


## Article

# Lightweight Certificate-Less Anonymous Authentication Key Negotiation Scheme in the 5G Internet of Vehicles

Guoheng Wei, Yanlin Qin <sup>\*</sup>, Guangyue Kou  and Zhihong Sun

Department of Information Security, Naval University of Engineering, Wuhan 430030, China

<sup>\*</sup> Correspondence: qinyanlincool@163.com

**Abstract:** In the current 5G vehicle network system, there are security issues such as wireless intrusion, privacy leakage, and remote control. To address these challenges, an improved lightweight anonymous authentication key negotiation scheme based on certificate-less aggregate signatures is proposed and its security and efficiency are analyzed. The result shows that the scheme can offer security attributes including anonymity, traceability, and revocability, as well as effective identity authentication, and it can resist forgery attacks, man-in-the-middle attacks, tampering attacks, and smart card loss attacks. Moreover, compared with similar schemes, it possesses superior security and more efficient computational efficiency and less communication overhead, thereby being more appropriate for high-speed, large-capacity, low-latency, and resource-constrained 5G vehicle network application scenarios.

**Keywords:** Internet of Vehicles; certificate-less aggregate signature; authentication key negotiation scheme; elliptic curve



**Citation:** Wei, G.; Qin, Y.; Kou, G.; Sun, Z. Lightweight Certificate-Less Anonymous Authentication Key Negotiation Scheme in the 5G Internet of Vehicles. *Electronics* **2024**, *13*, 3288. <https://doi.org/10.3390/electronics13163288>

Academic Editors: Yawen Chen, Fei (Travis) Dai and Fabio Grandi

Received: 3 July 2024

Revised: 11 August 2024

Accepted: 14 August 2024

Published: 19 August 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The 5G Internet of Vehicles regards vehicles as the fundamental unit and interacts with roadside infrastructure RSU (Road Side Unit), cloud servers, and humans. It depends on key technologies such as 5G communication technology, sensor technology, information security technology, vehicle autonomy [1], big data and cloud computing technology, and human–computer interaction technology to achieve the efficient unification and information interaction of “vehicle-person-road-cloud”. In the current 5G Internet of Vehicles system, there are numerous nodes, complex channels, an open network, and substantial information interaction. Security issues such as wireless intrusion, privacy leakage, and remote control keep emerging during the application of the Internet of Vehicles, seriously threatening the communication security and data privacy of the Internet of Vehicles. A secure and efficient authentication key negotiation protocol is a crucial means to counter such security attacks. The focus of this paper is to meet the requirements of authentication key negotiation of users in the Internet of Vehicles with a large number of vehicles and reduce the calculation cost of the authentication key negotiation scheme in the meanwhile.

## 2. Related Work

Currently, the authentication key negotiation schemes are mainly based on public key infrastructure, identity-based, and certificate-less cryptography. The authentication scheme based on PKI is the most commonly used one [2–5]. However, the PKI-based authentication scheme struggles to deal with the application scenarios in the Internet of Vehicles, because there are a large number of on-board units (OBUs) in circulation, whose certificates need to be issued, updated, and revoked frequently. At the same time, digital certificates are often broadcast during the interaction process, resulting in a large communication overhead and affecting system efficiency. To address the issue of large public key certificate management overhead in the authentication scheme based on PKI,

Internet of Vehicles authentication key negotiation schemes based on identity have been proposed [6–9]. The identity-based authentication scheme utilizes known user information as the public key to avoid the use of digital certificates, such as identity ID, telephone number, etc. But in the identity-based authentication scheme, the public–private key pair is generated by the key generation center based on user information, thereby causing a key escrow problem. To avoid this problem, scholars have proposed a certificate-less cryptography-based [10] authentication key negotiation scheme [11–14]. In the certificate-less authentication key negotiation scheme, the user controls his own secret value and the partial private key allocated by the KGC together as the private key, thereby avoiding the key escrow problem and also reducing the certificate management overhead. In 2020, Zhang et al. [11] introduced a pre-signature mechanism to achieve the identity authentication of vehicle users and designed an anonymous authentication key negotiation protocol for cloud services in the Internet of Vehicles, but due to the use of relatively fixed temporary identity information, it does not have strong anonymity, and the scheme fails to meet security characteristics such as resistance to temporary key leakage attack, perfect forward security, and resistance to spoofing attack [12]. In 2021, Zhang [13] proposed an efficient anonymous identity authentication and key negotiation scheme based on certificate-less aggregate signatures, but this scheme employs computationally intensive bilinear pair operations, which is not suitable for Internet of Vehicles systems with low latency and low computational overhead requirements. Xiong [14] proposed a lightweight group-based 5G V2X anonymous access authentication and digital transmission scheme, leveraging the advantages of low latency and high reliability of the 5G network to form a temporary group for a certain range of OBUs, combining certificate-less aggregate signature technology and the Chinese remainder theorem to achieve efficient management of group keys. In 2022, Liu et al. [15] proposed an elliptic curve certificate-less anonymous authentication scheme without bilinear pairs that supports batch verification, reducing the computational load of RSUs. However, this scheme is prone to user identity leakage when the vehicle smart card is lost and struggles to withstand spoofing attacks. Wang et al. [16] proposed a certificate-less aggregate signature algorithm for vehicular Ad hoc Network, which lacks anonymity and is unable to resist replay attack and simulation attack. Xi et al. [17] proposed a data sharing and security authentication scheme in the Internet of Vehicles. Compared to the scheme in [16], it has been improved, featuring authenticated identity anonymity and the ability to resist simulation attack, but it still cannot counter replay attack. Ye et al. [18] proposed an aggregate signature algorithm that introduces a timestamp to resist replay attack but still has a key escrow problem. Bao et al. [19] proposed a certificate-less anonymous authentication scheme for VANETs, which utilizes ring signature to achieve strong anonymity. However, it involves bilinear operation with a high calculation cost, and thus, its scalability in large-scale deployments is weak. In 2023, Shahidinejad A et al. [20] proposed a self-certified key exchange protocol for hybrid electric vehicles based on Blockchain. And then, in 2024, Shahidinejad et al. [21] proposed an anonymous lattice-based authentication protocol for vehicular communications; it is a post-quantum scheme.

The key contributions of our work are as follows. To offer more secure and efficient authentication and key negotiation for the 5G Internet of Vehicles with a large number of vehicles, an improved anonymous authentication key negotiation scheme based on certificate-less aggregate signatures is proposed. The scheme adopts a certificate-less public key cryptosystem, and the KGC generates a partial private key of the vehicle user. This, along with a random number generated by the vehicle user itself, constitutes the user's private key, thereby resolving the key escrow issue in the identity-based authentication scheme as well as the large public key certificate management overhead in the authentication scheme based on PKI. To protect the privacy of users, this scheme introduces long-term pseudonyms and short-term pseudonyms. The long-term pseudonym is employed to conceal the real identity of the vehicle user and generate the user's private key, and then the short-term pseudonym is used to hide the vehicle's long-term pseudonym to safeguard the identity privacy of the vehicle user during each authentication process. In the process

of signature authentication and key negotiation, to reduce the system's computational overhead and enhance the interaction efficiency between users, a lighter elliptic curve-based algorithm without the bilinear pair operation is adopted. Security analysis is provided to show that the proposed scheme meets the standard security objectives for the authentication key negotiation protocol. And a formal security proof is also provided to prove that it is secure under the random oracle security model and the assumption of ECDLP. Security and efficiency comparison analysis is provided to demonstrate the superiority of the proposed scheme against state-of-the-art methods.

### 3. Security Objectives and Models

#### 3.1. Security Objectives

This section will define the security objectives that need to be fulfilled by the efficient anonymous identity authentication and key agreement scheme of the 5G vehicle network based on the certificate-less aggregate signature, including the anonymity of vehicles [22], traceability and revocability, effective identity authentication, unlinkability, forward security and backward security, and the ability to resist various attacks.

##### 3.1.1. The Anonymity of Vehicles

The vehicle network system is required to encrypt or conceal the real identity information of vehicles to prevent attackers from obtaining the real identity information of vehicles by using the obtained messages when monitoring the communication channel. This anonymity of vehicles is conditional rather than absolute. Since the identity of vehicles cannot be derived from the exchanged messages, they can still be identified by the network activity.

##### 3.1.2. Traceability and Revocability

The vehicle network system needs to possess the ability to trace false identities and false information and revoke the identities of illegal vehicles.

##### 3.1.3. Effective Message Authentication

A secure vehicle network protocol needs to possess the ability to effectively authenticate information with a large amount of interaction, including the authentication of its timeliness and integrity, thereby ensuring the correctness and reliability of the source of the message.

##### 3.1.4. Unlinkability

Due to the close association among each node in the vehicle network system, it is necessary to guarantee unlinkability to prevent attackers from attacking one attribute and associating it with other secret information.

##### 3.1.5. Forward Security and Backward Security

In the large amount of data interaction in the vehicle network system, it is necessary to prevent attackers from analyzing the historical and future values of this secret information based on the obtained secret information, that is, it should have forward and backward security.

##### 3.1.6. The Ability to Resist Various Attacks

A secure vehicle network authentication key agreement protocol should be able to deal with the primary attack modes, such as eavesdropping attack, tampering attack, replay attack, man-in-the-middle attack, and simulation attack [23].

