SNB-PPB: Social-network-based-privacy-preserving Broadcast for Vehicular Communications

Yanheng Liu[1,2], Haifeng Zhu[1,2] and Jian Wang[1,2]
[1]*College of Computer Science and Technology, Jilin University, Changchun 130012, China*
[2]*Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China*

Keywords: Secure Vehicular Communications, Social Networks, Privacy Preservation.

Abstract: Internet Key Exchange (IKE) can be used in vehicular communications in vehicular ad hoc networks (VANETs), but it requires a stable communication procedure, which is usually difficult to achieve because of the highly dynamic context of a VANET. In this paper, we propose a new method named social-network-based-privacy-preserving broadcast (SNB-PPB) for secure vehicular communications, which can omit the key exchange procedure. It transfers the information of online social networks to offline vehicular communications, dividing messages into different privacy levels and vehicles into different trust levels based on trust relations on social networks. The sender uses attributes on social networks and the message's privacy level to encrypt messages; only those receivers who satisfy the corresponding trust level can decrypt the broadcast packet.To the best of our knowledge, this is the first time that social networks have been applied in VANETs to achieve secure vehicular communications and privacy preservation. Our simulation results show that our method enables the sender to decide the amount and percentage of vehicles that can decrypt the received broadcast.

1 INTRODUCTION

A vehicular ad hoc network (VANET) is a special form of wireless ad hoc network that allows information dissemination between vehicles (V2V) and with nearby roadside infrastructures (V2I) for facilitating numerous attractive and exciting applications with the aim of creating a safer and more efficient traffic environment (Xu et al., 2015). It can support vehicular intelligent control, such as collision avoidance, as well as an active safety system by allowing the exchange of a vehicle's information, such as speed, location, and acceleration. However, the information disseminated within the network contains users' private information and the communication procedure is at risk of eavesdropping, masquerading, trace tracking, and more problems, and therefore, more attention should be paid to the security of the communication procedure.

Many existing encryption algorithms have been ported and deployed to secure a range of vehicular applications, e.g., Internet Key Exchange version 2 (IKEv2) (Alsa'deh et al., 2013). However, simulation and experimental results show that the algorithms suffer from poor performance, because the key exchange procedure requires stable communication, which is difficult to achieve in a VANET (Wang and Li, 2009).

For this purpose, we use social networks and ciphertext-policy attribute-based encryption (CP-ABE)(Bethencourt et al., 2007) . Currently, social networks are becoming increasingly popular. Let us take Facebook as an example. Every account has a set of attributes, such as the ID, location, and sex of the holder. These attributes are sufficient to allow an individual to identify him/herself. It also provides a friends list, most members of which are the holder's families, classmates, friends, colleagues, and so on. In other words, most members are "trusted people." Furthermore, an individual's "trusted friend's friend" is frequently trustworthy but less so than his/her "friend." Thus, dissemination of trust exists among the social network nodes. As can be seen, the social network contains trust relationships. These attributes and trust relationships can help create the access structure and encrypt messages.

In this paper, we propose a method named social-network-based-privacy-preserving broadcast (SNB-PPB) for improving the security of information ex-

Liu, Y., Zhu, H. and Wang, J.
SNB-PPB: Social-network-based-privacy-preserving Broadcast for Vehicular Communications.
DOI: 10.5220/0006799901960203
In *Proceedings of the 4th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2018)*, pages 196-203
ISBN: 978-989-758-293-6

change in vehicular communications, which combines a trust-based privacy-preserving strategy and CP-ABE. When a vehicle user wishes to send a message to certain other vehicles, the vehicle user's social network attributes and the message's privacy level can be used to create an access structure and encrypt the message. Only those vehicles that satisfy the corresponding trust level can satisfy the access structure and decrypt the message packet.

The rest of the paper is structured as follows. Section 2 provides definitions used in our method and a problem statement. It also introduces previous work on the aspects of communication security and privacy preserving in VANETs. Section 3 presents related technologies used in our method. It also specifies the system model and the algorithms' definition. We give the specific simulations and analysis in Section 4. Finally, we conclude the paper in Section 5.

