Computer Systems Science and Engineering

Tech Science Press

DOI: 10.32604/csse.2024.051796

ARTICLE

Blockchain-Based Message Authentication Scheme for Internet of Vehicles in an Edge Computing Environment

Qiping Zou[1], Zhong Ruan[2,*] and Huaning Song[1]

[1]College of Artificial Intelligence and Manufacturing, Hechi University, Yizhou, 546300, China

[2]College of Big Data and Computer, Hechi University, Yizhou, 546300, China

*Corresponding Author: Zhong Ruan. Email: 05011@hcnu.edu.cn

Received: 15 March 2024 Accepted: 11 July 2024 Published: 13 September 2024

ABSTRACT

As an important application of intelligent transportation system, Internet of Vehicles (IoV) provides great convenience for users. Users can obtain real-time traffic conditions through the IoV's services, plan users' travel routes, and improve travel efficiency. However, in the IoV system, there are always malicious vehicle nodes publishing false information. Therefore, it is essential to ensure the legitimacy of the source. In addition, during the peak period of vehicle travel, the vehicle releases a large number of messages, and IoV authentication efficiency is prone to performance bottlenecks. Most existing authentication schemes have the problem of low authentication efficiency in the scenario. To address the above problems, this paper designs a novel reliable anonymous authentication scheme in IoV for Rush-hour Traffic. Here, our scheme uses blockchain and elliptic curve cryptography (ECC) to design authentication algorithms for message authentication between vehicles and roadside units (RSU). Additionally, we introduce the idea of edge computing into the scheme, RSU will select the most suitable vehicle as the edge computing node for message authentication. In addition, we used the ProVerif tool for Internet security protocols and applications to test its security, ensuring that it is secure under different network attacks. In the simulation experiment, we compare our scheme with other existing works. Our scheme has a significant improvement in computational overhead, authentication efficiency and packet loss rate, and is suitable for traffic scenarios with large message volume.

KEYWORDS

Internet of Vehicles; messages authentication; edge computing; blockchain; elliptic curve cryptography

1 Introduction

1.1 Background and Research Motivation

With the continuous maturity of intelligent vehicle technology, IoV technology has become one of the most popular topics today [1]. The main goal of IoV is to provide road services to achieve traffic management and road safety [2]. In the traditional IoV system, it mainly includes three types of communication entities: Trusted Authority (TA), RSU and vehicle [3]. Vehicles can collect daily driving data and road traffic information through sensing technology and periodically send them to

RSUs or other vehicles according to the DSRC [4] standard. RSU is an indispensable role [5], which is responsible for collecting and authenticating traffic messages sent by vehicles [6]. As a trusted center in the IoV, TA serves as a registration task for RSUs and vehicles [7,8]. The traditional model is shown in Fig. 1.

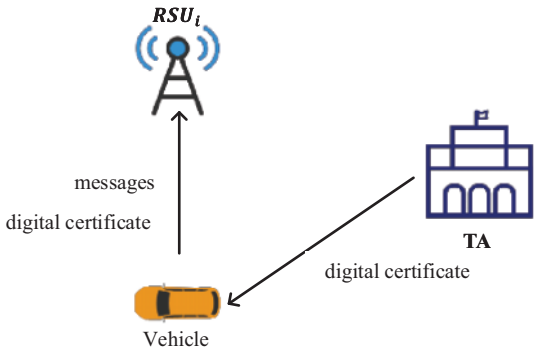

Figure 1: Traditional model of IoV

As an open network, IoV uses wireless electromagnetic wave communication technology to realize V2X communication [9]. In this communication environment, messages sent by vehicles can be easily intercepted and tampered by attackers, and their authenticity, integrity and legitimacy are uncertain [10,11]. In order to filter false information and improve the IoV's reliability, RSU needs to verify the legitimacy of traffic information generated by vehicles [12]. In addition, when the vehicle sends traffic-related information, it does not want others to know some sensitive information (such as real identity, location information, etc.) [13]. Therefore, vehicles can use pseudonyms to achieve privacy protection during communication. In IoV, the combination of privacy protection and traceability is called conditional privacy protection [14]. In today's IoV authentication mechanism, the higher the degree of privacy protection, the greater the computational and communication overhead. Especially in the traditional certificate authentication mechanism, the defect of excessive consumption of computing and storage resources is particularly obvious [15]. Nowadays, cloud computing technology has been widely used in IoV. But cloud computing also has limitations, that is, there is a higher transmission delay. During the peak period of vehicle travel, the vehicle releases a large number of messages, IoV authentication efficiency is prone to performance bottlenecks.

With the further development of technology, edge computing technology is introduced in IoV. As a new computing model that processes data away from the network center [16], edge computing can use flexible mechanisms and strategies to process sensitive information on edge terminal nodes. In addition, we no longer use the traditional certificate authentication mechanism, but use the encryption algorithm with lower computational complexity to reduce the consumption.

1.2 Contribution

This paper focuses on the research of IoV related applications, we combine blockchain and edge computing technology to design an efficient message authentication algorithm. The main research results of this paper are as follows:

(1) In the face of peak traffic scenarios, the message authentication work of the Internet of Vehicles is prone to bottlenecks. This paper is inspired by the idea of edge computing and applies it to the authentication scheme. RSU will select the most suitable vehicle as the edge computing node for

message authentication according to the real-time location distance and computing load. In addition, we also use blockchain and elliptic cryptographic curve to design the authentication algorithm, which reduce authentication latency, decrease packet loss ratio and alleviate network congestion indirectly.

(2) To verify security, we conducted ProVerif simulation experiments and theoretical analysis, which prove its security and correctness. In terms of performance, simulation experiments show that our scheme can also achieve fast authentication of messages during peak hours of traffic flow, and has lower computation and transmission over-head. With the addition of edge computing nodes, the packet loss rate (PLR) of IoV system was improved.

1.3 Organization

The article is divided into seven chapters. The first chapter mainly introduces the IoV and main contributions of the article. In the following chapters, we focus on the current research status of the IoV authentication scheme. The third chapter describes the background of the entire IoV system and some related theories of cryptography. The fourth chapter describes the specific authentication scheme designed by the article. In Chapter 5, we theoretically prove that the scheme is secure. In Chapter 6, the simulation results show that the scheme is efficient. Finally, the seventh chapter summarizes this paper and looks forward to the future development.

2 Related Work

2.1 PKI-Based Authentication Scheme

At present, Public Key Infrastructure (PKI) authentication schemes are very popular in IoV systems, this type of authentication scheme is recognized by the IEEE 1609.2 standard. Raya et al. [17] proposed an authentication scheme that can protect user identity privacy. TA will generate many key pairs and certificates for these vehicles in advance, so that these vehicles can sign the message. In reference [18], Scholars have optimized and improved the traditional PKI scheme and applied it to the IoV, which further improves the user's privacy while ensuring message integrity. Lu et al. [19] designed a new authentication mechanism using bilinear pairing cryptography algorithm. In this scenario, the certificate issuer is no longer TA, but RSU. RSU issues temporary certificates to passing vehicles for communication purposes. To sum up, these authentication mechanisms have great defects, and their overhead is very large, which is not suitable for IoV scenario. Especially when the traffic flow is particularly large, the system will face increasing pressure.

2.2 Identity-Based Authentication Scheme

Because PKI-based authentication schemes consume a lot, researchers have proposed a new scheme [20]. In these schemes, TA and RSU do not need to generate certificates for users, it greatly reduces the resource consumption. Kyung et al. [21] proposed to use pseudonym signature to protect user privacy in V2I communication. However, in this scheme, malicious nodes can modify information and cannot guarantee security. Vasudev et al. [22] implemented anonymous authentication between communicating entities through physical unclonable functions, but it cannot prevent man-in-the-middle attacks. Harsha et al. [23] designed a scheme with very low computational complexity. After mutual authentication, vehicle and RSU communicate through key negotiation. Although the performance of scheme is optimized, its security is weakened. Once a malicious event occurs, RSU cannot locate and track and know the identity of the malicious node. Zhang et al. [24] proposed the idea of batch authentication. Compared with other scheme in which RSU can only authenticate one message, this scheme can authenticate multiple messages in batches, which greatly improves the

efficiency. However, due to the limited computing resources of the RSU itself, it is still unable to quickly complete the message authentication and packet loss problem is serious during the peak travel time. Liu et al. [25] provided another new batch authentication scheme for IoV. The scheme uses bilinear algorithm to sign and encrypt messages. However, the scheme does not achieve a balance between security and efficiency. Wang et al. [26] used bilinear algorithm to realize segmented authentication of vehicles. The subsequent authentication process of vehicle is related to the previous authentication, which optimizes the authentication process and reduces the computational consumption, However, this scheme relies heavily on the stability of IoV system. If there is a line fault, RSU will not know the data related to vehicle authentication for the next calculation. If multiple vehicles simultaneously initiate identity authentication, it will also affect the efficiency of RSU. The scheme proposed by Islam et al. [27] is different from the conventional scheme. These vehicles in this scheme communicate by generating a group key and can freely choose groups, regardless of joining or leaving, the vehicle's behavior will continue to promote password updates. However, this scheme cannot support message batch authentication, which is very limited in the scenario of IoV. Xu et al. [28] applied blockchain technology to IoV authentication, which not only reduces the authentication overhead, but also avoids the single point problem in the network by designing the system model of multiple TAs. However, it needs to access the cloud center resources every time, and it will increase the delay time. Song et al. [29] proposed to use vehicles with rich computing resources as fog nodes to authenticate vehicles traveling around, which greatly reduces the burden of RSU authentication. Because vehicle has fast mobility, this will make the network topology of IoV change greatly, and the stability of the scheme needs to be further proved. Vijayakumar et al. [30] proposed a scheme for 6G scenarios, However, the computational complexity of the bilinear pairings is high, which leads to a large resource overhead. He et al. [31] combined fog computing with multi-TA model to give full play to their respective advantages. The fog computing is used to reduce the time delay of authentication, and the multi-TA structure is designed to prevent single-point problems in scheme.

In today's IoV authentication research, how to maintain low-cost consumption while ensuring system security is an urgent problem to be overcome. Therefore, a qualified authentication scheme must fit the actual application scenario of IoV and meet the real-time performance while meeting the security requirements.

3 Background and System Structure

3.1 System Model

IoV system involves four types of communication entities, which are trusted agency (TA), roadside unit (RSU), vehicles and edge computing node (ECN). Fig. 2 is the system model of IoV.

(1) TA: it has rich computing and storage resources, generally established by the government, and has full credibility [32]. It is responsible for maintaining the daily work of system operation and data storage. If malicious nodes appear in the IoV, TA can trace malicious nodes and make penalties at the same time. In the traditional IoV, it is often set up a TA, if the TA encounters a performance bottleneck, it will reduce the efficiency of authentication of system. And the setting of a single TA is also easy to cause the centralization problem. Therefore, this scheme places a TA in each region. These TA act as blockchain nodes to form an alliance chain and regularly publish relevant information on vehicle registration.

