CHINA COMMUNICATIONS

Key agreement and management scheme based on Blockchain for 5G-enabled Vehicular Networks

Zhihua Wang[1,2*], Shuaibo Wang[2], Jiaze Li[2], Haofan Wang[2], Yizhe Yao[2], Yongjian Wang[3*], Xiaolong Yang[1]

[1] School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China
[2] School of Cyberspace Security, Zhengzhou University, Zhengzhou 449001, China
[3] Lab of National Computer Network Emergency Response Technical Team/Coordination Center of China, Beijing 100029, China
[*] The corresponding author, email: wyj@cert.org.cn

Abstract: 5G technology has endowed mobile communication terminals with features such as ultra-wide band access and low transmission delay, which can complete the network access and interconnection of a large number of devices. 5G vehicular networks connect vehicle-to-vehicle and vehicle-to-roadside units through wireless communication channels, thereby providing a wide range of real-time traffic information related services. However, the fragility of wireless communication makes vehicle privacy and communication security a key issue that needs to be solved in vehicle-mounted networks. Large-scale communication makes higher communication efficiency an inevitable requirement. For efficient and safe communication while protecting vehicle privacy, this paper proposes a lightweight key agreement and key update scheme for 5G vehicular networks with the assistance of blockchain. The key agreement is accomplished through the public key in the certificateless system, and an efficient key updating method is designed based on the aggregate signature between the vehicle and the trusted institution. In addition, the introduction of blockchain and smart contract load vehicle public key table for key management, can track and dynamically revoke misbehaving vehicles. Finally, the security proof and comparative experimental analysis of the proposed scheme are carried out, and the results show that our proposed scheme has lower computational and communication overhead com-pared with other related schemes.

Keywords: key agreement; key management; blockchain; smart contract; 5G Vehicular Networks

Received: Aug. 14, 2021
Revised: xxx. xx, 2021
Editor: xx

I. INTRODUCTION

The traditional communication of Internet of vehicles (IOV) is mainly carried out through the vehicular ad hoc networks (VANETs),which specially designed for vehicle communication. VANETs enables wireless connection for vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) within a certain communication range, and automatically establish a mobile network [1]. In VANETs, the wireless communication of vehicles mostly uses the internationally formulated dedicated short-range communication (DSRC) technology. However, DSRC technology also has obvious limitations. For example, too many obstacles will affect the transmission efficiency, and the collision probability will increase when there are too many nodes [2], which cannot meet the goal of real-time stable communication of large-scale vehicle nodes, and restricts the development of IOV communication.

5G vehicular network uses cellular vehicle to everything(C-V2X) technology to form a vehicle communication network with multiple vehicles equipped with intelligent sensing devices, which can complete device-to-device(D2D) communication between terminal devices [3].C-V2X provides the characteristics of ultra-broadband access, low transmission delay, ultra-large connection, and ultra-large coverage for the

IOV communication [4], which is a good solution to the limitations of the DSRC technology.

5G vehicular network use wireless channel communication, which is vulnerable to eavesdropping, replay, and tampering attacks. Therefore, it is necessary to establish safe and reliable encryption for data transmission in the IOV. The Authentication and Key Agreement (AKA) allows both parties to authenticate in an open network environment and negotiate a secure shared session key, which can quickly establish a secure connection between vehicles and perform fast encrypted transmission of data.

However, the expansion of the user scale in the IOV has brought problems for the key management in the agreement. The most beginners use public key infrastructure (PKI) to issue certificates for vehicles, which also brings certificate management problems. Afterwards, scholars began to propose the use of identity-based public key cryptography, but the key escrow problem of forged signatures caused by the disclosure of private keys still cannot be solved. In the certificateless cryptosystem, part of the user's private key comes from the user, and this part of the private key is known to the user, and the other part is generated by the key generation center (KGC). It not only retains the user's identity characteristics, but also does not require a certificate, effectively solving the problems of certificate management and key escrow.

When vehicles use a AKA protocol based on public key authentication to communicate, the frequent use of fixed public keys for communication will inevitably increase the risk of private key leakage. The forward security of communication is difficult to be guaranteed. Therefore, it is necessary to study the efficient update method of the vehicle key. In the IOV scene, the number of users is huge and the duration of vehicle communication is short, key update should pay more attention to the efficiency of the update and reduce the pressure on the trusted center, share the computing overhead with edge devices or increase the computing and storage capacity of the system [5].

The calculation pressure of key update in the certificateless cryptographic system can be mainly borne by the vehicle itself. The vehicle updates the user's complete private key by updating part of the user's private key. The vehicle uses the old private key to sign the new public key, and the signature is handed over to a trusted authority to verify the legitimacy, which can complete an efficient key update. However, massive key updates will still bring huge computational pressure to the trusted center. Certificateless aggregate signature (CLAS) is an effective technology to improve message authentication. In CLAS, n signatures on n different messages from n different users are aggregated into a short signature, which can be verified immediately by merging [6]. Applying CLAS in the key update process can greatly improve computing efficiency and reduce communication overhead, as well as reduce the verification pressure of trusted authority.

There is also the problem of conditional anonymity in vehicular networks, which means that the private information of the vehicle can be viewed by trusted authority, and this information is invisible to any third party. IOV is in an exposed environment of wireless communication, so the broadcasted message should ensure the completion of identity verification, integrity and other security requirements. At the same time, it is necessary to protect the privacy of the vehicle from being obtained by malicious entities [7]. The method of generating a pseudonym of the vehicle is usually used to send messages in an anonymous form, which protects the privacy of the vehicle to a certain extent.

With the development of blockchain technology, its application in the field of IOV has also received extensive attention. Blockchain is essentially a data structure organized by linked lists, including other content derived from the data structure. It is based on algorithms in cryptography to ensure the characteristics of non-tampering, non-forgery, and decentralization. For the members of the alliance chain, the administrator can set access control policies for conditional data sharing. Using the alliance chain to store the key update records is helpful to track misbehaving vehicles, and can assist the identity authentication between vehicles by loading the vehicle public key table [1].

1.1 Our Research Contribution

In order to achieve the goal of protecting the security and privacy preservation, as well as promoting efficiency in large-scale communications, this paper proposes a lightweight key agreement scheme for 5G vehicular networks with the assistance of blockchain, and an efficient key update method suitable for large-scale vehicle key update is given. The main contribu-

tions of this paper can be summarized as follows:

1) In order to reduce computational overhead and promote communication efficiency, and solve the problem of key escrow at the same time, using a certificateless cryptographic system based on elliptic curve cryptosystems, a lightweight key generation and key agreement scheme suitable for 5G vehicular networks is designed.

2) In order to improve the forward security of the scheme, reduce the time overhead of key update and the calculation pressure of trusted authority, based on the aggregation signature method, a key update strategy suitable for 5G vehicular networks is designed. Trusted authority is able to complete a large number of user identity verifications at the same time.

3) In order to track and dynamically withdraw misbehaving vehicles, trace the abnormal behaviors of vehicles, and realize data sharing between trusted authority and vehicles in the system. This scheme stores the key update record through the alliance chain, and uses smart contracts to load the trusted vehicle public key table VPKT to summarize the mapping between vehicle pseudonyms and vehicle public keys, it can not only protect the privacy of the vehicle, but also assist the vehicle to perform identity authentication.

4) In order to verify the security and feasibility of the scheme, prove that the proposed scheme can meet the goals of communication security, privacy protection, and low overhead in the 5G vehicular network, the security proof and the comparative analysis of the performance of different schemes are carried out on the proposed scheme. The results show that the scheme can meet the security and privacy requirements of efficient communication, and the calculation and communication efficiency have also been greatly improved.

