[\Psi(X) \]

\[ \text{is computed as} \]

\[ \Psi(X) = \sum_{i \in [0, n]} \ell_i(X) y_i \text{ and its degree is less than} n, \]

\[ \text{where} \]

\[ \ell_i(X) = \prod_{j \in [0, n], j \neq i} \frac{X - x_j}{x_i - x_j}. \]

It is crucial that \[\ell_i(X) \]

\[ \text{has property that} \]

\[ \ell_i(x_i) = 1 \text{ and} \]

\[ \ell_i(x_j) = 0, \forall i, j \in [0, n) \text{ with} \]

\[ j \neq i. \]

D. Mathematical Assumptions

We elaborate the following assumptions as the basis for our

\[ \text{scheme.} \]

\[ \text{n-Diffie-Hellman Exponent problem (n-DHE): For one} \]

\[ \text{group} \]

\[ G \]

\[ \text{of prime order} q, \]

\[ \text{let} \]

\[ g_1 = g^{y_1}, \]

\[ \gamma \leftarrow \mathbb{Z}_q. \]

\[ \text{On input} \]

\[ (g_1, g_2, \ldots, g_{n+2}, \ldots, g_{2n}) \in \mathbb{G}^{2n}, \]

\[ \text{it is hard to output} \]

\[ g_{n+1}. \]

\[ \text{Elliptic Curve Diffie-Hellman Problem (ECDHP) \[49\]} \]

\[ \text{Setting an additive cyclic group} \]

\[ \mathbb{G} \]

\[ \text{constructed with an} \]

\[ \text{elliptic curve} \]

\[ E \]

\[ \text{and the generator of} \]

\[ G \]

\[ \text{is} \]

\[ P \]

\[ \text{the order is} q. \]

\[ \text{The ECDHP problem is defined as follows: given} \]

\[ \text{elements in} \]

\[ \mathbb{G} \]

\[ \text{of} \]

\[ V, R, \]

\[ \text{where} \]

\[ V = v \cdot P, \]

\[ R = r \cdot P, \]

\[ v, r \in \mathbb{Z}_q^* \]

\[ \text{it is hard to compute} \]

\[ vrP \]

\[ \text{with knowledge} \]

\[ V, R, P. \]

\[ \text{Elliptic Curve Discrete Logarithm Problem (ECDLP) \[50\]} \]

\[ \text{Setting an additive cyclic group} \]

\[ \mathbb{G} \]

\[ \text{constructed with an} \]

\[ \text{elliptic curve} \]

\[ E \]

\[ \text{and the generator of} \]

\[ G \]

\[ \text{is} \]

\[ P \]

\[ \text{the order is} q. \]

\[ \text{The ECDLP problem is defined as follows: given} \]

\[ \text{elements in} \]

\[ \mathbb{G} \]

\[ \text{of} \]

\[ G \]

\[ \text{and} \]

\[ P \]

\[ \text{it is intractable to compute} \]

\[ r \]

\[ \text{with knowledge} \]

\[ R \]

\[ \text{and} \]

\[ P \]

\[ \text{in probability polynomial time.} \]

IV. FRAMEWORK AND THREAT MODEL

Because of dynamic topology and high node mobility, IoVs suffer from a series of security vulnerabilities and privacy issues. In general, vehicles should be responsible for their disseminated safety and transport messages. Especially, there are many adversaries or misbehaving nodes trying to forge messages for benefit. Herein, vehicles in IoVs have a two-fold requirement. Firstly, vehicles should check the legitimacy of the communicating partner; secondly, vehicles should build their way of identification to showcase their legitimacy. Considering these requirements, we propose a privacy-preserving blockchain-based authentication protocol named PBAG. This section introduces our security framework, network components, security assumption, and threat model.

A. Framework and Components

We intend to provide an efficient authentication protocol in IoVs. The framework of our scheme is shown in Fig. 1, which comprises a three-layer architecture. The root authority in the first layer is responsible for the vehicle’s certificate issuance and generating system parameters and key pairs. The second layer is composed of Regional Trusted Authorities (RTAs) and blockchain, which is in charge of authenticating the vehicles or RSUs. Besides, RTAs are equipped with sufficient computing power to collect and analyze traffic messages. The third layer is composed of vehicles and RSUs. Vehicles communicate with others through the DSRC protocol \[51\]. RSUs communicate with RTAs or RA via a secure transport protocol (e.g., a wired transport layer security protocol). The main components are described as follows:

1) Root Authority (RA): RA should be an institution authorized by law, and it is responsible for issuing digital certificates for vehicles. Vehicles intending to join IoVs should register themselves with the RA first. Before issuing the vehicle’s
digital certificate, RA thoroughly verifies the vehicle’s information, including a physical examination. The information verified should include but is not limited to the driver’s license, the vehicle owner’s identity, and the actual license plate. This procedure is usually private but could be revealed in support of legal evidence under circumstances like accidental investigation. Besides, RA is the only entity that can track the actual ID of vehicles. In our paper, we assume the RA is a fully trusted institution and never compromised and conspired, and it has enough computing and storage capabilities.

2) Regional Trusted Authority: RTA can verify the received message from RSUs or vehicles. RTAs and RA maintain the blockchain. For a reasonable distribution of computing power, RTAs are responsible for collecting and analyzing traffic messages with sufficient computing power, which can help RTAs make sound responses in critical situations or optimize traffic light control. Besides, we assume the RTAs are trusted institutions and have access to the vehicle’s certificate information based on the blockchain. Therefore, RTAs are in charge of updating global commitment and certificates, which is elaborated in V-D.

3) Road Side Units: RSUs are located alongside the roads to organize and coordinate vehicular communications in an optimized manner. After confirming identity, vehicles apply for certificates by sending a request message to nearby RSUs within their communication range. RSU is also responsible for broadcasting the latest updated global commitment to vehicles within its range and assisting vehicles in completing the anonymous proof update process. Besides, RSU also has a certificate that can prove its legitimacy. In our scheme, we assume that RSUs are semi-trusted.

4) On-Board Unit: In this paper, we assume vehicles are untrusted entities. Each OBU is a computation and communication device in the vehicle. They are embedded with limited computing capability, which can be used to communicate with each other within the range. OBU contains a tamper-proof device [8], which can guarantee the security of stored information. We assume an OBU keeps synchronized with the RSUs. OBUs can carry out the V2V authentication without the help of RSUs.

5) Blockchain: In this paper, RTAs and RA have storage and retrieval permissions for the blockchain state database. RSUs can broadcast information in blockchain networks through links with RTAs. Blockchain has two main responsibilities. First, the blockchain is responsible for storing vehicles’ pseudonyms and certificates. Second, with the assistance of blockchain, the vehicle’s proof update records are broadcast to RTAs, RA, and RSUs.

B. Security Assumption and Threat Model

In our security hypothesis, RA and RTAs are trusted institutions, while RSUs are semi-trusted. The underlying cryptographic mechanisms used to set up a genesis block are secure for blockchain. We assume that vehicles in IoVs can be rational attackers, and they even have some prior knowledge when they launch attacks. Under the security assumption, our model can deal with the threat arising from the authentication procedure. The common threats to our scheme are as follows:

- **Authentication procedure**: The authentication protocol in IoVs does not run in secure channels. The adversary could launch attacks, particularly on the protocols. A successful attack on an authentication protocol usually does not rely on breaking the cryptographic complexity algorithm used in the protocol. Instead, the attacker obtains a certificate in an unauthorized way. We list several security issues and typical attacks below:
  - Correctness and integrity: In the authentication process, the authorized vehicles are able to provide evaluation proof that they are indeed legitimate vehicles. Besides, the management institutions can prove that messages sent by authorized vehicles are correct without being modified or forged.
  - Traceability: In the event of a dispute, it is necessary for RA to reveal the identity by associating it with the verification messages.

- **Privacy-preserving**: The proposed scheme must meet the following basic privacy-preserving requirements:
  - Anonymity: Anonymity is one of the most important aspects in wireless authentication [52]. All vehicles can participate in the IoVs communication scenario without revealing their identity in the proposed scheme.
  - Unlinkability: The vehicle’s ability to hinder the attacker from linking the relationship between two or multiple messages that have been published.

- **Resisting attacks**: The proposed scheme can resist the common attacks in IoVs, such as the replay attack, the impersonation attack, and the modification attack.
  - Replay attack: An attacker repeats a previously transmitted message to intercept and retransmit its modified version, thereby fooling the honest authentication party and giving genuine users the illusion that they have successfully completed the authentication protocol.
  - Modification attack: The attacker modifies the message or authentication materials. Afterwards, the attacker successfully deceives the target entity and passes its verification.
are listed in Table I and Table II.

Besides, we would elaborate on the authentication mechanism following fundamental principles: 1) reduce latency, 2) resist forgery attack: During the certificate issuance operation, adversaries are able to forge an encrypted ID and try to register to the RA to launch an impersonation attack where possible. Thus, the attack would hinder the certificate issuance of benign vehicles. While in the update and revocation procedure, an adversary can pretend to be the holder of the certificate and can initiate the update or revocation operation. The certificates of the benign nodes will be updated or revoked unexpectedly. In other words, the benign nodes lose control of their certificates.

V. THE PROPOSED SCHEME

Vehicles transmit time-critical messages such as emergency braking and lane crossing in IoVs to avoid road accidents. Thus, they should authenticate each other and then transmit the message as soon as possible to have a broader impact. To meet these requirements, we propose PBAG considering the following fundamental principles: 1) reduce latency, 2) resist impersonation attack: During the certificate issuance operation, adversaries are able to forge an encrypted ID and try to register to the RA to launch an impersonation attack where possible. Thus, the attack would hinder the certificate issuance of benign vehicles. While in the update and revocation procedure, an adversary can pretend to be the holder of the certificate and can initiate the update or revocation operation. The certificates of the benign nodes will be updated or revoked unexpectedly. In other words, the benign nodes lose control of their certificates.

A. System Initialization

In this subsection, we elaborate the system initialization process, which mainly includes system parameter generation phase and commitment parameter generation phase.

1) Phase I: System Parameter Generation Phase: In this phase, RA needs to perform the following steps to complete the system parameter generation process, which mainly contains ECC [53] parameters and commitment parameters.

- RA selects two large prime numbers $p$ and $q$. Afterwards, RA defines an elliptic curve $E: y^2 = x^3 + Ax + B \mod p$, where $p > 5$, $A, B \in \mathbb{Z}_p$ and satisfying $4A^3 + 27B^2 \neq 0$.
- RA conducts a additive cyclic group $G$ based on elliptic curve $E$. The generator of group $G$ is $P$, and the order is $q$.
- RA randomly chooses a random number $a \in \mathbb{Z}_q^*$ as its secret key $f_{sk}$, and then computes its public key $f_{pk} = a \cdot P$, which is the master key for the RA.

Finally, RA broadcasts parameters $\{f_{pk}, E, P, G\}$.

2) Phase II: Commitment Parameter Generation Phase: In this phase, RA needs to generate commitment parameters. First, RA defines the cryptographic hash function $H: \{0, 1\}^* \rightarrow \mathbb{Z}_q$ and clipping function $\bar{X} = \text{Clip}(H(X))$. The clipping function is to truncate the field $X$ and obtain its corresponding value $\bar{X}$. After that, RA defines $n$ pairs of parameters $\{(\omega, u_i) | 0 \leq i < n\}$, which will be assigned to registered vehicles in section V-B. Here, $\omega$ is a primitive $n$-th root of unity in $\mathbb{Z}_q$ [54]. Finally, RA generates a polynomial $\Psi(X) = \sum_{i \in [0,n]} \mathcal{L}_i(X)u_i$ using Lagrange polynomial interpolation, where $\mathcal{L}_i(X) = \prod_{j \in [0,n], j \neq i} \frac{X - u_j}{u_i - u_j}$. $\Psi(X)$ satisfies $\Psi(\omega^i) = u_i$. Further, RA generates a KZG commitment to $\Psi(X)$, that is $C = \prod_{i=0}^{n-1} (\Psi(\omega^i))^v = g^{\sum_{i=0}^{n-1} v^i} = g^{\Phi(v)}$. RA sends the global commitment $C$ to RSUs and RTAs, and RSU broadcasts $C$ within its management area.