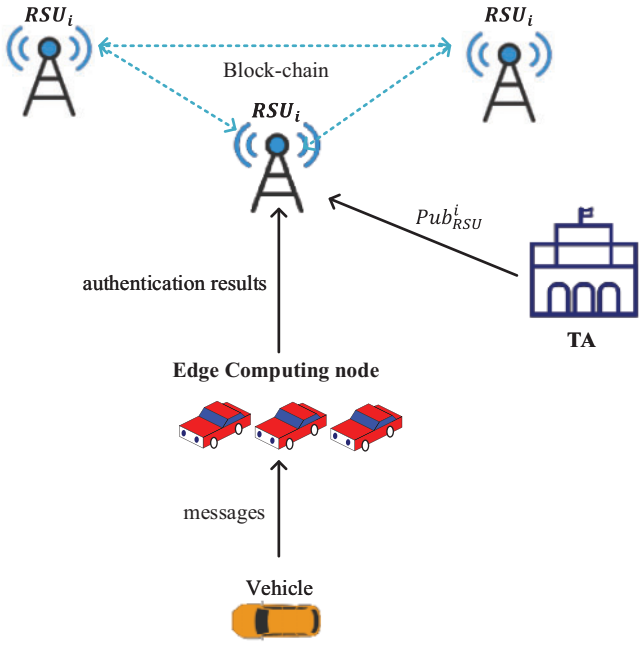

Figure 2: System model of IOV

(2) RSU: it is also a completely trusted third party and it can communicate directly with TA and vehicles [33]. In this scheme, it is responsible for publishing the anonymous identity of vehicles and verifying the legitimacy of messages collected by edge computing nodes. RSU can join the alliance chain after being registered and authenticated by TA. As a blockchain light node, it is used for maintaining the network information.

(3) Edge computing node (ECN): it is also essentially a legitimate vehicle and has a certain amount of computing resources. It is a node selected by RSU and is responsible for assisting the RSU in message authentication.

(4) Vehicle: it is the main user of IoV and can enjoy the service application of the system. Each vehicle has a communication device and tamper-proof device (TPD).

3.2 Related Cryptography Theory

In 1987, Miller and Koblitz proposed elliptic curve cryptography [34], which is based on the knowledge of groups and fields in mathematics. Because it has the characteristics of high safety strength and fast calculation speed, it is suitable for the environment with limited resources. ECC is more difficult to factorize the large composite number than RSA. It is generally defined as formula $y^2 = x^3 + ax + b \ mod \ q$, where a, b are coefficients, and $4a^3 + 27b^2 \neq 0 \, mod \, q$.

The addition principle of elliptic curve: take two points $A(X_1, \ Y_1)$, $B(X_2, \ Y_2)$ on curve. Select two points A and B to make a straight line and intersect with the elliptic curve, where the intersection point is $R(X_1, \ Y_1)$. Cross point R to make the vertical line of X axis intersect with the elliptic curve and point $-R$ (It is R's symmetric point on X axis), this is $A + B = -R$.

The multiplication principle of elliptic curve: it is different from our usual multiplication. It is calculated in the form of accumulation. For example, when $3Q$ needs to be calculated, it is generally reduced to $3Q = 2Q + Q = (Q + Q) + Q$, and the general form is $KQ = Q + Q + Q + Q + Q$ (K times). Fig. 3 shows the specific case of R = P + Q.

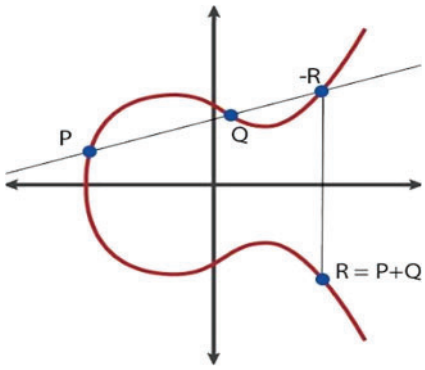

Figure 3: Elliptic curve algorithm example R = P + Q

Most encryption applications based on elliptic curves mainly use the following two generally accepted thorny issues:

The elliptic curve discrete logarithm problem: For example, $R = KP$, where R on elliptic curve, $K \in Z_q^*$. When P and K are known, R can be calculated, but when R and P are known, it is impossible to calculate K.

The elliptic curve computational Diffie-Hellman problem [35] (ECCDH). It is difficult to calculate abP for any known $P,\ aP,\ bP \in G$, where $a,\ b \in Z_q^*$.

3.3 Blockchain

Blockchain itself is a distributed system, It is characterized by chain connection, decentralization, group decision-making, and programmable customization. Each data block on the chain is stored in a chain structure and connected in chronological order.

In proposed scheme, all TA and RSU form a multi-server network based on a consortium chain. TA is a miner (full node) in the blockchain network with accounting rights, and RSU (light node) does not participate in the accounting. A new TA must be authorized before joining the network. Light nodes want to access information on the blockchain, must be authorized by the full node. Considering that the resources of IoV node are limited, we use the PBFT algorithm with higher efficiency and less computing resource consumption to make group decision between nodes. Our scheme also uses smart contract technology to automatically execute and manage data information in the authentication process according to predetermined rules, supporting users to achieve flexible and diverse automated operations, which helps to simplify the process and improve the authentication efficiency.

3.4 Security Model

The IoV system needs to have the following security features:

(1) Privacy protection: In the Internet era, users pay great attention to their privacy. In the process of communication, users do not want to expose their personal information and identity.

(2) Unforgeability: Any illegal vehicle cannot forge a legitimate signature.

(3) Traceability: If malicious behavior occurs in the system, TA can track and recover the identity of malicious nodes.

(4) Unlinkability: When a node receives multiple messages from the same vehicle, it is impossible to determine whether these messages are from the same message source.

(5) Resist man-in-the-middle attack: Attackers cannot implement this type of attack.

4 Proposed Scheme

In this chapter, we will explain each step of the authentication scheme in detail. The scheme has six steps: system set-up, registration, pseudonym generation, edge computing node selection, vehicle message signature, and vehicle message authentication. The complete authentication process will be shown in Fig. 4.

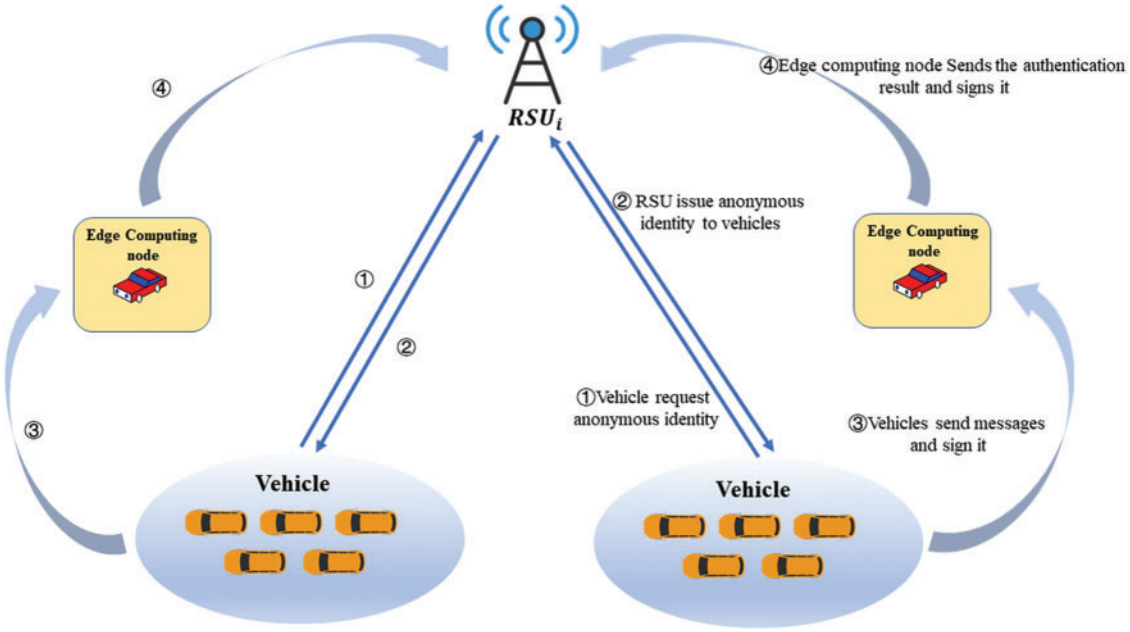

Figure 4: The process of messages authentication

When the system starts, TA performs the system set-up phase. If a vehicle wants to join IoV, it must obtain identity and key in system from the TA. These pre-operations are carried out in a secure network environment.

In our scheme, once vehicle gets (Pub^{i}_{OBU}, Key_i), TA will publish it to the blockchain for maintenance.

The symbols and parssssameter definitions that appear in the scheme will be explained and explained in detail in Table 1.

Table 1: Label description

Lable	Descriptions
Pub_{TA}	Public key of TA
S_T	Private key of TA
Pub^i_{OBU}	Public key of vehicle
S_V	Private key of vehicle
Pub^i_{RSU}	Public key of RSU
S_R	Private key of RSU
Key_i	A pointer to the relevant tuple
AID_i	The temporary anonymous identity of vehicle
T_{time}	Timestamp
$PID^i_v = \{PID^1_v,\ PID^2_v\}$	Anonymous identity that RSU gives vehicle
t	The key to vehicle's anonymous identity
$\sigma^i_m = \{\sigma^{,1}_m,\ \sigma^2_m,\ \}$	Vehicle's message signature
$HID_i = \{HID^1_i,\ HID^2_i\}$	Anonymous identity that RSU gives edge node
R_i	The key to edge node's anonymous identity

4.1 System Set-Up

At this stage, TA, as a fully trusted node, will complete the initialization of the system and broadcast the generated system parameters. It has a total of four steps:

Step (1): TA selects p and q (two large primes), an G (additive group) with order q and generator P.

Step (2): TA selects a random number $S_T \in Z^*_q$ as its own key, and calculates $Pub_{TA} = S_T \cdot P$.

Step (3): TA chooses three one-way hash functions: $h_1 : G \to G; h_2 : G \to Z^*_q; h_3 : \{0,\ 1\}^* \to Z^*_q$.

Step (4): TA publishes $\{G,\ p,\ q,\ P, Pub_{TA}, h_1, h_2, h_3\}$ to all.

4.2 Registration Phase

4.2.1 Registration of Vehicles

When a vehicle needs to be registered, its driver should submit the relevant information to TA. After TA checks that the received information is correct, it first randomly selects a number $S_V \in Z^*_q$ as vehicle key, and computes Pub^i_{OBU} for vehicle.

$$Pub^i_{OBU} = S_V \cdot P \tag{1}$$

After calculating public and private keys of the vehicle, TA calculates $Key_i = h(S_V \cdot Pub_{TA})$, and broadcasts $(Pub^i_{OBU},\ Key_i)$ to all TAs. In addition, Pub^i_{OBU} is not only public key of vehicle, but also can be can be used as a pointer on the blockchain to help RSU quickly find vehicle's other parameters. Through Pub^i_{OBU}, RSU retrieves the tuple $(Pub^i_{OBU},\ Key_i)$from blockchain. Finally, master node will initiate a consensus to write the transaction data $(Pub^i_{OBU},\ Key_i)$ of all TA broadcasts in the time period into the new block.

Finally, TA will send message $\{Pub^i_{OBU},\ S_V,\ Key_i\}$ to vehicle's tamper-proof device TPD in a secure environment. Fig. 5 shows the structure of our blockchain.

