A Tamper-Resistant Broadcasting Scheme for Secure Communication in Internet of Autonomous Vehicles

Jianfei Sun, Junyi Tao, Hao Zhang, Yanan Zhao, Liming Nie, Xiaochun Cheng, and Tianwei Zhang, *Member, IEEE*

***Abstract*— As increasingly prevalent technologies in autonomous driving, 5G and the Internet of Things (IoT), Internet of autonomous vehicle (IoAV) technology is recognized as a technique that is capable of disruptively changing the way people travel and greatly improving the travel experience. In the IoAV scenarios, information dissemination is inseparable from the interaction between autonomous vehicles and smart infrastructure. However, existing efforts rarely focus on the secrecy, authenticity of interactive data and flexible one-to-many communication between autonomous vehicles. In this paper, we propose a tamper-resistant broadcasting (TRBS) scheme for secure communication, which handles the inefficiencies and insecurity of existing identity-based broadcast signcryption solutions. Not only can our TRBS protect communication data from being illegally accessed, forged, or tampered with by malicious vehicles, but it can also enable efficient and flexible secure information dissemination between autonomous vehicles. We also exhibit strict security proofs and experimental evaluations to demonstrate our TRBS is secure and efficient for real-world applications.**

***Index Terms*— Internet of the autonomous vehicle, secure communication, tamper-resistant, identity-based broadcast signcryption.**

I. Introduction

With the fast advancement of the Internet of Things (IoT) and 5G-based wireless communication technologies, the evolution from the traditional vehicular ad hoc network (VANET) to the Internet of Autonomous Vehicles (IoAV) has been taking place [1], [2], [3], [4]. Unlike traditional VANET [5], [6], [7] only focuses on conventional vehicles communicating with infrastructure and other vehicles, IoAV as a combination technology of IoT and autonomous driving vehicle pays more attention to facilitating communication among autonomous driving vehicles and smart vehicular infrastructure. In a typical IoAV application, the data gathered from autonomous vehicles include various parameters such as vehicle speed, location, and movement direction while the data collected from smart vehicular infrastructure commonly includes overall traffic information. These data can be exchanged among internet-connected devices via IoT platforms to enhance better travel experiences, promote road safety and facilitate traffic management.

Despite the fact that the IoAV is generally recognized as a revolutionary technology that can significantly change the way and improve better experiences users travel, it also confronts some data security and privacy threats due to the massive data communication that exists between autonomous vehicles and smart infrastructure [8], [9], [10]. This could be illustrated from two perspectives: (I) how to ensure secure data communication between autonomous vehicles in case of data authenticity assurance is of prime consideration. To be more specific, for the confidentiality of data, in general, the interactive data between vehicles, including speed, location, movement direction, etc., are frequently highly sensitive and should be accessible only by legal vehicles which are only granted. If the communication data are transmitted in cleartext, this inevitably results in confidentiality and privacy breaches. For the authenticity of data, autonomous vehicles may receive instructions (*e.g.,* turning, decelerating, turning around) from other interacted autonomous vehicles to complete some tasks, if the instructions are forged and tampered with by other malicious vehicles, the implementation tasks probably are incomplete for cooperative vehicles, and more seriously, a traffic accident may occur; (II) how to efficiently and flexibly implement secure information dissemination between autonomous vehicles is another concern. In existing modes of information dissemination, most vehicle communication modes are primarily based on one-to-one secure communication, *i.e.,* a vehicle sender can only communicate with one vehicle recipient at any one time. If a vehicle intends to share the same information with a group

Manuscript received 13 December 2022; revised 22 February 2023; accepted 28 March 2023. This work was supported in part by the Nanyang Technological University (NTU)-DESAY SV Research Program under Grant 2018-0980, in part by NTU Start-Up Grant, and in part by the National Natural Science Foundation of China under Grant 61972359. The Associate Editor for this article was S. Mumtaz. *(Corresponding author: Yanan Zhao.)*

Jianfei Sun and Tianwei Zhang are with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798 (e-mail: jianfei.sun@ntu.edu.sg; tianwei.zhang@ntu.edu.sg).

Junyi Tao is with Amazon Web Services, Seattle, WA 98101 USA (e-mail: jaydent121208@gmail.com).

Hao Zhang is with the Science and Technology on Communication Security Laboratory, Chengdu 611731, China (e-mail: z.zhhao93@gmail.com).

Yanan Zhao is with the School of Transportation Science and Engineering, Beihang University, Beijing 100191, China (e-mail: zynbyxz@gmail.com).

Liming Nie is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798, and also with the School of Computer Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China (e-mail: liming.nie@ntu.edu.sg).

Xiaochun Cheng is with the Department of Computer Science, Swansea University, NW4 4BT Swansea, Wales, U.K. (e-mail: xiaochun.cheng@gmail.com).

Digital Object Identifier 10.1109/TITS.2023.3265403

of autonomous vehicles, the general solution is to allow the shared vehicle to respectively interact with each of the targets and send the respective encrypted data to them. Clearly, for the shared vehicle, this certainly produces some redundant copies of the same data since it needs to perform multiple encryption operations on the same data. Ideally, the vehicle only conducts encryption once to efficiently realize one-to-many secure data communication with the group of vehicles.

As we all know, the standard digital signature technique can address the data authenticity issue, however, it fails to ensure data confidentiality due to the lack of an encryption mechanism. As a novel cryptographic technology, the signcryption methodology enables simultaneous data authenticity and confidentiality by embedding a sender's private keys into the encrypted ciphertext of communication data, thus well-addressing the security concern (I). For another security problem (II), there are generally two kinds of solutions in technology to realize one-to-many secure information dissemination between autonomous vehicles: (a) attribute-based encryption (ABE) [11], [12], [13] and (b) identity-based broadcast encryption (IBBE) [14], [15], [16]. ABE techniques enable one-to-many data communications between vehicles via designating an access policy in the ciphertext to indicate the authorized vehicles whose attributes match the access policy. This method is conventionally inappropriate for solving concerns (II) due to the prohibitive computation costs and the inaccurate authorization (*i.e.*, the attribute-based access policy may indicate the authorized vehicles beyond those designated); IBBE approaches can elegantly handle the concern (II) via an access list indicating each unique identity of authorized vehicles, whereas they cannot solve the concern (I). In summary, the signcryption and IBBE approaches can only partially settle the concerns (I) & (II).

To simultaneously address the concerns (I) & (II), identity-based broadcast signcryption (IBBSC) [17], [18], [19], [20], [21] as a technical combination of IBBE and signcryption can realize one-to-many secure data communication between vehicles as well as prevent communication data from being edited, forged and even tampered with by other malicious attackers during the whole data communication. However, most existing IBBSC solutions either suffer from serious security issues or have inefficiency problems as their constructions are mostly based on symmetric prime-order groups or involve a large number of pairing computations. To mitigate the above issues, Zhao et al. [20] proposed a novel IBBSC scheme constructed on asymmetric prime-order groups, which is claimed to feature high efficiency and semantic security. However, this scheme suffers from collusion attacks, thus failing to offer the claimed security property, which is a crucial goal to ensure the confidentiality and integrity of communication data. That is to say, to date, there is no such IBBSC solution that securely enables efficient and flexible one-to-many information communication between vehicles under the premise of ensuring data authenticity.

A. Our Motivation and Contributions

An imperative motivation stimulates us to propose a secure IBBSC scheme to solve the concerns (I) & (II) mentioned above while remedying the security defects of Zhao et al.'s IBBSC. In this paper, we propose a tamper-resistant broadcasting (TRBS) scheme for secure communication in IoAV with our IBBSC technology, which not only protects communication data from not being accessed and forged or tampered with by illegal vehicles but also realizes efficient and flexible secure information dissemination between autonomous vehicles. As far as we are aware, our TRBS is the first scheme that focuses on challenges related to secure efficient one-to-many communication while maintaining the secrecy and integrity of communication data in IoAV. The main contributions of this paper are summarized as follows:

- *Confidentiality and authenticity*: Our TRBS scheme not only enables legal access to communication data by only eligible vehicles but also is capable of preventing communication data from being forged and tampered with during the whole communication, thus preserving the originality and secrecy of communication data. Compared to other solutions simply considering confidentiality, our TRBS is more comprehensive in security and privacy.
- *One-to-many secure communication*: Our TRBS scheme empowers a vehicle sender to securely broadcast the same encrypted communication data to multiple vehicle recipients, such that only authorized vehicle recipients can access the communication data. Compared to traditional one-to-one secure communication modes, our TRBS is more flexible, efficient, and practical.
- *Strong Security*: Our TRBS scheme is immune to forgery attacks, collusion attacks, impersonation attacks, and so on. Compared to other insecure works (See Section II), our TRBS is more appropriate for IoAV-based secure communication applications.

In addition, the formal security analysis indicates our TRBS is semantically-secure, and the theoretical analysis and experimental results show the practicability of our TRBS.

II. Related Work

Zheng [22] first proposed the primitive of signcryption. Different from the traditional form of signature and then encryption, the scheme of [22] allows the signature and encryption operations to be executed simultaneously, which greatly eliminates expensive computation and communication costs. Subsequently, Malone-Lee [23] suggested an identity-based signcryption (IBSC) to elegantly address the certificate management hassles associated with [22]. Inspired by [23], multiple IBSC works had been introduced in various areas [24], [25], [26], [27]. Yang et al. [26] applied the IBSC to an IoT-enabled maritime transportation system. In their scheme, blockchain technology is employed to further improve the performance and security of their solution. To realize secure data collection and delivery in industrial crowdsourcing environments, Karati et al. [28] constructed an IBSC methodology that is proven secure in the standard model. However, when there is a need for the sender to send a message to multiple recipients, IBSC is inappropriate because it will result in the fact that the sender has to send the same message multiple times. This weakness hampers the development of IBSC significantly.