B. Vehicle Registration

In our model, when RA has confirmed the legality of a vehicle applying for registration, RA will generate the vehicle’s pseudonym and certificate. Meanwhile, RA will update the global commitment and calculate the commitment proof for the vehicle. The Algorithm 1 and Fig. 2 illustrate the three phases of the vehicle registration process.

1) Phase I: Certificate Issuance: Vehicle generates local key pair ($nsk, npk$) based on the public elliptic curve parameters and computes the signature $\sigma_{nsk}$, that is $\sigma_{nsk} \leftarrow \text{Sig}(ID, nsk)$. Afterwards, the vehicle sends request message $R = (ID, registration, npk, \sigma_{nsk})$ to the RA, where $ID$ represents real identity information.
RA verifies the vehicle’s identity information and signature. Afterwards, RA computes the pseudonym \( E_{id} \) via encrypting its real ID, that is \( E_{id} \leftarrow \text{Encrypt}(ID, fpk) \). RA calculates \( \bar{E}_{id} = \text{Clip}(H(E_{id})) \) and checks the registration status of the vehicle. Specifically, with the assistance of smart contracts (defining the Search function), RA retrieves the \( \bar{E}_{id} \) in the blockchain state database. The return value determines the current status of a given vehicle. Details are as follows:

\[
\text{Search}(\bar{E}_{id}) = \begin{cases} 
\text{Cer}, & \text{If the vehicle has been registered} \\
\text{revoke}, & \text{If the vehicle has been revoked} \\
\text{null}, & \text{If the vehicle hasn’t been registered}
\end{cases}
\]

If \( \text{Search}(\bar{E}_{id}) = \text{null} \), RA signs the vehicle’s \( E_{id} \) with the secret key \( fsk \) and obtains \( \sigma_{fsk} \leftarrow \text{Sig}(E_{id}, fsk) \). RA conducts the certificate \( \text{Cer} = (E_{id}, \text{registration}, npk, T_{\text{expired}}, \sigma_{fsk}) \). Finally, RA stores \( \bar{E}_{id} \) and \( \text{Cer} \) in the blockchain state database in the form of key-value pairs.

It is worth noting that, during the \( E_{id} \) issuance, \( E_{id} \) issued for vehicles should contain enough redundancy so that other vehicles cannot reuse it. RA should retain the vehicle’s confidential material and its local secret key confidentiality. Meanwhile, a vehicle’s secret key is reserved in the tamper-proof device. Thus, we assume that there is no privacy disclosure and security attack risk in this phase.

Algorithm 1 Registration Procedure

**Input:** \( R \)

**Output:** \( \text{Cer} \)

1. \( R = (ID, \text{registration}, npk, \sigma_{nsk}) \)
2. \( E_{id} \leftarrow \text{Encrypt}(ID, fpk) \)
3. **if** \( \text{Search}(\bar{E}_{id}) = \text{null} \) **then** */check the \( E_{id} \) hasn’t been registered previously */
4. \( \sigma_{fsk} \leftarrow \text{Sig}(E_{id}, fsk) \)
5. \( \text{Cer} = (E_{id}, \text{registration}, npk, T_{\text{expired}}, \sigma_{fsk}) \); */RA generates the certificate for vehicle*/
6. \( \text{Mapping}(\bar{E}_{id} \rightarrow \text{Cer}) \); */RA maps \( \bar{E}_{id} \) to the certificate \( \text{Cer} \)*/
7. \( P_i \leftarrow (\omega^i, u_i, g^{\omega^i}, g^{u_i}, \pi_i) \); */RA calculates commitment parameters for authentication*/
8. \( U_{key} = (\rho_i, \mu_i, \omega^i) \); */RA calculates parameter \( U_{key} \) for proof update*/
9. **end if**

2) Phase II: Update of Global Commitment: RA is responsible for updating the global commitment. RA assigns a pre-generated pair of \( (\omega^i, \mu_i) \) to registered vehicles. Afterwards, based on the vehicle, RA can utilize the clipping function to calculate \( \bar{E}_{id} = \text{Clip}(H(E_{id})) \), \( npk = \text{Clip}(H(npk)) \), and corresponding value \( \mathcal{E} \) and \( \mathcal{P} \). Further, RA recalculates \( u_i = (\mathcal{E} \parallel \mathcal{P}) \), where \( \mathcal{R}P \) is the value corresponding to \( \text{Clip}(r \cdot npk) \) and \( r \) is a random number. We assume that the value of \( u \) changes by \( \delta \) after recalculating. Therefore, based on the description in V-A, the RA can calculate the newest polynomial \( \Psi(X) = \Psi(X) + \delta \cdot L_i(X) \) and the updated of commitment \( C \) is \( C' = C \cdot (l_i)^\delta \), where \( l_i = g^{L_i(\tau)} \) (calculation of polynomial \( \Psi(X) \) update is described in Appendix C). The newest \( C \) is broadcast via RSUs.

3) Phase III: Calculation of Commitment Parameters: After updating the global commitment, RA calculates the parameters related to KZG commitment, which mainly includes \( P_i \) and \( U_{key} \), where \( P_i = (\omega^i, u_i, g^{\omega^i}, g^{u_i}, \pi_i) \) and \( U_{key} = (\rho_i, \mu_i, \omega^i) \). Based on the description in III-C, RA can generate the commitment proof \( \pi_i \) corresponding to \( (\omega^i, u_i) \). In addition, RA calculates the update key \( U_{key} \) to assist the vehicle in updating proof \( \pi_i \). Specifically, RA computes \( \rho_i = g^{(r \cdot \omega^i)} \), where \( D(X) = X^{\mu_i} - 1 \) (the derivation of the \( D(X) \) has been shown in Appendix D).

After the registration process, RA sends \( \{\text{Cer}, P_i, U_{key}\} \) to the vehicle.

C. Message Authentication

In this subsection, we demonstrate the authentication process of vehicle-to-untrusted entities as well as the authentication process of vehicle-to-trusted/semi-trusted entities. Without loss of generality, we assume participants in the authentication process are the vehicle \( A \) (prover) and other entity \( B \) (verifier).

1) The Vehicle-to-Untrusted Entity: As shown in Fig. 3, we present the authentication process of the vehicle-to-untrusted entity. The specific details are as follows:

- **Messages generation**

  (1) The vehicle \( A \) selects a random number \( r \) and calculates \( npk \cdot r = \text{Clip}(H(npk \cdot r)) \) and corresponding value \( \mathcal{R}P \). Besides, vehicle \( A \) broadcasts \( npk_A \cdot r \). Then, the vehicle generates parameters related to message and identity as follows:

\[
E_A = \text{nsk}_A \cdot fpk \cdot r \oplus \bar{E}_{id}[t] \quad (2)
\]

\[
M_A = H \left[ m || t || E_A || (r_B \cdot npk_B \cdot r \cdot \text{nsk}_A) \right] \oplus r_B \cdot npk_B \cdot r \cdot \text{nsk}_A \quad (3)
\]
Fig. 3. The authentication process between the vehicle and untrusted entity.

In the equation (3), $r_B \cdot npk_B$ is broadcast by the verifier.

(2) The vehicle $A$ calculates the related parameters of the zero-knowledge proof:

$$
\pi_{au}^A = (\pi_u)^{r_{au}} = (g^{(\omega_u)})^r \\
\eta_{au}^A = g^{u_A} \cdot (\pi_{au}^A)^\emptyset \cdot g^{-1}
$$

(4)

In the equation (4), $u_A' = (-\emptyset \cdot P + u_A)$. Therefore, this algorithm outputs the authentication tuple $\sigma = \{\pi_{au}^A, \eta_{au}^A, \omega_{au}, g^{u_A}, (\pi_{au}^A)^\emptyset, E_A, M_A, m, t\}$.

- **Authentication**

  (1) The verifier receives the authentication tuple $\sigma$. It checks the freshness of the timestamp $t$ and calculates:

  $$
  H[m||t||E_A||(r_B \cdot nsk_B \cdot r \cdot npk_A)] + r_B \cdot nsk_B \cdot r \cdot npk_A \\ = M_A
  $$

  If the equation (5) holds, it means the message $m$ and key pairs $(nsk_A, npk_A)$ are valid. Afterwards, the verifier will verify that the message was sent by an authorized vehicle.

  (2) The verifier first calculates auxiliary parameter $g_{au} = g^{\emptyset P} \cdot g_{au}$, where $\emptyset P$ is the value corresponding to the $Clip(H(r \cdot npk_A))$. Therefore, the verifier can accomplish the authentication process by verifying:

  $$
  e(C \cdot g_{au}^{\pi_{au}^A}, g^{u_A}) = e(\pi_{au}^A, g^r \cdot g^{-1})
  $$

  If the equation (6) holds, it means the source of message $m$ is a legally authorized vehicle.

2) The Vehicle-to-Trusted/semi-Trusted Institutions: As shown in Fig. 4 and Fig. 5, we respectively present the single and batch authentication process of the vehicle-to-trusted/semi-trusted institutions. The specific details are as follows:

- **Messages generation**

  (1) The vehicle $A$ generates verification parameters related to message and identity as follows:

  $$
  E_A = nsk_A \cdot fpk \cdot r \oplus (E_{id}|t)
  $$

  (7)

  Fig. 4. The authentication process between the vehicle and trusted/semi-trusted institution.

  Assume $I = \{0, 1, 2, \ldots, k\}$

  $\sigma_i = \{P_i^1, E_i, M_i, m_i, t_i\}$

  $\sigma_f = \{P_i^2, E_i, M_i, m_i, t_i\}$

  $\sigma_g = \{P_i^3, E_i, M_i, m_i, t_i\}$

  $\sigma_i = \{P_i^4, E_i, M_i, m_i, t_i\}$

- **Single authentication**

  (1) The verifier receives the authentication tuple $\sigma$. It checks the freshness of the timestamp $t$ and verifies:

  $$
  H[m||t||E_A||(r_B \cdot nsk_B \cdot r \cdot npk_A)] + r_B \cdot nsk_B \cdot r \cdot npk_A \\ = M_A
  $$

  (9)

  If the equation (9) holds, it means the message $m$ and key pairs $(nsk_A, npk_A)$ are valid. Afterwards, the verifier would check that the message is sent by an authorized vehicle.
\( e(C / g^{\sigma_i}, g) = e(\pi, g^{g_i^\sigma}) \) (10)

If the equation (10) holds, it means the source of message \( m \) is a legally authorized vehicle.

**Batch authentication**

The verifier can simultaneously verify messages sent by multiple authorized vehicles. Batch verification is achieved via aggregating proofs with the assistance of Partial fraction decomposition (see Appendix D). Specifically, the verifier can receive \((\omega^j, u^j, \pi^j)\) \( j \in [0, n] \). According to previous work, given the point \((\omega^j, u^j)\), we have \( \Psi(\omega^j) = u^j = \exists q_t(X), \Psi(X) - u^j = q_t(X)(X - \omega^j) \). The \( \pi_t \) is a commitment to \( q_t(X) = \Psi(X) - u^j \). Similarly, \( \pi_t \) is the commitment to \( q_t(X) \), that is \( \pi_t = g^{q_t(\tau)} \). Thus, the proof aggregation is as follows:

1. The verifier calculates \( R(X) \) and \( D_I(X) \). \( R(X) \) is computed via Lagrange interpolation and satisfies \( R(\omega^i) = u^i, i \in I \). \( D_I(X) = \prod_{i \in I}(X - \omega^i) \).
2. Then, the verifier calculates \( c_I \equiv 1 / D_I(\omega^i) \) and \( \pi_I = \sum_{i \in I} c_I q_t(X) \). Thus, the verifier can obtain \( q_t(X) = [\Psi(X) - R(X)] / [D_I(X)] \).
3. Finally, the verifier can verify the multiple evaluation proof (according to III-C):

\[
e(C / g^{R_{\tau}(\pi)}, g) = e(\pi, g^{D_{I}(\tau)})
\]

Therefore, the verifier can verify multiple authorized vehicles simultaneously.