1.2 Organization of the Article

The rest of this article is organized as follows:Section 2 reviews the work related to the security and privacy protection of the IOV. In Section 3, we introduce the system model, design goals and preliminary knowledge. Section 4 introduces our proposed blockchain-assisted authentication and key agreement scheme, and the efficient key update strategy is given. Section 5 gives the security proof of the proposed scheme. Section 6 conducts experiments and analysis on the proposed scheme from the aspects of calculation and communication overhead. Finally, Section 7 summarizes this article.

II. RELATED WORK

In recent years, the security and privacy issues of the IOV have attracted the attention of many researchers. A variety of PKI-based authentication schemes [8–12] have been proposed. These schemes are constructed based on traditional public key cryptography (PKC). In PKI, each car first hands over its public key and identity to the certificate authority CA. After verification, the CA issues a certificate to it. The certificate provides the authenticity of the public key, each vehicle generates a pair of public and private key. However, all PKI-based authentication schemes have similar problems. It's that as the number of users continues to increase, certificate management (such as distribution, query or revocation) will generate high computational and communication overhead.

In 1984, Shamir [13] introduced an identity-based public key cryptosystem (IDC), which solved the problem of certificate management. IDC uses the user identity (email address, phone number, etc.) as the public key, and then uses the private key generator (PKG) to generate the corresponding private key. After that, ECC-based identity authentication schemes [14, 15] have been proposed to reduce computational overhead. However, in the IDC-based scheme, the key escrow problem that PKG can use the user's private key and forge the signature still cannot be solved.

In 2013, Al-Riyami and Paterson [16] introduced a new certificateless key (CLC) scheme, which eliminated the key escrow problem. First, the Key Generation Center (KGC) generates part of the private key and transmits it to the user. The user generates a complete private key using part of the private key and the random secret value selected by himself. Literature [17] studied the security model of certificateless cryptosystem, compared with PKI-based and identity-based keys, certificateless keys are not essentially different, and they are also efficient and safe for key agreement.

Due to the work of Al-Riyami and Paterson, in recent years, researchers have proposed many certificateless encryption [18, 19] and signature [20–22] schemes. In 2006, Yap et al. [23] designed a

certificate-free signature (CLS) scheme based on bilinear pairing for the difficulty of calculating Diffie-Hellman (CDH) in random oracle models (ROM). Since there is no bilinear pairing in the signature phase, and the verification phase only needs two bilinear pair calculations, this scheme is computationally efficient. But Zhang and Feng [24] analyzed and proved that the scheme in [23] is attackable under public key replacement attacks. He et al. [25] proposed a CLS scheme that does not use bilinear pairing operations in 2012. Although Tian and Huang [26] showed that the scheme in [25] cannot resist the attack of masquerading KGC, their CLS scheme provides a new way of thinking for reducing computational and communication costs. Since then, elliptic curve cryptography (ECC) has been widely used in the design of CLS and CLAS schemes.

In recent years, with the emergence of blockchain technology, some studies have tried to apply blockchain technology in virtual networks to establish a decentralized trust model. For example, Rowan et al. [27] used blockchain-based PKI and physical side channels for V2V secure communication, but this solution has some security problems in terms of autonomous driving requirements. Dorri et al. [28] proposed another privacy protection authentication based on blockchain and variable public keys, but it also has limitations such as user management and scalability. Lu et al. [29] and Kchaou et al. [30] used blockchain to optimize the trust management framework of the IOV and designed a privacy-aware reputation model. Assuming that the transactions in these two scenarios can safely record vehicle incidents, these incidents may become evidence for evaluating the reputation of the vehicle in the future. Although their programs all support strong accountability, they cannot prevent malicious behavior in advance.

With the advent of the 5G era, many researchers have begun to pay attention to 5G security and privacy [31–33]. The literature of [34, 35] investigated the security and privacy issues in 5G vehicular networks, and put forward some suggestions for improvement. Cui et al. [36] proposed a lightweight message authentication framework based on a reputation system for 5G vehicular networks, in which vehicles with poor reputation will not be able to join the communication. In this scheme, the author proposes an authentication scheme based on ECC, which supports batch authentication to reduce computational overhead, but it does not solve the problem of how to reduce communication overhead and how to revoke malicious users. The scheme of Ma et al. [37] uses an authentication key protocol without bilinear pairing to achieve mutual authentication, but this scheme cannot prevent short-term secret disclosure attacks. Recently, Cui et al. [38] proposed a scalable conditional privacy protection authentication scheme for secure IOV in a multi-cloud environment to solve the problem of cloud service provider (CSP) selection and resist known attacks. But we have noticed that the session key is generated by encrypting the pseudonym of the vehicle, the real identity of the CSP, and temporary information through a hash function. Since the pseudonym and hash function of the vehicle are public, the true identity of the CSP is exposed to the vehicle being authenticated. Therefore, the scheme in [38] cannot withstand short-term secret leakage attacks. Neither Ma et al.'s scheme [37] nor Cui et al.'s scheme [38] can prevent impersonation and man-in-the-middle attacks.

In order to protect privacy and the security of the transmitted message, it is necessary to use the session key to establish a secure connection, and to periodically update the key to protect the private key. Islam et al. [39] proposed a password-based conditional privacy protection authentication and group key agreement protocol, but the key update process requires TA support, and the key is sent through unicast. The efficiency of the program is low, and the program has not been proven to be safe. Hassan et al. [40] proposed an identity-based user authentication key protocol in a multi-server environment. The protocol uses ring signatures to allow users to anonymously verify their identities, providing unconditional anonymity. However, the above two schemes use bilinear pairing in the key agreement process, which makes the calculation overhead very large. Gervais et al. [41] proposed a certificateless authentication key agreement (CLAKA) for wireless local area networks. The second phase of the protocol describes the secure intermediary signature (SMC) for blockchain authentication. SMC has certain advantages in solving the problem of public key revocation, while maintaining the advantages of certificateless public key cryptosystems.

III. PREPARATORY WORK

3.1 System Structure

The communication model of the proposed scheme defines six participating entities in the entire system. The trusted authority (TA) provides trusted attributes for the communication in the system; 5G-RSU and Vehicle are the main roles in communication in the system, they are also the main service objects of information confidentiality in the system; Blockchain and smart contracts, as extended entities in the system, provide data security and data sharing functions for other entities in the system.

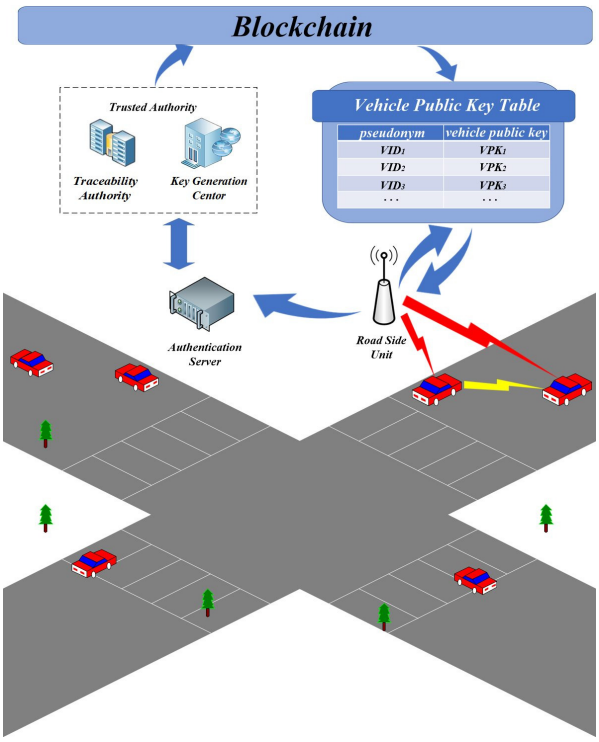


Figure 1. *The system model*

1) Trusted authority (TA): It exists as a trusted third-party agency, which is generally in charge of the government's traffic management center (TMC) department, composed of a key generation center (KGC) and a traceability authority (TRA), and has powerful computing and communication capabilities.