Fortunately, as a combination of broadcast encryption [29] and identity-based signcryption, the concept of identity-based broadcast signcryption (IBBSC) not only features a one-step signature and encryption but also enables a message to be delivered to multiple vehicles simultaneously. Due to such merits, many IBBSC solutions had also been proposed successively [17], [18], [19], [20]. On the basis of the work of Chen and Malone-Lee [30], Li et al. [17] introduced the idea of IBBSC, which elegantly achieves one-to-many data sharing with confidentiality and integrity at the same time. The fly in the ointment is that neither outside attacks nor inside attacks the scheme of [17] could be defended against. In the meanwhile, Seliv et al. [18] also pointed out that the work of [17] is easily vulnerable to forgery attacks. Moreover, a new secure IBBSC is constructed in [18] that significantly compensates for the shortcomings mentioned above. Later, Kim and Hwang [19] demonstrated an efficient IBBSC scheme. Wherein, the performance analysis shows that their scheme outperforms [18] in terms of computational and communication overhead. In order to further enhance the security, Luo [21] displayed a novel IBBSC solution based on symmetric prime-order groups. In this scheme, both register secrecy and forward secrecy are ensured. However, the constructions based on symmetric prime-order groups have been demonstrated to have serious security risks [31]. Very recently, Zhao et al. [20] presented an IBBSC scheme based on asymmetric prime-order groups to address the vehicle platoon communication in the condition of single-hop multicast. In addition to inheriting the advantages of IBBSC, their solution also implements constant-size ciphertext, dramatically reducing communication and unsigncryption overhead. Nevertheless, their solution does not guarantee security due to the collusion attacks. That is, until now, there has been no such IBBSC solution that securely and efficiently enables flexible one-to-many information communication between vehicles under the premise of ensuring data authenticity.

III. Definitions and Reviews

In this section, we introduce the definition of TRBS, security game definitions, and review and analyze Zhao et al.'s scheme.

A. Definition of TRBS Framework

There are four algorithms involved in our TRBS framework: **Setup**, **KeyGen**, **Signcrypt** and **Unsigncrypt**. In which, the first two algorithms are performed by a trusted authority, and the last two algorithms are conducted respectively by the data sender/owner and data recipients/receivers.

- **Setup**$(\lambda, \ell_{\mathsf{max}}) \rightarrow (\mathsf{msk}, \mathsf{pk})$: With the input a security parameter λ, the maximum number of allowed users ℓ_{max}, it outputs the master secret key msk and the public key pk.
- **KeyGen**$(\mathsf{pk}, \mathsf{id}) \rightarrow \mathsf{sk}$: Given pk, an identity id, it produces the secret key sk.
- **Signcrypt**$(\mathsf{pk}, \mathsf{sk}_s, \mathcal{L}, m) \rightarrow \sigma$: With the input pk, a message m, sender's secret key sk_s, a group of identities $\mathcal{L} = \{\mathsf{id}_i\}_{i=1}^{\ell}$, where $\ell \leq r_{\mathsf{max}}$, it outputs a ciphertext σ.
- **Unsigncrypt**$(\mathsf{pk}, \mathsf{sk}_{\mathsf{id}_i}, \mathcal{L}, m) \rightarrow \perp m/\perp$: Given the ciphertext σ and the receiver's secret key $\mathsf{sk}_{\mathsf{id}_i}$, it outputs $m = m'$ if the validity of signing on m holds; otherwise, it aborts and returns $\perp$.

The TRBS is *sound* if each algorithm is honestly performed. That is to say, for any ciphertext $\sigma \leftarrow$ **Signcrypt**$(\mathsf{pk}, \mathsf{sk}_s, \mathcal{L}, m)$ and secret key $\mathsf{sk} \leftarrow$ **KeyGen**$(\mathsf{pk}, \mathsf{id})$, where $(\mathsf{msk}, \mathsf{pk}) \leftarrow$ **Setup**$(\lambda, \ell_{\mathsf{max}})$, the decryption can always output plaintext m by implementing **Unsigncrypt**$(\mathsf{pk}, \mathsf{sk}_{\mathsf{id}_i}, \mathcal{L}, m) \rightarrow m$.

B. Definitions of Security Games

The security games are always conducted between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$, which are used to measure the ability of adversaries and ensure the properties of the TRBS scheme.

Definition 1: Our TRBS scheme is secure against chosen plaintext attacks if the general decisional Diffie-Hellman assumption holds. If the advantage of $\mathcal{A}$ in winning this game is negligible, then our TRBS is secure.

- **Init**: A group of identities $\mathcal{L} = \{\mathsf{id}_1^*, \ldots, \mathsf{id}_\ell^*\}$ $\mathcal{A}$ wants to attack is picked-then-sent to $\mathcal{C}$.
- **Setup**: To build the system public key, $\mathcal{C}$ performs **Setup**$(\lambda, \ell_{\mathsf{max}}) \rightarrow (\mathsf{msk}, \mathsf{pk})$ and sends the created public key pk to $\mathcal{A}$.
- **Phases 1 & 2**: $\mathcal{A}$ sends the secret key query to $\mathcal{C}$, in response, $\mathcal{C}$ performs **KeyGen**$(\mathsf{pk}, \mathsf{id}) \rightarrow \mathsf{sk}$ and sends the created secret key sk to $\mathcal{A}$.
- **Challenge**: $\mathcal{A}$ picks two equal-length messages m_0, m_1, $\mathcal{C}$ runs **Unsigncrypt**$(\mathsf{pk}, \mathsf{sk}_{\mathsf{id}_i}, \mathcal{L}, m_\xi)$ to create a ciphertext σ as follows:
- **Guess**: Finally, $\mathcal{A}$ outputs a guess ξ' as the result of $\mathcal{C}$ and if $\xi' = \xi$ and the game is won.

Definition 2: Our TRBS scheme is secure against forgery attacks if the computational bilinear Diffie-Hellman (CBDH) problem holds. If the advantage of $\mathcal{A}$ in winning this game is negligible, then our TRBS achieves existential unforgeability security.

- **Setup**: $\mathcal{C}$ performs **Setup**$(\lambda, \ell_{\mathsf{max}}) \rightarrow (\mathsf{msk}, \mathsf{pk})$ and sends $\mathcal{A}$ the public parameter pk.
- **Query phase**: $\mathcal{A}$ delivers the secret key query to $\mathcal{C}$, in response, $\mathcal{C}$ performs **KeyGen**$(\mathsf{pk}, \mathsf{id}) \rightarrow \mathsf{sk}$ and sends the created secret key sk to $\mathcal{A}$.
- **Forgery**: $\mathcal{A}$ sends partial challenge ciphertext $(c_2, \mathsf{id}, \mathsf{id}')$ to $\mathcal{C}$, where c_2 is a cihphertext produced by $\mathcal{C}$. In response, $\mathcal{C}$ outputs a result as the answer of $\mathcal{A}$ if the ciphertext is valid.

C. Review and Analysis of Zhao Et Al.'s Work

In this part, we first review Zhao et al.'s work [20] and give a brief security analysis to depict the security defect.

- **Setup**$(\lambda, \ell_{\mathsf{max}})$: With the input a security parameter λ, it picks a type-III bilinear group $\mathcal{B} = (p, \mathbb{G}_0, \mathbb{G}_1, \mathbb{G}_2, e, g, h)$ of prime order p, where g, h are corresponding generators of $\mathbb{G}_0$ and $\mathbb{G}_1$. It also chooses five hash functions: $\mathcal{H}_0 : \{0,1\}^* \rightarrow \mathbb{Z}_p$, $\mathcal{H}_1 : \{0,1\}^* \times \mathbb{G}_1 \rightarrow \mathbb{Z}_p$, $\mathcal{H}_2 : \mathbb{G}_2 \rightarrow \{0,1\}^\rho$. Next, it selects $\beta \in \mathbb{Z}_p$, sets the master secret key $\mathsf{msk} =$

