Privacy-Preserving Cloud-based Road Condition Monitoring with Source Authentication in VANETs

Yujue Wang, Yong Ding, Qianhong Wu,Yongzhuang Wei, Bo Qin, and Huiyong Wang

Abstract—The connected vehicular ad-hoc network (VANET) and cloud computing technology allows entities in VANET to enjoy the advantageous storage and computing services offered by some cloud service provider. However, the advantages do not come free since their combination brings many new security and privacy requirements for VANET applications. In this article, we investigate the cloud-based road condition monitoring (RCoM) scenario, where the authority needs to monitor real-time road conditions with the help of a cloud server so that it could make sound responses to emergency cases timely. When some bad road condition is detected, e.g., some geologic hazard or accident happens, vehicles on site are able to report such information to a cloud server engaged by the authority. We focus on addressing three key issues in RCoM. First, the vehicles have to be authorized by some roadside unit before generating a road condition report in the domain and uploading it to the cloud server. Second, to guarantee the privacy against the cloud server, the road condition information should be reported in ciphertext format, which requires that the cloud server should be able to distinguish the reported data from different vehicles in ciphertext format for the same place without compromising their confidentiality. Third, the cloud server and authority should be able to validate the report source, i.e., to check whether the road conditions are reported by legitimate vehicles. To address these issues, we present an efficient RCoM scheme, analyze its efficiency theoretically, and demonstrate the practicality through experiments.

Index Terms—Data privacy, vehicular ad hoc networks, VANET, cloud computing, authentication, auditability.

1 Introduction

VEHICULAR ad-hoc network has been envisioned as a promising technology to improve the travel efficiency and safety of transportation systems [1]. In VANET, each vehicle with an embedded on-board unit is able to collect and communicate the current road/traffic condition information at some location with others with the help of distributed roadside units (RUs). For example, vehicles may broadcast warning signals to the nearby vehicles (especially to the behind ones) when detecting some accident, congestion, jam, etc. [2]. In this way, every nearby recipient vehicle would be able to get better awareness of driving environment and change driving plan if needed. Indeed, the VANET technology has attracted great attentions from both industry and academia [3], [4].

- *Y. Wang, Y. Ding and Y. Wei are with the Guangxi Key Laboratory of Cryptography and Information Security, School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin, 541004, China. Y. Wang is also with the State Key Laboratory of Cryptology, P. O. Box 5159, Beijing, 100878, China*
  E-mail: yjwang@guet.edu.cn, stone_dingy@126.com, walker_wyz@guet.edu.cn
- *Q. Wu is with the School of Electronic and Information Engineering, Beihang University, Beijing 100191, China, and with Science and Technology on Information Assurance Laboratory, Beijing, China.*
  E-mail: qianhong.wu@buaa.edu.cn
- *B. Qin is with Key Laboratory of Data Engineering and Knowledge Engineering, Ministry of Education, School of Information, Renmin University of China, Beijing, China.*
  E-mail: bo.qin@ruc.edu.cn
- *H. Wang is with the School of Mathematics and Computing Science, Guilin University of Electronic Technology, Guilin, 541004, China.*
  E-mail: why6082015@gmail.com

Manuscript received XXXX, 2018

Many efforts have been focused on addressing the security issues in VANETs, for example, to guarantee the authentication, non-repudiation, integrity and privacy of messages [5]. Especially, message authentication in VANETs has been well studied. Chen, Ng and Wang [6] designed a threshold anonymous announcement system, where a recipient vehicle accepts the reported road/traffic information when at least τ different (anonymous) vehicles report the same information. The recipient can also validate whether the received information are sent from legitimate sources. Thus, their solution achieves reliable road/traffic information exchange. By employing the technique of direct anonymous attestation supporting user-controlled linkability of signatures, their proposal also provides distinguishability of origin by allowing the linking of signatures if a signer signs a message multiple times.

However, in real-world application scenario, the trusted authority (TA) in VANET may need to monitor road conditions in real-time so that it could respond quickly in emergency cases. Indeed, vehicles or RUs can be enabled to directly report the collected road conditions to TA. When τ or more road condition reports for the same location are received, where τ denotes the threshold in the monitoring system, TA takes it as an emergency case and then makes response. However, this approach requires TA to equip with powerful computing and storage resources (e.g., hardware and software resources), which would bring unaffordable costs to TA. Recently, Liu et al. [7] investigated a similar traffic monitoring problem in a vehicle-to-infrastructure framework. In their scheme, the distributed RUs forward reports from individual vehicles to the traffic monitoring

center. Therefore, the vehicles are unable to directly report the detected traffic information, and the traffic monitoring center must be powerful enough to process all reported traffic information in real-time as well as to maintain the report data.

Notice that the connected vehicular cloud computing (CVCC) has recently been brought out [8], [9], which combines VANET with the cloud computing technology. With CVCC, all entities in VANET are able to enjoy the advantages of cloud computing, that is, the computing and storage services offered by some cloud service provider. In this paper, we investigate the above mentioned road condition monitoring scenario in the CVCC framework. When a vehicle gets into the domain of some RU, it should interact with such RU to obtain a token. If some road condition is collected within the domain of RU, the vehicle generates a report using the issued token, and uploads the report to the cloud server for processing.

We observe that there remain three critical issues to be addressed. First, the cloud server may not be trustworthy [10] and may be curious about the contents of the stored road condition reports. Thus, to protect the confidentiality, all reports should be stored at the cloud server in ciphertext format such that only the root authority is able to decrypt reports. Second, the uploaded reports are in fact big data with the high volume and velocity characteristics, thus they cannot be directly forwarded to the trusted root authority for processing due to its limited storage and computing capabilities. It is preferable to allow the cloud server to undertake the most computations, for example, to identify which road domain has been reported for more than τ times for the same condition information. This functionality requires that the cloud server should be able to compare the ciphertext reports without decrypting their values. Note that existing homomorphic encryption [11] is not applicable here in big data scenario due to its inefficiency. Third, some malicious vehicles may impersonate others to upload forged road conditions. Thus, the source of road condition report should be verifiable by the cloud server and authority.

1.1 Our contributions

To address the above issues for secure road condition monitoring in the CVCC framework, this paper proposes a *privacy-preserving cloud-based road condition monitoring system with source authentication* (RCoM). We design a concrete RCoM scheme in bilinear groups, which provides the following functionalities.

- **Authorized reporting.** A vehicle can collect the real-time road condition information and encrypt it with the root authority's public key, its secret key and the token issued by the administrative RU before uploading to the cloud server, where the vehicle is currently running in the domain of RU. Without a valid token from some RU, the vehicle is unable to generate a road condition report without being caught. The report in ciphertext format can be partially validated by the cloud server without decrypting its value, in this way to filter out and discard invalid reports.
- **Privacy-preserving monitoring.** The cloud server is engaged to perform much of the monitoring work. Specifically, all reports are grouped into equivalence classes by the cloud server, where the reports in the same equivalence class are for the same road domain and the same road condition information. When the cloud server receives a new report from some vehicle, it compares this report with existing equivalence classes. In fact, only one report in each equivalence class needs to be compared. If some comparison returns true, then the fresh report is inserted into the corresponding equivalence class. When some equivalence class contains at least τ reports, then the cloud server informs the root authority to process immediately. The root authority only needs to decrypt one ciphertext report to obtain the road condition information, and takes actions if needed.
- **Source authentication.** The entities in RCoM system such as sub-authorities, vehicles and roadside units are recognized with their identities. Thus, our RCoM construction does not rely on complicated cryptographic certificates. When a vehicle or a RU receives a secret key from some sub-authority, it is able to validate the key using the identity of the administrative sub-authority. Moreover, the cloud server can check whether a report is generated by the claimed vehicle, and the root authority can verify whether the report is generated with a token issued by the claimed RU.

The security of our RCoM scheme is analyzed under the Computational Diffie-Hellman assumption, which implies that malicious vehicles cannot forge a valid road condition report under the selective identity and chosen message attacks, and the report enjoys one-way confidentiality under adaptive chosen ciphertext attack against the cloud server if the order of the message space is larger than any polynomial function. We also conduct extensive experiments of our RCoM scheme. Both theoretical analyses and experimental results demonstrate the practicality of our RCoM proposal in applications.

1.2 Related techniques

In this section, we briefly review some related cryptographic techniques in secure VANET applications and secure cloud storage. We also review some related cryptographic techniques such as encryption with equality test on ciphertexts and delegated/authorized data processing.

Secure VANET applications. In [12], Wu et al. presented a contributory broadcast encryption scheme, which allows vehicles in VANET to negotiate a common public encryption key, and meanwhile each vehicle can hold a decryption key. In this way, only these vehicles are able to exchange traffic information securely with their decryption keys. Guo et al. [13] designed a secure mechanism to collect traffic information through the Internet of Vehicles. Particularly, there is a trusted certification authority to issue certificates for all vehicles. For two types of traffic information such as business data and confidential data, the former can be transferred in plaintext format, whereas the later has to be encrypted.