2) 5G-RSU is a roadside infrastructure that uses 5G base stations as a carrier and can communicate with vehicles within a specific range. In this paper, 5G-RSU has a certain amount of computing power to complete the aggregation of signatures.

3) Vehicle: Vehicle is the main communication participant in the 5G vehicle network. It has various sensors that can complete wireless communication with other participants. As a terminal node, it has certain computing and storage capabilities [42], and can complete a small amount of calculations in a certificateless cryptographic system.

4) Authentication Server(AS): AS is an application server with strong computing and storage capabilities. It provides large-scale computing services for vehicular networks and uses secure wired connections to communicate with TA and 5G-RSU. AS also assumes the function of aggregated signature verification.

5) Blockchain: The blockchain is designed according to the alliance chain, the member nodes of the alliance chain include TA and vehicle terminal nodes. It mainly completes the functions of data sharing and data security. The update records of all keys in the system are stored for the members of the alliance chain to query.

6) Smart contract: A smart contract is a program that is deployed on the blockchain to execute automatically. It is compiled by the administrator of the blockchain and deployed on the blockchain to ensure reliable calculation results. It will load the vehicle public key table (VPKT) according to the information on the blockchain to summarize the mapping of all vehicle pseudonyms and public keys, thereby providing automatic and timely feedback of vehicle public key queries.

The data on the blockchain comes from the TA, the administrator of the blockchain. TA has the authority to write data in the blockchain, as well as deploy, update and revoke smart contracts, and verify the correctness of all transactions and smart contracts. The vehicle terminal node has the permission to read the blockchain data and obtain the public keys of other vehicles for secure communication by querying the VPKT.

3.2 Designed Goals

1) Key escrow. In order to improve the confidentiality of the user's private key in the system, reduce the possibility of private key leakage, as well as reduce the pressure of key escrow, the private key of the vehicle should only be known by itself, and the key generation center can participate in the key generation.

2) Single registration. In order to facilitate the communication in the 5G vehicular networks, the vehicle can generate public and private keys after single registration in the system.

3) Privacy protection of vehicles. The true identity of the vehicle should be invisible to other vehicles and RSU in the system, as well as cannot be analyzed from the information sent by the vehicle.

4) Efficient key agreement protocol. It is difficult for vehicles to maintain long-term communication in 5G vehicular networks. Vehicles should efficiently execute identity authentication and establish temporary session keys to reduce computational overhead.

5) Efficient key update strategy. In order to improve the forward security of vehicle-to-vehicle communication, the key of the vehicle should be able to be updated quickly, and the trusted center should be able to quickly complete the legitimacy verification of a large number of key update operations.

6) Traceability of malicious vehicle information. When malicious behavior occurs, the trusted center should be able to track the true identity of malicious vehicles. For example, malicious vehicles update their keys quickly after sending false information, thereby denying their malicious behavior.

7) Resist malicious attacks. Due to the openness of communication in the IOV, the system should be able to resist various known attacks, such as replay attacks.

IV. THE PROPOSED SCHEME

4.1 The Generation of Vehicle Key in Certificateless System

Step 1: System initialization phase.

The system selects a safe parameter λ. TA takes two larger prime numbers p, q respectively, and generates an elliptic curve $E : y^2 = x^3 + ax + b(\bmod p)$, where $a, b \in Z_q^*, \left(4a^3 + 27b^2\right) \bmod p \neq 0$.

KGC selects a point P from the elliptic curve E, and generates a group G_p with a prime order q from P. KGC chooses an $s \in Z_q^*$ as its master key and calculates the corresponding public key $K_{pub} = s \cdot P$, where s is only known by KGC.

TRA selects a random number $t \in Z_q^*$ as its master private key, which is used to track the vehicle identity and calculates the system public key $T_{pub} = t \cdot P$, where t is only known by TRA.

TA selects some secure hash functions and publishes the system parameters, params $= \{P, p, q, E, G, h_1, h_2, h_3, h_4, K_{pub}, T_{pub}\}$.

After the system parameters are released, any vehicle that has completed the registration with the TA can obtain the system parameters through the secure channel; similarly, any RSU can obtain the system parameters after the registration is completed.

Step 2: Vehicle registration and pseudo identity generation phase.

Vehicle V_i selects a random number $k_i \in Z_q^*$. calculates $P\mathrm{ID}_{1i}, P\mathrm{ID}_{2i}$. $P\mathrm{ID}_{1i} = k_i \cdot P$, $P\mathrm{ID}_{2i} = RID_i \oplus h_1\left(P\mathrm{ID}_{1i}, T_{pub}, \nabla, K_{pub}\right) \cdot k_i \cdot T_{pub}$, and then sends $\{P\mathrm{ID}_{1i}, P\mathrm{ID}_{2i}\}$ to TRA.

After the TRA receives $\{P\mathrm{ID}_{1i}, P\mathrm{ID}_{2i}\}$. calculates the $RID_i, RID_i = P\mathrm{ID}_{2i} \oplus h_1\left(P\mathrm{ID}_{1i}, T_{pub}, K_{pub}\right) \cdot t \cdot P\mathrm{ID}_{1i}$ to verify its real identity. If the verification is not passed, it will be discarded directly. Otherwise, TRA calculates $\mathrm{PID}_i, P\mathrm{ID}_i = RID_i \oplus h_2\left(t \cdot P\mathrm{ID}_{1i}, P\mathrm{ID}_{2i}, T_{pub}, K_{pub}, VP_i\right)$.

In order to ensure user privacy and malicious vehicle traceability, TRA will store the real identity RID_i and the pseudo identity $P\mathrm{ID}_i$ of the vehicle.

Finally, TRA sends $\{P\mathrm{ID}_i, VP_i\}$ to KGC through a secure channel, VP_i is the survival time of V_i.

Step 3: Partial private key generation phase.

After the KGC receives the pseudo identity $P\mathrm{ID}_i$ of the vehicle V_i, selects a random number $r_i \in Z_q^*$, calculates $R_i = r_i \cdot P, h_{ui} = h_3\left(P\mathrm{ID}_i, \mathrm{R}_i, T_{pub}, K_{pub},\right)$, and then calculates $ppk_i = r_i + s \cdot h_{ui}(\bmod q)$.

KGC transmits $\{R_i, ppk_i\}$ to the vehicle V_i through a secure channel, and the partial private key of the vehicle is ppk_i.

Step 4: Vehicle key generation phase.

After the vehicle V_i receives $\{R_i, ppk_i\}$, calculates whether $ppk_i \cdot P = R_i + h_{ui} \cdot K_{pub}$ is established to determine if the partial private key is valid. If it is not established, discard, otherwise the following steps are made.

Vehicle V_i randomly selects $vsk_i^b \in Z_q^*$, calculates $vpk_i^b = vsk_i^b \cdot P + ppk_i \cdot P$.

The vehicle private key is vsk_i^b, the vehicle public key is vpk_i^b.

4.2 Lightweight Key Agreement Protocol Design

The target of key agreement protocol is to achieve fast identity authentication and session key generation between entities. In this scheme, the mapping table of vehicle's pseudo identity, and vehicle's public key is stored in VPKT, which can be searched before vehicle communication, and it is not necessary for both parties to exchange certificates during communication. This protocol is suitable for V2V and V2I communication from the point of view of maximizing computing speed and reducing communication overhead. Key agreement between entities is performed according to the following protocol:

Step 1: A randomly selects a number $a_A \in Z_q^*$, calculates $W_{\mathrm{A}} = a_A \cdot ppk_A \cdot P$, and sends W_{A} to B.

Step 2: B randomly selects a number $b_B \in Z_q^*$, calculates $W_{\mathrm{B}} = b_B \cdot ppk_B \cdot P$, and sends W_{B} to A.