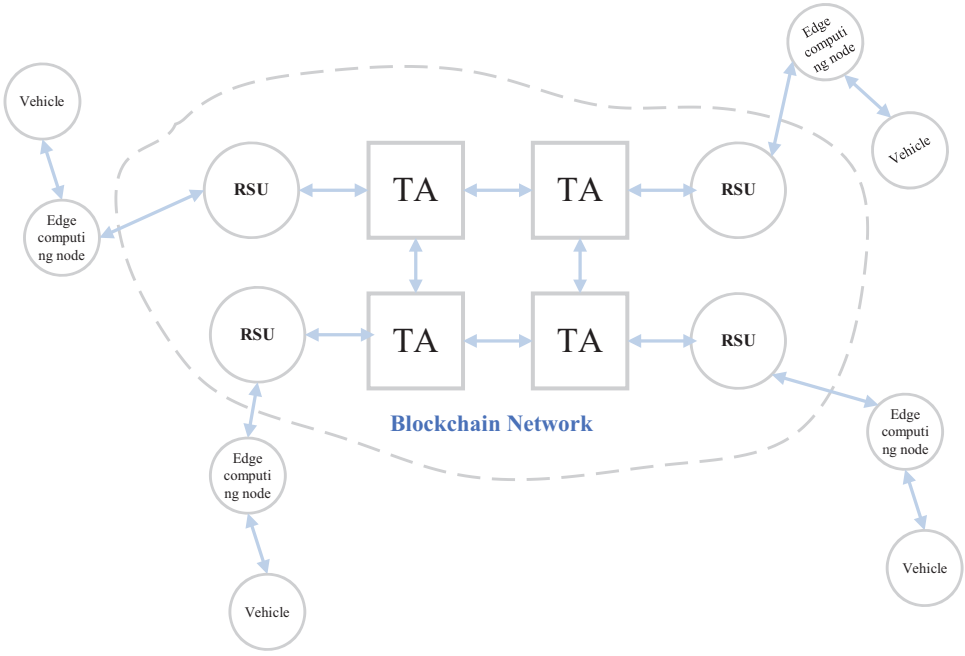

Figure 5: Blockchain structure of system

4.2.2 Registration of RSU

When roadside unit equipment RSU_i is installed, it needs to record the identity information at the TA. After receiving its registration application, TA will choose $S_R \in Z_q^*$ as its key, and calculates Pub^i_{RSU}:

$$Pub^i_{RSU} = S_R \cdot P \tag{2}$$

Finally, TA sends message $\{Pub^i_{RSU},\ S_R\}$ to RSU_i through secure channel. The entire registration process for each entity is shown in Fig. 6. After RSU_i registration is completed and successfully connected to IoV, it can apply to TA as a light node to obtain access to blockchain ledger, so that it can complete the issuance of anonymous identities for subsequent vehicles.

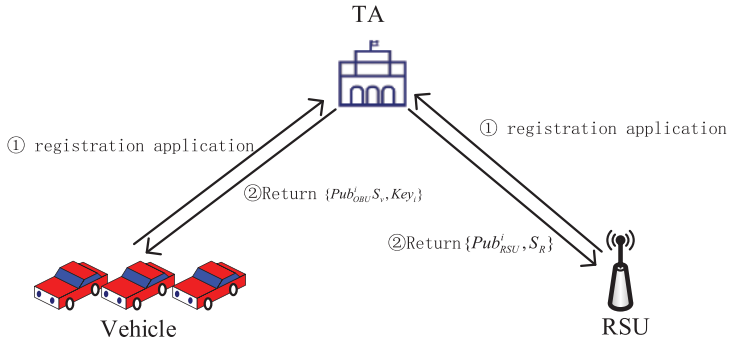

Figure 6: The registration process of vehicle and RSU

4.3 Anonymous Identity Generation of Vehicles

4.3.1 The Vehicle Applies for Anonymity

First, vehicle will request an anonymous identity from RSU, but it will first use a temporary anonymity to complete the application. The vehicle randomly selects a $r \in Z_q^*$, and calculates the temporary anonymity AID_i.

$$AID_i = \{AID_i^1,\ AID_i^2\} \tag{3}$$

where $AID_i^1 = r \cdot P, AID_i^2 = (r + S_V) \cdot Pub_{RSU}^i$. Vehicle then begins to calculate $Pid_{OBU}^i = Pub_{OBU}^i \oplus h(r \cdot Pub_{RSU}^i)$. T_{time} is time stamp generated by vehicle. Then, vehicle sends message $\{AID_i,\ Pid_{OBU}^i,\ T_{time}\}$ to RSU_i to apply for anonymous identity.

The specific implementation algorithm is as follows:

Algorithm 1: Application for anonymity of OBU_i (performed by OBU_i)

Input:(Pub_{RSU}^i, S_V)
Output:$(AID_i,\ Pid_{OBU}^i,\ T_{time})$
1.Select $r \in Z_q^*$
2.Generates a timestamp T_{time}
3. Compute $AID_i^1 = r \cdot P$
4. Compute $AID_i^2 = (r + S_V) \cdot Pub_{RSU}^i$
5. Compute $Pid_{OBU}^i = Pub_{OBU}^i \oplus h(r \cdot Pub_{RSU}^i)$
6. Broadcast $\{AID_i, Pid_{OBU}^i, T_{time}\}$ within the region of RSU_i

4.3.2 RSU Issues Anonymous Identity to Vehicles

When RSU_i receives application sent by vehicle, RSU_i starts to verify the validity of the message. If $T_{now} - T_{time} < \Delta T$, then the message has timeliness. After timeliness check is passed, RSU_i begins to calculate Pub_{OBU}^i of vehicle:

$$Pub_{OBU}^i = Pid_{OBU}^i \oplus h\left(S_R \cdot AID_i^1\right) \tag{4}$$

After RSU_i obtains Pub_{OBU}^i of vehicle, RSU_i will check whether there is information about Pub_{OBU}^i on the blockchain by executing a smart contract. If it exists, it means that a vehicle has indeed obtained the identity Pub_{OBU}^i. In this process, RSU_i will also obtain the parameter Key_i for further authentication calculations. After determining that Pub_{OBU}^i already exists, RSU_i will continue to confirm the authenticity of AID_i, that is, whether AID_i is Pub_{OBU}^i temporary anonymity. Next, RSU_i computes the equation

$$AID_i^2 == S_R \cdot AID_i^1 + S_R \cdot Pub_{OBU}^i \tag{5}$$

If the equation holds, then the message is indeed sent by vehicle, and AID_i's identity legitimacy is confirmed.

RSU_i is prepared to generate an anonymous identity that is used by vehicles to communicate with ECN for vehicle. First, the random number $t \in Z_q^*$ is selected as private key of vehicle's temporary identity. RSU_i saves $\{PID_v^i, Pub_{OBU}^i\}$ locally. The generation process of anonymous identity is as follows:

$$PID_v^1 = t \cdot P \tag{6}$$

$$PID_v^2 = Pub_{OBU}^i \oplus h(t \cdot S_R \cdot Pub_{TA}) \tag{7}$$

where anonymous identity is $PID_v^i = \{PID_v^1,\ PID_v^2\}$, After generating anonymous identity PID_v^i, RSU_i computes $S_t^{PID} = t \oplus h(Key_i)$, $Auth_{RSU}^i = h(S_R \cdot AID_i^1) \oplus h(Key_i)$. Finally, RSU_i generates a timestamp T_{time} based on the present time and sends $\{PID_v^i,\ S_t^{PID},\ Auth_{RSU}^i,\ T_{time}\}$ to vehicle. The algorithm is as follows:

Algorithm 2: Issuance of anonymous identity (performed by RSU_i)

Input:$(AID_i,\ Pid_{OBU}^i,\ T_{time})$
Output:$(PID_v^i,\ S_t^{PID},\ Auth_{RSU}^i,\ T_{time})$
1. **IF** $T_{now} - T_{time} < \Delta T$ **then**
2. Discard the message;
3. **Else**
Compute $Pub_{OBU}^i = Pid_{OBU}^i \oplus h(S_R \cdot AID_i^1)$
executes the smart contract
4. **IF** $\{Pub_{OBU}^i,\ Key_i\}$ is in the alliance blockchain
IF $(AID_i^2 == S_R \cdot AID_i^1 + S_R \cdot Pub_{OBU}^i)$
5. The AID_i is legally identified
6. Select $t \in Z_q^*$
7. Compute $PID_v^1 = t \cdot P$
Compute $PID_v^2 = Pub_{OBU}^i \oplus h(t \cdot S_R \cdot Pub_{TA})$
8. Compute $S_t^{PID} = t \oplus h(Key_i)$
9. Compute $Auth_{RSU}^i = h(S_R \cdot AID_i^1) \oplus h(Key_i)$
10. **Else**
11. identity of the AID_i is illegal
12. **Else**
13. The vehicle doesn't exist. It's not registered
14. Generates a timestamp T_{time}
15. Broadcast $\{PID_v^i,\ S_t^{PID},\ Auth_{RSU}^i,\ T_{time}\}$ within the region of *vehicle*

4.3.3 Vehicle Receives Anonymous Identity

When vehicle Pub_{OBU}^i receives message from RSU_i, the timeliness of the message must be verified first that $T_{now} - T_{time} < \Delta T$. The vehicle determines whether the PID_v^i is published by RSU_i. The vehicle first verifies the equation

$$h\,(Key_i) == Auth_{RSU}^i \oplus h(r \cdot Pub_{RSU}^i) \tag{8}$$

If not equal, then the PID_v^i is discarded; if it is equal, then the PID_v^i is published by RSU_i. Vehicle will also calculate private key t of anonymous identity through the following equation to indicate the message that needs to be published.

$$t = S_t^{PID} \oplus h(Key_i) \tag{9}$$

The specific implementation algorithm is as follows:

Algorithm 3: Confirm anonymous identity (performed by vehicle)

Input:(PID_v^i, S_t^{PID}, $Auth_{RSU}^i$, T_{time})
Output:(successful)

1. **IF** $T_{now} - T_{time} < \Delta T$ **then**
2. Discard the message;
3. **Else**
4. **IF** $h(Key_i) == Auth_{RSU}^i \oplus h(r \cdot Pub_{RSU}^i)$
5. Compute $t = S_t^{PID} \oplus h(Key_i)$
6. Get anonymous identity PID_v^i and key t
7. **Else**
8. Failed to obtain anonymous identity

The application process of anonymous identity is shown in the Fig. 7.

