electronics

MDPI

Article

A Noval and Efficient Three-Party Identity Authentication and Key Negotiation Protocol Based on Elliptic Curve Cryptography in VANETs

Wenping Yu [1], Rui Zhang [1], Maode Ma [2,*] and Cong Wang [1]

[1] College of Artificial Intelligence, Tianjin University of Science and Technology, Tianjin 300457, China; yuwenping@tust.edu.cn (W.Y.); zhangrui1113@mail.tust.edu.cn (R.Z.); wangcongjcdd@tust.edu.cn (C.W.)
[2] College of Engineering, Qatar University, Doha P.O. Box 2713, Qatar
* Correspondence: acadmmd@gmail.com

Abstract: In the process of vehicles transitioning from conventional means of transportation to mobile computing platforms, ensuring secure communication and data exchange is of paramount importance. Consequently, identity authentication has emerged as a crucial security measure. Specifically, effective authentication is required prior to the communication between the On-Board Unit (OBU) and Roadside Unit (RSU). To address vehicle identity authentication challenges in the Internet of Vehicles (VANETs), this paper proposes a three-party identity authentication and key agreement protocol based on elliptic curve public key cryptography. Considering issues such as vehicle impersonation attacks, RSU impersonation attacks, and vehicle privacy breaches in existing schemes within wireless mobile environments, this protocol introduces a trusted registry center that successfully enables mutual authentication between OBU and RSU. The proposed protocol not only enhances the VANETs system's ability to withstand security threats but also improves the credibility and efficiency of the authentication process.

Keywords: VANETs; elliptic curve; three-party authentication; key agreement

Citation: Yu, W.; Zhang, R.; Ma, M.; Wang, C. A Noval and Efficient Three-Party Identity Authentication and Key Negotiation Protocol Based on Elliptic Curve Cryptography in VANETs. *Electronics* **2024**, *13*, 449. https://doi.org/10.3390/electronics13020449

Academic Editor: Zbigniew Kotulski

Received: 6 December 2023
Revised: 27 December 2023
Accepted: 10 January 2024
Published: 22 January 2024

1. Introduction

The rapid advancement of computing technologies and the widespread implementation of communication mechanisms have facilitated the development of connected vehicles, which are revolutionizing the automotive field at a fast pace. Simultaneously, global urbanization and increasing traffic flow pose challenges to transportation systems, including congestion, accidents, pollution, and energy wastage [1–4]. To address these issues effectively, VANETs technology integrates a network system by establishing connections among vehicles, individuals, and sensing devices to enable bidirectional data transmission. Its functionalities [5] encompass information sharing, vehicle safety management, condition monitoring, driving behavior analysis, real-time traffic management as well as intelligent road network planning. Despite its potential for significant enhancements in vehicle safety standards along with improved driving comfort and energy efficiency levels; however, VANETs's widespread adoption necessitates addressing concerns related to communication security and vehicle privacy [6–9].

In the VANETs, identity authentication serves as a pivotal measure to ensure system security. By implementing effective authentication mechanisms, unauthorized vehicles can be prevented from accessing the network, thereby effectively mitigating information leakage and potential malicious activities. This endeavor not only aids in establishing a trusted vehicle communication environment but also assumes a crucial role in transportation coordination and intelligent scenarios. Simultaneously, it guarantees that vehicle communication and cooperation are founded on reliable identities, thus enhancing the overall security and reliability of the entire VANETs system.

1.1. Related Works

In the realm of VANETs, the pertinent technologies for implementing an identity authentication scheme can be categorized into three distinct groups: identity authentication schemes based on Public Key Infrastructure (PKI) [10,11]; batch identity authentication schemes [12–18]; blockchain-based authentication schemes [19–22]; and identity authentication scheme in multiserver architecture [23].

Currently, the prevailing authentication scheme in the field of Internet of Vehicles is based on Public Key Infrastructure (PKI). Xiong et al. [10] proposed a lightweight authentication protocol utilizing hash functions and exclusive OR operations, which is suitable for VANETs communication model. However, this scheme relies on a trusted third party (TA) during the authentication process, limiting its service coverage when TA is offline. Shohei et al.'s Meta-PKI [11], on the other hand, establishes trust relationships between certificate authorities outside the blockchain to construct an authentication path uniformly while leveraging blockchain technology to expedite certificate configuration and ensure path validity. However, this method solely utilizes edge connection state for building certification paths and suffers from limitations due to cross-certification and policy reasons, resulting in potential invalidity of some public key certificates. Furthermore, this scheme presents challenges such as high processing load for PKI clients and complexity in managing constraints and policies associated with them. Although PKI has made significant progress in enhancing verification efficiency, it also reinforces reliance on trust centers and increases system management complexity.

In order to provide a scheme for identity batch authentication, Tzeug et al. [12] proposed an identity-based anonymous batch authentication scheme with strong privacy protection that utilizes a tamper-proof device to store the system master key. By allowing vehicles to generate dynamic pseudonym identities before communicating with the RSU, it becomes difficult for attackers to determine whether different messages are sent by the same vehicle, effectively reducing the risk of vehicle tracking and solving complex replacement issues. Zhang et al. [13] proposed an identity-based batch authentication scheme for VANETs that does not require certificates on vehicles and provides a method for authenticating multiple messages simultaneously, significantly improving message verification efficiency; however, this scheme has limitations in resisting replay attacks and impersonation attacks. Lu et al. [14] proposed a V2I (vehicle-to-infrastructure) communication batch authentication scheme that is vulnerable to tampering attacks [24]. Chim et al. [15] improved upon Zhang et al.'s [13] work by providing anonymity for VANET users while having lower computational overhead; however, it still has limitations in effectively resisting impersonation attacks. Additionally, Azees et al.'s efficient anonymous authentication scheme prevents malicious vehicles from entering VANETs [16]. Zhou et al.'s [17] security-enhanced solution introduced in 2022 can resist signature forgery attacks in VANETs while further ensuring vehicle identity privacy; furthermore, Zhang et al.'s trust currency called TCoin was introduced as an anonymous reporting mechanism in VANETs [18]. Although these schemes have improved authentication efficiency, they still need further optimization to improve antiattack capabilities.

To establish a more secure and privacy-preserving authentication mechanism for the VANETs, several blockchain-based authentication schemes have been proposed. The BUA authentication protocol, introduced by Liu et al. [19], enables vehicles to generate multiple pseudonyms; however, this scheme lacks an effective revocation mechanism against malicious vehicles as the pseudonyms are verified based on specific blocks. Feng et al. [20] designed a blockchain-assisted privacy-preserving authentication system that utilizes smart contracts to record and revoke vehicle pseudonyms and public keys. Nevertheless, each authentication process requires querying the blockchain manager, resulting in significant verification overhead. Zheng et al. [21] proposed an innovative multi-TA network authentication protocol along with a multiserver network authentication protocol to distribute authentication tasks, reducing workload for individual entities and enhancing user authentication efficiency. Additionally, Chai et al.'s [22] blockchain-based authentication

framework transmits physical vehicle data to cyberspace through RSUs while establishing blockchains in third-party systems instead of executing consensus processes on physical nodes—aiming to create a lightweight blockchain that safeguards vehicle privacy and security while minimizing creation and maintenance costs. Although these schemes offer more effective options for identity verification in VANETs, they still face challenges related to insufficient revocation mechanisms and high verification overhead, thus, necessitating finding a balance between security and efficiency.

In the identity authentication system designed for multiserver architecture, the registry is responsible for user and server registration as well as generating master keys. Within the unified registration and authentication model, users only need to register once at the central registration center, enabling them to log in to all servers managed by the center and access corresponding network resources and services. When implementing identity authentication mode schemes under a multiserver architecture, there are primarily two types: two-party authentication mode and three-party authentication mode [25]. As a fundamental component of more complex high-level protocols, designing a secure and efficient identity authentication and key agreement protocol suitable for multiserver architecture is currently an active research area. The protocol must fulfill security requirements against impersonation attacks, password guessing attacks, known key attacks, replay attacks, etc., ensuring user privacy and forward security of the scheme [26,27]. In 2021, Chen et al.'s [23] scheme effectively resolved forgery issues along with adaptive chosen message attacks; nevertheless, its drawback lies in a high total cost of signature verification, which may impact network performance in dense scenarios. Overall, the research direction of multiserver authentication systems continues to be positive. However, researchers face quite complex challenges in balancing security and system efficiency and improving the ability to resist various attacks.

In the network environment, mutually authenticated entities are usually distributed in different physical locations, and there is a relationship between each other. In order to solve this problem, the second "what to have" and the third "what to know" technical means are widely used. Through the scheme proposed in this paper, the close connection between OBU, RSU and RC is established successfully, and the communication cost of the client is reduced through RC.

1.2. Motivation

To ensure reliable and secure communication services for the VANETs, while maintaining identity reliability and effectively protecting vehicle user privacy, numerous security protocols have been proposed by researchers. However, these protocols suffer from several limitations. For instance, the PKI-based identity authentication scheme proposed by Xiong et al. [10] proposed a lightweight authentication protocol that was suitable for the VANET communication model and effectively improved authentication efficiency; however, this scheme relied excessively on third parties. In the batch identity authentication scheme introduced by Zhang et al. [13], a method for simultaneous authentication of multiple messages without storing certificates on vehicles is proposed; however, it overlooks potential threats such as replay attacks and impersonation attacks. Furthermore, Feng et al.'s blockchain-based identity authentication scheme [20] utilizes smart contracts to record and revoke pseudonyms and public keys but significantly increases system overhead. Existing remote authentication schemes applicable to multiserver wireless mobile environments also face challenges in resisting vehicle impersonation attacks, RSU impersonation attacks, and vehicle privacy leakage issues. This paper introduces a trusted third party into the authentication process to achieve mutual authentication between vehicles and RSUs while ensuring high computational efficiency for vehicles—thus further enhancing security during VANETs communication.

1.3. Contributions

The main contributions of this paper are as follows:

Introduction of trusted third parties: The paper introduces a trusted third party (registration center RC) to actively participate in the specific authentication process, thereby achieving mutual authentication between OBU and RSU. This approach effectively mitigates security risks without introducing additional storage complexity to the intelligent terminal.

Eliminate duplicate registration: The identity authentication and key agreement protocol proposed in this paper within a multiserver architecture achieves the elimination of repetitive registration for vehicle users with each RSU, enabling unified management, authentication, and authorization of vehicles.

Efficient certification: The authentication protocol proposed in this paper demonstrates robust security across various attack scenarios while exhibiting efficient computational performance and occupying relatively minimal system memory space.

2. System Model and Preliminary

2.1. System Model

The present paper proposes an elliptic curve cryptography-based multiserver authentication and key agreement protocol for the VANET. The network model comprises three primary entities: RC, OBU, and RSU. Wired communication serves as the mode of communication between RSU and RC.

The OBU serves as the fundamental component of VANETs technology, being installed within vehicles to facilitate intervehicle and infrastructure communication, vehicle data collection and transmission, as well as real-time traffic information reception. These functionalities are achieved through integrated communication modules such as Bluetooth, Wi-Fi, GSM, LTE, 5G, among others.

The RSU, also known as the roadside unit, plays a pivotal role in VANETs technology and is typically deployed alongside roadways or at traffic intersections. It serves as a crucial link between vehicles and the transportation infrastructure, facilitating data exchange with on-board devices through wireless communication technologies such as Wi-Fi, LTE, 5G, etc.