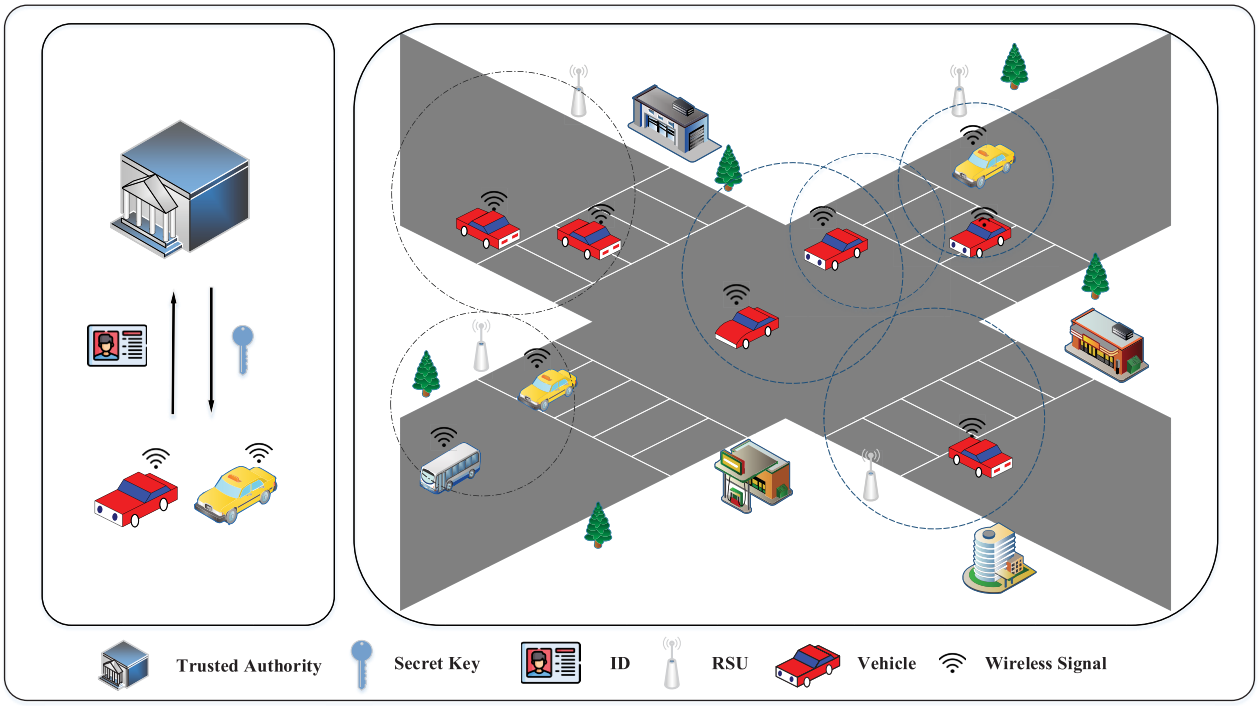


Fig. 1. System model of TRBS.

(g, β) and the public key $\mathsf{pk} = (\omega = g^{\beta}, h^{\beta}, \hat{\omega} = e(g,h), h, \ldots, h^{\beta^{r_{\max}}}, \{\mathcal{H}_i\}_{i=0}^{2})$.

- **KeyGen** $(\mathsf{pk}, \mathsf{id})$: Given pk, an identity id, it produces the secret key $\mathsf{sk}_{\mathsf{id}} = g^{\frac{1}{\beta+\mathcal{H}_0(\mathsf{id})}}$.
- **Signcrypt** $(\mathsf{pk}, \mathsf{sk}_s, \mathcal{L}, m)$: With the input pk, a message m, sender's secret key sk_s, a group of identities $\mathcal{L} = \{\mathsf{id}_i\}_{i=1}^{\ell}$, where $\ell \leq r_{\max}$, it picks $r_1, r_2 \in \mathbb{Z}_p$ and computes $X = \hat{\omega}^{r_1}$, $C_1 = \mathsf{sk}_{\mathsf{id}_s}^{r_2}$, $C_2 = \omega^{-r_1}$, $C_3 = m \oplus \mathcal{H}_2(X)$, $r = \mathcal{H}_1(m, X, C_1, C_2)$, $C_4 = h^{r_1 \prod_{i=1}^{\ell}(\beta+\mathcal{H}_0(\mathsf{id}_i))}$, $C_5 = r_2^{-1}(r_1 + r) \mod p$. Finally, it outputs a ciphertext $\sigma = (C_1, C_2, C_3, C_4, C_5, \mathcal{L})$.
- **Unsigncrypt** $(\mathsf{pk}, \mathsf{sk}_{\mathsf{id}_i}, \mathcal{L}, m)$: Given the ciphertext σ and the receiver's secret key $\mathsf{sk}_{\mathsf{id}_i}$, it first computes

$$X' = (e(C_2, h^{\rho}) e(C_4, \mathsf{sk}_{\mathsf{id}_i}))^{\frac{1}{\prod_{j=1, j\neq i}^{\ell} \mathcal{H}_0(\mathsf{id}_j)}}, Z = C_1^{C_5},$$
$$m' = C_3 \oplus \mathcal{H}_2(X'^{-1}), r' = \mathcal{H}_1(m, X'^{-1}, C_1, C_2),$$

where $\rho = \frac{1}{\beta}(\prod_{j=1, j\neq i}^{\ell} \mathcal{H}_0(\mathsf{id}_j) - \prod_{j=1, j\neq i}^{\ell}(\beta + \mathcal{H}_0(\mathsf{id}_j)))$. It next verifies whether $X'^{-1} = e(Z, h^{\beta} h^{\mathcal{H}_0(\mathsf{id}_s)}) \cdot \hat{\omega}^{r'}$. If the above equation holds, it outputs $m = m'$ and firmly believes in the validity of signing on m; otherwise, it aborts and returns $\perp$.

Analysis: Although the author claimed that the scheme is secure, we found that collusion attacks exist in this scheme, which makes the scheme insecure. In general construction, the user cannot leak his/her secret keys to adversaries but can share some computations. In the Zhao *et al.'s* scheme, since the data is shared with multiple users, these users can compute to obtain $r = \mathcal{H}_1(m, X, C_1, C_2)$. Once r is leaked, the scheme is insecure because the access control list does not work. That is to say, any unauthorized users can also access the plaintext by bypassing the authorized list via the collusion attacks. To be more specific, any adversary can perform the calculation $C_1^{C_5} = g^{\frac{r_1+r}{\beta+\mathcal{H}_0(\mathsf{id}_s)}}$ and $A = e(C_1^{C_5}, h^{\beta+\mathcal{H}_0(\mathsf{id}_s)}) = e(g,h)^{r_1+r}$. If an authorized user leaks r to the adversary, then the adversary can calculate $B = A \cdot e(g,h)^{-r} = e(g,h)^{r_1} = X$. Then, the adversary (*i.e.*, unauthorized users) can recover the plaintext m via colluding with any authorized user.

IV. Models, Goals and Workflow

In this section, we first introduce the system model of our TRBS, then illustrate the threat model and finally show the workflow of TRBS to be deployed in IoAV applications.

A. System Model

As illustrated in Fig. 1, our TRBS system is made up of three entities: a trusted authority (TA), roadside units (RSUs) and vehicles.

- A TA (*e.g.*, traffic authority) takes charge of the initialization of the system public parameters. When a vehicle requests registration, the TA is also responsible for authenticating it and issuing a private key for it.
- RSUs are usually installed on the roadside, which communicates with vehicles via wireless channels. In our system, the responsibility of RSUs is to collect information from vehicles and forward it to other vehicles to avoid traffic congestion.
- Every vehicle equipped with an On-Board Unit (OBU) is assumed to be a highly mobile node. Only registered vehicles are allowed to interact with RSUs or other vehicles by broadcasting the information. In the following

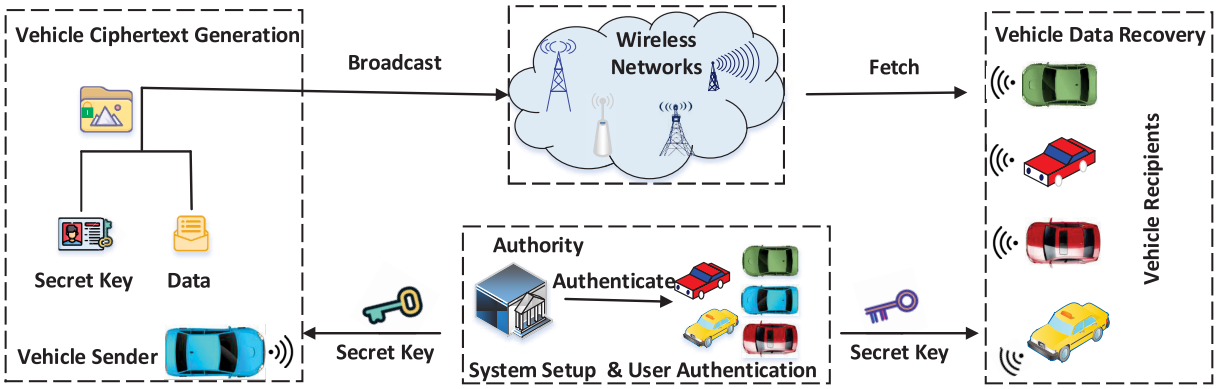


Fig. 2. IoAW workflow with our TRBS.

manuscript, we divide vehicles into two vehicle roles: vehicle senders and vehicle recipients.

B. Threat Model and Security Goals

In our threat model, the vehicle sender is considered a fully-trusted entity that embeds his/her secret key to the ciphertext of communication data for ensuring integrity and unforgeability [32], [33], [34]. The RSUs are assumed to be honest but curious, which broadcasts the encrypted data to all system vehicle users but may be controlled by malicious vehicles to learn some knowledge of encrypted broadcasting data. Vehicle recipients are divided into two categories. One case is unauthorized vehicle users who also attempt to launch various attacks including collusion attacks, forgery attacks, etc., to breach the integrity and authenticity of communication data [35], [36], [37], [38]. Besides, unauthorized vehicle users also try to access communication data without valid authorization. Another case is authorized vehicle users who are permitted to collaborate with ungranted vehicle users to capture valid access. Under the above threat model, our security goals are summarized as follows:

- *Confidentiality of communication data.* The communication data can only be accessed by authorized vehicle users. Without legitimate private keys, any malicious vehicle user can learn nothing from encrypted communication data.
- *Authenticity of communication data.* The communication data cannot be edited, forged, or tampered with by malicious vehicle users and cloud servers if they have no valid encryption keys.
- *Security assurance*. Even if malicious vehicle users launch various collusion attacks, forgery attacks, and other passive attacks, our security model can also be immune to them, which prevents adversaries from learning/forging some communication data.

