Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.073774

ARTICLE

A Distributed Anonymous Reputation System for V2X Communication

Shahidatul Sadiah[1,#] **and Toru Nakanishi**[2,*,#]

[1]Faculty of Electrical Engineering, Universiti Teknologi Malaysia, Johor Bahru, 81310, Malaysia
[2]Graduate School of Advanced Science and Engineering, Hiroshima University, Higashi-Hiroshima, 739-8527, Japan

*Corresponding Author: Toru Nakanishi. Email: t-nakanishi@hiroshima-u.ac.jp

[#]These authors contributed equally to this work

Received: 25 September 2025; Accepted: 19 December 2025; Published: 10 February 2026

ABSTRACT: V2X communication enables vehicles to share real-time traffic and road-condition data, but binding messages to persistent identifiers enables location tracking. Furthermore, since forged reports from malicious vehicles can distort trust decisions and threaten road safety, privacy-preserving trust management is essential. Lu et al. previously presented BARS, an anonymous reputation mechanism founded on blockchain technology to establish a privacy-preserving trust architecture for V2X communication. In this system, reputation certificates without a vehicle identifier ensure anonymity, while two authorities jointly manage certificate issuance and reputation updates. However, the centralized certificate updates introduce scalability limitations, and the authorities can trace vehicle behavioral information, which threatens privacy guarantees. Several subsequent systems derived from BARS still rely on centralized certificate management and are subject to authority-side privacy leakage. As a result, a key challenge in this line of research remains unresolved: how to decentralize the certificate-update process while preserving privacy against the authorities in privacy-preserving V2X trust management. In this paper, we propose a distributed anonymous reputation system for V2X communication, based on an anonymous reputation system for crowdsensing. In our proposed system for V2X communication, the server is distributed to a certificate authority (CA) and roadside units (RSUs). Each vehicle shows the reputation level to the nearest RSU at the beginning of each time interval, and registers a short-time public key. In the interval, the messages from the vehicle are authenticated under the public key and are scored. At the end of the interval, the nearest RSU updates the certificate anonymously. Our solution decentralizes the certificate-update process by assigning each update to the nearest RSU. A zero-knowledge-proof-based show protocol removes the need for any central authority to handle vehicle certificates and thus prevents the authorities from tracing vehicle activities. Compared with BARS, where centralized authorities must update the reputation certificates of many vehicles and may incur communication and processing delays, our system performs each update locally at the nearest RSU once per interval. The required interaction consists only of a few kilobytes of communication and a zero-knowledge proof that is almost fully precomputed on the vehicle side, while the RSU-side processing is estimated to take about 40 ms based on timing measurements of the underlying cryptographic operations. This distributed update model avoids the centralized bottleneck of BARS and simultaneously removes the privacy risk arising from authority collusion.

KEYWORDS: V2X communication; anonymous reputation system; proof of knowledge

1 Introduction

The privacy of one's location is a major concern, especially for drivers. However, real-time local information, such as traffic and road conditions, is essential for efficient urban management. Therefore, it is critical to collect and share such local information while respecting privacy. Consider a scenario in which

vehicles collect local information and exchange it via V2X communication, such as V2V and V2I. To deter message tampering and impersonation, messages from each vehicle should be authenticated. Nevertheless, since location can often be inferred from both the reported content and authentication metadata, linking authenticated messages to persistent vehicle identities can violate privacy.

V2X communication faces significant challenges due to its open and decentralized nature. When malicious vehicles disseminate falsified data, they undermine both the reliability of information exchange and the safety of ordinary drivers. Consequently, trust evaluation and management in V2X communication has attracted significant attention in recent years.

1.1 Previous Works

Vehicle ad-hoc network (VANET) reputation systems have been studied in the literature, but most approaches do not consider the privacy aspect of vehicles (e.g., [1–3]). These studies mainly focus on frameworks for evaluating vehicle behavior and message forwarding to detect misbehavior. In contrast, in the context of crowdsensing, a privacy-aware reputation system called ARTSense was proposed in [4]. In crowdsensing, numerous mobile users voluntarily collect sensing data, such as location and environmental information, which are submitted to and analyzed by a centralized server. Therefore, user privacy must be taken into account. To achieve the anonymity of users, ARTSense separates the data reporting process from the reputation updating process. No user identity information is revealed in individual sensing reports. Furthermore, the server cannot link multiple reports to the same participant due to the use of blind IDs. However, this work is not designed for V2X communication, which is characterized by high mobility and a dynamic network topology.

In 2016, Jaimes et al. [5] proposed a centralized anonymous reputation system (ARS) for V2X communication, including VANETs, in which vehicles interact with roadside units (RSUs) to submit feedback to a centralized reputation server (RepS) and to retrieve their current reputation levels under pseudonyms. The server updates the reputation level of each vehicle by associating anonymous identities with real identities. ARS introduced the notion of security states, which can help identify region-specific risks and support the evaluation of neighboring vehicles' reputation scores. The reputation level of a vehicle is the result of the aggregation of the reputation levels by the characteristics of generation and forwarding of messages. However, the centralized server causes the bottleneck in scalability, single point of failure, and privacy risk that the server can reveal the location history of vehicle.

In 2018, Lu et al. [6] proposed a blockchain-based anonymous reputation system (BARS) to establish a privacy-preserving trust model for V2X communication. In this system, a certificate authority (CA) issues certificates to vehicles and manages revocation. All activities of CA are recorded on the blockchain for transparency. The certificate includes no vehicle ID and is thus anonymous. On the other hand, a law enforcement authority (LEA) is responsible for managing the correspondence between public keys and real identities. In case of disputes, the LEA can trace a vehicle from a public key used for authentication. Furthermore, BARS incorporates a reputation system in which the LEA monitors and evaluates each vehicle's behavior, and updates its reputation score. The reputation is certified by the certificate, and updated by the CA with the assistance of the LEA. The reputation system is anonymous due to the hidden vehicle's ID. However, the two authorities CA and LEA cooperatively update the certificate of each vehicle. Thus, when lots of vehicles communicate messages, the centralized update process can become a bottleneck, i.e., BARS also have a scalability disadvantage. In addition, if the two authorities collude, they may reveal the location history of a vehicle.

1.2 Our Contributions

In this paper, toward a distributed privacy-preserving trust management in V2X communication, we propose an anonymous reputation system for V2X communication, which is derived from an anonymous reputation system for crowdsensing [7]. In the system for crowdsensing, the server updates the reputation certificate of each user s.t. the user's ID and even the reputation value are hidden. Using a zero-knowledge proof (ZKP), the user can prove the reputation level (an integer range that contains the reputation value). In our proposed system for V2X communication, the server is distributed across CA and RSUs (roadside units). Each vehicle shows the reputation level to the nearest RSU at the beginning of each time interval, similarly to the underlying system [7], and registers a short-time public key. In the interval, the messages from the vehicle are authenticated under the public key and are scored. At the end of the interval, the nearest RSU updates the certificate anonymously.

To situate our contributions within existing approaches, Table 1 summarizes the difference in the architecture, anonymity, and costs across ARS, BARS, and our proposed system. This distributed update in our system avoids the centralized certificate-update process in BARS [6], where two central authorities must update the certificates of many vehicles, potentially causing communication and processing delays. In our system, each update is performed locally at the nearest RSU once per interval, requiring only a few kilobytes of communication and a zero-knowledge proof that is almost fully precomputed on the vehicle side, with the RSU-side processing estimated to take about 40 ms based on timing measurements of the underlying cryptographic operations. This distributed mechanism removes the centralized bottleneck of BARS and mitigates the privacy risk arising from possible authority collusion. Per-message authentication during the interval uses the same pseudonym-certificate mechanism as in existing RSU-assisted V2X systems, and ZKPs are required only once per interval; detailed efficiency considerations are discussed in Section 6. From these efficiency considerations, we consider that the practicality of the proposed system is demonstrated without requiring mobility-level simulations. Network-level performance evaluations under specific traffic and mobility models (e.g., NS-3 or SUMO) are important as complementary research directions that depend on application scenarios. We therefore leave such system-level evaluations as future work.

Table 1: Comparison of BARS, ARS, and the proposed system

System	Architecture	Anonymity	Per-message cost	Reputation update cost
ARS [5]	Centralized certificate-update; Messages forwarding	Conditional anonymity[†]	Pseudonym-based certificate verification	Centralized update at RepS; no ZKP[§]
BARS [6]	Centralized certificate-update; Blockchain transparency	Conditional anonymity[†]	Pseudonym-based certificate verification	Centralized update at CA and LEA; no ZKP[§]
Proposed	Decentralized certificate-update	Full anonymity[‡]	Pseudonym-based certificate verification	Decentralized update at RSU; ZKP[§] needed (≈ 1.5 KB, vehicle-side precomputation, RSU verification ≈ 40 ms)

Note: [†]Conditional anonymity implies that anonymity is compromised by authorities. [‡] Full anonymity implies that anonymity is not compromised by authorities. [§]ZKP stands for "Zero-Knowledge Proof".

In our system, a malicious vehicle cannot be traced by authorities such as the LEA. However, the vehicle can be scored low in the anonymous reputation system, and the messages can be flagged as untrustworthy. The secret key of each RSU is unique, and thus a compromised RSU must also be revoked. The revocation is done using a complete subtree (CS) method [8] to compute revocation information for RSUs. In this paper, we do not address blockchain-based transparency, and the detailed reputation evaluation algorithm for scoring is also beyond the scope of our study. While both components are essential for constructing a complete and practical V2X trust management system, they are conceptually separable from the fundamental challenge we focus on in this research line. Since our main contribution is the distributed certificate-update mechanism with zero-knowledge-proof-based privacy protection, the practical evaluation presented in Section 6 concentrates on this fully specified component. The integration of more sophisticated reputation evaluation algorithms and scalable blockchain-based transparency mechanisms, which may require additional cryptographic and system-level design considerations, is left as future work.

1.3 Difference from the Conference Version [9]

A preliminary version of this paper was presented in ICCE 2024 [9], where security requirements are informally defined, and only the proof sketches for the security requirements are shown. In this paper, we formally show game-based security definitions, and prove the security based on the definitions. Furthermore, we extend the original system [9] by adding a revocation function for malicious RSUs, and we newly provide a detailed efficiency evaluation of the system.

1.4 Related Works

The recent works related to BARS and our proposed system are as follows.

In [10], Ahmed et al. propose a privacy-enhancing V2X trust management system that combines pseudo-identity–based anonymous authentication with blockchain-based revocation, similar to BARS. Compared with BARS, it integrates a more sophisticated context-aware trust (reputation) computation and improves verification efficiency through signature aggregation. However, the scheme relies on a fully trusted TA (Trusted Authority) that centrally generates and manages each vehicle's pseudo-identities and secret keys, and thus TA can compromise user's privacy by de-anonymization, as in BARS.