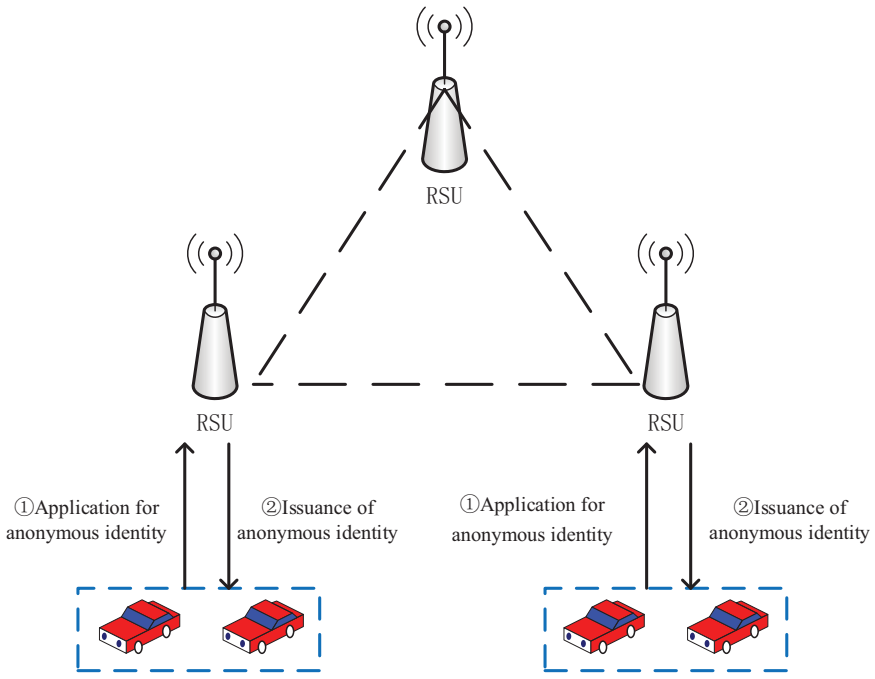

Figure 7: Application of anonymous identity

4.4 Selection of Edge Computing Nodes

All vehicles can apply to become ECN, but must meet two requirements, one is parked within the communication range of RSU_i, and the other is a certain amount of computing resources. The main task of ECN is to carry out the first round of authentication of the messages sent by other vehicles to RSU_i and feeds back results to RSU_i. Vehicle application to become an ECN proceeds mainly through the following three steps:

(1) First, vehicle should be a legal vehicle registered through IoV system. Vehicle can use private key t of temporary identity to sign the message $m_{request}$. The specific operations are as follows:

$$Request_i = t_i \cdot h\left(m_{request} || T_m\right) + S_v \cdot (PID_v^i) \tag{10}$$

The vehicle sends $\{PID_v^i,\ Request_i,\ m_{request},\ T_m\}$ to RSU_i. Immediately upon receipt of the message, RSU_i first queries vehicle's Pub_{OBU}^i according to PID_v^i. If PID_v^i does not exist, the first step of vehicle's legitimacy verification does not pass, and all data sent by the vehicle is discarded. If the vehicle passes, RSU_i continues the next calculation:

$$Request_i \cdot P == PID_v^i \cdot h\left(m_{request} || T_m\right) + Pub_{OBU}^i \cdot (PID_v^i) \tag{11}$$

If the equation is passed, then the legitimacy of the application vehicle is verified, and the vehicle can be listed as one of candidate ECN.

(2) Considering the success rate of message authentication, ECN should have sufficient computing resources and maintain a short communication distance with RSU_i. We will select ECN from candidate nodes according to two factors of distance and available computing resources. We set the communication radius of RSU_i to be x,and the straight distance between the vehicle and RSU_i is d. Distance will be an important factor affecting the vehicle to become an ECN. We will calculate d_i in two cases:

$$d_i = \begin{Bmatrix} d/X, 0 < d < X \\ \\ 0, d > X \end{Bmatrix} \tag{12}$$

In terms of computing resources, we are more inclined to select vehicles with more available computing resources. When a vehicle is applying as a bus, considering that it has more computing resources and is a fully trusted node, we will give priority to the bus.

The formula for selecting ECN is as follows:

$$ECN_i = \alpha_i \cdot d_i^{-1} + \beta_i \cdot \left(R_{use}^i\right)^{-1} + \gamma_i \tag{13}$$

α_i and β_i are the weights, respectively, $\alpha_i + \beta_i = 1$ and γ_i is a unique variable of bus. When the application node itself is a bus, the value of γ_i is 0, otherwise γ_i is 1. Among all vehicles, the vehicle with the smallest ECN_i value will be selected as ECN, and the number of ECNs is determined by RSU_i. For example, RSU_i needs to select two ECNs, which take the two candidate nodes with the smallest ECN_i as ECNs.

Algorithm 4: Selection of edge computing nodes (performed by RSU_i)

Input:$(PID_v^i, \sigma_m^i,\ m_{request},\ d_i,\ R_{use}^i,\ T_m)$
Output:$Edge_i[]$
1. for (each $m_{request}$) **do**
2. **if** (timestamp $T_{now} - T_m < \Delta t$) **then**
3. **if** $(\sigma_i^2 \cdot P == PID_v^{i,1} h\left(PID_v^{i,2}\right) + \sigma_i^1 \cdot h(m_i || T_m))$
4. Add $\{PID_v^i,\ \sigma_m^i,\ d_i,\ R_{use}^i\}$ to Legal vehicle[] i = 1, 2, 3 ..n

(Continued)

Algorithm 4 (continued)

5. Verify the next one
6. **End if**
7. **Else**
8. The $(m_{request}, PID_v^i, \sigma_m^i)$ is invalid.
9. **End if**
10. **End for**
11. **for** $(i = 0; i \ll n; i++)$
12. chose α_i, β_i from [0, 1]
13. **If** Legal vehicle [i] $.PID_i$ is Bus, **do**
14. $\gamma_i = 0$
15. **Else**
16. $\gamma_i = 1$
17. **End IF**
18. IF $(0 < d < X)$
19. $d_i = d/X$
20. else
21. $d_i = 0$
22. Compute Legal vehicle [i] $.ECN_i = \alpha_i \cdot d_i^{-1} + \beta_i \cdot \left(R_{use}^i\right)^{-1} + \gamma_i$
23. **End for**
24. Choose $Edge_i[] = minmum$(Legal vehicle [i] $.ECN_i$)
25. **Return** $Edge_i[]$

When vehicle becomes ECN, RSU_i randomly selects a number $R_i \in Z_q^*$ as it's key, and begins to issue new anonymous identity HID_i for ECN:

$$HID_i = \left\{HID_i^1,\ HID_i^2\right\} \tag{14}$$

$$HID_i^1 = R_i \cdot P \tag{15}$$

$$HID_i^2 = Pub_{OBU}^i \oplus h(R_i \cdot S_R \cdot Pub_{RSU}) \tag{16}$$

RSU_i sends $\{HID_i, R_i\}$ to TPD device of ECN through secure transmission. ECN will use HID_i to communicate with RSU_i for subsequent message authentication. The ECN sends information to RSU_i at regular intervals to maintain identity of ECN. At a certain time, if RSU_i does not receive daily feedback from ECN. RSU_i will cancel the identity of ECN. In the simulation experiment, we set 150 ms.

4.5 Messages Publish and Authentication

After vehicle is on the road, it first randomly selects a number $K_i \in Z_q^*$, and uses K_i and t_i to sign the traffic information m through the TPD device.

$$\sigma_i^1 = K_i \cdot P \tag{17}$$

$$\sigma_i^2 = t_i \cdot h\left(PID_v^2\right) + K_i \cdot h(m||T_m) \tag{18}$$

$$\sigma_m^i = \{\sigma_m^{,1}, \sigma_m^2, \} \tag{19}$$

The vehicle will select the nearest ECN $Edge_i$ in the current RSU_i communication range to send messages $\{PID_v^i,\ \sigma_m^i,\ m,\ T_m\}$.

Algorithm 5: Generation of vehicle traffic signatures (performed by *vehicle*)

Input:(m, t_i)
Output:$(PID_v^i,\ \sigma_m^i,\ m,\ T_m)$
1. Select $K_i \in Z_q^*$
2. Generates a timestamp T_m
3. Compute $\sigma_i^1 = K_i \cdot P$
4. Compute $\sigma_i^2 = t_i \cdot h\left(PID_v^2\right) + K_i \cdot h(m||T_i)$
5. Broadcast $\{PID_v^i,\ \sigma_m^i,\ m,\ T_m\}$ within the region of $Edge_i$

When ECN receives instruction of RSU_i, ENC performs batch authentication algorithm processing i messages$\{PID_v^i,\ \sigma_m^i,\ m_i,\ T_m\}$ $(1 < i < n)$ received in a certain time. The authentication method of this batch of messages is as follows:

$$\left(\sum_{i=1}^{n} a_i \cdot \sigma_i^2\right) P = \sum_{i=1}^{n} a_i \cdot PID_v^{i,1} \cdot h\left(PID_v^{i,2}\right) + \sum_{i=1}^{n} a_i \cdot \sigma_i^1 \cdot h(m_i||T_m) \tag{20}$$

ECN merge authenticated successful and time-sensitive messages into $M_T = \left(\sum_{i=1}^{n} M_i\right)||T_i$, and use R_i and t_i to sign M_T through the TPD device:

$$\theta_i = t_i \cdot h\left(HID_i^2\right) + R_i h(M_T) \tag{21}$$

Finally, the ECN sends message$\{M_T,\ \theta_i,\ HID_i,\ PID_v^i\}$ of its own authentication success to RSU_i.

Algorithm 6: Batch authentication algorithm (performed by edge node)

Input:$(PID_v^i,\ \sigma_m^i,\ m_i,\ T_i,\ i = 1 \ldots, n)$
Output:(M_T, θ_i, HID_i)
1. for (each message m_i) **do**
2. **if** $(T_{now} - T_i > \Delta t)$ **then**
3. The $(m_i, PID_v^i,\ \sigma_m^i)$ is invalid.
4. ECN discards the message.
5. **End if**
6. Compute $A_i = h\left(PID_v^{i,2}\right)$
7. Compute $B_i = h(m_i||T_i)$
8. End for
9. Let $\left\{\sigma_m^i\right\}$ $i = 1, 2 \ldots, n$ be the list of signatures
10. If $\left(\sum_{i=1}^{n} a_i \cdot \sigma_i^2\right) P = \sum_{i=1}^{n} a_i \cdot A_i PID_v^{i,1} + \sum_{i=1}^{n} a_i \cdot \sigma_i^1 \cdot B_i$ **then**
11. Accept the signatures σ_m^i and consume the respective messages $\{m_i\} i = 1, 2, 3 \ldots, n$
12. Else
13. Drop the messages $\{m_i\} i = 1, 2, 3 \ldots, n$
14. End if
15. Compute $M_T = \left(\sum_{i=1}^{n} M_i\right)||T_i, i = 1, 2, 3 \ldots, n$
16. Compute $\theta_i = t_i \cdot h\left(HID_i^2\right) + R_i h(M_T)$
Broadcast $(M_T,\ \theta_i,\ HID_i)$within the region of RSU_i

When RSU_i receives result from ECN, it begins the next authentication calculation to prove its validity.

$$\theta_i \cdot P == PID_v^1 \cdot h\left(HID_i^2\right) + HID_i^1 h\left(M_T\right) \tag{22}$$

If the equation holds, then RSU_i considers that the message sent by the ECN is legitimate. RSU_i will find the PID_i of each message sender based on M_T collection of traffic messages received, and broadcast $\{PID_i,\ m,\ Edge_i\}$to all vehicles in a secure way. If vehicle has objections to this result, then it can feedback to RSU_i. If the vehicle 's objection is established, then all the information received in the original time-period will be re-authenticated. If RSU_i does not receive any feedback in a fixed time period, then all messages authenticated by the ECN are verified and paid as a reward. So far, RSU_i have completed all the message authentication work.

5 Security Analysis

In terms of security, we have theoretically verified the scheme, and it can resist the network attack behavior in Chapter 3.4.

5.1 Formal Security Analysis

Theorem 1: The scheme can resist adaptive chosen message attack.

The whole proof process is carried out under the principle of ECDLP. The following is the proof process:

Adversary A forged a signature $\{PID_v^i,\ \sigma_m^i,\ m,\ T_m\}$, and give an ECDLP instance $Q = XP$, and then the C is challenger that can solve the discrete logarithm problem with a certain probability under A's query.

Initialization: Challenger C first selects a random number $t_i \in Z_q^*$, and compute $PID_v^1 = t_i \cdot P$.Next, C sets the system parameters $\{p,\ q,\ P,\ Pub_{TA},\ PID_v^1,\ h_1,\ h_2,\ h_3\}$ and sents to A. And C constructs three hash lists, that is, the form of L_{H_1} is $(\alpha,\ \tau h_1)$, the form of L_{H_2} is $\left(PID_v^1,\ PID_v^2,\ \tau h_2\right)$, and the form of L_{H_3} is $(m, T_m \tau h_2)$.