#### 3.2. Security Model

This section will elaborate on the security model of the proposed vehicle network authentication key agreement scheme [24] based on the certificate-less aggregate signature.

Firstly, we define two types of adversaries, namely  $A_I$  and  $A_{II}$ . Adversary  $A_I$  can query the private key of a legitimate vehicle and can also query and replace the public key of the legitimate vehicle with its own generated public key. Adversary  $A_{II}$  is an internal attacker, equivalent to a malicious but passive KGC, who can query the master key of the KGC and some private keys of the vehicle user, yet is unable to replace the public key.

The attack capabilities of these two types of adversaries are defined through the description of five random oracles:

Hash query, where the adversary queries this oracle and obtains the corresponding hash record on the vehicle list;

Partial private key extraction query, where the adversary makes this query to the oracle and acquires the partial private key on the record list;

Public key extraction query, where the adversary queries this oracle and retrieves the public key on the record list;

Secret value extraction query, where the adversary queries this oracle and obtains the private key of the user on the record list;

Signature query, where the adversary queries this oracle and obtains a legal signature of a message.

Next, we define two games, namely  $Game_0$  and  $Game_1$ . Adversary  $A_I$  plays  $Game_0$  with challenger C, and Adversary  $A_{II}$  plays  $Game_1$  with challenger C.

$Game_0$ : The system parameters  $p$ , adversary  $A_I$ , and challenger C are defined.

1. In the Initialization stage, the challenger C sends the system parameters, excluding the system master key after initialization, to the adversary  $A_I$ , and randomly selects an identity to await the start of the challenge.
2. In the Query stage, the adversary conducts hash queries, partial private key extraction queries, public key extraction queries, secret value extraction queries, signature queries, and other random oracle queries.
3. In the Forgery stage, adversary  $A_I$  generates a forged signature based on the information obtained from the query.

According to the forking lemma [25], if the adversary  $A_I$  successfully outputs three sets of legal signatures using the above queries, then it is said that the adversary  $A_I$  wins this game.

$Game_1$ : The system parameters  $p$ , adversary  $A_{II}$ , and challenger C are defined.

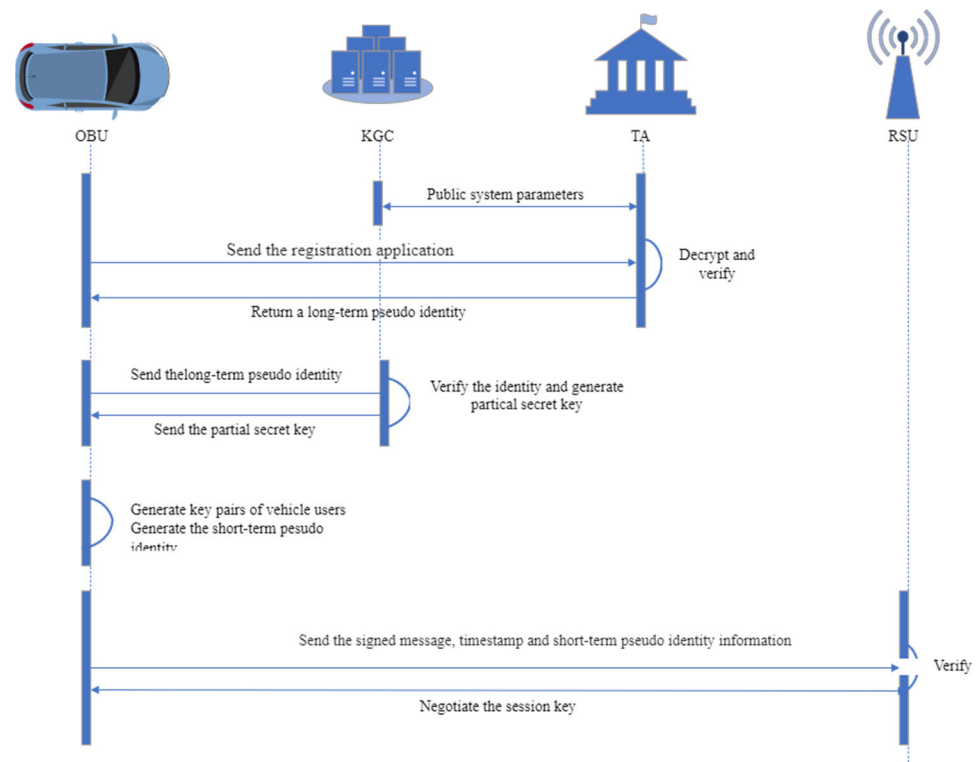
1. In the Initialization stage, the challenger C sends the initialized system parameters to the adversary  $A_{II}$ , and randomly selects an identity to await the start of the challenge.
2. In the Query stage, the adversary conducts hash queries, partial private key extraction queries, public key extraction queries, secret value extraction queries, signature queries, and other random oracle queries.
3. In the Forgery stage, adversary  $A_{II}$  generates a forged signature based on the information obtained from the query.

According to the forking lemma [25], if the adversary  $A_{II}$  successfully outputs three sets of legal signatures using the above queries, then it is said that the adversary  $A_{II}$  wins this game.

## 4. Lightweight Certificate-Less Anonymous Authentication Key Negotiation Scheme

### 4.1. Design of the Scheme

The overall design diagram of the scheme is shown in Figure 1, including the key generation center (KGC), the trusted center (TA), the Road Side Unit (RSU), and the on-board unit (OBU). The brief description of the scheme is as follows.



**Figure 1.** Design of the lightweight authentication key negotiation protocol based on certificate-less aggregate signature in 5G Internet of Vehicles.

**System initialization:** KGC and TA generate the public parameters and public–private key pairs of the system.

**Generation of long-term pseudonym:** The TA verifies the identity information of  $OBU_i$  and password  $PW_i$  in the system, and outputs the long-term pseudonym  $PID_i$  of the vehicle according to the identity information of the vehicle  $OBU_i$ , and sends it to the vehicle  $OBU_i$  through a secure channel, and the vehicle  $OBU_i$  stores the long-term pseudonym on the smart card.

**Generation of partial private key:** The vehicle  $OBU_i$  secretly sends the long-term pseudonym to the KGC, and the KGC generates the partial private key  $psk_i$  of the vehicle, and sends it to the TA and  $OBU_i$  through a secure channel.

**Generation of the vehicle public–private key pair:** After the vehicle successfully validates the partial private key, it combines this partial private key to generate the vehicle public–private key pair.

**Generation of short-term pseudonym:** The vehicle generates a short-term pseudonym  $LID_i$  based on the real identity  $OID_i$  and the long-term pseudonym  $PID_i$ .

**Signature:**  $OBU_i$  use the partial private key generated by the KGC and the secret value generated by itself to sign the message.

**Single verification:** The message recipient verifies the legitimacy of the signature on the message, and if it is legal, it outputs as true.

**Key negotiation:** The vehicle  $OBU_i$  negotiates a pair of safe and correct shared secret values  $K_{UR}$  and  $K_{RU}$  with  $RSU_i$ , and performs a calculation to obtain the shared session key SK.

**Aggregate signature:** Aggregate the message signatures of the vehicle  $OBU_i, OBU_i, \dots, OBU_n$  and output the aggregate signature  $\sigma = R_1, R_2, \dots, R_n, \tau$ .

**Batch verification:** Verify the legitimacy of the aggregate signature  $\sigma = R_1, R_2, \dots, R_n, \tau$  of the aggregated message  $M_1, M_2, \dots, M_n$  and if it is legal, it outputs as true.

A more detailed description will be given in Section 4.2.

#### 4.2. Description of the Scheme

This section will give a detailed description of the scheme. The lightweight authentication key negotiation protocol based on certificate-less aggregate signature includes the Algorithms 1–9.

---

##### Algorithm 1. System initialization.

---

Both KGC and TA are the trusted third parties that generate the public parameters and the public–private key pairs of the system. The system initialization algorithm is as follows:

1. TA constructs the elliptic curve  $G$  and the generator  $P$  on the elliptic curve.
  2. TA constructs a random number  $\alpha \in Z_q^*$ , calculates  $T_p = \alpha \cdot P$ , where TA secretly stores  $\alpha$  as its own master key.
  3. KGC constructs a random number  $\beta \in Z_q^*$ , calculates  $P_p = \beta \cdot P$ , where KGC secretly stores  $\beta$  as its own master key.
  4. Use  $\alpha$  and  $\beta$  jointly to form the master key of the Internet of Vehicles system.
  5. KGC selects hash functions  $H_1, H_2, H_3, H_4$ , where  $H_1 : \{0, 1\}^* \rightarrow Z_q^*$ ,  
 $H_2 : \{0, 1\}^* \times G \times G \rightarrow Z_q^*$ ,  $H_3 : G \times \{0, 1\}^* \rightarrow \{0, 1\}^*$ ,  
 $H_4 : G \times G \times \{0, 1\}^* \times G \times G \rightarrow Z_q^*$ , and TA and KGC publish the public parameter  
 $p = \{G, P, P_p, T_p, H_1, H_2, H_3, H_4\}$ .
- 

---

##### Algorithm 2. Long-term pseudonym generation.

---

The long-term pseudonym generation stage is divided into two parts; one part is for the TA to verify the real identity of the vehicle, and the other part is for the TA to generate a long-term pseudonym  $PID_i$  for the vehicle.