Step 3: Generate session keys:

1. When A accepts the information of B, calculates $K_{AB} = (a_A \cdot ppk_A) \cdot W_{\mathrm{B}}$.

2. When B accepts the information of A, calculates $K_{BA} = (b_B \cdot ppk_B) \cdot W_{\mathrm{A}}$.

3. Agreement session key:

$SK_{AB} = h_3\left(P\mathrm{ID}_A, P\mathrm{ID}_B,\ \mathrm{W}_A,\ \mathrm{W}_B, K_{AB}, T_A\right) = h_3\left(P\mathrm{ID}_A, P\mathrm{ID}_B,\ \mathrm{W}_A,\ \mathrm{W}_B, K_{BA}, T_B\right)$

SK_{AB} is a temporary session key of both parties in this communication, and the two parties use SK_{AB} for fast communication. Here we set the usage period T for the temporary session key SK_{AB}. Within the time T, SK_{AB} can be used for multiple communications between the two parties without repeating the above process again.

The transmitting of random numbers a_A and b_B ensures the confidentiality and freshness of the session key, and the application of timestamp can intuitively judge whether the message is a replay or not.

The characteristics of the above protocol are that the parameters involved in the communication are small enough and the time overhead of establishing the session key is small enough. However, its security depends heavily on the private key. As long as all the previous ciphertext is intercepted, all the parameters of the computing session key can be decrypted and obtained after obtaining the user's long-term private key. Therefore, it is necessary to introduce key update policy to ensure the forward-backward security of the protocol, and to compensate for the problem that its security depends on the private key.

4.3 Efficient Key Update Policy Design

Key updates are an important guarantee for forward-backward security. Setting key update policy can avoid the disclosure of user's private key and other problems caused by long-term use of a private key, and improve the difficulty of adversary analysis of communication data. Key update is performed by the following modules:

Step 1: Update and signature of private keys in certificateless system

The initiator of the key update is the vehicle, which affects the final public private key by changing its own secret values in the certificateless system. The vehicle performs the following operations:

When the survival time of the vehicle key expires, the vehicle updates the key. The vehicle selects the secret value $vsk_i^a \in Z_q^*$ again and calculates $vpk_i^a = vsk_i^a \cdot P$.

The updated private key of the vehicle is $vsk_i^a \in Z_q^*$, the public key is $vpk_i^a \in Z_q^*$, and the complete private key is $sk_i^a = (ppk_i, vsk_i^a)$.

The vehicle stores the public and private keys vsk_i^b, vpk_i^b and vsk_i^a, vpk_i^a generated twice before and after.

After the vehicle completes the private key update, the updated public key vpk_i^a is signed with the last private key vsk_i^b of the vehicle V_i. This signature operation can only be completed by the vehicle V_i. The operation equivalent to the vehicle V_i announcing the key update is completed by itself, which can effectively avoid other system roles such as malicious users and malicious KGC impersonating the vehicle to perform key update.

The vehicle performs the following signature steps:

The vehicle V_i randomly generates $u_i \in Z_q^*$, calculates $U_i = u_i \cdot P$, $h_{xi} = h_4\left(vpk_i^a, PID_i, K_{pub}, U_i, VP_i\right)$, $v_i = u_i + h_{xi} \cdot \left(vsk_i^b + ppk_i\right) (\mathrm{mod} q)$, and generates the signature $\sigma_i = (v_i, U_i)$ about the updated public key vpk_i^a.

Finally, the vehicle broadcasts the signature-message pair over the secure channel.

It should be noted that after completing the above steps, the vehicle itself initiates a key update behavior. After the update, the new public key vpk_i^a is currently

not recognized by the AS, nor is it updated in the vehicle public key table VPKT. Vehicle communication is subject to VPKT, and whether the key update is completed or not is subject to whether VPKT is updated.

When the RSU receives the updated public key vpk_i^a and signature of the vehicle V_i, it verifies the updated public key vpk_i^a, respectively calculates $h_{xi} = h_4\left(vpk_i^a, PID_i, K_{pub}, U_i, VP_i\right)$, and checks whether $v_i \cdot P = U_i + h_{xi} \cdot vpk_i^b$ is established. If not, then reject the public key, oth-erwise the authentication is passed.

Step 2: Aggregation of key update signature σ_i.

The idea of aggregate signature is to compress the length of the signature and aggregate the signature σ_i of all key update records in a certain area. This work is done by RSU.

The aggregation signature performs the following process.

When RSU receives multiple messages with pseudo identity $P\mathrm{ID}_i$, public key vpk_i^b, and message-signature pairs from different vehicles V_i, RSU plays the role of aggregate signature generator, aggregating multiple certificateless signatures into a short signature. That is, RSU calculates $V = \sum_{i=1}^{n} v_i$, $U = \sum_{i=1}^{n} U_i$ and outputs $\sigma = (V, U)$ as a certificateless aggregate signature to facilitate later aggregate signature verification.

Step 3: Aggregate signature verification.

After being aggregated by RSU, the signature is sent to AS for verification of the aggregated signature. At the same time, the pseudo identity of the vehicle $P\mathrm{ID}_i$ and the public key of the vehicle update vpk_i^a are also sent to AS for quick completion of VPKT update after verification. The verification of the aggregate signature follows the following process.

Once an AS receives the aggregate signature $\sigma = (V, U)$ of multiple certificateless message-signature pairs with pseudo identity $P\mathrm{ID}_i$, public key vpk_i^b from different vehicles V_i sent by RSU, the application server AS checks the validity of pseudo identity VP_i and performs the following steps to verify the aggregate signature if it is valid.

The AS calculates $h_{xi} = h_4\left(vpk_i^a, PID_i, K_{pub}, U_i, VP_i\right) i \in \{1, 2, 3, \cdots, n\}$.

The AS verifies whether $V \cdot P = U + \sum_{i=1}^{n} h_{xi} \cdot vpk_i^b$ is established, and if it is established, the aggregated signature verification is passed and the message is accepted. Otherwise, reject this message.

If the verification of the aggregated signature fails, AS rejects this message; the VPKT can be queried whether the vehicle key update is successful.

Step 4: The update of vehicle public key table (VPKT)

After the AS verifies the aggregate signature, it is equivalent to approving the legitimacy of the vehicle key update. The legal key update record after the completion of the aggregated signature verification can be recorded in the blockchain. AS forwards the verified legal information to TA, and the TA uploads the three sets of data of the pseudo identity $P\mathrm{ID}_i$, the new public key VPK, and the aggregate signature σ to the blockchain body. When the blockchain data is updated, the smart contract program is automatically executed to update the VPKT. In the VPKT, the new public key of the vehicle will replace the original public key. When the public key of the vehicle in VPKT is updated, the key update of the vehicle is finished, and the vehicle can check whether the current key update is completed through VPKT.

4.4 Design of Blockchain and Smart Contract

1) Design of Blockchain

Head: The head of each block in the blockchain consists of the hash value of the previous block header, the timestamp, and the hash value of all ag-gregate signature values of the previous block.

Body: The body of each block in the blockchain has the complete trade information in a certain amount of period. The data in the body is in form of form record, which contains the following three parts: vehicle pseudonym PID, public key VPK, and aggregate signature value.

Blockchain initialization: As the blockchain manager, TA starts the consortium blockchain among the nodes of the consortium blockchain with PBFT. At the same time, TA deploys an access control list on the blockchain. The trusted center TA has the permission to write data to the consortium blockchain, and the vehicle as a member node of the consortium blockchain has the permission to read data on the consortium blockchain.

2) Design and deployment of smart contract

Smart contract is originally proposed by Nick Szabo. Developers use smart contracts to design a set of rules and publish them online. The machine completes

the business part. It avoids the cheating behavior that caused by human [43]. The smart contract design of this scheme is learned from ideas of FENG et al [1]. The smart contract provides an application binary interface (ABI) for the vehicle public key table (VPKT) service, which supports the update, insertion, and upload of the public key, which supports the update, insertion, and upload of the public key. Algorithms 1 complete the initialization of the vehicle public key table. Algorithms 2 complete the update of the public key. Algorithms 3 complete the insertion of the public key. Algorithms 4 complete revocation of the public key in the smart contract.

