sensors

MDPI

Article

SmartVeh: Secure and Efficient Message Access Control and Authentication for Vehicular Cloud Computing

Qinlong Huang [1,*], Yixian Yang [1] and Yuxiang Shi [2]

[1] School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China; yxyang@bupt.edu.cn

[2] School of Science, Beijing University of Posts and Telecommunications, Beijing 100876, China; shiyx@bupt.edu.cn

[*] Correspondence: longsec@bupt.edu.cn; Tel.: +86-10-6228-3366

Received: 2 January 2018; Accepted: 15 February 2018; Published: 24 February 2018

Abstract: With the growing number of vehicles and popularity of various services in vehicular cloud computing (VCC), message exchanging among vehicles under traffic conditions and in emergency situations is one of the most pressing demands, and has attracted significant attention. However, it is an important challenge to authenticate the legitimate sources of broadcast messages and achieve fine-grained message access control. In this work, we propose SmartVeh, a secure and efficient message access control and authentication scheme in VCC. A hierarchical, attribute-based encryption technique is utilized to achieve fine-grained and flexible message sharing, which ensures that vehicles whose persistent or dynamic attributes satisfy the access policies can access the broadcast message with equipped on-board units (OBUs). Message authentication is enforced by integrating an attribute-based signature, which achieves message authentication and maintains the anonymity of the vehicles. In order to reduce the computations of the OBUs in the vehicles, we outsource the heavy computations of encryption, decryption and signing to a cloud server and road-side units. The theoretical analysis and simulation results reveal that our secure and efficient scheme is suitable for VCC.

Keywords: vehicular cloud computing; message access control; attribute-based encryption; message authentication; attribute-based signature

1. Introduction

Vehicular cloud computing (VCC) is an emerging and promising approach to exploit the latest advances in sensing, the Internet of Things, wireless communications, and cloud computing technologies for future transportation [1,2], which may improve road safety and satisfy emerging service demands through message broadcasting. VCC typically consists of road side units (RSUs) and on-board units (OBUs). Particularly, VCC is regarded as an important development that interconnects people, vehicles and information, since numerous services based on vehicle systems may require cooperation among vehicles and RSUs. In order to maximize the overall communication and computation efficiency in VCCs, adaptive resource management has been proposed to provide hard quality of service guarantees in some recent studies [3,4]. That means, with the wireless and sensor network, the driver can enjoy various services in-vehicle based on VCC. The wide application of VCC depends on an efficient mechanism to ensure secure and effective message sharing, which is critical to enable emerging services.

Specifically speaking, let us consider the following practical VCC scenarios [5,6]. Regarding the social aspect, for instance, drivers in vehicles are often glad to share their experiences and traffic information with others who are on the same journey, and may also wish to discuss common interests with friends. Regarding the safety aspect, if there is an emergency (such as a traffic accident or a pavement collapse) on a certain road, the passing drivers may broadcast a warning message to nearby vehicles. If this message can be shared among vehicles in a short time, more serious traffic jams or serious accidents can possibly be prevented. The passing driver may also want to notify a police car and ambulance which is near the affected areas to deal with incidents at the same time. Therefore, it is important to provide efficient access control methods in VCC to guarantee reasonable message access. Unfortunately, adversaries may easily inject false information into the communication network, or even broadcast forged messages to the transportation system; unexpected situations may be caused by these security issues. Hence, message confidentiality, message authentication and access control are the most important problems that affect the VCC services [6]. In order to solve these security issues, traditional encryption mechanisms might be unsuitable.

The attribute-based encryption (ABE) is a cryptographic technique which provides fine-grained access control for encrypted data [7]. In particular, the ciphertext in ciphertext-policy ABE (CP-ABE) scheme can be decrypted only if the attribute set associated with a secret key satisfies the access policy. Hence, a receiver needs to own enough attributes to decrypt the broadcast message [8]. With this technique, both message confidentiality and access control are ensured in VCC. However, applying ABE in VCC has several challenges. Firstly, it brings a heavy key management burden to attribute authority (AA). The attributes of vehicles can be divided into two types in VCC [9], persistent attributes, for which the values remain constant, such as vehicle type and brand, and dynamic attributes, for which the values change frequently, such as road, direction and location. Hence, AA has to renew both the persistent and dynamic attribute keys of the driving vehicle when any dynamic attribute changes to guarantee the decryption capability of vehicles, which brings extra computation and communication overhead. Secondly, ABE introduces heavy computational overhead in data encryption and decryption phases, and this presents a serious challenge for resource-limited OBUs [10].

To ensure the origin of a message, message authentication schemes based on identity based signature (IBS) and attribute-based signature (ABS) in VCC have been studied. However, an IBS scheme would disclose the identification of the signer, which is undesirable. In an ABS scheme, the signer can generate a signature with his attributes issued by AA. Then, from the signature, the recipient vehicle can verify the signature by checking that the sender's attributes satisfy the complex predicate policy without exposing the identity of the sender [11]. However, ABS also brings high computation costs, which cannot be adopted by OBUs directly.

In summary, it is important to maintain secure and reliable message broadcasting with low computation in VCC. In this work, we propose a secure and efficient message access control and authentication scheme for VCC, called SmartVeh, which features the following achievements.

(1) We provide a secure message access control framework in VCC based on hierarchical ABE (HABE). The framework consists of a trusted authority (TA), and a group of AAs which request secret parameters from the TA and generate persistent attribute keys or dynamic attribute keys for vehicles independently. Thus, vehicles can share confidential messages with other vehicles which satisfy the pre-defined access policy.

(2) We utilize ABS to enforce message authentication, which can authenticate messages by verifying whether the signer's attributes satisfy the predicate policy. It also ensures message integrity by checking and maintaining the anonymity of vehicles.

(3) We present a secure outsourcing construction in VCC by delegating the heavy computations from resource-limited OBUs to the cloud server and RSUs, which means that the computation complexity of OBUs is independent of the number of attributes.

The remainder of this paper is organized as follows. The related work is overviewed in Section 2, and technical preliminaries are provided in Section 3. The system framework, security model and system definition are provided in Section 4, and our construction of the proposed scheme is elaborated in Section 5. The security and performance analyses are described in Sections 6 and 7. The conclusions are given in Section 8.

2. Related Works

Over recent years, eavesdropping on messages, tampering with messages and forging warning messages by malicious attackers are security threats in VCC, and many related works have been proposed that have concentrated on confidentiality, access control, authentication, etc.

Pietrowicz et al. [12] adopted identity based encryption (IBE) algorithms to effectively address the challenges in providing secure communications in vehicle networks. Mallissery et al. [13] adopted the RSU geolocation key to encrypt the exchanged messages in a vehicular ad-hoc network (VANET), which provides location confidentiality against vehicles outside the zone. The weakness is that this scheme limits the scope of message sharing only to one RSU. Nema et al. [14] proposed an RSA-algorithm-based encryption and decryption approach to provide message confidentiality in VANETs. However, all of the above schemes do not consider the fine-grained access control of the transmitted message.

