Efficient Privacy-Preserving Chatting Scheme with Degree of Interest Verification for Vehicular Social Networks

Khaled Rabieh, Mohamed Mahmoud, Ambareen Siraj, and Jelena Misic

1Department of Electrical and Computer Eng., Tennessee Tech University, Cookeville, TN, USA
2Department of Computer Science, Tennessee Tech University, Cookeville, TN, USA
3Department of Computer Science, Ryerson University, Toronto, ON, Canada

Abstract—Wireless communication capabilities of Vehicular Ad Hoc Networks (VANETs) have been utilized in various cutting-edge applications such as Vehicular Social Networks (VSNs). One of the benefits of VSNs is sharing of common-interest information among vehicle drivers. Drivers may benefit from identifying neighbors that have interest common with them along with the extent of their interest. However, there are some privacy issues that should be addressed. Revealing the nature and degree of interests (DOI) of drivers can be in violation of their privacy. In this paper, we propose an efficient chatting scheme among drivers that preserves such privacy. We use attribute based encryption (ABE) technique for anonymous common interest verification and homomorphic encryption technique for anonymous DOI verification. Moreover, we propose an efficient search mechanism to enable vehicles to check if they have common interests with low computation and communication overhead. To secure conversation, a key agreement protocol is used to enable the drivers that have the same interest and the desired DOI to establish a shared secret key. Our extensive evaluations demonstrate that our scheme can successfully preserve drivers’ privacy with low communication and computation overhead.

Keywords: Attribute based encryption (ABE), homomorphic encryption, privacy preservation, chatting schemes, vehicular social networks, and VANETs.

I. INTRODUCTION

In Vehicular Ad Hoc Networks (VANETs), onboard units are installed in vehicles to enable them to communicate with each other and with roadside units (RSUs) installed along the roads [1], [2]. VANETs have received extensive attention from both industry and academia due to the need for better and safer driving conditions. Many new and useful applications can be implemented in VANETs. These applications can be categorized into safety applications such as forwarding collision warning and urgent braking messages, and non-safety applications such as finding free parking slot and chatting [3].

The focus of this paper is on the chatting applications of Vehicular Social Networks (VSNs). In these applications, drivers look for neighbors on the road who have common interests to share information. They may also want to find drivers who have a certain minimum degree of interest (DOI). A driver’s DOI represents the level of knowledge and experience he has in a certain interest. Chatting with experienced people sometimes helps drivers to obtain accurate replies to their inquiries. However, there are some privacy issues that should be addressed. Revealing the nature and degree of interests of drivers can be in violation of their privacy. Without preserving privacy, drivers will be discouraged to participate in chatting applications.

In this paper, we propose an efficient privacy-preserving chatting scheme with DOI verification. Our objective is not only protecting the privacy of the interests but also their degrees. We use attribute based encryption (ABE) technique for anonymous common interest verification and homomorphic encryption technique for anonymous DOI verification. Drivers who have common interests receive the same set of secret keys. When a driver initiates a chatting request to look for a neighboring driver who has a set of interests, he uses the interests’ corresponding keys and ABE scheme to encrypt a packet. Only the drivers who have the same interests should have the necessary secrets to decrypt the packet. However, the drivers who do not have the interests cannot decrypt the packet or even know the interests requested by the sender. We propose an efficient mechanism to enable the drivers to check whether they have the requested interests with low computation and communication overhead. In addition, an algorithm is used to enable the initiator driver to find drivers with a minimum DOI without revealing this value or the drivers’ DOIs. Finally, to secure the conversation, a key agreement protocol is used to enable the vehicles that have the same interest and meet the desired DOI requirement to establish a key.

Comparing to the existing schemes that require exchanging several packets to conclude whether drivers share common interests, drivers in our scheme can know whether they have the requested interests from the first packet sent by the vehicle with low communication overhead. This minimizes the computation/communication overhead and the required time to find a driver having the requested interests. Moreover, our scheme enables the initiator driver to request more than one interest in one packet. This is not only efficient but also more practical and realistic. We also discuss an interest revocation technique that uses the RSUs to distribute the revocation information without revealing the interests’ secret keys to...
the RSUs. Our extensive evaluations demonstrate that our scheme can successfully preserve drivers’ privacy with low communication and computation overhead.

