This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2023.3340723

PPRU: A Privacy-Preserving Reputation Updating Scheme for Cloud-Assisted Vehicular Networks

Zhiquan Liu, Lin Wan, Jingjing Guo, Feiran Huang, Xia Feng, Libo Wang, and Jianfeng Ma

***Abstract*—Vehicular networks have huge potential to improve road safety and traffic efficiency, especially in the context of large models. Cloud computing can significantly improve the performance of vehicular networks, and the concept of cloud-assisted vehicular networks comes into being. Reputation management plays a crucial role in vehicular networks, since it can help each vehicle evaluate the trustworthiness of the other vehicles and the received messages. Reputation updating is essential in reputation management and it is usually done by the Trusted Authority (TA) regularly after collecting, decrypting, and verifying a large number of reputation feedbacks, which leads to great computation and communication overheads on the TA side and even makes the TA become the bottleneck of reputation management system. In this paper, we propose a novel Privacy-Preserving Reputation Updating (PPRU) scheme for cloud-assisted vehicular networks based on the Elliptic Curve Cryptography (ECC) and Paillier algorithms, in which the reputation feedbacks are collected and preprocessed by the honest-but-curious Cloud Service Provider (CSP) in a privacy-preserving manner, and the computation and communication overheads on the TA side can be dramatically reduced by about 88.36% and 83.88% as a result, respectively. Meanwhile, the proposed scheme can provide strong privacy preservation, strong security, and robust reputation management with acceptable computation and communication overheads. Furthermore, the comprehensive theoretical analysis and simulation evaluation are conducted, and the results demonstrate that the proposed scheme is significantly superior to the existing schemes in several aspects.**

***Index Terms*—Vehicular networks, cloud-assisted, privacy-preserving, reputation updating, reputation management, privacy preservation.**

I. INTRODUCTION

NOWADAYS, vehicular networks have received extensive attention from the government, enterprise, and academe, due to their huge potential to improve road safety and traffic efficiency, especially in the context of large models [1]–[3]. With the increasing number of vehicles and continuous enrichment of vehicular applications, the traditional architectures of vehicular networks are facing more and more challenges in recent years, and it is imperative to exploit new architectures to further improve the performance of vehicular networks [4], [5].

Cloud computing can provide vehicular networks with on-demand computing and storage resources and greatly improve the performance of vehicular networks. As a result, the concept of cloud-assisted vehicular networks comes into being in recent years [6], [7]. Although cloud computing can bring lots of benefits to vehicular networks, cloud-assisted vehicular networks still face many security, privacy, and trust challenges due to their large, open, and highly dynamic characteristics [8], [9].

Reputation management plays a crucial role in vehicular networks, since it can help each vehicle evaluate the trustworthiness of the other vehicles and the received messages, so as to avoid the serious consequences caused by unreal messages from malicious vehicles [10], [13]. Reputation updating is an essential component of reputation management and it is usually done by the Trusted Authority (TA) regularly after collecting, decrypting, and verifying a large number of reputation feedbacks, which leads to great computation and communication overheads on the TA side and even makes the TA become the bottleneck of reputation management system [4], [14].

One potential way to dramatically reduce the computation and communication overheads on the TA side in reputation updating is to adopt the architecture of cloud-assisted vehicular networks, where the reputation feedbacks are collected and preprocessed by the Cloud Service Provider (CSP) [4]. However, the CSP is honest-but-curious. That is, it will perform the predefined operations honestly, but it is curious about a vehicle' privacy, such as unique identifier, reputation value, and feedback score. Thus, the above collecting and preprocessing operations must be conducted in a privacy-preserving manner, and one possible way is to adopt homomorphic encryption algorithms, such as the Paillier algorithm [15].

In addition, to improve the applicability to the large, open, and highly dynamic vehicular networks, an ideal scheme should provide strong security against multiple kinds of attacks, such as forgery, replay, Sybil, self-praise, and tampering attacks [11], [16], [17]. Besides, since the feedback scores

This work was supported in part by the National Natural Science Foundation of China under Grant 62272195; in part by the Fundamental Research Funds for the Central Universities under Grant 21622402; in part by the National Natural Science Foundation of China under Grant 62272203, Grant 61932010, and Grant 62032025; in part by the Natural Science Basic Research Program of Shaanxi under Grant 2022JQ-603; in part by the Science and Technology Program of Guangzhou of China under Grant 202201010421; and in part by Guangxi Key Laboratory of Trusted Software under Grant KX202303. (*Corresponding author: Jingjing Guo.*)

Z. Liu, L. Wan, F. Huang, and L. Wang are with the College of Cyber Security, Jinan University, Guangzhou 510632, China. (e-mail: zqliu@vip.qq.com; wanlinwan000@gmail.com; huangfr@jnu.edu.cn; wanglibo12b@mails.ucas.edu.cn).

Z. Liu is also with Guangdong Key Laboratory of Data Security and Privacy Preserving, Jinan University, Guangzhou 510632, China, with Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin 541004, China, and with Cyberdataforce (Beijing) Technology Ltd. Beijing 100036, China.

J. Guo and J. Ma are with the School of Cyber Engineering, Xidian University, Xi'an 710071, China. (e-mail: jjguo@xidian.edu.cn; jfma@mail.xidian.edu.cn).

X. Feng is with the Faculty of Data Science, City University of Macau, Macau 999078, China. (e-mail: xiafeng@cityu.mo).

from honest vehicles with high reputation values are usually more trustworthy than those from malicious vehicles with low reputation values, to improve the robustness of reputation management, the weighted average, rather than the simple average, of feedback score ciphertexts should be supported in an ideal scheme [12], [18], [19]. Meanwhile, to improve the practicality, in an ideal scheme, the computation and communication overheads should be acceptable, and some time-consuming operations, such as bilinear pairing, should be avoided if possible [20]. Besides, some attractive technologies, such as batch validation, should be adopted if possible to greatly reduce the total computation overhead [21].

In recent years, plenty of reputation updating schemes have been proposed for vehicular networks [4], [5], [14], [18], [22]. These schemes provide lots of brilliant ideas, but they have the following limitations. Gong *et al.* [22] and Liu *et al.* [14] completely ignored the privacy preservation for reputation feedbacks, which may leak the unique identifier and feedback score of a vehicle in reputation updating. Liu *et al.* [5] utilized the TA to collect, decrypt, and verify the reputation feedbacks one by one, instead of adopting a batch validation manner, without the assistance of CSP, which will lead to great computation overhead on the TA side and even make the TA become the bottleneck of reputation management system. Cheng *et al.* [4] utilized the honest-but-curious CSP to compute and store the reputation values of vehicles, which may leak the reputation value privacy of vehicles, and their scheme merely supports the simple average of feedback score ciphertexts and fails to resist the infamous Sybil attack. Zhang *et al.* [18] completely ignored the infamous Sybil attack and adopted the time-consuming bilinear pairing to verify the signatures, and the honest-but-curious CSP in their scheme may leak the reputation value and feedback score of a vehicle.

Aiming at dramatically reducing the computation and communication overheads on the TA side in a privacy-preserving manner and overcoming the aforementioned limitations in the existing schemes, we propose a novel Privacy-Preserving Reputation Updating (PPRU) scheme for cloud-assisted vehicular networks based on the Elliptic Curve Cryptography (ECC) [23] and Paillier [15] algorithms in this paper, and the major contributions of this work can be summarized as follows.

- This work proposes a novel PPRU scheme for reputation updating in vehicular networks, in which the reputation feedbacks are collected and preprocessed by the honest-but-curious CSP in a privacy-preserving manner, and the computation and communication overheads on the TA side can be dramatically reduced as a result.
- The proposed PPRU scheme can provide strong privacy preservation for the unique identifier, reputation value, and feedback score of a vehicle, and can provide strong security against the forgery, replay, Sybil, self-praise, and tampering attacks.
- The proposed PPRU scheme supports the weighted average, rather than the simple average, of feedback score ciphertexts and can provide robust reputation management. Meanwhile, it supports the batch validation of signatures and avoids the utilization of time-consuming bilinear pairing, and the computation and communication overheads are acceptable.
- This work conducts comprehensive theoretical analysis and simulation evaluation, and the results demonstrate that the proposed PPRU scheme is significantly superior to the existing schemes in several aspects, especially in the computation and communication overheads on the TA side, the privacy preservation for the reputation value and feedback score of a vehicle, and the security against the Sybil attack.

The remainder of this paper is structured as follows. Section II reviews some related work and its limitations, and Section III introduces the related preliminaries. Then, Section IV presents the system model, attack model, design goals, and formalized symbols, and Section V details the various stages in the PPRU scheme. Afterwards, Sections VI and VII detail the comprehensive theoretical analysis and simulation evaluation, respectively, followed by the conclusion and future work in Section VIII.

II. Related Work

Cloud-assisted vehicular networks are widely considered as a new architecture of vehicular networks, which can greatly improve the performance of vehicular networks [25], [27]. In recent years, the architectures, features, classifications, challenges, and potential applications of cloud-assisted vehicular networks have been analyzed in detail [26], [27]. In addition, some researchers [8], [9] pointed out that the cloud-assisted vehicular networks still face many security, privacy, and trust challenges due to their large, open, highly dynamic characteristics.

Reputation management plays a crucial role in vehicular networks, in which reputation updating is an essential component. In recent years, plenty of reputation updating schemes have been proposed for vehicular networks [13], [14], [22], [28]. Gong *et al.* [22] realized the reputation updating based on direct and indirect reputation parameters as well as the feedbacks of communication results, and completely ignored the security and privacy of reputation feedbacks. Liu *et al.* [14] adopted the digital signature technology to protect the authenticity and completeness of reputation feedbacks and completely ignored the privacy preservation for reputation feedbacks. In these schemes, the lack of privacy preservation for reputation feedbacks may result in the leakage of privacy-sensitive information of a vehicle, such as unique identifier, reputation value, and feedback score, in reputation updating.

To overcome the limitations of the above schemes, Liu *et al.* [5] adopted the asymmetric encryption and digital signature technologies to protect the authenticity, completeness, and privacy of reputation feedbacks, and utilized the TA to collect, decrypt, and verify the reputation feedbacks one by one. However, their scheme fails to support batch validation and does not take advantage of cloud. As a result, their scheme will lead to great computation and communication overheads on the TA side and even make the TA become the bottleneck of reputation management system.

To reduce the computation and communication overheads on the TA side and provide privacy preservation for the

TABLE I: An intuitive property comparison with the existing schemes

Properties	Gong *et al.*'s [22]	Liu *et al.*'s [14]	Liu *et al.*'s [5]	Cheng *et al.*'s [4]	Zhang *et al.*'s [18]	Ours
Cloud-assisted	×	×	×	✓	✓	✓
Batch validation	×	×	×	✓	✓	✓
Reputation value privacy	×	×	✓	×	×	✓
Feedback score privacy	×	×	✓	✓	×	✓
Sybil attack-resisted	×	✓	✓	×	×	✓
Weighted average	×	✓	✓	×	✓	✓
Bilinear pairing-free	✓	✓	✓	✓	×	✓

Note: × and ✓ denote support and nonsupport, respectively.

unique identifier and feedback score of a vehicle, Cheng *et al.* [4] proposed a privacy-preserving reputation updating scheme for cloud-assisted vehicular networks. In their scheme, the feedback score privacy is achieved via the Paillier algorithm, but the honest-but-curious CSP, responsible for computing and storing the reputation values of vehicles, is able to obtain the reputation values of all vehicles. However, as analyzed in many recent researches [10], [17], [19], reputation value is an important attribute of a vehicle, whose disclosure will lead to a reputation link attack and even expose the location and trajectory of a vehicle. Thus, the inability to provide privacy preservation for reputation value is a non-negligible limitation of their scheme. Meanwhile, their scheme fails to resist the infamous Sybil attack and merely supports the simple average of feedback score ciphertexts. However, the Sybil attack will greatly disturb the normal operations in a reputation management system [29], [30], and as revealed in many recent researches [5], [14], [31], the simple average will provide obviously weaker robustness against malicious feedback providers than the weighted average, in which the reputation values of feedback providers are adopted as important weights.

To realize the weighted average of feedback score ciphertexts, Zhang *et al.* [18] proposed a trust-based and privacy-preserving platoon recommendation scheme. In their scheme, the honest-but-curious CSP is able to obtain the feedback scores of all vehicles and the reputation values of head vehicles, as well as the incremental reputation values of user vehicles, which may leak the reputation value and feedback score of a vehicle. As analyzed earlier, the disclosure of reputation value may lead to a reputation link attack and even expose the location and trajectory of a vehicle, and the leakage of feedback score will lead to the possibility of a feedback provider being retaliated against and reduce the willingness of a feedback provider to submit reputation feedbacks [4]. Meanwhile, their scheme also completely ignores the infamous Sybil attack, and adopts the time-consuming bilinear pairing to verify the signatures, which greatly reduces the practicality of their scheme [20], [32].

Aiming at overcoming the aforementioned limitations in the existing schemes, we propose a novel PPRU scheme and an intuitive property comparison with the existing schemes is shown in Table I.

III. Preliminaries

In this section, we mainly introduce two important preliminaries involved in the PPRU scheme, namely ECC and Paillier algorithms.

A. ECC Algorithm

The ECC algorithm, first proposed by Miller and Koblitz [23], [33], is able to provide higher security level with shorter key size when compared with the other asymmetric cryptographic algorithms. Specifically, the ECC algorithm contains the following three main stages [34].