In [11], Feng et al. propose a blockchain-based privacy-preserving authentication system for V2X environments. The system adopts a structure similar to BARS by introducing two ID management entities and realizing vehicle anonymity through pseudonym-based public-key certificates. In addition, the system employs an asynchronous accumulator that is a hash-tree-based mechanism, to accelerate revocation verification. However, this approach focuses solely on authentication and revocation, and thus does not incorporate a reputation-based trust management mechanism. Moreover, if the two management entities collude, the linkage of pseudonyms becomes possible, which results in the leakage of vehicle behavioral information.

In [12], Feng et al. proposed a privacy-preserving authentication scheme for V2X communication. In the system, a certification authority issues certificates for pseudonymous identities and enables anonymous authentication, while leveraging polynomial commitments to achieve constant-cost revocation checking. However, the system also focuses solely on authentication and revocation, and does not provide a reputation-based trust management. Moreover, the certificate update process is centrally managed, allowing the authority to link successive certificates, and thus the privacy of vehicles is compromised.

In [13], Hou et al. introduce a double-layer blockchain architecture, consisting of an event chain and a reputation chain, and propose a sophisticated reputation-based trust management system for V2X

communication, which resists against on–off attacks and collusive attacks. However, the system does not address privacy protection. It assumes a fully trusted TA that issues public-key certificates, and thus the TA can easily deanonymize any vehicle even when pseudonyms are used. Moreover, since both reputation information and event reports are persistently stored on the blockchain, linkability among observations and historical behavior may still weaken privacy, even under pseudonym-based identifiers.

In [14], Fernandes et al. propose a V2X trust management system based on a consortium blockchain that employs a PoA (Proof of Authority) consensus mechanism, where RSUs collaboratively update reputation scores to improve efficiency. However, the system relies entirely on a centralized CA for certificate issuance and management, and it does not support certificate updates. As a result, it cannot also prevent deanonymization by the CA, and the reputation values tied to static IDs allow vehicle behavior to be tracked through linkable evaluation records.

2 Preliminaries

2.1 Bilinear Groups

We adopt the following bilinear groups:

1. $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ are multiplicative groups with same prime order p, where $g \in \mathbb{G}_1$ and $h \in \mathbb{G}_2$ are generators.
2. $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is a bilinear map s.t.
   - for all $u \in \mathbb{G}_1$ and $v \in \mathbb{G}_2$, and $a, b \in \mathbb{Z}$, $e(u^a, v^b) = e(u, v)^{ab}$ holds.
   - $e(g, h)$ is a generator of $\mathbb{G}_T$.

2.2 Assumptions

For the security of the proposed system, we use the q-SDH assumption [15].

Definition 1 (q-SDH assumption): *For all PPT algorithm $\mathcal{A}$, the probability*

$$Pr[\mathcal{A}(u, v, v^a, \ldots, v^{(a^q)}) = (b, v^{1/(a+b)}) \wedge b \in \mathbb{Z}_p]$$

is negligible, where $u \xleftarrow{R} \mathbb{G}_1$, $v \xleftarrow{R} \mathbb{G}_2$ and $a \xleftarrow{R} \mathbb{Z}_p$.

2.3 BB Signatures

We employ the scheme in [15] where a message and the signature can be proven by the zero-knowledge proofs.

Here are the descriptions of the algorithm.

- **BB-Setup:** Select bilinear group parameters $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g, h)$.
- **BB-KeyGen:** Compute $w = h^\gamma$ for $\gamma \xleftarrow{R} \mathbb{Z}_p^*$, where the public key is $pk = w$ and the secret key is $sk = \gamma$.
- **BB-Sign:** On input of a message $m \in \mathbb{Z}_p$, compute $A = g^{1/(m+\gamma)}$.
- **BB-Verify:** On inputs of a message m and a signature A, check if $e(A, wh^m) = e(g, h)$.

In [15], the security is proved under the q-SDH assumption.

2.4 BBS+ Signatures

We also use an extension of BB signature, BBS+ signature, which is informally introduced in [16], to sign a vector of numerous messages. The concrete structure is displayed in [17,18].

- **BBS+-Setup:** Select bilinear group parameters $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, g, h)$. Then, select $g_1, \ldots, g_{L+1} \xleftarrow{R} \mathbb{G}_1$.
- **BBS+-KeyGen:** Compute $w = h^\gamma$ for $\gamma \xleftarrow{R} \mathbb{Z}_p^*$, where the public key is $pk = w$ and the secret key is $sk = \gamma$.
- **BBS+-Sign:** On input of a vector $\mathcal{M}$ of L messages $(m_1, \ldots, m_L) \in \mathbb{Z}_p^L$, choose $\eta, \zeta \xleftarrow{R} \mathbb{Z}_p$, and compute $A = (g g_1^\zeta g_2^{m_1} \ldots g_{L+1}^{m_L})^{1/(\eta+\gamma)}$. The signature is $\sigma = (A, \eta, \zeta)$.
- **BBS+-Verify:** For the signature $\sigma = (A, \eta, \zeta)$ and $(m_1, \ldots, m_L)$, $checkif\ e(A, wh^\eta) = e(g g_1^\zeta g_2^{m_1} \cdots g_{L+1}^{m_L}, h)$.

  The security is proved in [18] under the q-SDH assumption.

2.5 Signature-Based Proofs of Knowledges (SPKs)

For Non-Interactive Zero-Knowledge Protocol (NIZK) proofs on representations, we adopt signature-based proofs of knowledge (SPKs), which are converted from zero-knowledge proofs of knowledge (PoKs) or Sigma protocols [19]. Concretely, we utilize the SPK to prove a representation of $C \in \mathbb{G}_1$ to $g_1, g_2, \ldots \in \mathbb{G}_1$ (or $\mathbb{G}_2$, or $\mathbb{G}_T$) which is denoted as $SPK\{(x_1, x_2, \ldots) : C = g_1^{x_1} g_2^{x_2} \cdots\}(M)$ where $x_1, x_2, \ldots \in \mathbb{Z}_p$. This SPK means a signature on message M by a signer with the secrets $x_1, x_2, \ldots$ s.t. the relation holds.

2.6 Complete Subtree (CS) Method

We adopt Complete Subtree (CS) method [8] to achieve efficient user revocation in group signatures [20–22]. First, a group manager (GM) generates a binary tree with the number of leaves equal to the total number of users, N. Each node is assigned a node ID, and each user is assigned to a leaf node. An example is shown in Fig. 1, where each user is assigned to nodes u_7 to u_{14}. The GM issues and publishes a membership certificate A for each node on the path from the root node to the leaf nodes $u_0, u_1, \ldots, u_l$. Additionally, whenever a user is revoked, the GM issues and publishes a revocation certificate R for each *cover node* that is a root of a subtree whose subtrees consist only of leaf nodes of non-revoked users. A non-revoked user can prove that she has not been revoked by demonstrating the existence of both A and R generated from the same node. In the case shown in Fig. 1, if the user of node 10 is revoked, the GM selects cover nodes (u_2, u_3, u_9) using the CS method and generates corresponding certificates (R_2, R_3, R_9). The revocation list thus becomes (R_2, R_3, R_9). For N total users and r revoked users, the size of the revocation list is $O(r \cdot \log(N/r))$, enabling efficient revocation with $O(1)$ time complexity for signature generation and verification in group signatures.

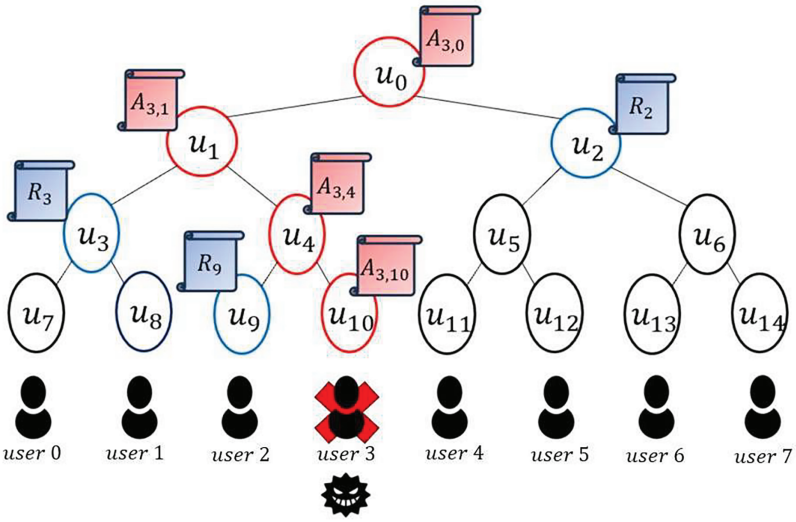

Figure 1: Example of CS method

3 Model of Proposed System

3.1 Syntax

The proposed anonymous reputation system consists of the following algorithm and protocols. The participants of this system are the certificate authority (CA), the roadside units (RSUs), and vehicles. In this system, messages sent from each anonymous vehicle are linkable via a pseudonym (i.e., short-time public key) during each time interval, and messages across intervals are unlinkable. At the first message in the interval, the vehicle shows the vehicle's reputation level to the nearest RSU, and the pseudonym at the interval is registered. The vehicle must ensure that the nearest RSU is not being revoked during this authentication. Each message is rated by the nearest RSU, and the scores at each interval are accumulated by RSUs. At the final message in the interval, the reputation of the vehicle is updated by the nearest RSU.

The anonymity of this system ensures that an adversary cannot obtain information about each vehicle other than the evaluation level and linkability (determining whether any two authenticated vehicles are the same) at each interval. This requirement means that the adversary cannot know the ID of the RSU with which each vehicle last communicated in the previous interval. Therefore, while hiding the public key of the last RSU that it communicated with, the vehicle performs a zero-knowledge proof of the certificate of the evaluation value by the RSU.

- **Setup:** CA takes a security parameter λ as input. This algorithm generates a key pair of public key spk and secret key ssk of the CA and initializes set SSet that keeps tags of used reputation certificates. This algorithm also prepares RSUs in a CS tree structure, generates a secret key rsk_i of each RSU i in the tree structure, and distributes the secret key to RSU i. SSet is shared with all RSUs.
- **Register:** This is an interactive protocol between a vehicle and the CA. The common input is spk, and the input of the CA is ssk. The vehicle is issued a reputation certificate $cert_0$ for initial reputation $rep_0 = 0$.
- **RevokeRSU:** This algorithm allows the CA to revoke an RSU. Given the ID of a revoked RSU, it outputs a revocation list RL_T at the current revocation interval T.
- **Show:** This protocol is called in the first communication at each time interval. This is an interactive protocol between a vehicle and the nearest RSU, where the vehicle shows that its reputation value is included in an integer range that is called reputation level. The vehicle also checks that the RSU is not

being revoked. The common input are spk and reputation level ℓ. The vehicle's input is $cert_{t-1}$ for its current reputation value rep_{t-1}. The RSU's input is SSet. The common outputs are a pseudonym $pseu_\tau$ of the vehicle at the current interval τ and the certificate $pcert_\tau$, the output of the vehicle is the secret key psk_τ of $pseu_\tau$, and the outputs of the RSU are a commitment $C'_{m,t}$ to messages to be certified for next **Update**, and the updated SSet to which the tag S_{t-1} of $cert_{t-1}$ is added. The pseudonym $pseu_\tau$, the reputation level ℓ, and $C'_{m,t}$ are shared with all RSUs.