The remainder of the paper is organized as follows. In Section II, the network and threat models are described. We discuss some preliminaries in Section III. Our scheme is presented in Section IV. Performance evaluations and security/privacy preservation analysis are explained in Sections V and VI, respectively. Finally, Section VII discusses the related works and Section VIII concludes the paper.

II. NETWORK AND THREAT MODELS

A. Network Model

As shown in Fig. 1, the considered network model has the following entities.

- **Vehicles**: The network has a large number of vehicles who can communicate with each other and with the RSUs. Each vehicle should contact the trusted party (TP) to receive the necessary credentials to run our scheme. The credentials are a group of secret keys associated to the drivers' interests.

- **TP**: The TP is a centralized authority that is responsible for generating the secret keys for the vehicles. It is also responsible for updating the drivers’ interests and revoking interests’ keys.

- **RSUs**: RSUs are access points that are deployed on roads. They are connected to the TP via a fast communication technology, e.g., wired cables, 4G, or WiMax. Vehicles can run our scheme without support from the RSUs that are used only for interest revocation. RSUs receive the revocation information from the TP and forward it to vehicles in its vicinity.

B. Adversary and Threat Model

In our scheme, the TP is fully trusted, and the RSUs cannot know the secret keys of the drivers’ interests. The drivers are honest but curious. They do not want to disrupt the communications, but they are curious to know the other drivers’ interests and DOIs. A possible attack scenario is illustrated in Fig. 2.

An attacker can falsely claim that he has an interest to chat with the drivers to obtain information about their activities, e.g., their work place, home address, etc.

Another attacker eavesdrops on the vehicles’ communications and try to obtain sensitive information, e.g., whether someone is heading to the stadium. Finally, we assume that the clocks of the vehicles are synchronized to enable them to identify replayed packets.

III. PRELIMINARIES

In this section, we first define interests and explain some basic techniques that will be used in our scheme.

A. Interests

Examples of interests are \{Sport\}, \{Sport, Football\}, \{Sport, Football, USA, TN\}, etc. If a driver looks for someone who is interested in any kind of sports, he should use the interest \{Sport\}. Similarly, if a driver looks for someone who is interested in football and football in Tennessee state in USA, he should use \{Sport, Football\} and \{Sport, Football, USA, TN\}, respectively. Every interest has a degree that can quantify the driver’s experience and knowledgable in the field of the interest. Our scheme allows drivers to search for more than one interest, e.g., \{Sport, Football\} and \{Sport, Swimming\}. It also allows the vehicles to select a driver who a minimum DOI value.

B. Attribute Based Encryption

Attribute Based Encryption schemes are usually used for secure multicast protocols. In these protocols, a message sender uses some attributes’ secret keys to encrypt a message and sends it alongside with the attributes. Only the nodes that have the attributes’ secret keys can decrypt the message.

In our scheme, we use attribute based encryption to identify the drivers who have a group of interests but without sending the interests to preserve the interest privacy from the drivers who do not have the interests. We use the attribute based encryption scheme proposed in [4] that uses the bilinear pairing mapping.
Let $q$ be a large prime number and $Z_q$ be a finite field of order $q$. $G_1$ is a cyclic multiplicative group with generator $g$, whose order is $q$, and $G_T$ is a cyclic multiplicative group with the same order $q$. Let $\hat{e}: G_1 \times G_1 \to G_T$ be a bilinear map that is called a bilinear pairing and has the following properties:

1. **Bilinearity**: $\hat{e}(g_1^a, g_2^b) = \hat{e}(g_1, g_2)^{ab}$ for all $g_1, g_2 \in G_1$ and $a, b \in \mathbb{Z}_q$.