The RC refers to the Registration Center, which serves as a fully trusted third party capable of withstanding various security attacks. Its primary role within this system involves generating essential parameters, such as a suitably large prime number for a well-constructed elliptic curve and constructing the system's master key to establish the public key. Additionally, RC acts as an intermediary facilitating communication between RSU and OBU.

The registry enables verification and authorization of vehicles and infrastructure, ensuring the validity and trustworthiness of both parties' identities. This mechanism differs from mutual authentication between OBU and RSU as it relies on a neutral entity for verification, reducing communication complexity and alleviating system configuration and maintenance burdens. Moreover, the third-party registry offers enhanced flexibility by allowing vehicles to access multiple applications and services using the same authentication credentials, simplifying operational processes while improving convenience and efficiency of communication. Additionally, this solution leverages advanced security features provided by registries, such as multifactor authentication, to enhance communication security while mitigating potential risks. The network model diagram is depicted in Figure 1.

However, in terms of security certification, the presence of an RC may introduce a single point of failure in the system, making it susceptible to potential large-scale monitoring. If an RC were to fail, it would impact the associated authentication process, rendering the entire system vulnerable. To counter this challenge, the implementation of multinode backup and load-balancing strategies is proposed. This ensures a seamless transition to a standby RC, thereby minimizing the risk associated with a single point of failure. Simultaneously, decentralizing the authentication process serves as an effective measure to mitigate potential threats posed by mass surveillance to the overall system.

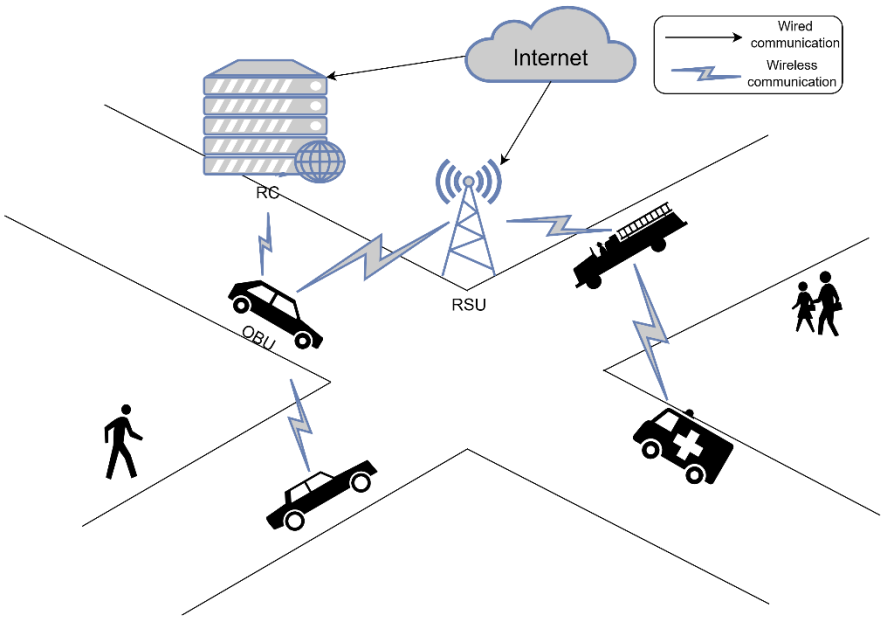

Figure 1. System network model diagram.

2.2. Preliminary

2.2.1. Elliptic Curve Public Key Cryptosystem

Elliptic Curve Cryptography (ECC) is an algorithm for public key encryption which leverages the mathematical principle of elliptic curves. In comparison to traditional cryptographic algorithms such as RSA, ECC offers enhanced performance at an equivalent security level and requires smaller key sizes. Consequently, it finds wider applicability in resource-constrained environments such as mobile devices and IoT devices.

The fundamental principle of ECC lies in leveraging the addition and multiplication operations on points residing on an elliptic curve to facilitate the generation of public and private keys, data encryption and decryption, as well as digital signature and other cryptographic operations. Its security is rooted in the discrete logarithm problem associated with elliptic curves, which involves finding a point P on the curve such that $k \times P = Q$, where k represents the private key while Q denotes the public key. Given this problem's computational infeasibility under elliptic curve settings, deriving the private key from its corresponding public key ensures ECC's robust security.

The elliptic curve group $E_p(a, b)$ is defined by the nonsingular equation E over a prime finite field F_p as shown in Formula (1), where p represents a prime number.

$$y^2 = x^3 + ax + b(mod\ p),\ a, b \in F_p,\ \Delta = 4a^3 + 27b^2 mod\ p \neq 0 \quad (1)$$

Lemma 1 (Elliptic Curve Discrete Logarithm Problem (ECDLP))**.** *Given a fixed point $G \in E_p(a, b)$ and $P = KG \in E_p(a, b)$, the computation of $k \in Z_q^*$ for the discrete logarithm problem exhibits both ease and difficulty.*

Lemma 2 (Elliptic Curve Diffie–Helman Problem (ECDHP))**.** *The security of the ECDHP key exchange system relies on the computational intractability of solving the Elliptic Curve Discrete Logarithm Problem (ECDLP). Given $G, xG, yG \in E_p(a, b)$, computing $xyG \in E_p(a, b)$ is considered to be a highly challenging task.*

2.2.2. Authentication Mode in Multiserver Architecture

The utilization of third-party authentication mitigates the risks associated with identity theft and fraud by incorporating advanced security features, such as multifactor authentication. Moreover, it ensures adherence to stringent security standards and supervision,

alleviates the burden of user authentication, enhances user-friendliness, and effectively reduces computational complexity.

The multiserver architecture described in this paper refers to the sharing of information sent by vehicles among multiple RSUs, eliminating the need for repetitive vehicle registration with each RSU. This approach enables unified management, authentication, and authorization of vehicle information. The authentication system based on the multiserver architecture primarily consists of two modes: two-party authentication and three-party authentication. In three-party authentication, an independent third-party agency is involved to verify the identity of individuals or entities. The chosen intermediary possesses the expertise and resources necessary for assuming responsibility for authentication while ensuring the validity of trusted identities for both parties involved. Compared to mutual authentication, third-party authentication simplifies the system complexity by eliminating the requirement for configuring encryption infrastructure like digital certificates. It streamlines the identity verification process and reduces costs and technical difficulties associated with it. Additionally, third-party authentication offers greater flexibility as users can utilize their credentials across multiple applications and services, thereby enhancing user convenience and experience. The third-party authentication mode is illustrated in Figure 2.

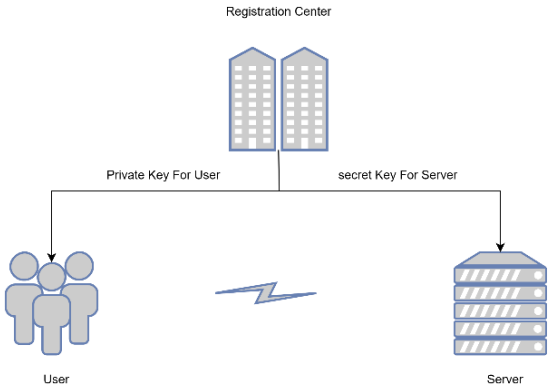

Figure 2. Third-party authentication mode.

2.3. Security Requirements

The existing authentication and key agreement schemes applicable in the V2I environment fail to withstand vehicle impersonation attacks, RSU impersonation attacks, and vehicle privacy leakage in the Internet of Vehicles. To address these issues, this paper proposes the integration of a trusted registry into the authentication process, enabling mutual authentication between vehicles and roadside units. Consequently, an identity authentication and key agreement protocol for VANETs should satisfy the following security requirements:

1. Ensuring message authentication and integrity. When receiving a message, both the vehicle and the RSU must authenticate the identity of the sender and verify whether the received message is indeed from the original sender;
2. Session key agreement. The vehicle and RSU can establish a secure session key, which is then utilized to encrypt subsequent sessions in order to ensure the confidentiality of the sessions;
3. Privacy protection. In light of the growing emphasis on privacy protection, VANETs must incorporate robust measures to safeguard sensitive vehicle information, including identity, session records, and driving location. The protocol should ensure that no privacy information can be extracted by potential attackers;
4. Can effectively withstand various types of security attacks. The security of VANETs is susceptible to a range of attacks, including impersonation and replay attacks. Therefore, it is imperative for the authentication scheme to possess robust resistance against diverse security threats in order to ensure its reliability and effectiveness.

3. Our Proposed Scheme

The present paper proposes an elliptic curve cryptography-based multiserver authentication and key agreement protocol for the VANETs. This protocol adopts a third-party authentication mode to address the issues of high system overhead and vulnerability against impersonation attacks found in existing protocols without introducing additional storage and computational complexity to the vehicle terminal. In this protocol, both the vehicle and roadside unit possess distinct keys, thereby increasing the registration process for RSU. The protocol comprises five phases: roadside unit registration phase, vehicle registration phase, vehicle login phase, authentication and key agreement phase, and password upgrade phase. Figure 3 illustrates the system process framework of this scheme.

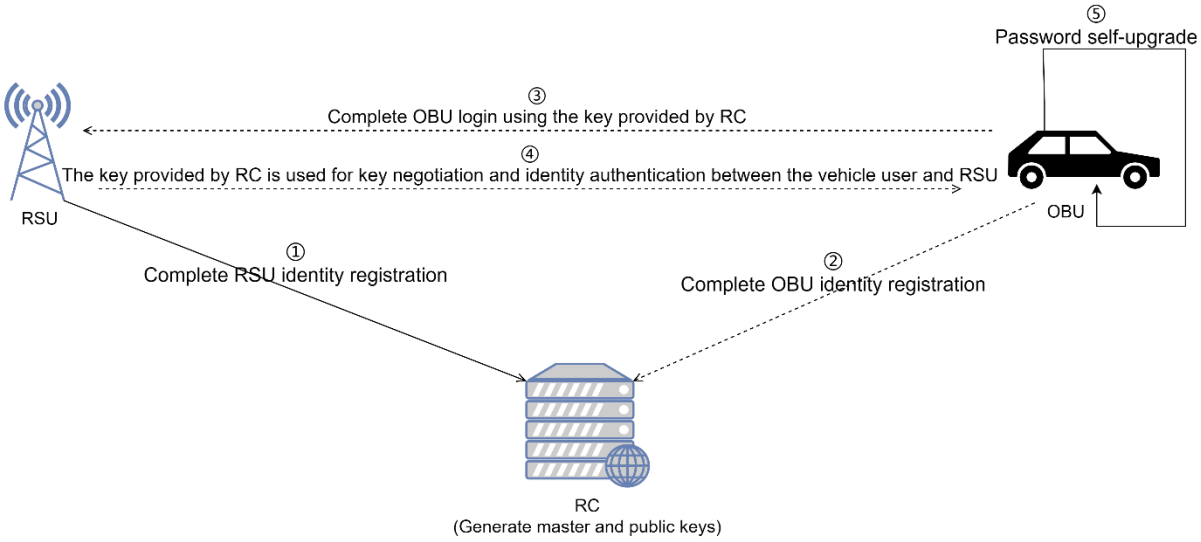

Figure 3. System flow framework of the scheme.