2 PROBLEM STATEMENT AND BACKGROUND

2.1 Problem Statement

Usually, a key exchange procedure is performed when one vehicle meets another and would like secure communication with the second vehicle in a VANET. The key exchange procedure can be considered a means of distinguishing between authorized and other vehicles. In social networks, the nodes other than the user's node can be divided into different trust levels based on the relations between them and the user ("me"), as shown in Fig. 1. This strategy can also be applied to differentiate different vehicles for vehicular communications. Every vehicle driver can connect the vehicle to his/her social network account. Then, the vehicle has attributes and relations with other vehicles. We can also classify the vehicle's surrounding vehicles into different trust levels. Since vehicles beyond the user's vehicle's communication radius cannot receive the broadcast, we pay more attention to the user's geographical neighbors. If a user sends a message encrypted with the attributes of his/her social network account and its privacy level is tl, then the user's geographical neighbors whose trust level is m can decrypt the message iff $m \leq tl$. As can be seen, we can set the message's privacy level to determine who is able to decrypt the message.

2.2 Definitions

N-Hops neighbor. Assuming that every social network account is one node, if B is A's friend in the social network, then a line is added between A and B. Then, we can draw a graph based on the social network. Exactly as in a computer network graph, if the length of the shortest path from node C to node D in the social network graph is N, we call D C's N-Hops neighbor.

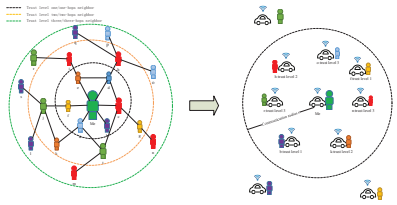


Figure 1: Social networks and vehicular communications.

Trust level tl and privacy level pl. tl is used to describe the level of trust one vehicle deserves. If B is A's i-Hops neighbor, then B's trust level for A is i. Trust level 1 is the highest trust level. The privacy level, pl, is used to describe the degree of privacy. It can determine the number of people who can decrypt the encrypted message. The value of pl is lower if fewer people are allowed to decrypt the message.

n and $Mhops$. This is an attribute set by the sender when encrypting messages that is used to help realize the privacy level. If the receiver is in trust level N and $N \leq n$, then the receiver is able to decrypt the message. n implies the privacy level of the message. The higher the privacy level of the message, the smaller is the value of n. $Mhops$ represents the max value of n. $1 \leq n \leq Mhops$.

Conditions C and k. C represents the plaintext in the broadcast, which consists of a set of attributes. It is used to partially identify the user and reduce the decryption calculation. $k \in ([0,1])$ represents C's ability to reduce calculation. The value of k is larger when the ability is lower.

Communication radius r and geographical neighbor. The broadcast sent by a vehicle can be received by other vehicles within an area. The range of the area is determined by the sending power of the sender, the sensitivity of receiver's sensor, and the environment (see (Ingelrest and Simplot-Ryl, 2006)). In our method, for simplicity we consider the area to be a circle, the radius of which is r. Then, vehicles located within the circle are called the sender's geographical neighbors.

Decryption percentage p. This represents the percentage of all the geographical neighbors of a vehicle that broadcasts a message that can decrypt the ciphertext of the encrypted message.

2.3 Related Work

Some progress on vehicular communication has already been achieved and presented in the literature. Wen-Bing et al. (Horng et al., 2012) proposed a novel group communication scheme for vehicular networks, in which a group is formed by a set of related vehicles having the same destination, such as a group of recreational vehicles traveling to the same tourist spot. Choi et al. (Choi and Jung, 2009) proposed a security framework with strong non-repudiation and privacy properties using a new approach constituting an ID-based cryptosystem in VANETs.

Goyal et al. (Goyal et al., 2006) developed a new cryptosystem for fine-grained sharing of encrypted data, called key-policy attribute-based encryption, in which ciphertexts are labeled with sets of attributes and private keys are associated with access structures that control which ciphertexts a user is able to decrypt. Liu et al. (Liu et al., 2016) proposed a communication model for VANETs by extending CP-ABE (Bethencourt et al., 2007) with a hierarchical structure of multiple authorities to achieve fine-grained access control of the transmitted messages.

In the methods in most previous studies, every vehicle needs authentication before communication is allowed. Some methods search for a third party to record every vehicle serially before communication; however, these methods are not practical, since the number of vehicles is very large, and they lack the required flexibility given that one vehicle may meet many other vehicles that have not previously appeared. Some methods are based on IKE, which suffers problems related to the high mobility and frequent topology changes in VANETs.