Once the ECC-based certificateless system is set up, the vehicle public key table will load the initial public keys of each vehicle in the system which were generated by the TA to complete the initialization of the VPKT. The VPKT only communicates with the TA once during initialization. After that, the vehicle public key table records the initial public keys of all vehicles in the system.

The logic of the smart contract algorithm is as follows:

Algorithm 1 is the initialization of the VPKT system which declares the required parameter settings. The update, insertion, and deletion of keys are considered as transactions to design the solutions.

Algorithm 1. *VPKT Initialization*

```
asset VPKT identified PID{
o string PID;
o address VPK;}
transaction UpdateTransaction {
–> VPKT asset;
o address newVPK;}
transaction InsertTransaction {
o string newPID;
o address newVPK;}
transaction RemoveTransaction {
o string PID;}
```

Algorithm 2 completes the update operation of the key in VPKT. After verifying the correctness of the aggregate signature (CLAS), the system uses this ABI to complete the update operation of the key.

Algorithm 3 completes the users insertion operation in VPKT. When a new vehicle generates a public key, the public key is inserted into the public key table through Algorithm 3.

Algorithm 2. *Update VPKT*

```
function updateVPKT (Form record)
{const assetRegistry = await getAssetRegisrty(VPKT.PID);
if Exist(VPKT[i].PID == Form record.PID) then
    {Form record.asset.VPK = Form record.newVPK;
    await assetRegistry.update(entry);}
end if
}
```

Algorithm 3. *Insert VPKT*

```
function insertVPKT (Form record){
const assetRegistry = await getAssetRegistry(VPKT.PID);
if Exist(VPKT[i].PID == Form record.PID) then
    var entry = newResource(VPKT);
end if
}
```

Algorithm 4 completes the users deletion operation in VPKT. When a vehicle exits the network due to various reasons such as breakdown or old, the public key of the vehicle needs to be revoked to reduce the threat of imitate attacks. Call algorithm 4 to complete the revocation of the user's public key in the public key table.

Algorithm 4. *Remove VPKT*

```
function removeVPKT (Form record){
const assetRegistry = await getAssetRegistry(VPKT.PID);
if Exist(VPKT[i].PID == Form record.PID) then
    assetRegistry.remove(VPKT[i].PID);
end if
return succ; }
```

Deployment of smart contracts:

The deployment of the smart contract is completed by the TA. After the TA accepts the smart contract algorithm, it compiles it and deploys the smart contract on the blockchain. TA also completes the update, insertion, and revocation operations of the public key as a blockchain manager.

The introduction of blockchain and smart contracts

in the plan is mainly for the following purposes:

The first reason is to ensure that the key update records are traceable. When malicious behaviors such as imitate attacks and key tampering attacks, TA can trace forward through the information on the chain. For example, when a malicious vehicle sends fake information to mislead others, the key is quickly updated. The introduction of the blockchain makes this behavior checkable. At the same time, it prevents malicious KGC from changing the user's public and private keys in the certificateless system, thereby impersonating users.

The second reason is to construct a trusted platform. The AS verifies the aggregate signature in the blockchain, the approved and trusted key will update the aggregate signature on the chain after verification. The public key update records of all vehicles are stored in the blockchain. The smart contract completes the modification of VPKT based on the information on the chain, which makes VPKT trusted.

V. SECURITY ANALYSIS

In this chapter, we will analyze the security of this program and prove that the program is safe. This scheme has forward security, unforgeability, security of known session keys, security of key control, resistance to malicious attacks, etc. Literature [44] proves that the signature technology used in this paper to complete the final session key agreement. In this case, it will not be quoted here.

1) Forward security In this scheme, the private keys of A and B will be updated regularly in a cycle, and at the same time, blockchain will verify data to ensure the security of the private key. Even if the attacker has the private key of the users, but because he does not have the temporary secret information a_A and b_B, the attacker will face the CDH problem and cannot effectively calculate W_A and W_B. With an efficient key update strategy, this scheme achieves the goals of forward security. 2) The security of the known session key In this scheme, the temporary secret information a_A and b_B are randomly generated by vehicles A and B. Even if the attacker executes the protocol multiple times, the final session key generated by the protocol will not be the same.

3) Security of key control

In this scheme, because the final calculation of the session key is related to PID_A, PID_B, W_A, W_B, K_{AB}, T_A, So neither user A nor B can predict the final session key in advance.

4) Traceability

When malicious behavior occurs, the trusted center should be able to trace the malicious vehicle's source and track the malicious vehicle's true identity. For example, the malicious vehicle quickly updates the key after sending false information to deny communication.

5) Resistance to replay attacks

The transmission of the numbers a_A and b_B also guarantees the confidentiality and freshness of the session key, the use of timestamps can intuitively determine whether it is a replayed message, which proves that the solution can resist replay attacks.

VI. PERFORMANCE ANALYSIS

1) Computational overhead analysis

We use the evaluation method used by He et al. [14] when performing computational overhead analysis. In the scheme proposed by He et al. [14], the running time of various cryptogram operations was calculated by considering the hardware platform. They used the clock frequency of 3.40GHz, 4GB of running memory, Intel I7-4670 processor and a computer running Windows7 operating system to run MIRACL library for the experiment. For the scheme based on the bilinear pairing, we construct the bilinear pairing as follows: the bilinear pairing $\bar{e} : G_1 \times G_1 \rightarrow G_2$ is constructed at the 80-bit safety level, where G_1 is an addition group of order $\hat{q}$ on a hypersingular elliptic curve $\bar{E} : y^2 = x^3 + x \bmod \hat{p}$ with embeddedness degree of 2. $\hat{p}$ is a 512-bit prime, $\hat{q}$ is a 160-bit prime. Elliptic curve encryption is constructed at an 80-bit security level: G is an addition group whose order is q, and the generator is a point P on a nonsingular elliptic curve $\bar{E} : y^2 = x^3 + ax + b \bmod p$, where $a, b \in Z_q^*$, p, q are 160-bit prime numbers. In our analysis, we do not consider the execution time required for general addition and general multiplication, because the time required to process these operations is negligible and can be ignored. The specific basic operation execution times of bilinear pairings and elliptic curve cryptography are shown in the table 1.

We compare the computational cost of the documents [45–49] and our scheme in the signature, ver-

ification phase, and aggregate signature verification phase. The schemes [45–49] are constructed based on bilinear pairing, and the schemes [48], [49] are constructed using the same elliptic curve cryptosystem as our scheme. The specific analysis of the calculation cost is listed as follows.

According to the table 2, it can be seen that Ali's scheme [46] requires a scalar multiplication operation in G_1 and a one-way hash for signature, and its overhead in the signature stage is $T_{sm\_bp}+T_h = 1.7091ms$. In the verification stage, a bilinear pairing, a scalar multiplication operation in G_1, a point addition operation in G_1 and two one-way hashes are needed, and the cost is $T_{bp}+T_{sm\_bp}+T_{pa\_bp}+2T_h = 5.9273ms$. In the signature verification stage of n messages aggregation, the scheme needs a bilinear pairing, n scalar multiplication operations in G_1, n point addition operations in G_1, and 2n one-way hashes, and the final cost is $T_{bp}+nT_{sm\_bp}+nT_{pa\_bp}+2nT_h = 1.7163n+4.211ms$.