3.1. System Initialization Phase

The registry RC initially generates a large prime number p and constructs an elliptic curve with desirable properties, carefully selecting a generator P of order p. Subsequently, the registry RC proceeds to choose a low entropy random number $sr \in Z_p$ as the system's master key, generating the parameter $Q = sr \times P(mod\ p)$, and publicly disclosing the public parameter $\{Q, P\}$. The identifiers utilized are presented in Table 1.

Table 1. Common identifiers.

Identification	Instructions
OBU_i	A vehicle unit
RSU_i	Some roadside unit
RC	Registry
ID_i	User identity
SID_j	Server identification
PW_i	User password
x	The master key of the system
SK	Session key
$E_k(M)$	The message M is encrypted with key K
$D_k(C)$	Indicates that key K decrypts ciphertext C
$\|\|$	String concatenation symbol
$\oplus$	Xor operation

3.2. Roadside Unit Registration Phase

The roadside unit RSU_i is required to register at the registration center RC to provide network resources to vehicle users through the registration center RC. The proposed multiserver identity authentication and key agreement protocol, based on three-party authentication, offers flexibility and scalability. It allows any RSU to register at any time, eliminating the need for an application process during the initialization phase of the authentication system. Initially, RSU_i selects its own identity SID_j and formulates a plaintext registration request that is transmitted through the authentication channel to the registration center RC. Upon receiving this request, registry RC performs further calculations $s_j = H(SID_j \| sr)$, which are then securely transmitted back to RSU_i via a security channel. Upon receipt of s_j, RSU_i stores it successfully completing the registration process. Finally, RSU_i announces its identity SID_j to vehicle users, as depicted in Figure 4 and Algorithm 1.

Algorithm 1 RSU_i registration

Input: SID_j;
Output: s_j;
1: $s_j = H\left(SID_j \| sr\right)$;
2: **return** $\mathrm{RSU_j} \leftarrow (s_j)$;

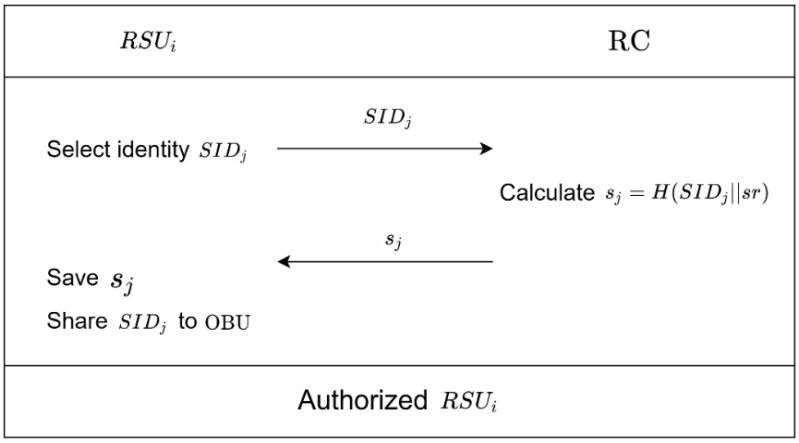

Figure 4. RSU registration steps.

3.3. Vehicle Registration Phase

The vehicle users can register at any time to acquire the corresponding access rights. Initially, the first vehicle user selects OBU_i identity ID_i and PW_i passwords, followed by generating a random number $t \in Z_p$ for calculating the hash value of the implicit vehicle user password $RPW_i = H(PW_i \| t)$. Subsequently, OBU_i sends the registration request and $\{ID_i, RPW_i\}$ to the registration center RC in plaintext form through channel authentication. Upon receiving the vehicle user requests, the registry generates random numbers $R_i \in Z_p$ and calculates $RID_i = H(ID_i \| R_i)$, $K_i = H(RID_i \| sr)$, $A_i = K_i \oplus H(ID_i \| RPW_i)$, $B_i = H(K_i)$. Following this, the registration center RC transmits authentication information $\{A_i, B_i, P, Q, H(\cdot), E_k\}$ to the vehicle terminal. The detailed process of RC authentication for registry is illustrated in Figure 5 and Algorithm 2.

Algorithm 2 OBU_i registration

Input: ID_j, PW_i,SID_j;
Output: $A_i, B_i, P, Q, H(\cdot), E_k, t$;
1: $t \in Z_N^*$; //generate a random number
2: $RPW_i = H(PW_i \| t)$;
3: **if** $(ID_j \in Z_N^*)$ then
4: $R_i \in Z_N^*$;
5: $RID_i = H(ID_i || R_i)$;
6: $K_i = H(RID_i || sr)$;
7: $A_i = K_i \oplus H(ID_i || RPW_i)$;
8: $B_i = H(K_i)$;
9: **else** return null;
4: **return** $OBU_i \leftarrow (A_i, B_i, P, Q, H(\cdot), E_k, t)$;

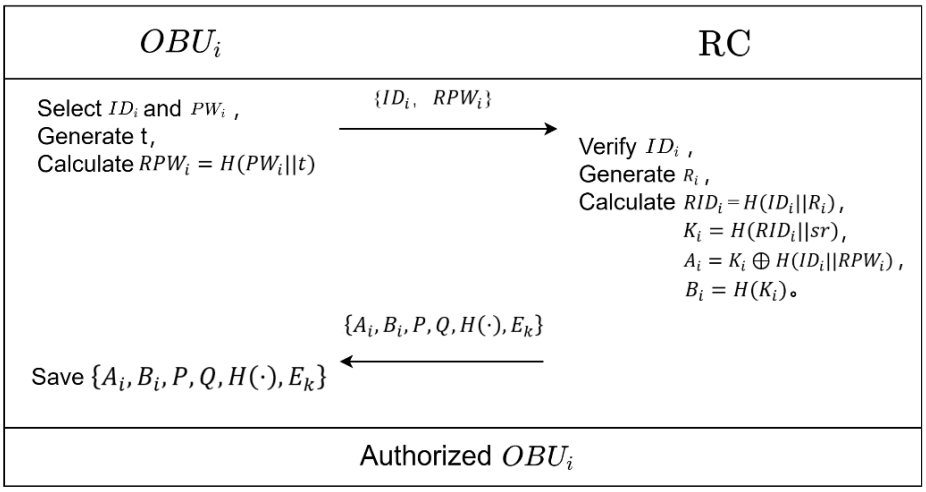

Figure 5. RC certification process.

Finally, the OBU_i of the vehicle user appends a randomly selected number t to the received information $\{A_i, B_i, P, Q, H(\cdot), E_k, t\}$, marking the end of the registration process for the vehicle user.

3.4. Vehicle Login Phase

When the OBU_i of the vehicle user executes the local verification module, it first needs to input its ID_i and corresponding password PW_i. Subsequently, OBU_i computes $K_i = A_i \oplus H(ID_i \| H(PW_i \| t))$, $B_i^* = H(K_i)$, and transmits $\{B_i^*\}$ to RC for validation against the stored check information $B_i^* = B_i$. In case of an invalid equation, RC returns an "error" prompting OBU_i to re-enter incorrect identity and password details. If successful, RC responds with a "pass". Consequently, OBU_i generates a random number $\alpha \in Z_p$ and calculates $X = \alpha \times P$, $X' = \alpha \times Q, CID_i = E_{H(X')}(RID_i, H(K_i \| SID_j))$. Finally, OBU_i sends the login request message $\{CID_i, X\}$ through a public network channel to interface RSU_i where the vehicle user intends to log in.

3.5. Authentication and Key Agreement Phase

After receiving OBU_i's login request, RSU_i initiates a response and completes the identity authentication of OBU_i with the assistance of RC registry. The session key for encrypting data transmission between the vehicle user and RSU_i is generated. The registry cannot recover the negotiated session key. RSU_i randomly selects $\beta \in Z_p$, calculates $Y = \beta \times P$, $M_j = H(X \| Y \| s_j \| SID_j)$, $V_j = E_{sj}(M_j, CID_i, X, Y)$, and then sends $\{SID_j, V_j\}$ to the RC registry. Upon receiving the message from RSU, RC computes $s_j = H(SID_j \| sr)$ and decrypts V_j using s_j to obtain $\{M_j, CID_i, X, Y\}$. Next, RC computes $M_j' = H\left(X \| Y \| s_j \middle| | SID_j\right)$

and compares it with M_j. If they match, it confirms that RSU_i is a legitimate roadside unit and proceeds to the next step; otherwise, RC rejects further operations and returns an error. Subsequently, RC continues by computing $H(sr \times X)$ to extract CID_i as a parameter in $\{RID_i, H(K_i \| SID_j)\}$. Then it calculates $K_i = H(RID_i || sr), H(K_i \| SID_j)$, and verifies whether $H(K_i \| SID_j)$ exists in the decamping parameters. If verification fails, the registry terminates the session with "Vehicle Verification Failed". Otherwise, the registry successfully completes identity verification of both roadside unit RSU_i and vehicle unit RSU_i, and proceeds to subsequent steps. RC computing $Y' = sr \times Y$, $SK_{ij} = H(RID_i \| SID_j \| X' \| Y)$, $C_i = E_{K_i}(SK_{ij}, X, Y)$ and $D_j = E_{s_j}(Y', C_i, SK_{ij})$. Upon completion of the calculation, RC transmits $\{D_j\}$ to RSU_i. After receiving the information $\{D_j\}$, RSU employs its own key s_j to decrypt D_j and obtain parameters $\{Y', C_i, SK_{ij}\}$. It then verifies $Y'? = \beta \times Q$. If they match, RSU_i considers OBU_i as trusted and sends the unsealed parameter $\{C_i\}$ to OBU_i. Subsequently, OBU_i decrypts C_i using its own key K_i to acquire parameters $\{SK_{ij}, X, Y\}$. Next, OBU_i verifies $H(RID_i \| SID_j \| X' || Y)? = SK_{ij}$. If it holds true, OBU_i deems RSU_i as trusted and computes $F_i = H(SID_j \| SK_{ij} \| X \| Y)$, which is sent as the authentication message for the session key to RSU_i. Otherwise, the response message from i is deemed invalid by OBU_i, and thus ends the session with a return of "Response message is not valid". Finally, based on existing data, RSU_i computes $F_i^* = H(SID_j \| SK_{ij} \| X \| Y)$ and verifies $F_i^*? = F_i$. if they are equal, RSU_i and OBU_i negotiate the session key $SK = \alpha \times \beta \times P$. The specific process of authentication and key negotiation can be seen in Figure 6 and Algorithm 3.