1. The vehicle inputs its real identity  $OID_i$  and password  $PWD_i$ , takes  $T_p$  as the public key of the encryption algorithm, and sends the encrypted identity information  $ID_i$  and  $PW_i$  to TA. Then, TA decrypts it to obtain the real identity information of the vehicle and compares it with the information in the vehicle list  $U_L$  to verify the legitimacy. If it exists in the list and the information is true, the vehicle can be regarded as a legitimate vehicle.
  2. TA decrypts the information sent by vehicle  $OBU_i$  to obtain the real identity of the vehicle  $OID_i$ , selects a random number  $t \in Z_q^*$ , calculates the long-term pseudonym  $PID_i = (t + \alpha H_1(OID_i)) \bmod q$ , and stores it in the smart card, then distributes the smart card to the vehicle.
- 

---

##### Algorithm 3. Key generation.

---

KGC and  $OBU_i$  perform the following operations to generate the public–private key pair of the vehicle.

1.  $OBU_i$  encrypts the long-term pseudonym with the public key  $P_p$  and transmits the encrypted message to KGC.
  2. KGC decrypts the message to obtain the long-term pseudonym of the vehicle; it should verify whether the long-term pseudonym exists in the vehicle list  $U_L$ . If the verification is successful, KGC selects a random number  $r_i \in Z_q^*$ , calculates  $R_i = r_i \cdot P$ ,  
 $h_i = H_2(PID_i \| R_i \| P_p)$ ,  $s_i = (r_i + h_i \cdot \beta) \bmod q$ , and generates the partial private key  $psk_i = (s_i, R_i)$  of the vehicle  $OBU_i$ .
  3. KGC sends  $(OID_i, PID_i, psk_i)$  to TA through a secure channel, and TA stores it in the vehicle list  $U_L$ .
  4. KGC sends the partial private key  $psk_i$  to the vehicle  $OBU_i$  through a secure channel.
  5. After the vehicle  $OBU_i$  receives the partial private key, it first verifies whether  $s_i \cdot P = R_i + h_i \cdot P_p$  is established, to obtain the legitimacy of  $psk_i$ . If it is established,  $psk_i$  can be regarded as the available partial private key.
  6.  $OBU_i$  selects a random number  $a_i \in Z_q^*$  as the other partial private key of the vehicle, so the public–private key pair of the vehicle is  $pk_i = a_i \cdot P$ ,  $sk_i = (a_i, psk_i)$ .
-

**Algorithm 4.** Short-term pseudonym generation.

$OBU_i$  generates the one-time short-term pseudonym  $LID_i$  using the real identity of the vehicle and the long-term pseudonym according to the following steps. The short-term pseudonym  $LID_i$  of the vehicle consists of two parts as follows.

1. The first part is  $LID_1^i$ .  $OBU_i$  randomly selects  $\delta \in Z_q^*$ , and calculates  $LID_1^i = \delta \cdot P$ .
2. The second part is composed of the real identity of the vehicle, the long-term pseudonym, the password, and the timestamp  $T_i$ .  $OBU_i$  is calculated.
3.  $LID_2^i = OID_i \oplus H_3[(\delta \cdot T_p) || PID_i || T_i || H_1(PWD_i)]$ .

Therefore, the short-term pseudonym  $LID_i$  of the vehicle is as follows:

$$LID_i = (LID_1^i, LID_2^i, T_i) = \begin{cases} LID_1^i = \delta \cdot P \\ LID_2^i = OID_i \oplus H_3[(\delta \cdot T_p) || PID_i || T_i || H_1(PWD_i)] \end{cases} \quad (1)$$

**Algorithm 5.** Signature generation.

$OBU_i$  constructs the signature of the vehicle according to the following steps and broadcasts the signed message to all other members in the system.

1.  $OBU_i$  randomly selects  $\gamma_i \in Z_q^*$ , and calculates  $D_i = \gamma_i \cdot P$ .
2.  $OBU_i$  calculates  $w_i = H_4(D_i, LID_1^i, LID_2^i, T_j, pk_i, R_i, P_p)$ ,  
 $v_i = H_4(m_i, D_i, LID_1^i, LID_2^i, T_j, pk_i, R_i, P_p)$ ,  $\tau_i = \gamma_i + v_i(w_i \cdot a_i + s_i) \bmod q$ .
3.  $OBU_i$  constructs the signature of the vehicle, that is, the tuple  $\sigma_i = (R_i, D_i, \tau_i)$ .
4.  $OBU_i$  constructs the tuple  $M_i = (LID_i, pk_i, m_i, h_i, \sigma_i, T_j)$ , and broadcasts this tuple to all other members in the Internet of Vehicles system.

**Algorithm 6.** Single verification.

After the message recipient (taking  $RSU_i$  as an example) receives the signed message broadcasted by  $OBU_i$ , it verifies the signature. If the verification result is true, the identity of the message sender is authenticated as legal; otherwise, the message sender is an illegal user.

1.  $RSU_i$  verifies whether the timestamp  $T_i$  is valid, to ensure the freshness of the short-term pseudonym  $LID_i$  of the vehicle  $OBU_i$ .
2.  $RSU_i$  verifies whether the timestamp  $T_j$  is valid. If it is invalid, the information will be discarded directly. If it is valid, the subsequent steps will be carried out.
3.  $RSU_i$  calculates  $w'_i = H_4(D_i, LID_1^i, LID_2^i, T_j, pk_i, R_i, P_p)$ ,  
 $v'_i = H_4(m_i, D_i, LID_1^i, LID_2^i, T_j, pk_i, R_i, P_p)$ .
4.  $RSU_i$  verifies whether the equation  $\tau_i \cdot P = D_i + v'_i(w'_i \cdot pk_i + R_i + h_i \cdot P_p)$  is established. If it is established, it is considered that the output is true and the vehicle's identity is legal and trustworthy; otherwise, the vehicle's identity is illegal and untrustworthy. The correctness proof is as follows:

$$\begin{aligned} \tau_i \cdot P &= [\gamma_i + v_i(w_i \cdot a_i + s_i)] \cdot P \\ &= \gamma_i \cdot P + v_i(w_i \cdot a_i + s_i) \cdot P \\ &= D_i + v_i(w_i \cdot a_i \cdot P + s_i \cdot P) \\ &= D_i + v_i(w_i \cdot pk_i + s_i \cdot P) \\ &= D_i + v_i[w_i \cdot pk_i + (r_i + h_i \cdot \beta) \cdot P] \\ &= D_i + v_i[w_i \cdot pk_i + (r_i \cdot P + h_i \cdot \beta \cdot P)] \\ &= D_i + v_i(w_i \cdot pk_i + R_i + h_i \cdot P_p) \end{aligned} \quad (2)$$

If the information in the broadcast message  $M_i = (LID_i, pk_i, m_i, h_i, \sigma_i, T_j)$  has not been modified, then  $w'_i = w_i$ ,  $v'_i = v_i$ , which means

$$D_i + v'_i(w'_i \cdot pk_i + R_i + h_i \cdot P_p) = D_i + v_i(w_i \cdot pk_i + R_i + h_i \cdot P_p) = \tau_i \cdot P \quad (3)$$

**Algorithm 7.** Key negotiation.

In this section, the two communicating parties need to negotiate a same-session key (taking the communication between  $OBU_i$  and  $RSU_i$  as an example). After the signature authentication, the key negotiation scheme is designed, as shown in Figure 2.

As shown in Figure 2,  $OBU_i$  and  $RSU_i$  take the following steps to negotiate the same-session key:

1.  $OBU_i$  selects a random number  $k_U \in Z_q^*$ , calculates  $h_U = H_2(LID_U \| R_U \| pk_U)$ ,  
 $TU = (k_U \cdot (a_U + s_U h_U))P$ .
2.  $OBU_i$  selects a random number  $k_R \in Z_q^*$ , calculates  $h_R = H_2(ID_R \| R_R \| pk_R)$ ,  
 $TR = (k_R \cdot (a_R + s_R h_R))P$ .
3.  $OBU_i$  sends  $TU$  to  $RSU_i$ , and at the same time,  $RSU_i$  sends  $TR$  to  $OBU_i$ .
4. After receiving  $TR$ ,  $OBU_i$  calculates the shared secret value  $K_{UR} = (k_U \cdot (a_U + s_U h_U))TR$ .
5. After receiving  $TU$ ,  $RSU_i$  calculates the shared secret value  $K_{RU} = (k_R \cdot (a_R + s_R h_R))TU$ .