C. IoAV Workflow

As defined in our TRBS framework shown in Section III-A, there are four algorithms including **Setup**, **KeyGen**, **Signcrypt**, **Unsigncrypt** for our IoAV scenarios, which are used for *system setup*, *user authentication*, *vehicle ciphertext generation* and *vehicle data recovery*. For more details of the above algorithms, please refer to Section V. For the secure data communication in the IoAV applications shown in Fig. 2, the trusted authority (*i.e.*, government's transportation sector) performs the **Setup** algorithm to build the system public information in the phase of *system setup* and implement the **KeyGen** algorithm to create secret keys for all registered vehicle users in the user authentication phase. In the vehicle ciphertext generation, the vehicle sender carries out the **Signcrypt** algorithm to produce a ciphertext related to his/her signature for broadcasting and sharing with other vehicle recipients via RSUs. The legitimate vehicle recipients can fetch the communication data ciphertext and recover and access the communication data by calling the **Unsigncrypt** algorithm in the vehicle data recovery phase.

V. Concrete Construction

In this section, a novel identity-based broadcast signcryption (IBBSC) is proposed to construct a tamper-resistant broadcasting (TRBS) scheme for secure communication in IoAV and its soundness is given for a better understanding of our construction.

A. Identity-Based Broadcast Signcryption (IBBSC)

- **Setup**$(\lambda, \ell_{\mathsf{max}})$: With the input a security parameter λ, it picks a type-III bilinear group $\mathcal{B} = (p, \mathbb{G}_0, \mathbb{G}_1, \mathbb{G}_2, e, g, h)$ of prime order p, where g, h are corresponding generators of $\mathbb{G}_0$ and $\mathbb{G}_1$. It also chooses five hash functions: $\mathcal{H}_0 : \{0,1\}^* \rightarrow \mathbb{G}_0$, $\mathcal{H}_1 : \{0,1\}^* \rightarrow \mathbb{G}_1$, $\mathcal{H}_2 : \{0,1\}^* \rightarrow \mathbb{Z}_p$, $\mathcal{H}_3 : \mathbb{G}_1 \times \mathbb{G}_2 \times \mathbb{G}_1^2 \times \mathbb{G}_2 \times \mathbb{Z}_p^{\ell_{\mathsf{max}}} \rightarrow \mathbb{Z}_p$, $\mathcal{H}_4 : \mathbb{G}_2 \rightarrow \mathbb{Z}_p$. Next, it selects $\alpha, \beta, \tau \in \mathbb{Z}_p$, sets the master secret key $\mathsf{msk} = (\alpha, \beta)$ and the public key $\mathsf{pk} = (\omega = g^\alpha, \omega' = h^\alpha, \varpi = g^\tau, \varpi' = h^\tau, \hat{\omega} = e(\omega, h), g, g^\beta, g^{\beta^2}, \ldots, g^{\beta^{r_{\mathsf{max}}}}, h^\beta, \ldots, h^{\beta^{r_{\mathsf{max}}}}, \{\mathcal{H}_i\}_{i=0}^4)$.
- **KeyGen** $(\mathsf{pk}, \mathsf{id})$: Given pk, an identity id, it produces the secret key $\mathsf{sk} = (\mathsf{sk}_{\mathsf{id},0}, \mathsf{sk}_{\mathsf{id},1})$, where $\mathsf{sk}_{\mathsf{id},0} = \mathcal{H}_1(\mathsf{id})^\alpha$, $\mathsf{sk}_{\mathsf{id},1} = g^{\frac{\alpha}{\beta + \mathcal{H}_2(\mathsf{id})}}$.
- **Signcrypt** $(\mathsf{pk}, \mathsf{sk}_s, \mathcal{L}, m)$: With the input pk, a message m, sender's secret key, a group of identities $\mathcal{L} = \{\mathsf{id}_i\}_{i=1}^{\ell}$, where $\ell \leq r_{\mathsf{max}}$, it picks $t, \gamma \in \mathbb{Z}_p$, computes $C_1 = \omega^t$. For each $\mathsf{id}_i \in \mathcal{L}$, it calculates $z_i = \mathcal{H}_4(e(\mathcal{H}_1(\mathsf{id}_i), \omega^t \cdot \mathcal{H}_0(\mathsf{id}_s)^\alpha))$ and $\prod_{i=1}^{\ell}(x - z_i) + \gamma = \sum_{i=0}^{\ell-1} a_i x^i + x^\ell \mod p$, where a_i are coefficients of the above polynomial.

It next counts $C_2 = h^{\gamma t}\prod_{i=1}^{\ell}(\beta+\mathcal{H}_2(\mathsf{id}_i))$, $C_3 = \hat{\omega}^t \cdot m$, $C_4 = \varpi^{\mathcal{H}_3(m,\hat{\omega}^t,C_1,C_2,C_3,a_0,\ldots,a_{\ell-1})t}$. Finally, it outputs a ciphertext $\sigma = (C_1, C_2, C_3, C_4, a_0, \ldots, a_{\ell-1}, \mathcal{L})$.

- **Unsigncrypt** $(\mathsf{pk}, \mathsf{sk}_{\mathsf{id}_i}, \mathcal{L}, m)$: Given the ciphertext σ and the receiver's secret key $\mathsf{sk}_{\mathsf{id}_i}$, it first computes $z_i = \mathcal{H}_4(e(\mathcal{H}_0(\mathsf{id}_s), \mathsf{sk}_{\mathsf{id}_i,0})e(\mathcal{H}_1(\mathsf{id}_i), C_1))$, $\gamma = \sum_{i=0}^{\ell-1} a_i(z_i)^i + z_i^{\ell} \mod p$. It next computes $A = (e(C_1, h^{\rho})e(C_2^{1/\gamma}, \mathsf{sk}_{\mathsf{id}_i,1}))^{\frac{1}{\prod_{j=1,j\neq i}^{\ell}\mathcal{H}_2(\mathsf{id}_j)}}$, where $\rho = \prod_{j=1,j\neq i}^{\ell}\mathcal{H}_2(\mathsf{id}_j) - \prod_{j=1,j\neq i}^{\ell}(\beta + \mathsf{id}_j)$ and deciphers $m' = C_3/A$. If both $e(C_2C_4, \omega') = e(h^{\gamma\prod_{i=1}^{\ell}(\beta+\mathcal{H}_2(\mathsf{id}_i))}\varpi^{\mathcal{H}_3(m',A,C_1,C_2,C_3,a_0,\ldots,a_{\ell-1})}, C_1)$ holds, it outputs $m = m'$ and firmly believes the validity of signing on m. Otherwise, it aborts and returns $\perp$.

Remark: It is worth noting that even if γ is leaked to unauthorized users by legitimate users, our IBBSC can also offer strong security against collusion attacks.

B. Soundness of Our IBBSC

If the ciphertext and the secret key are legitimate, namely $\sigma = (C_1, C_2, C_3, C_4, a_0, \ldots, a_{\ell-1}, \mathcal{L})$ and $\mathsf{sk} = (\mathsf{sk}_{\mathsf{id},0}, \mathsf{sk}_{\mathsf{id},1})$, then we perform the following computations:

$$\begin{aligned}
z_i &= \mathcal{H}_4(e(\mathcal{H}_0(\mathsf{id}_s), \mathsf{sk}_{\mathsf{id}_i,0})e(\mathcal{H}_1(\mathsf{id}_i), C_1))\\
&= \mathcal{H}_4(e(\mathcal{H}_0(\mathsf{id}_s), \mathcal{H}_1(\mathsf{id}_i)^{\alpha}) \cdot e(\mathcal{H}_1(\mathsf{id}_i), C_1))\\
&= \mathcal{H}_4(e(\mathcal{H}_1(\mathsf{id}_i), \omega^t \cdot \mathcal{H}_0(\mathsf{id}_s)^{\alpha})),\\
\gamma &= \sum_{i=0}^{\ell-1} a_i(z_i)^i + z_i^{\ell} \mod p,\\
A' &= e(C_1, h^{\rho})e(C_2^{1/\gamma}, \mathsf{sk}_{\mathsf{id}_i,1})\\
&= e(g^{\alpha t}, h^{\rho})e(h^t\prod_{i=1}^{\ell}(\beta+\mathcal{H}_2(\mathsf{id}_i)), g^{\frac{\alpha}{\beta+\mathcal{H}_2(\mathsf{id}_i)}})\\
&= e(\omega, h)^{t(\prod_{j=1,j\neq i}^{\ell}\mathcal{H}_2(\mathsf{id}_j) - \prod_{j=1,j\neq i}^{\ell}(\beta+\mathsf{id}_j))}.\\
&\quad e(\omega, h)^{t\prod_{j=1,j\neq i}^{\ell}(\beta+\mathsf{id}_j)}\\
&= e(\omega, h)^{t\prod_{j=1,j\neq i}^{\ell}\mathcal{H}_2(\mathsf{id}_j)},\\
A &= A'^{\frac{1}{\prod_{j=1,j\neq i}^{\ell}\mathcal{H}_2(\mathsf{id}_j)}} = e(\omega, h)^t.
\end{aligned}$$

After capturing A, $m' = C_3/A$ can be then derived. Then, we performs the signature verification by checking $e(C_2C_4, \omega') = e(h^{\gamma\prod_{i=1}^{\ell}(\beta+\mathcal{H}_2(\mathsf{id}_i))}\varpi^{\mathcal{H}_3(m',A,C_1,C_2,C_3,a_0,\ldots,a_{\ell-1})}, C_1)$. If the above equations hold, the plaintext $m = m'$ is deciphered and returned.