- **Authentication:** This is an interactive protocol between a vehicle and the nearest RSU. The common input is spk, and the inputs of the vehicle are message M, reputation level ℓ, $pseu_\tau$, $pcert_\tau$, and psk_τ. The outputs of the RSU is the validity bit 1 (accepted) or 0 (rejected), and $(M, pseu_\tau)$.

  Each message M is rated by the nearest RSU based on some evaluation method (e.g., ARTSense [4]) that uses the reputation level together with other information. A negative rating is possible by using a negative integer. For the rated score s, $(M, pseu_\tau, s)$ is shared with all RSUs.
- **Update:** This protocol is called in the final communication at each time interval, where the total score $\tilde{s}_\tau$ for each $pseu_\tau$ at the current interval τ, i.e., $\tilde{s}_\tau = \sum_i s_i$ for all $(M_i, pseu_\tau, s_i)$, is prepared. This is an interactive protocol between a vehicle and the nearest RSU. The common input is spk, and the input of the vehicle is $cert_{t-1}$ for the previous reputation rep_{t-1}, and the inputs of the RSU i are $pseu_\tau, C'_{m,t}, \tilde{s}_\tau, rsk_i$. The output of the vehicle is a new reputation certificate $cert_t$ for the updated reputation $rep_t = rep_{t-1} + \tilde{s}_\tau$.

3.2 Security Requirements

As in [7], we consider the following requirements:

- **Reputation Unforgeability:** Any vehicle cannot prove inappropriate reputation level, i.e., for the correct reputation $\tilde{rep}_{t-1}$ which added from $\tilde{s}_{\tau_1}, \ldots, \tilde{s}_{\tau_k}$, where $\tilde{s}_{\tau_i}$ is the total score for each $pseu_{\tau_i}$ assigned to the vehicle at the interval τ_i, the vehicle cannot prove any inappropriate level ℓ s.t. $\tilde{rep}_{t-1}$ is not included in the integer range of the level ℓ.
- **Anonymity:** Any adversary cannot obtain any information on each vehicle except the reputation level and the linkability (i.e., whether the vehicles of any two authentications are the same or not) in each interval from the protocols. This means that the adversary cannot determine whether the vehicles of any two authentications are the same or not across intervals. Furthermore, this requirement means that the adversary cannot know the ID of the nearest RSU that executes the **Update** protocol.

We adjust the security requirements in the underlying system [7] to our above-mentioned model in the V2X communication, as follows. In the underlying system, each authentication is rated, and the score is added to the reputation of the user, but in the proposed system, scores in each interval are summed and added to the the reputation of the vehicle. Thus, in the underlying system, all authentications are unlinkable w.r.t. the sameness of the user, but in the proposed system, the authentications during each interval are linkable (the authentications across intervals are unlinkable).

Furthermore, we require the following security properties in authentications and RSU revocation.

- **Misauthentication resistance:** In each **Authentication** protocol, any vehicle which does not succeed **Show** protocol in the current interval cannot be accepted.
- **RSU revocablity:** An RSU can be revoked, and then any vehicle with a reputation certificate issued from a revoked RSU does not succeed **Show** protocol.

We formally define the security requirements, as follows.

3.2.1 Reputation Unforgeability

In the definition of reputation unforgeability, we utilize the following oracles.

- O_{CV-Reg}: It takes as input vehicle ID k. A **Register** protocol is executed between the honest CA and a corrupted vehicle k controlled by the adversary $\mathcal{A}$. k is added to the set of corrupted vehicles CV.
- $O_{CV-Show}$: It takes as inputs vehicle ID $k \in$ CV, RSU ID i, and reputation level ℓ. A **Show** protocol is executed between an honest RSU i and a corrupted vehicle k controlled by the adversary $\mathcal{A}$, where the reputation level ℓ is proved, the pseudonym $pseu_\tau$ and the certificate $pcert_\tau$ for the current interval τ are outputted, SSet is updated, and an entry $(k, pseu_\tau, \ell, C'_{m,t}, \tilde{s} = 0)$ is kept.
- $O_{CV-Auth}$: It takes as inputs vehicle ID $k \in$ CV, RSU ID i, the pseudonym $pseu_\tau$ at the current interval τ, message M, and the rated score s. An **Authentication** protocol is executed between an honest RSU i and a corrupted vehicle k controlled by the adversary $\mathcal{A}$ with the current revocation list RL_T, where $\tilde{s}$ in the entry $(k, pseu_\tau, \ell, C'_{m,t}, \tilde{s})$ is updated as $\tilde{s}' = \tilde{s} + s$.
- $O_{CV-Update}$: It takes as inputs vehicle ID $k \in$ CV, RSU ID i, and total score $\tilde{s}$ at the current interval. An **Update** protocol is executed between an honest RSU i and a corrupted vehicle k controlled by the adversary $\mathcal{A}$, where the total score $\tilde{s}$ is added to the reputation rep_{t-1} of the vehicle k, and rep_k of the entry (k, rep_k) is updated as $rep'_k = rep_k + \tilde{s}$ (if no entry for k, new entry of $(k, rep_k = \tilde{s})$ is generated).
- $O_{RevokeRSU}$: It takes as an input RSU ID i. Using **RevokeRSU**, the new RL_T at the current interval T is outputted.

Then, consider the following reputation unforgeability game, where $O = (O_{CV-Reg}, O_{CV-Show}, O_{CV-Auth}, O_{CV-Update}, O_{RevokeRSU})$.

$Game^{RU}_{\mathcal{A}}(\lambda)$:

$(spk, ssk, \mathsf{SSet}, \{rsk_i\}) \leftarrow$ **Setup**(λ);

Run $\mathcal{A}^O(spk)$;

Return 1 if

the final $O_{CV-Show}$ oracle is accepted,

but rep_k is not in the integer range of level ℓ.

Return 0;

Definition 2 (Reputation Unforgeability): *An anonymous reputation system is reputation unforgeable, if for any PPT adversary* $\mathcal{A}$, $Pr[Game^{RU}_{\mathcal{A}}(\lambda) = 1]$ *is negligible in* λ.

3.2.2 Misauthentication Resistance

In the definition of misauthentication resistance, we utilize the oracles in the reputation unforgeability.

Then, consider the following misauthentication resistance game, where $O = (O_{CV-Reg}, O_{CV-Show}, O_{CV-Auth}, O_{CV-Update}, O_{RevokeRSU})$.

$Game^{MR}_{\mathcal{A}}(\lambda)$:

$(spk, ssk, \mathsf{SSet}, \{rsk_i\}) \leftarrow$ **Setup**(λ);

Run $\mathcal{A}^O(spk)$;

Return 1 if

the final $O_{CV-Auth}$ oracle is accepted,

but the vehicle k is not accepted in $O_{CV-Show}$ oracle previously executed during the same interval.

Return 0;

Definition 3 (Misauthentication Resistance): *An anonymous reputation system is misauthentication resistant, if for any PPT adversary $\mathcal{A}$, $Pr[Game_{\mathcal{A}}^{MR}(\lambda) = 1]$ is negligible in λ.*

3.2.3 RSU Revocability

In the definition of RSU revocability, we utilize the oracles in the reputation unforgeability.

Then, consider the following revocability game, where $O = (O_{CV-Reg}, O_{CV-Show}, O_{CV-Auth}, O_{CV-Update}, O_{RevokeRSU})$.

$Game_{\mathcal{A}}^{Rev}(\lambda)$:

$(spk, ssk, \mathsf{SSet}, \{rsk_i\}) \leftarrow$ **Setup**(λ);

Run $\mathcal{A}^{O}(spk)$;

Return 1 if

the final $O_{CV-Show}$ oracle is accepted,

but for the vehicle k, the RSU i in the previously executed $O_{CV-Update}$ oracle is revoked.

Return 0;

Definition 4 (RSU Revocability): *An anonymous reputation system is RSU revocable, if for any PPT adversary $\mathcal{A}$, $Pr[Game_{\mathcal{A}}^{Rev}(\lambda) = 1]$ is negligible in λ.*

3.2.4 Anonymity

In the definition of anonymity, we utilize the following oracles.

- O_{HV-Reg}: It takes as input vehicle ID k. A **Register** protocol is executed between the corrupted CA controlled by the adversary $\mathcal{A}$ and an honest vehicle k. k is added to the set of honest vehicles HV.
- $O_{HV-Show}$: It takes as inputs vehicle ID $k \in \mathsf{HV}$, RSU ID i, and reputation level ℓ. A **Show** protocol is executed between an honest vehicle k and a corrupted RSU i controlled by the adversary $\mathcal{A}$, where the reputation level ℓ is proved, the pseudonym $pseu_\tau$ and the certificate $pcert_\tau$ for the current interval τ are outputted, SSet is updated, and an entry $(k, pseu_\tau, \ell, C'_{m,t}, \tilde{s} = 0)$ is kept.
- $O_{HV-Auth}$: It takes as inputs vehicle ID $k \in \mathsf{HV}$, RSU ID i, the pseudonym $pseu_\tau$ at the current interval τ, message M, and the rated score s. An **Authentication** protocol is executed between an honest vehicle k and a corrupted RSU i controlled by the adversary $\mathcal{A}$ with the current revocation list RL_T, where $\tilde{s}$ in the entry $(k, pseu_\tau, \ell, C'_{m,t}, \tilde{s})$ is updated as $\tilde{s}' = \tilde{s} + s$.
- $O_{HV-Update}$: It takes as inputs vehicle ID $k \in \mathsf{HV}$, RSU ID i, and total score $\tilde{s}$ at the current interval. An **Update** protocol is executed between an honest vehicle k and a corrupted RSU i controlled by the adversary $\mathcal{A}$, where the total score $\tilde{s}$ is added to the reputation rep_{t-1} of the vehicle k, and rep_k of the entry (k, rep_k) is updated as $rep'_k = rep_k + \tilde{s}$ (if no entry for k, new entry of $(k, rep_k = \tilde{s})$ is generated).
- $O_{RevokeRSU}$: It takes as an input RSU ID i. Using **RevokeRSU**, the new RL_T at the current interval T is outputted.
- O_{LoR}: It takes as inputs vehicle IDs $k_0^*, k_1^* \in \mathsf{HV}$, RSU ID i, and reputation level ℓ. Return 0 if $rep_{k_0^*} \neq rep_{k_1^*}$ for entries $(k_0^*, rep_{k_0^*}), (k_1^*, rep_{k_1^*})$. Otherwise, select random bit b, and a **Show** protocol is executed between an honest vehicle k_b^* and a corrupted RSU i controlled by the adversary $\mathcal{A}$, where the reputation level ℓ is proved, the pseudonym $pseu_\tau$ and the certificate $pcert_\tau$ for the current interval τ are outputted, SSet is updated, and an entry $(k_b^*, pseu_\tau, \ell, C'_{m,t}, \tilde{s} = 0)$ is kept. After that, a **Show** protocol is executed between an honest vehicle $k_{\neg b}^*$ and a corrupted RSU i similarly. Then, **Authentication** protocols for k_b^* and for $k_{\neg b}^*$ with corrupted RSUs are executed, and **Update** protocols for k_b^* and for $k_{\neg b}^*$ with corrupted RSUs are executed, where the added score $\tilde{s}$ is the same.