ABE, introduced by Sahai and Waters, is cryptographic technique to implement fine-grained access control for encrypted messages [15,16]. In fact, ABE can be adopted in many applications to realize message confidentiality and access control in vehicular communication [17–20]. Huang et al. [17] proposed a security policy enforcement scheme to achieve secure message dissemination, which is the first one to introduce CP-ABE in VANET. The main drawback of this scheme is that the vehicles under different secure groups of RSUs cannot share messages with each other directly, which was improved in [18]. For emergency services, Yeh et al. [19] proposed an access control scheme in VANETs to send messages to nearby rescue vehicles securely with ABE. Xia et al. [9] divided the attributes of vehicles into two types, dynamic attributes and persistent attributes. Dynamic attribute values would change frequently, while persistent attributes such as police car and sprinkler would never change. This brings new challenges with respect to the heavy key management of AA, since it must re-generate secret keys for both persistent attributes and dynamic attributes when any dynamic attribute changes. To solve the issue of heavy key management by adopting ABE in VCC, Liu et al. [20] extended the CP-ABE algorithm with hierarchical authorities, which can reduce the key management of a single center authority. Nevertheless, none of the above ABE-based schemes can provide mechanisms to authenticate vehicles before handling the messages.

Message authentication of vehicles, which determines that a message is from a valid source, is another important security issue in vehicular communication networks. In consideration of the identity privacy of vehicles, the traditional IBS method is no longer applicable [21]. Sánchez-Garcíaby et al. [22] proposed an electronic identity (eID) based secure authentication scheme in VANETs, which can protect drivers' real identities. The vehicle broadcasts a message containing the certificate signed by eID to prove its identity when receiving the authentication request. Kang et al. [23] integrated pseudonyms with IBS in vehicular communication, which could not only authenticate the messages, but also protect the privacy of the message sender. Chim et al. [24] adopted anonymous credentials to guarantee the identity of driver to be unlinkable to any party. However, in these two anonymous schemes, the vehicle must preset a large number of anonymous keys in order to randomly choose one to sign messages, and the authority or RSU must hold the anonymous certificates of all the vehicles in order to authenticate vehicles, which creates a heavy overhead for key management. Instead of suffering from extra overhead, as in previous anonymous identity-based schemes, ABS is introduced in VCC to ensure anonymous authentication. In order to achieve message verification and maintain anonymity, Liu et al. [20] utilized ABS to enforce message authentication.

However, most existing ABE and ABS schemes introduce heavy computation overheads in the encryption, decryption and signing phases, and these computation costs grow linearly [25,26]. Therefore, OBUs that have limited resources may encounter serious challenges during these processes [27]. To reduce the computational burden of the OBUs of vehicles, Xia et al. [9] introduced an outsourced decryption construction for ABE in VCC, but this scheme requires each RSU to restore secret keys for all vehicles and ignores the high encryption cost of ABE. Liu et al. [28] proposed a secure message dissemination construction for vehicle networks, in which the local decryption computation cost can be outsourced to nearest RSU, but this scheme ignores the computation cost of message encryption with ABE. Ma et al. [29] proposed two CP-ABE based mechanisms for achieving both outsourced encryption and outsourced decryption. However, this scheme is not practical in VCC.

3. Technical Preliminaries

3.1. Bilinear Map

Let $\mathbb{G}_0$ and $\mathbb{G}_T$ be two multiplicative groups with the same prime order p. A map $e : \mathbb{G}_0 \times \mathbb{G}_0 \to \mathbb{G}_T$ with the following properties is said to be bilinear:

(1) Computability. There is a polynomial time algorithm to compute $e(g, h) \in \mathbb{G}_T$ for any $g, h \in \mathbb{G}_0$.
(2) Bilinearity. For all $g, h \in \mathbb{G}_0$ and $a, b \in \mathbb{Z}_p$, we have $e(g^a, h^b) = e(g, h)^{ab}$.
(3) Non-degeneracy. There exists $g, h \in \mathbb{G}_0$ such that $e(g, h) \neq 1$.

3.2. Access Tree

Let T be a tree representing an access policy. Each non-leaf node x of the tree represents a threshold gate. Let num_x denote the number of children of a node x, and k_x represent its threshold value, then $1 \leq k_x \leq num_x$. For each leaf node x of the tree, we have k_x = 1, and denote $attr_x$ as an attribute associated with it. For a non-leaf node x, the child nodes of x are numbered from 1 to num_x. The function $parent(x)$ represents a parent node of the node x, $index(x)$ returns the index value of node x.

We let T_x be the sub-tree rooted at node x in T. We denote the result as $T_x(r) = 1$ if the attribute set r satisfies the access tree T_x. Then the value of $T_x(r)$ is computed in the following. If x is a leaf node and $attr_x \in r$, $T_x(r)$ returns 1. If x is a non-leaf node, we compute $T_n(r)$ for all children n of node x. If at least k_x children return 1, $T_x(r)$ returns 1.

3.3. Ciphertext-Policy, Attribute-Based Encryption

In a typical CP-ABE system, the access policy is expressed as a tree over a set of attributes. The CP-ABE scheme is composed of the following four algorithms.

(1) *Setup*(1^λ): On input of a security parameter λ, the algorithm outputs a public key PK and a master key MK.
(2) *KeyGen*(MK, PK, S): On input of the master key MK, public key PK and a set S of attributes, the algorithm outputs a secret key SK.
(3) *Enc*(PK, M, T_a): On input of the public key PK, a message M and an access policy T_a, the algorithm outputs a ciphertext CT.
(4) *Dec*(PK, CT, SK): On input of the public key PK, a ciphertext CT associated with an access policy T_a and a secret key SK, the algorithm outputs the message M if $S \in T_a$.

3.4. Attribute-Based Signature

An ABS scheme that provides anonymous message authentication generally consists of the following four algorithms.

(1) *Setup*(1^λ). On input of a security parameter λ, AA generates the public key PK and master key MK.
(2) *KeyGen*(MK, PK, S). On input of the master key MK, public key PK, and a set of attributes S, AA generates the secret key SK.
(3) *Sign*(PK, SK, M, T_c). On input of the public key PK, a secret key SK of signer, a message M and a predicate policy T_c, the signer generates a signature ST for M.

(4) *Verify*(PK, M, T_c, ST). On input of the public key PK, a message M, a predicate policy T_c and a signature ST, the verifier checks ST. If the signer's attributes satisfy T_c, it outputs true.

4. System Overview

4.1. System Framework

The system framework of SmartVeh consists of the following parties: TA, AA, cloud server, RSUs and vehicles, as shown in Figure 1. The TA is viewed as a fully trusted party that takes charge of managing AAs and generating system parameters and secret parameter to AAs. The AAs are also trusted and independent of each other. According to the different types of attributes managed by the AA, persistent AA is responsible for generating the persistent attributes of vehicles, and dynamic AA is responsible for generating the dynamic attributes of vehicles. A semi-trusted cloud server which has powerful computation and storage capabilities is intended to perform the outsourced encryption and signing computations. The RSUs are interconnected through wired lines, and provide wireless connections to vehicles. We assume that there are the dense of RSUs deployed near the road in the city, and the RSUs are responsible for performing access control with vehicles, and authenticating the origin of messages by verifying the signature of vehicles. If the signature verification is passed, RSUs would partially decrypt the encrypted messages, and then broadcast them to vehicles. The vehicles with OBUs and powerful sensors are a set of nodes that are moving on the road, and communicate with each other through RSUs. When a vehicle communicates with others, it encrypts the message with an access policy and signs message with its attributes before broadcasting to others, and intended receivers can decrypt the ciphertext with their attributes.

