Cooperative Anonymity Authentication in Vehicular Networks

Jianmin Chen and Jie Wu
Department of Computer Science and Engineering
Florida Atlantic University
Boca Raton, FL 33431
Email: jchen8@fau.edu, jie@cse.fau.edu

Abstract

Data privacy in Vehicular Ad hoc Networks (VANETs) is a practical issue currently under investigation. Privacy-preserving anonymity authentication in networks is a challenging topic related to security, anonymity, network, and data privacy. Existing anonymity authentication in VANET is based on a k-anonymity model. An On Board Unit (OBU) chooses an adaptive anonymity group size in a request to an Road Site Unit (RSU). Upon response from an RSU, an OBU verifies the RSU's work probabilistically. But k-anonymity model group selection may leak information due to absence of diversity in the sensitive attribute, and an RSU lacks a rational strategy to adapt to an OBU's adaptive and probabilistic strategies. In this paper, we address anonymity authentication cooperative issues. Specifically, an RSU uses Cryptographic Client PUZZLE (CCPUZZLE) to enforce its maximum anonymity group's recommendation, and countermeasure Denial-of-Service (DoS) attacks. Cooperative design also relaxes restraints and supports anonymity group formation accommodating various anonymity models. In this paper, we clearly demonstrate that CCPUZZLE brings an OBU's cooperation and also efficiently countermeasures DoS attacks.

1. Introduction

With the rapid research and development of wireless communication technologies in recent years, more and more research has been done on the application of road-side vehicular communication in order to improve driver safety and traffic management, and expand the potential for Internet service.

On Board Units (OBUs) are registered with a vehicle, and Road Side Units (RSUs) are devices to provide road-side vehicular communication for commercial or government traffic management purposes. Vehicles can communicate with each other through OBUs as well as RSUs. The network formed among OBUs and RSUs is called a Vehicular Ad-hoc NETwork (VANET).

In a VANET, OBUs want to communicate with RSUs to get services, but sometimes OBUs must first be authenticated by an RSU. Research is also conducted to add privacy to the authentication process. In detail, anonymity authentication protocols [1][2][3] are employed when a user is authenticated anonymously in a group.

In practice, public key encryption required for anonymity authentication between an OBU and an RSU is expensive. Previous protocols [2][3] focus on an OBU's strategies to save time and computational cost. Therefore, via protocol extension, an OBU can customize its needs and choose RSU request anonymity group size adaptively. An OBU can then verify the RSU's work probabilistically after the RSU processes the anonymity authentication request. However, an RSU lacks a rational strategy to adapt to an OBU's adaptive and probabilistic strategies. An RSU is exposed to a malicious OBU's Denial-of-Service (DoS) attacks and further lacks a strategy to enforce OBU's cooperation in anonymity authentication service. An OBU can freely choose a big integer number as an anonymity group size and a small value as a verification probability without consideration of network status and an RSU's advice, since an OBU has zero trust of an RSU.

This paper analyzes the strategies RSUs and OBUs should use in anonymity authentication, and provides a cooperative anonymity authentication protocol based on the verifiable common secret encoding [1]. The authors consider cooperation from OBUs as an important factor in anonymity authentication. Equally important is data scalability through a distributed application across three tiers: terminal client (OBU), server (RSU), and corporate server (application server) in Figure (1).

The major contributions of this paper are threefold. First, we apply Cryptographic Client Puzzles (CCPUZZLE) [4] to enforce an OBU's cooperation and to countermeasure DoS attacks. Second, we propose an RSU to recommend a maximum anonymity group size to adapt to an OBU's adaptive and probabilistic strategies. Third, we propose an OBU to form an anonymity group according to its anonymity model's selection.