2. **Non-degeneracy**: There exist $g_1, g_2 \in G_1$ such that $\hat{e}(g_1, g_2) \neq 1$. In other words, the bilinear mapping does not map all pairs in $G_1 \times G_1$ to the identity in $G_T$.

3. **Computability**: There is an efficient algorithm to compute $\hat{e}(g_1, g_2)$ for all $g_1, g_2 \in G_1$.

### C. Yao’s Millionaires’ Problem

Yao’s Millionaires’ problem is a secure multiparty communication problem which was first introduced by Andrew Yao [5]. The problem discusses two millionaires who are interested in knowing which of them is richer without revealing their actual wealth. This problem is analogous to a more general problem called greater than “GT”, where there are two numbers $a$ and $b$ and the goal is to solve the inequality $a \geq b$ without revealing the actual values of $a$ and $b$. This is used in our scheme to verify whether a vehicle’s DOI is greater than a certain DOI value without revealing the two DOIs. In our scheme, we will use the proposed scheme in [6] that can solve the Millionaires’ problem efficiently. This scheme uses the homomorphic encryption.

The homomorphic encryption [7], [8] is a form of public key encryption that allows computations to be carried out on ciphertext, thus generating an encrypted result which, when decrypted, matches the result of operations performed on the plaintext. The homomorphic encryption can be expressed as follows:

$$E(m_1) \odot E(m_2) = E(m_1 \odot m_2)$$

The homomorphic encryptions used in our paper should do addition and bitwise exclusive OR operations.

### IV. PRIVACY-PRESERVING CHATTING SCHEME

In this section, we discuss the system initialization, our chatting scheme, and interest revocation.

#### A. System Initialization

$q$ is a large prime number and $Z_q$ is a finite field of order $q$. Let $G_1$ be a cyclic multiplicative group with generator $g$, whose order is $q$, and $G_T$ be a cyclic multiplicative group with the same order $q$. Let $\hat{e}: G_1 \times G_1 \to G_T$ be a bilinear map. Let $E()$ and $D()$ are homomorphic encryption and decryption schemes. The TP chooses a random element $t_i \in Z_q^*$ for each interest $i$. In addition, it chooses another random element $y \in Z_q^*$ and computes $\hat{e}(g, y^y) = Y$. The TP keeps $y$ and the set $\{y_0, t_1, \cdots, t_a\}$ secret, where $a$ is the total number of interests. Let $H$ be a secure cryptographic hash function, such that $H: \{0, 1\}^* \to G_1$. The TP publishes the public parameters of the system $\{q, g, g^i, \cdots, g^a, Y, H, \hat{e}, E(), D()\}$.

For every interest $i$, the TP chooses a random number $r_i \in Z_q^*$ and computes $d_i = g^{r_i}$, where $t_i$ is the secret key corresponding to the interest $i$. Then, for each combination of interests $j$, the TP computes a corresponding $D_j = g^{\gamma-j}$, where $\gamma = \sum_{i=1}^{a} r_i$, and distribute it to the vehicles that have the combination $j$.

For each vehicle, the TP computes a group of certified public/private key pairs. Then, it loads each vehicle with the group of certified public/private keys, $d_i$ for each interest $i$, $D_j$ for each combination of interests $j$. These will be used in our scheme to enable drivers to prove that they have a combination of interests. For each interest, each driver should quantify its degree of interest ($W$) which depicts the level of experience/knowledge the driver has in the relevant interest.

#### B. Privacy-Preserving Chatting Scheme

The exchanged packets in our scheme are illustrated in Fig. 3. Vehicle $A$ searches for a neighboring vehicle that has a list of $\gamma$ interests.