H_1 query: Challenger C creates and maintains list H_1. When receiving α queries from adversary A, C first confirms whether there is a tuple $(\alpha,\ \tau h_1)$ in table H_1. If it exists, C returns $\tau h_1 = h_1(\alpha)$ to adversary A. If not, C randomly selects a number $\tau h_1 \in Z_q^*$ to return it to A and store $(\alpha, \tau h_1)$ to table H_1.

H_2 query: Challenger C creates and maintains list H_2. When receiving PID_v^2 queries from adversary A, if $(PID_v^1,\ PID_v^2,\ \tau h_2)$ exists in table H_2, C tells $\tau h_2 = h_1(PID_v^2)$ to A. If not, C chooses a random number $\tau h_2 \in Z_q^*$ to return it to A and store$(PID_v^1,\ PID_v^2,\ \tau h_2)$ to table H_2.

H_3 query: Challenger C creates and maintains list H_3. After C receiving (m, T_m) queries from adversary A, C first confirms whether there is a tuple $(m,\ T_m,\ \tau h_3)$ in table H_3. If so, C will tell $\tau h_3 = h_1(m||T_m)$ to A. If not, C chooses a random number $\tau h_3 \in Z_q^*$ to return it to A and store $(m,\ T_m,\ \tau h_3)$ to table H_3.

Signature query: After C receives a message request from A, C randomly selects $h_{i,2},\ h_{i,3},\ \sigma_m^2 \in Z_q^*$, $\sigma_m^1 \in G$, Next, C calculates $PID_v^1 = \dfrac{\sigma_m^2 \cdot P - \sigma_m^1 h_{i,3}}{h_{i,2}}$. Then C increases $(PID_v^1,\ PID_v^2,\ \tau h_2)$ and $(m,\ T_m,\ \tau h_3)$ to H_2 and H_3, respectively. Finally, the response of C to A is $\{PID_v^i\ \sigma_m^i,\ m,\ T_m\}$.

Output: A final output message tuple $\{PID_v^i,\ \sigma_m^i,\ m,\ T_m\}$

$$\sigma_m^2 \cdot P = PID_v^1 h_{i,2} + \sigma_m^1 h_{i,3} \tag{23}$$

C uses Eq. (23) to check message tuples. If the equation does not hold, C ends the game.

In this paper, the above process is repeated with different values H_2, and A can obtain another effective message tuple $\{PID_v^i,\ \sigma_m^{i'},\ m,\ T_m\}$.In this case, the following equation can be obtained:

$$\sigma_m^{2'} \cdot P = PID_v^1 h'_{i,2} + \sigma_m^1 h_{i,3} \tag{24}$$

Through Eqs. (23) and (24), we can get

$$\begin{aligned} &\left(\sigma_m^2 - \sigma_m^{2'}\right) \cdot P \\ &= \sigma_m^2 \cdot P - \sigma_m^{2'} \cdot P \\ &= PID_v^1 h_{i,2} + \sigma_m^1 h_{i,3} - PID_v^1 h'_{i,2} + \sigma_m^1 h_{i,3} \\ &= \left(h_{i,2} - h'_{i,2}\right) PID_v^1 \\ &= \left(h_{i,2} - h'_{i,2}\right) t \cdot P \end{aligned} \tag{25}$$

Therefore, this paper can get $\left(\sigma_m^2 - \sigma_m^{2'}\right) = \left(h_{i,2} - h'_{i,2}\right) \cdot t \bmod q$, Therefore, C can solve the ECDL problem $PID_v^1 = t \cdot P$ by calculating $\mathrm{t} = \left(\sigma_m^2 - \sigma_m^{2'}\right)\left(h_{i,2} - h'_{i,2}\right)^{-1}$. However, this conclusion contradicts with the recognized difficulty of the ECDL problem. The above process is not established.

5.2 Informal Security Analysis

5.2.1 The Correctness of Signature

When the ECN starts the first step of authentication, it will use Eq. (26) to judge:

$$\begin{aligned} \left(\sum_{i=1}^{n} a_i \cdot \sigma_i^2\right) P &= \left(\sum_{i=1}^{n} a_i \cdot \left[t_i \cdot h\left(PID_v^2\right) + K_i \cdot h(m||T_m)\right]\right) \cdot P \\ &= \left[\sum_{i=1}^{n} a_i \cdot t_i \cdot h\left(PID_v^2\right) + \sum_{i=1}^{n} a_i \cdot K_i \cdot h(m||T_m)\right] \cdot P \\ &= \sum_{i=1}^{n} a_i \cdot PID_v^{i,1} h\left(PID_v^{i,2}\right) + \sum_{i=1}^{n} a_i \cdot \sigma_i^1 \cdot h\left(m_i||T_m\right) \end{aligned} \tag{26}$$

5.2.2 Vehicle Identity's Privacy Protection

In the process of IoV communication, in order to protect identity privacy and security, all communication entities use anonymous identity to communicate. Taking an ordinary vehicle as an example, we use t_i and Pub_{OBU}^i to obtain its anonymous identity PID_i=$\{PID_v^1,\ PID_v^2\}$, where $PID_v^1 = t_i \cdot Pub_{RSU}^i,\ PID_v^2 = Pub_{OBU}^i \oplus h(t_i \cdot S_R \cdot Pub_{TA})$.

If malicious node wants to understand vehicle's privacy data, then it must know t_i and S_R in advance. It is assumed that adversary A can calculate key t_i through vehicle's PID_v^1, that is, $Adv_A^{ECDLP}(t) = Prb\left[A\left(PID_v^1, P\right) = t_i\right]$. At the same time, A can calculate RSU_i's key S_R through Pub_{RSU}^i, that is, $Adv_A^{ECDLP}(t) = Prb\left[A\left(Pub_{RSU}^i,\ P\right) = S_R\right]$. Then A can calculate vehicle's Pub_{OBU}^i by cracking t_i and S_R. However, the above operation is contrary to ECDLP principle. We know that the probability

of $Adv_A^{ECDLP}(t) < \epsilon$ and $\epsilon > 0$ is very small, so it has the security characteristics of vehicle identity's privacy protection.

In addition, the anonymous identity of the vehicle can be changed at any time. When other vehicles receive multiple messages from the same vehicle, it cannot know vehicle's identity that sends these messages. That is to say, the scheme of this paper satisfies unlinkability.

5.2.3 Traceability

In this scheme, in order to protect their privacy, everyone uses anonymous identity to communicate. In order to prevent malicious behavior of legitimate users, RSU and TA can trace its real identity through anonymous identity. The formulas used by RSU and TA for identity traceability are as follows:

$$\text{RSU}: h(t_i \cdot S_R \cdot Pub_{TA}) \oplus PID_v^2 = h(t_i \cdot S_R \cdot Pub_{TA}) \oplus Pub_{OBU}^i \oplus h(t \cdot S_R \cdot Pub_{TA}) = Pub_{OBU}^i$$

$$\text{TA}: h(t_i \cdot S_R \cdot Pub_{TA}) \oplus PID_v^2 = h(t_i \cdot S_R \cdot Pub_{TA}) \oplus Pub_{OBU}^i \oplus h(t \cdot S_R \cdot Pub_{TA}) = Pub_{OBU}^i$$

Because other nodes do not know the secret key of RSU_i and TA, it can ensure the security and authenticity of identity traceability.

5.2.4 Resist Replay Attacks

The messages $\{PID_v^i,\ \sigma_m^i,\ m,\ T_m\}$ broadcast by the vehicle have time stamps, which are also part of the signature message. The time stamp cannot be modified. Checking T_m will be the first step for the receiver. If $T_{now} - T_m > \Delta$ (a fixed time interval), the message is discarded. This can well guarantee against replay attacks.

5.2.5 Resist Man-in-the-Middle Attackss

We can avoid adversaries impersonating others to participate in communication, in our scheme, anonymous identity applications and message sending require multi-stakeholder participation and mutual authentication. During the entire authentication session, the message sent by each node needs to be signed using the corresponding secret key. The pseudonym and the generated signature message are based on the ECDL problem, so it can effectively resist the man-in-the-middle attack.

6 Experimental Analysis

In this part, the team also used simulation tools to carry out security simulation experiments and performance analysis experiments.

6.1 Safety Simulation Experiment

We chose the widely accepted software tool ProVerif for experimental operation and the complete scheme described in Section 4 is implemented and verified in ProVerif. For a detailed description of the scheme, please see the Fig. A1.

In the experimental simulation, malicious node's goal is to launch an attack to obtain the privacy of the vehicle. Simulation results (as shown in Fig. 8) show that although malicious nodes launch network attacks, they still fail to achieve their goals. Malicious nodes can't get vehicle's private data, such as Pub_{OBU}^i and key t_i. All in all, our proposed scheme is secure.

Figure 8: The scheme's simulation results

6.2 Performance Simulation Experiment

In terms of performance, we mainly analyze several factors and compare the performance with CAEC [36], DCBA [37] and ABAH [38] schemes.

We use a popular cryptography library called Crypto + + library to measure all the cryptographic operations involved in the above scheme. The performance evaluation is carried out in a machine environment configured with Intel i7-9750 3 GHZ and Visual Studio 2019. In addition, we also use the Veins simulation platform to build simulation scenarios for experiments.

6.2.1 Computational Overhead

We set up a security level of 80 bits scheme, Elliptic curve cryptography is set as follows: E: $y^2 = x^3+ax+b \ mod \ q$, G is defined in the additive group of elliptic curves by order q and generator P, where P, q are primes of 160 bits. In the experiment,we average the execution time of each corresponding operation. Let T_{PM} represent point multiplication operation on the elliptic curve, and its execution time is 0.7358 ms; T_{PMS} represents point multiplication operation of elliptic curve vector, and its execution time is 0.0428 ms; T_{PA} represents the point addition operation of elliptic curve, and its execution time is 0.004 ms. T_{GM} represents the bilinear pairing point multiplication operation, and its run time is 2.6439 ms; T_{GE} represents the bilinear pairing operation, and its execution time is 6.4164 ms; T_{GA} represents bilinear point-to-point addition operation, and its execution time is 0.0146 ms; T_{MPT} represents the MapToPoint operation, and its execution time is 1.3277 ms; T_H represents a one-way hash function operation with a run time of 0.002 ms.

In a single RUS domain, we take the scenario where the vehicle only transmits one message as an example for analysis. In the whole process, the computational consumption generated by the vehicle 's signature is $T_{PM}+2T_H+T_{PA} \approx 0.7348$ ms. After the vehicle sends a message, ECN needs to authenate it and feedback the results to RSU_i, and these operations will result in the computational cost of $6T_{PM}+2T_{PA}+4T_H \approx 4.4268$ ms. We analyze the computational consumption of several other schemes in the same scenario, as shown in Table 2. By comparing the experimental data from Fig. 9, we clearly see that our consumption in the signature phase is the least, but it is not as good as the schemes DCBA and ABAK in the authentication phase.

Next, we consider the peak scenario of vehicle travel. In a short period of time, many vehicles send messages to ECN, ECN will batch authenticate the corresponding messages. In most cases, batch authentication of ECN is passed at one time.