Sucasas et al. [14] addressed the issue of guaranteeing vehicle location privacy when authenticating the received messages in intelligent transportation systems. They proposed an autonomous privacy-preserving authentication scheme such that the vehicles are able to renew their pseudonyms without interacting with the trusted authority. Malhi and Batra [15] considered the same authentication problem, where pseudonyms are used to achieve anonymous communication. Also, they designed a new privacy-preserving signature scheme for inter-vehicle communication, where the verification procedure is further improved to support aggregate verification and enhanced with bloom filters to prevent message drop in heavy busy traffic hour. Liu et al. [16] investigated the issue of user privacy protection in data aggregation without trusted third party, and designed an efficient scheme in bilinear groups.

Secure cloud storage. Security issues such as data auditability, confidentiality and provenance for cloud storage have been well studied in recent years [10], [17], [18]. *Provable Data Possession* [19], *Proofs of Retrievability* [20], and *Proofs of Storage* [21], [22] are introduced to guarantee the integrity of outsourced data in clouds. All these primitives allow an auditor to audit the outsourced data without retrieving the entire version from the cloud server. Note that compared to Provable Data Possession and Proofs of Storage, Proofs of Retrievability [23] also employs erasure code to support data recoverability to some extent when outsourced data was partially lost or corrupted.

Delegated/Authorized data processing. Wang, He and Tang [24] studied proxy-oriented data outsourcing in clouds, where the data owner can authorize a proxy to process her data and upload to the cloud server. Wang et al. [25] considered the similar problem scenario, where the authorized proxy can directly process data with his private key, while he has to generate a proxy key before processing data in [24]. Yang et al. [26] and He et al. [27] investigated data authentication and integrity protection in automatic dependent surveillance-broadcast system (ADS-B). In their proposed three-level framework, the top level authority (e.g., the International Civil Aviation Organization) issues secret keys for the second-level authorities (i.e., the airlines), and every airline is responsible for registration for its affiliated aircrafts. Different from [26], [27], Baek et al. [28] proposed a confidentiality framework for ADS-B by designing an efficient staged identity-based encryption scheme.

Equality test on ciphertexts. Public key encryption with equality test on ciphertexts allows users (e.g., the cloud server) to check whether two ciphertexts encrypt the same plaintext [29], where those ciphertexts may be produced under different public keys. Considerable efforts have been made to achieve controlled equality test on ciphertexts [30], [31], [32], [33], [34], [35], such that only an authorized/delegated tester is able to compare ciphertexts. Particularly, if the tester in [30], [31], [32] gets the authorization from two data owners, then it would be able to compare their ciphertexts. Recently, the functionality of equality test on ciphertexts has been used in achieving deduplication on encrypted data in clouds [36], [37], [38].

1.3 Paper organization

The remainder of this paper is organized as follows. We introduce the RCoM system architecture and security requirements in Section 2. The RCoM framework and the security model are formalized in Section 3. Section 4 introduces our RCoM scheme and Section 5 proves its security. In Section 6, we analyze our RCoM scheme both theoretically and empirically. Finally, Section 7 concludes the paper.

2 Architecture and Security Requirements of RCoM System

In this section, we formalize the architecture of RCoM and summarize its security requirements.

2.1 RCoM architecture

The RCoM system consists of five types of entities (see Figure 1), that is, a root authority (RA), many sub-authorities (SAs), many roadside units (RUs), a cloud server, and many vehicles. As in VANET, RA, SAs and RUs are the trusted participants. In real-world applications, RA can be the Department of Transportation. The goal of RA is to monitor the real-time road conditions with the help of a cloud server, so that it could make timely response to emergency cases. The cloud server is maintained by some cloud service provider (CSP), which has significant computing and storage resources, and provides on-the-move access to outsourced data (i.e., road condition information) to end users. In RCoM, the cloud server is a curious entity, which is engaged by RA to maintain and process all road information collected by vehicles.

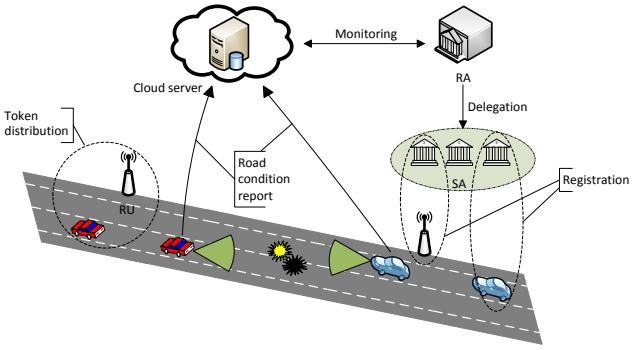

Fig. 1. System architecture of RCoM.

Initially, to join the system, all vehicles must be authorized by RA, in this way to get private keys extracted from their respective identities. Note that the number of vehicles may be significantly large. Thus, RA needs to delegate SAs to authorize vehicles and roadside units. In practice, each SA has a disjoint management region, which is only responsible for authorizing vehicles in its region. Also, each vehicle can only be authorized by one SA. Every vehicle can collect and report real-time road condition if a dangerous road condition is detected. Similarly, every RU also gets a private key from some SA, which is extracted from its identity. In applications, SAs can be separated into two categories to respectively deal with the registration of vehicles and RUs. In RCoM, all roads are divided into disjoint sections. For

ease of presentation, each road section is represented by the identity of its administrative RU in this paper. When some vehicle gets into a new section, the administrative RU issues a token which enables the vehicle to report the detected road condition information in this section.

Vehicles on road can collect road condition information and report to RA. All poor and dangerous road conditions (e.g., accidents, mudslides, etc.) have to be reported. Note that all these reports are uploaded to the cloud server rather than being directly sent to RA due to its limited computing and storage capabilities. Also, they should be outsourced in ciphertext format such that only RA is able to recover the information of road conditions. The cloud server is allowed to compare the reported ciphertexts and inform RA to make sound response when the number of the same road conditions reported for the same road section is larger than some predefined threshold.

Note that the RCoM system may confront some traditional attacks such as denial of service attack. However, since all registered vehicles direct upload road condition information to the cloud server, RA would not become a bottleneck to the RCoM system. The cloud server engaged by RA is maintained by some CSP. In real-world applications, a rational CSP would implement certain measures to protect the cloud server from traditional attacks. Thus, the remainder of this paper do not focus on the techniques to resist such traditional attacks.

2.2 Functionalities and design goals

As a common practice in designing VANET related protocols, it is assumed that every vehicle contains a tamper-resistant black box, which can keep data and perform basic cryptographic operations securely. We identify the functionalities and design goals of the RCoM system as follows.

Privacy of road conditions: The cloud server may be curious about the contents of maintained data, but does not collude with vehicles or RUs. Thus, all reports of road conditions should be sent to the cloud server in ciphertext format to guarantee the privacy, where only RA can decrypt the ciphertexts. The road condition information should specify the location (i.e., road section or roadside unit) where the condition is collected. Hence, all information regarding the road section and roadside unit must also be kept secret against the cloud server.

Source authentication and token verifiability: In the RCoM system, malicious vehicles may impersonate some other vehicle in reporting road conditions, and may forge a token from some RU. Its goal is to fool RA to accept a false report without being caught. To resist this attack, the sources of reports should be verifiable by the cloud server without interacting with RA. If they do not satisfy the verification conditions, then they would be discarded by the cloud server, which means they would not be further processed and presented to RA.

Road condition classification: In practice, many vehicles may report the same road condition for the same place in a reasonable period, which requires the cloud server to be able to distinguish them from the others. In other words, the cloud server must be able to check whether the ciphertexts encrypt the same road condition information and group them into equivalence classes accordingly.

3 DEFINITIONS

3.1 Framework of RCoM system

Formally, a RCoM system consists of eight polynomial-time computable algorithms/protocols, that is, Setup, SAdlg, VHreg, RUreg, TKdis, RCrep, CLpro, and RApro.