**D. Certificate Update**

In IoVs scenario, vehicles usually update their certificates under the following circumstances: 1) The identity owner initiates the update process for its own good; 2) When the vehicle intends to stay in IoVs after its certificate expires. A vehicle updates its certificate with the following update strategy. Firstly, a vehicle must prove that it has been registered before. Secondly, the updated vehicle must prove that it is the holder of the previous certificate. The certificate update is divided into two phases. The Phase I is mainly for certificate and authentication parameters updates. The Phase II is mainly for the global commitment update based on the newest certificate.

1) **Phase I: Certificate Update**: Before updating the certificate, the vehicle must prove ownership of the certificate, which can prevent attackers from obtaining \( E_{id} \) and \( npk \) through illegal means and updating legitimate certificates arbitrarily. The vehicle first signs the new public key with its old secret key, \( \sigma_i^{ver} = \text{Sig}(E_{id}\|npk_{t+1}, nsk) \), then signs the \( E_{id} \) with the new secret key \( \sigma_i^{nsk} = \text{Sig}(E_{id}, nsk_{t+1}) \), which are utilized for RTAs to verify whether the update is initiated by the owner of the previous certificate or not. The certificate update procedure is shown in Algorithm 2.

**Step 2-1**: The vehicle generates a new key pair \((npk_{t+1}, nsk_{t+1})\). Afterwards, the vehicle sends the tuple \( U \) to the RSU located within its communication range.

\[
U = (E_{id}, update, npk_i, npk_{t+1}, T_{expired}, \sigma_i^{ver}, \sigma_i^{nsk})
\] (12)

**Algorithm 2: Certificate Update Procedure**

**Input:** \( U \)

**Output:** \( Cer' \),

1. \( U \leftarrow (E_{id}, update, npk_i, npk_{t+1}, T_{expired}, \sigma_i^{ver}, \sigma_i^{nsk}) \)
2. if Search(\( E_{id} \)) == Cer then
3. if Check(npk_i, \( \sigma_i^{ver}, E_{id}\|npk_{t+1} \)) == 1
& Check(npk_{t+1}, \sigma_i^{nsk}, E_{id} == 1)

\& Check(npk_{t+1}, \sigma_i^{nsk}, E_{id} == 1)

**Step 2-2**: Firstly, smart contract leverages the function Search to check the vehicle’s current status. Only when the return value is Cer, the process continues.

**Step 2-3**: The RTA verifies whether the previous certificate’s owner initiates the update. RTA needs to verify two signatures \( \sigma_i^{ver} \) and \( \sigma_i^{nsk} \) and check whether \( npk_i \) is consistent with \( npk \) stored in the certificate Cer.

\[
\text{Check}(npk_i, \sigma_i^{ver}, E_{id}\|npk_{t+1}) \equiv 1 \quad (13)
\]

\[
\text{Check}(npk_{t+1}, \sigma_i^{nsk}, E_{id}) \equiv 1 \quad (14)
\]

\[
npk_i \equiv npk \quad (15)
\]

**Step 2-4**: The RTA calculates newest parameters for authentication. Specifically, here is mainly the update of the three parameters, \( u^i \), \( g^{u^i} \), and \( \pi_t \), respectively.

For \( u^i \), we have \( E_{id} = \text{Clip}(H(E_{id})) \) and \( npk_{t+1} = \text{Clip}(H(npk_{t+1})) \), and obtain corresponding value \( E_I \) and \( \pi_{i+1} \) and perform the following calculation:

\[
u_i^{t+1} = (E_{id}\|\pi_{i+1})
\] (16)

Therefore, the parameter \( g^{u^i} \) is also updated.

When the status of the vehicle’s certificate state changes, assuming that is a change \( \delta U \) to \( u_i \), so the newest \( \Psi'(X) = \Psi(X) + \delta \cdot L'(X) \) (see Appendix C). Thus, the proof \( \pi_t \) also needs to be updated. In order to express the proof update scheme explicitly, we assume that \( i \) and \( j \) are two vehicle nodes. We will discuss both cases of local proof update \((i = j) \)
and other proof update \((i \neq j)\) below. It should be noted that with the assistance of RSUs, vehicles performing certificate updates first need to broadcast parameters \(\{\delta, U_{key}\}\) in the blockchain network.

First, we discuss the case of \(i = j\). When one vehicle’s certificate state changes, the proof of this vehicle would be \(\pi_i \rightarrow \pi_{i+1}^{\delta}\). We have known \(\pi_i\) is a KZG commitment to \(q_i(X) = \Psi(X) - u_i\). Therefore, corresponding to the \(\Psi'(X)\), the newest \(q_i'(X)\) can be calculated:

\[
q_i'(X) = \frac{\Psi'(X) - (u_i + \delta)}{X - \omega^j} = \frac{(\Psi(X) + \delta \cdot L_i(X)) - u_i - \delta}{X - \omega^j} = \frac{\Psi(X) - u_i}{X - \omega^j} + \delta \cdot \frac{L_i(X) - 1}{X - \omega^j} = q_i(X) + \delta \cdot \left(\frac{L_i(X) - 1}{X - \omega^j}\right)
\]

(17)

Therefore, the vehicle also need a KZG commitment to \(\frac{L_i(X)}{X - \omega^j}\). From \(U_{key}\), the vehicle can obtain \(\mu_i = g^\beta, \beta = \frac{L_i(t)}{X - \omega^j}\) and compute the updated proof: \(\pi_{i+1}^{\delta+1} = \pi_i \cdot (\mu_i)^\delta\), which is uncomplicated to implement. It is worth noting that with the update of the certificate, \(U_{key}\) and \(\delta\) will be uploaded to the blockchain as an update record \(Rec\) via a transaction.

Second, we discuss the case of \(i \neq j\) and define this update situation as other proof updates. When other vehicle’s certificate (e.g. vehicle \(j\)) state changes, the proof of vehicle \(i\) would be \(\pi_i \rightarrow \pi_{i+1}^{\delta}\). Fig. 6 shows the update process in this case. We compute the updated quotient polynomial \(q_i'(X)\):

\[
q_i'(X) = \frac{\Psi'(X) - u_i}{X - \omega^j} = \frac{(\Psi(X) + \delta \cdot L_i(X)) - u_i}{X - \omega^j} = \frac{\Psi(X) - u_i}{X - \omega^j} + \delta \cdot \frac{L_i(X)}{X - \omega^j}
\]

(18)

Obviously, the vehicle \(i\) need a KZG commitment to \(\frac{L_i(X)}{X - \omega^j}\). We set \(\Phi_i,j(X) = \frac{L_i(X)}{X - \omega^j}\) and rewrite:

\[
\Phi_i,j(X) = \frac{L_i(X)}{X - \omega^j} = \frac{D(X)}{D'(\omega^j)(X - \omega^j)(X - \omega^j)}
\]

(19)

In the equation \(19\), \(D(X) = \prod_{i \in [0, n]}(X - \omega^i) = X^n - 1, D'(\omega^j) = n\omega^{-j}\). Further, we set \(\Upsilon_i,j(X) = \frac{D(X)}{(X - \omega^j)(X - \omega^j)}\). Therefore, the vehicle \(i\) only needs a KZG commitment \(\xi_i,j\) to \(\Upsilon_i,j(X)\). Based on the partial fraction decomposition (see Appendix D), we obtain:

\[
\Upsilon_i,j(X) = d_1 \cdot D(X)/(X - \omega^j) + d_2 \cdot D(X)/(X - \omega^j)
\]

(20)

In the equation \(20\), \(d_1 = \frac{1}{\omega^j - \omega^j}, d_2 = \frac{1}{\omega^j - \omega^j}\). Further, with the assistance of RSU, the vehicle \(i\) obtains \(\rho_j\) and \(\omega^j\) from \(U_{key}\) and compute \(\xi_i,j = \Omega_i,j^d \rho_j^2\). Finally, we can compute \(p_{i,j} = (\xi_i,j)^{D'(\omega^j)}\) and update the proof as \(\pi_{i+1}^{\delta+1} = \pi_i \cdot (p_{i,j})^\delta\).

Obviously, the calculation of proof \(\pi_{i+1}^{\delta+1}\) here follows the case of \(i = j\).

**Step 2-5:** The Mapping function maps the encrypted identity \(\overline{E}_{id}\) to the new certificate \(\overline{Cer}'\). Now, the vehicle stores \(\overline{Cer}'\) in its local OBU.

**Step 2-6:** If \(Search\) returns \(pend\), then RSU will inquire RTA to validate the vehicle’s identity.

1) The vehicle lost its secret key; 2) The vehicle’s certificate expired; 3) The vehicle wants to leave the network. Before a vehicle revokes its certificate, a verification process should be included to prevent the vehicle from being revoked maliciously. The vehicle’s certificate can be revoked with the master key in our scheme. The revocation message \(\nu\) is initiated upon receiving a request from a given vehicle. If the vehicle loses its secret key, it must revoke the certificate using the master key and signature.

The certificate revocation procedure is shown in Algorithm 3. Unlike the existing schemes requiring a CRL list, our scheme leverages a smart contract to check the current status of the vehicle’s certificate. Besides, the mechanism described in step 3-3 prevents the adversary from revoking certificates maliciously.

**Algorithm 3 Certificate Revocation Procedure**

**Input:** \(\nu\)

**Output:** \(\perp\)

1: \(\nu \leftarrow (E_{id}, revoke, npk, \sigma_{revoke}, T_{expired})\)
2: if \(Search(E_{id}) == Cer\) then
3: if \(Check(npk, \sigma_{revoke}, E_{id}, T_{expired}) == 1 \&\& \checkmark\)
4: \(\text{Mapping}(E_{id} \rightarrow revoke)\)
5: else return \(\perp\);
6: end if
7: end if

**Step 3-1:** The vehicle intending to revoke its certificate sends a tuple \(\nu\) to RSU.

\[
\nu = (E_{id}, npk, revoke, \sigma_{revoke}, T_{expired})
\]

(21)
The vehicle signs $E_{id}$ and $T_{expired}$ with its old secret key, 
\[ \sigma^\text{rev} = \text{sig}(E_{id}||T_{expired}, \text{nsk}_A) \]  
(22)

**Step 3-2:** First, RA leverages the function `Search` to check whether the vehicle’s $E_{id}$ is registered or revoked before.

**Step 3-3:** RSU verifies whether the revocation is initiated by the owner of the previous certificate by validating the signature $\sigma^\text{rev}$.

\[ \text{Check}(\text{nsk}_A, \sigma^\text{rev}, E_{id}||T_{expired}) \equiv 1 \]  
(23)

**Step 3-4:** RA utilizes the smart contract to update the mapping $E_{id} \rightarrow \text{revoke}$ to the blockchain state database.

Besides, vehicles may be forcibly revoked by RA and RTA due to malicious behaviour. In this case, RTA and RA can revoke vehicles by implementing the following steps (assuming the malicious vehicle $A$):

1. RTA captures the parameter $E_A$ in the authentication tuple $\sigma$. RTA sends parameters $\{E_A, \text{npk}_A \cdot r\}$ to the RA. Parameter $\text{npk}_A \cdot r$ is broadcast by the vehicle $A$.
2. RA obtains field $(E_{id}||t)$ by calculating $(E_{id}||t) = \text{npk}_A \cdot r \cdot fsk \oplus E_A$, $fsk$ is the secret key of RA.
3. Based on the parameter $E_{id}$, RA retrieves the corresponding certificate in the blockchain state database and re-maps the field “revoke” to $E_{id}$.