In this paper, we propose SNB-PPB to provide better information exchange security in vehicular communications. We use the relations between vehicle drivers on social networks to assist the authentication and use social network attributes to facilitate the encryption procedure. The method omits the key exchange procedure, which needs stable communication, and therefore it can be effective in a VANET.

3 MODELS

3.1 Preliminary Knowledge

3.1.1 Trust Evaluation Model

Like human beings, vehicles need to send messages having varying degrees of importance: some are "very important," some are "normally important," and some may be "not important." Correspondingly, it is expected that "very important" messages will be made known only to "very trusted vehicles and "normally important" messages to "very trusted" and "normally trusted" vehicles. Therefore, we propose a trust evaluation model to evaluate the extent to which one vehicle can trust another in a VANET.

In social networks, users have many "friends" who deserve trust. Then, these friends' friends also deserve trust; however, an attenuation in trust will exist. Thus, trust is disseminated among the social network nodes and an attenuation in trust exists during dissemination. We used this idea to create our trust evaluation model. In our model, we use trust levels to describe the concept "trust" and for simplicity consider the trust attenuation during dissemination to be linear. Each vehicle has a social network account and when the distance from B to A in a social network grows by one degree, B's trust level for A grows by one degree. When vehicle A is running on the road, there are many other vehicles nearby, e.g., B. If A is B's x-Hops neighbor, B is in trust level x for A. Thus, all vehicles can be divided into different trust levels for A.

3.1.2 Ciphertext-policy Attribute-based Encryption Model

Access structure T. Usually, a tree with root R is used to represent an access structure. Each non-leaf node of the tree represents a threshold gate, described by its children and a threshold value. If $num(x)$ is the number of children of a node x and k_x is its threshold value, then $0 < k_x \leq num(x)$. When $k_x = 1$, the threshold gate is an OR gate and when $k_x = num(x)$ it is an AND gate. Each leaf node x of the tree is described by an attribute and a threshold value $k_x = 1$. The parent of the node x in the tree is denoted by $parent(x)$. The function $att(x)$ is defined only if x is a leaf node and denotes the attribute associated with the leaf node x in the tree. The access tree T also defines an ordering between the children of every node; that is, the children of a node are numbered from 1 to num. The function $index(x)$ returns such a number associated with the node x, where the index values are uniquely assigned to nodes in the access structure for a given key in an arbitrary manner.

Satisfying the access tree. Let T be an access tree with root R. Denote by T_x the subtree of T rooted at the node x. Then, T is the same as T_R. If a set of attributes γ satisfies the access tree T_x, it is denoted by $T_x(\gamma) = 1$. $T_x(\gamma)$ is computed recursively as follows. If x is a non-leaf node, $T_{x'}(\gamma)$ is evaluated for all children x' of node x. $T_x(\gamma)$ returns 1 if and only if at least k_x

children return 1. If x is a leaf node, then $T_x(\gamma)$ returns 1 if and only if $att(x) \in \gamma$.

3.2 Our Construction

3.2.1 Setup

The setup algorithm selects a bilinear group $\mathbb{G}_0$ of prime order p with generator g. Then, it selects two random exponents $\alpha, \beta \in \mathbb{Z}_p$. The public key is published as

$$PK = \mathbb{G}_0, g, h = g^{\beta}, f = g^{1/\beta}, e(g,g)^{\alpha} . \quad (1)$$

and the master key MK is (β, g^{α}).

3.2.2 Encrypt and Send

The broadcast packet sent by the sender consists of two parts. The first part is conditions C (see Section 2), which describes part of the attributes of the sender's social network account in order to reduce the calculation when the receiver wants to decrypt the packet. The second part is the ciphertext, which is the result of the encryption. The sender sets two types of attribute to form the access structure. The first is a set of attributes S_{en}, which can identify the social network account itself uniquely. The second is an attribute n. n cannot be excessively large because of the "Six Degrees of Separation. Having succeeded in the real world, the theory in truth holds for online societies (Zhang and Tu, 2009). Therefore, $n \leq Mhops$. In our text, we use the attributes ID and location to form S_{en}. As an example, where the ID is windy123456 and the location is Changchun, we set $n = 3$ to complete the access tree: ("windy123456") AND ("changchun") AND ("n_1" OR "n_2" OR "n_3").

The encryption algorithm takes the public key, PK, the message, M, and the access structure, AC, as input, and outputs ciphertext, which implicitly contains AC. Then, the ciphertext is broadcast.