- $\mathsf{Setup}(1^\kappa) \rightarrow (\mathsf{par}, msk)$: On input 1^κ where κ is a security parameter, the RCoM system setup algorithm, which is run by the root authority RA, generates the public parameter par for the system and a master secret key *msk* for itself.
- $\mathsf{SAdlg}(\mathsf{par}, msk, SA_i) \rightarrow ssk_i$: On input the public parameter par, the master secret key *msk* and the identity of some sub-authority SA_i, the delegation algorithm, which is run by RA, generates a secret key ssk_i for SA_i. Sub-authority SA_i should be able to validate ssk_i before accepting it as secret key.
- $\mathsf{VHreg}(\mathsf{par}, SA_i, ssk_i, V_j) \rightarrow vsk_j$: On input the public parameter par, the identity SA_i and secret key ssk_i of some sub-authority, and the identity of some vehicle V_j, the vehicle registration algorithm, which is run by SA_i, generates a secret key vsk_j for V_j. Vehicle V_j should be able to validate vsk_j before accepting it as secret key.
- $\mathsf{RUreg}(\mathsf{par}, SA_\ell, ssk_\ell, RU_l) \rightarrow rsk_l$: On input the public parameter par, the identity SA_ℓ and secret key ssk_ℓ of some sub-authority, and the identity of some roadside unit RU_l, the roadside unit registration algorithm, which is run by SA_ℓ, generates a secret key rsk_l for RU_l. Roadside unit RU_l should be able to validate rsk_l before accepting it as secret key.
- $\mathsf{TKdis}(\mathsf{par}, (SA_i, V_j, vsk_j), (SA_\ell, RU_l, rsk_l)) \rightarrow \mathcal{T}_l/\perp$: On input the public parameter par, the token distribution protocol, which is jointly run by vehicle V_j and roadside unit RU_l with (SA_i, vsk_j) and (SA_ℓ, rsk_l), respectively, outputs an authentication tuple $\mathcal{T}_l$ including a token θ_l if both sides are honest, or $\perp$ otherwise. Here, SA_i and SA_ℓ denote the administrative sub-authorities of V_j and RU_l, respectively.
- $\mathsf{RCrep}(\mathsf{par}, vsk_j, \mathcal{T}_l, RU_l, I) \rightarrow (U, W)$: On input the public parameter par, the secret key vsk_j of vehicle V_j, an authentication tuple $\mathcal{T}_l$, a roadside unit identity RU_l and some road condition information I, the road condition report algorithm, which is run by vehicle V_j, outputs a ciphertext U and a tuple W.
- $\mathsf{CLpro}(\mathsf{par}, U, W) \rightarrow \{0, 1\}$: On input the public parameter par and a pair of (U, W), the cloud processing algorithm, which is run by the cloud server, outputs "1" if the pair (U, W) can be inserted into some group; otherwise it outputs "0".
- $\mathsf{RApro}(\mathsf{par}, msk, U, W) \rightarrow (RU_l, I)$: On input the public parameter par, the master secret key *msk* and a pair of (U, W), the RA processing algorithm, which is run by the root authority, outputs a decrypted pair of (RU_l, I).

A secure RCoM scheme must be *sound* in the sense that, if all involved entities are honest, then every algorithm/protocol would not output a symbol that denotes an error. Formally, for a security parameter $\kappa \in \mathbb{N}$ and

any $(\mathsf{par}, msk) \leftarrow \mathsf{Setup}(1^\kappa)$, the following conditions are satisfied:

- For any secret key $ssk_i \leftarrow \mathsf{SAdlg}(\mathsf{par}, msk, SA_i)$ issued by the RA, it can be validated as true and accepted by the sub-authority SA_i;
- For any secret key $vsk_j \leftarrow \mathsf{VHreg}(\mathsf{par}, SA_i, ssk_i, V_j)$ issued by the SA_i, it can be validated as true and accepted by the vehicle V_j;
- For any secret key $rsk_l \leftarrow \mathsf{RUreg}(\mathsf{par}, SA_\ell, ssk_\ell, RU_l)$ issued by the SA_ℓ, it can be validated as true and accepted by the roadside unit RU_l;
- For any vehicle V_j and any road section administrated by some roadside unit RU_l, the token distribution protocol $\mathsf{TKdis}(\mathsf{par}, (SA_i, V_j, vsk_j), (SA_\ell, RU_l, rsk_l))$ would output a valid authentication tuple $\mathcal{T}_l$;
- For any road condition information I collected at RU_l and any vehicle V_j with authentication tuple $\mathcal{T}_l$, both $\mathsf{CLpro}(\mathsf{par}, U, W) = 1$ and $(RU_l, I) \leftarrow \mathsf{RApro}(\mathsf{par}, msk, U, W)$ hold, where $(U, W) \leftarrow \mathsf{RCrep}(\mathsf{par}, vsk_j, \mathcal{T}_l, RU_l, I)$.

3.2 Formal security definitions

We present formal security definitions to capture the *unforgeability of ciphertexts against selective identity and chosen message attack* (UF-SI-CMA) launched by some malicious vehicle, and the *one-way confidentiality under adaptive chosen ciphertext attack* (OW-CCA2) against curious cloud server. Note that the cloud server should be able to compare the ciphertexts (i.e., encrypted road condition reports) from vehicles, so as to group them into equivalence classes, which means the ciphertexts in RCoM are distinguishable. Thus, the RCoM system cannot offer indistinguishability for road condition reports under chosen plaintext/ciphertext attacks. In [29], Yang et al. have discussed that ciphertext comparability and indistinguishability are irreconcilable, and indistinguishability-based security notions are not applicable to encryption schemes with ciphertext comparability.

We first consider the case where malicious vehicles may collude to forge a road condition report (U^*, W^*). Let $\mathcal{A}$ be a probabilistic polynomial-time (PPT) adversary, who plays the following game with a challenger $\mathcal{C}$ and tries to forge a valid pair (U^*, W^*).

Setup: The adversary $\mathcal{A}$ picks a target sub-authority SA^* and vehicle V^*, and sends them to $\mathcal{C}$. With a security parameter κ, the challenger $\mathcal{C}$ generates (par, msk) and publishes the public parameter par.

Queries: The adversary $\mathcal{A}$ can adaptively submit the following queries to $\mathcal{C}$.

- *Delegation:* The adversary $\mathcal{A}$ can ask for secret key for any sub-authority SA_i. The challenger $\mathcal{C}$ generates ssk_i and gives it to $\mathcal{A}$.
- *Vehicle registration:* In each query, the adversary $\mathcal{A}$ submits a pair (SA_i, V_j) to $\mathcal{C}$. If SA_i has not been queried for a secret key, then the challenger $\mathcal{C}$ first generates ssk_i. Then, the challenger $\mathcal{C}$ runs the vehicle registration algorithm and returns a secret key vsk_j of V_j with regard to SA_i.
- *Roadside unit registration:* In each query, the adversary $\mathcal{A}$ submits a pair (SA_ℓ, RU_l) to $\mathcal{C}$. If SA_ℓ has not been queried for a secret key, then the challenger $\mathcal{C}$ first generates ssk_ℓ. Then, the challenger $\mathcal{C}$ runs the roadside unit registration algorithm and returns a secret key rsk_l of RU_l with regard to SA_ℓ.
- *Road condition report:* In each query, the adversary $\mathcal{A}$ submits a pair $(V_j, \mathcal{T}_l, RU_l, I)$ to $\mathcal{C}$. Suppose both V_j and RU_l have been queried for secret keys before. If $\mathcal{T}_l$ is a valid authentication tuple jointly generated by V_j and RU_l, then the challenger $\mathcal{C}$ returns a pair (U, W).
- *RA processing:* In each query, the adversary $\mathcal{A}$ submits a pair (U, W) to $\mathcal{C}$. The challenger $\mathcal{C}$ returns the output of the RA processing algorithm.

End-Game: Eventually, the adversary $\mathcal{A}$ outputs a tuple (U^*, W^*) with regard to sub-authority SA^*, vehicle V^* and $(RU^*, \mathcal{T}^*, I^*)$. We say that the adversary $\mathcal{A}$ succeeds if all the following conditions hold:

- The adversary $\mathcal{A}$ has not been made a delegation query on sub-authority SA^* to get a secret key;
- The adversary $\mathcal{A}$ has not been made a registration query on vehicle V^*;
- $\mathsf{RApro}(\mathsf{par}, msk, U^*, W^*) = (RU^*, I^*)$, but (U^*, W^*) was not generated in a road condition report query with $(V^*, \mathcal{T}^*, RU^*, I^*)$.

Definition 1 *A RCoM scheme is UF-SI-CMA secure if any PPT adversary $\mathcal{A}$ who plays the above game with $\mathcal{C}$ has only negligible probability in winning the game, that is,*

$$\Pr[\mathcal{A}_{win}] \leq \epsilon(\kappa)$$

where the probability is taken over all coin tosses made by $\mathcal{C}$ and $\mathcal{A}$.

Remark 1 *Note that Definition 1 also implies that malicious vehicles cannot forge secret keys of SAs and vehicles. Otherwise, the adversary would be able to generate a valid ciphertext for some road condition information using the forged secret keys. Moreover, since the secret keys of RUs are generated in the same way as that of vehicles by SAs, the unforgeability of these keys can also be implied by Definition 1.*

We proceed to define the OW-CCA2 security of RCoM scheme against a curious cloud server.

Setup: On input a security parameter κ, the challenger $\mathcal{C}$ generates (par, msk) and gives public parameter par to the adversary $\mathcal{A}$.