In our scheme, the overhead of the signature stage consists of a scalar multiplication operation in G and a one-way hash, and the overhead of the signature is $T_{sm\_ecc} + T_h = 0.4421ms$. In the verification stage, two scalar multiplication operations in G, one point addition operation in G, and a hash function are required, with an overhead of $2T_{sm\_ecc} + T_{pa\_ecc} + T_h = 0.8859ms$. Since we perform the predicted calculation in the aggregate signature $U = \sum_{i=1}^{n} U_i$, at the stage of verifying the aggregate signature, our scheme only needs to perform n+1 scalar multiplication operations in G, n point addition operations in G and n one-way hashes, and the final overhead is $(n+1)T_{sm\_ecc} + nT_{pa\_ecc} + nT_h = 0.4439n + 0.442ms$.

Figures 2 and **??** compare the computational overhead of our scheme with the related schemes in [45–49] for message signature, single signature verification, and aggregate signature verification. It can be seen from the figure that our scheme has higher computational efficiency in message signature, single signature verification and aggregate signature verification.

Table 1. *Execution time of basic cryptography operations*

Notations	Description	Run time(ms)
T_{pa_bp}	Bilinear pairing point addition	0.0071
T_{sm_bp}	Bilinear pairing scalar multiplication	1.7090
T_{bp}	Bilinear pairing	4.2110
T_{pa_ecc}	Elliptic curve point addition	0.0018
T_{sm_ecc}	Elliptic curve scalar multiplication	0.4420
T_{mtp}	Map to point hash function	4.4060
T_h	One-way hash function	0.0001

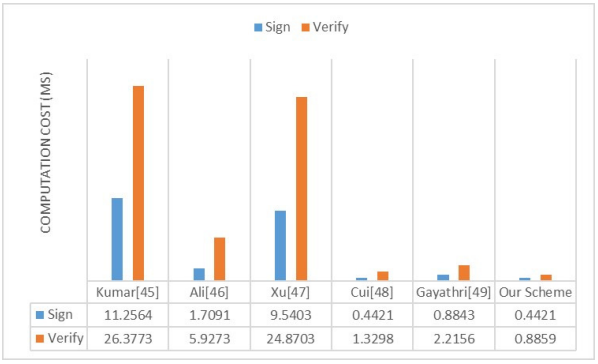


Figure 2. *Comparison of signature and single verification time*

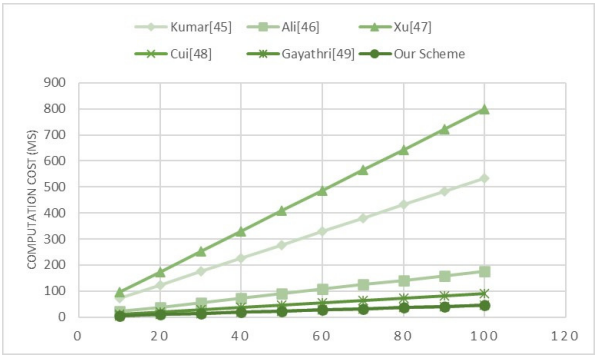


Figure 3. *Aggregated verification time versus number of signatures*

Next, the communication cost of our scheme is compared with the cost of signature, single signature verification and aggregate signature verification of related schemes in [45–49]. Compared with Kumar's scheme [14], the improvement percentages of our scheme in terms of signature, single signature verification and aggregate signature verification are $(11.2564 - 0.4421)/11.2564 \times 100\% \approx 96.07\%$, $(26.3773 - 0.8859)/26.3773 \times 100\% \approx 96.64\%$ and $5.1273n + 21.25 - (0.4439n + 0.442)/(5.1273n + 21.25) \times 100\% \approx 91.60\%$, where n represents the total number of signatures, assuming 100. For other schemes in [46–49], the percentage improvement is shown in the table 3.

In summary, it can be seen that our scheme is superior to existing schemes [45–49] in signature generation, single signature verification, aggregate signature verification, etc.

Table 2. *Analysis of the computation overhead for related schemes*

scheme	Sign	Verify	n Signature verify	pairing
Kumar [45]	$4T_{sm\_bp}+2T_{pa\_bp}+T_{mtp}+2T_h=11.2564ms$	$4T_{bp}+3T_{sm\_bp}+T_{mtp}+3T_h=26.3773ms$	$4T_{bp}+3nT_{sm\_bp}+T_{mtp}+3nT_h=5.1273n+21.25ms$	Yes
Ali [46]	$T_{sm\_bp}+T_h=1.7091ms$	$T_{bp}+T_{sm\_bp}+T_{pa\_bp}+2T_h=5.9273ms$	$T_{bp}+nT_{sm\_bp}+nT_{pa\_bp}+2nT_h=1.7163n+4.211ms$	Yes
Xu [47]	$3T_{sm\_bp}+T_{pa\_bp}+T_{mtp}+2T_h=9.5403ms$	$3T_{bp}+2T_{sm\_bp}+T_{pa\_bp}+2T_{mtp}+2T_h=24.8703ms$	$3T_{bp}+2nT_{sm\_bp}+nT_{pa\_bp}+(n+1)T_{mtp}+2nT_h=7.8313n+17.039ms$	Yes
Cui [48]	$T_{sm\_ecc}+T_h=0.4421ms$	$3T_{sm\_ecc}+2T_{pa\_ecc}+2T_h=1.3298ms$	$(n+2)T_{sm\_ecc}+2nT_{pa\_ecc}+2nT_h=0.4458n+0.884ms$	No
Gayathri [49]	$2T_{sm\_ecc}+3T_h=0.8843ms$	$5T_{sm\_ecc}+3T_{pa\_ecc}+2T_h=2.2156ms$	$(2n+1)T_{sm\_ecc}+(2n+1)T_{pa\_ecc}+2nT_h=0.8878n+0.4438ms$	No
Our Scheme	$T_{sm\_ecc}+T_h=0.4421m$	$2T_{sm\_ecc}+T_{pa\_ecc}+T_h=0.8859ms$	$(n+1)T_{sm\_ecc}+nT_{pa\_ecc}+nT_h=0.4439n+0.442ms$	No

Table 3. *Improvement in percentage*

Scheme	Sign	Single sig. verify	n sigs. verify
Kumar [45]	96.07%	96.64%	91.60%
Ali [46]	74.13%	85.05%	74.49%
Xu [47]	95.37%	96.44%	94.40%
Cui [48]	0%	33.38%	1.39%
Gayathri [49]	49.01%	60.02%	48.75%

2) Communication overhead analysis

The method in [14] is used to calculate the communication overhead of our scheme and related schemes in [45–49] and compare it.

The length of the elements in the various groups is shown in the table 4. In order to standardize the calculation and comparison of communication overheads, we assume that the length of the message signed by each scheme and the length of the timestamp are the same, so the communication overhead only calculates the sum of the length of the pseudo identity, the public key and the length of the signature. In the scheme of Kumar [45] pseudo identity $PID_i \in G_1$, public key $PK_i \in G_1$, signature $\sigma_i = (U_i, V_i) \in G_1$, and the aggregated signature is $\sigma = (U_1, U_2, \ldots\ldots, U_n, V)$. So the total communication overheads generated by [45] scheme are about $4 \times 128 = 512\ bytes$ and $(3n+1) \times 128 = 384n+128\ bytes$, respectively. In our scheme, pseudo identity $PID_i \in Z_q^*$, public key $PK_i \in G$, signature $\sigma_i = (v_i, U_i), v_i \in Z_q^*, U_i \in G$, and the aggregated signature is $\sigma = (V, U)$. Therefore, the total communication overheads of our scheme are approximately $2 \times 40 + 2 \times 20 = 120\ bytes$ and $(n+1) \times 40 + (n+1) \times 20 = 60n+60\ bytes$. It can be seen from the table 5 that the bilinear pairing-based schemes in [45], [46] and [47] have higher communication overhead than the ECC-based scheme. In addition, compared with the ECC-based scheme proposed in [48], the communication overhead of our scheme is smaller.