It should be noted that after the certificate status changes, the statuses of the vehicle’s proof $\pi$ in IoVs and global commitment $C$ will also change accordingly. Then, the method of changing the status of $\pi$ and $C$ here is the same as the certificate update; just set the change $\delta$ to $u$. Further, the change of $C$ will cause the revoked vehicle to fail the verification of equations (6) (10).

### VI. Security Analysis

We assume that the adversary in our model cannot break the standard cryptographic primitives. Firstly, we conduct a zero-knowledge proof analysis and correctness analysis of the proposed scheme. Secondly, we describe the privacy protection mechanism in our scheme. Lastly, we prove that our protocol can resist various attacks in IoVs.

#### A. Zero-Knowledge Proof

In our paper, the proposed protocol utilizes zero-knowledge proof to achieve authentication without exposing the additional private information of the prover. For clarity, we set the vehicle $A$ as prover and the verifier $B$.

**Theorem 1:** The proposed scheme is a non-interactive zero-knowledge proof on prover’s knowledge $\{\omega, u, \text{nsk}\}$.

From the following lemmas, the proof of the theorem shows that the proposed protocol satisfies the basic property: 1) completeness, 2) soundness, and 3) zero-knowledge.

**Lemma 1:** The proposed protocol satisfies completeness, i.e., the prover can pass the verifier’s validation as long as the prover has the correct knowledge. Moreover, the prover can convince the verifier with a sufficient probability.

**Proof:** We assume the vehicle $A$ is an honest prover in possession of the parameters $\{\omega^A, u_A, \text{nsk}_A\}$. Therefore, as described in V-C1, the vehicle $A$ can calculate the correct parameters using formula (2)(3)(4):

\[ \{E_A, M_A, \pi^A_{au}, g^A_{au}, g_{au}^A, g^r_{au}, g^{r-1}\} \]. Based on these parameters, the verifier can validate that vehicle $A$ has correct secret $\{\omega^A, u_A, \text{nsk}_A\}$ through formula (5)(6).

The correctness analysis of the equations involved in the certification process is detailed in section VI-B.

**Lemma 2:** The proposed protocol satisfies soundness, i.e., the prover cannot pass the verifier’s validation as long as the prover does not have the correct knowledge. Moreover, the probability that the prover successfully deceives the verifier can be ignored.

**Proof:** We assume one client $i$ participates in the validation process with the incorrect possession of the parameters $\{\omega^A, u_A, \text{nsk}_A\}$. Therefore, this client can calculate parameters based on the proposed protocol, including

\[ \{E_A, M_A, \pi^A_{au}, g^A_{au}, g_{au}^A, g^r_{au}, g^{r-1}\}. \]

Based on the description in V-C1, we first discuss whether the prover $A$ can pass the authentication with the parameters $\{E_A, M_A\}$. The prover calculates these two parameters based on the incorrect secret key $\text{nsk}_A$:

\[ E_A = \text{nsk}_A \cdot f pk \cdot r \oplus (E_{id}||t) \]  
(24)

\[ M_A = H[m||t||E_{AF}||E_{AF}||E_{AF}||E_{AF}||E_{AF}] \oplus r_B \cdot \text{npk}_B \cdot r \cdot \text{nsk}_A \]  
(25)

However, the verifier cannot validate the correctness of $M_A$ with the equation (5):

\[ H[m||t||E_{AF}||E_{AF}||E_{AF}||E_{AF}||E_{AF}] \oplus r_B \cdot \text{npk}_B \cdot r \cdot \text{nsk}_A \neq E_{AF} \]  
(26)

Next, we discuss whether the prover $A$ can pass the authentication with the parameters $\{\pi^A_{au}, g^A_{au}, g_{au}^A, g^r_{au}, g^{r-1}\}$. It should be emphasized that the global commitment $C$ is generated based on the total correct parameters $\{\{\omega^A, u_i\}|1 \leq i \leq n\}$, which does not include parameter $\{\omega^A, u_A\}$. With the assistance of global commitment $C$, the verifier cannot validate the correctness of $\{\pi^A_{au}, g^A_{au}, g_{au}^A, g^r_{au}, g^{r-1}\}$ with the equation (6), i.e.:

\[ e(C/g^A_{au}, g^r_{au}, g^{r-1}) \neq e(\pi^A_{au}, g^r/g_{au}^A) \]  
(27)

Therefore, the prover $A$ cannot pass the verifier’s validation as long as it doesn’t have the correct knowledge.

**Lemma 3:** The proposed protocol satisfies zero-knowledge, i.e., the prover can prove that it has the corresponding correct knowledge without disclosing any extra information about the knowledge.

**Proof:** In our paper, secret $\{\omega^A, u_A, \text{nsk}_A\}$ is masked in parameters $\{E_A, M_A, \pi^A_{au}, g^A_{au}, g_{au}^A, g^r_{au}, g^{r-1}\}$. It should be noted that this paper assumes mathematical assumptions cannot be broken. Thus, the verifier or adversary is unable to derive secret from existing knowledge (e.g. based on the ECDHP), the verifier $B$ cannot calculate $r_B \cdot \text{npk}_B \cdot r \cdot \text{nsk}_A$, even if the verifier already knows knowledge $r_B \cdot \text{npk}_B$ and $r \cdot \text{nsk}_A$. Obviously, the verifier does not learn any additional knowledge from
parameters \( \{E_A, M_A, \pi_{au}^A, g^\omega_{au}, g^\theta_{au}, g^{-1}\} \) and the prover’s secret knowledge \( \omega^A, u_A, nsk_A \) is not leaked.

**B. Correctness Analysis**

As described in III-C, the verifier can calculate equation
\[
\Psi(\tau) - u_A = q(\tau)(\tau - ao)
\]
and verify whether the vehicle has a legitimate authorization parameter \( ao, u \).

- **The vehicle-to-untrusted entity:** As described in V-C1, the verifier can calculate the equation
\[
e(C / g^{\omega_{au}}_{au}, g^{-1}) = e(\pi_{au}^A, g^\tau / g^{\omega_{au}}_{au})
\]
when the authentication process occurred between vehicles and the untrusted entity. The calculation of this equation can be derived as follows:
\[
es(C / g^{\omega_{au}}_{au}, g^{-1}) = e(\pi_{au}^A, g^\tau / g^{\omega_{au}}_{au})
\]
\[
\quad \Leftrightarrow e(g^{\Psi(\tau) - u_A - q(\tau)(\tau - 1)\omega^A}, g^{-1})
\]
\[
\quad \Leftrightarrow e(g^{\Psi(\tau) - u_A - q(\tau)(\tau - 1)\omega^A}, g^{-1})
\]
\[
\quad \Leftrightarrow e(g, g)^{\Psi(\tau) - u_A - q(\tau)(\tau - 1)\omega^A}(\tau - 1)
\]
\[
\quad \Leftrightarrow \Psi(\tau) - u_A = q(\tau)(\tau - \omega^A) + q(\tau)(\tau - 1)\omega^A
\]
\[
\quad \Leftrightarrow \Psi(\tau) - u_A = q(\tau)(\tau - \omega^A)
\]
(28)

where auxiliary parameter \( g^{\omega_{au}}_{au} = g^{\Psi(\tau) - u_A - q(\tau)(\tau - 1)\omega^A} \).

- **The vehicle-to-trusted/semi-trusted institutions:** As described in V-C2, the verifier can calculate the equation
\[
e(C / g^{\omega_{au}}_{au}, g) = e(\pi, g^\tau / g^{\omega_{au}}_{au})
\]
when the authentication process occurred between vehicles and the trusted entity. The calculation of this equation can be derived as follows:
\[
e(C / g^{\omega_{au}}_{au}, g) = e(\pi, g^\tau / g^{\omega_{au}}_{au})
\]
\[
\quad \Leftrightarrow e(g^{\Psi(\tau) - u_A - q(\tau)(\tau - 1)\omega^A}, g)
\]
\[
\quad \Leftrightarrow e(g, g)^{\Psi(\tau) - u_A - q(\tau)(\tau - 1)\omega^A}(\tau - 1)
\]
\[
\quad \Leftrightarrow \Psi(\tau) - u_A = q(\tau)(\tau - \omega^A)
\]
(29)

Besides, the verifier can implement batch authentication by calculating
\[
e(C / g^{\omega_{au}}_{au}, g) = e(\pi, g^\tau / g^{\omega_{au}}_{au})
\]
(28)

C. **Privacy Analysis**

It is worthy of note that our scheme predominantly intends to preserve the privacy of the identity data. Once the vehicle’s identity data is disclosed, adversaries can combine the owners’ information with their vehicles. During V2V authentication, we explore every step taken to showcase the privacy-preserving property of our scheme.

1) **Anonymous:**

**Proof:** Taking vehicle \( A \) as an example, the property of anonymity is achieved via three ways in the authentication process: 1) the vehicle’s real identity is masked by \( E_{id}, \tilde{E}_{id} \) and signature parameter \( E_A \) (see Section V-C1 for details), 2) the vehicle’s public key \( npk_A \) is randomized by random number \( \tau \), 3) the authentication is implemented with non-interactive zero-knowledge proof. For clarity, we can give the following proof sketches for anonymity.

With the assistance of signature parameter \( \{E_A, M_A\} \), the attackers attempting to obtain the real identity of vehicle \( A \) must perform three steps. Firstly, attackers calculate field \( \tilde{E}_{id} \). That is \( (\tilde{E}_{id})||\tau = E_A \oplus nsk_A \cdot fpk \cdot r \) or \( (\tilde{E}_{id})||\tau = E_A \oplus npk_A \cdot fsk \cdot r \). Secondly, attackers must retrieve the vehicle’s certificate \( Cer \) based on the field \( E_{id} \) from the blockchain state database. Thirdly, attackers decrypt the vehicle’s true identity using RA’s secret key \( fsk \).

However, our scheme supports that it is hard for attackers to perform the above steps. Firstly, based on the ECDLP assumption, attackers cannot derive RA’s secret key \( fsk \) from public key \( fpk \). Therefore, attackers cannot obtain field \( E_{id} \) by calculating \( E_A \oplus npk_A \cdot fsk \cdot r \). Besides, attackers cannot decrypt the vehicle’s true identity without RA’s secret key \( fsk \). Secondly, based on the ECDH assumption, attackers cannot calculate \( nsk_A \cdot fpk \cdot r \), even if the parameters \( npk_A \cdot r = nsk_A \cdot fsk \cdot r \) and \( fsk = fpk \cdot P \) are already known. Therefore, attackers cannot obtain field \( E_{id} \) by calculating \( E_A \oplus nsk_A \cdot fpk \cdot r \). Finally, our scheme is constructed based on the consortium blockchain, which is maintained by RA and RTA. In our scheme, RTA and RA are trusted institutions that are not compromised. Therefore, attackers do not have access to the blockchain state database, and our scheme can guarantee excellent anonymity.

2) **Unlinkability:**

**Proof:** The proposed scheme provides unlinkability, i.e., no verifier can tell whether two authentication tuples were derived from the same vehicle. For the authentication tuple \( \sigma = \{\pi_{au}^A, g^\omega_{au}, g^\theta_{au}, g^{-1}, E_A, M_A, m, t\} \), each element is generated with random number \( \tau \). Therefore, different signature parameter tuples exhibit various characteristics. Obviously, the verifier cannot link two or more signature information to the same vehicle. Similarly, based on the anonymity feature in the proposed scheme, the verifier cannot derive evaluation proof \( \pi \) or other identity information. Thus, our scheme can guarantee unlinkability in the authentication process.

3) **Traceability:**

**Proof:** In our scheme, RA can track and reveal malicious vehicles. Taking an authentication tuple and the master secret key \( fsk \), which outputs the \( E_{id} || | t \) by the following equation:
\[
\tilde{E}_{id} || | t = npk_A \cdot r_A \cdot fsk \oplus E_A
\]
(30)
The RA can compute \( E_{id} \) with its secret key to identify the vehicle with the assistance of blockchain. Consequently, based on the analysis of anyone cannot reveal the vehicle’s real identity without knowing the RA’s secret key.