VI. Security Analysis

In this section, we show a detailed security analysis via the following theorems to demonstrate that our TRBS scheme can achieve CPA security and authenticity.

Theorem 1: If the following general decisional Diffie-Hellman (GDDHE) assumption holds, our scheme is secure against chosen plaintext attacks (CPA).

Proof 1: Assuming that $\mathcal{A}$ enables breaching our scheme, then another algorithm $\mathcal{C}$ can be constructed via the interaction with $\mathcal{A}$ to solve the intractable GDDHE assumption. Given an instance of the GDDHE assumption:

$$g_0, g_0^{\beta}, \ldots, g_0^{\beta^{s-1}}, g_0^{\alpha f(\beta)}, g_0^{\tau f(\beta)}, g_0^{\alpha t f(\beta)}$$
$$h_0, h_0^{\beta}, \ldots, h_0^{\beta^{2n}}, h_0^{\alpha\beta}, \ldots, h_0^{\alpha\beta^{2n}}, h_0^{\alpha g(\beta)}, h_0^{\tau g(\beta)}, h_0^{tg(\beta)}, T$$

where $f(.), g(.)$ are two co-prime polynomials with pairwise various roots of respective orders s and n. The specific definitions of $f(.), g(.)$ are: $f(Z) = \prod_{i=1}^{s}(Z + x_i)$, $g(Z) = \prod_{i=s+1}^{s+n}(Z + x_i)$. For $i \in [1, s]$, $f_i(z) = \frac{f(z)}{z+x_i}$ and for $i \in [s+1, s+n]$, $g_i(z) = \frac{g(z)}{z+x_i}$. The goal of $\mathcal{C}$ is to judge $T = e(g_0, h_0)^{tf(\beta)}$ or T is a random element of $\mathbb{G}_0$.

- **Init**: A group of identities $\mathcal{L} = \{\mathsf{id}_1^*, \ldots, \mathsf{id}_{\ell}^*\}$ $\mathcal{A}$ wants to attack is picked-then-sent to $\mathcal{C}$.
- **Setup**: To build the system public key, $\mathcal{C}$ formalizes $g = g_0^{f(\beta)}$ and sets $h = h_0^{\prod_{i=s+\ell^*+1}^{s+n}(\beta+x_i)}$, $\omega = g_0^{\alpha f(\beta)}$, $\omega' = h_0^{\alpha\prod_{i=s+\ell^*+1}^{s+n}(\beta+x_i)}$, $\varpi = g_0^{\tau f(\beta)}$, $\varpi' = h_0^{\tau\prod_{i=s+\ell^*+1}^{s+n}(\beta+x_i)}$, $\hat{\omega} = e(g_0, h_0)^{\alpha f(\beta)\prod_{i=s+\ell^*+1}^{s+n}(\beta+x_i)}$. Then, $\mathcal{C}$ sends the defined public key $\mathsf{pk} = (\omega, \omega', \varpi = g^{\tau}, \varpi', \hat{\omega}, h, h^{\beta}, \ldots, h^{\beta^n})$ to $\mathcal{A}$.
- **Phases 1 & 2**: The following hash queries and secret key queries can be made by $\mathcal{A}$:
  - $\mathcal{H}_0$-Queries: A random oracle on any identity id_i can be queried. To respond to these queries, a list $\mathcal{L}_{\mathcal{H}_0}$ of the tuple (id_i, s_i) is maintained. If the list $\mathcal{L}_{\mathcal{H}_0}$ contains the identity id_i, $\mathcal{C}$ responds with the corresponding s_i. If the information of the identity id_i is excluded in the list $\mathcal{L}_{\mathcal{H}_0}$, $\mathcal{C}$ sets $\mathcal{H}_0(\mathsf{id}_i) = g_0^{f(\beta)s_i} = g^{s_i}$ and updates the list with (id_i, s_i).
  - $\mathcal{H}_1$-Queries: A random oracle on any identity id_i can be queried. To respond to these queries, a list $\mathcal{L}_{\mathcal{H}_1}$ of the tuple (id_i, s_i') is maintained. If the list $\mathcal{L}_{\mathcal{H}_1}$ contains the identity id_i, $\mathcal{C}$ responds with the corresponding s_i'. If the information of the identity id_i is out of the list $\mathcal{L}_{\mathcal{H}_1}$, $\mathcal{C}$ sets $\mathcal{H}_1(\mathsf{id}_i) = h_0^{\prod_{i=s+\ell^*+1}^{s+n}(\beta+x_i)s_i'} = h^{s_i'}$ and updates the list with (id_i, s_i').
  - $\mathcal{H}_2$-Queries: A random oracle on any identity id_i can be queried. To respond to these queries, a list $\mathcal{L}_{\mathcal{H}_2}$ of the tuple $(\mathsf{id}_i, x_i, \mathsf{sk}_{\mathsf{id}_i})$ is maintained. The tuple $(\mathsf{id}_i, x_i, \mathsf{sk}_{\mathsf{id}_i})$ contains $\{(*, x_i, *)_{i=1}^{s}, \{(\mathsf{id}_i, x_i, *)_{i=s+1}^{s+\ell^*}\}$. When $\mathcal{A}$ makes queries of the identity id_i, $\mathcal{C}$ responds to the queries as follows: If the list $\mathcal{L}_{\mathcal{H}_2}$ contains the identity id_i, $\mathcal{C}$ responds with the corresponding x_i. If the information of the identity id_i is not included in the list $\mathcal{L}_{\mathcal{H}_2}$, $\mathcal{C}$ sets $\mathcal{H}_2(\mathsf{id}_i) = x_i$ and updates the list with $(\mathsf{id}_i, x_i, *)$.
  - $\mathcal{H}_3$-Queries: For the query on the tuple $X_i = (m, \hat{\omega}^t, C_1, C_2, C_3, a_0, \ldots, a_{\ell-1})$, $\mathcal{C}$ sets $\mathcal{H}_3(X_i) = h_{3,i}$ and updates the list $\mathcal{L}_{\mathcal{H}_3}$ with $(X_i, h_{3,i})$.
  - Secret Key Queries: For the secret key query on identity id_i, $\mathcal{C}$ implements **KeyGen** on $\mathsf{id}_i \notin \mathcal{L}$ and sends the resulting secret key to $\mathcal{A}$. If the secret key query on id_i has been made, then $\mathcal{C}$ responds the secret key $\mathsf{sk}_{\mathsf{id}_i}$ in the list $\mathcal{L}_{\mathcal{H}_2}$ to $\mathcal{A}$; else, if a hash query on id_i has been also issued, $\mathcal{C}$ utilizes the corresponding x_i to calculate $\mathsf{sk}_{\mathsf{id},0} = \mathcal{H}_1(\mathsf{id}_i)^{\alpha} =$

TABLE I
EFFICIENCY COMPARISONS OF OUR TRBS WITH RELATED WORKS

Scheme	Secure Data Confidentiality	Data Authenticity	Ciphertext length	sk length	Signcryption costs	Unsigncryption costs
LZX [21]	✗	✓	$(n+3)\|\mathbb{G}\|$	$\|\mathbb{G}\|$	$(n+4)e_0+e_1+p$	$(n+2)e_0+4p$
ZWL+ [20]	✗	✓	$3\|\mathbb{G}\|+\|\mathbb{Z}_p^*\|$	$\|\mathbb{G}\|$	$(n+2)e_0+e_1$	$(n+1)e_0+2e_1+3p$
Our TRBS	✓	✓	$4\|\mathbb{G}\|+n\|\mathbb{Z}_p^*\|$	$2\|\mathbb{G}\|$	$(2n+3)e_0+np$	$(n+3)e_0+e_1+6p$

$h_0^{\alpha\prod_{i=s+\ell^*+1}^{s+n}(\beta+x_i)s_i'}$, $\mathsf{sk}_{\mathsf{id},1}=g_0^{\alpha g_i(\beta)}=g^{\frac{\alpha}{\beta+\mathcal{H}_2(\mathsf{id}_i)}}$; otherwise, $\mathcal{C}$ sets $\mathcal{H}_2(\mathsf{id}_i)=x_i$ and computes the corresponding secret key $\mathsf{sk}_{\mathsf{id}_i}$.