The remainder of this paper is organized as follows. Section 2 presents privacy-preservation authentication in a VANET. Section 3 further explains protocols extension in a VANET and its value. In Section 4, we evaluate the protocol

978-1-4244-5113-5/09/$25.00©2009 IEEE

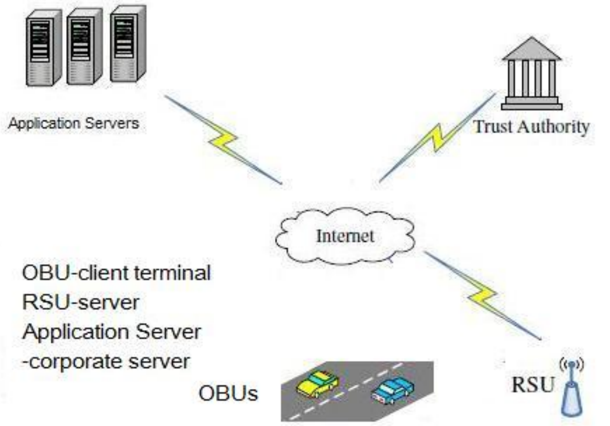

Figure 1. The model of VANET, 3-tier application: Application Server - Server corporate version, RSU - Server, OBU - Client terminal

basis and compare it with previous similar work. Section 5 introduces related work. Finally, Section 6 concludes this work.

2. Anonymity authentication in a VANET

A VANET, as seen in Figure 1, describes a two-layer vehicular network model. The lower layer is composed of vehicles and RSUs. An OBU has a unique public key and a corresponding private key. The communication among an RSU and an OBU is based on the Dedicated Short-Range Communication Protocol (DSRC). The upper layer is comprised of APplication Servers (APS), a Trust Authority (TA) and RSUs. RSUs communicate with APS using secure transmission protocols. APS may be dedicated to one service only. The secure transmission protocol, as an example, can be the wired Transport Layer Security (TLS) protocol. Figure 1 is the model of a VANET.

Next we present how we achieve our goal using anonymity authentication protocol. The protocol is based on previous protocols [2][3], but we make it cooperative to solve challenging problems in a VANET.

2.1. Key management and group formation

In our protocol extension, we choose a special linked hash map instead of a binary tree [2][3] to store network member information. Our goal of this design change is to accommodate anonymity group selection for various anonymity models, speed to search anonymity group members, and concern of a VANET's scalability.

An APS provides all network members information. We choose to use a special Java HashMap data structure, called LinkedHashmap [5].

Group Linked Hash Map, G_{LHM}: Linked hash map is a hash table and a linked list implementation of the Java Map interface with predictable iteration order. The implementation maintains a double-linked list running through all of its entries. This linked list defines the iteration ordering, which can be the insertion order.

Each network member, an OBU, has a pair of public/private keys. We insert OBU's information into a G_{LHM}. As we know, G_{LHM} has key/value pair for each map entry. The key of G_{LHM} is an OBU's unique public key. The value object of G_{LHM} includes the following six variables: (a) the member insertion order, which is a counter starting from 0 incremented by 1, denoted as j, (b) a group version, (c) time registered, (d) time revoked, (e) a bit data type storing validation information, which is zero bit if an OBU's public key is revoked, and (f) an OBU's registered data sharable.

After initialization, all the keys in the group are organized to a hash map, and a member's insertion order is kept as an index of group member. Each member (index j) can choose an index number inx and anonymity group size, g ($inx \leq j \leq inx + g$, if there is no revoked member yet) to represent an anonymity group - a subset.

An OBU may have to choose an anonymity group member from a large scale VANET. In that case, after network initialization, each OBU may only need to install a portion of a G_{LHM} to save the storage and transfer cost. Moreover, group data can be loaded into memory, to expedite the verification process (compared with loading data from a file on the hard disk).

In our protocol, an OBU still keeps an adaptive anonymity group size and probabilistic verification strategies from previous protocols [2][3]. As a result, we introduce a strategy for an RSU, which is to recommend a maximum anonymity group size, and to ensure an OBU's cooperation by implementing a CCPUZZLE technique in the protocol.

We can also assume that membership updates are infrequent based on stolen vehicle statistics in the United States. If a member is revoked, the G_{LHM} entry is updated with a bit of validation of public key. The order of G_{LHM} is not affected.

When a new member joins the group, it will be put into the G_{LHM}, and the insertion index increases by one. It takes constant time to search for a key value in a hashmap. A linked list is used to avoid transferring a group of public keys, especially if the group size is big and group members stay together in a linked list.