Table 2: The computational cost of a messages

Scheme	Signature consumption for a message	Authentication consumption for a message
Our scheme	$T_{PM} + 2T_H + T_{PA} \approx 0.7348$ ms	$6T_{PM} + 2T_{PA} + 4T_H \approx 4.4268$ ms
DCBA	$T_{PM} + T_H + T_{PA} \approx 0.7418$ ms	$5T_{PM} + 2T_H + 3T_{PA} \approx 2.9592$ ms
CAEC	$3T_{PM} + 2T_H + T_{PA} \approx 2.2154$ ms	$2T_{GE} + 2T_{PM} + 2T_H + T_{PA} \approx 14.312$ ms
ABAH	$T_{PM} + T_H \approx 0.7378$ ms	$2T_{PM} + T_{PA} + T_H \approx 1.4776$ ms

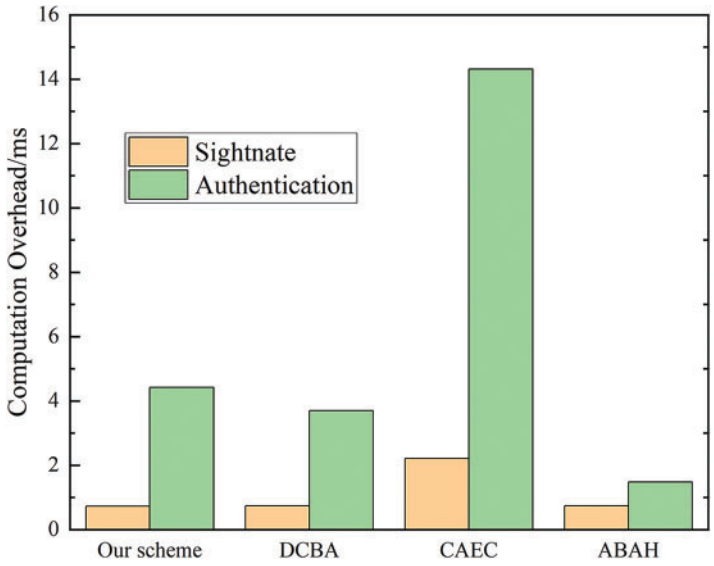

Figure 9: The authentication and signature computation overhead of a message

We assume that within the communication domain of RSU_i, RSU_i received n messages, and the number of ECN selected by RSU_i is x. We assume that each ECN authenticates the same number of messages, which are n/x. The computation cost to complete the authentication of n messages is $T_{MAX}\left\{T^1_{HID}, T^2_{HID}, T^3_{HID} \ldots\ldots T^x_{HID}\right\} + (2x+1)\,T_{PM} + xT_{PMS} + 2xT_H + T_{PA}$, T_{MAX} is the maximum time consumption for each ECN to verify n/x messages. Setting aside the impact of time consumption caused by network transmission, we assume that each ECN takes the same amount of time to authenticate the same number of messages. Therefore, the time to authenticate this batch of messages in the entire RSU domain is $\left[2\left(n + x^2 + x/x\right)\right] T_{PM} + 2\left(n + x^2\right) T_H + \left(n + x^2/x\right) T_{PMS} + 2T_{PA}$ ms.The computational consumption of scheme CAEC, DCBA and ABAH is $2nT_{GE} + (\mathrm{n}+1)\,T_{PM} + 2nT_H + nT_{PA}$ ms, $8nT_{PM} + nT_{PMS} + 3nT_H + 4nT_{PA}$ ms and $(n+1)T_{PM} + nT_{PMS} + nT_H + nT_{PA}$ ms, respectively. We conducted experiments on scenarios with 500, 1000, 1500, 2000, and 2500 messages generated in the entire RSU domain during a certain interval time. The experimental results are shown in Fig. 10. When 500 messages are generated in the domain, the computation time of our scheme is 383.61 ms ($x = 2$), which is close to ABAH. Our schemes are 392.53 ms. However, in the peak scenario, the advantages of our scheme become obvious.

We also consider the influence of the number of ECN. Fig. 11 reveals the correlation between time consumption and ECNs. With the addition of more ECN, the time consumption of message

authentication becomes less, because ECN greatly alleviates the authentication pressure of RSU and accelerates the whole process.

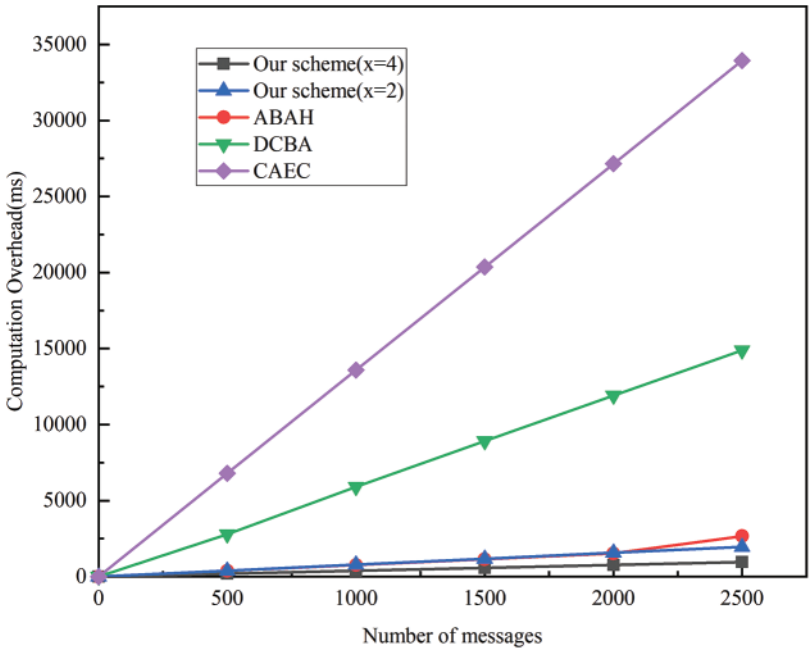

Figure 10: Comparison of computational consumption

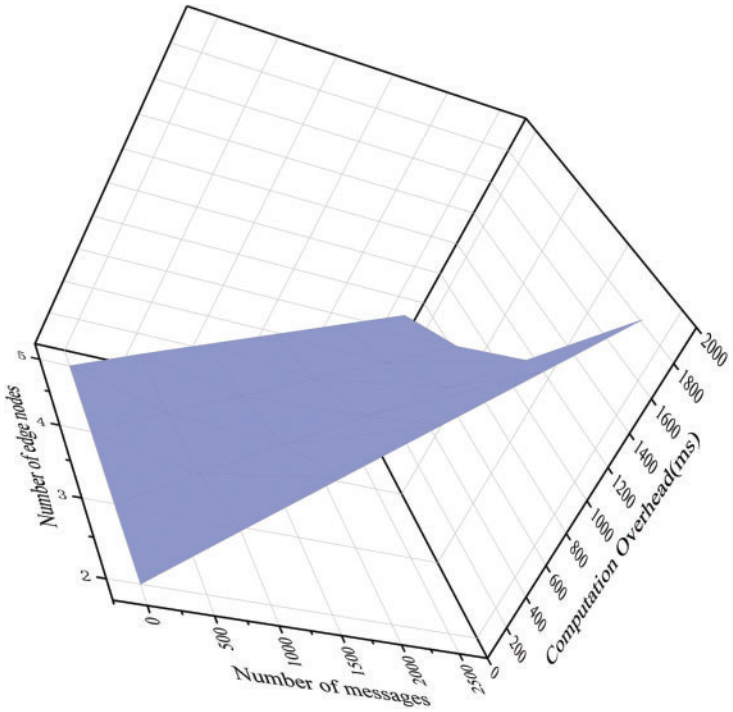

Figure 11: The correlation between the number of ECNs and computational consumption

6.2.2 Communication Overhead

In the part, we consider another factor, namely the communication overhead. We have done experiments on the communication consumption between DCBA, CAEC, ABAH and this scheme. Taking the communication domain of an RSU as an example. At a certain time, the sum of the transmission consumption of all vehicles and the transmission consumption of ENC is the total communication overhead in the domain.

In Chapter 6.1, we know that the P_1 is 64 bytes and P is 20 bytes. Therefore, the elements in group G_1 is 128 bytes and the elements in G are is 40 bytes. Suppose that the message m is 16 bytes and the time-stamp is 4 bytes.

We clearly know the communication consumption of each scheme from Table 3. In this scheme, the data sent from vehilce to ECN is $\{M_i,\ \delta_m^i,\ PID_V^i,\ T_i\}$, where is $PID_i^1,\ PID_i^2,\ \delta_m^1 \in \mathrm{G},\ \delta_m^2,\ \in Z_q^*$, So the communication overhead of sending a message is $3 \times 40 + 1 \times 20 + (16 + 4) = 160$ bytes. The result of the ECN feedback to the RSU is $\{M_T, \theta_i, HID_i, PID_v^i\}$ and the size of these messages transmitted is $4 \times 40 + 1 \times 20 + (16 + 4) = 200$ bytes, where is $HID_i \in \mathrm{G},\ \theta_i \in Z_q^*$. Similarly, the communication consumption of the scheme DCBA is 260 bytes, CAEC is 168 bytes, ABAH is 240 bytes. Fig. 12 indicates that Our scheme is better than others. (We set the number of ECNs to be 2). We derive the relationship between ECN and communication overhead. In the single message authentication scenario, This scheme is not as good as the other three schemes in terms of communication overhead, but we know that the impact of the number of ECN on transmission consumption can be ignored in the scenario of a large number of message authentication from Fig. 13.

Table 3: Communication consumption

Scheme	Communication consumption of a message	Communication consumption of n messages
Our scheme	360 bytes	160n + 200m bytes (m is number of edge computation nodes)
DCBA	260 bytes	260n bytes
CAEC	168 bytes	168n bytes
ABAH	240 bytes	240n bytes