- *Key Generation*: Given a large prime number p and a finite field Z_p, an elliptic curve $y^2 = x^3 + a \cdot x + b \bmod p$ can be generated, where $a, b \in Z_p$ and $4a^3 + 27b^2 \neq 0$. All the points in the elliptic curve and the infinity point constitute an additive cyclic group $\mathbb{G}$ with a q-order generator G. Then, given G and random $s \in Z_q^*$, computing $S = s \cdot G \in \mathbb{G}$ is efficient. However, given G and random $S \in \mathbb{G}$, computing $s \in Z_q^*$ satisfying $S = s \cdot G$ is infeasible in probabilistic polynomial time (which is also named Elliptic Curve Discrete Logarithm Problem (ECDLP) assumption [35]). As a result, the ECC public key and ECC private key are S and s, respectively.
- *Encryption*: Given a plaintext $m \in \{0,1\}^*$ and a ECC public key S, the m is firstly encoded to $M \in \mathbb{G}$, and then M's ciphertext is calculated as $C = (r \cdot G, M + r \cdot S)$, where $r \xleftarrow{R} Z_q^*$. It is obvious that $C \in \mathbb{G} \times \mathbb{G}$. To simplify the illustration, we define $C = \mathcal{E}_e(m, S)$.
- *Decryption*: Given a ciphertext C and a ECC private key s, the C's plaintext is calculated as $(M + r \cdot S) - s \cdot (r \cdot G) = M$, and then M is decoded to m. To simplify the illustration, we define $m = \mathcal{D}_e(C, s)$.

B. Paillier Algorithm

The Paillier algorithm, first proposed by Paillier [15], is able to provide more efficient additive homomorphic function than the other homomorphic algorithms. Specifically, the Paillier algorithm contains the following three main stages [4], [18].

- *Key Generation*: Given two random large prime numbers p' and q' satisfying $gcd(p' \cdot q', (p'-1) \cdot (q'-1)) = 1$, $n = p' \cdot q'$ and $\lambda = lcm(p'-1, q'-1)$ are calculated, where $gcd(x, y)$ and $lcm(x, y)$ denote the greatest common divisor and least common multiple of two numbers x and y, respectively. Next, a random value $g \in Z_{n^2}^*$ satisfying $gcd(\mathcal{L}(g^\lambda \bmod n^2), n) = 1$ is selected, where $\mathcal{L}(x) = \frac{x-1}{n}$. As a result, the Paillier public key and Paillier private key are (n, g) and (λ, μ), respectively, where $\mu = \mathcal{L}(g^\lambda \bmod n^2)^{-1} \bmod n$.

- *Encryption*: Given a plaintext $m \in Z_n$ and a Paillier public key (n, g), the m's ciphertext is calculated as $c = g^m \cdot (r')^n \bmod n^2$, where $r' \xleftarrow{R} Z_n^*$. It is obvious that $c \in Z_{n^2}^*$. To simplify the illustration, we define $c = \mathcal{E}_P(m, n, g)$.
- *Decryption*: Given a ciphertext c and a Paillier private key (λ, μ), the c's plaintext is calculated as $m = \mathcal{L}(c^\lambda \bmod n^2) \cdot \mu \bmod n$. To simplify the illustration, we define $m = \mathcal{D}_P(c, \lambda, \mu)$.

Besides, for $\forall m_1, m_2 \in Z_n$, the Paillier algorithm has the following two homomorphic properties.

- $\mathcal{D}_P(\mathcal{E}_P(m_1, n, g) \cdot \mathcal{E}_P(m_2, n, g) \bmod n^2, \lambda, \mu) = m_1 + m_2 \bmod n$.
- $\mathcal{D}_P(\mathcal{E}_P(m_1, n, g)^{m_2} \bmod n^2, \lambda, \mu) = m_1 \cdot m_2 \bmod n$.

IV. System Model, Attack Model, Design Goals, and Formalized Symbols

In this section, we first introduce the system model, attack model, and design goals of the PPRU scheme, and then list the formalized symbols in the PPRU scheme for ease of later illustration.

A. System Model

The system model of the PPRU scheme is illustrated in Fig. 1, where there exist five kinds of primary entities, namely a Trusted Authority (TA) and a Cloud Service Provider (CSP), as well as a number of Cellular Base Stations (CBSs), Road Side Units (RSUs), and Vehicles.

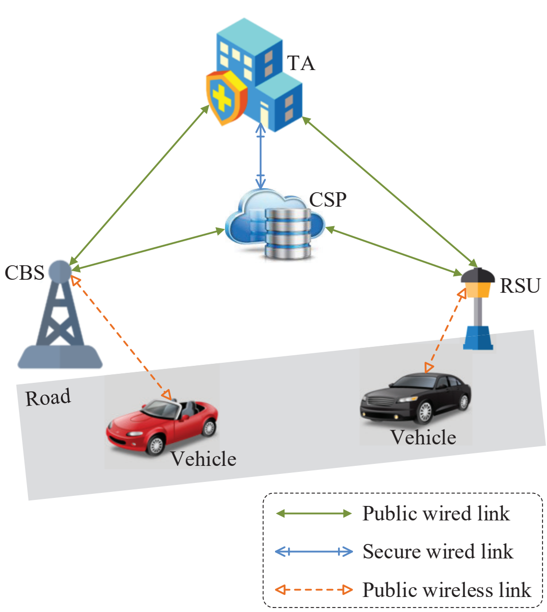


Fig. 1. System model of the PPRU scheme.

TA: The TA is mainly responsible for vehicle registration as well as storing and periodically updating vehicles' reputation values with the aid of CSP. Besides, it contains a clock and divides the time into a series of equal-length time intervals, and generates and distributes the reputation certificate and secret values to a vehicle when it receives the vehicle's request.

CSP: The CSP is equipped with a clock which keeps in sync with that in the TA. Besides, it is considered to have sufficient computational power, and it is mainly responsible for verifying and aggregating the reputation feedbacks and then sending the aggregated reputation feedback to the TA.

CBSs: The CBSs are regarded to be installed in the vicinity of the road and serve as the communication relays between the TA/CSP and nearby vehicles, and they generally connect to the TA/CSP and nearby vehicles via the wired manner and wireless manner, respectively.

RSUs: The RSUs are typically installed on the side of the road and also serve as the communication relays between the TA/CSP and nearby vehicles, and they generally connect to the TA/CSP and nearby vehicles via the wired manner and wireless manner, respectively.

Vehicles: Each vehicle is equipped with a clock which is in sync with that in the TA and a Trusted Platform Module (TPM) which can securely store its private information. Besides, each vehicle communicates with nearby CBSs and RSUs via the wireless manner, and periodically generates and submits a reputation feedback to the CSP with the relay of a nearby CBS or RSU for reputation updating.

B. Attack Model

Similar to many recent researches [4], [5], [14], we assume that the TA is fully trusted and will not collude with the other entities. Besides, the TA maintains a secure database which can securely store the vehicles' information. Meanwhile, the CSP, CBSs, and RSUs are considered to be honest-but-curious, that is, they will honestly perform predesigned operations but they are curious about the private information of a vehicle. For example, they may attempt to reveal the unique identifier, reputation value, and feedback score of a vehicle in the reputation feedback submitting and aggregation processes.

In addition, the vehicles may be malicious. Specifically, in the PPRU scheme, a malicious vehicle (the "vehicle" is also referred to as a feedback provider) may forge its reputation score or pseudonym in the reputation certificate (i.e., conduct the forgery attack), may submit an outdated reputation feedback (i.e., conduct the replay attack), may submit multiple reputation feedbacks for the same vehicle (the "vehicle" is also referred to as a feedback target) in a short period of time by adopting multiple pseudonyms (i.e., conduct the Sybil attack), and may submit a reputation feedback to improve its own reputation value (i.e., conduct the self-praise attack). Besides, the adversary (e.g., a malicious vehicle) may tamper with the feedback score or pseudonym of feedback target in the reputation feedback (i.e., conduct the tampering attack).

C. Design Goals

Based on the aforementioned attack model, the basic goal of the proposed PPRU scheme is to provide a privacy-preserving reputation updating scheme for cloud-assisted vehicular networks. Specifically, the following design goals should be achieved.

Strong Privacy Preservation: To provide strong privacy preservation, in the proposed PPRU scheme, the unique identifier, reputation value, and feedback score of a vehicle should not be revealed or linked by the adversary in the reputation feedback submitting and aggregation processes.

Strong Security: To provide strong security, the proposed PPRU scheme should be able to defend against multiple kinds of common attacks, including the forgery, replay, Sybil, self-praise, and tampering attacks in the reputation feedback submitting and aggregation processes.

Robust Reputation Management: To provide robust reputation management, the proposed PPRU scheme should support the weighted average, rather than the simple average, of feedback score ciphertexts, as the former can provide obviously stronger robustness against malicious feedback providers than the latter.

Acceptable Computation and Communication Overheads: To achieve acceptable computation and communication overheads as well as enhance the practicality of scheme, the proposed PPRU scheme should support the batch validation of signatures and avoid the utilization of time-consuming bilinear pairing. Specifically, the computation and communication overheads on the TA side should be dramatically reduced.

D. Formalized Symbols

For ease of later illustration, Table II lists the formalized symbols in the PPRU scheme.

TABLE II: Formalized symbols in the PPRU scheme

Symbols	Descriptions
p, a, b, q, G	Parameters in the ECC algorithm
S_T, s_T	TA's ECC public key and ECC private key, respectively
$\mathcal{E}_e(), \mathcal{D}_e()$	ECC encryption function and ECC decryption function, respectively
$(n, g), (\lambda, \mu)$	TA's Paillier public key and Paillier private key, respectively
$\mathcal{E}_p(), \mathcal{D}_p()$	Paillier encryption function and Paillier decryption function, respectively
$T_1, T_2, ...$	A series of equal-length time intervals
v_k	Secret value generated by the TA for T_k and securely stored by the TA and CSP
$\mathcal{H}()$	Hash function mapping any a bit string to a number in Z_o^*, where $o = min(q, n)$
V_i	Vehicle with a unique identifier i
S_i, s_i	V_i's ECC public key and ECC private key, respectively
$R_{i,0}, R_{i,k}$	V_i's initial reputation value and reputation value in T_k, respectively
$\mathcal{R}()$	Rounding function
η	Public parameter, where $\eta \in Z_n^*$ and $5 < \eta \ll n$
$Q_{i,k}^1$	Request generated by V_i in T_k
$\mathcal{S}(), \mathcal{V}()$	Signature generation function and signature verification function, respectively
$x_{i,k}^1, x_{i,k}^2$	Two secret values generated by the TA for V_i in T_k
$P_{i,k}, C_{i,k}$	V_i's pseudonym and reputation certificate in T_k, respectively
$r'_{i,k}$	Random value generated by the TA for V_i in T_k
$Q_{i,k}^2$	Response generated by the TA for V_i in T_k
$f_{i,j,k}, e_{i,j,k}$	Feedback score and encrypted feedback score generated by V_i's TPM for V_j in T_k, respectively
$r''_{i,j,k}, F_{i,j,k}$	Random value and reputation feedback generated by V_i's TPM for V_j in T_k, respectively
$y_{i,k}$	Random value generated by V_i's TPM in T_k
$D_{j,k}^1, D_{j,k}^2$	Two aggregated ciphertexts generated by the CSP for V_j in T_k
A_k	Aggregated feedback generated by the CSP in T_k
$R_{j,k}^I$	V_j's incremental reputation value in T_k
$\omega_{j,k}$	Weight value for calculating V_j's reputation value in T_{k+1}
ε, ξ	Control factor and decay factor for updating vehicles' reputation values, respectively

V. Various Stages in the PPRU Scheme

In this section, we detail the various stages in the PPRU scheme.

A. Scheme Initialization

1) Initialization of the TA and CSP

When the PPRU scheme is deployed in a vehicular network, the TA and CSP first set their clocks (which are assumed to be always synchronized) and divide the time into a series of equal-length time intervals $T_1, T_2, ...$. Next, the TA initializes the ECC and Paillier algorithms as shown in Section III and generates its ECC public key S_T, ECC private key s_T, Paillier public key (n, g), and Paillier private key (λ, μ), where s_T and (λ, μ) are always kept confidential by the TA. Then, the TA defines a hash function $\mathcal{H}()$ mapping any a bit string ς to a number in Z_o^* (where $o = min(q, n)$ denotes the minimum of q and n, in which q and n are the parameters of the ECC and Paillier algorithms, respectively, thus $\mathcal{H}(\varsigma) \in Z_q^*$ and $\mathcal{H}(\varsigma) \in Z_n^*$ hold simultaneously) and defines the range of the reputation values of vehicles as Z_η (where $\eta \in Z_n^*$ and $5 < \eta \ll n$). Besides, the TA sends S_T, (n, g), $\mathcal{H}()$, ECC algorithm (as well as its parameters p, a, b, q, G), and Paillier algorithm to the CSP via a secure wired link, and then both the TA and CSP store them locally. Furthermore, the TA generates a secret value $v_k \in Z_q^*$ for each T_k and sends v_k to the CSP via a secure wired link at the beginning of each T_k, where $k \in \{1, 2, ...\}$. After that, both the TA and CSP securely store v_k.

2) Initialization of the CBSs and RSUs

When the PPRU scheme is deployed in a vehicular network, the CBSs are installed in the vicinity of the road, and the RSUs are installed on the side of the road. Besides, the public wired links between each CBS and the TA, each CBS and the CSP, each RSU and the TA, and each RSU and the CSP are constructed. Then, both the CBSs and RSUs become the relays of the communication between the vehicles and TA as well as the vehicles and CSP.