Then, consider the following anonymity game, where $O = (O_{HV-Reg}, O_{HV-Show}, O_{HV-Auth}, O_{HV-Update}, O_{RevokeRSU}, O_{LoR})$.

$Game^{Ano}_{\mathcal{A}}(\lambda)$:

$(spk, ssk, \mathsf{SSet}, \{rsk_i\}) \leftarrow \mathbf{Setup}(\lambda)$;

$b' \leftarrow \mathcal{A}^{O}(spk, ssk, \{rsk_i\})$;

Return 1 if $b = b'$;

Return 0;

Definition 5 (Anonymity): *An anonymous reputation system is anonymous, if for any PPT adversary* $\mathcal{A}$, $|Pr[Game^{Ano}_{\mathcal{A}}(\lambda) = 1] - 1/2|$ *is negligible in* λ.

4 Proposed Scheme

4.1 Construction Idea

In BARS [6], two central authorities update a certificate for a short-time public key and the reputation to issue the vehicle, where the authorities have to evaluate each vehicle and update the reputation and the certificate. As a result, the update process is centralized and not scalable.

In our system, distributed RSUs in the V2X system manage the scores of each vehicle in each interval, and the nearest RSU updates a certificate for the new reputation using the RSU's secret key.

As the base system, we adopt the anonymous reputation system [7] for crowdsensing. In the system, a server and users participate. The server issues a certificate for the reputation to each user, where the certificate is a BBS+ signature on the user's secret, a certificate tag for checking one-time use of the certificate, and the reputation. Since the concrete value of the reputation can reveal the relevance to other authentications, the reputation level (an integer range where the reputation value is included) is shown in the **Show** protocol for authentication. In addition, while the reputation value is hidden using commitments, the certificate is updated by the server s.t. the certified reputation is reflected by the score for the authentication using the evaluation method of ARTSense [4].

We extend the system of [7] to construct the anonymous reputation system for V2X communication, as follows. In our system, a CA, RSUs, and vehicles participate. The central CA generates the CA's key pair, and each RSU's individual key pairs, and also generates the certificate of the RSU's public key as a BBS+ signature which can be proved by an SPKs. The BBS+ signature is also used for RSU revocation, as mentioned later. At first, a vehicle is issued as an initial certificate which is similar to the original certificate in [7], i.e., a BBS+ signature on the user's secret, a certificate tag, and the (initial) reputation. The original **Show** protocol is separated to **Show** protocol and **Update** protocol in our system. In **Show** called in the first time of each time interval, a vehicle proves the knowledge of the reputation certificate similarly to [7] to show the reputation level. In **Update** called in the final time of each time interval, instead of the central CA, the nearest RSU updates the reputation certificate reflected by the total score of the vehicle in the interval, where the BBS+ signature of the certificate is generated using the RSU's secret key.

The point in this construction is that, in **Show**, the vehicle needs to hide the ID of the RSU updating the certificate, since the RSU's ID allows one to link **Show** and **Update** by the same RSU. This is why we use an SPK where the RSU's public key of BBS+ signatures is hidden but the correctness is ensured by proving the knowledge of the certificate of the public key.

In addition, we introduce the authentication protocol for each vehicle accepted by **Show** protocol to send a message to the nearest RSU. In **Show** protocol, for a short time public key in an ordinary digital signature scheme, the public key certificate is issued from the RSU. In each message authentication, the sent

message is signed w.r.t. the public key. Thus, authentications during one interval can be linked. However, authentications across the interval are unlinkable.

Furthermore, we adopt a CS-based revocation method for RSUs. A CS tree is constructed, where each leaf corresponds to an RSU. The certificate of an RSU's key consists of BBS+ signatures on the RSU's secret key and on each node ID u_j along the path from the root to the RSU's leaf. To revoke an RSU, the CA selects cover nodes representing the revocation and publishes BBS+ signatures on these cover nodes together with the revocation interval, forming the revocation certificate. In the **Show** protocol, the vehicle additionally proves that for some node ID u_j—which is signed by a BBS+ signature for the public key of the RSU that issued the current reputation certificate—the u_j is a cover node, i.e., it is signed as part of the revocation certificate. This implies that the RSU has not been revoked.

4.2 Proposed Algorithm and Protocols

Fig. 2 illustrates the overall protocol flow among the vehicle, the nearest RSU, and the CA. The system public parameters are published to all participants. In **Setup**, the CA distributes secret keys to RSUs. In **Register**, the CA provides each vehicle with a vehicle secret key and an initial reputation certificate. The latest RSU revocation list is distributed in **RevokeRSU**. At the beginning of each interval, the vehicle executes the **Show** protocol with the nearest RSU to prove its reputation level and register a fresh short-term public key as its pseudonym key. During the interval, messages are authenticated using pseudonym-based signatures, and the RSU locally updates the scores. At the end of the interval, the vehicle anonymously obtains an updated reputation certificate from the nearest RSU through the **Update** protocol.

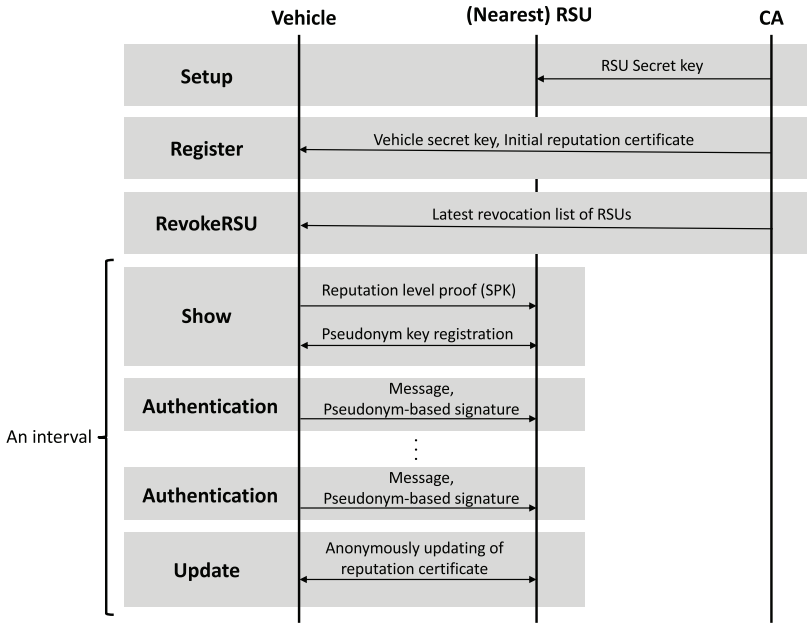

Figure 2: Protocol flow: **Setup**, **Register**, **RevokeRSU**, **Show**, per-message **Authentication** in an interval, and **Update**

The proposed algorithm and protocols are as follows.

Setup: In this algorithm, the CA generates key pairs of BB signatures and BBS+ signatures. Then, the CA computes the BB signature on every value in the integer range of reputation level $1 \le \ell \le L$ as the reputation level certificate. The CA also computes the secret key rsk_i of each RSU i and their BBS+ signatures. The RSU is authenticated using CS-method-based revocable scheme. In the previous work [7], a BB signature certificate for the public key $w_{2,i} = h_0^{\gamma_{2,i}}$ of RSU i was issued as $\tilde{A}_i = f_0^{1/(\gamma_1+\gamma_{2,i})}$. However, in the proposed system, the certificate is changed to a BBS+ signature certificate $\tilde{A}_{i,j}$. This allows it to serve not only as the certificate of RSU public key but also as the certificate A of node u_j in the CS method.

1. Select bilinear groups $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$, and a bilinear map e with a prime order $p > 2^\lambda$, where λ is the given security parameter. Then, select $g_0, g_1, g_2, g_3, g_4, f_0, f_1 \xleftarrow{R} \mathbb{G}_1,\ h_0, h_1 \xleftarrow{R} \mathbb{G}_2$. For all $1 \le \ell \le L$, choose $\gamma_{0,\ell} \xleftarrow{R} \mathbb{Z}_p^*$, and computes $w_{0,\ell} = h_0^{\gamma_{0,\ell}}$, where $\gamma_{0,\ell}$ is the secret key for the BB signature proving the reputation level ℓ. Choose $\gamma_1 \xleftarrow{R} \mathbb{Z}_p^*$, and compute $w_1 = h_0^{\gamma_1}$, where γ_1 is the secret key for the following certificate (BBS+ signature) for $w_{2,i}$. For all RSU $i \in [1, \mathsf{num}_{RSU}]$, where num_{RSU} is the number of RSUs, choose $\gamma_{2,i} \xleftarrow{R} \mathbb{Z}_p^*$, and compute $w_{2,i} = h_0^{\gamma_{2,i}}$ and $\tilde{w}_{2,i} = g_3^{\gamma_{2,i}}$ where $\gamma_{2,i}$ is the secret key of every RSU i in the BBS+ signatures. As the special secret key of the CA, choose $\gamma_{2,0} \xleftarrow{R} \mathbb{Z}_p^*$, and compute $w_{2,0} = h_0^{\gamma_{2,0}}$ and $\tilde{w}_{2,0} = g_3^{\gamma_{2,0}}$.
2. For all $1 \le \ell \le L$, generate the reputation level certificate $A_{\ell,R_{\ell,k}} = f_0^{1/(\gamma_{0,\ell}+R_{\ell,k})}$ (BB signature) for every value $R_{\ell,k}$ in the ℓ-th integer range indicating reputation level ℓ, where K_ℓ is the number of the values in the ℓ-th integer range.
3. The CA assigns RSU i to a leaf u_l of a binary tree in CS method, and $u_0, u_l, \cdots, u_\ell$ are the nodes on the path from the root node to the leaf node u_ℓ. For $j = 0, \cdots, \ell$, randomly choose $\eta'_{i,j}, \zeta'_{i,j} \xleftarrow{R} \mathbb{Z}_p^*$ and issue a BBS+ signature $(\tilde{A}_{i,j}, \eta'_{i,j}, \zeta'_{i,j})$ on $(u_j, \gamma_{2,i})$ s.t.

$$\tilde{A}_{i,j} = \left(g_0 g_1^{\zeta'_{i,j}} g_2^{u_j} g_3^{\gamma_{2,i}}\right)^{\frac{1}{\gamma_1+\eta'_{i,j}}}.$$

Then, send $\langle v_i \rangle := (u_0, u_1, \cdots, u_\ell)$ to RSU i. The public key of RSU i is $\tilde{w}_{2,i} = g_3^{\gamma_{2,i}}$.
4. For pseudonym certificates of each RSU $i \in [0, \mathsf{num}_{RSU}]$, generate a secret key $pcsk_i$ and the corresponding public key $pcpk_i$ in the ordinary digital signature scheme. For the public key certificates, generate a secret key $pcsk_{CA}$ of CA and the corresponding public key $pcpk_{CA}$ in the ordinary digital signature scheme. For every $i \in [1, \mathsf{num}_{RSU}]$, generate the public key certificate $pcpkcert_i$ as the digital signature on message $(pcpk_i, i)$ using the secret key $pcsk_{CA}$.
5. Initialize set SSet as empty, and output CA's public key

$$spk = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e, \{w_{0,\ell}\}_{\ell=1}^{L}, w_1, \{(w_{2,i}, \tilde{w}_{2,i}\}_{i=0}^{\mathsf{num}_{RSU}}, g_0, g_1, g_2, g_3, g_4, f_0, f_1, h_0, h_1,$$
$$\{\{A_{\ell,k}\}_{k=1}^{K_\ell}\}_{\ell=1}^{L}, \{\langle v_i \rangle, \{(\tilde{A}_{i,j}, \eta'_{i,j}, \zeta'_{i,j})\}_{j=0}^{\ell}\}_{i=0}^{\mathsf{num}_{RSU}}, pcpk_{CA}),$$

the CA's secret key $ssk = \gamma_{2,0}$, and the RSU's secret key $rsk_i = (\gamma_{2,i}, pcsk_i, pcpk_i, pcpkcert_i)$ for $i \in [1, \mathsf{num}_{RSU}]$.