Then, the same-session key can be calculated, respectively, by  $OBU_i$  and  $RSU_i$ , as follows:

$$SK = H_4(LID_U, TU, TR, K_{UR}) = H_4(LID_U, TU, TR, K_{RU}) \tag{4}$$

The correctness analysis is as follows:

$$\begin{aligned} K_{UR} &= (k_U \cdot (a_U + s_U h_U))TR \\ &= (k_U \cdot (a_U + s_U h_U))(k_R \cdot (a_R + s_R h_R))P \\ &= (k_R \cdot (a_R + s_R h_R))(k_U \cdot (a_U + s_U h_U))P \\ &= (k_R \cdot (a_R + s_R h_R))TU \end{aligned} \tag{5}$$

**Algorithm 8.** Aggregate signature.

When there are  $n$  messages in the system that need to be signed, the  $n$  signatures for these  $n$  messages are aggregated by  $RSU_i$ . The messages and signatures are  $M_1, \sigma_1 = (R_1, D_{n1}, \tau_1)$ ,  $M_2, \sigma_2 = (R_2, D_2, \tau_2) \dots, M_n, \sigma_n = (R_n, D_n, \tau_n)$ , and the aggregated certificate-less signature is  $\sigma = R_1, D_1, R_2, D_2, \dots, R_n, D_n, \tau$ , where  $\tau = \sum_{i=1}^n \tau_i$ .

**Algorithm 9.** Batch verification.

After the message recipient receives the aggregated certificate-less signature  $\sigma$ , it verifies the aggregated signature. If the verification is passed, the aggregated signature is considered legal; otherwise, it is not legal. The verification steps are as follows:

1. Calculate  $\tau' = \sum_{i=1}^n \tau'_i$ , and if  $\tau' = \tau$ , then the aggregated signature is considered valid; otherwise, the aggregated signature is invalid, and it will be discarded directly.
2. Calculate  $w'_i, v'_i, i = 1, 2, \dots, n$ , by following the steps in Algorithm 6.
3. Verify the equation

$$\tau' \cdot P = \sum_{i=1}^n D_i + \sum_{i=1}^n v'_i (w'_i \cdot pk_i + R_i + h_i \cdot P_p) \tag{6}$$

If it is established, the aggregated certificate-less signature is considered legal.



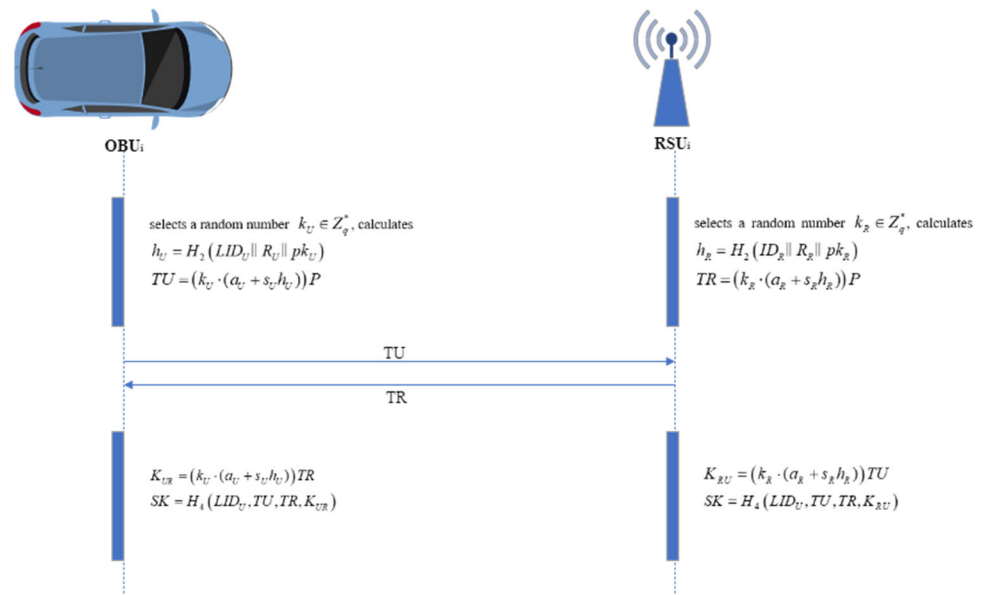


Figure 2. Key negotiation stage.

## 5. Security Analysis and Proof

### 5.1. Security Analysis

This section will analyze whether the protocol proposed in this paper meets the standard security objectives proposed in Section 3.1.

#### 5.1.1. Anonymity of Vehicles

During communication, the vehicles need to conceal their initial real identities. When a vehicle applies to the TA for a long-term pseudonym, it uses an encryption algorithm to encrypt the real identity  $OID_i$  with the encryption key  $T_p$ . Therefore, the attacker cannot calculate the real identity of the vehicle without knowing the master key  $\alpha$  of the TA; the TA calculates and distributes the long-term pseudonym  $PID_i = (t + \alpha H_1(OID_i)) \bmod q$ . Due to the one-way nature of the hash function and the fact that the attacker cannot obtain the random number  $t$ , the attacker cannot calculate the real identity  $OID_i$  of the vehicle through the long-term pseudonym  $PID_i$ , ensuring the anonymity of the real identity  $OID_i$  of the vehicle. To further ensure the security of the vehicle’s identity, this paper introduces the concept of a vehicle’s short-term pseudonym, so that the identity of the vehicle in the communication process is *sessionized*, like the session key. The short-term pseudonym consists of the real identity, long-term pseudonym and password of the vehicle, and a timestamp, which can be calculated as

$$LID_i = (LID_1^i, LID_2^i, T_i) = \begin{cases} LID_1^i = \delta \cdot P \\ LID_2^i = OID_i \oplus H_3[(\delta \cdot T_p) \| PID_i \| T_i \| H_1(PWD_i)] \end{cases} \quad (7)$$

If the attacker wants to obtain the real identity of the vehicle, he needs to calculate the equation  $OID_i = LID_2^i \oplus H_3[(\delta \cdot T_p) \| PID_i \| T_i \| H_1(PWD_i)]$ . However, the attacker cannot obtain  $\delta$ ; even if he obtains the smart card which stores the long-term pseudonym of the vehicle, he cannot obtain the password of the vehicle, so the attacker cannot calculate the result of this equation and thus cannot obtain the real identity  $OID_i$ , that is, the anonymity of the vehicle can be ensured.

#### 5.1.2. Traceability and Revocability

TA usually stores and maintains the vehicle list  $U_L$ , so when a vehicle has a dispute or other untrusted behavior during the communication process, TA can compare the suspect identity information and long-term or short-term pseudonym with the vehicle list  $U_L$ , and then it can calculate  $OID_i = LID_2^i \oplus H_3[(\alpha \cdot LID_1^i) \| PID_i \| T_i \| H_1(PWD_i)]$  to trace the real

identity and legality of this disputed vehicle; thus, the user behavior can be traced and verified in the Internet of Vehicles system. And the illegal vehicles will be revoked from the vehicle list. Therefore, this protocol can guarantee the traceability and revocability of the Internet of Vehicles system.

### 5.1.3. Effective Message Authentication

All of the communication entities in the  $RSU_i$  or domain of  $RSU_i$  can verify the legitimacy of the message  $m_i$  by validating the pseudonym  $LID_i$ , signature  $\sigma_i$ , and other information of the vehicle  $OBU_i$ .

In the Internet of Vehicles system, if the vehicle  $OBU_i$  wants to communicate with other entities, it first needs to be verified by the TA. Then, the TA will distribute a long-term pseudonym  $PID_i$  to the vehicle  $OBU_i$ , and at the same time, the TA will also save the  $OBU_i$  to the vehicle list  $U_L$ . Therefore, it can ensure that the attacker cannot attack the real identity of the vehicle during the communication process. Then, the vehicle can use its short-term pseudonym  $LID_i$ , which is composed of the real identity  $OID_i$ , long-term pseudonym  $PID_i$ , timestamp, and password, to interact with other communication entities. After the message recipient receives the message, it should first check the timestamp. If the time has expired, the message will be discarded; otherwise, it then checks whether the equation  $\tau_i \cdot P = D_i + v'_i(w'_i \cdot pk_i + R_i + h_i \cdot P_p)$  holds. If it holds, it can be considered that the signature is authenticated, and the message sender can be recognized as a legal user. If it does not hold, then it is considered that the message sender is illegal, and the message will be discarded. Therefore, the protocol designed in this paper can complete the authentication in terms of timeliness and integrity.

### 5.1.4. Unlinkability

In the certificate-less authentication scheme proposed in this paper, by constructing different random numbers  $t, \delta$ , the real identity  $OID_i$  of the vehicle is hidden during the communication with the long-term pseudonym  $PID_i$  and the short-term pseudonym  $LID_i$ . The existence of the random numbers  $t, \delta$  reduces the correlation between the long-term pseudonym  $PID_i$  and the short-term pseudonym  $LID_i$ , and the attacker cannot obtain one pseudonym through linking another pseudonym. Therefore, the protocol in this section can meet the unlinkability requirement of the Internet of Vehicles system.

### 5.1.5. Forward Security and Backward Security

The attacker may intercept the signature  $\sigma_i = (R_i, D_i, \tau_i)$  of the vehicle  $OBU_i$ , where  $\tau_i = \gamma_i + v_i(w_i \cdot a_i + s_i) \bmod q$ , and  $\gamma_i$  is randomly selected, so the attacker cannot obtain the previous and future signatures through the signature currently obtained. In addition, for the session key  $SK = H_4(LID_U, TU, TR, K_{UR}) = H_4(LID_U, TU, TR, K_{RU})$ , due to the existence of the random numbers  $k_U, k_R$  and the one-way nature of the hash function, the attacker also cannot obtain the previous and future session keys by using the current session key. Therefore, the proposed authentication key agreement scheme in this section satisfies the requirements for forward security and backward security.