**Chatting Request Packet**: Initially, vehicle $A$ picks a random element $r_a$, where $r_a \in G_T$ and encrypts it so that only the vehicles that have the $\gamma$ secret keys associated to the list of requested interests can decrypt it. To encrypt $r_a$, vehicle $A$ chooses a random element $s \in Z_q^*$. For every interest $i$ in the list of $\gamma$ interests, vehicle $A$ computes $T_i = g^{i+s}$ and then computes the ciphertext $\{\tilde{C}, \tilde{C}, \{C_i\}_{i \in \gamma}\}$ as follows:

$$\tilde{C} = r_a \times Y^s$$

$$\tilde{C} = g^s$$

$$C_i = \{T_i\}_{i \in \gamma}$$

Then, the vehicle $A$ computes $\partial$ to enable neighboring vehicles to verify whether they have the same $\gamma$ interests efficiently. $\partial$ is computed by hashing the concatenation of the current timestamp ($TS$) and the set of secret keys associated
to the interests, i.e., \( \partial \) to be \( H(TS, \{ d_i \}_{i \in \gamma}) \). Finally, the vehicle \( A \) composes a Chatting Request packet that has \( \{ C, \hat{C}, \{ C_i \}_{i \in \gamma}, \partial, TS, Cert_A \} \), where \( Cert_A \) is a public key certificate for a pseudonym used only for short time. Then, it broadcasts the packet to all vehicles in its vicinity, as illustrated in Fig. 3.

**Chatting Response Packet:** Only vehicles which have the same \( \gamma \) interests should be able to decrypt the ciphertext sent from vehicle \( A \) and reply back to vehicle \( A \). This is attributed to the fact that vehicles having the same interests receive the same secret keys \( \{ d_i \}_{i \in \gamma} \). When vehicle \( B \) receives the Chatting Request packet, it first uses \( \partial \) to check whether it has the \( \gamma \) interests requested by vehicle \( A \). For each combination of interests it has, vehicle \( B \) concatenates \( TS \) with the interests’ secrets \( \{ \{ d_i \} \} \) and hashes the concatenation. If the resultant hash function is similar to \( \partial \), vehicle \( B \) shares the \( \gamma \) interests requested by vehicle \( A \). Then, vehicle \( B \) uses the interests’ secrets to decrypt the ciphertext and recover \( r_a \) as follows:

\[
r_a = \frac{\hat{C}}{\hat{e}(C, D)} \prod_{i = 1}^{\gamma} \hat{e}(C_i, d_i)
\]

The verification of the decryption scheme is as follows:

\[
\frac{\hat{C}}{\hat{e}(C, D) \prod_{i = 1}^{\gamma} \hat{e}(C_i, d_i)} = \frac{r_a \times Y^s}{\hat{e}(g^s, g^{g^g - \tau}) \prod_{i = 1}^{\gamma} \hat{e}(g^i, g^r)} = \frac{r_a \times \hat{e}(g, g)^{ys}}{\hat{e}(g^s, g^{y - \tau}) \prod_{i = 1}^{\gamma} \hat{e}(g^i, g^r)}
\]

For vehicle \( B \) to prove to \( A \) that it has the requested \( \gamma \) interests, it should prove that it could decrypt the ciphertext correctly. To do that, it encrypts \( r_a \) with the public key of vehicle \( A \). As shown in Fig. 3, vehicle \( B \) selects a random element \( r_b \in Z_q^* \) and includes it in the encryption. The chatting key is \( K_{ba} = H(r_a, r_b, TS) \). Vehicle \( B \) generates public/private homomorphic encryption key pair \((PK, sk)\) that is used for DOI verification. It encrypts its DOI \((W_b)\) with the homomorphic public key to produce the ciphertext \( E_{PK}(W_b) \). Finally, it sends back a Chatting Response packet that has \( \{ E_{PK}(r_a, r_b), H(K_{ba}, 1), E_{PK}(W_b), PK \} \) to vehicle \( A \). Vehicle \( B \) adds the key confirmation code \( H(K_{ba}, 1) \) to the packet to enable vehicle \( A \) to ensure that it computed the same key computed by \( B \).