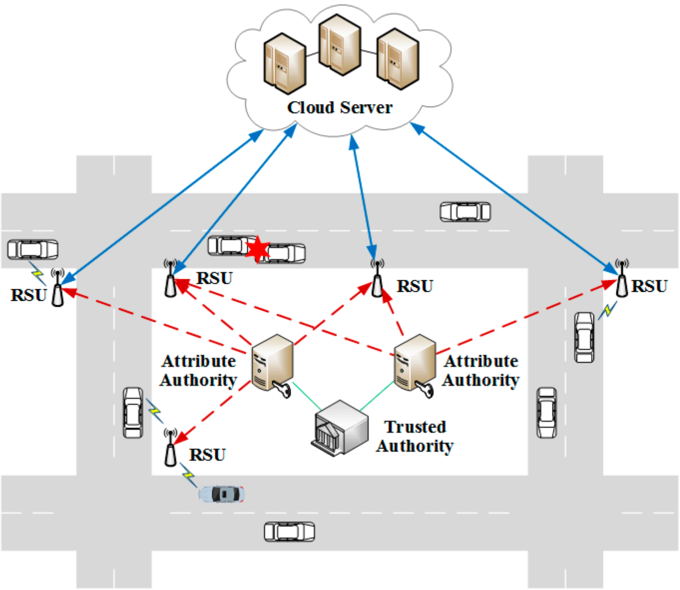

Figure 1. System framework of SmartVeh.

4.2. Security Model

In this work, we consider TA and AA to be trusted, while the cloud server and RSUs are honest but curious. It means they may learn sensitive information from the broadcast message. Specifically, the security requirements are defined as follows:

(1) Message confidentiality. The messages should be transmitted in encrypted form, and the vehicles which cannot satisfy the access policy defined by the message sender should not be allowed to

access the plaintext of the message. Meanwhile, the cloud server and RSUs cannot recover the broadcast message.

(2) Fine-grained access control. The vehicle can enforce an access policy for each broadcast message, which designates the messages that the vehicle is allowed to access.

(3) Message authentication. If message sender's attributes could not satisfy the predicate policy, the message broadcast should not succeed.

(4) Collusion resistance. The message access should not be successful if either of the vehicles cannot satisfy the access policy alone. Further, even if unauthorized vehicles collude with the RSU, the access should not take effect.

4.3. System Definition

According to the SmartVeh framework, our scheme consists of these ten algorithms.

(1) $Setup(1^{\lambda})$: On input of a security parameter λ, the TA outputs a system public key PK and a master key MK.

(2) $CreateAA(PK, MK, \mathbb{A})$: On input of PK and MK, a set of attributes $\mathbb{A}$ of AA, the TA outputs the master secret key MSK for AA.

(3) $KeyGen(PK, MSK, S_i)$: On input of PK and MSK, a set of managed attributes S_i of the vehicle, the AA outputs the secret key SK_i for each vehicle.

(4) $Cloud.Encrypt(PK, \{T_a^{(i)}\}_{i=1}^{2})$: On input of PK, access policies $\{T_a^{(i)}\}_{i=1}^{2}$ in different AAs, the cloud server outputs a partially encrypted ciphertext CT'.

(5) $Vehicle.Encrypt(PK, M, CT')$: On input of PK, a message M and a partial ciphertext CT', the vehicle outputs a ciphertext CT.

(6) $Cloud.Sign(CT, T_c, SK_i')$: On input of a ciphertext CT, a predicate policy T_c and an outsourced secret key SK_i' which is a part of secret key, the cloud server outputs a signing token SN and a partial signature ST'.

(7) $Vehicle.Sign(ST', SK)$: Given a partial signature ST' and secret key SK, the vehicle generates a thorough signature ST.

(8) $Verify(ST, SN, T_c)$: On input of a signature ST, a signing token SN and a predicate policy T_c, the RSU outputs true if the sender vehicle's attributes satisfy T_c.

(9) $RSU.Decrypt(PK, SK_i'', CT)$: On input of PK, a ciphertext CT, a outsourced secret key SK_i'' which is also a part of secret key, the RSU outputs a partially decrypted ciphertext CT_p if the attribute set satisfies the access policy.

(10) $Vehicle.Decrypt(CT_p, SK_i)$: The vehicle takes a CT_p and a secret key SK_i as input, and outputs the plaintext M.

5. Construction of SmartVeh

In order to achieve secure message broadcasting, we provided an access control framework for encrypted messages in VCC by employing a delegation mechanism based on HABE, and utilized ABS to enforce message authentication, which can authenticate messages by verifying that the sender's attributes satisfy T_c in the ciphertext.

5.1. System Setup

The TA first runs the $Setup$ algorithm to choose two multiplicative groups with prime order p, that are $\mathbb{G}_0$ and $\mathbb{G}_T$, and a bilinear map $e : \mathbb{G}_0 \times \mathbb{G}_0 \rightarrow \mathbb{G}_T$. Then, the TA randomly chooses $g, h \in \mathbb{G}_0$ and $\alpha, \beta \in \mathbb{Z}_p$, and chooses cryptographic hash functions $H_1 : \{0,1\}* \rightarrow \mathbb{Z}_p^*$, $H_2 : \{0,1\}* \rightarrow \mathbb{G}_0$. Finally, the TA outputs a system public key $PK = (g, g^{\alpha}, g^{\beta}, h, h^{\beta}, e(g,g)^{\alpha\beta})$ and a master key $MK = (\alpha, \beta)$.

5.2. Authority Setup

Our scheme divides the attributes of vehicle into two types, persistent attributes and dynamic attributes, which are managed by different AAs independently. The TA runs the *CreateAA* algorithm to select a random but unique value $\nu_i \in \mathbb{Z}_p$ for AA$_i$. For the attribute set $\mathbb{A}$ managed by AA$_i$, the TA chooses random $r_{i,j}$ for each attribute in it. Then the TA computes the master secret key for AA$_i$ as

$$MSK_i = (D'_i = g^{(\alpha+\nu_i)\beta}, D'_{i,1} = g^{\nu_i}, \{\overline{D}_{i,j} = g^{\nu_i\beta} H_1(j)^{r_{i,j}}, \overline{D}'_{i,j} = g^{r_{i,j}}\}_{j\in\mathbb{A}}) \tag{1}$$

5.3. Key Generation

For each vehicle, the AA$_i$ runs the *KeyGen* algorithm to choose a unique secret $\gamma_i \in \mathbb{Z}_p$ and a random $\varepsilon_i \in \mathbb{Z}_p$. For each attribute j in the attribute set S_i of vehicle in AA$_i$, the AA$_i$ chooses a random $u_{i,j} \in \mathbb{Z}_p$. Finally, AA$_i$ outputs the key as

$$AK_i = (\{\widetilde{D}_{i,j} = \overline{D}_{i,j} \cdot g^{\gamma_i\beta} H_1(j)^{u_{i,j}} = g^{(\nu_i+\gamma_i)\beta} H_1(j)^{r_{i,j}+u_{i,j}}, \overline{D}'_{i,j} = \overline{D}'_{i,j} \cdot g^{u_{i,j}} = g^{r_{i,j}+u_{i,j}}\}_{j\in S}) \tag{2}$$

Thus the vehicle's secret key in AA$_i$ is:

$$SK_i = (D_i = D'_i \cdot g^{\gamma_i\beta} = g^{(\alpha+\nu_i+\gamma_i)\beta}, D_{i,1} = D'_{i,1} \cdot g^{\gamma_i} h^{\varepsilon_i} = g^{\nu_i+\gamma_i} h^{\varepsilon_i}, D_{i,2} = g^{\varepsilon_i}, AK_i) \tag{3}$$

For example, an ambulance can get secret keys for vehicle type from the AA$_1$ for persistent attributes, and get secret keys for road and direction from the AA$_2$ for dynamic attributes.