### 5.1.6. Ability to Resist Attacks

The following analysis shows the proposed scheme's ability to resist regular attacks against the authentication key negotiation protocol.

1. Replay attack: In the authentication key agreement scheme designed in this section, a timestamp is introduced. During each authentication, the validity of the timestamp is first checked, and if it is valid, the subsequent steps will be carried out; otherwise, this message will be discarded. In this scheme, two timestamps need to be added. The first timestamp is added when generating the short-term pseudonym of the vehicle  $OBU_i$ , to ensure the timeliness of the short-term pseudonym. The second timestamp is added when broadcasting the message  $M_i = (LID_i, pk_i, m_i, h_i, \sigma_i, T_j)$ , to ensure the timeliness of the broadcast message. The introduction of timestamps can effectively

prevent the attacker from repeatedly sending the messages of the two communication parties in the channel and prevent the attacker from obtaining the secret information he expected. Therefore, the authentication key agreement protocol in this article can effectively resist replay attacks.

2. Man-in-the-middle attack: This protocol adopts a certificate-less authentication method, relying on the difficult problem of ECDLP. The adversary cannot completely simulate a vehicle to generate message  $\{LID_i, pk_i, m_i, h_i, \sigma_i, T_j\}$  as a middleman, and the vehicle uses the public parameters published by the trusted party in the communication process, so there is no opportunity for a middleman to deceive the communication participants. Therefore, this scheme can resist man-in-the-middle attacks.
3. Tampering attack: In this scheme, the message  $\{LID_i, pk_i, m_i, h_i, \sigma_i, T_j\}$  broadcasted by  $OBU_i$  is signed, where  $\sigma_i$  is the digital signature, which can ensure the integrity of the message. At the same time, this scheme has a traceability mechanism for suspicious information and identities. When the attacker tampers with the message in the communication channel, the traceability of the user identity can help to discover whether the information has been tampered with by the attacker. Therefore, the authentication scheme proposed in this article can resist tampering attacks.
4. Simulation attack: In the simulation attack, the attacker may imitate the structure of the pseudonym to disguise himself as a legitimate vehicle. However, under the assumption of ECDLP, the attacker cannot obtain the master key, which is protected by the trusted part, so he cannot forge a standardized pseudonym. Therefore, this scheme has the ability to resist simulation attacks.
5. Eavesdropping attack: Though malicious eavesdropping attacks on the Internet of Vehicles system are continuous and the occurrence of the eavesdropping behavior cannot be prevented, the proposed authentication key agreement scheme uses a secure channel or encryption to protect the secret information, and the session key negotiated will play an encryption role when the communication entities conduct dialogue interaction. Therefore, this scheme can prevent malicious attackers from obtaining confidential information and user privacy through eavesdropping.

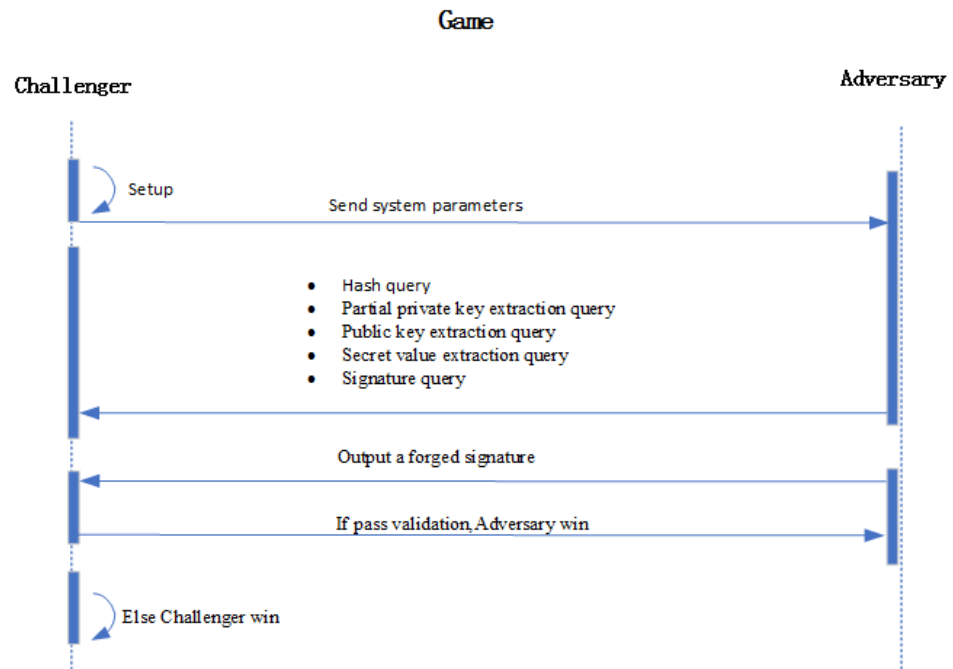
## 5.2. Security Proof

This section will formally prove that the proposed scheme is secure under the security model in Section 3.2 and the assumption that the Elliptic Curve Discrete Logarithm Problem (ECDLP) is difficult to resolve.

**Definition 1.** *If two types of adversaries  $A_I$  and  $A_{II}$  win the game with a non-negligible probability in polynomial time to solve the ECDLP, then this scheme satisfies nonforgeability in the random oracle model and is computationally secure.*

**Theorem 1.** *In a probabilistic polynomial time, assuming that the adversary  $A_I$  performs the game for time  $t$ , performs  $Q(h)$  hash queries,  $Q(sk)$  partial private key extraction queries,  $Q(pk)$  partial private key extraction queries, and  $Q(\sigma)$  signature queries, and finally forges a legal signature with an advantage as  $\epsilon$ , then within time  $t' \leq t + O(Q(1) + Q(3) + (Q(2) + Q(4) + Q(sk) + Q(pk) + Q(\sigma))t_s)$ , the adversary needs to solve the difficult ECDLP problem with a probability not lower than  $\frac{\epsilon}{Q(sk)} \left(1 - \frac{1}{Q(sk)}\right)^{Q(sk)}$ , where  $t_s$  represents the operation time of a scalar multiplication on the elliptic curve group  $G$ .*

**Proof of Theorem 1.** The outline of the proof is shown in Figure 3. The specific derivation process is as follows:



**Figure 3.** Outline of proof of Theorem 1.

Construct  $P, Q = aP$ , where  $a$  is a randomly selected number,  $a \in Z_q^*$ , and  $P$  is a generator on the elliptic curve  $G$ . According to the assumption of the ECDLP, the challenger  $C$  needs to find  $a$  in order to solve the ECDLP.

1. Initialization stage: The challenger  $C$  initializes the system parameters and sends them to the adversary, and the challenger  $C$  randomly selects an identity  $ID'$  as its challenge identity in this game, and the system parameters are  $p = \{G, P, P_p = Q, T_p, H_1, H_2, H_3, H_4\}$ .
2. Query stage: The adversary  $A_I$  will perform the following random oracle queries.
  - $H_1$  query: When the adversary  $A_I$  queries this oracle, the challenger records the interaction between the adversary  $A_I$  and the challenger  $C$  in the list  $L_1 = (ID_i, H_1(ID_i))$ . When the challenger  $C$  finds the corresponding record in the list  $L_1$ , it returns  $H_1(ID_i)$  to the adversary  $A_I$ ; otherwise, it randomly selects  $H_1(ID_i) \in Z_q^*$  and gives it to the adversary  $A_I$ , and adds  $(ID_i, H_1(ID_i))$  to the list  $L_1$ .
  - $H_2$  query: When the adversary  $A_I$  queries this oracle, the challenger records the interaction between the adversary  $A_I$  and the challenger  $C$  in the list  $L_2 = (PID_i || R_i || P_p, r_i)$ . When the challenger  $C$  finds the corresponding record in the list  $L_2$ , it returns  $r_i$  to the adversary  $A_I$ ; otherwise, it randomly selects  $r_i \in Z_q^*$  and gives it to the adversary  $A_I$ , and adds  $(PID_i || R_i || P_p, r_i)$  to the list  $L_2$ .
  - $H_3$  query: When the adversary  $A_I$  queries this oracle, the challenger records the interaction between the adversary  $A_I$  and the challenger  $C$  in the list  $L_3 = ((\delta \cdot T_p) || PID_i || T_i || H_1(PWD_i), u_i)$ . When the challenger  $C$  finds the corresponding record in the list  $L_3$ , it returns  $u_i$  to the adversary  $A_I$ ; otherwise, it randomly selects  $u_i \in Z_q^*$  and gives it to the adversary  $A_I$ , and adds  $((\delta \cdot T_p) || PID_i || T_i || H_1(PWD_i), u_i)$  to the list  $L_3$ .
  - $H_4$  query: When the adversary  $A_I$  queries this oracle, the challenger records the interaction between the adversary  $A_I$  and the challenger  $C$  in the list  $L_4 = (D_i || LID_1^i || LID_2^i || T_j || pk_i || R_i || P_p, l_i)$ . When the challenger  $C$  finds the corresponding record in the list  $L_4$ , it returns  $l_i$  to the adversary  $A_I$ ; otherwise, it randomly selects  $l_i \in Z_q^*$  and gives it to the adversary  $A_I$ , and adds  $(D_i || LID_1^i || LID_2^i || T_j || pk_i || R_i || P_p, l_i)$  to the list  $L_4$ .