**Degree of Interest Verification:** When \( A \) receives the Chatting Response packet, it first decrypts the ciphertext to verify that it has \( r_a \). Then, it computes the chatting session key \( K_{ba} = H(r_a, r_b, TS) \) and verify the key confirmation code sent from \( B \). It uses the homomorphic public key to encrypt its DOI and random number \( R \), i.e., \( E_{PK}(W_a) \) and \( E_{PK}(R) \), respectively. Then using the homomorphic properties, it computes \( E_{PK}((W_b - W_a) \oplus R) = (E_{PK}(W_a) - E_{PK}(W_b)) \oplus E_{PK}(R) \). Thanks to the homomorphic encryption properties, \( A \) can compute \( E_{PK}((W_b - W_a) \oplus R) \) without knowing \( W_b \). From Fig. 3, vehicle \( A \) sends to \( B \) the following: \( \{ H(K_{ba}, 2), E_{PK}((W_b - W_a) \oplus R) \} \). Vehicle \( A \) adds the key confirmation code \( H(K_{ba}, 2) \) to the packet to enable \( B \) to verify that it computed the same key computed by \( A \). Vehicle \( B \) decrypts the homomorphic ciphertext using the secret key \( sk \) to obtain \( (W_b - W_a) \oplus R \), then it sends the last bit to \( A \). It is obvious that \( B \) cannot know \( W_a \) because it does not know \( R \), and \( A \) cannot know \( W_b \) because it does not know \( sk \).

Finally, \( A \) takes the XOR of the obtained bit from \( B \) and the most significant bit of \( R \). If the resultant bit is \( 1 \) then \( W_b < W_a \). Actually, what \( A \) calculates is the last bit in \( W_b - W_a \) because \( ((W_b - W_a) \oplus R) \oplus R = (W_b - W_a) \), \( A \) does not have access to all the bits in \( (W_b - W_a) \), otherwise it can know \( W_b \). It knows only the last bit in \( (W_b - W_a) \), which represents the sign of \( (W_b - W_a) \). If the bit is \( 1 \), \( (W_b - W_a) \) is negative which indicates that \( W_a > W_b \), and if the bit is \( 0 \), \( (W_b - W_a) \) is positive or zero which indicates that \( W_a \leq W_b \).

**C. Interest Revocation**

Interest revocation is one of the most difficult problems in privacy preserving chatting applications and attribute based encryption scheme [9]. In our scheme, the TP is responsible for the revocation of vehicles’ interests. The common approach for interest revocation is by rekeying the vehicles’ interests. The TP picks a random element \( \hat{t}_i \in Z_q^* \) for every interest \( i \), and computes \( r_{ki} = \frac{\hat{t}_i}{r_{ki}} \) that is used to update the vehicles’ interests. Then, it securely sends the list \( \{ r_{ki} \}_{1 \leq i \leq n} \) to the RSUs, where \( n \) is the total number of interests. The RSUs send \( r_{ki} \) only to the vehicles that have the interest \( i \). Then, the vehicles update the secret of each interest \( i \) as follows: new \( d_i = d_i \oplus r_{ki} \). We delegate the distribution of the list \( \{ r_{ki} \}_{1 \leq i \leq n} \) to the RSUs because they can communicate to vehicles.

**V. PERFORMANCE EVALUATIONS**

**A. Communication Overhead**

In our scheme, there are four main packets transmitted between vehicles. The Chatting Request packet is the first packet broadcasted by an initiator vehicle \( A \) to discover the vehicles with common interests. This packet has \( \hat{C}, \hat{C}, \{ C_i \}_{i \in \gamma}, \partial, TS, Cert_A \). Using an elliptic curve of order 224 bits, each of \( \hat{C} \) and \( \hat{C} \) requires 56 bytes. \( \{ C_i \}_{i \in \gamma} \) requires 56 \( \times \gamma \), where \( \gamma \) is the number of interests requested by \( A \). \( \partial \) requires 56 bytes. Finally, the timestamp \( TS \) requires 8 bytes and the size of the certificate \( Cert_A \) is 60 bytes. The packet size is 56 + 56 + 56 \( \times \gamma \) + 56 + 8 + 60 = 236 + 56 \( \times \gamma \) bytes.
The Chatting Response packet has $E_{PK_a}(r_a, r_b)$, $H(K_{ba}, 1)$, $E_{PK}(W_b)$, $PK$. The public key encryption using a 2,048-bit key length produces 2,048 bits for all inputs less than 245 bytes. The size of $E_{PK_a}(r_a, r_b)$ and $H(K_{ba}, 1)$ are 256 and 56 bytes, respectively. The size of the homomorphic encryption’s ciphertext in $Z_n$ equals to double $n$. If $n$ is 2,048 bits, the ciphertext $E_{PK}(W_b)$ size equals to 256 × 2 = 512 bytes. The public key of the homomorphic function $PK$ is 256 bytes. We estimate the size of the Chatting Response packet to be 256 + 56 + 512 + 256 = 568 bytes.