- **Challenge**: $\mathcal{A}$ picks two equal-length messages m_0, m_1, $\mathcal{C}$ runs **Signcrypt** to create a ciphertext σ as follows:
  - $\mathcal{C}$ first sets $C_1=g_0^{\alpha t f(\beta)}=\omega^t$, then produces $\gamma, a_0, \ldots, a_{\ell-1}$ as described in **Signcryption** algorithm and computes $C_2=h_0^{t\gamma g(\beta)}=h_0^{\gamma t\prod_{i=s+\ell^*+1}^{s+n}(\beta+x_i)\prod_{i=s+1}^{s+\ell^*}(\beta+x_i)}=h^{\gamma t\prod_{i=s+1}^{s+\ell^*}(\beta+x_i)}$.
  - $\mathcal{C}$ then sets $C_3=m_\xi\cdot T^{\prod_{i=s+\ell^*+1}^{s+n}x_i}\cdot e(g_0^{t\alpha f(\beta)},h_0^{q(\beta)})$ and $C_4=\varpi^{h_{3,i}}$, where $q(\beta)=\prod_{i=s+\ell^*+1}^{s+n}(\beta+x_i)-\prod_{i=s+\ell^*+1}^{s+n}x_i$.
  - $\mathcal{C}$ finally returns the ciphertext $\sigma=(C_1,C_2,C_3,C_4,a_0,\ldots,a_{\ell-1})$ to $\mathcal{A}$.
- **Guess**: Finally, $\mathcal{A}$ outputs a guess ξ' as the result of $\mathcal{C}$ and if $\xi'=\xi$ and the game is won.

Theorem 2: Assuming that an adversary $\mathcal{A}$ can breach the authenticity with some overwhelming advantage, then an algorithm $\mathcal{C}$ can be created via interacting with $\mathcal{A}$ to output the result of the following computational bilinear Diffie-Hellman (CBDH) problem.

Proof 2: Given an instance $(g,g^a,g^b,g^c,h^a,h^b,h^c,T)$ of CBDH, the goal of $\mathcal{C}$ is to output $T=e(g,h)^{abc}$.

- **Setup**: $\mathcal{C}$ sends $\mathcal{A}$ the public parameter $\mathsf{pp}=(\mathcal{BG},\omega=g^b,\omega'=h^b,\{\mathcal{H}_i\}_{i\in\{0,1,2,4\}})$, where $\{\mathcal{H}_i\}_{i\in\{0,1,2,4\}}$ are the random oracles controlled by $\mathcal{C}$.
- **Query phase**: The following queries can be adaptively issued by $\mathcal{A}$:
  - $\mathcal{H}_0$-Query: If there exists a query id_i in a tuple $(\mathsf{id}_i,W_i,\eta_i,s_i)$, then return W_i; otherwise, produce random $\eta_i\in\mathbb{Z}_p$ and $s_i\in\{0,1\}$, such that the probability of $s_i=0$ is ξ. If $s_i=0$, then calculate $W_i=g^{\eta_i}$; otherwise let $W_i=g^{a\eta_i}$. Finally, the list $\mathcal{L}_0$ is added $(\mathsf{id}_i,W_i,\eta_i,s_i)$ and W_i is returned to $\mathcal{A}$.
  - $\mathcal{H}_1$-Query: If there exists a query id_i' in a tuple $(\mathsf{id}_i',W_i,\eta_i',s_i)$, then return W_i; otherwise, produce random $\eta_i'\in\mathbb{Z}_p$ and $s_i\in\{0,1\}$, such that the probability of $s_i=0$ is ξ. If $s_i=0$, then calculate $W_i=h^{\eta_i'}$; otherwise let $W_i=h^{c\eta_i'}$. Finally, the list $\mathcal{L}_1$ is added $(\mathsf{id}_i',W_i,\eta_i',s_i)$ and W_i is returned to $\mathcal{A}$.
  - $\mathcal{H}_2$-Query: The list $\mathcal{L}_2$ is maintained by $\mathcal{C}$ to store the tuples of (id_i,t_i). If the $\mathcal{H}_2$-query on id_i had been queried, then $\mathcal{C}$ returns the t_i; otherwise, $\mathcal{C}$ randomly chooses $t_i\in\mathbb{Z}_p$, adds the new tuple (id_i,t_i) into $\mathcal{L}_2$ and outputs t_i to $\mathcal{A}$.
  - $\mathcal{H}_4$-Query: The list $\mathcal{L}_3$ is maintained by $\mathcal{C}$ to store the tuples of $(Q_i,\hat{t}_i)$. If the query Q_i had been queried, then $\mathcal{C}$ returns the $\hat{t}_i$; otherwise, $\mathcal{C}$ randomly chooses $\hat{t}_i\in\mathbb{Z}_p$, adds the new tuple $(Q_i,\hat{t}_i)$ into $\mathcal{L}_3$ and outputs $\hat{t}_i$ to $\mathcal{A}$.
  - Secret Key Query: With the input id to the oracle of $\mathcal{H}_1$, the result $\mathcal{H}_1(\mathsf{id})=W_i$ can be obtained from $(\mathsf{id}_i,W_i,\eta_i,s_i)$ of $\mathcal{L}_1$. With the input id to the oracle of $\mathcal{H}_2$, the result $\mathcal{H}_2(\mathsf{id}_i)=t_i$ can be obtained from (id_i,t_i) of $\mathcal{L}_2$. If $s_i=1$, abort and return $\perp$; else, output $\mathsf{sk}_{\mathsf{id},0}=h^{b\eta_i}$ and $\mathsf{sk}_{\mathsf{id},1}=g^{b/(\beta+t_i)}$, where β is randomly chosen from $\mathbb{Z}_p$.
- **Forgery**: $\mathcal{A}$ sends $(c_2,\mathsf{id},\mathsf{id}')$ to $\mathcal{C}$. In response, $\mathcal{C}$ performs the steps as follows:
  1) Calculate $\mathcal{H}_0(\mathsf{id})=W$ and $\mathcal{H}_1(\mathsf{id})=W'$. If the tuples $(\mathsf{id},W,\eta,s)\in\mathcal{L}_0$ and $(\mathsf{id}',W',\eta',s')\in\mathcal{L}_1$ simultaneously do not have s,s' equal to 1, $\mathcal{C}$ aborts and returns $\perp$; otherwise, we can implicitly derive that $\mathsf{sk}_{\mathsf{id},0}=h^{cb\eta}$, $\mathcal{H}_0(\mathsf{id}')=g^{a\eta'}$, $\mathcal{H}_1(\mathsf{id}_i)=h^{\eta'}$, $C_1=g^{bt}$. So, $\mathcal{H}_4(e(\mathcal{H}_1(\mathsf{id}_i),\omega^t\cdot\mathcal{H}_0(\mathsf{id}_s)^\alpha))=\mathcal{H}_4(e(\mathcal{H}_0(\mathsf{id}'),\mathsf{sk}_{\mathsf{id}_i,0})e(\mathcal{H}_1(\mathsf{id}_i),C_1))$, where $e(\mathcal{H}_0(\mathsf{id}'),\mathsf{sk}_{\mathsf{id}_i,0})=e(g^{cb\eta},h^{a\eta'})$ and $e(\mathcal{H}_1(\mathsf{id}_i),C_1)=e(h^{\eta'},g^{bt})$.
  2) From the list $\mathcal{L}_3$, it is easy to obtain $(Q_i,\hat{t}_i)$, thus knowing $T=(Q_i\cdot e(h^{\eta'},g^{bt})^{-1})^{1/(\eta\eta')}$.

VII. PERFORMANCE EVALUATION

In this section, we present the performance evaluation including theoretical analysis and experimental analysis, which demonstrates the practicability of our TRBS.

A. Theoretical Analysis

In TABLE I, we provide an efficiency comparison of our solution with related schemes [20], [21] concerning functionality, computational cost, and communication cost. Wherein, "✓" denotes that the feature can be satisfied. In contrast, the "✗" represents that the functionality is unable to be fulfilled by this solution. Data confidentiality indicates that no adversary can recover the ciphertext without the valid keys. Data authenticity suggests that there is no way for an adversary to forge or modify ciphertext, thus realizing unforgeability. e_0, e_1 and p indicates the time for exponentiation operation in $\mathbb{G}_0$, exponentiation operation in $\mathbb{G}_2$ and bilinear pairing operation, respectively. Since the length of $\mathbb{G}_0$, $\mathbb{G}_1$, and $\mathbb{G}_2$ is the identical, we uniformly use $|\mathbb{G}|$ to display the sizes of element in $\mathbb{G}_0$, $\mathbb{G}_1$, and $\mathbb{G}_2$. In the meanwhile, $|\mathbb{Z}_p^*|$ is shown as the sizes of element in $\mathbb{Z}_p^*$. n refers to the number of receivers.

It can be seen from TABLE I that the work in LZX [21] supports data authenticity. Nevertheless, LZX [21] cannot guarantee the confidentiality of the data because its scheme is based on symmetric prime-order groups [31]. Similarly

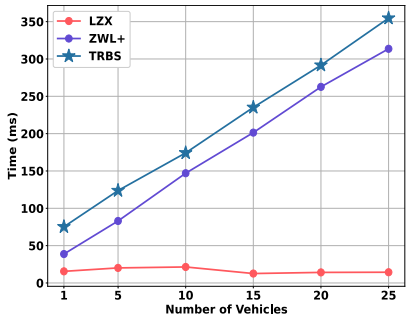


Fig. 3. Time consumption of setup.

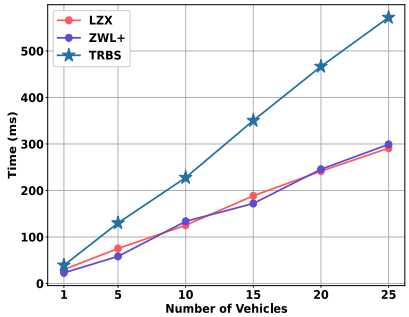


Fig. 4. Time consumption of KeyGen.