- Partial private key extraction query: When the adversary  $A_I$  queries this oracle, the challenger records the interaction between the adversary  $A_I$  and the challenger  $C$  in the list  $L_{sk} = (PID_i, psk_i)$ . When the challenger  $C$  finds the corresponding record in the list  $L_{sk}$ , it returns  $l_i$  to the adversary  $A_I$ ; otherwise, if  $PID_i \neq PID'_i$ , it randomly selects  $psk_i \in Z_q^*$  and gives it to the adversary  $A_I$ , and adds  $(PID_i, psk_i)$  to the list  $L_{sk}$ , and if  $PID_i = PID'_i$ , the game ends.
  - Public key extraction query: When the adversary  $A_I$  queries this oracle, the challenger records the interaction between the adversary  $A_I$  and the challenger  $C$  in the list  $L_{pk} = (PID_i, a_i, pk_i)$ . When the challenger  $C$  finds the corresponding record in the list  $L_{pk}$ , it returns  $pk_i$  to the adversary  $A_I$ ; otherwise, if  $PID_i \neq PID'_i$ , it randomly selects  $a_i \in Z_q^*$  and gives it to the adversary  $A_I$ , and adds  $(D_i \parallel LID_1^i \parallel LID_2^i \parallel T_j \parallel pk_i \parallel R_i \parallel P_p)$  and  $(PID_i, a_i, pk_i)$  to  $L_{sk}$  and  $L_{pk}$ , respectively.
  - Secret value extraction query: When the adversary  $A_I$  queries this oracle, if  $PID_i = PID'_i$ , the challenger  $C$  quits and ends the game; otherwise, if there is a record  $(PID_i, a_i, pk_i)$ , the challenger  $C$  returns  $a_i$  to the adversary  $A_I$ , and otherwise, the challenger  $C$  adds the record  $(a_i, pk_i)$  to the list  $L_{pk}$  and returns  $a_i$  to the adversary  $A_I$ .
  - Signature query: When the adversary  $A_I$  queries this oracle, the challenger  $C$  obtains  $H_1(PID_i)$ ,  $H_2(PID_i \parallel R_i \parallel P_p)$ ,  $H_3((\delta \cdot T_p) \parallel PID_i \parallel T_i \parallel H_1(PWD_i))$ ,  $H_4(D_i \parallel LID_1^i \parallel LID_2^i \parallel T_j \parallel pk_i \parallel R_i \parallel P_p)$  from the lists  $L_1, L_2, L_3, L_4$ , respectively. If  $PID_i \neq PID'_i$ , the challenger  $C$  outputs the signature  $\sigma_i$  corresponding to the message  $m_i$  and returns it to the adversary  $A_I$ ; otherwise, it calculates  $D_i = \gamma_i \cdot P$ ,  $w_i = H_4(D_i, LID_1^i, LID_2^i, T_j, pk_i, R_i, P_p)$ ,  $v_i = H_4(m_i, D_i, LID_1^i, LID_2^i, T_j, pk_i, R_i, P_p)$ ,  $\tau_i = \gamma_i + v_i(w_i \cdot a_i + s_i) \bmod q$ , and returns the correct signature  $\sigma_i = (R_i, D_i, \tau_i)$  of the message  $m_i$  to the adversary  $A_I$ .
3. Forgery stage: After the adversary  $A_I$  completes the above queries, it outputs a forged signature. If  $PID_i = PID'_i$ , the challenger  $C$  ends the game; otherwise, if the adversary wants to win the game, it needs to find out the corresponding signature information from the information obtained from the queries, and it needs to make the equation  $\tau_i \cdot P = D_i + v'_i(w'_i \cdot pk_i + R_i + h_i \cdot P_p)$  hold.

According to the forking lemma [25], the adversary  $A_I$  also needs to obtain two other valid signatures  $\sigma_i^{(\lambda)}$ ,  $(\lambda = 2, 3)$ , and all three signatures need to make the equation  $\tau_i \cdot P = D_i + v'_i(w'_i \cdot pk_i + R_i + h_i \cdot P_p)$  hold. Since there is  $pk_i = a_i \cdot P$ ,  $P_p = \beta \cdot P$ ,  $D_i = \gamma_i \cdot P$ , then  $\tau_i^\lambda \cdot P = D_i + v'_i(w'_i \cdot pk_i^\lambda + R_i + h_i \cdot P_p)$ ,  $\lambda = 1, 2, 3$ .

The challenger  $C$  needs to solve this linearly independent equation and output  $a$  as the solution to the ECDLP.

In the partial private key extraction stage, the challenger  $C$  has a probability of at least  $(1 - \frac{1}{Q(sk)})^{Q(sk)}$  to not abandon the operation, and in the forgery stage, the challenger  $C$  has a probability of at least  $\frac{1}{Q(sk)}$  to not abandon the operation. Therefore, the challenger  $C$  successfully solves the ECDLP within time  $t' \leq t + O(Q(1) + Q(3) + (Q(2) + Q(4) + Q(sk) + Q(pk) + Q(\sigma))t_s)$  with a probability of at least  $\frac{\epsilon}{Q(sk)} (1 - \frac{1}{Q(sk)})^{Q(sk)}$ . Since the adversary  $A_I$  cannot win the game with a negligible probability in polynomial time, then this scheme has nonforgeability security in the random oracle model.  $\square$

The proof is completed.

**Theorem 2.** In a probabilistic polynomial time, assuming that the adversary  $A_{II}$  performs the game for time  $t$ , performs  $Q(h)$  hash queries,  $Q(x)$  secret value extraction queries,  $Q(pk)$  partial private key extraction queries, and  $Q(\sigma)$  signature queries, and finally forges a legal signature with an advantage as  $\epsilon$ , then within time  $t' \leq t + O(Q(1) + Q(3) + (Q(2) + Q(4) + Q(pk) + Q(\sigma))t_s)$ , the

adversary needs to solve the ECDLP with a probability not lower than  $\frac{\epsilon}{Q(pk)} \left(1 - \frac{1}{Q(pk)}\right)^{Q(pk)+Q(x)}$ , where  $t_s$  represents the operation time of a scalar multiplication on the elliptic curve group  $G$ .

**Proof of Theorem 2.** The proof process is similar to that of Theorem 1, but the adversary  $A_{II}$  does not have the ability to perform partial private key extraction queries, and this will not be elaborated here.  $\square$

## 6. Discussion of Performance

This section will conduct a performance comparison and analysis of the proposed authenticated key agreement scheme in this paper from the aspects of security, computational overhead, and communication overhead.

### 6.1. Security Comparison

The following part will compare the protocol in this paper with the protocols proposed in other studies from the perspective of security. Table 1 shows the comparison results of this scheme and other similar schemes [15–18] in terms of anonymity, traceability and revocability, identity privacy, message authenticity, unlinkability, resistance to man-in-the-middle attack, resistance to replay attack, resistance to simulation attack, key escrow resilience [26], and batch verification.

The security comparison is shown in Table 1. The aggregate signature scheme proposed in [16] does not have anonymity, for the real identity of the vehicle is used in the authentication process. And it is proved in [17] that the scheme in [16] cannot resist replay attack and simulation attack, with relatively low security. These drawbacks make it unsuitable for large-scale IoV networks with high complexity and uncertainty. The scheme in [17] has been greatly improved compared to [16]; it possesses higher security for its identity anonymity in the authentication and the ability to resist simulation attack, but it still cannot resist replay attack, for it does not include a timestamp in the signature message. The scheme in [18] introduces a timestamp to resist replay attack, but it is constructed by using identity-based cryptography, and all user keys are generated by a third party, which will lead to a key escrow problem; this drawback makes it unsuitable for large-scale 5G IoV networks. The scheme proposed in [15] stores the vehicle's real identity and password in an anti-tampering smart card, which is costly. Furthermore, once the smart card is lost, the user's identity will be exposed. At the same time, when the trusted authority calculates the user's long-term fake identity, it does not use its own master key, which means that the malicious user without the trusted authority's master key can simulate and forge the long-term fake identity of the vehicle. In addition, when the KGC generates a partial private key for the user, it does not use a one-time random number, so attackers can analyze the historical and future values of the partial private key for the vehicle based on the obtained partial private key value, that is, the scheme proposed in [15] does not meet the requirement of forward security and backward security. According to the security analysis and proof in Section 5, the authenticated key agreement scheme proposed in this paper can solve the security problems in [15–18] and can resist more types of attacks. Vehicle users also do not need to use expensive anti-tampering devices. Even if the smart card is lost, as previously analyzed, the attacker cannot calculate the vehicle's real identity  $OID_i$  through the long-term pseudonym  $PID_i$  stored in the smart card. Therefore, the scheme can ensure the anonymity of the vehicle's real identity  $OID_i$  and has stronger security.