Register: This is an interactive protocol between a vehicle **V** and the CA. The CA issues an initial reputation certificate $cert_0$ for the vehicle. The common input is spk, and the CA's input is ssk.

1. **[V]:** Select secret $x \xleftarrow{R} \mathbb{Z}_p^*$, a reputation certificate's tag $S_0 \xleftarrow{R} \mathbb{Z}_p^*$, and a random factor $\zeta'_0 \xleftarrow{R} \mathbb{Z}_p^*$. Compute the commitment to the messages (x, S_0) to be signed by $C'_{m,0} = g_1^{\zeta'_0} g_2^{x} g_3^{S_0}$. Then, prove to the

CA that $C'_{m,0}$ is correctly formed by the following *SPK* on a random message $\hat{M}$.

$$SPK\{(\zeta'_0, x, S_0) : C'_{m,0} = g_1^{\zeta'_0} g_2^{x} g_3^{S_0}\}(\hat{M})$$

2. **[CA]:** Set the initial reputation as $rep_0 = 0$, and choose random factors $\zeta''_0, \eta_0 \xleftarrow{R} \mathbb{Z}_p^*$. Then, using the secret key $\gamma_{2,0}$ of BBS+ signatures, sign the vector of messages (x, S_0, rep_0) as $B_0 = (g_0 g_1^{\zeta''_0} C'_{m,0} g_4^{rep_0})^{1/\gamma_{2,0}+\eta_0}$, and send back $\tilde{\sigma}'_0 = (B_0, \eta_0, \zeta''_0)$ to the vehicle.
3. **[V]:** Set $C_{m,0} = C'_{m,0} g_4^{rep_0}$ for $rep_0 = 0$, compute $\zeta_0 = \zeta'_0 + \zeta''_0$, and set the BBS+ signature on the messages (x, S_0, rep_0) as $\tilde{\sigma}_0 = (B_0, \eta_0, \zeta_0)$, where $B_0 = (g_0 g_1^{\zeta_0} g_2^{x} g_3^{S_0} g_4^{rep_0})^{1/\gamma_{2,0}+\eta_0}$. Output $cert_0 = (x, rep_0, \tilde{\sigma}_0, S_0, C_{m,0})$.

RevokeRSU: This algorithm enables the CA to revoke an RSU. For the current tree, the cover nodes obtained using the CS method are denoted as $\{u'_0, u'_1, \cdots, u'_{\text{num}}\}$, where $\text{num} \le r \cdot \log(N/r)$. For all $i \in [0, \text{num}]$, random values $\eta''_{T,j}, \zeta''_{T,j} \xleftarrow{R} \mathbb{Z}_p^*$ are chosen, and the revocation certificate is calculated as a BBS+ signature $(R_{T,j}, \eta''_{T,j}, \zeta''_{T,j})$ on (u'_j, T) s.t.

$$R_{T,j} = \left(g_0 g_1^{\zeta''_{T,j}} g_2^{u'_j} g_3^{T}\right)^{\frac{1}{\gamma_1+\eta''_{T,j}}}.$$

The revocation list is output as $RL_T = \{(R_{T,j}, \eta''_{T,j}, \zeta''_{T,j})\}_{j=1}^{\text{num}}$.

Show: This is an interactive protocol between a vehicle **V** and the nearest RSU, where the vehicle shows RSU its reputation level ℓ, a pseudonym $pseu_\tau$ is registered for a time interval τ, and the corresponding secret key psk_τ is kept in **V**. At the beginning of the protocol, the vehicle's reputation value is proved on $cert_{t-1}$, where the vehicle's inputs are $cert_{t-1} = (x, rep_{t-1}, \tilde{\sigma}_{t-1}, S_{t-1}, C_{m,t-1})$, where $\tilde{\sigma}_{t-1} = (B_{t-1}, \eta_{t-1}, \zeta_{t-1})$. Here, t indicates the number of updates in the reputation certificates for the vehicle. Let RSU i be the RSU that issued $\tilde{\sigma}_{t-1}$. The input of RSU is SSet.

1. **[V]:** From *spk*, retrieve a reputation level certificate $A_{\ell,rep_{t-1}}$ such that its current reputation rep_{t-1} is in the ℓ-th range. Choose $r_{A_\ell} \xleftarrow{R} \mathbb{Z}_p$ and compute the commitment $C_{A_\ell} = A_{\ell,rep_{t-1}} f_1^{r_{A_\ell}}$ and $\rho = r_{A_\ell} \cdot rep_{t-1}$. Retrieve a certificate $(\tilde{A}_{i,j}, \eta'_{i,j}, \zeta'_{i,j})$ for a node u_j of the CS-method tree s.t. the certificate was issued to the RSU i and u_j is a cover node at the current revocation time T. Choose $\hat{\zeta}' \xleftarrow{R} \mathbb{Z}_p$ and compute the commitment $C_{\tilde{A}_{i,j}} = \tilde{A}_{i,j} g_1^{\hat{\zeta}'}$. Choose $r_{\tilde{w}_{2,i}} \xleftarrow{R} \mathbb{Z}_p$ and compute the commitment $C_{\tilde{w}_{2,i}} = \tilde{w}_{2,i} f_1^{r_{\tilde{w}_{2,i}}}$. Set $\theta' = \zeta'_{i,j} + \hat{\zeta} \cdot \eta'_{i,j}$. Retrieve the revocation certificate $(R_{T,j}, \eta''_{T,j}, \zeta''_{T,j})$ for the cover node u_j at T. Choose $\hat{\zeta}'' \xleftarrow{R} \mathbb{Z}_p$, and compute $C_{R_{T,j}} = R_{T,j} \cdot g_1^{\hat{\zeta}''}$. Set $\theta'' = \zeta''_{T,j} + \hat{\zeta}'' \cdot \eta''_{T,j}$ Then, choose $\hat{\zeta}, r_{w_{2,i}} \xleftarrow{R} \mathbb{Z}_p$, compute the commitments $C_{B_{t-1}} = B_{t-1} g_1^{\hat{\zeta}}$ and $C_{w_{2,i}} = w_{2,i} \cdot h_1^{r_{w_{2,i}}}$, and set $\theta = \zeta_{t-1} + \hat{\zeta}\eta_{t-1}$. Choose $\zeta'_t \xleftarrow{R} \mathbb{Z}_p^*$ and $S_t \xleftarrow{R} \mathbb{Z}_p^*$, and compute $C'_{m,t} = g_1^{\zeta'_t} g_2^{x} g_3^{S_t} g_4^{rep_{t-1}}$ as the commitment to the vector of (x, S_t, rep_{t-1}). Set $v = \hat{\zeta} \cdot r_{w_{2,i}}$ and compute the commitments $C_{\hat{\zeta}} = g_0^{\hat{\zeta}} g_1^{r_{\hat{\zeta}}}$ and $C_v = g_0^{v} g_1^{r_v}$ for a randomly chosen $r_{\hat{\zeta}}, r_v \in \mathbb{Z}_p$. Set $\hat{\beta} = r_v - r_{w_{2,i}} \cdot r_{\hat{\zeta}}$. Send $C_{A_\ell}, C_{\tilde{A}_{i,j}}, C_{\tilde{w}_{2,i}}, C_{R_{T,j}}, C_{B_{t-1}}, C_{w_{2,i}}, C'_{m,t}, C_{\hat{\zeta}}, C_v$, and S_{t-1} to the nearest RSU, and using the following *SPK* on a random message $\hat{M}$, prove that the reputation rep_{t-1} is in the ℓ-th range, $cert_{t-1}$ is valid, $C'_{m,t}$ is correct, and the RSU i is not revoked.

$$SPK\{(r_{A_\ell}, rep_{t-1}, \rho, \theta', u_j, r_{\tilde{w}_{2,i}}, \eta'_{i,j}, \hat{\zeta}', \theta'', \eta''_{T,j}, \hat{\zeta}'', \theta, x, r_{w_2}, \eta_{t-1}, \hat{\zeta}, v, r_{\hat{\zeta}}, r_v, \hat{\beta}, \zeta'_t, S_t) :$$
$$e(C_{A_\ell}, w_{0,\ell}) \cdot e(f_0, h_0)^{-1} = e(f_1, w_{0,\ell})^{r_{A_\ell}} \cdot e(C_{A_\ell}, h_0)^{-rep_{t-1}} \cdot e(f_1, h_0)^{\rho} \quad (1)$$

$$\wedge e(C_{\tilde{A}_{i,j}}, w_1) \cdot e(C_{\tilde{w}_{2,i}}, h_0)^{-1} \cdot e(g_0, h_0)^{-1} = e(g_1, h_0)^{\theta'} \cdot e(g_2, h_0)^{u_j} \cdot e(f_1, h_0)^{-r_{\tilde{w}_{2,i}}} \cdot e(C_{\tilde{A}_{i,j}}, h_0)^{-\eta'_{i,j}} \cdot e(g_1, w_1)^{\hat{\zeta}'} \quad (2)$$

$$\wedge e(C_{R_{T,j}}, w_1) \cdot e(g_3, h_0)^{-T} \cdot e(g_0, h_0)^{-1} = e(g_1, h_0)^{\theta''} \cdot e(g_2, h_0)^{u_j} \cdot e(C_{R_{T,j}}, h_0)^{-\eta''_{T,j}} \cdot e(g_1, w_1)^{\hat{\zeta}''} \quad (3)$$

$$\wedge e(C_{B_{t-1}}, C_{w_{2,i}}) \cdot e(g_3, h_0)^{-S_{t-1}} \cdot e(g_0, h_0)^{-1} = e(g_1, h_0)^{\theta} \cdot e(g_2, h_0)^{x} \cdot e(g_4, h_0)^{rep_{t-1}} \cdot e(C_{B_{t-1}}, h_1)^{r_{w_{2,i}}} \cdot e(C_{B_{t-1}}, h_0)^{-\eta_{t-1}} \cdot e(g_1, C_{w_{2,i}})^{\hat{\zeta}} \cdot e(g_1, h_1)^{-\nu} \quad (4)$$

$$\wedge C_{\hat{\zeta}} = g_0^{\hat{\zeta}} g_1^{r_{\hat{\zeta}}} \wedge C_{\nu} = g_0^{\nu} g_1^{r_{\nu}} \quad (5)$$

$$\wedge C_{\nu} = C_{\hat{\zeta}}^{r_{w_{2,i}}} \cdot g_1^{\hat{\beta}} \quad (6)$$

$$\wedge e(C_{\tilde{w}_{2,i}}, h_0) \cdot e(g_3, C_{w_2,i})^{-1} = e(f_1, h_0)^{r_{\tilde{w}_{2,i}}} \cdot e(g_3, h_1)^{-r_{w_{2,i}}} \quad (7)$$

$$\wedge C'_{m,t} = g_1^{\zeta'_t} g_2^{x} g_3^{S_t} g_4^{rep_{t-1}}\}(\hat{M}). \quad (8)$$