D. **Resistance to Attacks**

We analyze several common attacks in IoVs and illustrate that our scheme can resist these attacks.

1) **Impersonation Attack:**

**Proof:** According to the registration algorithm, the certification posted to the blockchain is composed of a tuple \( \mathcal{R} = (E_{id}, \text{register}, npk, T_{\text{expired}}, \sigma^{fsk}) \). If an adversary \( \mathcal{A} \) tries to forge a certificate, there are several routines to follow. First, \( \mathcal{A} \) attempts to figure out the real identity of a benign vehicle. However, after thoroughly examining the vehicle, the
RA encrypts and reserves its real identity. Besides, RA has a sufficient safety level. Therefore, there is no way for \( A \) to access the real identity. Second, RA encrypts and reserves its real identity. Besides, RA has a tamper-proof device installed in the OBU is responsible for storing confidential material. Therefore, adversary \( A \) is unable to steal vehicle’s \( E_{id} \). Besides, based on the generation process of \( \overline{E}_{id} \triangleq Cli(p(H(E_{id}))) \), adversary \( A \) cannot derive \( E_{id} \) from \( \overline{E}_{id} \). Third, the RSUs usually request the RA to perform a validity check during the registration. The fake identity and \( \overline{E}_{id} \) cannot pass this validity check. Thus, we can conclude that our scheme can resist impersonation attacks.

2) Replay Attack:

Proof: An adversary \( A \) launches an attack by replaying the critical identity information between the participants involved in the authentication process. The global polynomial commitment \( C = g^{\Psi(X)} \) is dynamic following each vehicle certificate’s state in our scheme. Therefore, to realize the correspondence between the proof and the global certificate status, the proof (e.g., \( \pi^A \)) should be updated before the vehicle implements the authentication process, as shown in V-D. It is worth noting that the information required for the proof update is hidden (e.g., \( \omega^A \)). Therefore, an attacker cannot utilize the old proof to pass the verification based on the latest commitment and cannot update the old proof. On the other hand, the tuple \( \pi = \{\pi_{au}^A, g_{au}^{\omega^A}, g_{au}^{\omega^{-1}}, E_A, M_A, m, t\} \) sent by authentication launcher includes timestamp \( t \), therefore, parameter \( \pi \) changes with \( t \), which prevents the outdated message and resist the replay attack.

3) Modification Attack:

Proof: The modification of the signature \( \pi = \{\pi_{au}^A, g_{au}^{\omega^A}, g_{au}^{\omega^{-1}}, E_A, M_A, m, t\} \) can be divided into two categories: 1) the attacker tamper with parameters \( \{E_A, M_A, m, t\} \); 2) the attacker tamper with parameters \( \{\pi_{au}^A, g_{au}^{\omega^A}, g_{au}^{\omega^{-1}}\} \).

Firstly, if the attacker modifies any information in parameter set \( \{E_A, M_A, m, t\} \), the verifier can calculate \( H(m[||E_A][||rB \cdot nskB \cdot r \cdot npkA] \oplus rB \cdot nskB \cdot r \cdot npkA \neq M_A) \). More concretely, based on the ECDHP assumption, attackers cannot calculate \( nskA \cdot r \cdot nskB \cdot rB \), even if the parameters \( npkA \cdot r = nskA \cdot r \cdot P \) and \( fpk = fsk \cdot P \) are already known. Therefore, attackers cannot forge reasonable parameters \( M_A \) and \( H(m[||E_A][||rB \cdot npkB \cdot r \cdot nskA] \) \) (or \( H(m[||E_A][||rB \cdot nskB \cdot r \cdot npkA] \) ) and to deceive the verifier via modifying parameters in \( \{E_A, M_A, m, t\} \).

Secondly, if the attacker modifies any information in parameter set \( \{\pi_{au}^A, g_{au}^{\omega^A}, g_{au}^{\omega^{-1}}\} \), the verifier can calculate \( e(C/g_{au}^{\omega}, g^{\omega^{-1}}) \neq e(C_{au}^A, g^{\omega^{-1}}/g_{au}^{\omega}) \). Therefore, any modifications can cause the message to fail validation. Therefore, our scheme can resist the modification attack.

4) Forgery Attack:

Proof: Consider there is a benign vehicle (i.e., possessing an authorized certificate) maliciously forging a certificate \( Cer_F \) based on its valid key pair and trying to pass a verification equation successfully. To pass the verification, the vehicle needs to generate parameters \( (u_F, g^{u_F}, \pi_F) \) to construct zero-knowledge proof corresponding to the forged certificate. However, in the authentication process, the global commitment \( C \) is generated by RTA, which is based on the total authorized certificate, that is:

\[
C = \prod_{i=0}^{d} (g^{\Psi_i}) = g^{\sum_{i=0}^{d} \Psi_i t^i} = g^{\Psi(t)}
\]

where \( \Psi(X) \) is a polynomial of degree \( d \leq n \) with coefficients \( s_0, s_1, \ldots, s_d \) in \( \mathbb{Z}_q \). \( \Psi(X) \) is generated by Lagrange interpolation based on \( (\omega^i, u_i) \in e(0, n) \). Obviously, the computation set of global commitment \( C \) does not contain \( u_F \), and the vehicle cannot change the state of \( C \). Thus, the vehicle cannot pass the verification of global commitment \( C \) using parameters \( (u_F, g^{u_F}, \pi_F) \), i.e.:

\[
C \neq C_F \Rightarrow e(C/g_{au}^{\omega_F}, g^{\omega_F^{-1}}) \neq e(C_{au}^F, g^{\omega / g_{au}^{\omega}})
\]

VII. PERFORMANCE EVALUATION

In this section, we evaluate the performance of our scheme through constructing comprehensive analyses and simulations. More concretely, we first introduce the complexity of crucial algorithms and operations involved in the proposed model. Afterwards, we implement computation cost analysis and comparison and measure the overhead regarding certificate operations. Finally, we further measure the communication overhead incurred by the scheme and compare our proposed scheme with existing state-of-the-art methods in terms of overall authentication latency to highlight the advantages regarding practical feasibility.

A. Experiment Settings

We simulate PBAG on a machine with an AMD Ryzen 7 5800H with Radeon Graphics CPU @ 3.20GHz CPU and 16GB RAM. The prototype is implemented in Hyperledger Fabric v2.0.0 with Raft consensus, and its chaincodes are developed in Golang [56].

In the blockchain settings, furthermore, we construct one specific channel and set up two organizations representing RA and RTAs. RA and RTAs are utilized as endorsing peers. Following the prescribed endorsement policy, one transaction will be successfully committed when more than \( 2n + 1 \) signatures from the endorsing peers are valid. It is worth noting that the vehicle in our scheme would not be the endorsing peer and have no rights to access the channel due to its mobility and limited communication resources. In the simulation, we construct four peer nodes belonging to two organizations, three order nodes, and one client node in one channel.

In our scheme, we implement bilinear pair programs with the MIRACL library. In order to estimate the time consumption of certificate operations, we run the smart contracts with various numbers of vehicles in the network scaled at 10, 10^2, 10^3, and 10^4. In the experiment, we combine the vehicle’s plate with the Vehicle Identification Number (VIN) to form an 18-digit number as their real ID parameter. With this ID, we apply the crypto series library in Golang to generate \( E_{id} \) and master key pairs.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
B. Complexity of Key Algorithms and Operations

This subsection illustrates the complexity of key algorithms and operations involved in the proposed scheme. Meanwhile, the overhead analysis of the proposed scheme in terms of computation, communication, and overall authentication will be shown in subsection VII-E and VII-D, which also includes the comparison with other schemes.

- Lagrange polynomial interpolation: In our proposal, this algorithm is extremely crucial for the system initialization phase (subsection V-A) and batch authentication operation (subsection V-C). More concretely, in the system initialization phase, the protocol needs to compute the polynomial function \( \Psi(X) = \sum_{i \in [0,n]} \mathcal{L}_i(X)u_i \) of degree \( < n \) based on \( n \) pairs of parameters \( \{(\omega_i, u_i)|0 \leq i < n\} \). The time complexity is \( O(n \log^2 n) \). Similarly, when the verifier calculates the polynomial function \( R(X) \) based on \( |I| \) received parameters involved in batch authentication operations, the time complexity is \( O(|I| \log^2 |I|) \).

- Calculation and aggregation of proofs: Essentially, the proof \( \pi_I \) is just a KZG evaluation proof for \( \Psi(\omega_i) \). Thus, in the batch authentication operation for \( |I| \) received parameters, except calculating \( R(X) \) using Lagrange polynomial interpolation, the protocol also requires the calculation of other crucial parameters: a KZG batch proof \( \pi_I \) for all \( \Psi(\omega_i) \) \( i \in I \). Here, it should be noted that all proofs \( \pi_i \) can be computed with the assistance of the Feist-Khovratovich (FK) technique [57] in \( O(n \log n) \) time, which allows the protocol faster to calculate the intermediate parameters \( D_I(X) \) and \( c_i \) required for the aggregation proof (the relevant computation complexity has been illustrated in Appendix D) [47]. Finally, the batch proof \( \pi_I \) can be computed with an \( O(|I|) \)-sized multi-exponentiation.

- Global commitment update: In the protocol, the global commitment \( C \) can be computed via \( C = g^{\Psi(\tau)} \). During global commitment update process, we can set \( l_i = g^{\mathcal{L}_i(\tau)} \) [58], thus \( C \) can be regarded as a calculation of complexity \( O(n) \): \( C = \prod_{i \in [0,n]} l_i^{u_i} \). When the value of \( u_i \) changes by \( \delta \), the update of global commitment \( C \) can be calculated via \( C' = C \cdot (l_i)^\delta \), obviously, which is the commitment to the updated polynomial \( \Psi(X)' = \Psi(X) + \delta \cdot \mathcal{L}_i(X) \).

C. Certificate Operation

We carry out simulations to estimate the time consumption of the certificate operations in PBAG with smart contracts, as presented in Fig. 7. Specifically, Fig. 7(a) and (b) show the time incurred for calling the issuance and updating contracts, respectively. Fig. 7(c) shows the time duration for calling revocation with different certificate scales. All certificate operations involve validating whether the vehicle has a legal
signature and a legitimate key pair. Therefore, we simulate this verification process additionally, and the related results are shown in Fig. 7(d). We observe that the data presented in each subfigure did not alter significantly, and the data is distributed in a specific interval. According to our simulation, the average time consumption for the contract issuance is 6.95ms, while verification is 0.39ms. This experimental result is due to the issuance and update process needed to regenerate the certificate, including the key generation process, which is relatively time-consuming. Fig. 7 indicates that our scheme can lead to small time fluctuations when running the smart contract multiple times in the Hyperledger Fabric. Our scheme is technically feasible according to the time overhead.

Besides, the authentication is accomplished based on zero-knowledge proof in our scheme. There are three crucial elements involved in the zero-knowledge proof protocol: certificate, proof, and global commitment. Proof and global commitment are utilized to implement authentication by the verifier through bilinear pairing, and both need to be updated following the state of the certificate (elaborated in V-D).

- **Update of global commitment** The update of global commitment is divided into two phases: the verification of evaluation proof $\pi$ and the calculation of the newest global commitment. We simulate the update process with smart contracts, as shown in Fig. 8(a). We can observe that the time duration curves of both phases have an upward trend as the number of certificate updates increases. Based on the batch-enabled mechanism, when there are 50 certificate update records, the verification time is 21.956ms, and the calculation of the newest global commitment can be accomplished within 35.906ms.

- **Update of evaluation proof** For vehicles, the situation of proof update can be divided into two categories, local proof update and other proof update. We implement the simulation with smart contracts to estimate the update performance, as shown in Fig. 8(b). For local proof update, we can observe that the time overhead is basically constant at 0.279ms. In addition, the time duration curve shows an apparent upward trend for other proof updates as the number of proof update records increases. More concretely, the time duration is about 34.95ms when there are 50 proof update records.