Algorithm 3 RSU_i authenticates OBU_i

Input: ID_i, PW_i,A_i, B_i, P, Q, $H(\cdot)$, E_k, t, CID_i, X;
Output: SK;
1: $\beta \in Z_p$; //generate a random number
2: $Y = \beta \times P$;
3: $M_j = H(X \| Y \| s_j \| SID_j)$;
4: $V_j = E_{sj}(M_j, CID_i, X, Y)$;
5: $s_j = H(SID_j || sr)$;
6: $M_j' = H(X \| Y \| s_j || SID_j)$;
7: **if** $(M_j' == M_j)$ then
8: $K_i' = H(RID_i || sr)$;
9: **if** $(H(K_i') == H(K_i \| SID_j)$ then
10: $Y' = sr \times Y$;
11: $SK_{ij} = H(RID_i \| SID_j \| X' \| Y)$;
12: $C_i = E_{K_i}(SK_{ij}, X, Y)$;
13: $D_j = E_{s_j}(Y', C_i, SK_{ij})$;
14: **if** $(Y' == Y)$ then
15: $SK_{ij}' = H(RID_i \| SID_j \| X' || Y)$;
16: **if** $(SK_{ij}' == SK_{ij})$ then
17: $F_i = H(SID_j \| SK_{ij} \| X || Y)$;
18: $F_i^* = H(SID_j \| SK_{ij} \| X || Y)$;
19: **if** $(F_i == F_i^*)$ then
20: return $SK = \alpha \times \beta \times P$;
21: **else** return null;
22: **else** return null;
23: **else** return null;
24: **else** return null;
25: **else** return null;

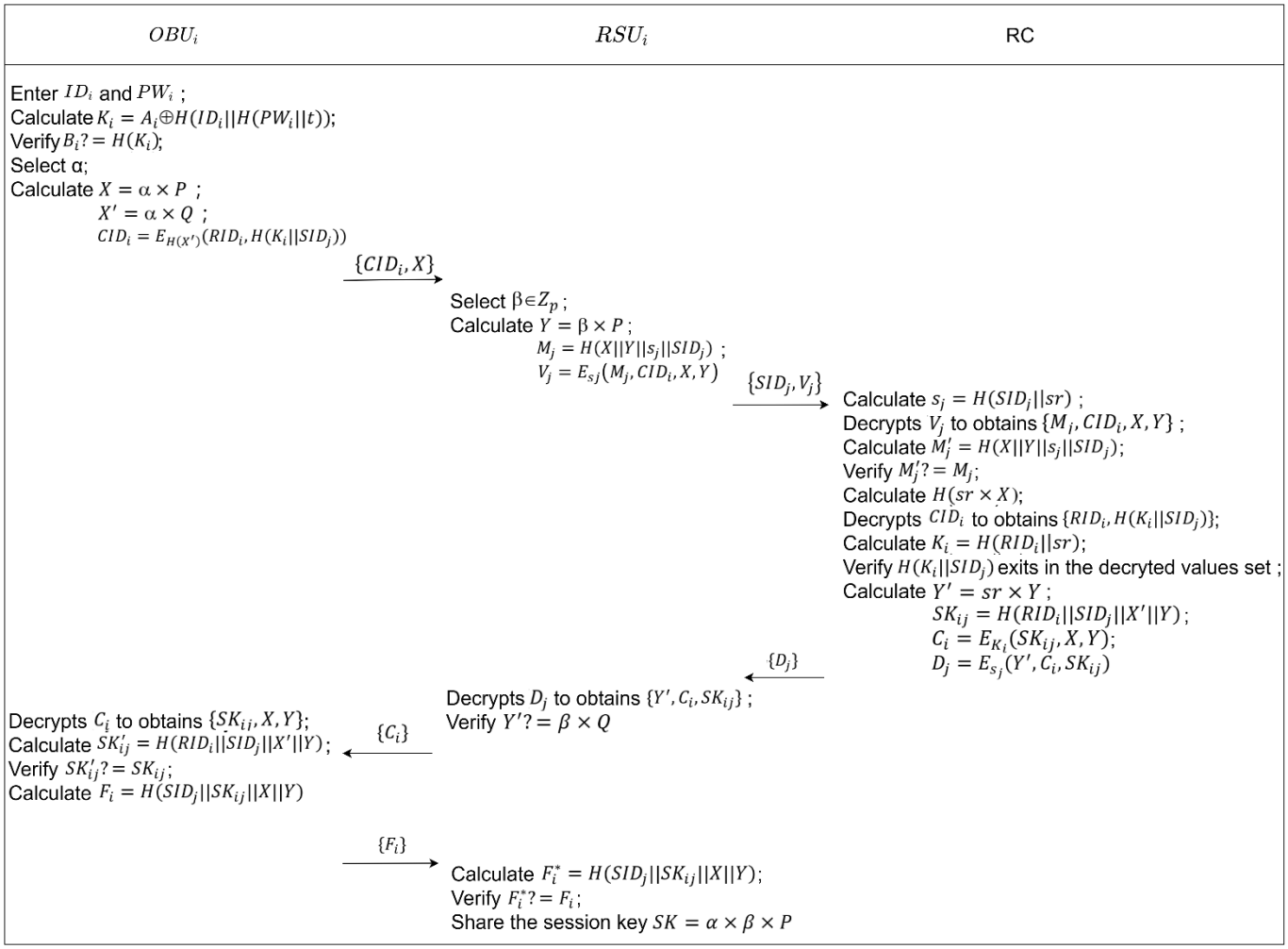

Figure 6. Authentication negotiation process between RSU and vehicle users.

3.6. Password Upgrade Phase

The registration center will implement a password security mechanism for the vehicle, requiring timely password updates within a specified period to mitigate security risks associated with password loss. Vehicle users can independently update their passwords on OBU_i without the need to communicate with RSU or RC. To initiate the password upgrade module, the vehicle user enters their ID ID_i and original password PW_i into OBU_i as prompted by the intelligent terminal. OBU calculates $K_i = A_i \oplus H(ID_i \| H(PW_i||r))$ and verifies $B_i? = H(K_i)$. In case of inconsistency, an incorrect ID or password is indicated, prompting re-entry. If three consecutive incorrect attempts are made, OBU activates the security interrupt mechanism and restricts further attempts to upgrade the password within a designated security period. Conversely, if consistency is confirmed, the vehicle user is prompted to enter a new password twice consecutively; this new password must differ from the original one. In case of inconsistency between entered passwords, OBU prompts re-entry twice; otherwise, OBU calculates $A_i^{new} = K_i \oplus H(ID_i \| H(PW_i^{new}||r))$, replacing A_i with $A_i = A_i^{new}$, thereby completing the password upgrade operation. The password upgrade process is shown in Figure 7.

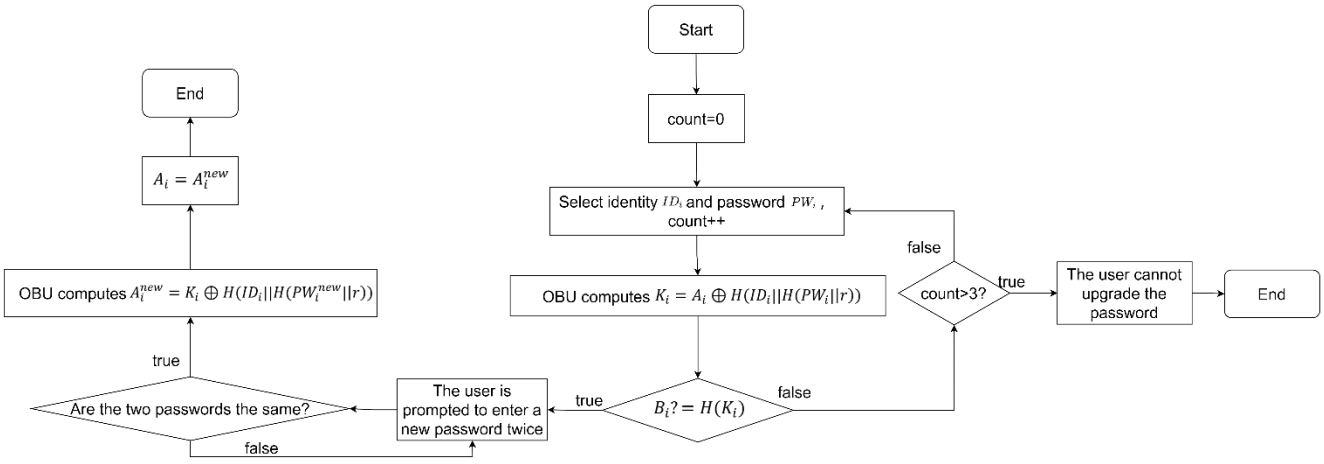

Figure 7. Password upgrade flowchart.

4. Security Analysis

4.1. Formal Security Analysis

The stochastic oracular model, proposed by Bellare and Rogaway [28] in 1993, offers a method for designing protocols that are highly effective. It can be considered the ideal state of the hash function, known as the random oracle model, which possesses three fundamental properties.

1. Consistency: If the input value is the same, the output must be the same;
2. Computability: The output will be calculated in polynomial time;
3. Uniform distribution: The output value space range is evenly distributed. The biggest advantage of this model is that it applies provable security methodology to practical applications.

The steps to verify the security of a scheme using a random oracle model are:

1. The security of the scheme is formally defined, assuming that an adversary can successfully steal the key with a nonnegligible probability in a probability polynomial of time;
2. An adversarial algorithm C is established to provide a simulation environment for the adversarial algorithm C. The adversarial algorithm C uses a unique attack method to ask the protocol, and the protocol answers all the questions asked by the opponent;
3. Try to solve mathematical puzzles with information obtained from your opponent. If the mathematical problem can be solved successfully, it proves that the protocol does not pass the random oracle model verification. If the mathematical problem is not solved successfully, it is proved that the protocol is successfully verified by the random oracle model.

We will choose the random oracle model as the simulation environment and the CK adversarial model as the adversarial algorithm to analyze the security of the whole scheme.

Theorem: Let A be an adversary against a probabilistic polynomial protocol. Adversary A can compromise the semantic security of the protocol through Send queries, Execute queries, and Hash queries. The probability of opponent A's success is shown in Formula (2), where N represents a set of integers, l denotes the size of the hash function space, q_h indicates the number of hash queries executed by adversary A, and q_s represents the number of Send queries executed by adversary A.

$$adv^{AKA}(A) \leq \frac{q_s^2}{N} + \frac{q_h^2 + q_s + 2q_h}{2^l} + 2q_h max\left\{adv_{CMBDLP}^{AKA}(A)\right\} \quad (2)$$

Proof. Define a sequence of events, denoted as G_0 through G_5, where the event G_0 represents a genuine attack with opponent A having no advantage. In event G_1, opponent A is granted the ability to passively attack and launches passive attacks on both sides. Building upon event G_1, event G_2 grants opponent A the capability to send queries and initiates active attacks on both sides. Event G_3 enables adversary A to send a Hash query based on event G_2, which is then utilized by adversary A to make guesses about the values of W_3 and W_4. Extending from event G_3, event G_4 empowers adversary A with ESReveal queries and Corrupt queries. Finally, in event G_5, derived from event G_4, adversary A gains the ability to send a Test query. Subsequently, adversary A is presented with a mathematical problem related to the CMBDLP hypothesis and the CMBDHP hypothesis. The progression of these events is illustrated in Figure 8. □

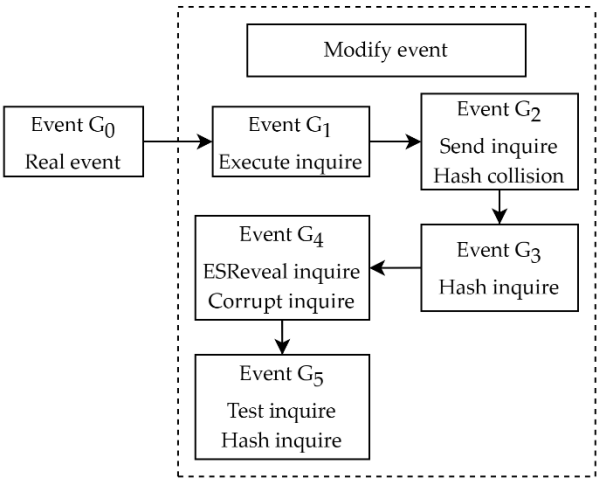

Figure 8. Flow of events.