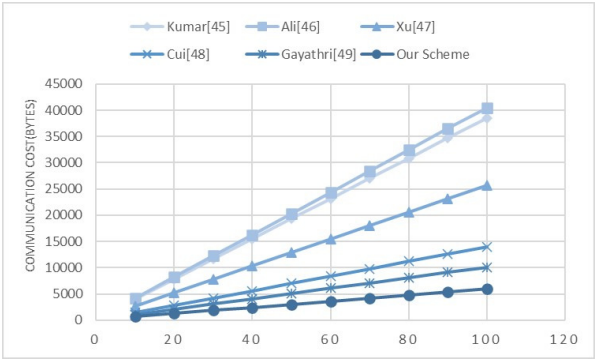


Figure 4. *Aggregate signature size versus number of participants*

VII. CONCLUSION

1) The lightweight key agreement protocol designed in this paper has a good adaptability to the key update strategy, which effectively protects the privacy of vehicles, solves the problem of key escrow, as well as completes efficient identity authentication in 5G vehicular networks and the establishment of session key.

2) The method of key-update shares the computational pressure on the edge devices, and the comparative experiment proves that the key update algorithm based on aggregated signature is more efficient in communication and calculation.

3) The blockchain enhances the security of the system and completes the tracking of malicious vehicles. The vehicle public key table loaded by the smart contract provides a public key query platform for the vehicle to complete the key update and query.

4) The scheme can resist various known network attacks, and the security proof verifies the security of the scheme.

In order to make the scheme proposed in this paper better achieve the goal of high efficiency, it is still necessary to optimize various parameters in the process of public and private key establishment and key update.

Table 4. *Length of variables in bilinear pairing and ECC*

Type of the System	Type of the Curve	$\|p\|$	$\|G\|$	Length of elements of the group
Bilinear Pairing	$\bar{E}: y^2 = x^3 + x \bmod p$	$\|p\|$=512 bits (64 bytes)	q=160 bits (20 bytes)	$\|G_1\|$=1024 bits (128 bytes)
ECC	$\bar{E}: y^2 = x^3 + ax + b \bmod p \quad a, b \in Z_q^*$	$\|p\|$=160 bits (20 bytes)	q=160 bits (20 bytes)	$\|G_1\|$=320 bits (40 bytes)

Table 5. *Communication overhead comparison*

Scheme	Single sig. transmit	n sigs. transmit
Kumar [45]	$4\|G_1\| = 512\ bytes$	$(3n+1)\|G_1\| = 384n + 128\ bytes$
Ali [46]	$4\|G_1\| + \left\|Z_q^*\right\| = 532\ bytes$	$(3n+1)\|G_1\| + n\left\|Z_q^*\right\| = 404n + 128\ bytes$
Xu [47]	$3\|G_1\| + \left\|Z_q^*\right\| = 404\ bytes$	$(2n+1)\|G_1\| + \left\|Z_q^*\right\| = 256n + 147\ bytes$
Cui [48]	$3\|G\| + 2\left\|Z_q^*\right\| = 160\ bytes$	$3n\|G\| + (n+1)\left\|Z_q^*\right\| = 140n + 20\ bytes$
Gayathri [49]	$3\|G\| + 3\left\|Z_q^*\right\| = 180\ bytes$	$2(n+1)\|G\| + (n+1)\left\|Z_q^*\right\| = 100n + 100\ bytes$
Our Scheme	$2\|G\| + 2\left\|Z_q^*\right\| = 120\ bytes$	$(n+1)\|G\| + (n+1)\left\|Z_q^*\right\| = 60n + 60\ bytes$

References

[1] FENG Q, HE D, ZEADALLY S, et al. Bpas: Blockchain-assisted privacy-preserving authentication system for vehicular ad hoc networks [J]. IEEE Transactions on Industrial Informatics, 2019, 16(6):4146-4155.

[2] ZHAO J, ZHANG B, PAN X, et al. Research on telematics communication technol-ogy and application prospects[J]. Auto Time, 2021, 6(3): 15-16+32.

[3] XIAO Y, LIU H, CHENG X. Key technologies of internet of vehicles and their development trends and challenges[J]. Communication technology, 2021, 1(8):1-8.

[4] GU W. Opportunities and challenges for the develop-ment of internet of vehicles in the 5g era [J]. Technology Vision, 2019, 19(3):1-3.

[5] XIONG H, CHEN J, MEI Q, et al. Conditional privacy-preserving authentication protocol with dynamic membership updating for vanets [J]. IEEE Transactions on Dependable and Secure Computing, 2020(01):1-1.

[6] LAI C, LU R, ZHENG D, et al. Security and privacy challenges in 5g-enabled vehicular networks[J]. IEEE Network, 2020, 34(2):37-45.

[7] CUI J, WEN J, HAN S, et al. Efficient privacy-preserving scheme for real-time location data in vehicular ad-hoc network[J]. IEEE Internet of Things Journal, 2018, 5(5):3491-3498.

[8] RAYA M, HUBAUX J P. Securing vehicular ad hoc networks[J]. Journal of computer security, 2007, 15(1):39-68.

[9] LU R, LIN X, ZHU H, et al. Ecpp: Efficient conditional privacy preservation protocol for secure vehicular communications[C]//IEEE INFOCOM 2008-The 27th Conference on Computer Communications. [S.l.]: IEEE, 2008: 1229-1237.

[10] WASEF A, SHEN X. Emap: Expedite message authentication protocol for vehicular ad hoc networks[J]. IEEE transactions on Mobile Computing, 2011, 12(1):78-89.

[11] CINCILLA P, HICHAM O, CHARLES B. Vehicular pki scalability-consistency trade-offs in large scale distributed scenarios[C]//2016 IEEE Vehicular Networking Conference (VNC). [S.l.]: IEEE, 2016: 1-8.

[12] ASGHAR M, DOSS R R M, PAN L. A scalable and efficient pki based authentication protocol for vanets[C]//2018 28th International Telecommunication Networks and Applications Conference (ITNAC). [S.l.]: IEEE, 2018: 1-3.

[13] SHAMIR A. Identity-based cryptosystems and signature schemes[C]//Workshop on the theory and application of cryptographic techniques. [S.l.]: Springer, 1984: 47-53.

[14] HE D, ZEADALLY S, XU B, et al. An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks[J]. IEEE Transactions on Information Forensics and Security, 2015, 10(12):2681-2691.

[15] ALI I, LAWRENCE T, LI F. An efficient identity-based signature scheme without bilinear pairing for vehicle-to-vehicle communication in vanets[J]. Journal of Systems Architecture, 2020, 103:101692.

[16] AL-RIYAMI S S, PATERSON K G. Certificateless public key cryptography[C]//International conference on the theory and application of cryptology and information security. [S.l.]: Springer, 2003: 452-473.

[17] DENT A W. A survey of certificateless encryption schemes and security models[J]. International Journal of Information Security, 2008, 7 (5):349-377.

[18] DENT A W, LIBERT B, PATERSON K G. Certificateless encryption schemes strongly secure in the standard model[C]//International Workshop on Public Key Cryptography. [S.l.]: Springer, 2008: 344-359.

[19] SUN Y, LI H. Short-ciphertext and bdh-based cca2 secure certificateless encryption[J]. Science China Information Sciences, 2010, 53(10):2005-2015.

[20] FENG S R, MO J, ZHANG H, et al. Certificateless short signature scheme from bilinear pairings[C]//Applied Mechanics and Materials: volume 380. [S.l.]: Trans Tech Publ, 2013: 2435-2438.

[21] HE D, HUANG B, CHEN J. New certificateless short signature scheme[J]. IET Information Security, 2013, 7(2):113-117.

[22] XU Z, LIU X, ZHANG G, et al. A certificateless signature scheme for mobile wireless cyber-physical systems[C]//2008 The 28th International Conference on Distributed Computing Systems Workshops. [S.l.]: IEEE, 2008: 489-494.