B. Vehicle Registration

When a new vehicle registers with the TA in T_k, the TA first assigns a unique identifier i to it, and then the new vehicle is named V_i for ease of illustration. Next, the TA generates a ECC public key S_i and a ECC private key s_i for V_i, and then equips V_i with a TPM to maintain i, s_i, S_T, (n, g), $\mathcal{H}()$, ECC algorithm (as well as its parameters p, a, b, q, G), Paillier algorithm, a clock which is always in sync with that in the TA and CSP, V_i's reputation certificate $C_{i,k}$, and V_i's secret values $x_{i,k}^1$ and $x_{i,k}^2$, where $C_{i,k}$, $x_{i,k}^1$, and $x_{i,k}^2$ will be detailed in Section V.*C*. Then, inspired by the previous work [5], [14], [19], [36], the TA sets an initial reputation value $R_{i,0}$ for V_i

according to the category of V_i as

$$R_{i,0} = \begin{cases} \mathcal{R}(0.9 \cdot \eta), \text{ if } V_i \text{ is a law enforcement vehicle} \\ \mathcal{R}(0.5 \cdot \eta), \text{ if } V_i \text{ is a public service vehicle} \\ \mathcal{R}(0.1 \cdot \eta), \text{ if } V_i \text{ is a private vehicle} \end{cases} \quad (1)$$

where $\mathcal{R}()$ denotes the rounding function. That is, V_i's reputation value in T_k is set as $R_{i,k} = R_{i,0}$. From Eq. (1), we can easily find that $R_{i,0} \in Z_\eta$, thus $R_{i,k} \in Z_\eta$. Afterwards, the TA stores V_i's information (i.e., i, S_i, k, $R_{i,k}$, etc.) in the secure database.

C. Reputation Certificate and Secret Value Requesting

At the beginning of each time interval T_k, each vehicle (e.g., V_i) requests the TA for its new reputation certificate and secret values via the relay of a nearby CBS or RSU. Specifically, V_i first generates a request $Q^1_{i,k} = \mathcal{E}_e(i||k||\sigma^1_{i,k}, S_T)$, where $||$ denotes the concatenation of bit strings (the same below), and $\sigma^1_{i,k} = \mathcal{S}(i||k, s_i)$ denotes the signature with s_i on "$i||k$", in which $\mathcal{S}()$ denotes the signature generation function which can be realized by utilizing the Elliptic Curve Digital Signature Algorithm (ECDSA) [34], [37]. Next, V_i sends $Q^1_{i,k}$ to the TA via the relay of a nearby CBS or RSU.

After receiving $Q^1_{i,k}$, the TA first decrypts it with s_T to obtain i, k, and $\sigma^1_{i,k}$, and then derives the current time interval's serial number k' from its clock. Next, the TA verifies the validity (including timeliness, integrity, and authenticity, the same below) of $Q^1_{i,k}$ by checking whether $k = k'$ and $\mathcal{V}(i||k, \sigma^1_{i,k}, S_i) = \text{TRUE}$ hold, where S_i can be obtained by querying the secure database and $\mathcal{V}()$ denotes the signature verification function which can be realized by utilizing the ECDSA algorithm [34], [37]. Then, the TA tries to retrieve the reputation certificate $C_{i,k}$ and secret values $x^1_{i,k}$, $x^2_{i,k}$ in T_k of V_i from the secure database.

- If the result set is empty, the TA first randomly picks a secret value $x^1_{i,k} \in Z^*_q$ and generates a pseudonym $P_{i,k} = x^1_{i,k} \cdot G$ for V_i, and then generates another secret value $x^2_{i,k} = \mathcal{H}(k||v_k||P_{i,k})$ for V_i. Besides, the TA picks a random value $r'_{i,k} \in Z^*_n$ for V_i and retrieves V_i's reputation value $R_{i,k}$ in T_k from the secure database, and then generates a reputation certificate $C_{i,k} = (P_{i,k}, k, C^1_{i,k}, C^2_{i,k})$ for V_i, where
$$\begin{cases} C^1_{i,k} = g^{R_{i,k}} \cdot (r'_{i,k} \cdot \mathcal{H}(k||v_k))^n \bmod n^2 \\ C^2_{i,k} = g^{R_{i,k} \cdot v_k + x^2_{i,k}} \cdot (r'_{i,k})^{n \cdot v_k} \bmod n^2 \end{cases} \quad (2)$$
Next, the TA stores $(i, C_{i,k}, x^1_{i,k}, x^2_{i,k})$ in the secure database.
- If the result set is non-empty, the TA adopts the existing $C_{i,k}$, $x^1_{i,k}$, and $x^2_{i,k}$ in the result set, instead of generating new ones. This strategy can ensure that V_i can merely obtain a group of $C_{i,k}$, $x^1_{i,k}$, and $x^2_{i,k}$ for each T_k even though it requests the TA for multiple times, and enhance the security of the PPRU scheme against the infamous Sybil attack.

Afterwards, the TA generates a corresponding response $Q^2_{i,k} = \mathcal{E}_e(i||C_{i,k}||x^1_{i,k}||x^2_{i,k}||\sigma^2_{i,k}, S_i)$ for V_i, where $\sigma^2_{i,k} = \mathcal{S}(i||C_{i,k}||x^1_{i,k}||x^2_{i,k}, s_T)$ denotes the signature with s_T on "$i||C_{i,k}||x^1_{i,k}||x^2_{i,k}$". Next, the TA sends $Q^2_{i,k}$ to V_i via the relay of a CBS or RSU near to V_i.

After receiving $Q^2_{i,k}$, V_i's TPM first decrypts it with s_i to obtain i, $C_{i,k}$, $x^1_{i,k}$, $x^2_{i,k}$, and $\sigma^2_{i,k}$, and then obtains its unique identifier i' from the storage, extracts k from $C_{i,k}$, and derives the current time interval's serial number k' from its clock. Next, V_i's TPM verifies the validity of $Q^2_{i,k}$ by checking whether $i = i'$, $k = k'$, and $\mathcal{V}(i||C_{i,k}||x^1_{i,k}||x^2_{i,k}, \sigma^2_{i,k}, S_T) = \text{TRUE}$ hold, where S_T can be obtained from V_i's TPM. Next, V_i's TPM securely stores $C_{i,k}$, $x^1_{i,k}$, and $x^2_{i,k}$.

D. Reputation Feedback Generation and Submitting

In T_k, if a vehicle V_i with the pseudonym $P_{i,k}$ is to generate a reputation feedback $F_{i,j,k}$ for another vehicle V_j with the pseudonym $P_{j,k}$ (where V_i is referred to as a feedback provider, V_j is referred to as a feedback target), V_i's TPM first generates a feedback score $f_{i,j,k} \in Z_\eta$ according to the quality of V_j's messages (the detailed generation method of $f_{i,j,k}$ is discussed in [14], [31] and beyond the scope of this paper due to limited space), and then generates a random value $r''_{i,j,k} \in Z^*_n$ (which is kept confidential by V_i's TPM) and an encrypted feedback score $e_{i,j,k}$ as

$$\begin{aligned} & e_{i,j,k} \\ =& (C^1_{i,k})^{f_{i,j,k}} \cdot (r''_{i,j,k})^n \bmod n^2 \\ =& g^{R_{i,k} \cdot f_{i,j,k}} \cdot ((r'_{i,k} \cdot \mathcal{H}(k||v_k))^{f_{i,j,k}} \cdot r''_{i,j,k})^n \bmod n^2 \end{aligned} \quad (3)$$

Afterwards, V_i's TPM generates a random value $y_{i,k} \in Z^*_q$ (which is kept confidential by V_i's TPM and unique for a certain pair of i and k) and calculates $Y_{i,k} = y_{i,k} \cdot G$, and then generates the reputation feedback $F_{i,j,k} = (C_{i,k}, P_{j,k}, e_{i,j,k}, Y_{i,k}, \mathcal{F}_{i,j,k})$ for V_j, where $\mathcal{F}_{i,j,k} = x^1_{i,k} \cdot \mathcal{H}(C_{i,k}||P_{j,k}||e_{i,j,k}||Y_{i,k}) + x^2_{i,k} + y_{i,k} \bmod q$ denotes the signature with $x^1_{i,k}$ and $x^2_{i,k}$ on "$C_{i,k}||P_{j,k}||e_{i,j,k}||Y_{i,k}$" (Note that V_i's TPM can merely obtain the pseudonym $P_{j,k}$ from V_j's messages and is ignorant of its unique identifier j, thus $P_{j,k}$, instead of j, is included in $F_{i,j,k}$). Next, V_i submits $F_{i,j,k}$ to the CSP via the relay of a nearby CBS or RSU.

E. Reputation Feedback Verification and Aggregation

Whenever receiving a reputation feedback marked as $F_{i,j,k}$, the CSP first extracts $P_{i,k}$ and k from $C_{i,k}$, retrieves v_k from its local storage, and derives the current time interval's serial number k'' from its clock, and then verifies the validity of $C_{i,k}$ by checking whether $k = k''$ and

$$\begin{aligned} & (C^1_{i,k})^{v_k} \cdot g^{\mathcal{H}(k||v_k||P_{i,k})} \bmod n^2 \\ =& C^2_{i,k} \cdot \mathcal{H}(k||v_k)^{n \cdot v_k} \bmod n^2 \end{aligned} \quad (4)$$

hold. The correctness of Eq. (4) is proved as

$$\begin{aligned} & (C^1_{i,k})^{v_k} \cdot g^{\mathcal{H}(k||v_k||P_{i,k})} \bmod n^2 \\ =& (g^{R_{i,k}} \cdot (r'_{i,k} \cdot \mathcal{H}(k||v_k))^n)^{v_k} \cdot g^{x^2_{i,k}} \bmod n^2 \\ =& (g^{R_{i,k} \cdot v_k} \cdot (r'_{i,k})^{n \cdot v_k} \cdot \mathcal{H}(k||v_k)^{n \cdot v_k}) \cdot g^{x^2_{i,k}} \bmod n^2 \\ =& (g^{R_{i,k} \cdot v_k + x^2_{i,k}} \cdot (r'_{i,k})^{n \cdot v_k}) \cdot \mathcal{H}(k||v_k)^{n \cdot v_k} \bmod n^2 \\ =& C^2_{i,k} \cdot \mathcal{H}(k||v_k)^{n \cdot v_k} \bmod n^2 \end{aligned} \quad (5)$$

$$\begin{aligned}&(\mathcal{F}_{i,j,k}-\mathcal{H}(k||v_k||P_{i,k}) \bmod q)\cdot G\\=&((x^1_{i,k}\cdot\mathcal{H}(C_{i,k}||P_{j,k}||e_{i,j,k}||Y_{i,k})+x^2_{i,k}+y_{i,k})-x^2_{i,k} \bmod q)\cdot G\\=&(x^1_{i,k}\cdot\mathcal{H}(C_{i,k}||P_{j,k}||e_{i,j,k}||Y_{i,k})+y_{i,k} \bmod q)\cdot G\\=&\mathcal{H}(C_{i,k}||P_{j,k}||e_{i,j,k}||Y_{i,k})\cdot(x^1_{i,k}\cdot G)+y_{i,k}\cdot G\\=&\mathcal{H}(C_{i,k}||P_{j,k}||e_{i,j,k}||Y_{i,k})\cdot P_{i,k}+Y_{i,k}\end{aligned}\tag{7}$$

$$\begin{aligned}&(\sum_{P_{i,k}\in\mathcal{I}_k}(\sum_{P_{j,k}\in\mathcal{J}_{i,k}}\mathcal{F}_{i,j,k})-|\mathcal{J}_{i,k}|\cdot\mathcal{H}(k||v_k||P_{i,k}) \bmod q)\cdot G\\=&\sum_{P_{i,k}\in\mathcal{I}_k}(\sum_{P_{j,k}\in\mathcal{J}_{i,k}}\mathcal{H}(C_{i,k}||P_{j,k}||e_{i,j,k}||Y_{i,k}) \bmod q)\cdot P_{i,k}+|\mathcal{J}_{i,k}|\cdot Y_{i,k}\end{aligned}\tag{9}$$

Next, the CSP verifies the validity of $F_{i,j,k}$ by checking whether $P_{i,k}\neq P_{j,k}$ and

$$\begin{aligned}&(\mathcal{F}_{i,j,k}-\mathcal{H}(k||v_k||P_{i,k}) \bmod q)\cdot G\\=&\mathcal{H}(C_{i,k}||P_{j,k}||e_{i,j,k}||Y_{i,k})\cdot P_{i,k}+Y_{i,k}\end{aligned}\tag{6}$$

hold. The correctness proof of Eq. (6) is shown as Eq. (7).

If the above verifications pass, the CSP considers $F_{i,j,k}$ as valid and stores it locally; otherwise, the CSP drops it directly.