### 6.2. Computational Overhead Comparison

To analyze the computational overhead of the scheme, we define  $T_{bp}$  as the execution time of a bilinear operation,  $T_{em}$  as the execution time of a point multiplication operation on ECC, and  $T_{ea}$  as the execution time of a point addition operation on ECC. For the simulations, a PC with Inter Core i7-10710CPU and 16 GB of DDR3 memory was employed, and algorithms were chosen from MIRACL cryptographic library. The simulation steps are as follows. Construct elliptic curve  $\bar{E} : y^2 = x^3 + x(\text{mod } p_1)$  and  $E : y^2 = x^3 + ax + b(\text{mod } p)$ , where the lengths of integers  $p_1$  and  $p$  are 64 bytes and 20 bytes, respectively,  $G_1$  is a cyclic subgroup of elliptic curve  $\bar{E}$ , and  $G$  is a cyclic subgroup of elliptic curve  $E$ . Perform the bilinear pair operation  $e: G_1 \times G_1 \rightarrow G_T$ , point multiplication operation, and point addition operation on the cyclic group  $G$ , with an average of 5000 executions for each operation. The simulation results show that  $T_{bp}$  is approximately equal to 9.15 ms,  $T_{em}$  is approximately equal to 5.69 ms, and  $T_{ea}$  is approximately equal to 0.01 ms. Since the time consumption of hash function operation and modular multiplication operation is significantly lower than that of bilinear operation and point multiplication operation on ECC, they are not discussed in the comparison range of the computational overhead of the scheme. The comparison of the computational overhead of this scheme and similar schemes is shown in Table 2.

**Table 2.** Computational overhead comparison.

Scheme	Sign/ms	Verify/ms	Total/ms
Scheme in [15]	$T_{ea}$	$4T_{em} + 2T_{ea}$	$5T_{em} + 2T_{ea}$
Scheme in [16]	$4T_{em} + 2T_{ea}$	$3T_{em} + T_{ea} + 3T_{bp}$	$7T_{em} + 3T_{ea} + 3T_{bp}$
Scheme in [17]	$3T_{em} + 2T_{ea}$	$3T_{em} + 2T_{ea}$	$6T_{em} + 4T_{ea}$
Scheme in [18]	$3T_{em} + T_{ea}$	$3T_{em} + 2T_{ea}$	$7T_{em} + 4T_{ea}$
This paper's scheme	$T_{ea}$	$4T_{em} + 3T_{ea}$	$5T_{em} + 3T_{ea}$

The calculation overhead of each scheme is presented in Table 2, covering the execution time in the signature stage and the verification stage, as well as the total execution time. The time unit is ms. As shown in Table 2, this scheme does not utilize the bilinear operation. Instead, it employs point multiplication and point addition operations on the elliptic curve, which have a smaller computational overhead. In the signature process, the vehicle only employs one elliptic curve point multiplication operation. This is highly suitable for the vehicle end with limited computing power to generate signatures, and this relatively low calculation cost enables the scheme to be applicable in large-scale deployment environments with numerous vehicles. In the verification stage, this scheme also has a relatively smaller computational overhead compared with [16–18]. Although it performs one more elliptic curve point addition operation in the verification stage compared with the scheme in [15], this scheme has better security than the scheme in [15], and the time consumption of one point addition operation is significantly lower than that of one point multiplication operation, so the impact on the operational efficiency of the scheme is minor. Meanwhile, the processes of aggregate signature and batch verification enhance the overall efficiency of the scheme. Therefore, this scheme is more appropriate for the large-scale 5G vehicle network than other similar schemes. Figure 4 shows the comparison of the operation time of this scheme and similar schemes.



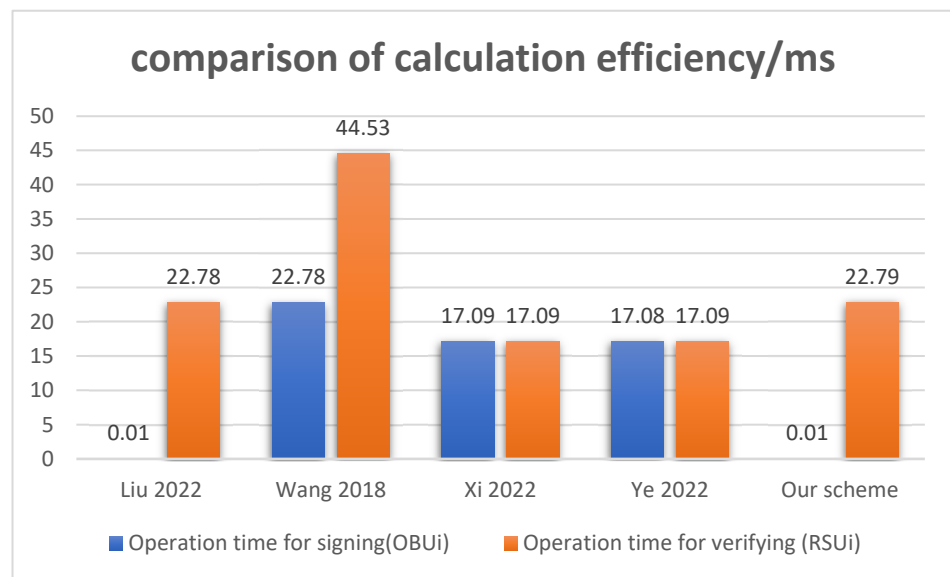


Figure 4. Operation time comparison [15–18].

### 6.3. Communication Overhead Comparison

This section will compare the scheme in this paper with other similar certificate-less aggregation schemes [15–18] from the perspective of communication overhead. The comparison contents include the signature length of a single message and the length of the broadcast message  $M_i$  minus the message  $m_i$ . Elliptic curves  $\bar{E}$  and  $E$  are constructed. The element  $|G_1|$  in the bilinear mapping group  $\bar{E}$  has a length of 128 bytes, and the integer has a length of 64 bytes. The element  $|G|$  in the group  $E$  has a length of 40 bytes, and the integer has a length of 20 bytes. The comparison of communication overhead with other schemes is shown in Table 3.

Table 3. Communication overhead comparison.

Scheme	Length of Signature	Length of Broadcast Message
Scheme in [15]	60 Bytes	188 Bytes
Scheme in [16]	272 Bytes	272 Bytes
Scheme in [17]	120 Bytes	300 Bytes
Scheme in [18]	100 Bytes	248 Bytes
This paper’s scheme	100 Bytes	228 Bytes

In the proposed scheme, the broadcast message  $M_i = (FID_i, pk_i, m_i, h_i, \sigma_i, T_j)$ , where the short-term pseudonym  $LID_i$  has a length of  $|G| + |Z_q^*| + |T| = 64Bytes$ , the public key  $pk_i$  has a length of  $|G| = 40Bytes$ ,  $h_i$  has a length of  $|Z_q^*| = 20Bytes$ , the timestamp  $T_j$  has a length of  $|T| = 4Bytes$ , the signature  $\sigma_i$  has a length of  $2|G| + |Z_q^*| = 100Bytes$ , and the length of the broadcast message is  $4|G| + 3|Z_q^*| + 2|T| = 228Bytes$ .

Using the same method, we can calculate the communication overhead of the schemes in [15–18]:

In the scheme proposed in [15], the broadcast message  $M_i = (m_i, \sigma_i, QID_i, FID^i, V_{pub_i}, T_j)$ , where the long-term pseudonym  $QID_i$  has a length of  $|Z_q^*| = 20Bytes$ , the short-term pseudonym  $FID^i$  has a length of  $|G| + |Z_q^*| + |T| = 64Bytes$ , the public key  $V_{pub_i}$  has a length of  $|G| = 40Bytes$ , the timestamp  $T_j$  has a length of  $|T| = 4Bytes$ , the signature  $\sigma_i$  has a length of  $|G| + |Z_q^*| = 60Bytes$ , and the broadcast message length is  $3|G| + 3|Z_q^*| + 2|T| = 188Bytes$ .

For the broadcast message  $M_i = (m_i, \sigma_i)$  in the scheme of [16], only the signature  $\sigma_i$  is included, and its length is  $2|G| + |G_1| + |Z_q^*| = 272\text{Bytes}$ , then the length of the broadcast message is also 272 bytes.

In the scheme of [17], the broadcast message  $M_i$  includes the pseudonym  $AID_i$ , the user's public key  $SPK_{AID_i}$ , the signature  $\sigma_i$ , where the pseudonym has a length of  $|G| + |Z_q^*| = 60\text{Bytes}$ , the user's public key has a length of  $3|G| = 120\text{Bytes}$ , the signature has a length of  $|G| + 4|Z_q^*| = 120\text{Bytes}$ , then the total length of the broadcast message is  $5|G| + 5|Z_q^*| = 300\text{Bytes}$ .

In the scheme of [18], the broadcast message  $M_i$  includes the signature  $\sigma_i$ , the timestamp TS, the fake identity PID, the secret value  $Q_i$ , the public key  $PK_{V_i}$ , where the signature  $\sigma_i$  has a length of  $|G| + 3|Z_q^*| = 100\text{Bytes}$ , the timestamp TS has a length of  $|T| = 4\text{Bytes}$ , the fake identity PID has a length of  $|G| + |Z_q^*| + |T| = 64\text{Bytes}$ , the secret value  $Q_i$  has a length of  $|G| = 40\text{Bytes}$ , the public key  $PK_{V_i}$  has a length of  $|G| = 40\text{Bytes}$ , so the total length of the broadcast message is  $4|G| + 4|Z_q^*| + 2|T| = 248\text{Bytes}$ .