Phase 1: The adversary $\mathcal{A}$ can adaptively submit queries to $\mathcal{C}$ as in Definition 1.

- *Delegation:* Same to Definition 1.
- *Vehicle registration:* Same to Definition 1.
- *Roadside unit registration:* Same to Definition 1.
- *Road condition report:* Same to Definition 1.
- *RA processing:* Same to Definition 1.

Challenge: At the end of Phase 1, the challenger randomly picks a vehicle V^*, a roadside unit RU^* and a road condition I^*, and computes $(U^*, W^*) \leftarrow \mathsf{RCrep}(\mathsf{par}, vsk^*, \mathcal{T}^*, RU^*, I^*)$, where V^* and RU^* are respectively administrated by SA^* and SA'^*, and $\mathcal{T}^* \leftarrow \mathsf{TKdis}(\mathsf{par}, (SA^*, V^*, vsk^*), (SA'^*, RU^*, rsk^*))$. The challenger $\mathcal{C}$ gives (U^*, W^*) to $\mathcal{A}$.

Phase 2: The adversary $\mathcal{A}$ can issue queries to $\mathcal{C}$ as in Phase 1, except that (U^*, W^*) cannot be submitted for RA processing.

Guess: At the end of Phase 2, the adversary outputs a guess $(\hat{RU}, \hat{I})$ and succeeds in the game if $(\hat{RU}, \hat{I}) = (RU^*, I^*)$.

Definition 2 *A RCoM scheme is OW-CCA2 secure for reported road conditions if any PPT adversary $\mathcal{A}$ has only negligible advantage in κ in winning the above game, that is,*

$$Adv^{ow-cca2} = \Pr[(\hat{RU}, \hat{I}) = (RU^*, I^*)] \leq \epsilon(\kappa)$$

where the probability is taken over all coin tosses made by $\mathcal{C}$ and $\mathcal{A}$.

4 Our Proposal

In this section, we propose a concrete RCoM scheme in bilinear groups. Table 1 summarizes the frequently used notations, which will be explained as they are used.

TABLE 1
Notation.

Symbol	Meaning
G, G_T	Cyclic groups with bilinear mapping $\hat{e}: G \times G \to G_T$
g, h	Two generators of G
p	A large prime, the order of G and G_T
x, z	The master secret key
y, w	The public key of RA
H_i	Cryptographic hash functions for $1 \leq i \leq 7$
τ	A threshold to trigger an emergency case
ssk_i	The secret key of SA_i
vsk_j	The secret key of vehicle V_j
rsk_l	The secret key of RU_l
r, v, s	Random values in Z_p^*
$\mathcal{T}_j, \mathcal{T}_l$	The tuples generated by V_j and RU_l in TKdis
θ_j, θ_l	The pairs contained in $\mathcal{T}_j, \mathcal{T}_l$, respectively
t_j, t_l	Time stamps
$\mathbb{T}_d$	Valid period of θ_l
I	Collected road condition information
(U, W)	Encrypted road condition information
$\mathcal{G}$	Equivalence class of road condition information

Suppose $G = \langle g \rangle$ and G_T are cyclic groups of prime order p. The mapping $\hat{e}: G \times G \to G_T$ is bilinear if the following properties are satisfied:

- Bilinearity: $\forall \mu, \nu \in G$ and $\forall a, b \in Z_p^*$, $\hat{e}(\mu^a, \nu^b) = \hat{e}(\mu, \nu)^{ab}$;
- Non-degeneracy: $\hat{e}(g, g) \neq 1$;
- Efficiency: The mapping $\hat{e}$ is efficiently computable.

Our RCoM scheme will rely on the following complexity assumption.

Computational Diffie-Hellman assumption (CDH). Let $G = \langle g \rangle$ be a cyclic group of prime order p. Given a tuple (g, g^a, g^b) for some random values $a, b \in_R Z_p^*$, any PPT algorithm $\mathcal{E}$ would have negligible probability in computing $g^{ab} \in G$.

4.1 System setup

The root authority RA generates a bilinear mapping $\hat{e}: G \times G \to G_T$, where G and G_T are cyclic groups with prime order p, and g, h are two distinct generators of G. RA then selects random values $x, z \in_R Z_p^*$, sets the master secret key $msk = (x, z)$, and computes $y = g^x$ and $w = g^z$. RA also picks seven cryptographic hash functions such as $H_i: \{0,1\}^* \to Z_p^*$ for $1 \leq i \leq 5$, $H_6: G \to \{0,1\}^{\lambda_{ru}+\lambda_I+2\log p}$ and $H_7: \{0,1\}^* \to G$, where λ_{ru} and λ_I denote the length of the identities of RU and road condition information I, respectively.

RA also generates an alert threshold τ (e.g., $\tau = 10$) such that when τ or more vehicles report the same road condition at the same place, it would be looked as an emergency case and need fast response from RA. Finally, RA sets the public system parameters $\mathsf{par} = (\hat{e}, G, G_T, g, h, p, y, w, H_1, H_2, \cdots, H_7, \tau)$.

4.2 Delegation to sub-authority

Sub-authorities are delegated by RA to authorize vehicles, in this way to improve the authorization efficiency. In the delegation phase, each SA_i would obtain a secret key from RA, that is, RA picks a random value $r_i \in_R Z_p^*$, calculates the secret key

$$ssk_i = (ssk_{i,1}, ssk_{i,2}) = \left(g^{r_i}, h^{r_i + xH_1(SA_i \| ssk_{i,1})}\right)$$

and sends ssk_i to SA_i securely. Sub-authority SA_i can verify ssk_i as follows

$$\hat{e}(ssk_{i,2}, g) \stackrel{?}{=} \hat{e}\left(h, ssk_{i,1} \cdot y^{H_1(SA_i \| ssk_{i,1})}\right) \quad (1)$$

4.3 Vehicle registration

In the registration phase, every vehicle V_j gets the authorization (e.g., a secret key) from its administrative sub-authority SA_i. SA_i picks a random value $r_{i,j} \in_R Z_p^*$, calculates the secret key $vsk_j = (vsk_{j,1}, vsk_{j,2}, vsk_{j,3})$ where

$$vsk_{j,1} = ssk_{i,1},\ vsk_{j,2} = g^{r_{i,j}}$$

and

$$vsk_{j,3} = ssk_{i,2} \cdot h^{r_{i,j} H_2(SA_i \| V_j \| vsk_{j,1} \| vsk_{j,2})}$$

and gives vsk_j to V_j securely. Vehicle V_j is able to verify vsk_j as follows

$$\hat{e}(vsk_{j,3}, g) \stackrel{?}{=} \hat{e}\left(h, vsk_{j,1} \cdot y^{H_1(SA_i \| vsk_{j,1})} \cdot vsk_{j,2}^{H_2(SA_i \| V_j \| vsk_{j,1} \| vsk_{j,2})}\right) \quad (2)$$

4.4 Roadside unit registration

As in the vehicle registration phase, every roadside unit RU_l obtains a secret key from its administrative sub-authority SA_ℓ. That is, SA_ℓ picks a random value $r_{\ell,l} \in_R Z_p^*$, calculates the secret key $rsk_l = (rsk_{l,1}, rsk_{l,2}, rsk_{l,3})$ where

$$rsk_{l,1} = ssk_{\ell,1},\ rsk_{l,2} = g^{r_{\ell,l}}$$

and

$$rsk_{l,3} = ssk_{\ell,2} \cdot h^{r_{\ell,l} H_2(SA_\ell \| RU_l \| rsk_{l,1} \| rsk_{l,2})}$$

and gives rsk_l to RU_l securely. Roadside unit RU_l can validate rsk_l as follows

$$\hat{e}(rsk_{l,3}, g) \stackrel{?}{=} \hat{e}\left(h, rsk_{l,1} \cdot y^{H_1(SA_\ell \| rsk_{l,1})} \cdot rsk_{l,2}^{H_2(SA_\ell \| RU_l \| rsk_{l,1} \| rsk_{l,2})}\right) \quad (3)$$

4.5 Token distribution

When some vehicle V_j enters into a new road section, it interacts with the administrative roadside unit RU_l. Specifically, V_j picks a random value $v_j \in_R Z_p^*$ and computes

$$\theta_{j,1} = g^{v_j},\ \theta_{j,2} = vsk_{j,3} \cdot h^{v_j H_3(V_j \| RU_l \| t_j \| \theta_{j,1})}$$

where t_j denotes the time stamp. Vehicle V_j sends the tuple $\mathcal{T}_j = (SA_i, V_j, vsk_{j,1}, vsk_{j,2}, t_j, \theta_{j,1}, \theta_{j,2})$ to RU_l, where SA_i denotes the administrative sub-authority of V_j. This step implies that some malicious vehicle cannot impersonate V_j to request a token from RU_l.