5.4. Message Broadcasting

Before broadcasting the message to the RSUs, the vehicle first selects a symmetric key $DK \in \mathbb{Z}_p$ randomly. Then the vehicle encrypts M by utilizing a symmetric encryption algorithm, and the result is outputted as $C = SE_{DK}(M)$. Then the vehicle defines a collection of access policies $\{T_a^{(i)}\}_{i=1}^{2}$, where $T_a^{(i)}$ is the access tree in AA$_i$, such as "police car OR ambulance", "(normal road AND east) AND (eall road AND north)".

5.4.1. Cloud Encryption

The cloud server runs the *Cloud.Encrypt* algorithm to execute outsourcing encryption. First, the cloud server chooses a polynomial p_x for each node x in $T_a^{(i)}$. The polynomials are selected in a top-down manner. For each node x in $T_a^{(i)}$, the cloud server sets the degree d_x of p_x to be $k_x - 1$.

The algorithm selects a random $s_i \in \mathbb{Z}_p$ and sets $p_R(0) = s_i$ for the root node R. Then the algorithm chooses d_R other points of p_R randomly to complete the definition. For the other node x, the algorithm sets $p_x(0) = p_{parent(x)}(index(x))$ and chooses d_x other points randomly to complete the definition. In $T_a^{(i)}$, let Y_i be the set of leaf nodes. Then, the cloud server returns the result as

$$CT_i = (T_a^{(i)}, \{\widetilde{C}_{i,y} = g^{p_y(0)}, \widetilde{C}'_{i,y} = H_1(attr_y)^{p_y(0)}\}_{y\in Y_i}) \tag{4}$$

Finally, the cloud server outputs a partial ciphertext CT' as

$$CT' = (\left\{C'_{i,3} = g^{\beta s_i}, C'_{i,4} = h^{\beta s_i}, CT_i\right\}_{i\in\{1,2\}}) \tag{5}$$

5.4.2. Vehicle Encryption

With the partial ciphertext CT', the vehicle runs the *Vehicle.Encrypt* algorithm to randomly choose $t \in \mathbb{Z}_p$, compute $C_1 = DK \cdot e(g,g)^{\alpha\beta t}$ and $C_2 = g^t$. Then, the vehicle computes $C_{i,3} = C'_{i,3} \cdot g^{\beta t}, C_{i,4} = C'_{i,4} \cdot h^{\beta t}$ and outputs the ciphertext CT as

$$CT = (C = SE_{DK}(M), C_1 = DK \cdot e(g,g)^{\alpha\beta t}, C_2 = g^t, \{C_{i,3} = g^{\beta(s_i+t)}, C_{i,4} = h^{\beta(s_i+t)}, CT_i\}_{i\in\{1,2\}}) \quad (6)$$

5.4.3. Cloud Signing

The encrypted messages must be authenticated, since the messages may be forged by attackers. Then the vehicle computes $S_0 = H_2(CT)$, and sends the ciphertext CT, a predicate policy T_c, such as "(middle road AND east) AND location of accident", an outsourced secret key $SK'_k = \{AK_k\}$ corresponding to attribute set S_k in AA to the cloud server through RSUs. The cloud server runs the $Cloud.Sign$ algorithm to execute computation outsourcing. For each node x of predicate policy T_c, the cloud server chooses polynomial q_x in a top-down manner, and sets the degree d'_x of q_x to be $k'_x - 1$.

Starting from R, the algorithm first selects a random $r \in \mathbb{Z}_p$ and sets $q_R(0) = r$. Then, the algorithm randomly chooses d'_R other points of q_R to complete the definition. For the other node x, it sets $q_x(0) = q_{parent(x)}(index(x))$ and then selects d'_x other points randomly to define q_x completely.

In T_c, let Z be the set of leaf nodes. Then, the cloud server outputs the signing token SN as

$$SN = \{\widetilde{K}_z = g^{q_z(0)}, \widetilde{K}'_z = H_1(attr_z)^{q_z(0)}\}_{z\in Z} \quad (7)$$

The cloud server randomly chooses $t_j \in \mathbb{Z}_p$ for each node $j \in Z$, and computes with SK'_k as follows.

(1) If $j \in S_k \cap Z$, the cloud server computes $\widetilde{S}_j = (\widetilde{D}_{k,j} \cdot H_1(j)^{t_j})^{1/r} = g^{(v_k+\gamma_k)\beta/r} \cdot H_1(j)^{(r_{k,j}+u_{k,j}+t_j)/r}$, and $\widetilde{S}'_j = (\widetilde{D}'_{k,j} \cdot g^{t_j})^{1/r} = g^{(r_{k,j}+u_{k,j}+t_j)/r}$.

(2) If $j \in Z/S_k \cap Z$, the cloud server computes $\widetilde{S}_j = (H_1(j)^{t_j})^{1/r} = H_1(j)^{t_j/r}$, and $\widetilde{S}'_j = (g^{t_j})^{1/r} = g^{t_j/r}$.

Finally, the cloud server randomly selects $\lambda \in \mathbb{Z}_p$ and outputs the partial signature ST' as

$$ST' = (S'_1 = H_2(CT)^{\lambda}, S'_2 = g^{\lambda}, S_3 = \{\widetilde{S}_j, \widetilde{S}'_j\}_{j\in Z}) \quad (8)$$

5.4.4. Vehicle Signing

With the partial signature generated by the cloud server, the vehicle first runs the *Vehicle.Sign* algorithm to randomly choose $\mu \in \mathbb{Z}_p$ and compute $S_1 = S'_1 \cdot (S_0)^{\mu} \cdot D_k$ and $S_2 = S'_2 \cdot g^{\mu}$. At last, the vehicle generates the encrypted message's signature ST as

$$ST = (S_1 = H_2(CT)^{\lambda+\mu} \cdot g^{(\alpha+v_k+\gamma_k)\beta}, S_2 = g^{\lambda+\mu}, S_3) \quad (9)$$

The vehicle sends the signature ST with encrypted message to the connected RSUs, and the message will be broadcasted to other vehicles.

5.5. Message Decryption

When receiving the encrypted and signed message, the recipient RSU runs the $Verify$ algorithm to verify that the message is from an authorized source.

5.5.1. RSU Verifying

The RSU runs the *VerNode* algorithm, which takes as input ST, SN and a node x of T_c.