**D. Computation Cost Analysis and Comparison**

Here, we analyze the computation cost of our proposed PBAG scheme. Meanwhile, we compare PBAG with the state-of-the-art schemes, including EADA [17], RCoM [18], P2BA [19]. First, we calculate the execution time of some basic cryptographic operations. For more accurate and convincing measurement results, we run the cryptographic operations 1000 times and take the average values as the final execution time. The execution time of the necessary cryptographic parameters and the authentication overhead of some related schemes are shown below:

- $T_{bp}$ is the time required to perform a bilinear pairing, $T_{bp} \approx 3.803$ms
- $T_h$ is the time required to implement the hash function, $T_h \approx 0.001$ms
- $T_{sm}$ is the time required to implement one scalar multiplication under $G_1$, $T_{sm} \approx 0.141$ms
- $T_{ex}$ is time required to implement one exponentiation in $G_T$, $T_{ex} \approx 0.138$ms.

To evaluate the authentication process accurately, we calculate the computation cost of the various phases in the authentication operation. First, we calculate the computation cost for generating one authentication message. For one vehicle, the computation process mainly needs to perform three exponentiation operations and one hash operation. Thus, the message-generating time per vehicle is $3T_{ex} + T_h \approx 0.415$ms. Afterwards, we evaluate the performance of the RTA/RSU’s message verification phase. The infrastructure needs to perform two bilinear pairing operations, two scalar multiplication operations, and one hash operation when verifying a single message. Thus, the authentication time per vehicle is $2T_{bp} + 2T_{sm} + T_h \approx 7.889$ms. Finally, our scheme supports batch message verification, significantly facilitating authentication performance. We can compute that it only takes about 36.171ms to verify 100 messages simultaneously. Our scheme
can satisfy the requirement latency to support applications in the IoVs scenario.

Table III gives the comparison of computation overhead among recently proposed schemes approaches, including EADA, RCoM, and P2BA. The primary operations involved in the related schemes are $T_{bp}$, $T_{sm}$, $T_{ex}$, and $T_{mtp}$. For authentication with the EADA scheme, the cryptographic operations require four bilinear pairing operations, three scalar multiplication under $G_1$, and four MapToPoint hash operations of the bilinear pairing, respectively. Thus, the total computation cost is $4T_{bp} + 3T_{sm} + 4T_{mtp}$ approx. 16.003ms. For authentication with the RCoM scheme, the cryptographic operations require performing $2s + 6$ bilinear pairing operations, seven exponentiation under $G_T$, respectively. Thus, the total computation cost is $(2s + 6)T_{bp} + 7T_{ex} \approx 38.996$ ms when the number of equivalence classes is assumed to be 5. For authentication with the P2BA scheme, the cryptographic operations require performing four bilinear pairing operations, ten scalar multiplication under $G_1$, and ten exponentiation under $G_T$ of the bilinear pairing, respectively. Thus, the total computation cost is $4T_{bp} + 10T_{sm} + 10T_{ex} \approx 18.002ms$. When addressing one message authentication, our scheme outperforms above at least 50.70%, 79.77%, and 56.18%, respectively, for EADA, RCoM, and P2BA.

In order to highlight the efficiency of the proposed PBAG, we compute the comparison times of $n$ authentication in the proposed scheme with three state-of-the-art schemes, EADA, RCoM, and P2BA, as shown in Fig. 9. It should be noted that $T_T$ is the time to calculate using Lagrangian interpolation, which is 0.0179ms, 0.0811ms, 0.2237ms, 0.2893ms, 0.3672ms, respectively, for the 20, 40, 60, 80, 100, 120 vehicles. More concretely, when there are $n$ authentication requests, EADA scheme requires $3n + 1 + \kappa$ bilinear pairing operations, $n + 2$ scalar multiplication operations, and $3 + n + \kappa$ MapToPoint hash operations. The total computation costs is $(3\kappa + 1)T_{bp} + (n + 2)T_{sm} + (3 + n + \kappa)T_{mtp}$. Similarly, for $n$ vehicles authentication, the RCoM scheme involves $(2s + 6)n$ bilinear pairing operations and $7n$ exponentiation operations in $G_T$. The total computation costs is $(2s + 6)nT_{bp} + 7nT_{ex}$. P2BA needs to perform four bilinear pairing operations, $6n - 1$ scalar multiplication operations. The total computation cost is $4T_{bp} + (6n - 1)T_{sm}$. In comparison, our scheme only requires two bilinear pairing operations, $n$ scalar multiplication operations, $n + 2$ exponentiation operations, one lagrangian interpolation operation, and $n$ hash operations. The total computation cost is $2T_{bp} + n T_{sm} + (n + 2)T_{ex} + T_L + nT_h$.

Therefore, when there are 100 authentication requests, the computation cost is 1180.25ms, 3899.6ms, and 99.671ms, respectively, for EADA ($\kappa = 4$), RCoM ($s = 2$), and P2BA. Compared with the other three schemes, the time delay of the PBAG scheme is 36.171ms, which outperforms above at least 96.97%, 99.07%, 63.7%, respectively, for EADA, RCoM, and P2BA. Therefore, the proposed scheme is more efficient than the comparison schemes.

E. Authentication Latency and Comparison

This subsection analyses the authentication latency based on the more complex authentication scenario that vehicle to RTA/RSUs. We define authentication latency containing three main parts: the computation time of the vehicle $T_{gen}$, the message propagation time $T_P$, and the computation time of the verifier $T_{ver}$. Therefore, the total authentication latency is $T_{gen} + T_{P} + T_{ver}$ involved in the process from the vehicle generating a message to when the message is successfully verified. Among them, $T_{gen}$ and $T_{ver}$ have been elaborated in the previous subsection. Therefore, we will analyze the calculation of $T_P$ detailedly in the following contents. Besides, for accurately calculating the $T_p$, we construct the communication scenario in Network Simulator v2 based on the prespecified parameters listed in Table IV.

First of all, we analyze the message size in our proposal and compared schemes. As described in subsection III-A, the bilinear pairing operation is defined as $e: G_1 \times G_2 \rightarrow G_T$, and $G_1$, $G_2$, and $G_T$ are multiplicative cyclic group. The size of the elements in these groups is $64 \times 2 = 128$ bytes. Besides, as shown in III-D, based on the points on the elliptic curve $E$ and the point $\theta$ at infinity, the elliptic curve addition group $G$ is set, and the element size of the group is $20 \times 2 = 40$ bytes. The timestamp size, entity identification field, hash value, and

| Scheme     | Generate one message | For one message authentication | For $n$ messages authentication |
|------------|----------------------|-------------------------------|----------------------------------|
| RCoM       | $7T_{bp} + 4T_{ex}$   | $(2s + 6)T_{bp} + 7T_{ex}$    | $(2s + 6)nT_{bp} + 7nT_{ex}$    |
| EADA       | $2T_{bp} + 3T_{sm} + 3T_{mtp}$ | $4T_{bp} + 3T_{sm} + 4T_{mtp}$ | $(3\kappa + 1)T_{bp} + (n + 2)T_{sm} + (3 + n + \kappa)T_{mtp}$ |
| P2BA       | $2T_{bp} + 11T_{sm} + 12T_{ex}$ | $2T_{bp} + 2T_{sm} + T_h$ | $2T_{bp} + nT_{sm} + (n + 2)T_{ex} + T_L + nT_h$ |

Table III: Comparison for Computational Costs

Fig. 9. Comparison of computation cost.
the element in integer field $Z_p^n$ are 4 bytes, 20 bytes, 20 bytes, and 20 bytes, respectively.

In RCoM, the vehicle $V_j$ needs to upload the triple $W = (SA_i, SA_j, V_j, vsk_{j,1}, vsk_{j,2}, t_i, t_j, T_{d}, \theta, Time)$ and ciphertext $U$ to the cloud server (verifier). Here, $U = (u_1, u_2, u_3, u_4)$, and the computation analyses present that $u_3$ is hash value, $\{u_1, u_2, u_4\} \in G_1$. In addition, $\{SA_i, SA_j, V_j\}$ are the entity identification fields, and partial secret key $\{vsk_{j,1}, vsk_{j,2}\} \in G_1$. Parameters $\{t_i, t_j, Time\}$ denote the time stamp. $\theta_i$ is a token with 20 bytes issued by the administrative RSU, and $T_d$ represents the valid period of the token. Therefore, the size of the uploaded authentication message in RCoM is $5|G_1| + 20 \ast 5 + 4 \ast 4 = 756$ bytes. In comparison, EADA requires to perform multiple interactions between the vehicle and edge nodes (verifier). Specifically, in the Auth-I phase, vehicle $\hat{V}$ needs to send the request $req = (ID_\hat{V}, Y, R1, R2, TS)$ to the leader edge node. For these parameters, $ID_\hat{V}$ is a unique code used to identify individual vehicles (e.g., VIN code with 17 bytes), and $\{Y, R1, R2\} \in G_1$. TS represents a timestamp. Thus, the size of this request message is $3|G_1| + 17 + 4 = 141$ bytes. Afterwards, all edge nodes in set $T$ (assume $|T| = \kappa$, including the leader edge node) need to provide feedback parameter $C$ with 128 bytes and pairs $(V_k, W_k)_{k \in T}$ to the vehicle $\hat{V}$, where $(V_k, W_k)_{k \in T} \in G_1$. Thus, the size of this feedback message is $2\kappa|G_1| + 128 = 128 + 80k$ bytes. Finally, the vehicle $\hat{V}$ execute the Auth-II phase. The vehicle $\hat{V}$ send the request $req = (ID_\hat{V}, EXP, Y_2, R_3, R_4, TS)$ to the edge node. Here, $EXP$ is a composed string with 128 bytes. Parameters $\{Y_2, R_3, R_4\} \in G_1$ and $TS$ is a timestamp. Thus, the size of this request message is $3|G_1| + 128 + 17 + 4 = 269$ bytes. In summary, Therefore, the size of the authentication message in EADA is $(6 + 2\kappa)|G_1| + 128 \ast 2 + 17 \ast 2 + 4 \ast 2 = 538 + 80k$ bytes. In P2BA, the uploaded authentication message contains the blinded certificate $\tilde{Cert}$, message signature $\sigma$, and timestamp $T_{stamp}$. Briefly, the blinded certificate $\tilde{Cert}$ includes three elements in $G_1$; signature $\sigma$ consists of eight $G_T$ group elements and four integer field elements. Therefore, the size of the uploaded authentication message is $3|G_1| + 8|G_T| + 4|Z_p^n| + 4 = 1620$ bytes. In our proposal, the uploaded authentication message contains parameters $\{o, u, g^o, g^u, \pi, E_A, M_A, m, t\}$, and the size is $3|G_1| + |G_T| + 2|Z_p^n| + 20 + 4 = 448$ bytes. The comparison of message size is presented in Table V.

In addition, based on these analyses of authentication message size, we construct the communication scenario in Network Simulator v2 to observe the communication time. As shown in Fig. 10, the communication time of P2BA is the highest, based on the analysis of the previous contexts, which is due to it having a large number of parameter transmission requirements. Meanwhile, the growth rate of the communication time curve of this scheme is also the highest. In comparison, obviously, our proposal can complete the authentication process with the lowest communication time, and the curve growth rate is also lower than other schemes, which provides a significant advantage for achieving large-scale authentication.