Roadside unit RU_l would not respond if $\mathcal{T}_j$ does not satisfy the following condition

$$\hat{e}(\theta_{j,2}, g) \stackrel{?}{=} \hat{e}\left(h, vsk_{j,1} \cdot vsk_{j,2}^{H_2(SA_i \| V_j \| vsk_{j,1} \| vsk_{j,2})} \cdot y^{H_1(SA_i \| vsk_{j,1})} \cdot \theta_{j,1}^{H_3(V_j \| RU_l \| t_j \| \theta_{j,1})}\right) \quad (4)$$

RU_l selects a random value $v_l \in_R Z_p^*$ and computes a token $\theta_l = (\theta_{l,1}, \theta_{l,2})$ as follows

$$\theta_{l,1} = g^{v_l},\ \theta_{l,2} = rsk_{l,3} \cdot h^{v_l H_4(V_j \| RU_l \| t_j \| t_l \| \mathbb{T}_d \| \theta_{j,1} \| \theta_{l,1})}$$

where t_l and $\mathbb{T}_d$ denote the time stamp and valid period of token θ_l, respectively. Roadside unit RU_l returns the authentication tuple $\mathcal{T}_l = (SA_\ell, rsk_{l,1}, rsk_{l,2}, t_l, \mathbb{T}_d, \theta_l)$, where SA_ℓ is the administrative sub-authority of RU_l.

Vehicle V_j accepts the authentication tuple $\mathcal{T}_l$ if it satisfies the following condition

$$\hat{e}(\theta_{l,2}, g) \stackrel{?}{=} \hat{e}\left(h, rsk_{l,1} \cdot rsk_{l,2}^{H_2(SA_\ell \| RU_l \| rsk_{l,1} \| rsk_{l,2})} \cdot y^{H_1(SA_\ell \| rsk_{l,1})} \cdot \theta_{l,1}^{H_4(V_j \| RU_l \| t_j \| t_l \| \mathbb{T}_d \| \theta_{j,1} \| \theta_{l,1})}\right) \quad (5)$$

The procedure of token distribution is shown in Fig. 2.

Vehicle V_j	Roadside unit RU_l
Choose $v_j \in_R Z_p^*$ and t_j	
Generate $(\theta_{j,1}, \theta_{j,2})$	
$\mathcal{T}_j = (SA_i, V_j, vsk_{j,1}, vsk_{j,2}, t_j, \theta_{j,1}, \theta_{j,2})$ $\longrightarrow$	
	Verify $\mathcal{T}_j$. If true, then do:
	Choose $v_l \in_R Z_p^*$, t_l and $\mathbb{T}_d$
	Compute $\theta_l = (\theta_{l,1}, \theta_{l,2})$
$\mathcal{T}_l = (SA_\ell, rsk_{l,1}, rsk_{l,2}, t_l, \mathbb{T}_d, \theta_l)$ $\longleftarrow$	
Verify $\mathcal{T}_l$. If true, then accept.	

Fig. 2. A procedure of token distribution between vehicle V_j and roadside unit RU_l

4.6 Road condition report

Suppose vehicle V_j collects road condition I on some section administrated by roadside unit RU_l at time Time. Then vehicle V_j performs as follows to generate a report. It selects a random value $s \in_R Z_p^*$ and computes the ciphertext $U = (u_1, u_2, u_3, u_4)$, where

$$u_1 = g^s$$
$$u_2 = H_6(w^s) \oplus (RU_l \| I \| rsk_{l,1} \| rsk_{l,2})$$
$$u_3 = H_7(RU_l \| I)^s$$
$$u_4 = vsk_{j,3} \cdot h^{sH_5(V_j \| \theta_l \| u_1 \| u_2 \| u_3 \| \mathsf{Time})}$$

Finally, vehicle V_j uploads the ciphertext U and tuple $W = (SA_i, SA_\ell, V_j, vsk_{j,1}, vsk_{j,2}, t_l, t_j, \mathbb{T}_d, \theta_l, \mathsf{Time})$ to the cloud server, where SA_i and SA_ℓ are the administrative sub-authorities of V_j and RU_l, respectively.

4.7 Cloud processing

Upon receiving a report from some vehicle, the cloud server performs the following processing steps.

Step 1: *Soundness verification*. In this step, the cloud server filters out forged information from some malicious vehicles. The cloud server checks whether the following equality holds

$$\hat{e}(u_4, g) \stackrel{?}{=} \hat{e}\left(h, vsk_{j,1} \cdot vsk_{j,2}^{H_2(SA_i \| V_j \| vsk_{j,1} \| vsk_{j,2})} \cdot y^{H_1(SA_i \| vsk_{j,1})} \cdot u_1^{H_5(V_j \| \theta_l \| u_1 \| u_2 \| u_3 \| \mathsf{Time})}\right) \quad (6)$$

The cloud server also checks whether the report was generated in period $\mathbb{T}_d$. If all conditions are satisfied, then the received elements in U and V are sound; otherwise, they are discarded by the cloud server.

Step 2: *Privacy-preserving monitoring*. For all sound tuples, the cloud server groups them into different equivalence classes such that the tuples in the same class report the same road condition for the same location (road section) within a reasonable time period. Originally there exists no any equivalence class. For a new sound tuple (U, W), the cloud server sequentially compares it with every existing equivalence class. Note that only one element in every equivalence class needs to be compared. Suppose (U', W') is an element in some equivalence class $\mathcal{G}'$. The cloud server checks whether the following condition is satisfied.

$$\hat{e}(u_1, u_3') \stackrel{?}{=} \hat{e}(u_1', u_3) \quad (7)$$

If true, then the tuple (U, W) is inserted into $\mathcal{G}'$; otherwise, the cloud server continues to compare it with another equivalence class. Eventually, if there exists no matching equivalence class, then a new one is constructed with only one element (U, W).

Step 3: For any equivalence class $\mathcal{G}$, if $|\mathcal{G}| \geq \tau$, then the cloud server sends an element (i.e., some tuple (U, W)) in $\mathcal{G}$ to RA, which implies that an emergency case is detected and requires RA to make response.

4.8 RA processing

The root authority runs the following steps to decrypt ciphertext U with the master secret key *msk*. RA computes

$$RU_l \| I \| rsk_{l,1} \| rsk_{l,2} \leftarrow u_2 \oplus H_6(u_1^z)$$

and checks whether Equality (5) and the following condition are satisfied

$$\hat{e}(u_1, H_7(RU_l \| I)) \stackrel{?}{=} \hat{e}(u_3, g) \quad (8)$$

If both are true, then RA accepts the reported road condition I at RU_l, and takes action if needed.

5 Soundness and Security

In this section, we show that our RCoM is sound and enjoys the UF-SI-CMA and OW-CCA2 security.

Theorem 1 *In a successful delegation, SA accepts the secret key generated by RA; In a successful registration, a vehicle or a RU accepts the secret key generated by SA; In a round of successful token distribution protocol, the vehicle accepts the authentication tuple generated by the corresponding RU; For any two ciphertexts*

encrypting the same road condition information, the cloud server classifies them into the same group; For a sound ciphertext, RA is able to recover the reported road condition information.

Proof *We only need to show Equations (1)-(8) hold.*

For a secret key ssk_i of SA_i, Equation (1) holds as follows

$$\hat{e}\left(h^{r_i+xH_1(SA_i\|ssk_{i,1})}, g\right) = \hat{e}\left(h, g^{r_i}(g^x)^{H_1(SA_i\|ssk_{i,1})}\right) = \hat{e}\left(h, ssk_{i,1}\cdot y^{H_1(SA_i\|ssk_{i,1})}\right)$$

For a secret key vsk_j of some vehicle V_j issued by SA_i, Equation (2) holds as follows

$$\begin{aligned}&\hat{e}(vsk_{j,3}, g)\\ &= \hat{e}(ssk_{i,2}, g)\cdot\hat{e}\left(h^{r_{i,j}H_2(SA_i\|V_j\|vsk_{j,1}\|vsk_{j,2})}, g\right)\\ &= \hat{e}\left(h, vsk_{j,1}\cdot y^{H_1(SA_i\|vsk_{j,1})}\right.\\ &\qquad\left.\cdot vsk_{j,2}^{H_2(SA_i\|V_j\|vsk_{j,1}\|vsk_{j,2})}\right)\end{aligned}$$

The correctness of Equation (3) can be proved in a similar way as Equation (2).

In the token distribution protocol, the correctness of tuple $\mathcal{T}_j$ generated by vehicle V_j with regard to the roadside unit RU_l is straightforward, since $\theta_{j,2} = vsk_{j,3}\cdot h^{v_j H_3(V_j\|RU_l\|t_j\|\theta_{j,1})}$. Similarly, the correctness of Equations (5) and (6) can also be verified.