In addition to the one-by-one verifications in Eq. (4) and Eq. (6), the CSP can also perform two batch verifications at the end of each T_k. Specifically, for multiple reputation feedbacks in T_k, whose set is denoted as $\{F_{i,j,k}\}$ (in which, without loss of generality, we assume that $P_{i,k}\in\{P_{i_1,k},P_{i_2,k},...\}\triangleq\mathcal{I}_k$ and $P_{j,k}\in\{P_{j_1,k},P_{j_2,k},...\}\triangleq\mathcal{J}_k$, where $\mathcal{I}_k\neq\emptyset$ and $\mathcal{J}_k\neq\emptyset$), the CSP first derives the corresponding feedback target set $\{P_{j_{i,1},k},P_{j_{i,2},k},...\}\triangleq\mathcal{J}_{i,k}$ for each $P_{i,k}\in\mathcal{I}_k$ (where $\mathcal{J}_{i,k}\subseteq\mathcal{J}_k$ and $\mathcal{J}_{i,k}\neq\emptyset$), and then performs two batch verifications as Eq. (8) and Eq. (9), where $|\mathcal{J}_{i,k}|$ denotes the number of elements in $\mathcal{J}_{i,k}$. The correctness proofs of Eq. (8) and Eq. (9) are shown as Eq. (10) and Eq. (11), respectively.

$$\begin{aligned}&\sum_{P_{i,k}\in\mathcal{I}_k}(C^1_{i,k})^{v_k}\cdot g^{\mathcal{H}(k||v_k||P_{i,k})} \bmod n^2\\=&(\sum_{P_{i,k}\in\mathcal{I}_k}C^2_{i,k})\cdot\mathcal{H}(k||v_k)^{n\cdot v_k} \bmod n^2\end{aligned}\tag{8}$$

$$\begin{aligned}&\sum_{P_{i,k}\in\mathcal{I}_k}(C^1_{i,k})^{v_k}\cdot g^{\mathcal{H}(k||v_k||P_{i,k})} \bmod n^2\\\overset{(5)}{=}&\sum_{P_{i,k}\in\mathcal{I}_k}C^2_{i,k}\cdot\mathcal{H}(k||v_k)^{n\cdot v_k} \bmod n^2\\=&(\sum_{P_{i,k}\in\mathcal{I}_k}C^2_{i,k})\cdot\mathcal{H}(k||v_k)^{n\cdot v_k} \bmod n^2\end{aligned}\tag{10}$$

Furthermore, at the end of each T_k, for each $P_{j,k}\in\mathcal{J}_k$, the CSP first derives the corresponding feedback provider set $\{P_{i_{j,1},k},P_{i_{j,2},k},...\}\triangleq\mathcal{I}_{j,k}$ (where $\mathcal{I}_{j,k}\subseteq\mathcal{I}_k$ and $\mathcal{I}_{j,k}\neq\emptyset$), and then calculates two aggregated ciphertexts $D^1_{j,k}$ and $D^2_{j,k}$ as Eq. (12) and Eq. (13), respectively.

Afterwards, the CSP sends an aggregated feedback $A_k=(k,\{(P_{j,k},D^1_{j,k},D^2_{j,k},|\mathcal{I}_{j,k}|)|P_{j,k}\in\mathcal{J}_k\})$ to the TA via a secure wired link, where $|\mathcal{I}_{j,k}|$ denotes the number of elements in $\mathcal{I}_{j,k}$.

F. Aggregated Feedback Verification and Reputation Updating

After receiving A_k, the TA first extracts k from A_k and derives the current time interval's serial number k''' from its clock, and then verifies the timeliness of A_k by checking whether $k=k'''$ holds. Next, for each $P_{j,k}\in\mathcal{J}_k$, the TA derives the unique identifier j corresponding to $P_{j,k}$ by retrieving the secure database, and calculates V_j's incremental reputation value $R^I_{j,k}$ in T_k as

$$R^I_{j,k}=\mathcal{R}(\frac{\mathcal{D}_P(D^2_{j,k},\lambda,\mu)}{\mathcal{D}_P(D^1_{j,k},\lambda,\mu)})\tag{14}$$

As described earlier, in the PPRU scheme, $\eta\in Z^*_n$, $5<\eta\ll n$, and $\mathcal{I}_{j,k}\neq\emptyset$, thus $1\leq|\mathcal{I}_{j,k}|<\frac{n}{\eta^2}$ always holds in a practical vehicular network. Besides, in the PPRU scheme, $f_{i,j,k}\in Z_\eta$, and we assume $R_{i,k}\in Z_\eta$ holds for each vehicle and each $k\in\{1,2,...\}$ (which is also named as *Assumption-I* for ease of later illustration, and we will prove it in the subsequent analysis), thus Eq. (15) holds. That is, $\sum_{P_{i,k}\in\mathcal{I}_{j,k}}R_{i,k}\in Z_n$ and $\sum_{P_{i,k}\in\mathcal{I}_{j,k}}R_{i,k}\cdot f_{i,j,k}\in Z_n$. As a result, the correctness of Eq. (14) can be proved as

$$\begin{aligned}R^I_{j,k}=&\mathcal{R}(\frac{\mathcal{D}_P(D^2_{j,k},\lambda,\mu)}{\mathcal{D}_P(D^1_{j,k},\lambda,\mu)})\\=&\mathcal{R}(\frac{\sum_{P_{i,k}\in\mathcal{I}_{j,k}}R_{i,k}\cdot f_{i,j,k} \bmod n}{\sum_{P_{i,k}\in\mathcal{I}_{j,k}}R_{i,k} \bmod n})\\=&\mathcal{R}(\frac{\sum_{P_{i,k}\in\mathcal{I}_{j,k}}R_{i,k}\cdot f_{i,j,k}}{\sum_{P_{i,k}\in\mathcal{I}_{j,k}}R_{i,k}})\end{aligned}\tag{16}$$

That is, $R^I_{j,k}$ is essentially calculated as the weighted average of the corresponding feedback providers' feedback scores for V_j, where the corresponding feedback providers' reputation values are adopted as important weights. In addition, we can easily find that $R^I_{j,k}\in Z_\eta$. Next, the TA calculates V_j's reputation value in T_{k+1} as

$$R_{j,k+1}=\mathcal{R}(\omega_{j,k}\cdot R_{j,k}+(1-\omega_{j,k})\cdot R^I_{j,k})\tag{17}$$

where $\omega_{j,k}$ is a weight value and is defined as a function of $|\mathcal{I}_{j,k}|$, namely $\omega_{j,k}=e^{-\varepsilon\cdot|\mathcal{I}_{j,k}|}$, in which $|\mathcal{I}_{j,k}|\in\{1,2,...\}$, and ε is a control factor in the range of $(0,1)$. We can easily find that $\omega_{j,k}\in(0,1)$. Specifically, the larger $|\mathcal{I}_{j,k}|$ is, the

$$\begin{aligned}
&(\sum_{P_{i,k}\in\mathcal{I}_k}(\sum_{P_{j,k}\in\mathcal{J}_{i,k}}\mathcal{F}_{i,j,k}) - |\mathcal{J}_{i,k}|\cdot\mathcal{H}(k||v_k||P_{i,k}) \bmod q)\cdot G\\
=&(\sum_{P_{i,k}\in\mathcal{I}_k}(\sum_{P_{j,k}\in\mathcal{J}_{i,k}}\mathcal{F}_{i,j,k} - \mathcal{H}(k||v_k||P_{i,k})) \bmod q)\cdot G\\
=&\sum_{P_{i,k}\in\mathcal{I}_k}\sum_{P_{j,k}\in\mathcal{J}_{i,k}}(\mathcal{F}_{i,j,k} - \mathcal{H}(k||v_k||P_{i,k}) \bmod q)\cdot G\\
\overset{(7)}{=}&\sum_{P_{i,k}\in\mathcal{I}_k}\sum_{P_{j,k}\in\mathcal{J}_{i,k}}\mathcal{H}(C_{i,k}||P_{j,k}||e_{i,j,k}||Y_{i,k})\cdot P_{i,k} + Y_{i,k}\\
=&\sum_{P_{i,k}\in\mathcal{I}_k}(\sum_{P_{j,k}\in\mathcal{J}_{i,k}}\mathcal{H}(C_{i,k}||P_{j,k}||e_{i,j,k}||Y_{i,k}) \bmod q)\cdot P_{i,k} + |\mathcal{J}_{i,k}|\cdot Y_{i,k}
\end{aligned} \tag{11}$$

$$\begin{aligned}
D^1_{j,k} =& \prod_{P_{i,k}\in\mathcal{I}_{j,k}} C^1_{i,k} \bmod n^2\\
=& \prod_{P_{i,k}\in\mathcal{I}_{j,k}} g^{R_{i,k}}\cdot(r'_{i,k}\cdot\mathcal{H}(k||v_k))^n \bmod n^2\\
=& \prod_{P_{i,k}\in\mathcal{I}_{j,k}} g^{R_{i,k}}\cdot\prod_{P_{i,k}\in\mathcal{I}_{j,k}}(r'_{i,k}\cdot\mathcal{H}(k||v_k))^n \bmod n^2\\
=& g^{\sum_{P_{i,k}\in\mathcal{I}_{j,k}} R_{i,k}}\cdot(\prod_{P_{i,k}\in\mathcal{I}_{j,k}} r'_{i,k}\cdot\mathcal{H}(k||v_k))^n \bmod n^2
\end{aligned} \tag{12}$$

$$\begin{aligned}
D^2_{j,k} =& \prod_{P_{i,k}\in\mathcal{I}_{j,k}} e_{i,j,k} \bmod n^2\\
=& \prod_{P_{i,k}\in\mathcal{I}_{j,k}} g^{R_{i,k}\cdot f_{i,j,k}}\cdot((r'_{i,k}\cdot\mathcal{H}(k||v_k))^{f_{i,j,k}}\cdot r''_{i,j,k})^n \bmod n^2\\
=& \prod_{P_{i,k}\in\mathcal{I}_{j,k}} g^{R_{i,k}\cdot f_{i,j,k}}\cdot\prod_{P_{i,k}\in\mathcal{I}_{j,k}}((r'_{i,k}\cdot\mathcal{H}(k||v_k))^{f_{i,j,k}}\cdot r''_{i,j,k})^n \bmod n^2\\
=& g^{\sum_{P_{i,k}\in\mathcal{I}_{j,k}} R_{i,k}\cdot f_{i,j,k}}\cdot(\prod_{P_{i,k}\in\mathcal{I}_{j,k}}(r'_{i,k}\cdot\mathcal{H}(k||v_k))^{f_{i,j,k}}\cdot r''_{i,j,k})^n \bmod n^2
\end{aligned} \tag{13}$$

$$\begin{cases}
0 \le \sum_{P_{i,k}\in\mathcal{I}_{j,k}} R_{i,k} < \sum_{P_{i,k}\in\mathcal{I}_{j,k}} \eta = |\mathcal{I}_{j,k}|\cdot\eta < \frac{n}{\eta^2}\cdot\eta = \frac{n}{\eta} < n\\
0 \le \sum_{P_{i,k}\in\mathcal{I}_{j,k}} R_{i,k}\cdot f_{i,j,k} < \sum_{P_{i,k}\in\mathcal{I}_{j,k}} \eta\cdot\eta = |\mathcal{I}_{j,k}|\cdot\eta\cdot\eta < \frac{n}{\eta^2}\cdot\eta\cdot\eta = n
\end{cases} \tag{15}$$

closer $\omega_{j,k}$ is to 0, and the larger weight $R^I_{j,k}$ has in Eq. (17); otherwise, the closer $\omega_{j,k}$ is to 1, and the larger weight $R_{j,k}$ has in Eq. (17).

In addition, for each $P_{j,k} \notin \mathcal{J}_k$ (i.e., there is no reputation feedback for V_j being submitted in T_k), the TA calculates V_j's reputation value $R_{j,k+1}$ in T_{k+1} as

$$R_{j,k+1} = \mathcal{R}(\xi\cdot R_{j,k}) \tag{18}$$

where ξ denotes a decay factor in the range of $(0, 1)$. Moreover, Eq. (17) and Eq. (18) can be combined as Eq. (19), and we can easily find that $R_{j,k+1} \in Z_\eta$.

From the calculations in Eq. (12) - Eq. (19), we can easily find that as long as the reputation value in T_k of each vehicle falls in Z_η, the calculated reputation value of each vehicle in T_{k+1} will also fall in Z_η, where $k \in \{1, 2, ...\}$. Meanwhile, as described in Section V.*B*, the initial reputation value of each vehicle belongs to Z_η (Specifically, for any a vehicle marked as V_i, if it registers with the TA in T_k, then $R_{i,k} = R_{i,0} \in Z_\eta$, where $k \in \{1, 2, ...\}$). Thus, we can easily find that the *Assumption-I* holds based on the classic mathematical induction method [38].

Moreover, after calculating the reputation value of each vehicle in T_{k+1} based on Eq. (19), the TA updates the reputation value of each vehicle in the secure database.

VI. Theoretical Analysis

In this section, we present the detailed theoretical analysis for the strong privacy preservation, strong security, robust reputation management, acceptable computation overhead, and acceptable communication overhead in the PPRU scheme, respectively.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2023.3340723

$$R_{j,k+1} = \begin{cases} \mathcal{R}(\omega_{j,k} \cdot R_{j,k} + (1-\omega_{j,k}) \cdot R^I_{j,k}), \text{ if } P_{j,k} \in \mathcal{J}_k \\ \mathcal{R}(\xi \cdot R_{j,k}), \text{ otherwise} \end{cases} \tag{19}$$

A. Strong Privacy Preservation

In this part, we mainly analyze the strong privacy preservation capability of the PPRU scheme for the unique identifier, reputation value, and feedback score of each vehicle in the reputation feedback submitting and aggregation processes.

Firstly, in the PPRU scheme, the pseudonyms $P_{i,k}$ and $P_{j,k}$ (instead of the unique identifiers i and j) of feedback provider V_i and feedback target V_j are included in the reputation feedback $F_{i,j,k}$. Without knowing the correspondence between unique identifiers and pseudonyms, the adversary cannot reveal i and j from $F_{i,j,k}$. Meanwhile, the pseudonyms in different time intervals corresponding to the same unique identifier are different. As a result, the unique identifier of each vehicle cannot be revealed or linked for a long time by the adversary in the reputation feedback submitting and aggregation processes.

Besides, in the PPRU scheme, the Paillier ciphertext $C^1_{i,k}$ (instead of the plaintext) of reputation value $R_{i,k}$ is included in the reputation feedback $F_{i,j,k}$. According to the properties of Paillier algorithm [15], the adversary cannot reveal $R_{i,k}$ from $F_{i,j,k}$, since it does not own the Paillier private key (λ, μ). Meanwhile, the Paillier ciphertexts in different time intervals of the same reputation value are different, due to the adoption of $r'_{i,k}$ and $\mathcal{H}(k||v_k)$ in Eq. (2). As a result, the reputation value of each vehicle cannot be revealed or linked by the adversary in the reputation feedback submitting and aggregation processes.