Theorem (differential theorem): R1, R2, R3 denote events in a certain probability distribution. If the equation R1 ∧ ¬R2 ⇔ R3 ∧ ¬R2 exists. Then $|\Pr[R1] - \Pr[R3]| \leq \Pr[R2]$

The following is a detailed description of the G_0 to G_5 events:

Event G_0: This event is A simulation of a real attack by adversary A in a random language model. According to the above definition, Formula (3) is derived:

$$\Pr[Succ_1] = \Pr[Succ_0] \tag{3}$$

Event G_1: In this event, opponent A initiates a passive attack by using the Execute (OBU_i, RSU_i) function to query the interaction information $\{CID_i, X, SID_j, V_j, D_j, C_j, F_j\}$ from RC. However, opponent A is unable to obtain the parameter s_j that RC sends to RSU_i during the registration phase. Consequently, adversary A cannot derive the session key solely based on this information, Formula (4) is derived:

$$adv^{AKA}(A) = |2Pr[Succ_0] - 1| \tag{4}$$

Event G_2: First event G_2 simulates event G_1 and adversary A has the ability to Send an active attack using the SEND query. When peer A sends different Send queries, peer A receives different interaction messages. There are several cases:

For a Send$(RSU_i, \{CID_i, X\})$ query: when RSU_i received $\{CID_i, X\}$, RSU_i will select a random number beta and calculate the Y, M_j, V_j. Finally, $\{Y, M_j, V_j\}$ to opponent A.

For a Send $(RC, \{SID_j, V_j\})$ query: RC calculates $s_j = H(SID_j \| sr)$, using s_j unlock $\{M_j, CID_i, X, Y\}$, and calculate the $M' = H(X \| Y \| sj \| SID_j)$, and then judge $M'? = M_j$. Next, calculate H(sr × X), The parameter $\{RID, H(K_i \| SID_j)\}$ is obtained by unlocking CID_i, and calculate the $H(RID_i \| sr), M'' = H(K_i \| SID_j)$, and then judge $M''? = M'$. If is

equal to the calculated by $Y' = \mathrm{sr} \times Y, SK_{ij} = \mathrm{H}(RID_i \| SID_j \| X' \| \mathrm{Y})$,$\mathrm{C}_i = \mathrm{E}_{K_i}(SK_{ij}, X, Y)$, $\mathrm{D}_j = \mathrm{E}_{s_i}(Y', C_i, SK_{ij})$. Finally, RC sends $\{\mathrm{D}_j\}$ to opponent A.

For a Send(RSU_i,$\{\mathrm{D}_i\}$) query: RSU_i decrypts D_i with s_j to obtain parameters $\{SK_{ij}, C_i, Y'\}$, and judge $Y'? = Q \times \beta$. If so, unseal C_i. Finally, RSU_i sends $\{C_i\}$ to opponent A.

For a Send(OBU_i,$\{\mathrm{C}_i\}$) query: OBU_i with K_i decryption C_i get parameters $\{SK_{ij}, X, Y\}$, calculate $\mathrm{SK}'_{ij} = \mathrm{H}(RID_i \| SID_j \| X' \| \mathrm{Y})$, then judge $\mathrm{SK}'_{ij}? = SK_{ij}$. If this is true, then $F_i = H(SID_j \| SK_{ij} \| X \| Y)$ is calculated. Finally, OBU_i sends $\{F_i\}$ to opponent A.

For a Send(RSU_i,$\{F_i\}$) query: RSU_i computing $\mathrm{F}^*_i = \mathrm{H}(SID_j \| SK_{ij} \| \mathrm{X} \| \mathrm{Y})$, the final inspection $\mathrm{F}^*_i? = F_i$.

In this scheme, the random numbers α and β come from the set of positive integers Z^*. According to the birthday paradox theorem, the probability of a collision between two random numbers after sending a $\mathrm{q_s}$ query is $\mathrm{q_s^2}/2\mathrm{N}$. If a hash query collision occurs, the event ends immediately, and the probability of a hash collision occurring is $\mathrm{q_h^2}/2^{\mathrm{l}+1}$, Formula (5) is derived: (the length of each hash value is l)

$$| \, Pr[Succ_2 - Succ_1] \, | \leq q_s^2/2N + q_h^2/2^{l+1} \tag{5}$$

Event G_3: If opponent A cannot correctly guess the values of X and Y without using the hash oracle query scenario. Then, the event G_3 is the same as the previous event, and Formula (6) is derived:

$$| \, Pr[Succ_3 - Succ_2] \, | \leq q_s/2^{l+1} \tag{6}$$

Event G_4: This event contains concerns about session key security. In addition to inheriting the ability of event G_3, adversary A in event G_4 can also obtain $\{SID_j, ID_i, s_j, K_i, \alpha, \beta\}$ and other information through ESReveal query and Corrupt query.

Case 1: Send ESReveal(OBU_i) and ESReveal(RSU_i) queries: When opponent A sends ESReveal(OBU_i) and ESReveal(RSU_i) queries, opponent A will get the information $\{\alpha\}$ in OBU_i and $\{\beta\}$ in RSU_i. But this information is different in every conversation.

Case 2: Send ESReveal(OBU_i) and Corrupt(RSU_i) query: When opponent A sends ESReveal(OBU_i) and Corrupt(RSU_i) queries, opponent A will get $\{\alpha\}$ in (OBU_i) and $\{s_j, SID_j\}$ in (RSU_i).

Case 3: Send Corrupt(OBU_i) and ESReveal(RSU_i) query: When opponent A sends Corrupt(OBU_i) and ESReveal(RSU_i) queries. Opponent A will have access to the information $\{K_i, ID_i\}$ in OBU_i and $\{\beta\}$ in RSU_i.

Case 4: Sending Corrupt(OBU_i) and Corrupt(RSU_i) queries: When adversary A sends Corrupt(OBU_i) and Corrupt(RSU_i) queries, adversary A will get the information $\{K_i, ID_i\}$ in OBU_i and $\{s_j, SID_j\}$ in RSU_i.

But since the session key is associated with X', Y. When opponent A does not get $\mathrm{H}(RID_i \| SID_j \| X' \| \mathrm{Y})$ or solve CMBDLP and CMBDHP assumptions, opponents will not be able to get A session key. That is, event G_4 is the same as event G_3 as long as the assumptions of CMBDLP and CMBDHP hold. Then we get:

$$\left| Pr[Succ_4 - Succ_3] \right| \leq q_h max\left\{ adv^{AKA}_{CMBDLP}(A), adv^{AKA}_{CMBDHP}(A) \right\}$$

Event G_5: Event G_5 simulates event G_4. The opponent A issues a Test query, if your opponent A $\mathrm{H}(RID_i \| SID_j \| X' \| \mathrm{Y})$ query. Then, the Test query for event G_5 terminates. Since the adversary A can obtain SK_{ij} by hash query. The probability that SK_{ij} is obtained by hash query is $q_h/2^l$. Then, Formula (7) is derived:

$$|Pr[Succ_5 - Succ_4]| \leq \frac{q_h}{2^l} \tag{7}$$

The adversary A issues a Test query and makes a guess about the value of b. Then, we get the probability that adversary A guessed correctly, Formula (8) is derived:

$$Pr[Succ_5] = 1/2 \tag{8}$$

Combining the above formula, Formula (9) is derived:

$$adv^{AKA}(A) \leq \frac{q_s^2}{N} + \frac{q_h^2 + q_s + 2q_h}{2^l} + 2q_h max\left\{adv_{CMBDLP}^{AKA}(A), adv_{CMBDHP}^{AKA}(A)\right\} \quad (9)$$

Then the formula follows: there exists a number ε, $\varepsilon > 0$, such that the inequality $\mathrm{adv}^{\mathrm{AKA}}(\mathrm{A}) < \varepsilon$ holds. That is, in the CK model, the protocol scheme is considered secure.

4.2. Informal Security Analysis

OBU impersonation attack: OBU impersonation attack means that an attacker masquerades as a legitimate vehicle user to obtain information services illegally. First of all, the attacker can forge a login request $\{CID_i, X\}$, including $CID_i = E_{H(X')}\left(RID_i, H\left(K_i \parallel SID_j\right)\right) = E_{H(\alpha \times Q)}\left(RID_i, H\left(K_i \parallel SID_j\right)\right)$ and be RSU and RC validation through. Among them, RID_i and K_i are the key information, but from the above introduction of this protocol, we know that they are secure, and the corresponding key K_i is also secure. Therefore, vehicle impersonation attack is not possible in the proposed protocol.

RSU impersonation attack and RC impersonation attack: In order to launch an RSU impersonation attack, the attacker must forge a valid authentication reply information $\{C_i\}$ and successfully pass the verification process of vehicle OBU_i. The keys used by each vehicle and RSU are unique and independently generated by the registration center, with no correlation between them. Without obtaining the key of the target RSU or the key of the vehicle communicating with it, the attacker will be unable to recover or calculate $\{C_i\}$ in D_j, thus making it impossible to forge a legitimate reply $\{C_i\}$ that can be authenticated by vehicle OBU_i. Consequently, our proposed protocol effectively mitigates against RSU impersonation attacks. Furthermore, an RC impersonation attack can only be launched if the attacker gains access to the system master key sr. However, this master key is exclusively stored and accessible by the registry center itself. The security of this master key directly determines overall security within our multiserver authentication system framework. It is assumed in this paper that proper measures are taken by the registry center to safeguard this system master key at all times; hence rendering any effective RC masquerading attacks unfeasible.

User privacy protection function: This scheme proposes the random generation of RID_i to replace the actual vehicle information identification in the login request information. The dynamic parameter is encrypted using the key $H(\alpha \times Q)$. Only a trusted third-party registry RC can compute $X' = sc \times X$ using the system master key. By utilizing parameters obtained from this calculation, $CID_i = E_{H(X')}\left(RID_i, H\left(K_i \| SID_j\right)\right) = E_{H(\alpha \times Q)}\left(RID_i, H\left(K_i \| SID_j\right)\right)$ recovery RID_i can be unlocked. However, there are only two ways for an attacker to recover the vehicle user key: illegal theft of the system master key or solving an NP problem. It is evident that successful retrieval by attackers is not possible; hence, ensuring anonymity in this proposed protocol. Furthermore, none of the fixed parameters are present in the vehicle login request information $\{CID_i, X\}$, as they are all randomized by parameter α. Therefore, tracing fixed parameters within login information cannot reveal any vehicle-related details. Consequently, this proposed scheme ensures nontraceability.

Offline password guessing attack: In this study, it is assumed that the attacker has obtained $\{A_i, B_i, r\}$ of the vehicle through unauthorized means, where $B_i = H(A_i \oplus H(ID_i \parallel H(PW_i \parallel r)))$. To recover the password in B_i, the attacker must acquire both the vehicle's ID and password. However, during registration services provided by the registration center (RC), vehicles are required to adhere to specific rules for selecting their identity and password. This inevitably enhances the difficulty for attackers attempting to crack these two crucial parameters. Consequently, our proposed scheme effectively withstands offline password-guessing attacks launched by potential adversaries.