Based on the crucial measurement of time $T_{gen}$, $T_p$ and $T_{ver}$, we can obtain the final authentication latency. However, it should be noted that some traffic variables (such as vehicle speed and density) may still affect the final authentication latency, which requires further experimentation and analysis. Therefore, in order to comprehensively explore the factors affecting authentication latency, we first simulate vehicles driving at different speeds in Network Simulator v2. We assume that there is a certain number of vehicles (e.g., 20) are distributed over different lanes, and the speed of vehicles in each lane is roughly distributed in 5-30 m/s. As shown in Fig. 11, for one scheme and a certain number of vehicles, the latency data did not alter significantly during the simulation process. This is good evidence that speed only slightly affects latency. Afterwards, we observe the curve of the relationship...
between the average authentication latency and the traffic density through the simulation presented in Fig. 12. For various schemes, we can observe that the latency curve has an upward trend with the increase in vehicle density. Among that, although the RCoM has a certain advantage over other comparison schemes in terms of communication cost (as shown in Fig. 10), it has an unaffordable computational cost, which leads to the highest authentication latency. The authentication latency of RCoM and EADA is similar, but the numerical performance of the curve is still high, including the growth rate. In comparison, the curve matching our scheme has the smallest slope, which means the average latency of our proposal increases slowly compared with the other approaches. Besides, our scheme can authenticate 80 vehicles within 400ms in the simulation, which is 82.18% better than P2BA, 82.38% better than EADA, and 93.22% better than RCoM, respectively.

VIII. CONCLUSION AND FUTURE WORK

Research on addressing the authentication latency issue caused by verifiers having to connect with the blockchain network in advance. We have proposed a privacy-preserving blockchain-based authentication protocol (PBAG). Without connecting to the blockchain, the verifier can efficiently
authenticate the user to be an authorized member by utilizing the global commitment through bilinear pairing. Moreover, our scheme can prevent vehicles with invalid certificates from accomplishing the authentication, thus avoiding the time consumption of checking the CRL. In terms of performance evaluation, we have conducted an exhaustive theoretical analysis and simulation to indicate the privacy properties and performance of the proposed scheme. In future work, we plan to implement our scheme in an automated way and facilitate it to interface with real-world data. Besides, based on the proposed scheme, how to achieve batch authentication in V2V is also one of our focuses.

APPENDIX A
CALCULATE THE DERIVATIVE VALUE OF THE FUNCTION $D(X)$ WHEN $X = \omega^j$

Let $D(X) = \prod_{i \in [0,n]} (X - \omega^j) = X^n - 1$ and $\varphi(X) = D(X)/(X - \omega^j)$, then:

$$D'(X) = \sum_{j \in [0,n]} D(X)/(X - \omega^j)$$

$\Rightarrow D'(\omega^j) = \prod_{j \in [0,i]} (\omega^j - \omega^j). \prod_{r \in (i,n)} (\omega^j - \omega^j) = \varphi(\omega^j)$

Thus, rewriting $\varphi(X)$:

$$\varphi(X) = \frac{X^n - 1}{X - \omega} = (\omega^j)^0 X^{n-1} + (\omega^j)^1 X^n - 2 + (\omega^j)^2 X^{n-3} + \ldots + (\omega^j)^n X^0$$

$$= \sum_{y=0}^{n-1} (\omega^j)^y X^{n-1-y}$$

Finally, evaluating $D'(\omega^j)$ at $X = \omega^j$:

$$D'(\omega^j) = \varphi(\omega^j) = (\omega^j)^0 \omega^j(\omega^j - 1) + (\omega^j)^1 \omega^j(\omega^j - 2) + \ldots (\omega^j)^n \omega^j(\omega^j - n)$$

$$= \sum_{y=0}^{n-1} (\omega^j)^y \omega^j(\omega^j - n) = n\omega^j(\omega^j - n) = n\omega^j - i$$

Thus, we can compute $D'(\omega^j) = n\omega^j - i$.

APPENDIX B
MULTIPLE EVALUATIONS PROVING AND PARAMETER $c_i$ DERIVATION PROCESS

The first is the multiple evaluation proving process. We have known $c_i = 1/D_i(\omega^j)$, $\pi_i = \prod_{i \in [0,n]} \pi_i$ and $q_1(X) = \Psi(X) - R(X)$. For the equation $e(C/g^{R(t)}, g) = e(\pi_i, g^{D_i(t)})$, we provide the following calculation details:

$$e(C/g^{R(t)}, g) = e(\pi_i, g^{D_i(t)})$$

$\Leftrightarrow e(C/g^{R(t)}, g) = e(\prod_{i \in I} \pi_i, g^{D_i(t)})$

According to the construction of multiple evaluation proving. We will show the process of finding $\pi_i$ that satisfies the conditions of the multiple evaluation proving:

$$q_1(X) = \frac{\Psi(X) - R(X)}{D_i(X)}$$

$= \frac{\Psi(X)}{D_i(X)} - \frac{R(X)}{D_i(X)}$

$= \frac{\Psi(X)}{D_i(X)} - \frac{1}{\sum_{i \in I} D_i(\omega^j)(X - \omega^j)}$

$= \frac{\Psi(X)}{D_i(X)} - \sum_{i \in I} D_i(\omega^j)(X - \omega^j)$

$= \frac{1}{\sum_{i \in I} D_i(\omega^j)} \cdot q_i(X)$

Thus, we can compute $c_i = 1/D_i(\omega^j)$ and $\pi_i = \prod_{i \in I} \pi_i$.

APPENDIX C
UPDATE OF FUNCTION $\Psi(X)$

Without loss of generality, we assume the polynomial $\Psi(X)$ is generated for given points $(x_i, y_i) \in [0,n]$ and satisfies $\Psi(x_i) = y_i$. Based on the Lagrangian interpolation, we can obtain:

$$\Psi(X) = \sum_{i \in [0,n]} L_i(X)y_i$$

where $L_i(X) = \prod_{j \in [0,n], j \neq i} \frac{X - x_j}{x_i - x_j}$. We assume the change in $y_i$ is $\delta$, then:

$$\Psi'(X) = \sum_{k \in [0,i]} L_k(X)y_k$$

$+ L_i(X)(y_i + \delta) + \sum_{k \in [0,k]} L_k(X)y_k$

$= \sum_{k \in [0,i]} L_k(X)y_k + L_i(X)y_i + L_i(X)\delta$

$= \Psi(X) + L_i(X)\delta$

Finally, we can obtain the newest $\Psi'(X) = \Psi(X) + \delta \cdot L_i(X)$.

APPENDIX D
PARTIAL FRACTION DECOMPOSITION

We define $D_i(X) = \prod_{i \in I} (X - x_i)$, $I \subset [0, n]$ and rewriting the Lagrange polynomial for interpolating $\Psi(X)$ given all
where $D'_i(x) = \sum_{j \in I} D_i(x)/(x - x_j)$. Noting that $D'_i$ is the derivative of $D_i(X)$ \[54\]. Therefore, we can rewrite the Lagrange interpolation as:

$$
\Psi_i(X) = \sum_{i \in I} \xi_i \prod_{j \in I, j \neq i} \frac{X - x_j}{x_i - x_j} = \frac{D_i(X)}{\prod_{j \in I, j \neq i} (x_i - x_j)} \quad (41)
$$

From the properties of Lagrangian polynomials, we know that $\Psi(x_i) = y_i$ for the point $(x_i, y_i)$. Therefore, for the $\Psi(X) = 1$, this implies:

$$
\Psi(|x| = 1) = \frac{1}{D_i(X)} = \frac{1}{\prod_{i \in I} (x_i - x)} = \sum_{i \in I} c_i \cdot \frac{1}{x_i - x}, c_i = \frac{1}{D'_i(x)} \quad (42)
$$

we can compute the $D_i(X)$ in $O(|I|^2 \log^2|I|)$ time, $D'_i(X)$ in $O(|I|)$ time and evaluated at all $x_i$ in $O(|I| \log^2|I|)$ time \[54\]. Thus, all $c_i$’s can be computed in $O(|I| \log^2|I|)$ time. When the $I = [0, n]$, we have:

$$
D_i(X) = \prod_{i \in [0, n]} (X - x_i) = X^n - 1 \quad \text{and} \quad D'_i(x) = nx^{i-1} \quad (44)
$$

The derivation details of the formula are in Appendix A. Therefore, we can compute any $c_i$ in $O(1)$ time.

REFERENCES

[1] H. Tan, W. Zheng, and P. Vijayakumar, “Secure and efficient authenticated key management scheme for UAV-assisted infrastructure-less IoVs,” IEEE Trans. Intell. Transp. Syst., vol. 24, no. 6, pp. 6389–6400, Jun. 2023.

[2] J. Ma, T. Li, J. Cui, Z. Ying, and J. Cheng, “Attribute-based secure announcement sharing among vehicles using blockchain,” IEEE Internet Things J., vol. 8, no. 13, pp. 10873–10883, Jul. 2021.

[3] X. Zhou, M. Luo, P. Vijayakumar, C. Peng, and D. He, “Efficient certificateless conditional privacy-preserving authentication for VANETs,” IEEE Trans. Veh. Technol., vol. 71, no. 7, pp. 7863–7875, Jul. 2022.

[4] X. Li et al., “PAPU: Pseudonym swap with provable unlinkability based on differential privacy in VANETs,” IEEE Internet Things J., vol. 7, no. 12, pp. 11789–11802, Dec. 2020.

[5] C. Adams and S. Lloyd, Understanding Public-key Infrastructure: Concepts, Standards, and Deployment Considerations. Indianapolis, IN, USA: Sams, 1999.

[6] C. Zhang, R. Lu, X. Lin, P-H. Ho, and X. Shen, “An efficient identity-based batch verification scheme for vehicular sensor networks,” in Proc. 27th Conf. Comput. Commun., 2008, pp. 246–250.

[7] L. Wu, J. Fan, Y. Xie, J. Wang, and Q. Liu, “Efficient location-based conditional privacy-preserving authentication scheme for vehicle ad hoc networks,” Int. J. Distrib. Sensor Netw., vol. 13, no. 3, Mar. 2017, Art. no. 155014771700089.

[8] Y. Liu, L. Wang, and H-H. Chen, “Message authentication using proxy vehicles in vehicular ad hoc networks,” IEEE Trans. Veh. Technol., vol. 64, no. 8, pp. 3697–3710, Aug. 2015.

[9] C. D. Jung, C. Sur, Y. Park, and K-H. Rhee, “A robust conditional privacy-preserving authentication protocol in Vanet,” in Proc. Int. Conf. Secure Privacy Mobile Inf. Commun. Syst., 2009, pp. 35–45.

[10] Z. Lu, Q. Wang, G. Qu, H. Zhang, and Z. Liu, “A blockchain-based privacy-preserving authentication scheme for VANETs,” IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 27, no. 12, pp. 2792–2801, Dec. 2019.

[11] V. De Maio, R. B. Uriarte, and I. Brandic, “Energy and profit-aware proof-of-stake offloading in blockchain-based VANETs,” in Proc. 12th IEEE/ACM Int. Conf. Utility Cloud Comput., Dec. 2019, pp. 177–186.

[12] X. Li, T. Jing, R. Li, H. Li, X. Wang, and D. Shen, “BDORA: Blockchain and decentralized identifiers assisted secure registration and authentication for VANETs,” IEEE Internet Things J., vol. 10, no. 14, pp. 12140–12155, Jul. 2023.

[13] C. Lin, D. He, X. Huang, N. Kumar, and K-K. R. Choo, “BCCPA: A blockchain-based conditional privacy-preserving authentication protocol for vehicular Ad Hoc networks,” IEEE Trans. Intell. Transp. Syst., vol. 22, no. 12, pp. 7408–7420, Dec. 2021.

[14] C. Lin, X. Huang, and D. He, “EBCPA: Efficient blockchain-based conditional privacy-preserving authentication protocol for VANETs,” IEEE Trans. Dependable Secur. Comput., vol. 20, no. 3, pp. 1818–1832, May 2023.

[15] X. He, X. Niu, Y. Wang, L. Xiong, Z. Jiang, and C. Gong, “A hierarchical blockchain-assisted conditional privacy-preserving authentication scheme for vehicular ad hoc networks,” Sensors, vol. 22, no. 6, p. 2299, Mar. 2022.