Vehicle A replies back with a packet that has $H(K_{ba}, 2)$, $E_{PK}((W_b - W_a) \oplus R)$. The size of $H(K_{ba}, 2)$ and $E_{PK}((W_b - W_a) \oplus R)$ are 56 and 512 bytes, respectively. The packet size is 56 + 512 = 568 bytes.

B. Computation Overhead

The computation times of the multiplication, pairing, and exponentiation operations are measured in [10] using 3 GHz processor and 512 MB RAM. The measurements are 6.4 ms, 20 ms, and 12.4 ms for the multiplication, pairing, and exponentiation operations, respectively. Using these measurements, we compute the computation overhead of our scheme.

The composition of a Chatting Request packet requires one exponential and one multiplication operations to compute $C$, one exponentiation operation to compute $C$, one exponentiation operation to compute $C$, and one hashing operation to compute $\partial$. The required time to compose a packet is $(12.4 + 6.4) + 12.4 + (12.4 \times \gamma) = 31.2 + 12.4 \times \gamma$ ms. To decrypt the same packet, it requires $\gamma + 1$ pairing and one multiplication operations. It takes $(\gamma + 1) \times 20 + 6.4 = 26.4 + 20 \times \gamma$ ms to decrypt one message.

The computational times of the public key encryption and decryption of $r_a$, $r_b$ are 0.5 and 37.6 ms respectively. The homomorphic encryption requires one exponential operation to encrypt or decrypt a packet. The composition time of the packet $E_{PK_a}(r_a, r_b)$, $H(K_{ba}, 1)$, $E_{PK}(W_b)$, $PK$ takes 0.5 + 12.4. To decrypt the same packet, it requires one public key decryption and one homomorphic decryption. It takes 37.6 + 12.4 = 50 ms.

In order to compose the packet $\{H(K_{ba}, 2), E_{PK}((W_b - W_a) \oplus R)\}$, it requires two homomorphic encryption operations, which takes $2 \times 12.4 = 24.8$ ms. Finally, in order to decrypt this packet and obtain the value $(W_b - W_a) \oplus R)$, it requires only one exponentiation operation that takes 12.4 ms.

A driver can determine whether he shares the requested interests using hashing operations. Let the maximum number of interests a driver can have is $n$, the vehicle has to hash each $\gamma$ interests. Therefore, the maximum number of hashing operations is given by the equation

$$\frac{n!}{(n - \gamma)! \times \gamma!}$$

For example, the maximum number of hashing operations is 252 if $n$ and $\gamma$ are 10 and 5, respectively. It takes 252 × 0.3 = 75.6 ms to verify whether the vehicle has the requested interests, assuming computing one hash value requires 0.3 ms.

VI. Security/Privacy Preservation Analysis

In this section, we investigate the security/privacy preservation features provided by our scheme.

Mutual verification: In our scheme, the chatting request initiator A can ensure that the responding vehicle $B$ has the requested interests by ensuring that $B$ could decrypt the ciphertext and obtain $r_a$. On the other hand, vehicle $B$ can ensure that $A$ has the interests from the hash value of the interest’ secrets sent by $A$.

Interest Privacy: In our scheme, vehicle $A$ hashes the requested interests and broadcasts it. The vehicles that do not have the requested interests cannot know the interests the chatting request initiator looks for. This is because the hash functions are one way, i.e., it is infeasible to compute the input of the hash function which has information about the requested interests.