The Eq. (1) implies the verification of the (variant of) BB signature $\tilde{A}_\ell$ on message rep_{t-1} for level ℓ, as in [7]. The Eq. (2) implies the verification of the BBS+ signature $(\tilde{A}_{i,j}, \eta'_{i,j}, \zeta'_{i,j})$ on messages $u_j, \gamma_{2,i}$ w.r.t. the CA's public key w_1, which ensures the node u_j on the path to RSU and the RSU's public key $w_{2,i}$. The Eq. (3) implies the verification of BBS+ signature $(R_{T,j}, \eta''_{T,j}, \zeta''_{T,j})$ on message u_j, T w.r.t. public key w_1, which ensures that the cover node u_j of the tree at time T is the same as the node u_j for $\tilde{A}_{i,j}$. The Eqs. (5) and (6) show $\nu = \hat{\zeta} \cdot r_{w_{2,i}}$, and thus the Eq. (4) implies the verification of the BBS+ signature $(B_{t-1}, \eta_{t-1}, \zeta_{t-1})$ on messages x, S_{t-1}, rep_{t-1} w.r.t. the RSU i's public key $w_{2,i}$. The Eq. (7) shows the same secret key $\gamma_{2,i}$ of $w_{2,i}$ and $\tilde{w}_{2,i}$. These are proved in **Lemma 1** in the next section.

2. **[RSU]:** To check the freshness of the proved certificate, check if $S_{t-1} \in$ SSet. If it is true, abort. Otherwise, add tag S_{t-1} in set SSet. Verify the *SPK*. If it is invalid, abort.
3. **[V]:** Generate a short-time key pair in the ordinary digital signature scheme, where the public key is opk_τ and the secret key is osk_τ. Send opk_τ as the pseudonym at the current interval τ.
4. **[RSU]:** As the certificate on opk_τ, generate the ordinary signature $sig_{i,\tau}$ on message $(opk_\tau, \ell, \tau, i)$ using the secret key $pcsk_i$ in rsk_i, for the reputation level ℓ of the vehicle, and send $pseu_\tau = opk_\tau$ and $pcert_\tau = (sig_{i,\tau}, pcpk_i, pcpkcert_i)$. Output $pseu_\tau$, $pcert_\tau$, $C'_{m,t}$, and the updated SSet.
5. **[V]:** Verify the signature $sig_{i,\tau}$ on $(opk_\tau, \ell, \tau, i)$ using the public key $pcpk_i$ in the ordinary digital signatures. Verify the signature $pcpkcert_i$ on $(pcpk_i, i)$ using the public key $pcpk_{CA}$ in the ordinary digital signatures. If either is invalid, abort. Otherwise, output $pseu_\tau = opk_\tau$, $pcert_\tau = (sig_{i,\tau}, pcpk_i, pcpkcert_i)$, and $psk_\tau = osk_\tau$.

Authentication: This is an interactive protocol between a vehicle **V** and the nearest RSU. The common input is spk, and the input of the vehicle is message M, $pseu_\tau$, $pcert_\tau$, and psk_τ. The output of the RSU is the validity bit 1 (accepted) or 0 (rejected), and $(M, pseu_\tau)$.

1. **[V]:** For message M, using the secret key $psk_\tau = osk_\tau$, compute the digital signature $authsig$ on M and send $(M, authsig, pseu_\tau, pcert_\tau, \ell)$.
2. **[RSU]:** Using $pcpk_i$ (resp., $pcpk_{CA}$), verify the signature $sig_{i,\tau}$ (resp., $pcpkcert_i$) on $(opk_\tau, \ell, \tau, i)$ (resp., $(pcpk_i, i)$) where $pcert_\tau = (sig_{i,\tau}, pcpk_i, pcpkcert_i)$ and $pseu_\tau = opk_\tau$. Using opk_τ, verify the signature $authsig$ on M. If either one is not valid, abort. Otherwise, this vehicle is accepted. Output $(M, pseu_\tau)$.

Update: This is an interactive protocol between a vehicle and the nearest RSU i. The common input is spk, and the input of the vehicle is $cert_{t-1}$ for the previous reputation rep_{t-1}, and the inputs of the RSU i are

$pseu_\tau, C'_{m,t}, \tilde{s}_\tau, rsk_i$. The output of the vehicle is a new reputation certificate $cert_t$ for the updated reputation $rep_t = rep_{t-1} + \tilde{s}_\tau$.

1. **[RSU]:** Compute $C_{m,t} = C'_{m,t} g_4^{\tilde{s}_\tau}$, and using $\gamma_{2,i}$ in $rsk_i = (\gamma_{2,i}, pcsk_i, pcpk_i, pcpkcert_i)$, generate $B_t = (g_0 g_1^{\zeta''_t} C_{m,t})^{1/\gamma_{2,i}+\eta_t} = (g_0 g_1^{\zeta''_t} g_1^{\zeta'_t} g_2^{x} g_3^{S_t} g_4^{rep_{t-1}} g_4^{\tilde{s}_\tau})^{1/\gamma_{2,i}+\eta_t}$ for $\zeta''_t, \eta_t \xleftarrow{R} \mathbb{Z}_p^*$. Then, send back $\tilde{\sigma}'_t = (B_t, \eta_t, \zeta''_t)$ to the vehicle.
2. **[V]:** Compute $\zeta_t = \zeta'_t + \zeta''_t$, $rep_t = rep_{t-1} + \tilde{s}_\tau$ and set the signature on the vector of messages (x, S_t, rep_t) as $\tilde{\sigma}_t = (B_t, \eta_t, \zeta_t)$, where $B_t = (g_0 g_1^{\zeta_t} g_2^{x} g_3^{S_t} g_4^{rep_t})^{1/\gamma_{2,i}+\eta_t}$. Output $cert_t = (x, rep_t, \tilde{\sigma}_t, S_t, C_{m,t})$.

5 Security

For the security of our system, we show the following lemma.

Lemma 1: *The SPK in* **Show** *proves the knowledge of* $A'_\ell, \xi, rep_{t-1}, \tilde{A}_{i,j}, \zeta'_{i,j}, \eta'_{i,j}, u_j, R_{T,j}, \zeta''_{T,j}, \eta''_{T,j}, B_{t-1}, \zeta_{t-1}, \eta_{t-1}, x$ *such that*

$$A'_\ell = (f_0 f_1^{\xi})^{1/(\gamma_{0,\ell}+rep_{t-1})}, \quad \tilde{A}_{i,j} = \left(g_0 g_1^{\zeta'_{i,j}} g_2^{u_j} g_3^{\gamma_{2,i}}\right)^{\frac{1}{\gamma_1+\eta'_{i,j}}}, \quad R_{T,j} = \left(g_0 g_1^{\zeta''_{T,j}} g_2^{u_j} g_3^{T}\right)^{\frac{1}{\gamma_1+\eta''_{T,j}}},$$

$$B_{t-1} = (g_0 g_1^{\zeta_{t-1}} g_2^{x} g_3^{S_{t-1}} g_4^{rep_{t-1}})^{1/(\gamma_{2,i}+\eta_{t-1})}, \quad w_{2,i} = h_0^{\gamma_{2,i}}, \text{ and } \tilde{w}_{2,i} = g_3^{\gamma_{2,i}}.$$

Proof: The equation for A'_ℓ can be shown as in [7], since the Eq. (1) proved in the SPK is the same. From Eq. (2), we obtain

$$e(C_{\tilde{A}_{i,j}}, w_1 h_0^{\eta'_{i,j}}) \cdot e(g_1^{-\hat{\zeta}'}, w_1) \cdot e(g_1, h_0)^{-\hat{\zeta}'\eta'_{i,j}} = e(C_{\tilde{w}_{2,i}}, h_0) \cdot e(g_2, h_0)^{u_j} \cdot e(f_1^{-r_{\tilde{w}_{2,i}}}, h_0) \cdot e(g_0, h_0) \cdot e(g_1^{\theta'}, h_0) \cdot e(g_1, h_0)^{-\hat{\zeta}'\eta'_{i,j}}$$

$$e(C_{\tilde{A}_{i,j}} g_1^{-\hat{\zeta}'}, w_1 h_0^{\eta'_{i,j}}) = e(C_{\tilde{w}_{2,i}} f_1^{-r_{\tilde{w}_{2,i}}} g_2^{u_j} g_1^{\theta'-\hat{\zeta}\eta'_{i,j}} g_0, h_0)$$

Setting $\tilde{A}_{i,j} = C_{\tilde{A}_{i,j}} g_1^{-\hat{\zeta}'}$, $\tilde{w}_{2,i} = C_{\tilde{w}_{2,i}} f_1^{-r_{\tilde{w}_{2,i}}}$ and $\zeta'_{i,j} = \theta' - \hat{\zeta}'\eta'_{i,j}$, we have $e(\tilde{A}_{i,j}, w_1 h_0^{\eta'_{i,j}}) = e(\tilde{w}_{2,i} g_2^{u_j} g_1^{\zeta'_{i,j}} g_0, h_0)$. For $w_1 = h_0^{\gamma_1}$ and $\tilde{w}_{2,i} = g_3^{\gamma_{2,i}}$, this implies $\tilde{A}_{i,j} = (g_0 g_1^{\zeta'_{i,j}} g_2^{u_j} g_3^{\gamma_{2,i}})^{1/(\gamma_1+\eta'_{i,j})}$. From the Eq. (3), we can show $R_{T,j} = (g_0 g_1^{\zeta''_{T,j}} g_2^{u_j} g_3^{T})^{1/(\gamma_1+\eta''_{T,j})}$ similarly. In addition, from the Eqs. (5), (6), we have $v = \hat{\zeta} \cdot r_{w_{2,i}}$, and we can transform the Eq. (4) to

$$e(C_{B_{t-1}} g_1^{-\hat{\zeta}}, C_{w_{2,i}} h_1^{-r_{w_{2,i}}} h_0^{\eta_{t-1}}) = e(g_0 g_1^{\theta-\hat{\zeta}\cdot\eta_{t-1}} g_2^{x} g_3^{S_{t-1}} g_4^{rep_{t-1}}, h_0).$$