Replay attack: A replay attack occurs when an attacker records or repeats information during network communication with the intention of deceiving the system. In our proposed scheme, we have integrated random numbers and timestamps into the interactive

information exchanged between the on-board unit OBU_i, roadside unit RSU_i, and registration center (RC). This integration enables any recipient to determine whether a particular piece of information has been replayed or not, thereby effectively filtering out any illicitly replayed data.

Forward security: Forward security means that the leakage of a long-used master key does not lead to the leakage of past session keys. In the proposed scheme, the session key $SK = \alpha \times \beta \times P$ negotiated by the on-board unit OBU_i and the roadside unit RSU_i integrates the random numbers $\{\alpha, \beta\}$ generated by them respectively. These two random numbers are generated independently by the on-board unit OBU_i and the roadside unit RSU_i, which cannot be obtained by anyone else. The attacker can only calculate SK by $X = \alpha \times \mathrm{P}, Y = \beta \times P$. In other words, the session key security of the proposed scheme can be reduced to the computational DH problem, so it satisfies forward security.

Interoperability: The protocol meticulously adheres to and integrates VANET standards, such as IEEE 802.11p [29] and SAE J2735 [30], ensuring coherence with industry benchmarks across vital elements like communication protocols, message formats, and data fields. This adherence to standards aims to facilitate the seamless integration of protocols into the existing VANET ecosystem, mitigate compatibility issues among diverse protocols, and enhance the overall system's stability and maintainability. Simultaneously, to accommodate the resource constraints of the VANET environment, we opted for ECC with a focus on a concise key length. The rationale behind this choice lies in the ability of a short key length to alleviate computational burdens during communication, rendering the protocol more suitable for the resource-limited vehicle communication milieu in VANET while also maintaining compatibility with traditional encryption algorithms.

5. Performance Analysis

The proposed scheme will be compared with the schemes of Shohei et al. [11], Zhang et al. [13], Liu et al. [19], Guo et al. [25], Kumari et al. [26], Jangirala et al. [27], and Chen et al. [23] in this section to analyze and compare their security functions. Furthermore, a comparison will be made between our scheme and the schemes of Zhou et al. [17], Liu et al. [19], and Chen et al. [23] in terms of encryption operations, computation delay, and communication cost.

5.1. Safety Function Analysis

According to the comparison in Table 2, it is evident that the protocols proposed by Zhang et al. [13] and Liu et al. [19] lack effective resistance against known session key attacks and password-guessing attacks. Guo et al.'s scheme [25] fails to withstand offline password guessing attacks and replay attacks. On the other hand, Kumari et al.'s protocol [26] and Jangirala et al.'s protocol [27] are susceptible to impersonation attacks from OBU and RSU, while also failing to ensure vehicle privacy. In recent years, although the new protocols proposed by Shohei et al. [11] and Chen et al. [23] have enhanced identity authentication efficiency, they still possess certain flaws, such as susceptibility, to replay attacks and absence of vehicle privacy protection. Distinguished from other solutions, our proposed protocol comprehensively fulfills all security requirements listed in the table while providing a more comprehensive and efficient solution for addressing limitations present in existing approaches.

Table 2. Comparison of security functions.

	OBU and RSU Feinting	User Privacy Protection	Resist Known Session Key Attacks	Resist Password Guessing Attacks	Replay Attack	Forward Safety
[11]	√	√	×	×	√	√
[13]	√	×	√	√	×	√
[19]	√	√	×	×	√	√
[25]	√	√	√	×	×	√
[26]	×	×	√	√	√	√
[27]	×	×	√	√	√	√
[23]	√	×	√	√	×	√
Ours	√	√	√	√	√	√

5.2. Computational Cost

The encryption operations employed in the four compared schemes include hash operation, scalar multiplication operation, point addition operation, symmetric key encryption operation, and symmetric key decryption operation. Their corresponding symbols are Has, Mul, Add, Enc and Dec. To facilitate a more intuitive comparison of performance data between the schemes, we conducted simulations using OpenSSL library [31], GMP library [32], and PBC Library [33] on two Ubuntu 16.04 devices. The end device hardware utilized was a RASPBERRY PI 3B+ with 1 GB LPDDR2 SDRAM and a BCM2837B0 system chip operating at a frequency of 1.4 GHz. The AG hardware consisted of a computer with 4 GB RAM and an INTEL(R) CELERON (R) J1900 CPU. The execution time of the basic operation is shown in Table 3.

Table 3. Execution time of basic operations (ms).

	Definition	BCM2837B0	Intel J1900
T_{Has}	Execution time of the SHA 256 algorithm (160 bit)	0.0729 ms	0.0023 ms
T_{Mul}	Curve25519 Execution time of point multiplication	23.4405 ms	2.2260 ms
T_{Add}	Curve25519 Execution time of point addition	0.1652 ms	0.0288 ms
T_{Enc}	Execution time for AES 128 encryption (ECB mode and 128 bit)	0.0500 ms	0.0130 ms
T_{Dec}	Execution time of AES 128 decryption (ECB mode and 128 bit)	0.0810 ms	0.0170 ms
T_{Xor}	Curve25519 Execution time of the XOR operation	0.1650 ms	0.0490 ms
T_{EXP}	Curve25519 Indicates the execution time of the exponential operation	3.3280 ms	0.0390 ms

To evaluate the performance of the schemes in terms of encryption operations, we conducted a comprehensive analysis of the utilization of different encryption techniques in each scheme, as presented in Table 4. In terms of encryption operations, it is observed that the remaining three schemes exhibit higher usage rates, potentially leading to increased computational latency.

Table 4. Comparison of the number of different encryption operations.

	Our		Zhou et al. [17]		Liu et al. [19]		Chen et al. [23]	
	User	Server	User	Server	User	Server	User	Server
Has	5	8	2	2	0	2	1	4
Mul	2	5	3	5	4	3	5	4
Add	0	0	2	3	1	0	1	1
Enc	0	3	0	0	0	0	0	0
Dec	1	3	0	0	0	0	0	0
Xor	1	0	0	0	0	0	0	0
EXP	0	0	0	0	6	4	1	2
Total	58.8955 ms		82.0601 ms		120.7338 ms		129.7886 ms	

Based on the data in Tables 3 and 4, we computed the duration of cryptographic operations for the compared schemes. Among them, Zhou et al. [17] performed 17 cryptographic operations, taking 82.0601 ms. Lin et al. [19] conducted 20 cryptographic operations and required 120.7338 ms. Chen et al. [23] executed 19 cryptographic operations, which consumed 129.7886 ms of time. The proposed scheme involved a total of 28 cryptographic operations with a corresponding time expenditure of 58.8955 ms.

Through simulation, we obtained the results as shown in Figure 9, which shows the comparison of different schemes of average computing latency between the client side and the server side under the same attack ratio. The vertical axis represents the average computation delay, while the horizontal axis represents the four different schemes, with the blue and orange columns representing the client and server sides, respectively. According to the data in Table 4, the scheme proposed in this paper mainly completes the encryption operation on the server side. It can be clearly seen from Figure 9 that compared with other schemes, the scheme proposed in this paper has relatively short computation latency on both the client and server side.

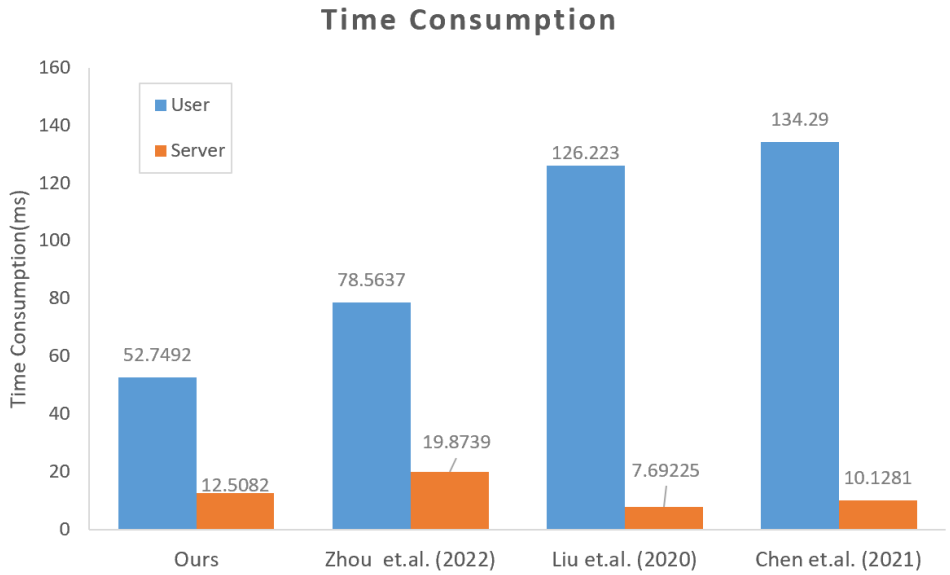

Figure 9. Comparison of computation latency between user and server in different schemes [17,19,23].

The security properties of the proposed scheme have been analyzed in Section 4, and it has been proven that the scheme can withstand certain attacks. Attacks that can be defended through security analysis are considered as known attacks, while all other potential attacks are classified unknown attacks due to their unpredictable nature. It is assumed that an unknown attack may disrupt the authentication process of these four schemes, whereas known attacks cannot. Under known attacks, each scheme exhibits a fixed computation delay; however, under unknown attacks, the computation delay may vary. To assess the impact of this uncertainty on performance, we conducted simulations using C++ to evaluate how the authentication schemes perform under unknown attacks. During these simulations, we measured the computation delay for each scheme as we varied the ratio between unknown and known attacks. The parameter used to measure performance is defined as the average successful computation delay according to Equation (10).

$$\text{delay}_{\text{AVG}} = \frac{\text{delay}_{\text{unknown}} * \text{times}_{\text{unknown}} + \text{delay}_{\text{known}} * \text{times}_{\text{known}}}{\text{times}_{\text{known}}} \tag{10}$$

where $\text{times}_{\text{unknown}} = \text{ratio}_{\text{unknown}} * \text{times}_{\text{ALL-ATTACK}}$.

The correlation between the average computation delay of each scheme and the percentage of unidentified attacks is depicted in Figure 10, with the Y-axis representing the average computation delay and the X-axis indicating the proportion of unknown attacks

relative to the total number of attacks. Each scheme is represented by a distinctive color curve. As outlined in Table 4, our computations primarily occur on the server side, and as indicated in Table 3, the server side's computational time significantly outperforms that of the user side. Figure 10 illustrates that, with the escalating percentage of unknown attacks, the average computation delay for each scheme gradually increases. Notably, our proposed scheme consistently exhibits the lowest computation latency even when facing a higher proportion of unknown attacks, underscoring its robust security. Hence, our scheme stands out with considerable advantages in this context.