The encryption procedure is as follows. The algorithm first selects a polynomial q_x for each node x (including the leaves) in the tree T. These polynomials are selected as follows in a top-down manner, starting from the root node R. For each node x in the tree, the degree d_x of the polynomial q_x is set to be one less than the threshold value k_x of that node, that is, $d_x = k_x - 1$.

Starting with the root node R the algorithm selects a random $s \in \mathbb{Z}_p$ and sets $q_{\text{R}}(0) = s$. Then, it selects d_{R} other points of the polynomial q_{R} randomly to define it completely. For any other node x, it sets $q_x(0) = q_{parent(x)}(index(x))$ and selects d_x other points randomly to completely define q_x.

Let Y be the set of leaf nodes in T. The ciphertext is then constructed by giving the tree access structure T and computing

$$CT = (T, \tilde{C} = Me(g,g)^{\alpha s},$$
$$C = h^s, \forall y \in Y : C_y = g^{q_y(0)}, C_y' = H(att(y))^{q_y(0)}) . \quad (2)$$

3.2.3 Decryption

The sender's geographical neighbors can receive the packet and would like to decrypt it. We provide two methods to decrypt the packet: a centralized method and a distributed method.

First, we introduce the centralized decryption method. The receiver uploads the packet together with his/her social network account ID to the server and applies for decryption. The server responds to the application and attempts to decrypt the ciphertext. Since a huge calculation is required to traverse all-hops neighbors, the server can reduce the search range by means of the plaintext conditions in the packet. The decryption algorithm is described in Algorithm 1.

Algorithm 1: Tentative decryption algorithm in the server.

Require:
social network account ID (id); ciphertext in the broadcast (cip);
plaintext in broadcast conditions C; $Mhops$; PK; MK;

1: $i = 1$;
2: **while** $i <= Mhops$ **do**
3: **for all** id's i-Hops neighbors that satisfy C **do**
4: Take $S = S_{\text{kg}} \cup \{n\_i\}$ and MK as input and output SK by the key generation algorithm;
5: **if** SK can help decrypt cip with PK successfully **then**
6: Return the result;
7: **end if**
8: **end for**
9: $i++$;
10: **end while**
11: return fail;

In Step 3, we consider that C is satisfied when $C \subseteq S_{pi}$.

Step 4 comprises the key generation algorithm. S and MK are used to generate SK. The algorithm first selects a random $r \in \mathbb{Z}_p$ and then random $r_j \in \mathbb{Z}_p$ for each attribute $j \in S$. Then, it computes the key as

$$SK = (D = g^{(\alpha+r)/\beta}, \forall j \in S : D_j = g^r \cdot H(j)^{r_j},$$
$$D_j' = g^{r_j}) . \quad (3)$$

Step 5 comprises the decryption algorithm. We specify our decryption procedure as a recursive algorithm. For ease of exposition, we present the simplest form of the decryption algorithm.

We first define a recursive algorithm $DecryptNode(CT, SK, x)$ that takes as input a ciphertext CT, a private key SK, and a node x from T .

If the node x is a leaf node, then we let $i = att(x)$ and define as follows. If $i \in S$, then

$$DecryNode(CT, SK, x) = \frac{e(D_i, C_x)}{e(D_i', C_x')} = e(g,g)^{rq_x(0)} . \quad (4)$$

If $i \notin S$, then we define $DecryptNode(CT, SK, x) = \perp$. We now consider the recursive case where x is a non-leaf node. The algorithm $DecryptNode(CT, SK, x)$ then proceeds as follows. For all nodes z that are children of x, it calls DecryptNode(CT,SK,z) and stores the output as F_z. Let S_x be an arbitrary $k_x - sized$ set of child nodes z such that $F_z \neq \perp$. If no such set exists, then the node was not satisfied and the function returns $\perp$. Otherwise, we compute

$$\begin{aligned} F_x &= \prod_{z \in S_x} F_z^{\triangle_{i,S_x'}(0)}, where \quad i = index(z), \\ S_x' &= \{index(z) : z \in S_x\} \\ &= e(g,g)^{r \cdot q_x(0)} \\ &(using \quad polynomial \quad interpolation) . \end{aligned} \quad (5)$$