(1) If x is a leaf node, then we set $w = attr_x$. If $w \in S \cap Z$, then

$$VerNode(ST, SN, x) = \frac{e(\widetilde{S}_w, \widetilde{K}_x)}{e(\widetilde{S}'_w, \widetilde{K}'_x)} = \frac{e(g^{(\nu_k+\gamma_k)\beta/r} H_1(z)^{(r_{k,w}+u_{k,w}+t_w)/r}, g^{q_x(0)})}{e(g^{(r_{k,w}+u_{k,w}+t_w)/r}, H_1(attr_x)^{q_x(0)})} = e(g,g)^{(\nu_k+\gamma_k)\beta/r \cdot q_x(0)} \tag{10}$$

If $w \in Z/S \cap Z$, then

$$VerNode(ST, SN, x) = \frac{e(\widetilde{S}_w, \widetilde{K}_x)}{e(\widetilde{S}'_w, \widetilde{K}'_x)} = \frac{e(H_1(w)^{t_w/r}, g^{q_x(0)})}{e(g^{t_w/r}, H_1(attr_x)^{q_x(0)})} = 1 \tag{11}$$

(2) If x is a non-leaf node, the algorithm $VerNode(ST, SN, x)$ computes as follows. It calls the $VerNode(ST, SN, n)$ algorithm for each child node n of x, and outputs the result as I_n.

We denote S_x as an arbitrary k_x-sized set of child nodes n such that $I_n \neq \perp$. If no such set exists, it returns $\perp$. Otherwise, the algorithm computes the I_x.

$$\begin{aligned} I_x &= \prod_{n \in S_x} {I_n}^{\Delta_{j,S'_x}(0)} = \prod_{n \in S_x} \left(e(g,g)^{(\nu_k+\gamma_k)\beta/r \cdot q_{parent(n)}(index(n))}\right)^{\Delta_{j,S'_x}(0)} = \prod_{n \in S_x} e(g,g)^{(\nu_k+\gamma_k)\beta/r \cdot q_x(j) \cdot \Delta_{j,S'_x}(0)} \\ &= e(g,g)^{(\nu_k+\gamma_k)\beta/r \cdot q_x(0)} \end{aligned} \tag{12}$$

where $j = index(n)$ and $S'_x = \{index(n) : n \in S_x\}$. Then, we can define the evaluation result for predicate tree T_c as I, if T_c is satisfied.

$$I = VerNode(ST, SN, R) = e(g,g)^{(\nu_k+\gamma_k)\beta/r \cdot q_R(0)} = e(g,g)^{(\nu_k+\gamma_k)\beta/r \cdot r} = e(g,g)^{(\nu_k+\gamma_k)\beta} \tag{13}$$

Finally, the RSU checks whether the equation holds.

$$\frac{e(g, S_1)}{e(H_2(CT), S_2) \cdot I} = \frac{e(g, H_2(CT)^{\lambda+\mu} \cdot g^{(\alpha+\nu_k+\gamma_k)\beta})}{e(H_2(CT), g^{\lambda+\mu}) \cdot e(g,g)^{(\nu_k+\gamma_k)\beta}} = e(g,g)^{\alpha\beta} \tag{14}$$

If the equation holds, then RSU accepts ST and partially decrypts the encrypted message for vehicles that satisfy the access policy.

5.5.2. RSU Decryption

With part of the secret key $SK''_k = (D_{k,1}, D_{k,2}, AK_k)$ from the vehicle corresponding to attribute set S_k, the RSU runs the $RSU.Decrypt$ algorithm to decrypt the CT. In order to evaluate whether the vehicle's attributes satisfy $T_a^{(k)}$ or not, the RSU runs the *DecNode* algorithm, which takes as input CT_k, SK''_k, and a node x from $T_a^{(k)}$.

(1) If x is a leaf node, then we let $w = attr_x$ and compute the following. If $w \in S_k$, then

$$DecNode(CT_k, SK''_k, x) = \frac{e(\widetilde{D}_{k,w}, \widetilde{C}_{k,x})}{e(\widetilde{D}'_{k,w}, \widetilde{C}'_{k,x})} = \frac{e(g^{(\nu_k+\gamma_k)\beta} H_1(w)^{r_{k,w}+u_{k,w}}, g^{p_x(0)})}{e(g^{r_{k,w}+u_{k,w}}, H_1(attr_x)^{p_x(0)})} = e(g,g)^{(\nu_k+\gamma_k)\beta p_x(0)} \tag{15}$$

If $z \notin S_k$, then $DecNode(CT_k, SK''_k, x) = \perp$.

(2) If x is a non-leaf node, the algorithm $DecNode(CT_k, SK''_k, x)$ computes the following. It calls $DecNode(CT_k, SK''_k, n)$ for each child node n of x, and generates the result as $F_{k,n}$. Let S_x be an arbitrary k_x-sized set of child nodes n such that $F_{k,n} \neq \perp$. Similar to the verifying process, the algorithm computes as follows.

$$\begin{aligned} F_{k,x} &= \prod_{n \in S_x} {F_{k,n}}^{\Delta_{j,S'_x}(0)} = \prod_{n \in S_x} \left(e(g,g)^{(\nu_k+\gamma_k)\beta \cdot p_{parent(n)}(index(n))}\right)^{\Delta_{j,S'_x}(0)} = \prod_{n \in S_x} e(g,g)^{(\nu_k+\gamma_k)\beta \cdot p_x(j) \cdot \Delta_{j,S'_x}(0)} \\ &= e(g,g)^{(\nu_k+\gamma_k)\beta \cdot p_x(0)} \end{aligned} \tag{16}$$

If the receiver owns enough attributes to satisfy $T_a^{(k)}$, we set the evaluation result as F_k.

$$F_k = DecNode(CT_k, SK_k'', R) = e(g,g)^{(\nu_k+\gamma_k)\beta p_R(0)} = e(g,g)^{(\nu_k+\gamma_k)\beta s_k} \tag{17}$$

RSU computes

$$B_k = \frac{e(D_{k,1}, C_{k,3})}{e(D_{k,2}, C_{k,4})} = \frac{e(g^{\nu_k+\gamma_k}h^{\varepsilon_k}, g^{\beta(s_k+t)})}{e(g^{\varepsilon_k}, h^{\beta(s_k+t)})} = e(g,g)^{(\nu_k+\gamma_k)\beta(s_k+t)} \tag{18}$$

and

$$A_k = B_k/F_k = e(g,g)^{(\nu_k+\gamma_k)\beta(s_k+t)}/e(g,g)^{(\nu_k+\gamma_k)\beta s_k} = e(g,g)^{(\nu_k+\gamma_k)\beta t} \tag{19}$$

Hence, if the vehicle's attributes satisfy $T_a^{(k)}$, the RSU sends the result $CT_p = (C, C_1, C_2, A_k)$ to the vehicle.

5.5.3. Vehicle Decryption

After receiving the result from the RSU, the vehicle runs the *Vehicle.Decrypt* algorithm to recover *DK* with its own secret key.

$$DK = \frac{C_1 \cdot A_k}{e(C_2, D_k)} = \frac{DK \cdot e(g,g)^{\alpha\beta t} \cdot e(g,g)^{(\nu_k+\gamma_k)\beta t}}{e(g^t, g^{(\alpha+\nu_k+\gamma_k)\beta})} = \frac{DK \cdot e(g,g)^{\alpha\beta t}}{e(g^t, g^{\alpha\beta})} \tag{20}$$

Finally, the vehicle can recover the message M with DK based on the symmetric decryption algorithm, while the unauthorized vehicles are prevented from accessing it.