Degree of Interest Privacy: In our scheme, we consider the degree of interest private information and we hide it. However, we enable the vehicles to know whether they have more or less degree of interest without disclosing the exact value.

Unlinkability/anonymity: It is infeasible to link chatting requests even if they are for the same interests, i.e., it is infeasible to know if the requested interest are similar to old requests or not. Timestamp is used to calculate the hash value sent by $A$ so that every time a vehicle looks for the same set of interests, the hash value looks different. Moreover, chatting requests cannot also be linked from the ciphertexts sent from $A$ because every time it uses a new random value $r_a$ and $s$ which can make the ciphertext looks different even if the vehicle looks for the same interests. Therefore, there is no way to link two ciphertexts sent from the same vehicle for the same interests.

The real identity of the vehicles are fully preserved from both the vehicles that have the interests and the ones that do not have them. Each vehicle has a large number of certified pseudonyms and each pseudonym is used for a short time. Linking pseudonyms is not feasible. Moreover, the vehicles’ real identities are not used in any packet in our scheme.

Revocation: Once an interest is revoked, the vehicle cannot send valid Chatting Request packets with the interest. It cannot also know whether a chatting request has the interest. However, the revocation does not affect the other vehicles that still have the interest. Although the RSUs distribute the revocation information, they do not know the interests’ secrets. This is because they distribute the value $r_{k_i} = \frac{t_i}{t_{i+1}}$ that needs the old secrets to calculate the new ones, where new $d_i = d_i^k$. The RSU knows $r_{k_i}$ but it does not know $d_i$ to calculate the new secrets.

VII. Related Work

In [3], Smaldone et. al. propose a framework for building chatting groups from vehicles driving on the same road. The drivers can create the groups and other drivers can join them.
if they can match to the groups’ profiles. A privacy-preserving like-minded vehicle finding protocol is proposed in [11]. The drivers search for the vehicles that have common interests without revealing these secrets to the vehicles that do not have them. However, unlike our scheme, the protocol focuses on search for a vehicle that has one interest. Moreover, the vehicles should exchange several messages until they figure out whether they have the common interest or not. On contrary, our scheme enables the vehicles to verify whether they have the interests locally, i.e., only the vehicles that have the same interests should responds. This not only reduces the communication overhead but also shortens the required time to find a vehicle that has the same interests.

Zhu et. al. [12] propose proximity-based matching algorithms to discover potential friends based on common attributes without violating their privacy. Bethencourt et. al. [13] propose a ciphertext policy based attribute based encryption scheme. In order to communicate with other users who have specific attributes, the user should send an access tree in plaintext and encrypt a message with the secrets of the attributes in the access tree. Unlike our scheme, this scheme does not aim to protect the privacy of the attributes, i.e., an adversary can know the attributes of the user from the access tree sent with the message.

In sensor networks, the location privacy of the node that sends an event has been investigated [14], [15], [16]. In the context of smart grid, several schemes have been proposed to preserve the privacy of the power consumption and injection data [17], [18], [19]. Several schemes have been proposed to preserve the privacy of the nodes involved in a communication session in ad hoc networks [20], [21], [22]. However, these scheme cannot solve the problem addressed in this paper because they are designed for different network and threat models.

VIII. Conclusion

In this paper, we have proposed an efficient privacy-preserving chatting scheme with degree of interest verification in VSNs. In our scheme, drivers look for neighboring drivers that have a set of interests to chat without exposing the interest privacy. Our scheme also enables drivers to select other drivers that are more experienced and knowledgeable, without revealing their degree of interest. In order to improve the efficiency of our scheme, vehicles can decide whether they have the requested interests without exchanging several packets or doing excessive computations, i.e., instead of trying several combinations of the secret keys to decrypt a message, they need to perform an efficient search. The vehicles can compute shared secret key to secure their conversation. Our evaluations demonstrated that our scheme can preserve drivers’ privacy with low communication and computation overhead.

REFERENCES