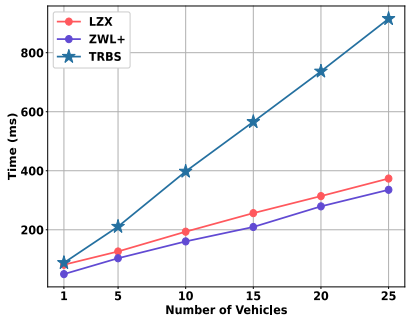


Fig. 5. Time consumption of Signcryption.

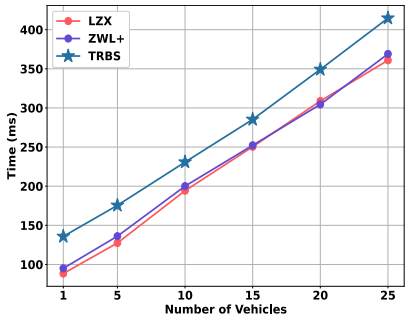


Fig. 6. Time consumption of Unsigncryption.

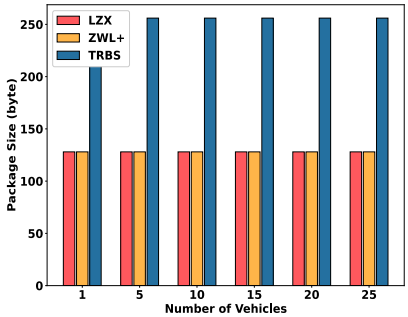


Fig. 7. Communication costs of sk.

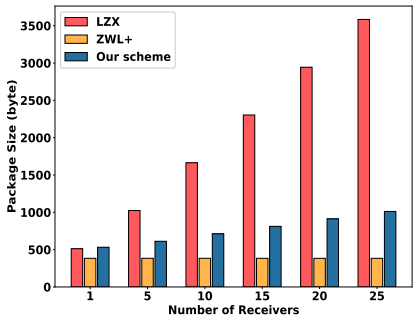


Fig. 8. Communication costs of ciphertext.

to LZX [21], ZWL+ [20] also fails to achieve data confidentiality. Worse still, data authenticity cannot be ensured in ZWL+ [20] either. Luckily, both data confidentiality and data authenticity are realized in our framework. In terms of ciphertext length, it is evident that LZX [21] and our solution increase as the number of receivers grows, while ZWL+ [20] remains constant with the number of receivers rises. The sk length in our TRBS is relatively higher than LZX [21] and ZWL+ [20]. It is visible that the time consumption of signcryption in all schemes is linearly related to the number of receivers. In terms of unsigncryption phase, the computational cost of LZX [21], ZWL+ [20], and our TRBS increases proportionally with the number of vehicle receivers.

B. Experimental Analysis

To further demonstrate the efficiency of our TRBS, we implement experiments to evaluate the performance comparison between our TRBS and other relevant schemes [20], [21] by using Python 3.8.14 with Charm framework [40]. The experimental simulations were performed on a computer with a 3.61 GHz Intel(R) Core(TM) i7-12700KF CPU running 64-bit Ubuntu 18.04.5 LTS. In order to guarantee the 80-bit security level, we adopt the SS512 curve for parsing.

Fig. 3 displays the time-consuming comparison between our TRBS and other related schemes in terms of setup algorithm. From Fig. 3, we can observe that the computational time in ZWL+ [20] and our TRBS grow linearly as the number of vehicles (receivers) increases, while that in LZX [21] is not affected by the number of vehicles. Since the setup algorithm is executed only once when the system is initialized, the computational overhead it generates has no impact on the overall efficiency of the solution. Fig. 4 shows the execution time of our TRBS compared with LZX [21] and ZWL+ [20] in the keygen algorithm. As described in Fig. 4, as the number of recipients grows, LZX [21] and ZWL+ [20] take almost the same amount of time to generate the secret key, while our TRBS takes a little more time to generate the secret key than LZX [21] and ZWL+ [20]. Fig. 5 illustrates the time consumption of our TRBS compared to other solutions in signcryption phase to generate ciphertexts. From Fig. 5, we can learn that all schemes increase linearly as the number of vehicles, and our solution is higher than that in LZX [21] and ZWL+ [20]. Fig.6 depicts a comparison of the computational

cost of our TRBS and other works for recovering plaintexts in unsigncryption algorithm. From Fig. 6, we can conclude that both LZX [21], ZWL+ [20] and our rise in proportion to the number of vehicles. Wherein, compared to other works, our time consumption of unsigncryption is slightly higher. This can be forgiven since our solution guarantees not only the confidentiality of the data but also the authenticity of data.

Fig. 7 represents the communication overhead of our TRBS compared to LZX [21] and ZWL+ [20] with respect to sk. From Fig. 7, we can find that the size of sk does not vary with the number of vehicles for all solutions. Compared with other schemes, the communication overhead of sk in our TRBS is slightly larger. Fig. 8 describes the communication cost comparison of ciphertexts between our solution and other works. From Fig. 8, we can summarize that the communication cost in LZX [21] and our TRBS rise as the number of vehicles grows, while remaining fixed in ZWL+ [20]. Besides, our TRBS and ZWL+ [20] consume lower communication costs than LZX [21].

In a nutshell, our TRBS solution features small ciphertexts, which implies fast time execution. Even though the computational overhead in our TRBS is relatively higher than other schemes, it is tolerable for IoVA because the time consumed for unsigncryption takes only about 0.4 seconds when the number of vehicles reaches 25. Furthermore, secure data confidentiality and authenticity can be ensured in our TRBS, which demonstrates its feasibility for IoAV-based secure communication applications.

VIII. Conclusion

In this paper, a tamper-resistant broadcasting (TRBS) scheme for secure communication in IoAV was suggested, which enables secure one-to-many vehicle communication under the premise of ensuring the authenticity and secrecy of communication data. Compared to other tamper-resistant broadcasting (TRBS) solutions (*i.e., IBBSC*), our TRBS scheme constructed on the asymmetric prime-order groups is more secure. In addition, the rigorous security proofs indicate our TRBS can realize CPA security and authenticity. We also implement theoretical and experimental performance evaluations to demonstrate the appropriateness of our TRBS for real-world applications. In the future, we will seek to construct a more secure and efficient tamper-resistant broadcasting (TRBS) scheme against chosen ciphertext attacks.

References

[1] A. Nanda, D. Puthal, J. J. P. C. Rodrigues, and S. A. Kozlov, "Internet of Autonomous Vehicles communications security: Overview, issues, and directions," *IEEE Wireless Commun.*, vol. 26, no. 4, pp. 60–65, Aug. 2019.

[2] S. Mumtaz, A. Al-Dulaimi, H. Gacanin, and A. Bo, "Block chain and big data-enabled intelligent vehicular communication," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 3904–3906, Jul. 2021.

[3] Y. Chen, M. Alam, and S. Mumtaz, "Aiden: Association-learning-based attack identification on the edge of V2X communication networks," *IEEE Trans. Green Commun. Netw.*, vol. 6, no. 3, pp. 1377–1385, Sep. 2022.

[4] F. Jameel, Z. Chang, J. Huang, and T. Ristaniemi, "Internet of Autonomous Vehicles: Architecture, features, and socio-technological challenges," *IEEE Wireless Commun.*, vol. 26, no. 4, pp. 21–29, Aug. 2019.

[5] H. Khelifi et al., "Named data networking in vehicular ad hoc networks: State-of-the-art and challenges," *IEEE Commun. Surveys Tuts*, vol. 22, no. 1, pp. 320–351, 1st Quart., 2020.

[6] H. Fatemidokht, M. K. Rafsanjani, B. B. Gupta, and C.-H. Hsu, "Efficient and secure routing protocol based on artificial intelligence algorithms with UAV-assisted for vehicular ad hoc networks in intelligent transportation systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 7, pp. 4757–4769, Jul. 2021.

[7] M. Obaidat, M. Khodjaeva, J. Holst, and M. Ben Zid, "Security and privacy challenges in vehicular ad hoc networks," in *Connected Vehicles in the Internet of Things*. Cham, Switzerland: Springer, 2020, pp. 223–251.

[8] J. Sun, G. Xu, T. Zhang, M. Alazab, and R. H. Deng, "A practical fog-based privacy-preserving online car-hailing service system," *IEEE Trans. Inf. Forensics Security*, vol. 17, pp. 2862–2877, 2022.

[9] Z. Su, Y. Wang, Q. Xu, and N. Zhang, "LVBS: Lightweight vehicular blockchain for secure data sharing in disaster rescue," *IEEE Trans. Dependable Secure Comput.*, vol. 19, no. 1, pp. 19–32, Jan. 2022.

[10] J. Sun, G. Xu, T. Zhang, X. Cheng, X. Han, and M. Tang, "Secure data sharing with flexible cross-domain authorization in autonomous vehicle systems," *IEEE Trans. Intell. Transp. Syst.*, early access, Mar. 16, 2022, doi: 10.1109/TITS.2022.3157309.

[11] C. Feng, K. Yu, M. Aloqaily, M. Alazab, and S. Mumtaz, "Attribute-based encryption with parallel outsourced decryption for edge intelligent IoV," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13784–13795, Nov. 2020.

[12] J. Cui, X. Chen, J. Zhang, Q. Zhang, and H. Zhong, "Toward achieving fine-grained access control of data in connected and autonomous vehicles," *IEEE Internet Things J.*, vol. 8, no. 10, pp. 7925–7937, May 2021.