6. Security Analysis

The construction of SmartVeh is based on CP-ABE [25] and ABS [26], which have been proved secure, thus our scheme has the same security property as these. Then we discuss the security properties of SmartVeh, which not only provides message confidentiality, but also guarantees fine-grained access control, efficient message authentication and collusion resistance.

6.1. Message Confidentiality

The broadcast message in our scheme is first encrypted with a symmetric encryption technique. Then the *DK* is encapsulated by access policy. Hence, message confidentiality against outside vehicles which do not have enough attributes can be guaranteed. In the message broadcasting phase, the cloud server executes most of encryption computations for the vehicle. However, the cloud server cannot access the plaintext of message without the secret key. Moreover, if the attribute set of the vehicle cannot satisfy the T_a in the ciphertext, the value $A_k = e(g,g)^{(\nu_k+\gamma_k)\beta t}$ cannot be computed by the RSUs to get *DK* in the message decryption phase. Therefore, only vehicles that satisfy T_a can decrypt the encrypted message, and message confidentiality against a semi-trusted cloud server and RSUs is also guaranteed.

6.2. Fine-Grained Access Control

Our work used the CP-ABE mechanism to protect *DK*, and ensure flexibility by specifying the access policies of vehicles. In the message encryption phase, the sender is able to protect the symmetric key with an expressive access policy, and broadcast the encrypted message through RSUs. Specifically, the access policy in the ciphertext can be represented by flexible access tree. In this way, our scheme can dramatically increase the flexibility and represent any desired access conditions.

6.3. Message Authentication

In our work, the ABS technique was adopted to achieve message authentication with privacy preservation. The adversary, such as a malicious vehicle, may want to forge a signature with an unsatisfied predicate policy, so that fake messages have a reliable source. However, as proved in [26], our work is secure under the computational Diffie-Hellman assumption, since the adversary cannot forge a valid ST with a non-negligible probability.

6.4. Collusion Resistance

Malicious vehicles may collude to combine their secret keys to decrypt a ciphertext that each of them cannot access individually. However, the secret key outputted by AA in our scheme is generated with random γ_i, which is unique for each vehicle. Thus, even if two or more vehicles combine their attributes to satisfy the access policies, the value $F_k = e(g,g)^{(v_k+\gamma_k)\beta s_k}$ cannot be computed. Moreover, even if malicious vehicles collude with RSUs to decrypt the encrypted message, the collusion will not succeed.

7. Performance Analysis

7.1. Functionality Comparisons

In this part, we will analyze the performance of several ABE-based message sharing schemes. The results are shown in Table 1. The functionality comparison of our scheme with these schemes in VCC is in terms of message confidentiality, hierarchical authorities, persistent attribute key generation, anonymous authentication and computation outsourcing.

Table 1. Attribute-based message sharing schemes in vehicular cloud computing.

Functions	Yeh et al. [19]	Liu et al. [28]	Chim et al. [24]	Xia et al. [9]	Liu et al. [20]	Our Scheme
Message confidentiality	CP-ABE	CP-ABE	CP-ABE	CP-ABE	HABE	HABE
Hierarchical authorities	No	No	No	No	Yes	Yes
Persistent attribute key generation	-	-	-	Every	Once	Once
Anonymous authentication	No	No	IBS with pseudonym	No	ABS	ABS
Encryption outsourcing	No	No	No	No	No	Yes
Decryption outsourcing	No	Yes	No	Yes	No	Yes
Signing outsourcing	-	-	No	-	No	Yes

First, the compared schemes all adopt the ABE technique to grant fine-grained access control for vehicular messages. Moreover, only Xia et al. [9], Liu et al. [20] and our scheme clearly define the attributes of vehicles that include persistent attributes and dynamic attributes. However, a persistent attribute key is generated only once in Liu et al. [20] and our scheme, while in Xia et al. [9] it needs to be generated when the vehicles move into another RSU. Further, we can see that in our scheme, Xia et al. [9] and Liu et al. [28] achieve decryption outsourcing, which incur less computation costs for message decryption for resource-limited OBUs in vehicles. This is because the RSU helps the OBU to decrypt the ciphertext. However, the origin of the message is not authenticated in Xia et al. [9] and Liu et al. [28], which may bring security concerns, such as forged messages and man-in-the-middle attacks. Chim et al. [24] and Liu et al. [20] adopt IBS with pseudonym and ABS, respectively, to achieve anonymous authentication, but the pseudonym method creates large extra storage overheads and the standard ABS method would bring large computation costs.

Compared to these schemes, our scheme first introduces HABE to reduce the overhead for key management on a single TA by dividing dynamic and persistent attributes managed by different AAs, which also resolves the problem of single point failure to a certain extent, and the complexity of operations of AAs in the key generation phase is independent of the number of vehicles, which means that our scheme is scalable enough to handle a case where the number of authorized vehicles increases dynamically. Further, our scheme proposes an outsourced architecture to satisfy the lightweight demand of resource-limited OBUs in VCC.

7.2. Performance Analysis

We discuss the efficiency of our scheme in terms of message encryption, decryption and signing, and compare the results with Liu et al. [28], Xia et al. [9] and Liu et al. [20], which are related schemes in a vehicular network. Table 2 shows the comparison results. Let T_r, T_0, T_t, N_c, N_u and N_d denote the computation cost of the pairing operation, the computation cost of the exponentiation operation in $\mathbb{G}_0$, the computation cost of the exponentiation operation in $\mathbb{G}_T$, the number of attributes in the ciphertext, the total number of attributes of the vehicle, and the number of dynamic attributes, respectively. The symmetric encryption and decryption, hash and simple multiplication operations are ignored.

Table 2. Computation cost.

Schemes	Key Generation (AA)	Message Encryption (OBU)	Message Decryption (OBU)	Message Signing (OBU)
Liu et al. [28]	$(3+N_u)T_0$	$(3N_c+1)T_0+T_t$	T_r	-
Xia et al. [9]	$(3+N_u)T_0$	$(3N_c+1)T_0+T_t$	T_r	-
Liu et al. [20]	$(2+4N_d)T_0$	$(2N_c+1)T_0+T_t$	$(2N_c+1)T_p+N_cT_t$	$3N_uT_0+2T_t$
Our scheme	$(4+2N_d)T_0$	$3T_0+T_t$	T_r	$2T_0$

First, we analyzed the computation cost in the key generation phase. As vehicles are moved through different RSUs dynamically along with time, the secret keys should be generated for vehicles by TA. Xia et al. [9] and Liu et al. [28] both need to perform $(3+N_u)T_0$ to generate all secret keys for vehicles. Our scheme and Liu et al. [20] both divide attributes into two types, namely persistent attributes and dynamic attributes. The AA only needs to generate secret keys according to dynamic attributes for vehicles since the value of persistent attributes are not changed. From the table, we can notice that the computation cost of our scheme in this phase is less than that in Liu et al. [20] which needs to generate extra signing keys at the same time.

Second, we discuss the overhead of encryption and decryption of the message. Since Liu et al. [28], Xia et al. [9] and Liu et al. [20] all execute the complex ABE algorithm, the encryption computation costs on the vehicle side of these schemes are $(3N_c+1)T_0+T_t$, $(3N_c+1)T_0+T_t$ and $(2N_c+1)T_0+T_t$, respectively, which increase with N_c. Conversely, the result stay constant in our scheme. For the message decryption phase, the vehicles use secret keys to decrypt the encrypted message recursively in Liu et al. [20], and the computation cost is $(2N_c+1)T_r+N_cT_t$. In Liu et al. [28], Xia et al. [9] and our scheme, most of decryption computations are outsourced to nearby RSUs, and the OBUs in vehicles only need one pairing operation to decrypt the partially decrypted message.