Having defined our function DecryptNode, we can now define the decryption algorithm. The algorithm begins by simply calling the function on the root node R of the tree T. If the tree is satisfied by S, $A = DecryptNode(CT, SK, r) = e(g,g)^{rq_R(0)} = e(g,g)^{rs}$ is set, and the algorithm now decrypts by computing

$$\tilde{C}/(e(C,D)/A) = \tilde{C}/(e(h^s, g^{(\alpha+r)/\beta})/e(g,g)^{rs}) = M . \quad (6)$$

Next, we introduce the distributed decryption method. One difference exists between the centralized and the distributed method in the encryption step. The attribute n is not used in encryption, but instead the plaintext of the packet is used. The decryption procedure is completely different from that in the centralized method. The main idea is that when A receives the packet, he/she tries using every friend's attributes on the friend list of his/her social network account to decrypt the packet. If A cannot decrypt the packet, he/she sends the packet to his/her friends for decryption (by the cellular network). If one of A's friends can decrypt it, the friend sends the result back; if not, the friend sends the packet to his/her friends' friends. It is a recursive procedure. The decryption algorithm is described in Algorithm 2.

Algorithm 2: Recursive decryption algorithm RDA.

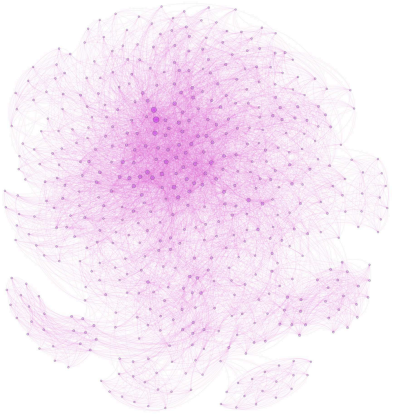
Require:
social network account ID (id); ciphertext in the broadcast packet (cip);
Conditions C; n; SOURCE ID (source_id)); PK; MK;

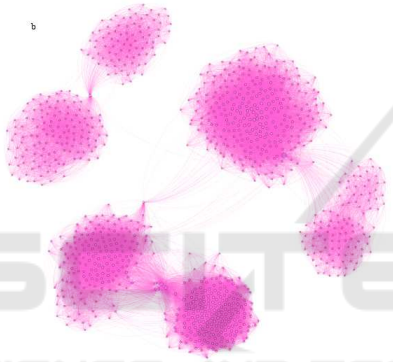
1: **for all** id's friends that satisfy C **do**
2: Take $S = S_{\text{kg}}$ and MK as input and output SK by the key generation algorithm;
3: **if** SK can help decrypt cip with PK successfully **then**
4: Return the result to source_id;
5: **end if**
6: **end for**
7: **if** $n == 1$ **then**
8: return fail to source_id;
9: **else**
10: $n = n - 1$;
11: send the packet to all friends except source_id;
12: **end if**
13: **for all** id's friends **do**
14: **if** id obtains the successful result from some friend **then**
15: return the result to source_id;
16: **end if**
17: **end for**
18: return fail to source_id;

4 SIMULATIONS AND ANALYSIS

We needed to conduct experiments to explore which factors affect the decryption percent p. Indirectly, we can infer this by the distribution of the hops from geographical neighbors to the sender. There are three groups of social networks from the Internet. Figure 2 depicts the social networks relations of Twitter, Facebook, and Hamsterster, respectively. There are also three groups of trajectories. We performed a random mapping between trajectories and social networks to form three groups of data. Then, every vehicle had one trajectory and a social network account in each group. Since the decryption method type does not affect whether a packet can be decrypted or not, we do not need to distinguish the centralized and distributed decryption method in the feasibility analysis.

First, we explored the manner in which the decryption percentage p changes according to the communication radius r. In each group, every vehicle had





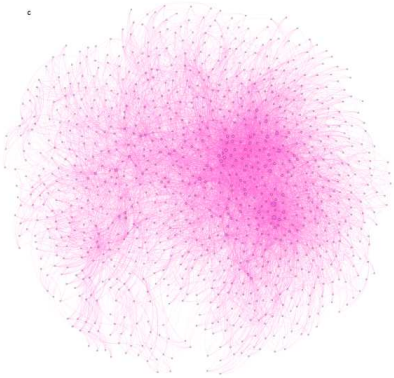
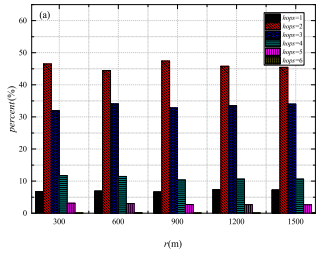


Figure 2: Social networks.