Similarly, in the PPRU scheme, the Paillier ciphertext $e_{i,j,k}$ (instead of the plaintext) of feedback score $f_{i,j,k}$ is included in the reputation feedback $F_{i,j,k}$. According to the properties of Paillier algorithm [15], the adversary cannot reveal $f_{i,j,k}$ from $F_{i,j,k}$, since it does not own the Paillier private key (λ, μ). Meanwhile, the Paillier ciphertexts in different time intervals of the same feedback score are different, due to the adoption of $r'_{i,k}$, $\mathcal{H}(k||v_k)$, and $r''_{i,j,k}$ in Eq. (3). As a result, the feedback score of each vehicle cannot be revealed or linked by the adversary in the reputation feedback submitting and aggregation processes.

B. Strong Security

In this part, we mainly demonstrate the strong security of the PPRU scheme against multiple kinds of common attacks, including the forgery, replay, Sybil, self-praise, and tampering attacks in the reputation feedback submitting and aggregation processes. The detailed analysis is as follows.

Theorem 1: The PPRU scheme is resistant to the reputation value forgery attack.

Proof: In the PPRU scheme, a malicious feedback provider V_i may conduct the reputation value forgery attack (i.e., forge its reputation value $R_{i,k}$ in $C^1_{i,k}$) to gain higher weight in the reputation updating process. Firstly, we prove that the one-by-one verification in Eq. (4) is resistant to the reputation value forgery attack. Specifically, we assume that V_i can forge $R_{i,k}$ as $R^*_{i,k} = R_{i,k} + \Delta R_{i,k}$, where $\Delta R_{i,k} \in Z_\eta$. That is, it can forge $C^1_{i,k}$ as $C^{1*}_{i,k}$, where

$$\begin{aligned} &C^{1*}_{i,k} \\ =&C^1_{i,k} \cdot g^{\Delta R_{i,k}} \bmod n^2 \\ =&g^{R_{i,k}+\Delta R_{i,k}} \cdot (r'_{i,k} \cdot \mathcal{H}(k||v_k))^n \bmod n^2 \end{aligned} \tag{20}$$

Accordingly, to enable the forged $C^{1*}_{i,k}$ to pass the verification in Eq. (4), V_i needs to forge $C^2_{i,k}$ as $C^{2*}_{i,k}$, in which

$$\begin{aligned} &C^{2*}_{i,k} \\ =&C^2_{i,k} \cdot g^{\Delta R'_{i,k}} \bmod n^2 \\ =&g^{R_{i,k} \cdot v_k + x^2_{i,k} + \Delta R'_{i,k}} \cdot (r'_{i,k})^{n \cdot v_k} \bmod n^2 \end{aligned} \tag{21}$$

where $\Delta R'_{i,k} \in Z$. To conduct the reputation value forgery attack successfully, $C^{1*}_{i,k}$ and $C^{2*}_{i,k}$ should be able to pass the verification in Eq. (4), namely

$$\begin{aligned} &(C^{1*}_{i,k})^{v_k} \cdot g^{\mathcal{H}(k||v_k||P_{i,k})} \bmod n^2 \\ =&C^{2*}_{i,k} \cdot \mathcal{H}(k||v_k)^{n \cdot v_k} \bmod n^2 \end{aligned} \tag{22}$$

By combining Eq. (4) and Eq. (22), we can easily derive

$$g^{\Delta R_{i,k} \cdot v_k} = g^{\Delta R'_{i,k}} \bmod n^2 \tag{23}$$

Based on the Carmichael theorem [39], we can further derive

$$\Delta R_{i,k} \cdot v_k - \Delta R'_{i,k} = \kappa \cdot \lambda(n^2) \tag{24}$$

where $\kappa \in Z$, $\lambda(n^2) = lcm(p' \cdot (p'-1), q' \cdot (q'-1))$, and $n = p' \cdot q'$. Due the difficulty of factoring a large integer [34], V_i cannot derive p', q', and $\lambda(n^2)$ from n. Meanwhile, v_k is also unknown to V_i. As a result, except for setting $\Delta R_{i,k} = 0$, $\Delta R'_{i,k} = 0$, and $\kappa = 0$ (i.e., without forging $R_{i,k}$), V_i cannot effectively set $\Delta R_{i,k}$, $\Delta R'_{i,k}$, and κ such that Eq. (24) holds. Thus, the one-by-one verification in Eq. (4) is resistant to the reputation value forgery attack. Similarly, we can easily prove that the batch verification in Eq. (8) is also resistant to the reputation value forgery attack.

Theorem 2: The PPRU scheme is resistant to the feedback provider pseudonym forgery attack.

Proof: In the PPRU scheme, a malicious feedback provider V_i may conduct the feedback provider pseudonym forgery attack (i.e., forge its pseudonym $P_{i,k}$ which is the first part of $C_{i,k}$) to disrupt the normal reputation updating. Firstly, we prove that the one-by-one verification in Eq. (4) is resistant to the feedback provider pseudonym forgery attack. Specifically, we assume that V_i can forge $P_{i,k}$ as $P^*_{i,k}$. To conduct the feedback provider pseudonym forgery attack successfully, $P^*_{i,k}$ should be able to pass the verification in Eq. (4), namely

$$\begin{aligned} &(C^1_{i,k})^{v_k} \cdot g^{\mathcal{H}(k||v_k||P^*_{i,k})} \bmod n^2 \\ =&C^2_{i,k} \cdot \mathcal{H}(k||v_k)^{n \cdot v_k} \bmod n^2 \end{aligned} \tag{25}$$

By combining Eq. (4) and Eq. (25), we can easily derive

$$g^{\mathcal{H}(k||v_k||P^*_{i,k})} = g^{\mathcal{H}(k||v_k||P_{i,k})} \bmod n^2 \tag{26}$$

Based on the Carmichael theorem [39], we can further derive

$$\mathcal{H}(k||v_k||P^*_{i,k}) - \mathcal{H}(k||v_k||P_{i,k}) = \kappa' \cdot \lambda(n^2) \quad (27)$$

where $\kappa' \in Z$. Similar to the previous analysis, V_i cannot obtain v_k and $\lambda(n^2)$. Besides, $\mathcal{H}()$ is irreversible. As a result, V_i cannot effectively set $P^*_{i,k}$ such that Eq. (27) holds. Thus, the one-by-one verification in Eq. (4) is resistant to the feedback provider pseudonym forgery attack. Similarly, we can easily prove that the batch verification in Eq. (8) is also resistant to the feedback provider pseudonym forgery attack.

Theorem 3: The PPRU scheme is resistant to the replay attack.

Proof: In the PPRU scheme, the adversary may conduct the replay attack by utilizing an outdated reputation certificate $C_{i,k}$ (containing an outdated k) or by modifying the outdated k (i.e., the second part of $C_{i,k}$). However, as detailed in Section V.*E*, a reputation feedback $F_{i,j,k}$ with an outdated $C_{i,k}$ cannot pass the timeliness verification of CSP (i.e., checking whether $k = k''$). Next, we prove that the one-by-one verification in Eq. (4) is resistant to the replay attack by modifying the k. Specifically, we assume the adversary can modify the outdated k as the timely k^*. Accordingly, the CSP will pick the secret value v_{k^*} in T_{k^*} (instead of v_k) to perform the verification in Eq. (4). To conduct the replay attack successfully, k^* should be able to pass the verification in Eq. (4), namely

$$\begin{aligned}&(C^1_{i,k})^{v_{k^*}} \cdot g^{\mathcal{H}(k^*||v_{k^*}||P_{i,k})} \bmod n^2\\ =&C^2_{i,k} \cdot \mathcal{H}(k^*||v_{k^*})^{n \cdot v_{k^*}} \bmod n^2\end{aligned} \quad (28)$$

By combining Eq. (4) and Eq. (28), we can easily derive

$$\begin{aligned}&(C^1_{i,k})^{v_{k^*} - v_k} \cdot g^{\mathcal{H}(k^*||v_{k^*}||P_{i,k}) - \mathcal{H}(k||v_k||P_{i,k})} \bmod n^2\\ =&\mathcal{H}(k^*||v_{k^*})^{n \cdot v_{k^*}} \cdot \mathcal{H}(k||v_k)^{-n \cdot v_k} \bmod n^2\end{aligned} \quad (29)$$

Similar to the previous analysis, the adversary cannot obtain v_{k^*} and v_k. Besides, $\mathcal{H}()$ is irreversible. As a result, the adversary cannot effectively set k^* such that Eq. (29) holds. Thus, the one-by-one verification in Eq. (4) is resistant to the replay attack. Similarly, we can easily prove that the batch verification in Eq. (8) is also resistant to the replay attack.

Theorem 4: The PPRU scheme is resistant to the Sybil attack.

Proof: In the PPRU scheme, the adversary may conduct the Sybil attack by requesting the TA for multiple pseudonyms or by forging multiple pseudonyms in a time internal T_k. However, as detailed in Section V.*C*, in the PPRU scheme, each vehicle (e.g., V_i) can merely obtain a reputation certificate $C_{i,k}$ (containing a pseudonym $P_{i,k}$) for each T_k even though it requests the TA for multiple times. Besides, as analyzed in Theorem 2, V_i cannot effectively set the other pseudonyms except for $P_{i,k}$ in each time interval T_k. As a result, V_i cannot effectively submit multiple reputation feedbacks for the same feedback target in a short period of time by adopting multiple pseudonyms. Thus, the PPRU scheme is resistant to the Sybil attack.

Theorem 5: The PPRU scheme is resistant to the self-praise attack.

Proof: In the PPRU scheme, the adversary may conduct the self-praise attack by setting $P_{j,k} = P_{i,k}$ in the reputation feedback $F_{i,j,k}$ or by forging multiple pseudonyms in a time internal T_k. However, as detailed in Section V.*E*, each reputation feedback (e.g., $F_{i,j,k}$) for self-praise (where $P_{i,k} = P_{j,k}$) cannot pass the rationality verification of the CSP (i.e., checking whether $P_{i,k} \neq P_{j,k}$). Besides, as analyzed in Theorem 2, each vehicle (e.g., V_i) cannot effectively set the other pseudonyms except for $P_{i,k}$ for self-praise in each time interval T_k. Thus, the PPRU scheme is resistant to the self-praise attack.

Theorem 6: The PPRU scheme is resistant to the feedback score tampering attack.

Proof: In the PPRU scheme, the adversary may conduct the feedback score tampering attack (i.e., tamper with the feedback score $f_{i,j,k}$ which is contained in the encrypted feedback score $e_{i,j,k}$) to disrupt the normal reputation updating. Firstly, we prove that the one-by-one verification in Eq. (6) is resistant to the feedback score tampering attack. Specifically, we assume that the adversary can tamper with $f_{i,j,k}$ to $f^*_{i,j,k} = f_{i,j,k} + \Delta f_{i,j,k}$. That is, it can tamper with $e_{i,j,k}$ to $e^*_{i,j,k}$, where

$$\begin{aligned}&e^*_{i,j,k}\\ =&e_{i,j,k} \cdot (C^1_{i,k})^{\Delta f_{i,j,k}}\\ =&(C^1_{i,k})^{f_{i,j,k} + \Delta f_{i,j,k}} \cdot (r''_{i,j,k})^n \bmod n^2\end{aligned} \quad (30)$$

To conduct the feedback score tampering attack successfully, the adversary needs to tamper with the signature $\mathcal{F}_{i,j,k}$ to $\mathcal{F}^*_{i,j,k}$, where $e^*_{i,j,k}$ and $\mathcal{F}^*_{i,j,k}$ should be able to pass the verification in Eq. (6), namely

$$\begin{aligned}&(\mathcal{F}^*_{i,j,k} - \mathcal{H}(k||v_k||P_{i,k}) \bmod q) \cdot G\\ =&\mathcal{H}(C_{i,k}||P_{j,k}||e^*_{i,j,k}||Y_{i,k}) \cdot P_{i,k} + Y_{i,k}\end{aligned} \quad (31)$$

By combining Eq. (6) and Eq. (31), we can easily derive

$$\begin{aligned}&(\mathcal{F}^*_{i,j,k} - \mathcal{F}_{i,j,k}) \cdot G\\ =&(h^*_{i,j,k} - h_{i,j,k}) \cdot P_{i,k}\\ =&(h^*_{i,j,k} - h_{i,j,k}) \cdot x^1_{i,k} \cdot G\end{aligned} \quad (32)$$

where "mod q" is omitted for ease of expression, $h^*_{i,j,k} = \mathcal{H}(C_{i,k}||P_{j,k}||e^*_{i,j,k}||Y_{i,k})$, $h_{i,j,k} = \mathcal{H}(C_{i,k}||P_{j,k}||e_{i,j,k}||Y_{i,k})$. Thus,

$$(\mathcal{F}^*_{i,j,k} - \mathcal{F}_{i,j,k}) \cdot (h^*_{i,j,k} - h_{i,j,k})^{-1} \cdot G = x^1_{i,k} \cdot G \quad (33)$$

As a result, the adversary can output $(\mathcal{F}^*_{i,j,k} - \mathcal{F}_{i,j,k}) \cdot (h^*_{i,j,k} - h_{i,j,k})^{-1}$ as a solution of deriving $x^1_{i,k}$ from $x^1_{i,k} \cdot G$. That is, the probability that the adversary solves the ECDLP problem is obviously non-negligible, which is contradictory to the difficulty of ECDLP problem [35]. Thus, the one-by-one verification in Eq. (6) is resistant to the feedback score tampering attack. Similarly, we can easily prove that the batch verification in Eq. (9) is also resistant to the feedback score tampering attack.

Theorem 7: The PPRU scheme is resistant to the feedback target pseudonym tampering attack.