In order to analyze the time cost of signing the message, we compared our scheme with Liu et al. [20], which achieves anonymous authentication based on ABS as well, and needs to perform $3N_uT_0+2T_t$ in signing the algorithm, while in our scheme, the cloud server is able to partially sign the ciphertext with a predicate policy and outsourced secret key, which are both sent by the vehicles. The OBUs in the vehicles only need to perform two exponent operations in $\mathbb{G}_0$. Thus, most of the laborious signing operations in the vehicle are delegated to the cloud server through RSUs, so that the computation overhead of the vehicles can be reduced.

7.3. Simulation Evaluation

Next, we analyze the computation cost of our scheme by conducting experiments on a simulated RSU with an Intel CPU at 2.53 GHz and 4 GB RAM. The OBU in the vehicle, which has limited processing power, is simulated by an Android phone with a 1.2 GHz processor [27]. The simulations are developed with a pairing-based cryptography library [30]. A type A elliptic curve of 160-bit group order is chosen. We assume that each vehicle has the same number of persistent attributes and dynamic attributes, which means that each of them has half of the whole attributes.

From Figure 2, we can observe that the computation costs for key generation in these schemes all grow with N_c, while those for our scheme and Liu et al. [20] grow at a slower pace than Xia et al. [9], and our scheme costs almost the same as Liu et al. [20].

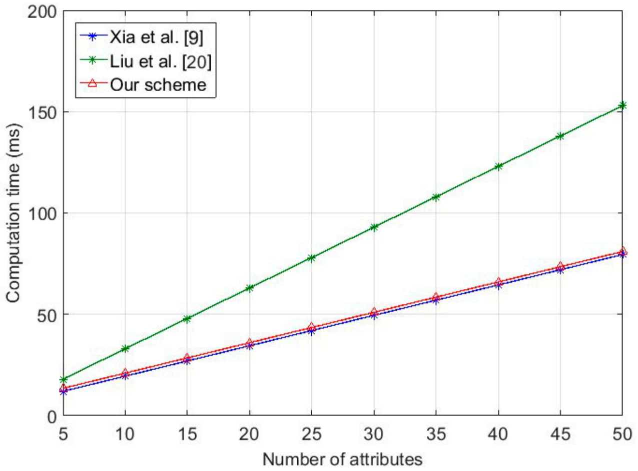

Figure 2. Computation cost of key generation on attribute authority.

In the message broadcasting phase, the OBU in our scheme encrypts the message with a predefined access policy, and signs the ciphertext. To compare the efficiency of Xia et al. [9], Liu et al. [20] and our scheme, we evaluated the computation costs under two situations, namely non-authentication and authentication. Figure 3 shows that the computation time for message broadcasting is related with N_c in T_a. Firstly, the cost of Xia et al. [9] and Liu et al. [20] without authentication increase with N_c in T_a, while remaining constant at a low level in our scheme. Then, we compared our scheme with Liu et al. [20] with authentication, to illustrate the encryption efficiency of our authentication scheme. As shown in the figure, the time cost of Liu et al. [20] is related to N_c in T_a. Although the results for our scheme are slightly greater than the previous situation, they are still constant, which illustrates that our scheme is more efficient. Figure 4 illustrates the computation time for the OBU by decrypting the ciphertext. The data decryption time of Liu et al. [20] also increased with N_c in the T_a, while Xia et al. [9], while, on the contrary, our scheme, based on decryption outsourcing, remained constant.

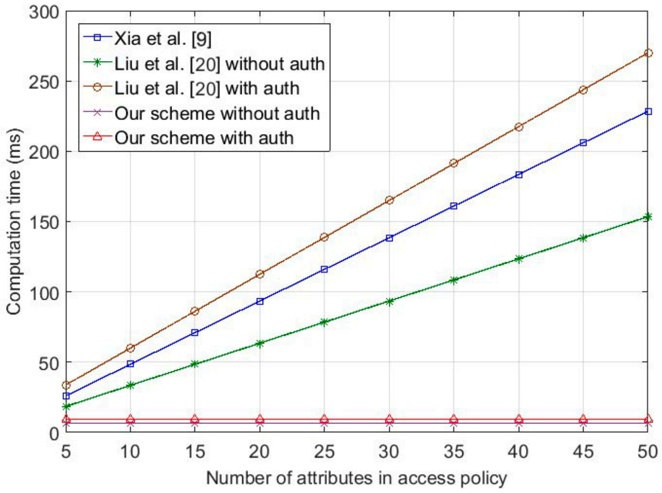

Figure 3. Computation cost of message broadcasting for on-board unit.

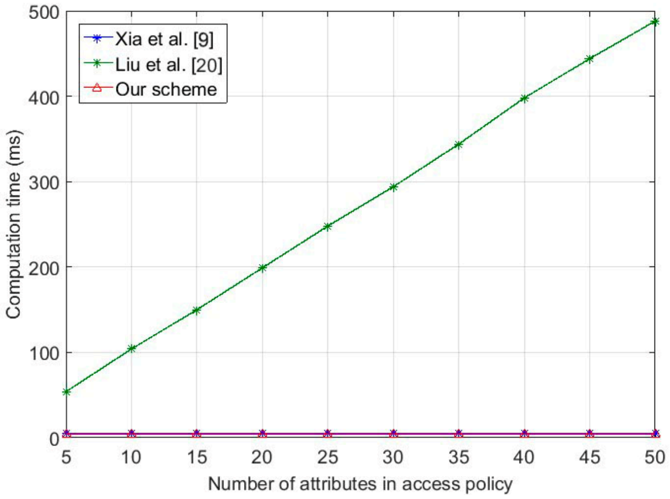

Figure 4. Computation cost of message decryption for on-board unit.

8. Conclusions

This paper proposes a secure and efficient message access control and authentication scheme for VCC based on HABE and ABS. In our scheme, the attributes of vehicle are divided into persistent attributes and dynamic attributes. These two kinds of attributes are managed by different AAs, which reduces the key management for single TAs. To prevent the forging of messages, we adopt ABS to anonymously authenticate the origin of messages in VCC. Considering the resource-limited OBUs in vehicles, our scheme outsources the heavy computations from OBUs to cloud servers and RSUs. The analysis shows that our scheme achieves efficient access control and authentication of messages in VCC.

Acknowledgments: This work was supported by the National Key Research and Development Program of China under Grant No. 2016YFB0800605, by the National Natural Science Foundation of China under Grant No. 61572080, by the Key Program of Joint Funds of the National Natural Science Foundation of China under Grant No. U1736212.

Author Contributions: Qinlong Huang contributed to the original ideas and designed the simulations. Yixian Yang contributed to the scheme design and revised the work. Yuxiang Shi analyzed the simulation results and drafted the manuscript.