Setting $B_{t-1} = C_{B_{t-1}} g_1^{-\hat{\zeta}}$, $w_{2,i} = C_{w_{2,i}} h_1^{-r_{w_{2,i}}}$ and $\zeta_{t-1} = \theta - \hat{\zeta} \cdot \eta_{t-1}$, we have $e(B_{t-1}, w_{2,i} h_0^{\eta_{t-1}}) = e(g_0 g_1^{\zeta_{t-1}} g_2^{x} g_3^{S_{t-1}} g_4^{rep_{t-1}}, h_0)$. For $w_{2,i} = h_0^{\gamma_{2,i}}$, this implies $B_{t-1} = (g_0 g_1^{\zeta_{t-1}} g_2^{x} g_3^{S_{t-1}} g_4^{rep_{t-1}})^{1/(\gamma_{2,i}+\eta_{t-1})}$. Finally, from Eq. (7), we obtain $e(C_{\tilde{w}_{2,i}} f_1^{-r_{\tilde{w}_{2,i}}}, h_0) = e(g_3, C_{w_{2,i}} h_1^{-r_{\tilde{w}_{2,i}}})$, which means that the discrete log $\gamma_{2,i}$ of $\tilde{w}_{2,i} = C_{\tilde{w}_{2,i}} f_1^{r_{\tilde{w}_{2,i}}}$ with base g_3 is the same as the discrete log of $w_{2,i} = C_{w_{2,i}} h_1^{-r_{\tilde{w}_{2,i}}}$ with base h_0. □

As mentioned in [7], $A'_\ell = (f_0 f_1^{\xi})^{1/(\gamma_{0,\ell}+rep_{t-1})}$ is modified from the original BB signature $A_\ell = f_0^{1/(\gamma_{0,\ell}+rep_{t-1})}$, but forging A'_ℓ can be reduced to forging the BB signature.

Here, we prove the security of the proposed system.

Theorem 1: *The proposed scheme is reputation unforgeable, under the security of BB signatures, BBS+ signatures, commitments, and digital signatures in the random oracle model.*

Proof: Assume an adversary $\mathcal{A}$ that wins the reputation unforgeability game with non-negligible probability. In the game, we can extract the proved secrets from SPKs in each **Show** protocol via $O_{CU-Show}$. For the winning game, we consider the following four cases.

- **Case 1:** In a **Show** protocol, an extracted BBS+ signature $((\tilde{A}_{i,j}, \eta'_{i,j}, \zeta'_{i,j}), (R_{T,j}, \eta''_{T,j}, \zeta''_{T,j})$, or $(B_{t-1}, \eta_{t-1}, \zeta_{t-1}))$ has not been issued by CA or RSUs.
- **Case 2:** In a **Show** protocol, an extracted (variant of) BB signature A'_ℓ has not been issued by CA.
- **Case 3:** In a **Register** or **Show** protocol, the extracted committed values compromise the binding property of commitments.
- **Case 4:** In an **Authentication** protocol, a digital signature $sig_{i,\tau}$, or $pcpkcert_i$ is forged.

When all of Cases 1–4 do not happen, any corrupted vehicle cannot prove the incorrect level ℓ s.t. the concrete reputation value rep_{t-1} is not included in the range, as follows. The reputation value rep_{t-1} of each vehicle is ensured by a BBS+ signature $\tilde{\sigma}_{t-1} = (B_{t-1}, \eta_{t-1}, \zeta_{t-1})$ on x, S_{t-1}, rep_{t-1} using the RSU i's (or CA's) key pair $(\gamma_{2,i}, w_{2,i})$. The BBS+ signature $(\tilde{A}_{i,j}, \eta'_{i,j}, \zeta'_{i,j})$ on $(u_j, \gamma_{2,i})$ using the CA's key pair (γ_1, w_1) implies that the proven key pair $(\gamma_{2,i}, w_{2,i})$ is ensured by the CA, i.e., it is not a forged key pair, and a node u_j is on the path to the RSU i. Due to the BBS+ signature $(R_{T,j}, \eta''_{T,j}, \zeta''_{T,j})$ on (u_j, T), the node u_j is a cover node at time T, i.e., the RSU i is not revoked. Furthermore, a variant of BB signature A'_ℓ on the rep_{t-1} using the key pair $(\gamma_{0,\ell}, w_{0,\ell})$ implies that the proved rep_{t-1} belongs to the integer range of reputation level ℓ. In each **Authentication**, due to digital signatures $authsig$, $sig_{i,\tau}$, and $pcpkcert_i$, the score s is added to $\tilde{s}$ for the corrupted vehicle via the pseudonym $pseu_\tau$. Since the SPK in **Show** proves $C'_{m,t} = g_1^{\zeta'_t} g_2^x g_3^{S_t} g_4^{rep_{t-1}}$, for the correct $rep_t = rep_{t-1} + \tilde{s}_\tau$, the new BBS+ signature (B_t, η_t, ζ_t) on x, S_t, rep_t is issued in **Update** protocol. In **Show** protocol, S_{t-1} is checked for the freshness of the proved certificate, and thus each rep_t is correctly updated. Therefore, for the correct reputation $\tilde{rep}_{t-1}$ which is added from $\tilde{s}_{\tau_1}, \ldots, \tilde{s}_{\tau_k}$ where $\tilde{s}_{\tau_i}$ is the total score for each $pseu_{\tau_i}$ assigned to the vehicle at the interval τ_i, the vehicle cannot prove any inappropriate level ℓ s.t. $\tilde{rep}_{t-1}$ is not inculded in the integer range of the level ℓ.

Therefore, one of Case 1–4 happens with some non-negligible probability. Case 1 (resp., 2–4) can be reduced to an adversary $\mathcal{A}_{\text{BBS+}}$ (resp., $\mathcal{A}_{\text{BB}}$, $\mathcal{A}_{\text{com}}$, and $\mathcal{A}_{\text{DS}}$) against BBS+ signatures (resp., BB signatures, commitments (binding property), and digital signatures). The reductions are similar to the proof of **Lemma 2** shown in the journal version [23] of the underlying system [7]. Here, we show the outline and key points, as follows.

- $\mathcal{A}_{\text{BBS+}}$: Given a public key of BBS+ signatures, generate other parameters in **Setup** and run $\mathcal{A}$ on spk, where in case of the reduction to $(\tilde{A}_{i,j}, \eta'_{i,j}, \zeta'_{i,j})$, the BBS+ signature is obtained via the BBS+ signing oracle. For oracles where the target type of BBS+ signature $((R_{T,j}, \eta''_{T,j}, \zeta''_{T,j})$, or $(B_{t-1}, \eta_{t-1}, \zeta_{t-1}))$ is issued, extract the secrets from the SPKs if needed, and access the BBS+ signing oracle to obtain the BBS+ signature. Then, in Case 1, a non-issued BBS+ signature is extracted in a **Show** protocol, which is outputted as $\mathcal{A}_{\text{BBS+}}$.
- $\mathcal{A}_{\text{BB}}$: This case is the same as the proof in the underlying system [23], since the construction is the same. Given a public key of BB signatures, randomly guess level $\tilde{\ell}$, and obtain BB signatures $A_{\tilde{\ell},R_{\tilde{\ell},k}}$ via the BB signing oracle. Generate other BB signatures $A_{\ell,R_{\tilde{\ell},k}}$ with $\ell \neq \tilde{\ell}$ and other parameters as in **Setup**, and run $\mathcal{A}$ on spk, where each oracle is addressed as in the real protocols. Then, in Case 2, a variant of non-issued BB signature is extracted in a **Show** protocol. Similarly to the proof in the underlying system [23], it is transformed to a non-issued BB signature, which is outputted as $\mathcal{A}_{\text{BB}}$.
- $\mathcal{A}_{\text{com}}$: Given public parameters of commitments, generate other parameters in **Setup** and run $\mathcal{A}$ on spk, where each oracle is addressed as in the real protocols except that the committed values are extracted from SPKs. Check the consistency between **Update** and **Show** in the same commitment. If

inconsistency happens, the collision values are outputted as $\mathcal{A}_{\mathrm{com}}$, which compromises the binding property of commitments.

- $\mathcal{A}_{\mathrm{DS}}$: Given a public key of the digital signatures, use the public key as $pcpk_i$ or $pcpk_{CA}$ according to the target type of signatures, generate other parameters in **Setup**, and run $\mathcal{A}$ on spk, where in case of the reduction to $pcpkcert_i$, the digital signatures is obtained via the signing oracle. Each oracle is addressed as in the real protocols except the following. When the target type of digital signature is $sig_{i,\tau}$, for oracles where $sig_{i,\tau}$ is issued, access the signing oracle to obtain the signature. Then, in Case 4, a non-issued digital signature is used in an **Authentication** protocol, which is outputted as $\mathcal{A}_{\mathrm{DS}}$. □

Theorem 2: *The proposed scheme is misauthentication resistant, under the security of digital signatures.*

Proof: Assume an adversary $\mathcal{A}$ that wins the misauthentication resistance game with non-negligible probability. In this game, $\mathcal{A}$ is not issued $psert_\tau$ for pseudonym $pseu_\tau$, but $\mathcal{A}$ is accepted in $O_{CV-Auth}$ in the interval τ. Thus, since $\mathcal{A}$ successfully forges the digital signature $sig_{i,\tau}$ or $pcpkcert_i$ in $O_{CV-Auth}$, we can construct an adversary $\mathcal{A}_{\mathrm{DS}}$ against the digital signatures, as in the proof of Theorem 1. □

Theorem 3: *The proposed scheme is RSU revocable, under the security of BBS+ signatures in the random oracle model.*

Proof: Assume an adversary $\mathcal{A}$ that wins the RSU revocability game with non-negligible probability. In the game, we can extract the proved secrets from SPKs in each **Show** protocol via $O_{CU-Show}$. In the RSU revocability game, for the extracted BBS+ signature $(\tilde{A}_{i,j}, \eta'_{i,j}, \zeta'_{i,j})$ on node u_j, the issuing RSU i is revoked. Thus, for the extracted BBS+ signature $(R_{T,j}, \eta''_{T,j}, \zeta''_{T,j})$ on node u'_j, the revocation based on CS method means $u_j \neq u'_j$, if these extracted BBS+ signatures were issued by the honest CA. However, since the verification for the SPKs is accpeted, we have $u_j = u'_j$, due to the soundness of the SPK for the same secret. This means that either of the extracted BBS+ signatures is forged by $\mathcal{A}$. Thus, we can construct an adversary $\mathcal{A}_{\mathrm{BBS+}}$ against the BBS+ signatures, as in the proof of Theorem 1. □

Theorem 4: *The proposed scheme is anonymous in the random oracle model.*

Proof: In the random oracle model, SPKs can be simulated. Let **Game 0** be the original anonymity game. Consider **Game 1** where the followings are modified from **Game 0**: In the **Show** protocols for O_{LoR} request, as honest vehicles k^*_b and $k^*_{\neg b}$, execute the zero-knowledge simulations instead of the SPKs, and replace the commitments $C_{A_\ell}, C_{\tilde{A}_{i,j}}, C_{\tilde{w}_{2,i}}, C_{R_{T,j}}, C_{B_{t-1}}, C_{w_{2,i}}, C'_{m,t}, C_{\hat{\zeta}}$, and C_v with random elements of the corresponding group.