Proof: In the PPRU scheme, the adversary may conduct the feedback target pseudonym tampering attack (i.e., tamper with the feedback target's pseudonym $P_{j,k}$ in the reputation feedback $F_{i,j,k}$) to disrupt the normal reputation updating. Firstly, we prove that the one-by-one verification in Eq. (6) is resistant to the feedback target pseudonym tampering attack. Specifically, we assume that the adversary can tamper with $P_{j,k}$ to $P_{j,k}^{**}$. To conduct the feedback target pseudonym tampering attack successfully, the adversary needs to tamper with the signature $\mathcal{F}_{i,j,k}$ to $\mathcal{F}_{i,j,k}^{**}$, where $P_{j,k}^{**}$ and $\mathcal{F}_{i,j,k}^{**}$ should be able to pass the verification in Eq. (6), namely

$$\begin{aligned}&(\mathcal{F}_{i,j,k}^{**}-\mathcal{H}(k||v_k||P_{i,k}) \bmod q)\cdot G\\=&\mathcal{H}(C_{i,k}||P_{j,k}^{**}||e_{i,j,k}||Y_{i,k})\cdot P_{i,k}+Y_{i,k}\end{aligned} \tag{34}$$

By the similar analysis to that in Theorem 6, we can easily conclude that the one-by-one verification in Eq. (6) is resistant to the feedback target pseudonym tampering attack. Similarly, we can easily prove that the batch verification in Eq. (9) is also resistant to the feedback target pseudonym tampering attack.

C. Robust Reputation Management

As revealed in Eq. (16), the incremental reputation value of each vehicle is essentially calculated as the weighted average of the corresponding feedback providers' feedback scores, where the corresponding feedback providers' reputation values are adopted as important weights. As demonstrated in many recent researches [5], [14], [31], the weighted average can provide obviously stronger robustness against malicious feedback providers than the simple average, and the quantitative robustness evaluation is shown in Section VII.*A*.

D. Acceptable Computation Overhead

In this part, we mainly analyze the computation overheads of vehicles, CSP, and TA in the reputation feedback submitting and aggregation processes. For ease of expression, we define $|\mathcal{I}_k|=u$, $|\mathcal{J}_k|=u'$, and define the average values of $|\mathcal{J}_{i,k}|$ and $|\mathcal{I}_{j,k}|$ as v and v', respectively. The detailed analysis is as follows.

When a vehicle (e.g., V_i) is to generate a reputation feedback $F_{i,j,k}$ for another vehicle (e.g., V_j), it first needs to generate an encrypted feedback score $e_{i,j,k}$, which requires to perform two Paillier modular exponential operations and one Paillier modular multiplication operation, and then needs to generate a signature $\mathcal{F}_{i,j,k}$, which requires to perform one hash operation, one ECC modular multiplication operation, and two ECC modular addition operations. Besides, it needs to compute $Y_{i,k}$ for once (even it is to generate multiple reputation feedbacks) in each T_k, which requires to perform one ECC point multiplication operation. Specifically, we denote the computation overheads of conducting one Paillier modular exponential operation, one Paillier modular multiplication operation, one hash operation, one ECC modular multiplication operation, one ECC modular addition operation, and one ECC point multiplication operation as T_{exp}^p, T_{mul}^p, T_{hash}, T_{mul}^e, T_{add}^e, and T_{pmul}^e, respectively, and then the total computation overhead on the vehicles side in the reputation feedback submitting and aggregation processes can be approximately calculated as $u\cdot v\cdot(2T_{exp}^p+T_{mul}^p+T_{hash}+T_{mul}^e+2T_{add}^e)+u\cdot T_{pmul}^e$.

After receiving the reputation feedbacks in each T_k, the CSP first needs to verify the validity of reputation certificates and reputation feedbacks by adopting the one-by-one verification or batch verification.

- When the one-by-one verification is adopted, for verifying each reputation feedback, the CSP needs to perform three Paillier modular exponential operations, three Paillier modular multiplication operations, four hash operations, two ECC point multiplication operations, one ECC point addition operation, and one ECC modular subtraction operation.
- When the batch verification is adopted, for verifying $u\cdot v$ reputation feedbacks, the CSP needs to perform $2(u+1)$ Paillier modular exponential operations, $u+2$ Paillier modular multiplication operations, $2u-2$ Paillier modular addition operations, $2u+u\cdot v+1$ hash operations, u ECC modular multiplication operations, $2u\cdot v-u-1$ ECC modular addition operations, u ECC modular subtraction operations, $2u+1$ ECC point multiplication operations, and $2u-1$ ECC point addition operations.

Then, the CSP needs to generate an aggregated feedback A_k, which requires to perform $2u'\cdot(v'-1)$ Paillier modular multiplication operations. Specifically, we denote the computation overheads of conducting one ECC point addition operation, one ECC modular subtraction operation, and one Paillier modular addition operation as T_{padd}^e, T_{sub}^e, and T_{add}^p, respectively, and then we can derive the following conclusions.

- When the one-by-one verification is adopted, the total computation overhead on the CSP side in the reputation feedback submitting and aggregation processes can be approximately calculated as $u\cdot v\cdot(3T_{exp}^p+3T_{mul}^p+4T_{hash}+2T_{pmul}^e+T_{padd}^e+T_{sub}^e)+2u'\cdot(v'-1)\cdot T_{mul}^p$.
- When the batch verification is adopted, the total computation overhead on the CSP side in the reputation feedback submitting and aggregation processes can be approximately calculated as $2(u+1)\cdot T_{exp}^p+(u+2+2u'\cdot(v'-1))\cdot T_{mul}^p+(2u-2)\cdot T_{add}^p+(2u+u\cdot v+1)\cdot T_{hash}+u\cdot T_{mul}^e+(2u\cdot v-u-1)\cdot T_{add}^e+u\cdot T_{sub}^e+(2u+1)\cdot T_{pmul}^e+(2u-1)\cdot T_{padd}^e$.

After receiving the aggregated feedback A_k, the TA needs to conduct two Paillier decryption operations to calculate the incremental reputation value of each vehicle in $\mathcal{J}_k$, which requires to perform $2u'$ Paillier modular exponential operations and $2u'$ Paillier modular multiplication operations. Note that the computation overheads of other operations are negligible when compared with those of Paillier-based operations, thus the total computation overhead on the TA side in the reputation feedback submitting and aggregation processes can be approximately calculated as $2u'\cdot(T_{exp}^p+T_{mul}^p)$.

As revealed in Table I, among the state-of-the-art schemes, the PPTM scheme [5] has the security, privacy, and trust properties closest to the PPRU scheme, thus we adopt it as a baseline in the theoretical analysis and simulation evaluation. To make a fair comparison between the PPRU and PPTM schemes, we assume that in the PPTM scheme, $Mc_{\mathcal{E},V_i}^{\alpha,\kappa}$, $Ts_{\mathcal{E},V_i}^{\alpha,\kappa}$, and $Ds_{\mathcal{E},V_i}^{\alpha,\kappa}$ are not contained in the reputation feedback $Rf_{\mathcal{E},V_i,V_j}^{\alpha,\kappa}$, and both the verification on $Ds_{\mathcal{E},V_i}^{\alpha,\kappa}$ and the acknowledgement for $Rf_{\mathcal{E},V_i,V_j}^{\alpha,\kappa}$ are omitted in the reputation feedback submitting process. Besides, we assume in the PPTM scheme, the encryption/decryption operations for reputation

TABLE III: Formalized computation overhead comparisons of vehicles, CSP, and TA in the PPRU and PPTM [5] schemes

Schemes	Vehicles	CSP	TA
PPRU-one	$u \cdot v \cdot (2T^p_{exp} + T^p_{mul} + T_{hash} + T^e_{mul} + 2T^e_{add}) + u \cdot T^e_{pmul}$	$u \cdot v \cdot (3T^p_{exp} + 3T^p_{mul} + 4T_{hash} + 2T^e_{pmul} + T^e_{padd} + T^e_{sub}) + 2u' \cdot (v' - 1) \cdot T^p_{mul}$	$2u' \cdot (T^p_{exp} + T^p_{mul})$
PPRU-batch	$u \cdot v \cdot (2T^p_{exp} + T^p_{mul} + T_{hash} + T^e_{mul} + 2T^e_{add}) + u \cdot T^e_{pmul}$	$2(u+1) \cdot T^p_{exp} + (u + 2 + 2u' \cdot (v'-1)) \cdot T^p_{mul} + (2u-2) \cdot T^p_{add} + (2u + u \cdot v + 1) \cdot T_{hash} + u \cdot T^e_{mul} + (2u \cdot v - u - 1) \cdot T^e_{add} + u \cdot T^e_{sub} + (2u+1) \cdot T^e_{pmul} + (2u-1) \cdot T^e_{padd}$	$2u' \cdot (T^p_{exp} + T^p_{mul})$
PPTM [5]	$u \cdot v \cdot (T_{hash} + T^e_{exp} + 2T^e_{mul} + T^e_{add} + 3T^e_{pmul} + T^e_{padd})$	0	$u \cdot v \cdot (T_{hash} + 2T^e_{exp} + 2T^e_{mul} + 3T^e_{pmul} + 2T^e_{padd})$

Note: PPRU-one and PPRU-batch denote the PPRU scheme with one-by-one verification and with batch verification, respectively.

feedbacks are realized by utilizing the ECC algorithm and the signature generation/verification operations for reputation feedbacks are realized by utilizing the ECDSA algorithm.

Thus, by the similar analysis to that in the PPRU scheme, we can easily derive that in the PPTM scheme, for $u \cdot v$ reputation feedbacks, the total computation overheads of vehicles, CSP, and TA in the reputation feedback submitting process can be approximately calculated as $u \cdot v \cdot (T_{hash} + T^e_{exp} + 2T^e_{mul} + T^e_{add} + 3T^e_{pmul} + T^e_{padd})$, 0, and $u \cdot v \cdot (T_{hash} + 2T^e_{exp} + 2T^e_{mul} + 3T^e_{pmul} + 2T^e_{padd})$, respectively, where T^e_{exp} denotes the computation overhead of conducting one ECC modular exponential operation. The formalized computation overhead comparisons of vehicles, CSP, and TA in the PPRU and PPTM schemes are shown in Table III, and the quantitative computation overhead comparisons are revealed in Section VII.*B*.

E. Acceptable Communication Overhead

In this part, we mainly analyze the communication overheads of vehicles, CSP, and TA in the reputation feedback submitting and aggregation processes. For the convenience of expression, we still define $|\mathcal{I}_k| = u$, $|\mathcal{J}_k| = u'$, and define the average values of $|\mathcal{J}_{i,k}|$ and $|\mathcal{I}_{j,k}|$ as v and v', respectively. Meanwhile, we set the parameters in the PPRU scheme as shown in Table IV, and then the detailed analysis is as follows.

TABLE IV: Parameter setting in the PPRU scheme

Parameters	Definitions	Bit lengths
$\|k\|$	Bit length of time interval's serial number k	32
$\|q\|$	Bit length of ECC parameter q	160
$\|S\|$	Bit length of ECC public key	320
$\|n^2\|$	Bit length of Paillier parameter n^2	2048
$\|v'\|$	Bit length of parameter $\|\mathcal{I}_{j,k}\|$	32

In the PPRU scheme, each reputation certificate (e.g., $C_{i,k}$) contains four parts, namely $P_{i,k}$, k, $C^1_{i,k}$, and $C^2_{i,k}$, whose bit lengths are $|S|$, $|k|$, $|n^2|$, and $|n^2|$, respectively, thus the bit length of $C_{i,k}$ can be calculated as $|S| + |k| + 2|n^2|$. Similarly, each reputation feedback (e.g., $F_{i,j,k}$) consists of five parts, namely $C_{i,k}$, $P_{j,k}$, $e_{i,j,k}$, $Y_{i,k}$, and $\mathcal{F}_{i,j,k}$, where the bit lengths of $P_{j,k}$, $e_{i,j,k}$, $Y_{i,k}$, and $\mathcal{F}_{i,j,k}$ are $|S|$, $|n^2|$, $|S|$, and $|q|$, respectively, thus the bit length of $F_{i,j,k}$ can be calculated as $3|S| + |k| + 3|n^2| + |q|$. Besides, each aggregated feedback (e.g., A_k) contains two parts, namely k and $\{(P_{j,k}, D^1_{j,k}, D^2_{j,k}, |\mathcal{I}_{j,k}|)|P_{j,k} \in \mathcal{J}_k\}$, where the bit lengths of k, $P_{j,k}$, $D^1_{j,k}$, $D^2_{j,k}$, and $|\mathcal{I}_{j,k}|$ are $|k|$, $|S|$, $|n^2|$, $|n^2|$, and $|v'|$, respectively, thus the bit length of A_k can be calculated as $|k| + u' \cdot (|S| + 2|n^2| + |v'|)$.

For each T_k, the vehicles need to submit $u \cdot v$ reputation feedbacks, the CSP needs to receives $u \cdot v$ reputation feedbacks and sends an aggregated feedback, and the TA needs to receive an aggregated feedback, thus the communication overheads on the vehicles, CSP, and TA sides can be calculated as $u \cdot v \cdot (3|S| + |k| + 3|n^2| + |q|)$, $u \cdot v \cdot (3|S| + |k| + 3|n^2| + |q|) + |k| + u' \cdot (|S| + 2|n^2| + |v'|)$, and $|k| + u' \cdot (|S| + 2|n^2| + |v'|)$, respectively.

As a contrast, in the PPTM scheme [5], based on the same assumptions as detailed in Section VI.*D*, the plaintext of each reputation certificate (e.g., $Rf^{\alpha,\kappa}_{\mathcal{E},V_i,V_j}$) contains five parts, namely α, $Ps^{\alpha,\kappa}_{V_i}$, j, $Fs^{\alpha,\kappa}_{\mathcal{E},V_i,V_j}$, and $Ds^{\alpha,\kappa}_{\mathcal{E},V_i,V_j}$, whose bit lengths are assumed to be $|k|$, $|k|$, $|k|$, 1, and $|S|$, respectively, thus the bit length of each reputation certificate (which is encrypted by utilizing the ECC algorithm) can be approximately calculated as $\lceil (3|k| + 1 + |S|)/|q| \rceil \cdot 2|S|$, where $\lceil * \rceil$ denotes ceiling function.