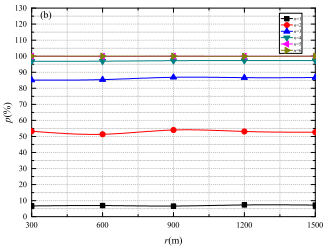
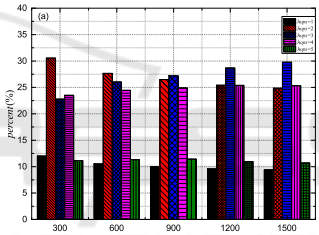


Figure 3: Decryption percentage p vs. communication radius r in Twitter.



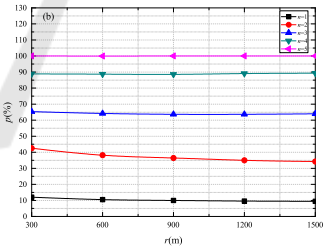


Figure 4: Decryption percentage p vs. communication radius r in Facebook.

a trajectory and a social network account. Then, we chose one moment and recorded the location of each vehicle. We changed r and observed the change in p. Figures. 3, 4, and 5 show that r does not affect p. The number of the sender's geographical neighbors changes when r changes. However, p does not change, since the hops from other vehicles to the sender are fixed after mapping. We provide a proof in the following.

Then, we explored the manner in which the decryption percentage p changes according to time t. In

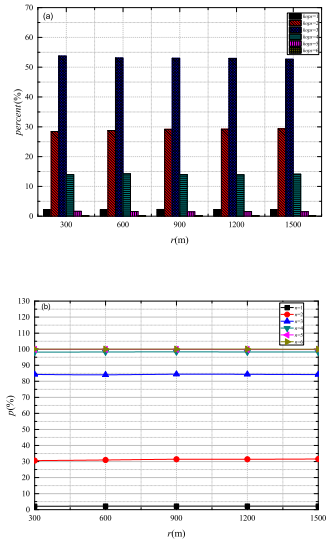


Figure 5: Decryption percentage p vs. communication radius r in Hamsterster.

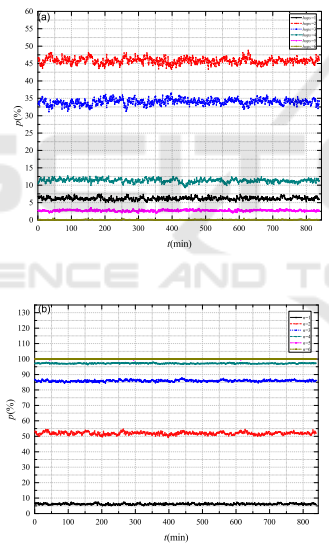


Figure 6: Decryption percentage p vs. time t in Twitter.

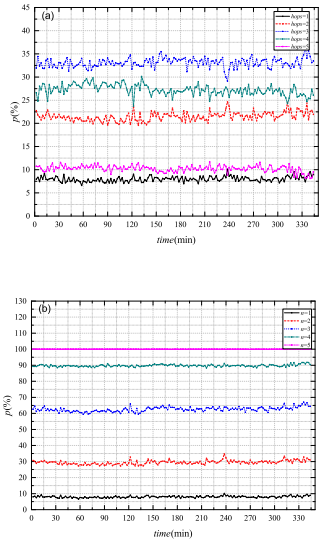


Figure 7: Decryption percentage p vs. time t in Facebook.

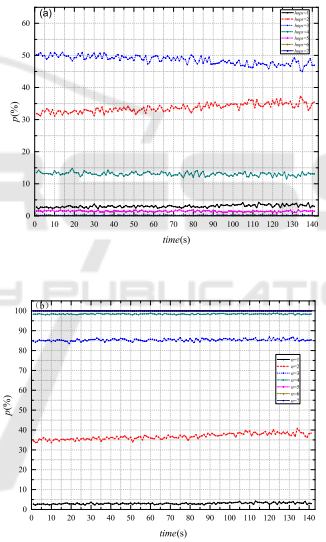


Figure 8: Decryption percentage p vs. time t in Hamsterster.

each group, every vehicle had a trajectory and a social network account. Then, we set the communication radius as 300 m. We changed t and observed the change in p. According to the results shown in Figs. 6, 7, and 8, time also does not affect p. Therefore, we can conclude that r and time do not affect p. We provide a proof.