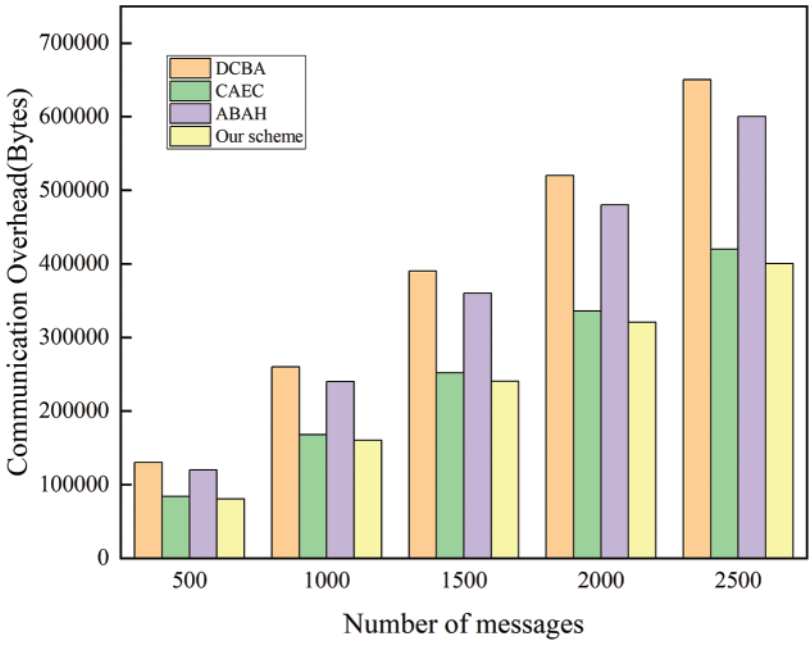

Figure 12: Comparison of communication consumption of message authentication

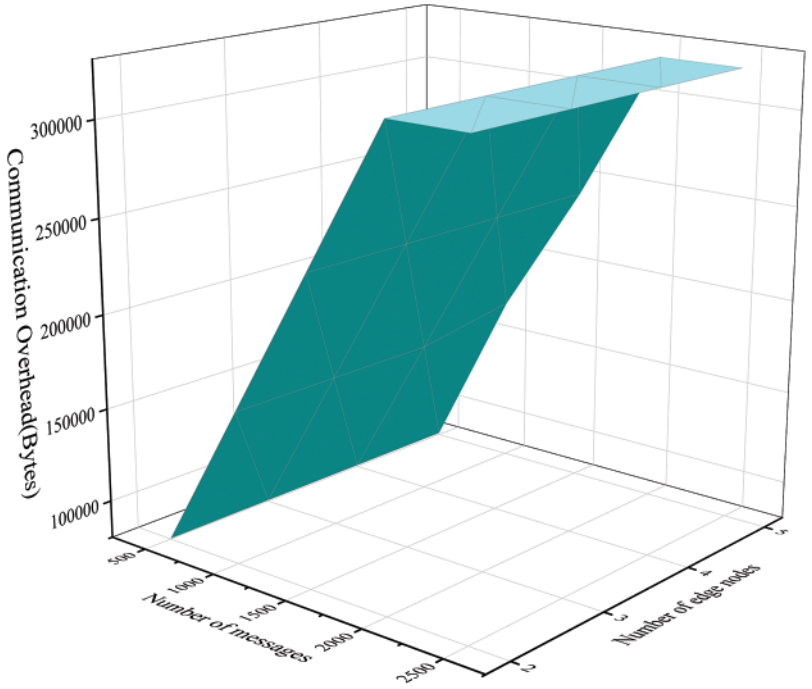

Figure 13: Impact of the number of ECN on communication consumption

6.2.3 *Packet Loss Rate*

In high-density traffic scenarios, it is easy to cause packet loss events during transmission. To reflect advantages of this scheme in this aspect, in the part, we use the Veins simulation architecture to conduct experiments. The calculation formula of packet loss rate is PLR = A − B/A. A is the total packet, and B is the received packet. Table 4 is the setting of simulation parameters.

Table 4: The related parameters

Parameter	Default value
Running time	100 s
Vehicle broadband	200 kbit/s
Speed range	40–120 km/h
Signal transmission range of RSU	800 m
The interval of information sending	300 ms
The setting spacing of RSU	3 km
Road length	12 km
Road lane	two-way four-lane

Fig. 14 is the PLR results of the experiment under different vehicle traffic scenarios. From the experimental results, it can be known that the throughput of IoV is easily affected by the size of traffic flow. The larger the traffic flow, the higher the packet loss rate. In our scheme (only one ECN is set), the traffic flow is in the range of 2∼16 vehicles per metre, the PLR is 2.91%∼3.06%, and we increased ECN to 2 vehicles, the PLR is 2.89%∼2.93%. All in all, our scheme still has certain advantages and is suitable for authentication in peak traffic scenarios.

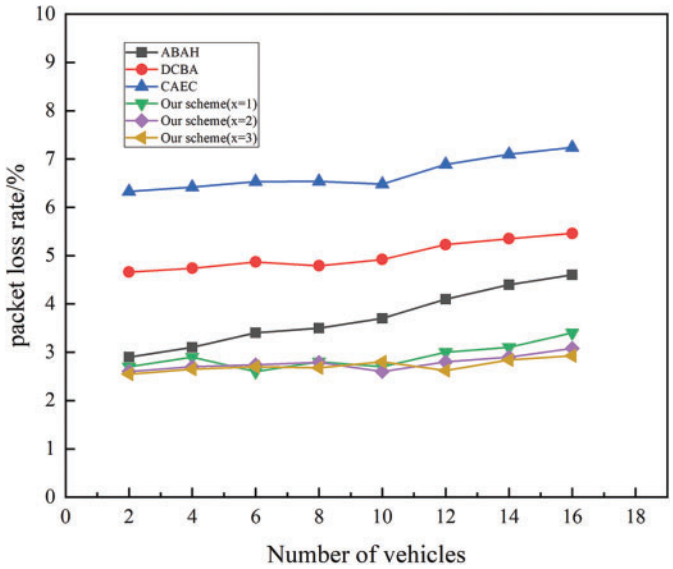

Figure 14: Comparison of PLR

7 Conclusions and Prospected

This paper is devoted to the application research of IoV, we combine blockchain and edge computing technology to design a novel scheme for message authentication. Unlike in traditional authentication scheme, ECN in the scheme will assume part of the responsibility of RSU for the first round of message authentication, which can make full use of the idle computing resources and accelerate the overall authentication process of messages. In addition, TA and RSU in the scheme are used as blockchain nodes to maintain vehicle information. By accessing the information on the blockchain, RSU can authenticate vehicles from different domains and enable vehicles to achieve cross-domain authentication. The scheme we designed also has limitations. Considering that ECN are temporarily parked vehicles, there is uncertainty in identity authentication work. At the same time, frequent replacement of ECN will also increase the resource consumption of the system.

At present, most of authentication mechanisms are mainly aimed at the security characteristics required by IoV. The higher the security of the scheme, the greater the corresponding computational communication consumption. How to achieve a perfect balance between the two and design a high security and performance authentication scheme is still our goal in the future. Nowadays, blockchain, artificial intelligence and other technologies can be used in authentication schemes, and it is also feasible to design authentication schemes by combining multiple technologies.

Acknowledgement: Throughout the submission process, we are very grateful for the patient guidance of editors and reviewers. We thank our colleagues and experts for their guidance and support.

Funding Statement: This research was funded by Guangxi Natural Science Foundation General Project—Research on Visual Positioning and Navigation Robot Based on Deep Learning, Project Number: 2023GXNSFAA026025.

Author Contributions: The authors confirm contribution to the paper as follows: study conception and design: Qiping Zou, Zhong Ruan; data collection: Zhong Ruan; analysis and interpretation of results: Zhong Ruan; draft manuscript preparation: Huaning Song. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] A. Hammoud, "AI, blockchain, and vehicular edge computing for smart and secure IoV: Challenges and directions," *IEEE Internet Things Mag.*, vol. 3, no. 2, pp. 68–73, 2020. doi: 10.1109/IOTM.0001.1900109.

[2] P. M. Rao, S. Jangirala, S. Pedada, A. K. Das, and Y. H. Park, "Blockchain integration for IoT-Enabled V2X communications: A comprehensive survey, security issues and challenges," *IEEE Access*, vol. 3281844, no. 11, pp. 54476–54494, 2023.

[3] M. T. Abbas and M. Afaq, "SD-IoV: SDN enabled routing for internet of vehicles in road-aware approach," *J. Ambient Intell. Hum. Comput.*, vol. 11, no. 3, pp. 1265–1280, 2020. doi: 10.1007/s12652-019-01319-w.

[4] M. El, S. Samira, and E. G. Abdelaziz, "Internet of vehicles: Concept, process, security aspects and solutions," *Multimed. Tools Appl.*, vol. 81, no. 12, pp. 16563–16587, 2020.

[5] J. Cui, L. Wei, J. Zhang, Y. Xu, and H. Zhong, "An efficient message-authentication scheme based on edge computing for vehicular ad hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 1621–1632, 2018. doi: 10.1109/TITS.2018.2827460.
[6] S. O. Ogundoyin, "An autonomous lightweight conditional privacy-preserving authentication scheme with provable security for vehicular ad-hoc networks," *Int. J. Comput. Appl.*, vol. 42, no. 2, pp. 196–211, 2020. doi: 10.1080/1206212X.2018.1477320.
[7] S. Jiang, X. Zhu, and L. Wang, "An efficient anonymous batch authentication scheme based on HMAC for VANETs," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 8, pp. 2193–2204, 2016. doi: 10.1109/TITS.2016.2517603.
[8] C. Yang, J. Peng, Y. Xu, Q. Wei, L. Zhou and Y. Tang, "Edge computing-based VANETs' anonymous message authentication," *Symmetry*, vol. 14, no. 12, pp. 2662, 2022. doi: 10.3390/sym14122662.
[9] B. Palaniswamy, S. Camtepe, E. Foo, and J. Pieprzyk, "An efficient authentication scheme for intra-vehicular controller area network," *IEEE Trans. Inf. Forensics Secur.*, vol. 15, pp. 3107–3122, 2020. doi: 10.1109/TIFS.2020.2983285.
[10] Z. Shen, F. Ren, and H. Wang, "Combining blockchain and crowd-sensing for location privacy protection in Internet of vehicles," *Veh. Commun.*, vol. 45, pp. 100724–100732, 2024. doi: 10.1016/j.vehcom.2023.100724.
[11] M. Azees, P. Vijayakumar, and L. J. Deboarh, "EAAP: Efficient anonymous authentication with conditional privacy-preserving scheme for vehicular Ad Hoc networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 9, pp. 2467–2476, 2017. doi: 10.1109/TITS.2016.2634623.
[12] J. Cui, D. Wu, J. Zhang, Y. Xu, and H. Zhong, "An efficient authentication scheme based on semi-trusted authority in VANETs," *IEEE Trans. Vehicular Technol.*, vol. 68, no. 3, pp. 2972–2986, 2019. doi: 10.1109/TVT.2019.2896018.
[13] J. Cui, J. Wen, S. Han, and H. Zhong, "Efficient privacy-preserving scheme for real-time location data in vehicular ad-hoc network," *IEEE Internet Things J.*, vol. 5, no. 5, pp. 3491–3498, 2018. doi: 10.1109/JIOT.2018.2797206.
[14] L. Zhang, "OTIBAAGKA: A new security tool for cryptographic mix-zone establishment in vehicular ad hoc networks," *IEEE Trans. Inf. Forensics Secur.*, vol. 12, no. 6, pp. 2998–3010, 2017. doi: 10.1109/TIFS.2017.2730479.
[15] W. Xiong, R. Wang, Y. Wang, F. Zhou, and X. Luo, "CPPA-D: Efficient conditional privacy-preserving authentication scheme with double-insurance in VANETs," *IEEE Trans. Vehicular Technol.*, vol. 70, no. 4, pp. 3456–3468, 2021. doi: 10.1109/TVT.2021.3064337.
[16] W. Shi, X. Zhang, Y. Wang, and Q. Zhang, "Edge computing: State-of-the-art and future directions," *J. Comput. Res. Dev.*, vol. 56, no. 1, pp. 69–89, 2019.
[17] M. Raya and J. -P. Hubaux, "Securing vehicular ad hoc networks," *J. Comput. Secur.*, vol. 15, no. 1, pp. 39–68, 2007. doi: 10.3233/JCS-2007-15103.
[18] Y. Sun, R. Lu, X. Lin, X. Shen, and J. Su, "An efficient pseudonymous authentication scheme with strong privacy preservation for vehicular communications," *IEEE Trans. Vehicular Technol.*, vol. 59, no. 1, pp. 3589–3603, 2010. doi: 10.1109/TVT.2010.2051468.
[19] R. Lu, X. Lin, H. Zhu, P. -H. Ho, and X. Shen, "ECPP: Efficient conditional privacy preservation protocol for secure vehicular communications," in *Proc. IEEE INFOCOM*, Phoenix, AZ, USA, 2008, pp. 1229–1237.
[20] Y. Qing, Z. Xiao, Z. Jing, and X. L. Wang, "A novel authentication and key agreement scheme for Internet of Vehicles," *Future Gener. Comput. Syst.*, vol. 145, no. 1, pp. 415–428, 2023. doi: 10.1016/j.future.2023.03.037.
[21] K. -A. Shim, "CPAS: An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks," *IEEE Trans. Vehicular Technol.*, vol. 61, no. 4, pp. 1874–1883, 2012. doi: 10.1109/TVT.2012.2186992.
[22] H. Vasudev, D. Das, and A. V. Vasilakos, "Secure message propagation protocols for IoVs communication components," *Comput. Electr. Eng.*, vol. 82, no. 1, pp. 106555, 2020. doi: 10.1016/j.compeleceng.2020.106555.