For each T_k, the vehicles need to submit $u \cdot v$ reputation feedbacks, the CSP is not involved, and the TA needs to receive $u \cdot v$ reputation feedbacks, thus the communication overheads on the vehicles, CSP, and TA sides can be calculated as $u \cdot v \cdot \lceil (3|k| + 1 + |S|)/|q| \rceil \cdot 2|S|$, 0, and $u \cdot v \cdot \lceil (3|k| + 1 + |S|)/|q| \rceil \cdot 2|S|$, respectively.

The formalized communication overhead comparisons on the vehicles, CSP, and TA sides in the PPRU and PPTM schemes are illustrated in Table V, and the quantitative communication overhead comparisons on the vehicles and TA sides in the PPRU and PPTM schemes are revealed in Fig. 2, where u varies from 1 to 1000, $u' = u$, and $v = v' = u \cdot 10\%$.

As revealed in Fig. 2(a), the communication overhead on the vehicles side in the PPRU scheme is slightly higher than that in the PPTM scheme. As revealed in Fig. 2(b), for the most values of u (i.e., $u \geq 22$), the communication overhead on the TA side in the PPRU scheme is significantly lower than that in the PPTM scheme, thus the PPRU scheme can dramatically reduce the communication overhead on the TA side when compared with the PPTM scheme. Specifically, when u varies from 1 to 1000, the average reducing percentage of the communication overhead on the TA side is about 83.88%.

This article has been accepted for publication in IEEE Transactions on Vehicular Technology. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVT.2023.3340723

TABLE V: Formalized communication overhead comparisons on the vehicles, CSP, and TA sides in the PPRU and PPTM [5] schemes

Schemes	Vehicles	CSP	TA
PPRU	$u \cdot v \cdot (3\|S\| + \|k\| + 3\|n^2\| + \|q\|)$	$u \cdot v \cdot (3\|S\| + \|k\| + 3\|n^2\| + \|q\|) +$ $\|k\| + u' \cdot (\|S\| + 2\|n^2\| + \|v'\|)$	$\|k\| + u' \cdot (\|S\| + 2\|n^2\| + \|v'\|)$
PPTM [5]	$u \cdot v \cdot \lceil (3\|k\| + 1 + \|S\|)/\|q\| \rceil \cdot 2\|S\|$	0	$u \cdot v \cdot \lceil (3\|k\| + 1 + \|S\|)/\|q\| \rceil \cdot 2\|S\|$

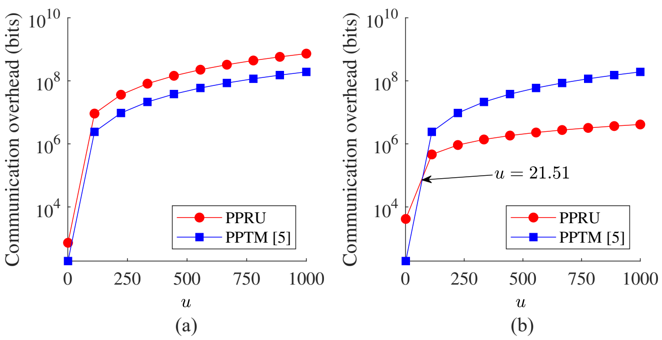


Fig. 2. Quantitative communication overhead comparisons on the (a) vehicles and (b) TA sides in the PPRU and PPTM [5] schemes.

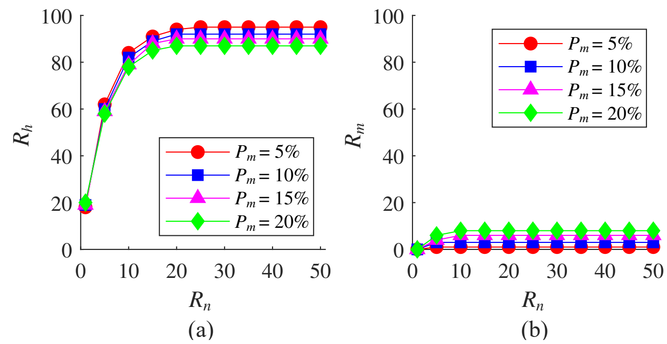


Fig. 3. Variation curve comparisons of (a) R_h and (b) R_m versus R_n when P_m takes different values.

VII. Simulation Evaluation

In this section, we conduct quantitative robustness evaluation for the reputation updating and quantitative computation overhead evaluation for the vehicles, CSP, and TA in the PPRU scheme on a laptop with the 11th Gen Intel Core i5-1135G7 CPU, 2.40GHz and 2.42GHz dual-core processors, 16G memory, and 64-bit Windows 10 operating system.

A. Robustness Evaluation

In this part, we first conduct quantitative robustness evaluation for the reputation updating in the PPRU scheme (where the weighted average is adopted) by adjusting the percentage of malicious vehicles and adopting two common evaluation indexes [5], [10], [14], [22], namely the average reputation value of honest vehicles (marked as R_h) and the average reputation value of malicious vehicles (marked as R_m). Specifically, we assume the total number of vehicles is 1000, and the percentages of law enforcement vehicles, public service vehicles, and private vehicles are 5%, 10%, and 85%, respectively, where the law enforcement vehicles and public service vehicles are honest, and the private vehicles may be malicious. Meanwhile, we assume that $u = u' = 1000$, $v = v' = u \cdot 10\% = 100$, and $\eta = 100$.

In addition, we adjust the malicious percentage of private vehicles (denoted as P_m) from 5% to 20%, and for each P_m, we carry out the reputation updating in the PPRU scheme for 50 rounds, where the serial number of each round (denoted as R_n) is 1, 2, ..., or 50. The above operations are repeated 1000 times for each P_m, and the average results are illustrated in Fig. 3.

As revealed in Fig. 3(a), for each P_m, in the first 20 or so rounds, the R_h continually increases, and the increasing speed of R_h decreases with the increase of P_m; in the subsequent rounds, the R_h basically remains stable (as a relatively high value), and the stable value of R_h decreases with the increase of P_m. As revealed in Fig. 3(b), for each P_m, in the first 10 or so rounds, the R_m continually increases, and the increasing speed of R_m increases with P_m; in the subsequent rounds, the R_m basically remains stable (as a relatively low value), and the stable value of R_m increases with P_m. Besides, for each P_m and each R_n, the R_h is significantly higher than the R_m, which indicates the PPRU scheme can provide robust reputation management.

As a contrast, based on the same assumptions as those in the above simulation, we also conduct quantitative robustness evaluation for the reputation updating in the PPRU scheme when the simple average, instead of the weighted average, is adopted, and the quantitative comparison results are shown in Fig. 4.

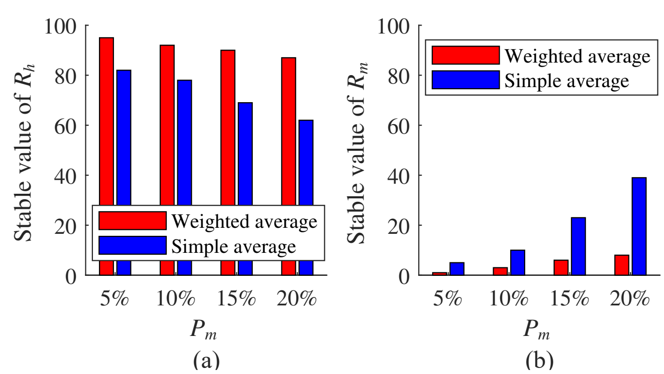


Fig. 4. Stable value comparisons of (a) R_h and (b) R_m when the weighted average and simple average are adopted for the reputation updating and P_m takes different values.

As revealed in Fig. 4(a), for each P_m, the stable value of R_h when the weighted average is adopted is obviously higher than that when the simple average is adopted, and the stable value difference of R_h when the weighted average and simple average are adopted increases with P_m. As revealed in Fig. 4(b), for each P_m, the stable value of R_m when the weighted average is adopted is obviously lower than that when the simple average is adopted, and the stable value difference of R_m when the weighted average and simple average are adopted increases with P_m. Besides, for each P_m, the stable value difference of R_h and R_m when the weighted average is adopted is obviously larger than that when the simple average is adopted, which indicates that the weighted average can provide obviously stronger robustness against malicious feedback providers than the simple average.

B. Computation Overhead Evaluation

In this part, we first measure the runtimes of various cryptographic operations in the PPRU and PPTM [5] schemes for 10^6 times based on the Java programming language[1] and Java Pairing-Based Cryptography (JPBC) library[2], respectively, and the average results are revealed in Table VI.

TABLE VI: Runtimes (ms) of various cryptographic operations in the PPRU and PPTM [5] schemes

Notations	Definitions	Runtimes
T^p_{exp}	Time of one Paillier modular exponential operation	3.956170
T^p_{mul}	Time of one Paillier modular multiplication operation	0.011560
T^p_{add}	Time of one Paillier modular addition operation	0.000690
T_{hash}	Time of one hash operation	0.000990
T^e_{exp}	Time of one ECC modular exponential operation	0.014890
T^e_{mul}	Time of one ECC modular multiplication operation	0.000410
T^e_{add}	Time of one ECC modular addition operation	0.000310
T^e_{sub}	Time of one ECC modular subtraction operation	0.000290
T^e_{pmul}	Time of one ECC point multiplication operation	1.685240
T^e_{padd}	Time of one ECC point addition operation	0.007710

Next, based on the data in Table III and Table VI, we conduct quantitative computation overhead evaluation for the vehicles and TA in the PPRU and PPTM [5] schemes when u varies from 1 to 1000, $u' = u$, and $v = v' = u \cdot 10\%$, respectively, and the quantitative comparison results are illustrated in Fig. 5.

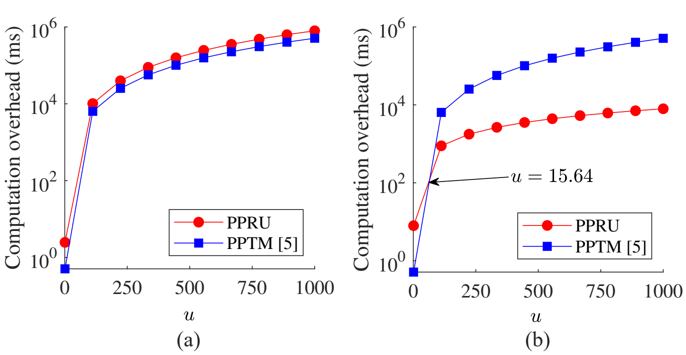


Fig. 5. Quantitative computation overhead comparisons of (a) vehicles and (b) TA in the PPRU and PPTM [5] schemes.

[1]https://www.java.com/.

[2]http://gas.dia.unisa.it/projects/jpbc/.

As revealed in Fig. 5(a), for each u, the computation overhead on the vehicles side in the PPRU scheme is slightly higher than that in the PPTM scheme. Note that the computation overhead shown in Fig. 5(a) is shared by u vehicles and the computation overhead on each vehicle side is far less than that shown in Fig. 5(a), thus the computation overhead on the vehicles side in the PPRU scheme is acceptable. As revealed in Fig. 5(b), for the most values of u (i.e., $u \geq 16$), the computation overhead on the TA side in the PPRU scheme is significantly lower than that in the PPTM scheme, which indicates that the PPRU scheme can dramatically reduce the computation overhead on the TA side when compared with the PPTM scheme. Specifically, when u varies from 1 to 1000, the average reducing percentage of the computation overhead on the TA side is about 88.36%.

Then, based on the data in Table III and Table VI, we conduct quantitative computation overhead evaluation for the CSP in the PPRU scheme when the batch verification is adopted, u varies from 1 to 1000, $u' = u$, $v = v' = u \cdot \varrho$ (where ϱ varies from 5% to 20%), respectively, and the concrete comparison results are illustrated in Fig. 6(a).

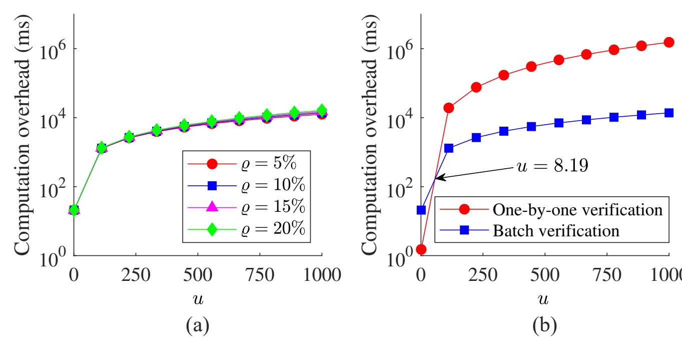


Fig. 6. Variation curve comparisons of the computation overhead on the CSP side versus u (a) when ϱ takes different values and (b) when the one-by-one verification and batch verification are adopted.

As revealed in Fig. 6(a), for each ϱ, the computation overhead on the CSP side first rapidly increases with u (when u is smaller than 100 or so), and then slowly increases with u (when u is larger than 100 or so). Meanwhile, for each u, the computation overhead on the CSP side slightly increases with ϱ. As we well know, the computational power of CSP in actual vehicular networks is much greater than that in our simulation, thus the computational overhead on the CSP side in actual vehicular networks is far less than that shown in Fig. 6(a), thus the computation overhead on the CSP side in the PPRU scheme is acceptable.

Afterwards, based on the data in Table III and Table VI, we conduct quantitative computation overhead evaluation for the CSP in the PPRU scheme when the one-by-one verification and batch verification are adopted, u varies from 1 to 1000, $u' = u$, $v = v' = u \cdot 10\%$, respectively, and the concrete comparison results are revealed in Fig. 6(b).