Given N vehicles and corresponding random mapping social network accounts and their trajectories, assuming vehicles have a random distribution in the map, vehicle density is $\rho(vehicles/m^2)$, and communication radius is r, then r and time t do not affect the decryption percentage p.

Proof. One vehicle A is located in *Location*; then, the number of A's geographical neighbors is $m = h(Location(time), r, \rho)$, where $h()$ is a function to calculate the number of vehicles and *Location* changes according to time. Since the hops distribution from every other vehicle to A in the social network is fixed after mapping, we can define it as $\vec{P} = (p_1, p_2 \ldots p_n), \sum_{i=1}^{n} p_i = 1$. All the distribution of other vehicles is random, and therefore, for these m vehicles, the mathematical expectation of their distribution number is $m \cdot \vec{P}$. The percentage of hops distribution is $m \cdot \vec{P}/m = \vec{P}$. Therefore, m does not affect hops distribution. In other words, the time and the commu-

nication radius do not affect decryption percentage p when vehicles are mapped to social network accounts one to one randomly. □

We can conclude that p is related only to the type of social network in random mapping and the sender can set a different value of n to achieve a different decryption percentage in our method. Therefore, the sender can set n based on the privacy level of the message and decide which individuals are trustworthy to decrypt the message. The decryption percentage can reach 100%. Therefore, our method is practical.

5 CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a new method for secure vehicular communications, called SNB-PPB, that does not require key exchange, in order to handle the frequently changing vehicular context. Our method uses social networks to facilitate message encryption and decryption. We explored all the factors that may affect the decryption percentage. We can conclude that parameters can be set to ensure that only the vehicles allowed by the sender can decrypt his/her encrypted messages. Moreover, we offered two types of decryption mode to suit different potential scenarios. We will do further researches on performance evaluation and improvement in the future.

ACKNOWLEDGMENT

This paper was partially supported by: National Nature Science Foundation, Grant/Award Number: 61373123, 61572229 and U1564211; Scientific Research Foundation for Returned Scholars; International Scholar Exchange Fellowship (ISEF) program of Korea Foundation for Advanced Studies (KFAS); Jilin Provincial Science and Technology Development Foundation, Grant/Award Number: 20170204074GX; Jilin Provincial International Cooperation Foundation, Grant/Award Number: 20150414004GH; Premier-Discipline Enhancement Scheme supported by Zhuhai Government; Premier Key-Discipline Enhancement Scheme supported Guangdong Government Funds.

REFERENCES

Alsa'deh, A., Meinel, C., Westphal, F., Gawron, M., and Groneberg (2013). Cga integration into ipsec/ikev2 authentication. In *Proceedings of the 6th International Conference on Security of Information and Networks*. ACM.

Bethencourt, J., Sahai, A., and Waters, B. (2007). Ciphertext-policy attribute-based encryption. In *Security and Privacy, 2007. SP'07. IEEE Symposium on*. IEEE.

Choi, J. and Jung, S. (2009). A security framework with strong non-repudiation and privacy in vanets. In *2009 6th IEEE Consumer Communications and Networking Conference*. IEEE.

Goyal, V., Pandey, O., Sahai, A., and Waters, B. (2006). Attribute-based encryption for fine-grained access control of encrypted data. In *Proceedings of the 13th ACM conference on Computer and communications security*. Acm.

Horng, W.-B., Lee, C.-P., and Peng, J.-W. (2012). Privacy preservation in secure group communications for vehicular ad hoc networks. *Telecommunication Systems*.

Ingelrest, F. and Simplot-Ryl, David, I. (2006). Optimal transmission radius for energy efficient broadcasting protocols in ad hoc and sensor networks. *Parallel and Distributed Systems, IEEE Transactions on*.

Liu, X., Shan, Z., Zhang, L., Ye, W., and Yan, R. (2016). An efficient message access quality model in vehicular communication networks. *Signal Processing*.

Wang, Y. and Li, F. (2009). Vehicular ad hoc networks. In *Guide to wireless ad hoc networks*. Springer.

Xu, J., Liu, Y., Wang, J., Deng, W., and Ernst, T. (2015). Vike: vehicular ike for context-awareness. *Wireless Networks*.

Zhang, L. and Tu, W. (2009). Six degrees of separation in online society.