[16] A. Vangala, A. K. Das, A. Mitra, S. K. Das, and Y. Park, “Blockchain-enabled authenticated key agreement scheme for mobile vehicles-assisted precision agricultural IoT networks,” IEEE Trans. Inf. Forensics Security, vol. 18, pp. 904–919, 2023, doi: 10.1109/TIFS.2022.3231121.

[17] A. Yang, J. Weng, K. Yang, C. Huang, and X. Shen, “Delegating authentication to edge: A decentralized authentication architecture for vehicular networks,” IEEE Trans. Intell. Transp. Syst., vol. 23, no. 2, pp. 1284–1298, Feb. 2022.

[18] Y. Wang, Y. Ding, Q. Wu, Y. Wei, B. Qin, and H. Wang, “Privacy-preserving cloud-based road condition monitoring with source authentication in VANETs,” IEEE Trans. Inf. Forensics Security, vol. 14, no. 7, pp. 1779–1790, Jul. 2019.

[19] X. Feng, Q. Shi, Q. Xie, and L. Wang, “P2BA: A privacy-preserving protocol with batch authentication against semi-trusted RSUs in vehicular ad hoc networks,” IEEE Trans. Inf. Forensics Security, vol. 16, pp. 3888–3899, 2021.

[20] J. S. Alshudduki, Z. G. Al-Mekhlafi, and B. A. Mohammed, “A lightweight authentication with privacy-preserving scheme for vehicular ad hoc networks based on elliptic curve cryptography,” IEEE Access, vol. 9, pp. 15633–15642, 2021.

[21] 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. Depend. Secur. Comput., vol. 18, no. 2, pp. 722–735, Mar. 2021.

[22] F. Wei, S. Zeadally, P. Vijayakumar, N. Kumar, and D. He, “An intelligent terminal based privacy-preserving multi-modal implicit authentication protocol for Internet of connected vehicles,” IEEE Trans. Intell. Transp. Syst., vol. 22, no. 7, pp. 3939–3951, Jul. 2021.

[23] J. Zhang, Q. Zhang, X. Lu, and Y. Gan, “A novel privacy-preserving authentication protocol using bilinear pairings for the VANET environment,” Wireless Commun. Mobile Comput., vol. 2021, pp. 1–13, Jun. 2021.

[24] S. K. A. Theodore, K. R. Gandhi, and V. Palanisamy, “A novel lightweight authentication and privacy-preserving protocol for vehicular ad hoc networks,” Complex Intell. Syst., vol. 7, no. 7, pp. 2981–2991, Jun. 2023.

[25] F. Alataf 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.

[26] I. Ali, Y. Chen, N. Ullah, M. Afzal, and W. He, “Bilinear pairing-based hybrid signcryption for secure heterogeneous vehicular communications,” IEEE Trans. Veh. Technol., vol. 70, no. 6, pp. 5974–5989, Jun. 2021.

[27] Q. Feng, D. He, S. Zeadally, and K. Liang, “BPAS: Blockchain-assisted privacy-preserving authentication system for vehicular ad hoc networks,” IEEE Trans. Ind. Inform., vol. 16, no. 6, pp. 4146–4155, Jun. 2020.

[28] F. Wang and Y. Liu, “SEMA: Secure and efficient message authentication protocol for VANETs,” IEEE J. Sel. Top. Signal Process., vol. 15, no. 1, pp. 846–855, Mar. 2021.

[29] U. Bansal, J. Kar, I. Ali, and K. Naik, “ID-CEPPA: Identity-based computationally efficient privacy-preserving authentication scheme for vehicle-to-vehicle communications,” J. Syst. Archit., vol. 123, Feb. 2022, Art. no. 102387.
[30] Z. Liu, M. Yuan, Y. Ding, and B. Wang, “Efficient small-batch verification and identification scheme with invalid signatures in VANETs,” *IEEE Trans. Veh. Technol.*, vol. 70, no. 12, pp. 18236–18246, Dec. 2021.

[31] P. Vijayakumar, M. Azees, S. A. Kozlov, and J. J. P. C. Rodrigues, “An anonymous batch authentication and key exchange protocols for 6G enabled VANETs,” *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 1630–1638, Feb. 2022.

[32] C. Maurya and V. K. Chaurasia, “Efficient anonymous batch authentication scheme with conditional privacy in the Internet of Vehicles (IoV) applications,” *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 9, pp. 9670–9683, Sep. 2023.

[33] Y. Yang, H. Yu, Y. Zhao, H. Jiang, and Y. Ren, “Towards load balancing in IoV system: A vehicle-assisted batch verification scheme,” *Veh. Commun.*, vol. 44, Dec. 2023, Art. no. 100683.

[34] J. Liu et al., “CPAHP: Conditional privacy-preserving authentication scheme with hierarchical pseudonyms for 5G-enabled IoV,” *IEEE Trans. Veh. Technol.*, vol. 72, no. 2, pp. 8892–8940, Jul. 2023.

[35] H. Tan, W. Zheng, P. Vijayakumar, K. Sakurai, and N. Kumar, “An efficient vehicle-assisted aggregate authentication scheme for infrastructure-less vehicular networks,” *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 12, pp. 15590–15600, Dec. 2023.

[36] S. Nakamoto. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[37] C. Wang, J. Shen, J.-F. Lai, and J. Liu, “B-TSCA: Blockchain assisted trustworthiness scalable computation for V2I authentication in VANETs,” *IEEE Trans. Emerg. Topics Comput.*, vol. 9, no. 3, pp. 1386–1396, Jul. 2021.

[38] S. Son, J. Lee, Y. Park, Y. Park, and A. K. Das, “Design of blockchain-based lightweight V2I handover authentication protocol for VANET,” *IEEE Trans. Netw. Sci. Eng.*, vol. 9, no. 3, pp. 1346–1358, May 2022.

[39] J. Zhang, H. Fang, H. Zhong, J. Cui, and D. He, “Blockchain-assisted privacy-preserving traffic route management scheme for fog-based vehicular ad-hoc networks,” *IEEE Trans. Netw. Service Manage.*, vol. 20, no. 3, pp. 2854–2868, Jul. 2023.

[40] K. Xue, X. Luo, Y. Ma, J. Li, J. Liu, and D. S. Wei, “A distributed authentication scheme based on smart contract for roaming service in mobile vehicular networks,” *IEEE Trans. Veh. Technol.*, vol. 71, no. 5, pp. 5284–5297, May 2022.

[41] J. Zhang, Y. Jiang, J. Cui, D. He, I. Bolodurina, and H. Zhong, “DBCPA: Dual blockchain-assisted conditional privacy-preserving authentication framework and protocol for vehicular ad hoc networks,” *IEEE Trans. Mobile Comput.*, vol. 23, no. 2, pp. 1–15, Sep. 2022.

[42] X. Zhou, D. He, M. K. Khan, W. Wu, and K. R. Choo, “An efficient blockchain-based conditional privacy-preserving authentication protocol for VANETs,” *IEEE Trans. Veh. Technol.*, vol. 72, no. 1, pp. 81–92, Jan. 2023.

[43] A. Joux, “A one round protocol for tripartite Diffie–Hellman,” in *Proc. Int. Algorithmic Number Theory Symp.*, 2000, pp. 385–393.

[44] A. J. Menezes, T. Okamoto, and S. A. Vanstone, “Reducing elliptic curve logarithms to logarithms in a finite field,” in *Proc. Int. Algorithmic Number Theory Symp.*, 2001.

[45] J.-P. Berrut and L. N. Trefethen, “Barycentric Lagrange interpolation,” *SIAM Rev.*, vol. 46, no. 3, pp. 501–517, Jan. 2004.

[46] A. Kate, G. M. Zaverucha, and I. Goldberg, “Constant-size commitments to polynomials and their applications,” in *Proc. 16th Int. Conf. Theory Appl. Cryptol. Inf. Security*, Dec. 2010, pp. 177–194.

[47] A. Tomescu, I. Abraham, V. Buterin, J. Drake, D. Feist, and D. Khovratovich, “Aggregatable subvector commitments for stateless cryptocurrencies,” in *Proc. Int. Conf. Secur. Cryptogr. Netw.*, 2020, pp. 45–64.

[48] S. Bowe, A. Gabizon, and J. Miers, “Scalable multi-party computation zk-snark parameters in the random beacon model,” Cryptol. ePrint Arch., Tech. Paper 2017/1050, 2017. [Online]. Available: https://eprint.iacr.org/2017/1050

[49] P. Kushwaha, “Towards the equivalence of Diffie–Hellman problem and discrete logarithm problem for important elliptic curves used in practice,” in *Proc. ISEA Asia Secur. Privacy (ISEASP)*, Jan. 2017, pp. 163–169.

[50] A. Menezes, *Evaluation of Security Level of Cryptography: The Elliptic Curve Discrete Logarithm Problem*, vol. 14. Waterloo, Canada: University of Waterloo, 2001.

[51] J. B. Kenney, “Dedicated short-range communications (DSRC) standards in the united states,” *Proc. IEEE*, vol. 99, no. 7, pp. 1162–1182, 2011, doi: 10.1109/JPROC.2011.2132790.

[52] L. Yang, J. Ma, Z. Liu, and R. Zheng, “A trusted authentication scheme for wireless networks using direct anonymous attestation,” in *Proc. 4th Int. Conf. Intell. Netw. Collaborative Syst.*, Sep. 2012, pp. 279–285.

[53] D. Johnson, A. Menezes, and S. Vanstone, “The elliptic curve digital signature algorithm (ECDSA),” *Int. J. Inf. Security*, vol. 1, no. 1, pp. 36–63, 2001.

[54] J. Von Zur Gathen and J. Gerhard, *Modern Computer Algebra*. Cambridge, U.K.: Cambridge Univ. Press, 2013.

[55] IEEE Standard for Wireless Access in Vehicular Environments–Security Services for Applications and Management Messages, Standard IEEE 1609.2, 2016.

[56] (2023). Golang. [Online]. Available: https://golang.org/

[57] D. Feist and D. Khovratovich. (2020). Fast Amortized Kate Proofs. [Online]. Available: https://github.com/khovratovich/Kate

[58] J. Camenisch, M. Dubovitskaya, K. Haralambiev, and M. Kohlweiss, “Composable and modular anonymous credentials: Definitions and practical constructions,” in *Proc. Int. Conf. Theory Appl. Cryptol. Inf. Secur.*, 2015, pp. 1–15.

Xia Feng received the B.S. degree in computer science and technology from Jiangsu University in 2008 and the Ph.D. degree from the Computer Science and Technology Department, Anhui University, in 2017. She is currently with the Faculty of Data Science, City University of Macau. Her research interests include authentication protocols in the IoT, blockchain, and applied cryptography.

Kaiping Cui is currently pursuing the Ph.D. degree with the Faculty of Data Science, City University of Macau. His research interests include identity privacy-preserving authentication, privacy and security in the IoVs, and secure aggregation in federated learning.

Liangmin Wang (Member, IEEE) received the B.S. degree in computational mathematics from Jilin University, Changchun, China, in 1999, and the Ph.D. degree in cryptography from Xidian University, Xi’an, China, in 2007. He is currently a Full Professor with the School of Cyber Science and Technology, South-East University, Nanjing, China. He has published over 60 technical papers in international journals and conferences. His current research interests include security protocols and the Internet of Things. He is a member of ACM.

Zhiquan Liu received the B.S. degree from the School of Science, Xidian University, Xi’an, China, in 2012, and the Ph.D. degree from the School of Computer Science and Technology, Xidian University, in 2017. He is currently a Lecturer with the College of Cyber Security, Jinan University, Guangzhou, China. His current research interests include trust management and privacy preservation in vehicular networks.

Jianfeng Ma (Member, IEEE) received the M.E. degree from the School of Computer Science and Technology, Xidian University, Xi’an, China, in 1988, and the Ph.D. degree from the School of Information and Telecommunication Engineering, Xidian University, in 1995. He is currently a Professor with the School of Cyber Engineering, Xidian University. His current research interests include information security, coding theory, and cryptography.

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.