For two ciphertext $U = (u_1, u_2, u_3, u_4)$ and $U' = (u_1', u_2', u_3', u_4')$, we have

$$\hat{e}(u_1, u_3') = \hat{e}(g^s, H_7(RU_l'\|I')^{s'}) = \hat{e}(g, H_7(RU_l'\|I'))^{s\cdot s'}$$

and

$$\hat{e}(u_1', u_3) = \hat{e}(g^{s'}, H_7(RU_l\|I)^s) = \hat{e}(g, H_7(RU_l\|I))^{s\cdot s'}$$

Thus, Equation (7) holds if and only if $RU_l\|I = RU_l'\|I'$.

For a valid ciphertext U, Equation (8) in the RA processing phase holds as follows

$$\begin{aligned}\hat{e}(u_1, H_7(RU_l\|I)) &= \hat{e}(g^s, H_7(RU_l\|I))\\ &= \hat{e}(g, H_7(RU_l\|I)^s) = \hat{e}(u_3, g)\end{aligned}$$

Theorem 2 *Suppose the CDH assumption holds in bilinear group G. The proposed RCoM scheme is EU-SI-CMA secure for road condition reports against adaptive impersonation attacks. That is, any vehicle cannot forge road condition reports of other vehicles.*

Proof *The proof follows the standard framework established in [27, Theorem 1]. He et al.'s scheme [27] is proven existentially unforgeable against selective identity and chosen message attacks assuming that the CDH assumption holds in bilinear group G. Our proof for Theorem 2 in the random oracle model follows mostly in [27, Theorem 1] except that in the setup phase, the challenger $\mathcal{C}$ chooses $z \in_R Z_p^*$ and computes $w = g^z$, and in the query phase, the challenger $\mathcal{C}$ needs to answer two more types of queries, i.e., road condition report and RA processing queries. In fact, the road condition report queries can be answered in a similar way as in vehicle/roadside unit registration queries since only the public parameter y is involved in generating u_1, u_2, u_3; whereas the RA processing queries can be answered directly using z. Thus, our RCoM scheme is EU-SI-CMA secure if the CDH assumption holds in G.*

Theorem 3 *Suppose the CDH assumption holds in bilinear group G. The proposed RCoM scheme is EU-SI-CMA secure for the secret keys of sub-authorities, vehicles and roadside units against adaptive impersonation attacks. That is, any vehicle cannot forge a valid secret key of another vehicle, sub-authority or roadside unit.*

As noted in Remark 1, the proof directly follows from Theorem 2.

Theorem 4 *Suppose the CDH assumption holds in bilinear group G. The proposed RCoM scheme offers OW-CCA2 confidentiality for road condition reports against the cloud server.*

Proof *The proof follows the standard framework established in [29, Theorem 3]. Yang et al.'s scheme [29] is proven OW-CCA2 secure assuming that the CDH assumption holds in bilinear group G. Our proof for Theorem 4 in the random oracle model follows mostly in [29, Theorem 3] except that in the setup phase, the challenger $\mathcal{C}$ chooses $x \in_R Z_p^*$ and computes $y = g^x$, and in the query phase, the challenger $\mathcal{C}$ needs to answer four more types of queries, i.e., delegation, vehicle registration, roadside unit registration and road condition report queries. In fact, these queries can be answered directly using x and y without leaking any information about $RU^*\|I^*$. Thus, our RCoM scheme is OW-CCA2 secure if the CDH assumption holds in G.*

6 Comparison and Analysis

6.1 Functionality comparison

We now compare our RCoM construction with existing schemes in Table 2. In [24], [25], the authors studied the delegated data outsourcing scenario, such that the data owner is able to authorize a proxy to process her data and upload to the cloud server. Particularly, the data owner needs to generate a warrant and sign it with some signature scheme E, and gives the authorization pair (warrant, signature) to the proxy for verification. Note that if the signature scheme E is secure, then anyone including the designated proxy cannot forge a (warrant, signature) pair. This authorization mechanism also implies that the delegation/authorization can be publicly verified in the comprehensive auditing phase on outsourced data. Compared with our RCoM scheme, the proposals in [24], [25] did not consider data privacy protection, thus they cannot support equality test on ciphertexts without decryption.

TABLE 2
Comparison with related techniques.

Functionality	Ours	[24]	[25]	[26]	[27]	[28]
Delegation	√	√	√	√	√	√
Source authentication	√	√	√	√	√	√
Integrity guarantee	√	√	√	√	√	×
Privacy protection	√	×	×	×	×	√
Third party equality test	√	×	×	×	×	×

In [26], Yang et al. designed a framework to authenticate messages in ADS-B system based on the three-level hierarchical identity-based signature scheme. They also noticed that the verification costs need to be reduced, especially when the recipient receives lots of (message, signature) pairs. This issue was well addressed in their proposed two

concrete schemes with partial and full batch verification, respectively. He et al. [27] studied the same problem and proposed a more efficient construction without using hash-to-point operations. Compared with our scheme, there are no data privacy protection in the ADS-B authentication framework and constructions in [26], [27].

Baek et al. [28] presented a confidentiality framework for ADS-B. They noted that key management and efficiency are two key issues in this framework. To address these issues, a staged identity-based encryption scheme (SIBE) is designed from identity-based encryption (IBE) and symmetric encryption. IBE does not need complicated key management mechanism as in traditional PKI-based crypto systems, however, many IBE schemes may require resource-intensive computations such as bilinear mapping. To reduce the computation costs, only the symmetric key is transferred as IBE ciphertext in the first stage of SIBE, all the subsequent communication are secured by the symmetric encryption scheme. Compared with RCoM scheme, their confidentiality framework and SIBE scheme do not consider data integrity protection and the ciphertexts do not allow equality test.

6.2 Theoretical analysis

We analyze the computational complexity of our RCoM scheme in terms of computation costs of every algorithm/protocol at each entity side, which is summarized in Table 3. Our analysis focuses on the most time-consuming operations in the scheme such as exponentiations in group G and bilinear pairing $\hat{e}$. In the table, `Exp` and `Pair` denote the evaluation times of an exponentiation in G and a bilinear pairing, respectively.

TABLE 3
Computational complexity of each algorithm/protocol in RCoM scheme.

Algorithm/protocol	Entity	Computations
Setup	RA	2Exp
SAdlg	RA	2Exp
	SA	1Exp + 2Pair
VHreg	SA	2Exp
	V	2Exp + 2Pair
RUreg	SA	2Exp
	RU	2Exp + 2Pair
TKdis	V	5Exp + 2Pair
	RU	5Exp + 2Pair
RCrep	V	4Exp
CLpro	CS	3Exp + $(2n+2)$Pair
RApro	RA	4Exp + 4Pair

The computation costs in the table are analyzed for one evaluation, that is, one delegation in the SAdlg algorithm, one registration in both VHreg and RUreg algorithms, one round of token distribution in the TKdis protocol, and one road condition report is generated and processed in the RCrep, CLpro and RApro algorithms, respectively. In the Setup phase, RA only needs to perform two exponentiations in G to produce the public parameters y and w. The delegation algorithm SAdlg requires RA to take two exponentiations in G to generate a secret key for some SA, while SA needs to perform one exponentiation and two bilinear pairings to validate its secret key. For the VHreg and RUreg algorithms, SA follows the same method to generate a secret key for a vehicle or a roadside unit. In detail, SA takes two exponentiations in G to generate such a secret key, whereas the vehicle (or the roadside unit) takes two more bilinear pairings to do verification.

The protocol TKdis contains four steps between a vehicle V_j and a roadside unit RU_l. The tuple $\mathcal{T}_j$ is generated by V_j with two exponentiations in G, while it is validated by RU_l with three exponentiation and two bilinear pairings. If $\mathcal{T}_j$ passes the validation, then an authentication tuple $\mathcal{T}_l$ would be produced and verified by RU_l and V_j, respectively, with the same computation costs for $\mathcal{T}_j$. In generating a road condition report (U, W), the vehicle V_j only needs to take four exponentiations in G to compose the elements in U. The generation of W does not involve any complicated computation. In the CLpro algorithm, every tuple (U, W) is verified and compared with the stored equivalence classes. As shown in Equation (6), the verification step takes three exponentiations and two bilinear pairings. Suppose there are n equivalence classes at the cloud server side. For a new report (U, W), it is compared with only one element in each equivalence class. Thus, the second step of equality test in CLpro takes at most $2n$ bilinear pairings, that is, this step enjoys the linear computation complexity. In the RApro algorithm, RA recovers $RU_l \| I$ from U, which requires four exponentiation and four bilinear pairings.

6.3 Experimental analysis

We conducted the experiments of our RCoM scheme using the Pairing Based Cryptography Library (PBC, http://crypto.stanford.edu/pbc/). The details of hardware and software environments are summarized in Table 4. The elliptic curve is of Type A ($y^2 = x^3 + x$) such that p is a 160-bit prime and the element size in G is 256 bits.