[23] YAP W S, HENG S H, GOI B M. An efficient certificateless signature scheme[C]// International Conference on Embedded and Ubiquitous Computing. [S.l.]: Springer, 2006: 322-331.

[24] ZHANG Z, FENG D. Key replacement attack on a certificateless signature scheme.[J]. IACR Cryptol. ePrint Arch., 2006, 2006:453.

[25] HE D, CHEN J, ZHANG R. An efficient and provably-secure certificateless signature scheme without bilinear pairings[J]. International Journal of Communication Systems, 2012, 25(11): 1432-1442.

[26] TIAN M, HUANG L. Cryptanalysis of a certificateless signature scheme without pairings [J]. International Journal of Communication Systems, 2013, 26(11):1375-1381.

[27] ROWAN S, CLEAR M, GERLA M, et al. Securing vehicle to vehicle communications using blockchain through visible light and acoustic side-channels[J]. arXiv preprint arXiv:1704.02553, 2017.

[28] DORRI A, STEGER M, KANHERE S S, et al. Blockchain: A distributed solution to automotive security and privacy[J]. IEEE Communications Magazine, 2017, 55(12):119-125.

[29] LU Z, WANG Q, QU G, et al. Bars: a blockchain-based anonymous reputation system for trust management in vanets[C]//2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE). [S.l.]: IEEE, 2018: 98-103.

[30] KCHAOU A, ABASSI R, GUEMARA S. Toward a distributed trust management scheme for vanet[C]//Proceedings of the 13th International Conference on Availability, Reliability and Security. [S.l.: s.n.], 2018: 1-6.

[31] QIU Q, LIU S, XU S, et al. Study on security and privacy in 5g-enabled applications[J]. Wireless Communications and Mobile Computing, 2020, 2020.

[32] ZHANG Y, LI J, ZHENG D, et al. Privacy-preserving communication and power injection over vehicle networks and 5g smart grid slice[J]. Journal of Network and Computer Applications, 2018, 122:50-60.

[33] CAO J, MA M, FU Y, et al. Cppha: Capability-based privacy-protection handover authentication mechanism for sdn-based 5g hetnets[J]. IEEE transactions on dependable and secure computing, 2019.

[34] LAI C, LU R, ZHENG D, et al. Security and privacy challenges in 5g-enabled vehicular networks[J]. IEEE Network, 2020, 34(2):37-45.

[35] SAĞLAM E T, BAHTIYAR Ş. A survey: Security and privacy in 5g vehicular networks[C]// 2019 4th International Conference on Computer Science and Engineering (UBMK). [S.l.]: IEEE, 2019: 108-112.

[36] CUI J, ZHANG X, ZHONG H, et al. Rsma: Reputation system-based lightweight message authentication framework and protocol for 5g-

enabled vehicular networks[J]. IEEE Internet of Things Journal, 2019, 6(4):6417-6428.

[37] MA M, HE D, WANG H, et al. An efficient and provably secure authenticated key agreement protocol for fog-based vehicular ad-hoc networks [J]. IEEE Internet of Things Journal, 2019, 6(5): 8065-8075.

[38] CUI J, ZHANG X, ZHONG H, et al. Extensible conditional privacy protection authentication scheme for secure vehicular networks in a multi-cloud environment[J]. IEEE Transactions on Information Forensics and Security, 2019, 15: 1654-1667.

[39] ISLAM S H, OBAIDAT M S, VIJAYAKUMAR P, et al. A robust and efficient password-based conditional privacy preserving authentication and group-key agreement protocol for vanets [J]. Future Generation Computer Systems, 2018, 84:216-227.

[40] HASSAN A, OMALA A A, ALI M, et al. Identity-based user authenticated key agreement protocol for multi-server environment with anonymity[J]. Mobile Networks and Applications, 2019, 24(3):890-902.

[41] GERVAIS M, SUN L, WANG K, et al. Certificateless authenticated key agreement for decentralized wbans[C]//International Conference on Frontiers in Cyber Security. [S.l.]: Springer, 2019: 268-290.

[42] ZHANG J, ZHONG H, CUI J, et al. Edge computing-based privacy-preserving authentication framework and protocol for 5g-enabled vehicular networks[J]. IEEE Transactions on Vehicular Technology, 2020, 69(7):7940-7954.

[43] SAYEED S, MARCO-GISBERT H, CAIRA T. Smart contract: Attacks and protections[J]. IEEE Access, 2020, 8:24416-24427.

[44] ZHANG L, ZHANG F. A method to construct a class of certificateless signature schemes[J]. Chinese Journal of Computers, 2009, 32(5):940-945.

[45] KUMAR P, KUMARI S, SHARMA V, et al. Secure cls and cl-as schemes designed for vanets [J]. The Journal of Supercomputing, 2019, 75 (6):3076-3098.

[46] ALI I, GERVAIS M, AHENE E, et al. A blockchain-based certificateless public key signature scheme for vehicle-to-infrastructure communication in vanets[J]. Journal of Systems Architecture, 2019, 99:101636.

[47] XU Z, HE D, KUMAR N, et al. Efficient certificateless aggregate signature scheme for performing secure routing in vanets[J]. Security and Communication Networks, 2020, 2020.

[48] CUI J, ZHANG J, ZHONG H, et al. An efficient certificateless aggregate signature without pairings for vehicular ad hoc networks[J]. Information Sciences, 2018, 451:1-15.

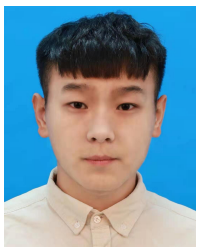
[49] GAYATHRI N, THUMBUR G, KUMAR P R, et al. Efficient and secure pairing-free certificateless aggregate signature scheme for healthcare wireless medical sensor networks[J]. IEEE Internet of Things Journal, 2019, 6(5):9064-9075.

Biographies

Zhihua Wang (Member, IEEE) received the M.S. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2005. He is currently pursuing the Ph.D. degree in communication and information engineering with the University of Science and Technology Beijing, Beijing, China. He is currently an Associate Professor with the School of Cyberspace Security, Zhengzhou University, Zhengzhou, China. His research interests include security of network and information, privacy protection, and information processing.

Shuaibo Wang is an undergraduate student in the School of Cyberspace Security at Zhengzhou University. His research interests include Certificateless cryptography, Artificial intelligence security, Key agreement.

Jiaze Li is an undergraduate student in the School of Cyberspace Security at Zhengzhou University. Her research interest is information security.



Haofan Wang is currently pursuing the master's degree in computer science with Zhengzhou University, Zhengzhou, China. His research interests include security in the Internet of vehicles, Certificateless cryptography and edge computing.

Yizhe Yao is an undergraduate student in the School of Cyberspace Security at Zhengzhou University. His research interests include Blockchain,Key agreement.

Yongjian Wang received the M.Sc. and Ph.D. degrees in communication engineering from the Harbin Institute of Technology, Harbin, China, in 2008 and 2012, respectively. From 2006 to 2008, he was a Research Fellow with Research Room on Communications, National Institute of Advanced Industrial Science and Technology, Tsukuba, Japan. Since 2010, he has been an Associate Professor with the Laboratory of the National Computer Network Emergency Response Technical Team/Coordination Center of China (CNCERT/CC), Beijing, China, where he is currently a Professor. His research interests focus on communication signal processing, cognitive radio, 5G, and the security of Internet of Things (IoT).

Xiaolong Yang (Member, IEEE) received the B.Eng., M.S., and Ph.D. degrees in communication and information systems from University of Electronic Science and Technology of China, Chengdu, China, in 1993, 1996, and 2004, respectively. He is currently a Professor with the School of Computer and Communication Engineering, Institute of Advanced Networking Technologies and Services, University of Science and Technology Beijing, Beijing, China. He has fulfilled more than 30 research projects. He has authored more than 80 articles. His current research interests include the next-generation Internet, network security and defense, and anonymity networking. He holds 16 patents in these areas.