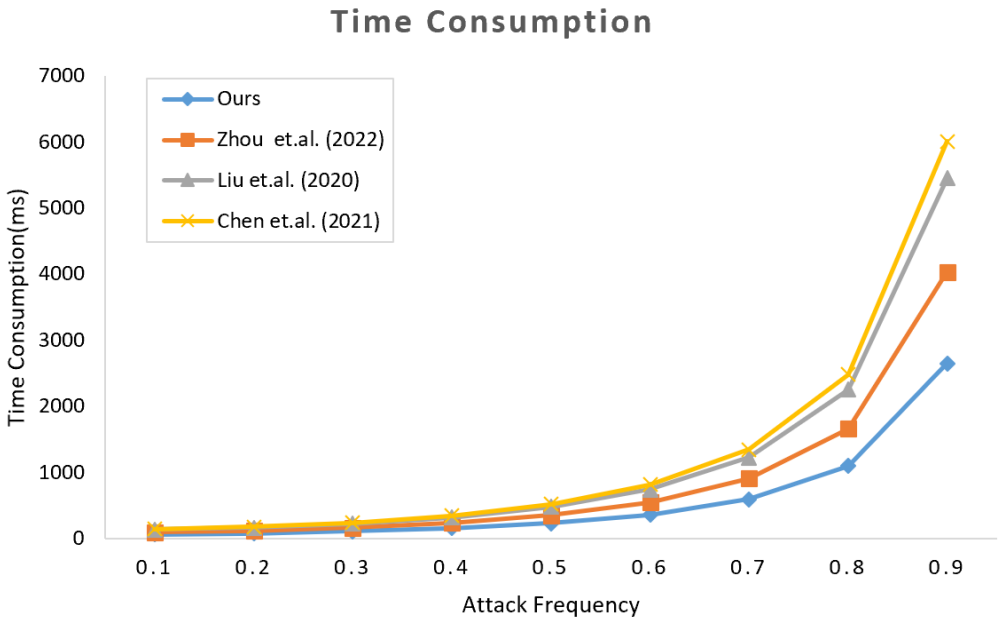

Figure 10. Comparison of computation delay of different schemes under unknown attack ratio [17,19,23].

5.3. Memory Overhead

The authentication process relies heavily on the relationship between memory and performance, as it involves handling a substantial amount of information and keys. The efficient utilization of memory directly impacts the overall performance of the authentication system. Sufficient memory enhances concurrency, enabling simultaneous authentication for multiple users and thereby improving system performance. Conversely, inadequate memory can lead to delays in authentication requests, reduced system responsiveness, or even denial-of-service situations. We will evaluate each scheme's size to assess its pros and cons in terms of performance. If a scheme requires less memory, it consumes fewer system resources for identity authentication and offers certain advantages in terms of performance.

The memory consumption of different operands varies across systems. In this paper, it was assumed that the memory footprint for timestamp S_T random number S_R, and identifier S_I was 32 bits; the memory footprint for data obtained through point multiplication S_M was 320 bits; the memory footprint for hash value S_H and packet S_S was 160 bits; and the memory footprint for symmetric key encryption S_E was 128 bits. Table 5 presents an analysis of the proportion of memory usage in four schemes on both the user side and the server side. Figure 11 illustrates a comparison of the total occupied memory by these four schemes.

Table 5. Memory usage operand analysis.

Scheme	User	Server	Total
Our	$2S_E + S_M + S_H$	$S_I + 3S_E$	$5S_E + S_M + S_H + S_I$
Zhou et al. [17]	$8S_S + S_T + S_H$	$2S_M$	$8S_S + S_T + S_H + 2S_M$
Liu et al. [19]	$3S_E + S_M + 2S_I$	$S_E + S_R + 3S_M + 2S_I$	$4S_E + S_R + 4S_M + 4S_I$
Chen et al. [23]	$7S_S + S_T$	$2S_M$	$7S_S + S_T + 2S_M$

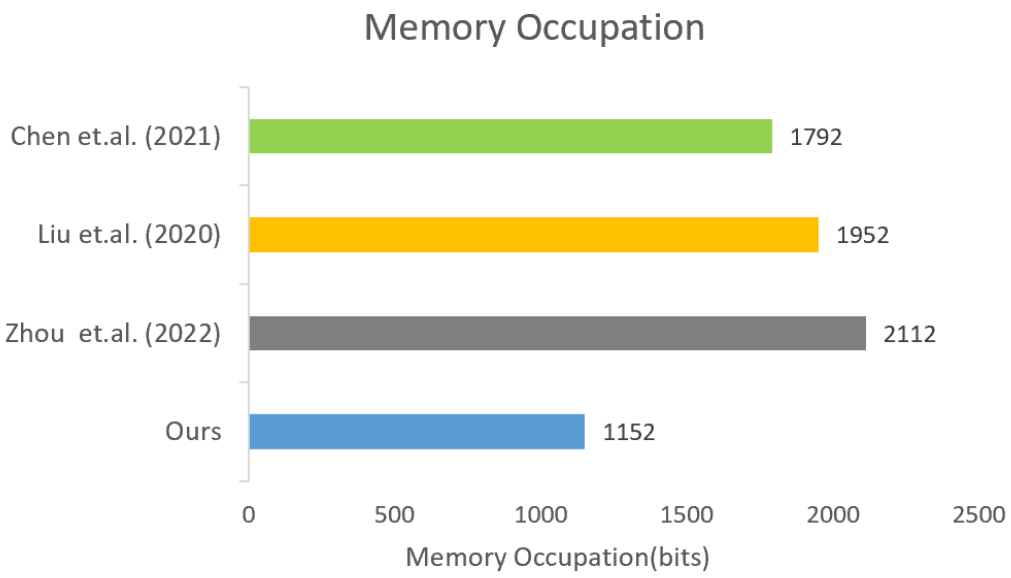

Figure 11. Comparison of memory usage when processing a single message [17,19,23].

The proposed scheme exhibits the minimum total memory occupancy value, as evidenced by Table 5 and Figure 11. Moreover, it demonstrates superior performance in terms of memory occupation on the user side with limited system resources. A smaller memory footprint on the user side ensures smooth operation even with fewer system resources available. In conclusion, our proposed scheme offers significant advantages in terms of memory utilization.

To verify the performance of the scheme in terms of packet loss rate, we set up a simulation scenario using the Vein simulation platform. Reasonable deployment of Rsus, such as key intersections, high-traffic areas, and emergency lanes, is the key to ensuring efficient coverage. The OBU is placed inside the vehicle, paying attention to the antenna position and causing minimal interference to the vehicle. In terms of communication protocols, IEEE 802.11p [29] and SAE J2735 [30] protocols are used to support a variety of communication technologies. The simulation scenario Settings are shown in Table 6: (1) On a two-way four-lane road with a total length of 12 km, an RSU with a communication radius of 800 m is placed every 3 km. (2) Allow different numbers of vehicles to pass through, introducing changes in traffic density. (3) The vehicle speed is controlled within the range of 40~120 km/h to simulate the real traffic conditions. (4) The vehicle communication radius is set to 250 m to establish a local communication network. (5) The broadband of the vehicle is fixed at 200 kbit/s, representing the typical data transmission rate of vehicle communication. The packet loss rate is defined as the ratio of the total number of lost signed packets transmitted by a vehicle to the total number of signed packets.

Table 6. Simulation Scenario.

Parameter	Setting
Simulation Scenario Range	12 km
RSU Communication Radius	800 m
Vehicle Speed	40–120 km/h
Vehicle Communication Radius	250 m
Vehicle Broadband	200 kbit/s

As depicted in Figure 12, the packet loss rate exhibits an upward trend with the increase of traffic volume in a given scenario. This phenomenon is mainly attributed to the elevated vehicle density, which leads to vehicles moving out of communication range before message processing by RSUs is completed. Under high traffic conditions, vehicles may enter and leave the communication range of RSUs more rapidly, resulting in a shorter time window for message processing and a higher likelihood of packet loss.

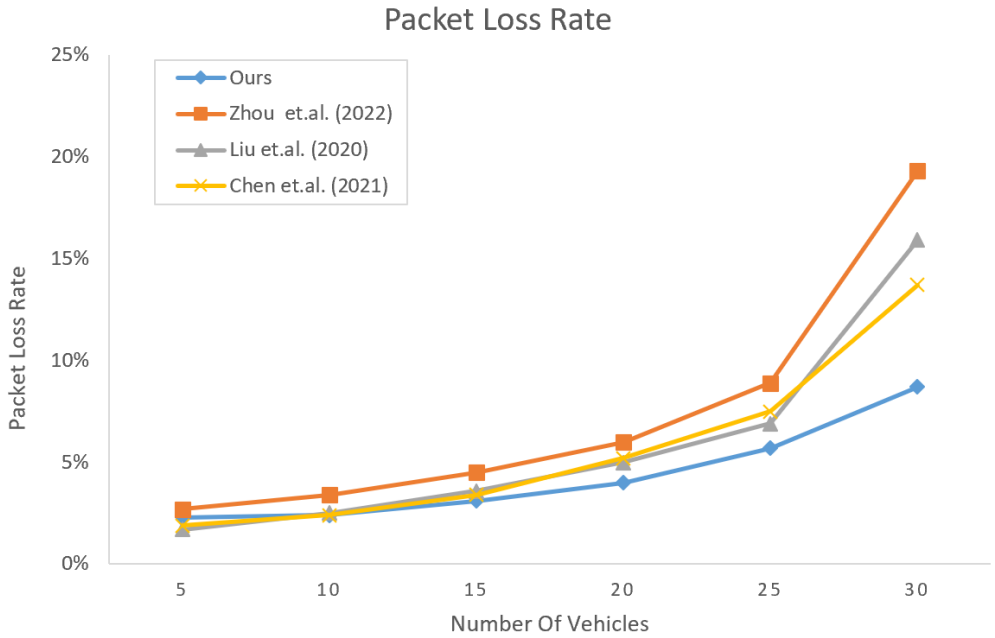

Figure 12. Packet loss rate [17,19,23].

Traditional authentication and key management can become cumbersome and computationally intensive as traffic increases. In our proposed scheme, RC serves as a centralized entity that simplifies the communication process between vehicles and RSUs. By reporting to RC to obtain authentication information and keys, vehicles do not need to communicate directly with multiple RSUs frequently, thereby reducing communication complexity. Meanwhile, RC handles complex computational tasks such as key management and authentication more efficiently using specialized algorithms and resources. The advantages offered by RC alleviate the computational burden on both vehicles and RSUs during authentication and key exchange processes while reducing packet loss probability significantly.

Observations from Figure 12 indicate that when there are few vehicles present, our proposed scheme's packet loss rate is almost identical to other schemes; however, its superiority becomes more pronounced when there are many vehicles present. Our scheme demonstrates strong adaptability when dealing with high traffic flow while providing reliability and stability for communications within vehicular networking environments—further proving its practical applicability.

6. Open Challenges and Future Research Directions

The protocol introduced in this paper presents a significant advancement in the realm of autonomous driving and intelligent transportation. The incorporation of a trusted RC effectively addresses security challenges prevalent in existing protocols, ensuring highly reliable bidirectional authentication between vehicles and RSUs. This not only strengthens the resilience of vehicle communications but also enhances the security posture for autonomous vehicles. Under a multiserver architecture, the protocol achieves substantial progress in identity authentication and key negotiation, eliminating the burdensome and repetitive registration of vehicle users at each RSU. This optimization improves system efficiency, fosters seamless integration of vehicle management, and delivers a more efficient solution for intelligent transportation systems.