Figure 5 shows the comparison of communication overhead. It can be observed that the proposed scheme has a lower communication overhead than the similar schemes in [16–18], and it is slightly higher than the scheme in [15], but it has higher security.

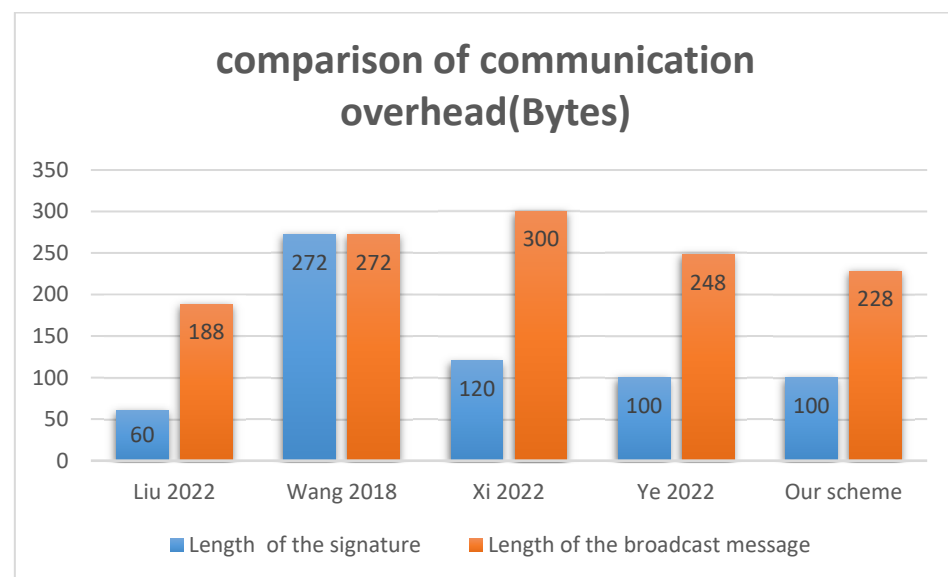


Figure 5. Communication overhead comparison [15–18].

## 7. Conclusions

In this paper, an anonymous authentication key negotiation scheme for 5G vehicle networks is proposed based on certificate-less aggregate signature. Two types of pseudonyms, namely long-term and short-term ones, are constructed to conceal the real identity of vehicles. Meanwhile, the partial private key of the vehicle is generated and distributed according to the long-term pseudonym to solve the problem of key escrow. Through security and performance analysis and verification, it is shown that the proposed scheme in this paper is provably secure under the random oracle model and meets the characteristics of anonymity, traceability and revocability, identity privacy, etc. Additionally, it can resist simulation attacks, man-in-the-middle attacks, and smart card loss attacks. Compared with similar schemes, it possesses stronger security and better computing efficiency and communication efficiency, making it more suitable for application in the 5G vehicle network. In future work, we will focus on potential extensions of the scheme, such as adaptations for different IoV environments, integration with emerging technologies like Blockchain,

and further discuss the practical implementation challenges and potential deployment scenarios in real-world IoV systems.

**Author Contributions:** Conceptualization, G.W. and Y.Q.; methodology, Y.Q.; validation, G.W. and Y.Q.; formal analysis, G.W. and Y.Q.; writing—original draft preparation, G.W. and Y.Q.; writing—review and editing, G.W., Y.Q., G.K. and Z.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the National Natural Science Foundation of China, grant number (No. 62270273).

**Data Availability Statement:** The data presented in this study are also available upon request from the corresponding author.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Wiseman, Y. Autonomous Vehicles, Encyclopedia of Information Science and Technology. 2020, Volume 1, Chapter 1. pp. 1–11. Available online: <https://u.cs.biu.ac.il/~wisemav/Autonomous-Vehicles-Encyclopedia.pdf> (accessed on 20 July 2024).
2. Wei, Z.; Lu, X.; Shi, T. Cross-domain key agreement protocol based on PKI system. *Comput. Sci.* **2017**, *44*, 155–158+182.
3. Wang, Q.; Qiao, R.; Fan, N.; Duan, Z. An efficient conditional anonymo authentication scheme for VANETs. *J. Beijing Jiaotong Univ.* **2019**, *43*, 80–86.
4. Pu, W. A Lightweight Group-based Secure Authentication and Communication Scheme in VANETs. Master's Thesis, Wuhan University, Wuhan, China, 2019.
5. Huang, Y.; Wang, Y.; Chen, W.; Zhang, Z. PKI cross-domain authentication model based on alliance chain. *Comput. Eng. Des.* **2021**, *42*, 3043–3051.
6. Lv, L.; Zheng, D.; Zhang, Y.; Yan, M.; Su, H. Identity-based aggregated signature verification in vehicular ad hoc network. *Comput. Eng. Des.* **2018**, *39*, 1866–1871.
7. Yao, R. Research on Efficient Authentication Schemes with Conditional Privacy-Preserving for VANETs. Master's Thesis, Chongqing University, Chongqing, China, 2021.
8. Zhang, H.; Chen, Z.; Huang, H.; He, X. Intra-group mutual authentication key agreement protocol based on Chinese remainder theorem in VANET system. *J. Commun.* **2022**, *43*, 182–193.
9. Zhang, G. Research on Security and Privacy Traceability in Internet of Vehicle Based on 5G. Master's Thesis, Xidian University, Xi'an, China, 2021.
10. Al-Riyami, S.S.; Paterson, K.G. Certificate-less secure upload for drive-thru Internet. *Lect. Notes Comput. Sci.* **2003**, 452–473.
11. Zhang, W.; Lei, L.; Wang, X.; Wang, Y. Secure and Efficient Authentication and Key Agreement Protocol Using Certificateless Aggregate Signatu re for Cloud Service Oriented VANET. *Acta Electron. Sin.* **2020**, *48*, 1814–1823.
12. Wei, G.; Qin, Y.; Fu, W. Secure and efficient certificateless authentication key agreement protocol in VANET. In *Communications in computer and Information Science, CCIS, Proceedings of Emerging Information Security and Applications-3rd International Conference, EISA 2022*; Springer: Cham, Switzerland, 2022; Volume 1641, pp. 160–172.
13. Zhang, Z. Research on Certificateless Anonymous Authentication Scheme and Group Key Agreement Scheme in VANETs. Master's Thesis, Chongqing University, Chongqing, China, 2021.
14. Xiong, L. Research on Group-based Authentication and Key Management Mechanism in 5G V2X. Master's Thesis, Xidian University, Xi'an, China, 2021.
15. Liu, X.; Wang, L.; Huan, L.; Du, X.; Niu, S. Certificateless Anonymous Authentication Scheme for Internet of Vehicles. *J. Electron. Inf. Technol.* **2022**, *44*, 295–304.
16. Wang, D.; Teng, J. Probably Secure Cetificateless Aggregate Signature Algorithm for Vehicular Ad hoc Network. *J. Electron. Inf. Technol.* **2018**, *1*, 11–17.
17. Xi, W. Research on Data Sharing and Security Authentication Scheme in Internet of Vehicles Environment. Master's Thesis, Northwest Normal University, Lanzhou, China, 2022.
18. Ye, X. Research on Efficient Digital Signature Technology in Internet of Vehicles. Master's Thesis, University of Electronic Science and Technology of China, Chengdu, China, 2022.
19. Bao, J.; Luo, M.; Chen, Y.; Peng, C.; Bao, Z. A Certificateless Anonymous Authentication Scheme for VANETs Based on Ring Signature. *J. Circuits Syst. Comput.* **2024**, *33*, 245–253. [[CrossRef](#)]
20. Shahidinejad, A.; Abawajy, J. Blockchain-based self-certified key exchange protocol for hybrid electric vehicles. *IEEE Trans. Consum. Electron.* **2023**, *70*, 543–553. [[CrossRef](#)]
21. Shahidinejad, A.; Abawajy, J.; Huda, S. Anonymous Lattice-Based Authentication Protocol for Vehicular Communications. *Veh. Commun.* **2024**, *48*, 100803. [[CrossRef](#)]
22. Shahidinejad, A.; Abawajy, J. Anonymous blockchain-assisted authentication protocols for secure cross-domain IoD communications. *IEEE Trans. Netw. Sci. Eng.* **2023**, *11*, 2661–2674. [[CrossRef](#)]

23. Shahidinejad, A.; Abawajy, J. An all-inclusive taxonomy and critical review of blockchain-assisted authentication and session key generation protocols for IoT. *ACM Comput. Surv.* **2024**, *56*, 1–38. [[CrossRef](#)]
24. Shahidinejad, A.; Abawajy, J. Efficient provably-secure authentication protocol for multi-domain IIoT using a combined off-chain and on-chain approach. *IEEE Internet Things J.* **2023**, *9*, 15241–15251.
25. Gope, P. PMAKE: Privacy-aware multi-factor authenticated key establishment scheme for advance metering infrastructure in smart grid. *Comput. Commun.* **2020**, *152*, 338–344. [[CrossRef](#)]
26. Wang, L. Research on Certificateless Anonymous Authentication and Conditional Privacy Preservation Scheme for Internet of Vehicles. Master's Thesis, Northwest Normal University, Lanzhou, China, 2022.

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.