As revealed in Fig. 6(b), for the most values of u (i.e., $u \geq 9$), the computation overhead on the CSP side when the batch verification is adopted is significantly lower than that when the one-by-one verification is adopted, thus the batch verification can effectively reduce the computation overhead on the CSP side in the PPRU scheme when compared with the one-by-one verification.

VIII. Conclusion and Future Work

In this work, we have put forward a novel PPRU scheme for cloud-assisted vehicular networks based on the ECC and Paillier algorithms. Specifically, the reputation feedbacks are collected and preprocessed by the honest-but-curious CSP in a privacy-preserving manner, and the computation and communication overheads on the TA side can be dramatically reduced by about 88.36% and 83.88% as a result. Besides, the results of comprehensive theoretical analysis and simulation evaluation demonstrate that the proposed scheme can provide strong privacy preservation, strong security, and robust reputation management with acceptable computation and communication overheads, and is significantly superior to the existing schemes in several aspects. In future work, we will further improve the PPRU scheme by taking more potential attacks into consideration, and evaluate its performance in various kinds of simulational and real vehicular networks.

Acknowledgment

The authors would like to thank the anonymous reviewers for their constructive comments and suggestions.

References

[1] L. Wei, J. Cui, H. Zhong, I. Bolodurina, and L. Liu, "A lightweight and conditional privacy-preserving authenticated key agreement scheme with multi-TA model for fog-based VANETs," *IEEE Trans. Depend. Secure*, vol. 20, no. 1, pp. 422-436, Jan. 2023.

[2] L. Zhao, H. Chai, Y. Han, K. Yu, and S. Mumtaz, "A collaborative V2X data correction method for road safety," *IEEE Trans. Reliab.*, vol. 71, no. 2, pp. 951-962, June 2022.

[3] J. Guo, X. Li, Z. Liu, J. Ma, C. Yang, J. Zhang, and D. Wu, "TROVE: A context awareness trust model for VANETs using reinforcement learning," *IEEE Internet Things J.*, vol. 7, no. 7, pp. 6647-6662, July 2020.

[4] H. Cheng, M. Shojafar, M. Alazab, R. Tafazolli, and Y. Liu, "PPVF: Privacy-preserving protocol for vehicle feedback in cloud-assisted VANET," *IEEE Trans. Intell. Transp.*, vol. 23, no. 7, pp. 9391-9403, July 2022.

[5] Z. Liu, J. Weng, J. Guo, J. Ma, F. Huang, H. Sun, and Y. Cheng, "PPTM: A privacy-preserving trust management scheme for emergency message dissemination in space-air-ground integrated vehicular networks," *IEEE Internet Things J.*, vol. 9, no. 8, pp. 5943-5956, Apr. 2022.

[6] W. Li, C. Xia, C. Wang, and T. Wang, "Secure and temporary access delegation with equality test for cloud-assisted IoV," *IEEE Trans. Intell. Transp.*, vol. 23, no. 11, pp. 20187-20201, Nov. 2022.

[7] Y. Wang, W. Zhang, X. Wang, M. K. Khan, and P. Fan, "Efficient privacy-preserving authentication scheme with fine-grained error location for cloud-based VANET," *IEEE Trans. Veh. Technol.*, vol. 70, no. 10, pp. 10436-10449, Oct. 2021.

[8] H. Hou, J. Ning, Y. Zhao, and R. H. Deng, "A traitor-resistant and dynamic anonymous communication service for cloud-based VANETs," *IEEE Trans. Serv. Comput.*, vol. 15, no. 5, pp. 2551-2564, Sept. 2022.

[9] Q. Huang, N. Li, Z. Zhang, and Y. Yang, "Secure and privacy-preserving warning message dissemination in cloud-assisted Internet of vehicles," in *Proc. CNS*, 2019, pp. 1-8.

[10] Y. Cheng, J. Ma, Z. Liu, Y. Wu, K. Wei, and C. Dong, "A lightweight privacy preservation scheme with efficient reputation management for mobile crowdsensing in vehicular networks," *IEEE Trans. Depend. Secure*, vol. 20, no. 3, pp. 1771-1788, May 2023.

[11] J. Guo, Z. Liu, S. Tian, F. Huang, J. Li, X. Li, K. K. Igorevich, and J. Ma, "TFL-DT: A trust evaluation scheme for federated learning in digital twin for mobile networks," *IEEE J. Sel. Areas Comm.*, vol. 41, no. 4, pp. 3548-3560, Nov. 2023.

[12] S. Zhang, R. He, Y. Xiao, and Y. Liu, "A three-factor based trust model for anonymous bacon message in VANETs," *IEEE Trans. Veh. Technol.*, vol. 72, no. 9, pp. 11304-11317, Sept. 2023.

[13] S. Gyawali, Y. Qian, R. Q. Hu, "Deep reinforcement learning based dynamic reputation policy in 5G based vehicular communication networks," *IEEE Trans. Veh. Technol.*, vol. 70, no. 6, pp. 6136-6146, June 2021.

[14] Z. Liu, J. Weng, J. Ma, J. Guo, B. Feng, Z. Jiang, and K. Wei, "TCEMD: A trust cascading-based emergency message dissemination model in VANETs," *IEEE Internet Things J.*, vol. 7, no. 5, pp. 4028-4048, May 2020.

[15] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proc. Eurocrypt*, 1999, pp. 223-238.

[16] Y. Liang, H. Yan, and Y. Liu, "Unlinkable signcryption scheme for multi-receiver in VANETs," *IEEE Trans. Intell. Transp.*, vol. 24, no. 9, pp. 10138-10154, Sept. 2023.

[17] Z. Liu, J. Ma, J. Weng, F. Huang, Y. Wu, L. Wei, and Y. Li, "LPPTE: A lightweight privacy-preserving trust evaluation scheme for facilitating distributed data fusion in cooperative vehicular safety applications," *Inform. Fusion*, vol. 73, no. 1, pp. 144-156, Sept. 2021.

[18] C. Zhang, L. Zhu, C. Xu, K. Sharif, K. Ding, X. Liu, X. Du, and M. Guizani, "TPPR: A trust-based and privacy-preserving platoon recommendation scheme in VANET," *IEEE Trans. Serv. Comput.*, vol. 15, no. 2, pp. 806-818, Apr. 2022.

[19] Z. Liu, F. Huang, J. Weng, K. Cao, Y. Miao, J. Guo, and Y. Wu, "BTMPP: Balancing trust management and privacy preservation for emergency message dissemination in vehicular networks," *IEEE Internet Things J.*, vol. 8, no. 7, pp. 5386-5407, Apr. 2021.

[20] J. Cui, L. Wei, H. Zhong, J. Zhang, Y. Xu, and L. Liu, "Edge computing in VANETs-an efficient and privacy-preserving cooperative downloading scheme," *IEEE J. Sel. Areas Comm.*, vol. 38. no. 6, pp. 1191-1204, June 2020.

[21] X. Feng, Q. Shi, Q. Xie, and L. Wang, "P2BA: A privacy-preserving protocol with batch authentication against semi-trusted RSUs in vehicular ad hoc networks," *IEEE Trans. Inf. Foren. Sec.*, vol. 16. no. 1, pp. 3888-3899, Jan. 2021.

[22] C. Gong, C. Xu, Z. Zhou, T. Zhang, and S. Yang, "A reputation management scheme for identifying malicious nodes in VANET," in *Proc. HPSR*, 2019, pp. 1-6.

[23] V. S. Miller, "Use of elliptic curves in cryptography," in *Proc. Crypto*, 1985, pp. 417-426.

[24] M. R. Hajidavalloo, Z. Li, X. Xia, A. Louati, M. Zheng, and W. Zhuang, "Cloud-assisted collaborative road information discovery with Gaussian process: Application to road profile estimation," *IEEE Trans. Intell. Transp.*, vol. 23, no. 12, pp. 23951-23962, Dec. 2022.

[25] A. Mchergui, T. Moulahi, and S. Nasri, "BaaS: Broadcast as a service cross-layer learning-based approach in cloud assisted VANETs," *Comput. Netw.*, vol. 182, no. 1, pp. 107468, Nov. 2020.

[26] T. Mekki, I. Jabri, A. Rachedi, and M. B. Jemaa, "Vehicular cloud networks: Challenges, architectures, and future directions," *Veh. Commun.*, vol. 9, no. 1, pp. 268-280, July 2017.

[27] M. R. Jabbarpour, A. Marefat, A. Jalooli, and H. Zarrabi, "Could-based vehicular networks: A taxonomy, survey, and conceptual hybrid architecture," *Wirel. Netw.*, vol. 25, no. 1, pp. 335-354, Jan. 2019.

[28] X. Liu, O. Ma, W. Chen, Y. Xia, and Y. Zhou, "HDRS: A hybrid reputation system with dynamic update interval for detecting malicious vehicles in VANETs," *IEEE Trans. Intell. Transp.*, vol. 23, no. 8, pp. 12766-12777, Aug. 2022.

[29] B. Hammi, Y. M. Idir, S. Zeadally, R. Khatoun, and J. Nebhen, "Is it really easy to detect sybil attacks in C-ITS environments: A position paper," *IEEE Trans. Intell. Transp.*, vol. 23, no. 10, pp. 18273-18287, Oct. 2022.

[30] M. Baza, M. Nabil, M. M. Mahmoud, N. Bewermeier, K. Fidan, W. Alasmary, and M. Abdallah, "Detecting sybil attacks using proofs of work and location in vanets," *IEEE Trans. Depend. Secure*, vol. 19, no. 1, pp. 39-53, Jan. 2022.

[31] Z. Liu, J. Ma, Z. Jiang, and Y. Miao, "FCT: A fully-distributed context-aware trust model for location based service recommendation," *Sci. China Inform. Sci.*, vol. 60, no. 8, pp. 1-16, July 2017.

[32] 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. Veh. Technol.*, vol. 70, no. 2, pp. 1278-1291, Feb. 2021.

[33] N. Koblitz, "Elliptic curve cryptosystems," *Math. Comp.*, vol. 48, no. 177, pp. 203-209, Jan. 1987.

[34] W. Stallings, Cryptography and Network Security: Principles and Practice, 6th Edition, NJ: Prentice Hall, 2014.

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

[36] X. Yao, X. Zhang, H. Ning, and P. Li, "Using trust model to ensure reliable data acquisition in VANETs," *Ad Hoc Networks*, vol. 55, no. 1, pp. 107-118, Feb. 2017.

[37] D. Johnson, A. Menezes, and S. Vanstone, "The elliptic curve digital signature algorithm (ECDSA)," *Int. J. Inf. Secur.*, vol. 1, no. 1, pp. 36-63, Aug. 2001.

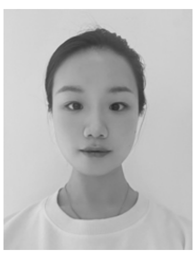
[38] H. Poincaré, Science and Hypothesis, MA: Courier Corporation, 1952.

[39] T. Peng, W. Zhong, G. Wang, S. Zhang, E. Luo, and T. Wang, "Spatiotemporal-aware privacy-preserving task matching in mobile crowdsensing," *IEEE Internet Things J.*, to be published, doi: 10.1109/JIOT.2023.3292284.

[40] D. Pointcheval and J. Stern, "Security arguments for digital signatures and blind signatures," *J. Cryptol.*, vol. 13, no. 3, pp. 361-396, Mar. 2000.

Zhiquan Liu received the B.S. degree from the School of Science, Xidian University, Xi'an, China, in 2012, and the Ph.D. degree from the School of Computer Science and Technology, Xidian University, Xi'an, China, in 2017.

He is currently an associate professor with the College of Cyber Security, Jinan University, Guangzhou, China. His current research focuses on security, trust, and privacy in vehicular networks, and his homepage is https://www.zqliu.com.



Lin Wan received the B.E. degree in the School of Network Engineering from Wuhan University of Science and Technology, Wuhan, China, in 2021.

She is currently working toward the M.S. degree with the College of Cyber Security, Jinan University, Guangzhou, China. Her current research interests include information security and privacy preservation in mobile crowdsensing and vehicular networks.

Jingjing Guo received the M.E. and Ph.D. degrees from the School of Computer Science and Technology, Xidian University, Xi'an, China, in 2012 and 2015, respectively.

She is currently an associate professor with the School of Cyber Engineering, Xidian University, Xi'an, China. Her current research interests include trust management, social network, access control, and information security.

Feiran Huang received the B.S. degree from the School of Physics and Electronics, Central South University, Changsha, China, in 2011, and the Ph.D. degree from the School of Computer Science and Engineering, Beihang University, Beijing, China, in 2019.

He is currently an associate professor with the College of Cyber Security, Jinan University, Guangzhou, China, and his current research focuses on multi-modal data analysis, social media analysis, and network security.

Xia Feng received the B.S. degree in computer science and technology from Jiangsu University in 2008, and the Ph.D. degree in computer science and technology department from Anhui University in 2017.

She is currently with the Faculty of Data Science, City University of Macau, and her research interests include authentication protocols in IoT, blockchain, and applied cryptography.

Libo Wang received the B.S. degree in mathematics from Xiangtan University, Xiangtan, China, in 2012, and the Ph.D. degree in applied mathematics from Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China, in 2017.

He is currently working in the College of Information Science and Technology, Jinan University, Guangzhou, China. His research fields include cryptography, coding theory, and information security.

Jianfeng Ma received the M.E. degree in School of Computer Science and Technology from Xidian University in 1988, and the Ph.D. degree in School of Information and Telecommunication Engineering from Xidian University in 1995.

He is currently a Professor with School of Cyber Engineering, Xidian University, and his current research interests include information security, coding theory, and cryptography.