Nevertheless, in the current landscape, the security of multiserver authentication and key protocols based on elliptic curve cryptography faces considerable challenges due to the emergence of quantum computing. The vulnerabilities of traditional encryption algorithms become more apparent in a quantum computing environment, particularly the susceptibility of elliptic curve cryptography to factorization and other attacks. This raises critical questions about the overall protocol security, necessitating urgent exploration of future research directions to counter the new challenges posed by quantum computing.

To address this challenge, future research directions should concentrate on implementing cryptographic algorithms resistant to quantum computing attacks, such as hash function-based message authentication code (HMAC) [34] and lattice-based encryption algorithms (NTRUEncrypt) [35]. These cryptographic algorithms demonstrate heightened resistance in the current quantum computing environment and can effectively counter potential quantum computing attacks. Additionally, future research should delve into advanced strategies for key management, incorporating regular key updates to ensure the adaptability of system security to the continuous evolution of cryptographic techniques. Research in this direction will offer profound insights and crucial solutions for addressing the security challenges associated with elliptic curve cryptography in the era of quantum computing.

7. Conclusions

In this paper, we propose a three-party identity authentication and key agreement protocol based on elliptic curve public key cryptography to address the challenges of identity authentication and key agreement between OBU and RSU in VANETs. To mitigate vehicle impersonation attacks, RSU impersonation attacks, and vehicle privacy leakage in multiserver remote authentication within a wireless mobile environment, we introduce a trusted third-party registration center that facilitates identity verification for OBU and RSU. During the login and mutual authentication process, the registration center assumes most of the identity verification tasks without imposing additional storage or computational complexity on the vehicle terminal. This ensures efficient storage utilization and computing efficiency even for vehicles with limited computing power while enabling successful mutual authentication between OBU and RSU. By adopting this scheme, we effectively address security risks present in existing approaches without increasing the storage complexity of the vehicle terminal. Future research will focus on vehicle efficient authentication through RSU, exploring this direction in order to enhance system stability, reduce costs, reduce dependence on third parties, and thus improve the sustainability of the overall solution.

Author Contributions: Methodology, M.M.; Software, W.Y., R.Z. and C.W.; Validation, W.Y., M.M. and C.W.; Formal analysis, W.Y., R.Z., M.M. and C.W.; Investigation, W.Y.; Resources, W.Y.; Writing—original draft, W.Y. and R.Z.; Writing—review & editing, W.Y., M.M. and C.W.; Visualization, R.Z.; Supervision, M.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Data are contained within the article.

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

References

1. Kamal, A.S.; Bukhari, S.M.A.H.; Khan, M.U.S.; Maqsood, T.; Fayyaz, M.A.B. *Traffic Pattern Plot: Video Identification in Encrypted Network Traffic*; Intelligent Sustainable Systems: Selected Papers of WorldS4 2022; Springer Nature Singapore: Singapore, 2023; Volume 2, pp. 77–84.
2. Rathore, R.S.; Hewage, C.; Kaiwartya, O.; Lloret, J. In-vehicle communication cyber security: Challenges and solutions. *Sensors* **2022**, *22*, 6679. [CrossRef]
3. Tomar, I.; Sreedevi, I.; Pandey, N. State-of-Art review of traffic light synchronization for intelligent vehicles: Current status, challenges, and emerging trends. *Electronics* **2022**, *11*, 465. [CrossRef]
4. Agbaje, P.; Anjum, A.; Mitra, A.; Oseghale, E.; Bloom, G.; Olufowobi, H. Survey of interoperability challenges in the internet of vehicles. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 22838–22861. [CrossRef]
5. Marwein, P.S.; Sur, S.N.; Gao, X.Z.; Kandar, D. Recent Survey on Internet of Vehicles: Architecture, Applications, Challenges, and Its Solutions. *J. Test. Eval.* **2024**, *52*, 20230095. [CrossRef]
6. Liang, B.; Lu, W.; Ran, B. Deploying Roadside Unit Efficiently in VANETs: A Multi-Objective Delay-Based Optimization Strategy Using Lagrangian Relaxation. *IEEE Trans. Intell. Transp. Syst.* **2023**. [CrossRef]
7. Guan, T.; Han, Y.; Kang, N.; Tang, N.; Chen, X.; Wang, S. An overview of vehicular cybersecurity for intelligent connected vehicles. *Sustainability* **2022**, *14*, 5211. [CrossRef]
8. Xie, Q.; Ding, Z.; Zheng, P. Provably Secure and Anonymous V2I and V2V Authentication Protocol for VANETs. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 7318–7327. [CrossRef]
9. Tengilimoglu, O.; Carsten, O.; Wadud, Z. Infrastructure-related challenges in implementing connected and automated vehicles on urban roads: Insights from experts and stakeholders. *IET Intell. Transp. Syst.* **2023**, *17*, 2352–2368. [CrossRef]
10. Li, X.; Liu, T.; Obaidat, M.S.; Wu, F.; Vijayakumar, P.; Kumar, N. A Lightweight Privacy-Preserving Authentication Protocol for VANETs. *IEEE Syst. J.* **2020**, *14*, 3547–3557. [CrossRef]
11. Kakei, S.; Shiraishi, Y.; Mohri, M.; Nakamura, T.; Hashimoto, M.; Saito, S. Cross-Certification Towards Distributed Authentication Infrastructure: A Case of Hyperledger Fabric. *IEEE Access* **2020**, *8*, 135742–135757. [CrossRef]
12. Tzeng, S.F.; Horng, S.J.; Li, T.; Wang, X.; Huang, P.-H.; Khan, M.K. Enhancing security and privacy for identity-based batch verification scheme in VANETs. *IEEE Trans. Veh. Technol.* **2015**, *66*, 3235–3248. [CrossRef]
13. Zhang, C.; Lu, R.; Lin, X.; Ho, P.-H.; Shen, X. An efficient identity-based batch verification scheme for vehicular sensor networks. In Proceedings of the IEEE INFOCOM 2008—The 27th Conference on Computer Communications, Phoenix, AZ, USA, 13–18 April 2008; IEEE: Piscataway, NJ, USA, 2008; pp. 246–250.
14. Lu, R.; Lin, X.; Zhu, H.; Ho, P.-H.; Shen, X. ECPP: An efficient Conditional Privacy Protection Protocol for Secure Vehicle Communication. In Proceedings of the IEEE INFOCOM 2008—The 27th Conference on Computer Communications, Phoenix, AZ, USA, 13–18 April 2008; IEEE: Piscataway, NJ, USA, 2008.
15. Chim, T.W.; Yiu, S.M.; Hui, L.C.K.; Li, V.O.K. SPECS: VANET's Security and Privacy Enhanced Communication Scheme. *AD Hoc Netw.* **2011**, *9*, 189–203. [CrossRef]
16. Azees, M.; Vijayakumar, P.; Deboarh, L.J. EAAP: Efficient Anonymous Authentication With Conditional Privacy-Preserving Scheme for Vehicular Ad Hoc Networks. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 2467–2476. [CrossRef]
17. Zhou, X.; Luo, M.; Vijayakumar, P. Efficient Certificateless Conditional Privacy-Preserving Authentication for VANETs. *IEEE Trans. Veh. Technol.* **2022**, *71*, 7863–7875. [CrossRef]
18. Zhang, L.; Xu, J. Blockchain-based anonymous authentication for traffic reporting in VANETs. *Connect. Sci.* **2022**, *34*, 1038–1065. [CrossRef]
19. Liu, J.; Li, X.; Jiang, Q.; Obaidat, M.S.; Vijayakumar, P. BUA: A Blockchain-based Unlinkable Authentication in VANETs. In Proceedings of the ICC 2020–2020 IEEE International Conference on Communications (ICC), Dublin, Ireland, 7–11 June 2020.
20. Feng, Q.; He, D.; Zeadally, S.; Liang, K. BPAS: Blockchain-Assisted Privacy-Preserving Authentication System for Vehicular Ad Hoc Networks. *IEEE Trans. Ind. Inform.* **2020**, *16*, 4146–4155. [CrossRef]
21. Zheng, J.; Wang, X.; Yang, Q.; Xiao, W.; Sun, Y.; Liang, W. A blockchain-based lightweight authentication and key agreement scheme for internet of vehicles. *Connect. Sci.* **2022**, *34*, 1430–1453. [CrossRef]
22. Chai, H.; Leng, S.; He, J.; Zhang, K.; Cheng, B. CyberChain: Cybertwin empowered blockchain for lightweight and privacy-preserving authentication in Internet of Vehicles. *IEEE Trans. Veh. Technol.* **2021**, *71*, 4620–4631. [CrossRef]
23. Chen, Y.; Chen, J. CPP-CLAS: Efficient and conditional privacy-preserving certificateless aggregate signature scheme for VANETs. *IEEE Internet Things J.* **2021**, *9*, 10354–10365. [CrossRef]
24. Cheng-Chi, L.; Lai, Y.-M. Secure batch validation through VANET's group testing. *Wirel. Netw.* **2013**, *19*, 1441–1449.
25. Guo, D.; Wen, F. Analysis and improvement of a robust smart card based-authentication scheme for multi-server architecture. *Wirel. Pers. Commun.* **2014**, *78*, 475–490. [CrossRef]
26. Kumari, S.; Om, H. Cryptanalysis and improvement of an anonymous multi-server authenticated key agreement scheme. *Wirel. Pers. Commun.* **2017**, *96*, 2513–2537. [CrossRef]
27. Jangirala, S.; Mukhopadhyay, S.; Das, A.K. A multi-server environment with secure and efficient remote user authentication scheme based on dynamic ID using smart cards. *Wirel. Pers. Commun.* **2017**, *95*, 2735–2767. [CrossRef]

28. Bellare, M.; Rogaway, P. Random oracles are practical: A paradigm for designing efficient protocols. In Proceedings of the 1st ACM Conference on Computer and Communications Security, Fairfax, VA, USA, 3–5 November 1993; pp. 62–73.
29. Wang, N.; Hu, J. Performance Analysis of IEEE 802.11 p for the Internet of Vehicles with Bursty Packet Errors. In Proceedings of the 2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), Wuhan, China, 9–11 December 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1435–1440.
30. Özyilmaz, B.; Paker, S. SAE J2735 message suggestion for traffic light-vehicles communication. In Proceedings of the 2018 26th Signal Processing and Communications Applications Conference (SIU), Izmir, Turkey, 2–5 May 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–4.
31. OpenSSL. 2018. Available online: https://www.openssl.org/ (accessed on 3 November 2023).
32. GMP. 2016. Available online: https://gmplib.org/ (accessed on 3 November 2023).
33. PBC Library. 2019. Available online: https://crypto.stanford.edu/pbc/ (accessed on 3 November 2023).
34. Castellon, C.E.; Roy, S.; Kreidl, O.P.; Dutta, A.; Bölöni, L. Towards an Energy-Efficient Hash-based Message Authentication Code (HMAC). In Proceedings of the 2022 IEEE 13th International Green and Sustainable Computing Conference (IGSC), Pittsburgh, PA, USA, 24–25 October 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 1–7.
35. Zhu, Y.; Liu, Y.; Wu, M.; Li, J.; Liu, S.; Zhao, J. Research on Secure Communication on In-Vehicle Ethernet Based on Post-Quantum Algorithm NTRUEncrypt. *Electronics* **2022**, *11*, 856. [CrossRef]