[13] I. E. Carvajal-Roca and J. Wang, "A semi-decentralized security framework for connected and autonomous vehicles," in *Proc. IEEE 94th Veh. Technol. Conf.*, Sep. 2021, pp. 1–6.

[14] A. Ge and P. Wei, "Identity-based broadcast encryption with efficient revocation," in *Proc. Public Key Cryptogr. Conf.* Cham, Switzerland: Springer, 2019, pp. 405–435.

[15] L. Liu, Y. Zhang, and X. Li, "KeyD: Secure key-deduplication with identity-based broadcast encryption," *IEEE Trans. Cloud Comput.*, vol. 9, no. 2, pp. 670–681, Apr. 2021.

[16] K. He, J. Weng, J.-N. Liu, J. K. Liu, W. Liu, and R. H. Deng, "Anonymous identity-based broadcast encryption with chosen-ciphertext security," in *Proc. 11th ACM Asia Conf. Comput. Commun. Secur.*, May 2016, pp. 247–255.

[17] F. Li, X. Xin, and Y. Hu, "Indentity-based broadcast signcryption," *Comput. Standards Interface*, vol. 30, nos. 1–2, pp. 89–94, Jan. 2008.

[18] S. S. D. Selvi, S. S. Vivek, R. Gopalakrishnan, N. N. Karuturi, and C. P. Rangan, "Provably secure ID-based broadcast signcryption (IBBSC) scheme," *IACR Cryptol. ePrint Arch.*, vol. 2008, p. 225, Apr. 2008.

[19] I. T. Kim and S. O. Hwang, "An efficient identity-based broadcast signcryption scheme for wireless sensor networks," in *Proc. Int. Symp. Wireless Pervasive Comput.*, Feb. 2011, pp. 1–6.

[20] Y. Zhao, Y. Wang, Y. Liang, H. Yu, and Y. Ren, "Identity-based broadcast signcryption scheme for vehicular platoon communication," *IEEE Trans. Ind. Informat.*, early access, Sep. 2, 2022, doi: 10.1109/TII.2022.3203724.

[21] M. Luo, C. Zou, and J. Xu, "An efficient identity-based broadcast signcryption scheme," *J. Softw.*, vol. 7, no. 2, pp. 366–373, Feb. 2012.

[22] Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption) $\ll$ cost (signature) + cost (encryption)," in *Advances in Cryptology—CRYPTO*. Cham, Switzerland: Springer, 1997, pp. 165–179.

[23] J. Malone-Lee, "Identity-based signcryption," *Cryptol. ePrint Arch.*, p. 98, 2002. [Online]. Available: http://eprint.iacr.org/2002/098

[24] H. Xiong, Y. Hou, X. Huang, and Y. Zhao, "Secure message classification services through identity-based signcryption with equality test towards the Internet of Vehicles," *Veh. Commun.*, vol. 26, Dec. 2020, Art. no. 100264.

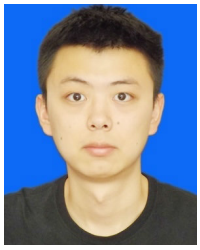
[25] H. Zhu, Y. Wang, C. Wang, and X. Cheng, "An efficient identity-based proxy signcryption using lattice," *Future Gener. Comput. Syst.*, vol. 117, pp. 321–327, Apr. 2021.

[26] Y. Yang, D. He, P. Vijayakumar, B. B. Gupta, and Q. Xie, "An efficient identity-based aggregate signcryption scheme with blockchain for IoT-enabled maritime transportation system," *IEEE Trans. Green Commun. Netw.*, vol. 6, no. 3, pp. 1520–1531, Sep. 2022, doi: 10.1109/TGCN.2022.3163596.

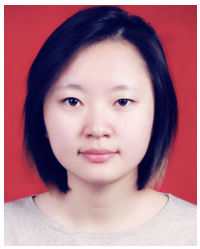
[27] X.-J. Lin, L. Sun, and H. Qu, "Generic construction of public key encryption, identity-based encryption and signcryption with equality test," *Inf. Sci.*, vol. 453, pp. 111–126, Jul. 2018.
[28] A. Karati, S. H. Islam, G. Biswas, M. Z. A. Bhuiyan, P. Vijayakumar, and M. Karuppiah, "Provably secure identity-based signcryption scheme for crowdsourced industrial Internet of Things environments," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2904–2914, Aug. 2018.
[29] H. Zhong, S. Zhang, J. Cui, L. Wei, and L. Liu, "Broadcast encryption scheme for V2I communication in VANETs," *IEEE Trans. Veh. Technol.*, vol. 71, no. 3, pp. 2749–2760, Mar. 2022, doi: 10.1109/TVT.2021.3113660.
[30] L. Chen and J. Malone-Lee, "Improved identity-based signcryption," in *Proc. Int. Workshop Public Key Cryptogr.* Cham, Switzerland: Springer, 2005, pp. 362–379.
[31] S. Agrawal and M. Chase, "FAME: Fast attribute-based message encryption," in *Proc. CCS*, 2017, pp. 665–682.
[32] P. K. Chithaluru, M. S. Khan, M. Kumar, and T. Stephan, "ETH-LEACH: An energy enhanced threshold routing protocol for WSNs," *Int. J. Commun. Syst.*, vol. 34, no. 12, p. e4881, Aug. 2021.
[33] A. Aggarwal, A. Rani, and M. Kumar, "A robust method to authenticate car license plates using segmentation and ROI based approach," *Smart Sustain. Built Environ.*, vol. 9, no. 4, pp. 737–747, Oct. 2019.
[34] M. Kumar, J. Aggarwal, A. Rani, T. Stephan, A. Shankar, and S. Mirjalili, "Secure video communication using firefly optimization and visual cryptography," *Artif. Intell. Rev.*, vol. 55, pp. 1–21, Oct. 2022.
[35] H. Ren, H. Li, D. Liu, G. Xu, N. Cheng, and X. Shen, "Privacy-preserving efficient verifiable deep packet inspection for cloud-assisted middlebox," *IEEE Trans. Cloud Comput.*, vol. 10, no. 2, pp. 1052–1064, Apr. 2022.
[36] C. Dhasarathan, M. Kumar, A. K. Srivastava, F. Al-Turjman, A. Shankar, and M. Kumar, "A bio-inspired privacy-preserving framework for healthcare systems," *J. Supercomput.*, vol. 77, pp. 11099–11134, Mar. 2021.
[37] H. Chen, H. Li, Y. Wang, M. Hao, G. Xu, and T. Zhang, "PriVDT: An efficient two-party cryptographic framework for vertical decision trees," *IEEE Trans. Inf. Forensics Security*, vol. 18, pp. 1006–1021, 2023.
[38] H. Ren, H. Li, D. Liu, G. Xu, and X. S. Shen, "Enabling secure and versatile packet inspection with probable cause privacy for outsourced middlebox," *IEEE Trans. Cloud Comput.*, vol. 10, no. 4, pp. 2580–2594, Oct. 2022.
[39] G. Ateniese, D. Francati, D. Nuñez, and D. Venturi, "Match me if you can: Matchmaking encryption and its applications," *J. Cryptol.*, vol. 34, no. 3, pp. 1–50, Jul. 2021.
[40] J. A. Akinyele et al., "Charm: A framework for rapidly prototyping cryptosystems," *J. Cryptograph. Eng.*, vol. 3, no. 2, pp. 111–128, Jun. 2013.

Jianfei Sun is currently a Research Fellow with the School of Computer Science and Engineering, Nanyang Technological University. His research interests include network security and the IoT security.

Junyi Tao received the B.Eng. degree in software engineering from the University of Electronic Science and Technology of China and the M.Sc. degree in computer science from Stony Brook University. He currently works as a Software Development Engineer at Amazon Web Services. His research interests include applied cryptography, network security, and usable security.



Hao Zhang received the M.S. degree from the School of Information and Software Engineering, University of Electronic Science and Technology of China. He is currently an Engineer with the Science and Technology on Communication Security Laboratory, Chengdu, China. His research interests include asymmetric cryptography, malicious code detection formation security, and the *Journal of Cybersecurity and Privacy*.



Yanan Zhao received the M.S. degree from the University of Electronic Science and Technology of China in 2020. She is currently pursuing the Ph.D. degree with the School of Transportation Science and Engineering, Beihang University. Her research interests include blockchain and the security of the Internet of Vehicles.

Liming Nie received the Ph.D. degree from the Dalian University of Technology in 2017. He subsequently joined Zhejiang Sci-Tech University as a Faculty Member. He is currently a Senior Research Fellow with Nanyang Technological University since 2021. His current research interests include intelligent software development, big code data analysis, and profiling open-source software.

Xiaochun Cheng received the Ph.D. degree in computer science from Jilin University in 1996. He has been Computer Science EU Project Coordinator at Middlesex University since 2012. He is currently with the Department of Computer Science, Swansea University. His research interests include information security.

Tianwei Zhang (Member, IEEE) received the bachelor's degree from Peking University, Beijing, China, in 2011, and the Ph.D. degree from Princeton University in 2017. He is currently an Assistant Professor with the School of Computer Science and Engineering, Nanyang Technological University. He is particularly interested in security threats and defenses in machine learning systems, autonomous systems, computer architecture, and distributed systems. His research interests include computer system security.