TABLE 4
Experiment environments.

Environment		Details
Hardware	CPU	Intel(R) Pentium(R) G645 @ 2.90 GHz
	Memory	2 GB
Software	Operating system	Microsoft Windows 7
	Programming language	C
	Library	Pairing Based Cryptography

The performance of the Setup, SAdlg, VHreg and RUreg algorithms are shown in Figure 3. The evaluation shows that the Setup algorithm can be completed in less than 20 msec, which is mainly determined by two exponentiations in G. For a delegation, RA can generate a secret key ssk_i for some sub-authority SA_i in roughly 7 msec, whereas SA_i is able to validate ssk_i with less than 8 msec. These two procedures are presented by DelGen and DelVrf in the figure. The vehicle registration enjoys the comparable performance of the roadside unit registration, that is, the secret keys vsk_j and rsk_l can be generated by sub-authorities in roughly the same time, and the verification at respective sides of vehicle and roadside unit also takes the similar time. This is consistent with the theoretical analysis in Section 6.2 that both VHreg and RUreg algorithms follow the same approach to fulfill the registration. Note that in the figure, VKeyGen and VKeyVrf respectively denote the generation and verification procedures of vsk_j for vehicle V_j, and RUKeyGen

and RUKeyVrf represent the generation and verification procedures of rsk_l for vehicle RU_l, respectively.

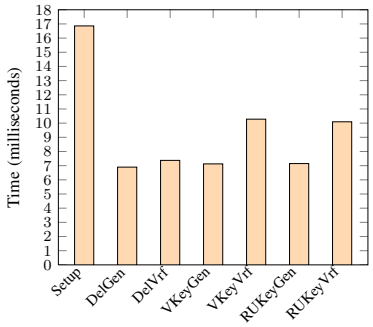

Fig. 3. Performance evaluation of the Setup, SAdIg, VHreg and RUreg algorithms.

Figure 4 plots the performance of the token distribution TKdis protocol, and the RCrep and RApro algorithms. As shown in Section 4.5, the vehicle V_j (resp. RU_l) needs to generate a tuple $\mathcal{T}_j$ and validate $\mathcal{T}_l$ (resp. to validate $\mathcal{T}_j$ and generate $\mathcal{T}_l$). Thus, in the TKdis protocol, both sides of vehicle V_j and roadside unit RU_l require roughly the same computation time, which are shown in Figure 4 with TK_V and TK_{RU}, respectively. When vehicle V_j collects a road condition I, it is able to generate a report in about 15 msec (see RepGen in the figure). The report can be verified through Equation (6) in roughly 13 msec (see RepVrf), which is run in the first step of the CLpro algorithm. For a sound report, RA is able to recover the road condition as shown in Section 4.8 in 22 msec, which is depicted as RepDec in Figure 4. In fact, the computation time of the RApro is determined by the verification time of Equalities (5) and (8).

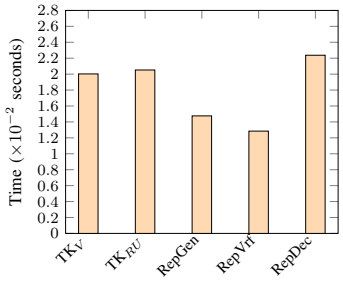

Fig. 4. Performance evaluation of the TKdis protocol, and the RCrep and RApro algorithms.

For the comparison performance between a new sound road condition report and each existing equivalence class, several cases with different number of equivalence classes are considered, that is, the step 2 of Section 4.7 is run to compare $10, 20, \cdots, 100$ pairs of $\{(u_1, u_3), (u'_1, u'_3)\}$, where (u'_1, u'_3) denotes the pair in some equivalence class. The experiment results are shown in Figure 5, which demonstrate that the performance of the step 2 of Section 4.7 is linearly determined by the number of equivalence classes at the cloud server side. It is easy to see that the average execution time of comparing with a single equivalence class is roughly 4 msec.

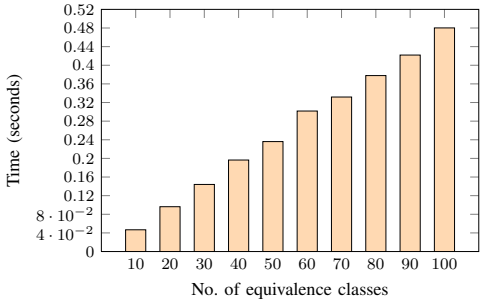

Fig. 5. Performance evaluation of the CLpro algorithm.

7 Conclusion and Remark

In this article, we considered the problem of privacy-preserving cloud-based road condition monitoring with source authentication (RCoM). There are two levels of authorities such that the root authority delegates sub-authorities to perform registration for vehicles and RUs. RA monitors real-time road conditions through a third party intermediary, that is, vehicles report the detected road conditions to the cloud server for verification and processing, in this way, only the valid information sent from legitimate vehicles will be picked out for RA to make response. To protect the privacy against the cloud server, the road condition report should be uploaded in ciphertext format, which brings another challenge for the cloud server to distinguish the same road condition for the same place from lots of reports. In response to these functionalities and security and privacy requirements in RCoM, we presented an efficient scheme and compared it with related techniques. Through extensive theoretical and experimental analyses, we demonstrate that the proposed RCoM scheme is practical in application settings.

Acknowledgments

This article is supported in part by the National Key R&D Program of China through project 2017YFB0802500, the National Natural Science Foundation of China under projects 61772150, 61772538, 61672083, 91646203, 61472429, 61402029, 61862012, 61862011, and 61602125, the National Cryptography Development Fund of China under projects MMJJ20170217 and MMJJ20170106, the Foundation of Science and Technology on Information Assurance Laboratory through project 61421120305162112006, the Guangxi Natural Science Foundation under Grant 2018JJA170035, the Guangxi Young Teachers' Basic Ability Improvement Program under Grant 2018KY0194, and the open program of Guangxi Key Laboratory of Cryptography and Information Security under projects GCIS201622 and GCIS201702. Y. Wei was supported in part by the Natural Science Foundation of China under Grant 61572148, in part by the Guangxi Natural Science Foundation under Grant 2015GXNSFGA139007.

References

[1] L. Zhang, C. Hu, Q. Wu, J. Domingo-Ferrer, and B. Qin, "Privacy-preserving vehicular communication authentication with hierarchical aggregation and fast response," *IEEE Transactions on Computers*, vol. 65, no. 8, pp. 2562–2574, Aug. 2016.

[2] Q. Wu, J. Domingo-Ferrer, and U. Gonzalez-Nicolas, "Balanced trustworthiness, safety, and privacy in vehicle-to-vehicle communications," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 2, pp. 559–573, Feb 2010.

[3] F. Qu, Z. Wu, F. Y. Wang, and W. Cho, "A security and privacy review of vanets," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 6, pp. 2985–2996, Dec 2015.

[4] "IEEE Standard for Wireless Access in Vehicular Environments–Security Services for Applications and Management Messages," *IEEE Std 1609.2-2016 (Revision of IEEE Std 1609.2-2013)*, pp. 1–240, March 2016.

[5] L. Zhang, Q. Wu, J. Domingo-Ferrer, B. Qin, and C. Hu, "Distributed aggregate privacy-preserving authentication in vanets," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 3, pp. 516–526, March 2017.

[6] L. Chen, S. L. Ng, and G. Wang, "Threshold anonymous announcement in vanets," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 3, pp. 605–615, March 2011.

[7] Y. Liu, J. Ling, Q. Wu, and B. Qin, "Scalable privacy-enhanced traffic monitoring in vehicular ad hoc networks," *Soft Computing*, vol. 20, no. 8, pp. 3335–3346, Aug 2016.

[8] R. Yu, Y. Zhang, S. Gjessing, W. Xia, and K. Yang, "Toward cloud-based vehicular networks with efficient resource management," *IEEE Network*, vol. 27, no. 5, pp. 48–55, September 2013.

[9] J. A. Guerrero-ibanez, S. Zeadally, and J. Contreras-Castillo, "Integration challenges of intelligent transportation systems with connected vehicle, cloud computing, and internet of things technologies," *IEEE Wireless Communications*, vol. 22, no. 6, pp. 122–128, December 2015.

[10] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia, "A view of cloud computing," *Commun. ACM*, vol. 53, no. 4, pp. 50–58, Apr. 2010.

[11] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the Forty-first Annual ACM Symposium on Theory of Computing*, ser. STOC'09. New York, NY, USA: ACM, 2009, pp. 169–178.

[12] Q. Wu, B. Qin, L. Zhang, J. Domingo-Ferrer, O. Farràs, and J. A. Manjón, "Contributory broadcast encryption with efficient encryption and short ciphertexts," *IEEE Transactions on Computers*, vol. 65, no. 2, pp. 466–479, Feb 2016.