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

References

1. Li, X.; Qiao, C.; Yu, X.; Wagh, A.; Sudhaakar, R. Toward effective service scheduling for human drivers in vehicular cyber-physical systems. *IEEE Trans. Parallel Distrib. Syst.* **2012**, *23*, 1775–1789. [CrossRef]
2. Lee, U.; Lee, J.; Park, J.S.; Gerla, M. FleaNet: A virtual market place on vehicular networks. *IEEE Trans. Veh. Technol.* **2010**, *59*, 344–355.
3. Shojafar, M.; Cordeschi, N.; Baccarelli, E. Energy-efficient adaptive resource management for real-time vehicular cloud services. *IEEE Trans. Cloud Comput.* **2016**, *PP*. [CrossRef]
4. Cordeschi, N.; Amendola, D.; Shojafar, M.; Baccarelli, E. Distributed and adaptive resource management in cloud-assisted cognitive radio vehicular networks with hard reliability guarantees. *Veh. Commun.* **2015**, *2*, 1–12. [CrossRef]
5. Alam, K.; Saini, M.; Saddik, A. tNote: A social network of vehicles under Internet of Things. In Proceedings of the 1st International Conference on Internet of Vehicles (IOV 2014), Beijing, China, 1–3 September 2014; pp. 227–236.

6. Smaldone, S.; Han, L.; Shankar, P.; Iftode, L. RoadSpeak: Enabling voice chat on roadways using vehicular social networks. In Proceedings of the 1st Workshop on Social Network Systems, Glasgow, Scotland, 1 April 2008; pp. 43–48.
7. Huang, Q.; Ma, Z.; Yang, Y.; Fu, J.; Niu, X. EABDS: Attribute-based secure data sharing with efficient revocation in cloud computing. *Chin. J. Electron.* **2015**, *24*, 862–868. [CrossRef]
8. Bethencourt, J.; Sahai, A.; Waters, B. Ciphertext-policy attribute based encryption. In Proceedings of the 2007 IEEE Symposium on Security and Privacy, Berkeley, CA, USA, 20–23 May 2007; pp. 321–334.
9. Xia, Y.; Chen, W.; Liu, X.; Zhang, L.; Li, X.; Xiang, Y. Adaptive multimedia data forwarding for privacy preservation in vehicular ad-hoc networks. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 2629–2641. [CrossRef]
10. Green, M.; Hohenberger, S.; Waters, B. Outsourcing the decryption of ABE ciphertexts. In Proceedings of the 20th USENIX Conference on Security, San Francisco, CA, USA, 8–12 August 2011; p. 34.
11. Maji, H.; Prabhakaran, M.; Rosulek, M. Attribute-based signatures. In Proceedings of the 11th Cryptographers' Track at the RSA Conference 2011: Topics in Cryptology, San Francisco, CA, USA, 14–18 February 2011; pp. 376–392.
12. Pietrowicz, S.; Shim, H.; Crescenzo, G.D.; Zhang, T. VDTLS—Providing secure communications in vehicle networks. In Proceedings of the INFOCOM Workshops 2008, Phoenix, AZ, USA, 13–18 April 2008; pp. 1–6.
13. Mallissery, S.; Pai, M.; Pai, R.; Smitha, A. Cloud enabled secure communication in vehicular ad-hoc networks. In Proceedings of the 2014 International Conference on Connected Vehicles and Expo, Vienna, Austria, 3–7 November 2014; pp. 596–601.
14. Nema, M.; Stalin, S.; Tiwari, R. RSA algorithm based encryption on secure intelligent traffic system for VANET using Wi-Fi IEEE 802.11p. In Proceedings of the 2015 International Conference on Computer, Communication and Control, Indore, India, 10–12 September 2015; pp. 1–5.
15. Sahai, A.; Waters, B. Fuzzy identity-based encryption. In Proceedings of the 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, 22–26 May 2005; pp. 457–473.
16. Goyal, V.; Pandey, O.; Sahai, A.; Waters, B. Attribute-based encryption for fine-grained access control of encrypted data. In Proceedings of the 13th ACM Conference on Computer and Communications Security, Alexandria, VA, USA, 30 October–3 November 2006; pp. 89–98.
17. Huang, D.; Verma, M. ASPE: Attribute-based secure policy enforcement in vehicular ad hoc networks. *Ad Hoc Netw.* **2009**, *7*, 1526–1535. [CrossRef]
18. Ruj, S.; Nayak, A.; Stojmenovic, I. Improved access control mechanism in vehicular ad hoc networks. In Proceedings of the 10th International Conference on Ad-Hoc, MOBILE, and Wireless Networks, Paderborn, Germany, 18–20 July 2011; pp. 191–205.
19. Yeh, L.; Chen, Y.; Huang, J. ABACS: An attribute-based access control system for emergency services over vehicular ad hoc networks. *IEEE J. Select. Areas Commun.* **2011**, *29*, 630–643. [CrossRef]
20. Liu, X.; Shan, Z.; Zhang, L.; Ye, W.; Yan, R. An efficient message access quality model in vehicular communication networks. *Signal Process.* **2016**, *120*, 682–690. [CrossRef]
21. Zhang, L.; Wu, Q.; Domingo-Ferrer, J. Distributed aggregate privacy-preserving authentication in VANETs. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 516–526. [CrossRef]
22. Sánchez-García, J.; García-Campos, J.M.; Reina, D.G.; Toral, S.L.; Barrero, F. On-siteDriverID: A secure authentication scheme based on Spanish eID cards for vehicular ad hoc networks. *Future Gener. Comput. Syst.* **2016**, *64*, 50–60. [CrossRef]
23. Kang, Q.; Liu, X.; Yao, Y.; Wang, Z.; Li, Y. Efficient authentication and access control of message dissemination over vehicular ad hoc network. *Neurocomputing* **2016**, *181*, 132–138. [CrossRef]
24. Chim, T.; Yiu, S.; Hui, L.; Li, V. VSPN: VANET-based secure and privacy-preserving navigation. *IEEE Trans. Comput.* **2014**, *63*, 510–524. [CrossRef]
25. Zhang, P.; Chen, Z.; Liu, J.; Liang, K.; Liu, H. An efficient access control scheme with outsourcing capability and attribute update for fog computing. *Future Gener. Comput. Syst.* **2016**, *78*, 753–762. [CrossRef]
26. Huang, Q.; Yang, Y.; Shen, M. Secure and efficient data collaboration with hierarchical attribute-based encryption in cloud computing. *Future Gener. Comput. Syst.* **2017**, *72*, 239–249. [CrossRef]
27. Studer, A.; Shi, E.; Bai, F.; Perrig, A. Tacking together efficient authentication, revocation, and privacy in VANETs. In Proceedings of the 6th Annual IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks, Rome, Italy, 22–26 June 2009; pp. 1–9.

28. Liu, X.; Xia, Y.; Chen, W.; Xiang, Y.; Hassan, M.; Alelaiwi, A. SEMD: Secure and efficient message dissemination with policy enforcement in VANET. *J. Comput. Syst. Sci.* **2016**, *82*, 1316–1328. [CrossRef]
29. Ma, H.; Zhang, R.; Wan, Z.; Lu, Y.; Lin, S. Verifiable and exculpable outsourced attribute-based encryption for access control in cloud computing. *IEEE Trans. Dependable Secur. Comput.* **2017**, *14*, 679–692. [CrossRef]
30. The Pairing-Based Cryptography Library. Available online: http://crypto.stanford.edu/pbc (accessed on 24 December 2017).