Then, consider the responses to O_{LoR} request in **Game 1**. In the **Show** protocols, the zero-knowledge simulation, the replaced random elements, and one-time used random S_{t-1} are sent to $\mathcal{A}$, but these values have no information on b. Furthermore, $sig_{i,\tau}$ is one-time digital signature for one-time public key opk_τ only during the interval τ. Therefore, the probability that $\mathcal{A}$ correctly guesses b is 1/2.

On the other hand, due to the zero-knowledge-ness of the SPK and the perfect hiding of the commitments, both games are indistinguishable. Therefore, in **Game 0**, the probability that $\mathcal{A}$ correctly guesses b is also 1/2. □

6 Efficiency

In this section, we discuss the efficiency of our system, compared to BARS [6]. In BARS, two authorities (CA and LEA) cooperatively update the reputation of each vehicle and issue the updated certificate to the vehicle. Thus, the centralized issuing process involving two authorities is a bottleneck. In our system, the issuing process is distributed, which is executed between a vehicle and the nearest RSU, and thus the bottleneck is resolved.

On the other hand, although BARS adopts an ordinary digital signature scheme for the public key and the certificate, our system utilizes the pairing-based computations in **Show** protocol, whose costs are heavier than the ordinary public key cryptosystems such as RSA and ECC. In Table 2, we present the computational costs of a vehicle and an RSU in the **Show** protocol, measured as the number of pairings and exponentiations on $\mathbb{G}_1$, $\mathbb{G}_2$, and $\mathbb{G}_T$, excluding precomputable operations. In [22], implementation results of a pairing-based group signature scheme are presented, using the Barreto–Lynn–Scott (BLS) curve with embedding degree 12 over a 455-bit prime field to achieve 128-bit security. The results show that the computation time of an exponentiation on a $\mathbb{G}_1$ element is approximately 0.25 ms, while those of exponentiation on $\mathbb{G}_2$ and $\mathbb{G}_T$ elements and a pairing are about 0.53, 0.74, and 1.45 ms, respectively, on a Core i7-7700K (4.20 GHz) CPU. Based on these measurements, the processing time required by a vehicle in **Show** protocol is approximately 30 ms, while that of an RSU is about 43 ms. Although the computational costs are heavier than BARS, the computations of commitments and SPK in the vehicle can be pre-computed before **Show** protocol, and the needed online computations are only multiplications in the responses in SPK.

Table 2: Computational costs of our **Show** protocol

	Number of exponentiations			**Number of pairings**
	$\mathbb{G}_1$	$\mathbb{G}_2$	$\mathbb{G}_T$	
Vehicle	23	1	21	6
RSU	14	0	28	13

As for the data size submitted by a vehicle in **Show** protocol, the data contains 8 $\mathbb{G}_1$-elements, 1 $\mathbb{G}_2$-elements, and 24 $\mathbb{Z}_p$-elements. In the implementation results reported in [22], a $\mathbb{G}_1$-element is represented using 58 bytes, whereas a $\mathbb{G}_2$-element and a $\mathbb{Z}_p$-element are represented using 115 and 39 bytes, respectively. Using the results, the submitted data size is approximately 1500 bytes.

We emphasize that **Show** protocol is required only at the beginning of each interval. The interval length can be configured to practical values (e.g., several hours or a day), and the interval boundaries of vehicles can be offset so that **Show** protocols do not concentrate on a single RSU. During the interval, per-message authentication is performed using the same pseudonym-certificate mechanism as in existing RSU-assisted V2X systems such as BARS, and therefore the communication pattern and verification cost at RSUs remain unchanged from prior work. The SPK-related load is limited to a single proof of approximately 1.5 KB, most of whose computation can be performed offline by the vehicle; The verification cost at the RSU is estimated to be about 40 ms, based on timing measurements of the underlying cryptographic operations. Because the SPK is executed only once per interval and the per-message operations are identical to existing systems, we consider that the proposed system does not impose additional constraints on RSU coverage or handoff latency in high-mobility V2X environments, and that its scalability with respect to the number of RSUs and vehicles is unlikely to become a major concern.

7 Conclusion

In this paper, a distributed anonymous reputation system for V2X communication is proposed. The proposed system distributes the task to update the vehicles' reputation certificates to RSUs, in which the nearest RSU updates the certificate anonymously at the end of each interval. This approach resolves the bottleneck in the certificate update process and improves the scalability.

Our future works include the implementation of the proposed system, network-level performance evaluations under specific traffic and mobility models (e.g., NS-3 or SUMO), realizing the transparency based on blockchain, and a detailed reputation evaluation algorithm.

Acknowledgement: Not applicable.

Funding Statement: The authors received no specific funding for this study.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Shahidatul Sadiah and Toru Nakanishi; methodology, Shahidatul Sadiah and Toru Nakanishi; validation, Shahidatul Sadiah and Toru Nakanishi; formal analysis, Shahidatul Sadiah and Toru Nakanishi; investigation, Shahidatul Sadiah and Toru Nakanishi; writing—original draft preparation, Shahidatul Sadiah; writing—review and editing, Shahidatul Sadiah and Toru Nakanishi; supervision, Toru Nakanishi. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Not applicable.

Ethics Approval: Not applicable.

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

References

1. Li Q, Malip A, Martin KM, Ng S, Zhang J. A reputation-based announcement scheme for VANETs. IEEE Trans Veh Technol. 2012;61(9):4095–108.
2. Yang N. A similarity based trust and reputation management framework for VANETs. Int J Future Gener Commun Netw. 2013;6(2):25–34.
3. Jesudoss A, Raja SK, Sulaiman A. Stimulating truth-telling and cooperation among nodes in VANETs through payment and punishment scheme. Ad Hoc Netw. 2015;24:250–63.
4. Wang XO, Cheng W, Mohapatra P, Abdelzaher T. ARTSense: anonymous reputation and trust in participatory sensing. In: Proceedings of IEEE INFOCOM 2013; 2013 Apr 14–19; Turin, Italy. p. 2517–25.
5. Jaimes LMS, Ullah K, Moreira ES. ARS: anonymous reputation system for vehicular ad hoc networks. In: Proceedings of the 8th IEEE Latin-American Conference on Communications (LATINCOM); 2016 Nov 15–17; Medellin, Colombia. p. 1–6.
6. Lu Z, Liu W, Wang Q, Qu G, Liu Z. A privacy-preserving trust model based on blockchain for VANETs. IEEE Access. 2018;6:45655–64.
7. Sadiah S, Nakanishi T. An efficient anonymous reputation system for crowd sensing. In: Seventh International Symposium on Computing and Networking Workshops (CANDAR 2019 Workshops), WICS 2019; 2019 Nov 26–29; Nagasaki, Japan. p. 374–80.
8. Naor D, Naor M, Lotspiech J. Revocation and tracing schemes for stateless receivers. In: Advances in cryptology-CRYPTO 2001. LNCS 2139. Berlin/Heidelberg, Germany: Springer-Verlag; 2001. p. 41–62.
9. Sadiah S, Nakanishi T. A distributed anonymous reputation system for V2X communication. In: Proceedings of the IEEE International Conference on Consumer Electronics (ICCE); 2024 Jan 6–8; Las Vegas, NV, USA. p. 1–6.
10. Ahmed W, Wu D, Mukathe D. Blockchain-assisted privacy-preserving and context-aware trust management framework for secure communications in VANETs. Sensors. 2023;23(12):5766.
11. Feng X, Shi Q, Xie Q, Liu L. An efficient privacy-preserving authentication model based on blockchain for VANETs. J Syst Archit. 2021;117:102158.
12. Feng X, Cui K, Wang L, Liu Z, Ma J. PBAG: a privacy-preserving blockchain-based authentication protocol with global-updated commitment in IoVs. IEEE Trans Intell Transp Syst. 2024;25(10):13524–13545.
13. Hou B, Xin Y, Zhu H, Yang Y, Yang J. VANET secure reputation evaluation & management model based on double layer blockchain. Appl Sci. 2023;13(9):5733.

14. Fernandes CP, Montez C, Adriano DD, Boukerche A, Wangham MS. A blockchain-based reputation system for trusted VANET nodes Ad Hoc Networks. 2023;140:103071.
15. Boneh D, Boyen X. Short signatures without random oracles. In: Advances in cryptology-EUROCRYPT 2004. LNCS 3072. Berlin, Heidelberg: Springer-Verlag; 2004. p. 56–73.
16. Boneh D, Boyen X, Shacham H. Short group signatures. In: Advances in cryptology-CRYPTO 2004. LNCS 3152. Berlin/Heidelberg, Germany: Springer-Verlag; 2004. p. 41–55.
17. Au MH, Susilo W, Mu Y. Constant-size dynamic k-TAA. In: Security and cryptography for networks (SCN 2006). LNCS 4116. Berlin/Heidelberg, Germany: Springer-Verlag; 2006. p. 111–25.
18. Camenisch J, Drijvers M, Lehmann A. Anonymous attestation using the strong diffie-Hellman assumption revisited. In: Trust and trustworthy computing (TRUST 2016). Cham, Switzerland: Springer; 2016. p. 1–20.
19. Damgård I. On Σ-Protocols. [cited 2025 Dec 18]. Available from: http://www.daimi.au.dk/~ivan/Sigma.pdf.
20. Libert B, Peters T, Yung M. Group signatures with almost-for-free revocation. In: Advances in cryptology-CRYPTO 2012. LNCS 7417. Berlin/Heidelberg, Germany: Springer-Verlag; 2012. p. 571–89.
21. Ohara K, Emura K, Hanaoka G, Ishida A, Ohta K, Sakai Y. Shortening the Libert-Peters–Yung revocable group signature scheme by using the random oracle methodology. IEICE Trans Fundam. 2019;102-A(9):1101–17.
22. Emura K, Hayashi T. A revocable group signature scheme with scalability from simple assumptions. IEICE Trans Fundam. 2020;103-A(1):125–40.
23. Sadiah S, Nakanishi T. An efficient anonymous reputation system for crowdsensing. J Inf Process. 2022;30:694–705.