[13] L. Guo, M. Dong, K. Ota, Q. Li, T. Ye, J. Wu, and J. Li, "A secure mechanism for big data collection in large scale internet of vehicle," *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 601–610, April 2017.

[14] V. Sucasas, G. Mantas, F. B. Saghezchi, A. Radwan, and J. Rodriguez, "An autonomous privacy-preserving authentication scheme for intelligent transportation systems," *Computers & Security*, vol. 60, pp. 193–205, 2016.

[15] A. Malhi and S. Batra, "Privacy-preserving authentication framework using bloom filter for secure vehicular communications," *International Journal of Information Security*, vol. 15, no. 4, pp. 433–453, Aug 2016.

[16] Y. Liu, W. Guo, C.-I. Fan, L. Chang, and C. Cheng, "A practical privacy-preserving data aggregation (3pda) scheme for smart grid," *IEEE Transactions on Industrial Informatics*, pp. 1–1, 2018.

[17] D. Song, E. Shi, I. Fischer, and U. Shankar, "Cloud data protection for the masses," *IEEE Computer*, vol. 45, no. 1, pp. 39–45, Jan 2012.

[18] B. Wang, H. Li, X. Liu, F. Li, and X. Li, "Efficient public verification on the integrity of multi-owner data in the cloud," *Journal of Communications and Networks*, vol. 16, no. 6, pp. 592–599, Dec 2014.

[19] G. Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner, Z. Peterson, and D. Song, "Provable data possession at untrusted stores," in *Proceedings of the 14th ACM Conference on Computer and Communications Security*, ser. CCS'07. New York, NY, USA: ACM, 2007, pp. 598–609.

[20] A. Juels and B. S. Kaliski, Jr., "Pors: Proofs of retrievability for large files," in *Proceedings of the 14th ACM Conference on Computer and Communications Security*, ser. CCS'07. New York, NY, USA: ACM, 2007, pp. 584–597.

[21] G. Ateniese, S. Kamara, and J. Katz, "Proofs of storage from homomorphic identification protocols," in *Advances in Cryptology–ASIACRYPT 2009*, M. Matsui, Ed. Springer Berlin Heidelberg, 2009, pp. 319–333.

[22] Y. Wang, Q. Wu, B. Qin, X. Chen, X. Huang, and J. Lou, "Ownership-hidden group-oriented proofs of storage from pre-homomorphic signatures," *Peer-to-Peer Networking and Applications*, vol. 11, no. 2, pp. 235–251, Mar 2018.

[23] H. Shacham and B. Waters, "Compact proofs of retrievability," *Journal of Cryptology*, vol. 26, no. 3, pp. 442–483, Jul 2013.

[24] H. Wang, D. He, and S. Tang, "Identity-based proxy-oriented data uploading and remote data integrity checking in public cloud," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 6, pp. 1165–1176, June 2016.

[25] Y. Wang, Q. Wu, B. Qin, W. Shi, R. H. Deng, and J. Hu, "Identity-based data outsourcing with comprehensive auditing in clouds," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 4, pp. 940–952, April 2017.

[26] A. Yang, X. Tan, J. Baek, and D. S. Wong, "A new ads-b authentication framework based on efficient hierarchical identity-based signature with batch verification," *IEEE Transactions on Services Computing*, vol. 10, no. 2, pp. 165–175, March 2017.

[27] D. He, N. Kumar, K. K. R. Choo, and W. Wu, "Efficient hierarchical identity-based signature with batch verification for automatic dependent surveillance-broadcast system," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 2, pp. 454–464, Feb 2017.

[28] J. Baek, E. Hableel, Y. J. Byon, D. S. Wong, K. Jang, and H. Yeo, "How to protect ads-b: Confidentiality framework and efficient realization based on staged identity-based encryption," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 3, pp. 690–700, March 2017.

[29] G. Yang, C. H. Tan, Q. Huang, and D. S. Wong, "Probabilistic public key encryption with equality test," in *Topics in Cryptology - CT-RSA 2010: The Cryptographers' Track at the RSA Conference 2010, San Francisco, CA, USA, March 1-5, 2010. Proceedings*, J. Pieprzyk, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 119–131.

[30] Q. Tang, "Public key encryption supporting plaintext equality test and user-specified authorization," *Security and Communication Networks*, vol. 5, no. 12, pp. 1351–1362, 2012.

[31] S. Ma, Q. Huang, M. Zhang, and B. Yang, "Efficient public key encryption with equality test supporting flexible authorization," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 3, pp. 458–470, March 2015.

[32] S. Ma, M. Zhang, Q. Huang, and B. Yang, "Public key encryption with delegated equality test in a multi-user setting," *The Computer Journal*, vol. 58, no. 4, pp. 986–1002, 2015.

[33] Y. Wang and H. Pang, "Probabilistic public key encryption for controlled equijoin in relational databases," *The Computer Journal*, vol. 60, no. 4, pp. 600–612, 2017.

[34] H. Pang and X. Ding, "Privacy-preserving ad-hoc equi-join on outsourced data," *ACM Trans. Database Syst.*, vol. 39, no. 3, pp. 23:1–23:40, Oct. 2014.

[35] Y. Wang, H. Pang, N. H. Tran, and R. H. Deng, "Cca secure encryption supporting authorized equality test on ciphertexts in standard model and its applications," *Information Sciences*, vol. 414, pp. 289–305, 2017.

[36] Z. Yan, W. Ding, X. Yu, H. Zhu, and R. H. Deng, "Deduplication on encrypted big data in cloud," *IEEE Transactions on Big Data*, vol. 2, no. 2, pp. 138–150, June 2016.

[37] H. Cui, R. H. Deng, Y. Li, and G. Wu, "Attribute-based storage supporting secure deduplication of encrypted data in cloud," *IEEE Transactions on Big Data*, vol. PP, no. 99, pp. 1–1, 2017.

[38] Z. Yan, L. Zhang, W. Ding, and Q. Zheng, "Heterogeneous data storage management with deduplication in cloud computing," *IEEE Transactions on Big Data*, vol. PP, no. 99, pp. 1–1, 2017.

Yujue Wang received the Ph.D. degrees from the Wuhan University, Wuhan, China, and City University of Hong Kong, Hong Kong, under the joint Ph.D. program, in 2015. He was a Research Fellow with the School of Information Systems, Singapore Management University. He is currently with the School of Computer Science and Information Security, Guilin University of Electronic Technology, China. His research interests include applied cryptography, database security and cloud computing security.

Yong Ding received his PhD in Cryptography from the School of Communication Engineering, Xidian University, China, in 2005. He is currently a Professor at School of Computer Science and Information Security, Guilin University of Electronic Technology, China. He was a research fellow of Computer Science at City University of Hong Kong from April, 2008 to September, 2009. His research interests include cryptography and information security.

Qianhong Wu received his Ph.D. in Cryptography from Xidian University in 2004. Since then, he has been with Wollongong University (Australia) as an associate research fellow, with Wuhan University (China) as an associate professor, and with Universitat Rovira i Virgili (Spain) as a research director. He is currently a professor in Beihang University in China. His research interests include cryptography, information security and privacy, VANET security and cloud computing security. He has been a holder/co-holder of 8 China/Australia/Spain funded projects. He has authored more than 20 patents and over 120 publications in leading journals and conferences. He has served in the program committee of several international conferences in information security and privacy. He is a member of IACR and IEEE.

Yongzhuang Wei received the M.S. and the Ph.D. degrees in cryptology from Xidian University, Xian, China, in 2004 and 2009, respectively. Since July 2011, he has been doing research with the State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences, Beijing, China. Since September 2014, he joined the Guangxi Key Laboratory of Cryptography and Information Security at Guilin University of Electronic Technology, where he is currently employed as a full professor. He is now a member of Chinese Association for Cryptologic Research (CACR). His current research interests include Boolean functions, stream ciphers, block ciphers, and hash functions.

Bo Qin received her Ph.D. degree in Cryptography from Xidian University in 2008 in China. Since then, she has been with Xi'an University of Technology (China) as a lecturer and with Universitat Rovira i Virgili (Catalonia) as a postdoctoral researcher. She is currently a lecturer in the Renmin University in China. Her research interests include pairing-based cryptography, data security and privacy, and VANET security. She has been a holder/co-holder of 5 China/Spain funded projects. She has authored over 80 publications in well-recognized journals and conferences and served in the program committee of a number of international conferences in information security.

Huiyong Wang received his Ph.D. degree in software theory and applications from Chinese Academy of Sciences in 2017 in China. He is currently a Lecture at the School of Mathematics and Computing Science, Guilin University of Electronic Technology, China. His research interests include privacy-preserving computation, information security, cyber security, multi-party computation and homomorphic encryption.