[23] H. Vasudev, V. Deshpande, D. Das, and S. K. Das, "A lightweight mutual authentication protocol for V2V communication in internet of vehicles," *IEEE Trans. Vehicular Technol.*, vol. 69, no. 6, pp. 6709–6717, 2020. doi: 10.1109/TVT.2020.2986585.

[24] C. Zhang, X. Lin, R. Lu, P. -H. Ho, and X. Shen, "An efficient message authentication scheme for vehicular communications," *IEEE Trans. Vehicular Technol.*, vol. 57, no. 6, pp. 3357–3368, 2008. doi: 10.1109/TVT.2008.928581.

[25] B. Liu and L. Zhang, "An improved identity-based batch verification scheme for VANETs," in *Proc. 2013 5th Int. Conf. Intell. Netw. Coll. Syst.*, Xi'an, China, 2013, pp. 809–814.

[26] C. Wang, R. Huang, J. Shen, J. Liu, P. Vijayakumar and N. Kumar, "A novel lightweight authentication protocol for emergency vehicle avoidance in VANETs," *IEEE Internet Things J.*, vol. 8, no. 18, pp. 14248–14257, 2021. doi: 10.1109/JIOT.2021.3068268.

[27] S. H. Islam, M. S. Obaidat, P. Vijayakumar, E. Abdulhay, F. Li and M. K. C. Reddy, "A robust and efficient password-based conditional privacy preserving authentication and group-key agreement protocol for VANETs," *Future Gener. Comput. Syst.*, vol. 84, no. 1, pp. 216–227, 2018. doi: 10.1016/j.future.2017.07.002.

[28] Z. Xu, W. Liang, K. -C. Li, J. Xu, and H. Jin, "A blockchain-based roadside unit-assisted authentication and key agreement protocol for internet of vehicles," *J. Parallel Distr. Comput.*, vol. 149, no. 1, pp. 29–39, 2021. doi: 10.1016/j.jpdc.2020.11.003.

[29] L. Song, G. Sun, H. Yu, X. Du, and M. Guizani, "FBIA: A fog-based identity authentication scheme for privacy preservation in internet of vehicles," *IEEE Trans. Vehicular Technol.*, vol. 69, no. 5, pp. 5403–5415, 2020. doi: 10.1109/TVT.2020.2977829.

[30] P. Vijayakumar, M. Azees, S. A. Kozlov, and J. J. Rodrigues, "An anonymous batch authentication and key exchange protocols for 6G enabled VANETs," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 1630–1638, 2021. doi: 10.1109/TITS.2021.3099488.

[31] Y. He, G. Li, and J. Liu, "Conditional privacy protection authentication scheme based on fog computing and multi-TA in Internet of vehicles," *Appl. Res. Comput.*, vol. 1, pp. 6–23, 2022.

[32] S. -F. Tzeng, S. -J. Horng, T. Li, X. Wang, P. -H. Huang and M. K. Khan, "Enhancing security and privacy for identity-based batch verification scheme in VANETs," *IEEE Trans. Vehicular Technol.*, vol. 66, no. 4, pp. 3235–3248, 2015.

[33] X. Zhu, S. Jiang, L. Wang, and H. Li, "Efficient privacy-preserving authentication for vehicular ad hoc networks," *IEEE Trans. Vehicular Technol.*, vol. 63, no. 2, pp. 907–919, 2013. doi: 10.1109/TVT.2013.2294032.

[34] N. Andola, Raghav, and V. K. Yadav, "A lightweight blockchain for authentication and anonymous authorization in IoD," *Wirel. Pers. Commun.*, vol. 119, no. 1, pp. 343–362, 2021. doi: 10.1007/s11277-021-08214-8.

[35] G. Xu, H. Bai, and J. Xing, "SG-PBFT: A secure and highly efficient distributed blockchain PBFT consensus algorithm for intelligent Internet of vehicles," *J. Parallel Distr. Comput.*, vol. 164, no. 1, pp. 1–11, 2022. doi: 10.1016/j.jpdc.2022.01.029.

[36] W. Peng, N. Han, and C. Song, "CAEC: Certificateless IOV identity authentication scheme in edge computing environment," (in Chinese), *J. Beijing Univ. Posts Telecommun.*, vol. 45, no. 2, pp. 46–51, 2022.

[37] A. K. Sutrala, P. Bagga, A. K. Das, N. Kumar, J. J. Rodrigues and P. Lorenz, "On the design of conditional privacy preserving batch verification-based authentication scheme for Internet of vehicles deployment," *IEEE Trans. Vehicular Technol.*, vol. 69, no. 5, pp. 5535–5548, 2020. doi: 10.1109/TVT.2020.2981934.

[38] I. Ali, Y. Chen, N. Ullah, R. Kumar, and W. He, "An efficient and provably secure ECC-based conditional privacy-preserving authentication for vehicle-to-vehicle communication in VANETs," *IEEE Trans. Vehicular Technol.*, vol. 70, no. 2, pp. 1278–1291, 2021. doi: 10.1109/TVT.2021.3050399.

Appendix A

```
----------RSU----------
let RSU=
in(Sch,(xPubTA:bitstring,xPubRSU:bitstring,xKey:bitstring,xSR:bitstring))
in(Pch,(xAID1:bitstring,xAID2:bitstring;xPIDOBU:bitstring))
new PubOBU: bitstring
let PubOBU=xor(xPIDOBU,H(MULT(xSR,xAID1))) in
new mid:bitstring
let mid=Add(MULT(xSR,XAID1),MULT(xSR,PubOBU)) in
if mid=xAID2 then
event AuthVehicle(xAID1,xAID2);
new t:bitstring
new PID1:bitstring
let PID1=ECPM(t,P) in
new PID2:bitstring
let PID2=xor(PubOBU,H(MULT(t,MULT(xSR,xPubTA)))) in
new t2:bitstring
let xPID21=ECPM(t2,P) in
let xPID22=xor(PubOBU2,h(t2,SR,xPubTA) in
new StPID:bitstring
let StPID=xor(t,H(xKey)) in
new AuthRSU:bitstring
let AuthRSU=xor(H(MULT(xSR,xAID1)),H(xKey)) in
out(Pch,(PID21:bitstring,PID22:bitstring,t2:bitstring))
out(Pch,(PID1:bitstring,PID2:bitstring,StPID:bitstring,AuthRSU:bitstring))
new Rbitstring
new HID1:bitstring
let HID1=ECPM(R,P) in
new HID2:bitstring
let HID2=xor(PubOUB2,H(MULT(R,MULT(SR2,xPubRSU)))) in
out(Pch,(HID1:bitstring,HID2:bitstring,Rbitstring))
in(Pch,(xSig:bitstring,xHID2:bitstring,xxPID21:bitstring,xxPID22:bitstring,xM:bitstring));
new RSUmid:bitstring
let RSUmid=ECPM(xSig,P) in
new A:bitstring
let A=Add(MULT(xxPID21,H3(xxHID2),MULT(xxHID1,H3(xM)) in
if RSUmid = A then
event RSUAuthEdgenode(xxPID21,XXPID22);
else
0
```

```
----------Basic definition----------
free Sch:channel [private].
free Pch:channel.
const P.bitstring.
free PubOBU2:bitstring
free SR2:bitstring [private]
free t2:bitstring [private]
free m:bitstring.
free st:bitstring [private]. //TA的秘钥
free xPID1:bitstring.
free xPubOBU2:bitstring.
free Sv2:bitstring[private]
fun H1(bitstring,bitstring,bitstring):bitstring
fun H2(bitstring,bitstring):bitstring.
fun H(bitstring):bitstring.
fun ECPM(bitstring,bitstring):bitstring
fun MULT(bitstring,bitstring):bitstring.
fun ADD(bitstinrg,bitstring):bitstring
fun xor(bitstring,bitstring):bitstring.
equation forall m:bitstring,n:bitstring;xor(xor(m,n),n)=m
event AuthVehicle(xAID1,xAID2);
event RSUAuthEdgenode(xxPID21,XXPID22);
query XAID1:bitstring,xAID2:bitstring;xxPID21:bitstring,XXPID22:bitstring,
inj-event(RSUAuthEdgenode(xxPID21,XXPID22))==>inj-event(AuthVehicle(xAID1,xAID2))
query attacker(Key).
query attacker(Sv).
query attacker(r).
query attacker(xt).
query attacker(K).
query attacker(PubOBU).
```

```
----------TA----------
let TA=
new PubTA:bitstring
let PubTA=ECPM(St,P) in
new PubRSU:bitstring
new SR:bitstring [private]
let PubRSU=ECPM(SR,P) in
new PubOBU:bitstring
new Sv:bitstring [private]
let PubOBU=ECPM(Sv,P) in
new Key:bitstring
let Key=H(MULT(Sv,PubTA)) in
out(sch,(PubTA:bitstring,PubRSU:bitstring,Key:bitstring,Sv:bitstring));
out(sch,(PubTA:bitstring,PubOUB:bitstring,Key:bitstring,PubRSU:bitstring,SR:bitstring));
else
0.
```

```
----------Edge Computing nodes----------
let Edge Computing nodes=
in(Pch,(xPID21:bitstring,xPID22:bitstring,xt2:bitstring))
in(Pch,(xHID1:bitstring,xHID2:bitstring,xRbitstring))
in(Pch,(xxPID1:bitstring,xxPID2:bitstring,xSig1:bitstring,xSig2:bitstring,xm:bitstring));
new mid:bitstring
let mid=Add(MULT(xxPID1,H(xxPID2)),MULT(xSig1,H(xm))) in
if ECPM(xSig2,P)=mid then
event EdgenodeAuthVehicle(xxPID1,XXPID2);
new Sig:bitstring
let Sig =Add(MULT(t2,H(xHID2)),MULT(xR,H(xm)) in
out(Pch,(Sig:bitstring,HID2:bitstring,xPID21:bitstring,xPID22:bitstring,xm:bitstring));
else
0
```

```
Process
(       (!Vehicle)|
        (!RSU)|
        (!EdgeComputingnodes)|
        (!TA)
)
```

```
----------Vehicle----------
let Vehicle=
in(Sch,(xPubTA:bitstring,xPubOBU:bitstring,xKey:bitstring,xPubRSU:bitstring,xSv:bitstring));
new r:bitstring [private]
new AID1: bitstring
let AID1=ECPM(r,P) in
new AID2:bitstring
let AID2=MULT(add(r,xSv)) in
new PIDOUB:bitstring
let PIDOBU=xor(xPubOBU,H(MULT(r,xPubRSU)) in
out(Pch,(AID1:bitstring,AID2:bitstring;PIDOBU:bitstring));
in(Pch,(xPID1:bitstring,xPID2:bitstring,xStPID:bitstring,xAuthRSU:bitstring))
new mid:bitstring
let mid=xor(xAuthRSU,H(MULT(r,xPubRSU))) in
if H(xKey)=mid then
new t:bitstring
let t=xor(xStPID,H(xKey)) in
event AuthRSU(xPubRSU);
new K:bitstring
new Sig1:bitstring
let Sig1=ECPM(K,P) in
new Sig2:bitstring
let sig2=Add(MULT(t,H(PID2)),MULT(K,H(m))) in
out(Pch,(xPID1:bitstring,xPID2:bitstring,Sig1:bitstring,Sig2:bitstring,m:bitstring));
else
0.
```

Figure A1: Coding of the scheme