2.2. Protocol description

Our protocol is based on the previous work [1][2]. We understand privacy is a challenging issue in a VANET, and previous protocols have weaknesses. Therefore, we look into a cooperative solution in anonymity authentication. We avoid the prisoner dilemma effect [6] problem, in which OBUs and

RSU cannot cooperate to reach optimal solution for both. Moreover, we allow an RSU to give a client puzzle [4] to enforce cooperation from an OBU and countermeasure attacks from malicious OBUs.

There are six steps. Steps three and four are optional, depending on whether CCPUZZLE is required as determined by an RSU's analysis.

(1): RSU→OBU: $Cert(Pub_s, recGrpSize, timeout)$.

In the first step, an RSU broadcasts the message periodically with its certificate, recommended group size $recGrpSize$, and time out value $timeout$ for anonymity authentication request, in which Pub_s is a public key of an RSU.

Step 2 has two versions depending on whether or not random selection of an anonymity group is used.

(2)(a): OBU→RSU:
$Pub_s(true, inx, g, T_1, K_{session}, optional)$.

In this step, an OBU constructs an encrypted message using an RSU's public key Pub_s to allow an RSU to decrypt the message, and sends the encrypted message to an RSU.

The message has six parameters. A subset of the group uniquely identifying an anonymity group in G_{LHM}, current time T_1, anonymity group member starting index inx, anonymity group size g, and a session key $K_{session}$. Parameter $optional$ can hold some value, e.g., a probability value used for verification by an OBU.

(2)(b): OBU→RSU:
$Pub_s(false, cusGroup, T_1, K_{session})$.

In this step $(2)b$, an OBU customizes anonymity group by randomly choosing members from the group using an index number. So, a customized anonymity group is formed by randomly selection of n number used for a collection of $(inx_1, inx_2, ..., inx_n)$, in which inx_i is an index value in G_{LHM}. But $cusGroup$ is identified furthered by n public key associated with inx_i, $(Pub_1, Pub_2, ..., Pub_n)$. An OBU can also choose authentication using its public key only, which has no anonymity.

(3) RSU→OBU: $K_{session}(clientPuzzle, extInfo)$.

In this step 3, an RSU will determine whether or not to require a client puzzle solution from an OBU depending on the OBU's request and network status.

An RSU will attach a cryptographic puzzle to an OBU and require the solution to the puzzle to be attached in the reply before the RSU processes time consuming anonymity authentication Verifiable Common Secret(VCS). An RSU can also send out a puzzle if the anonymity group is bigger than an RSU's recommendation in step 1. Parameter $extInfo$ can be simplified as a value of 0 or 1 to communicate the RSU's decision: 0 for busy traffic and 1 for anonymity group size bigger than $recGrpSize$.

(4) OBU→RSU: $K_{session}(clientPuzzleSolution)$.

An OBU sends out the client puzzle solution to an RSU and waits for its anonymity authentication. Typically, solving a client puzzle requires a brute-force search in the solution space, while solution verification is trivial.

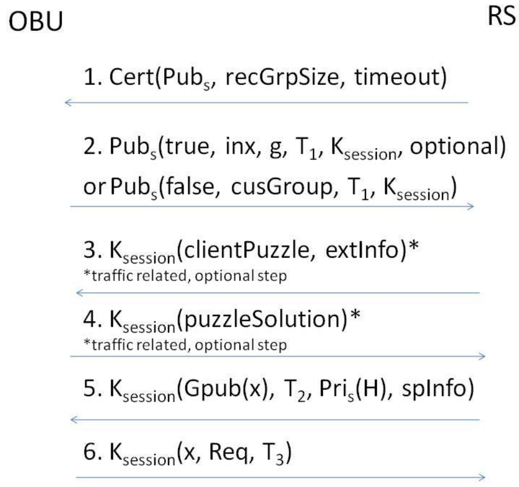
OBU
RSU
1. Cert(Pub_s, recGrpSize, timeout)
2. Pub_s(true, inx, g, T_1, $K_{session}$, optional)
or Pub_s(false, cusGroup, T_1, $K_{session}$)
3. $K_{session}$(clientPuzzle, extInfo)*
*traffic related, optional step
4. $K_{session}$(puzzleSolution)*
*traffic related, optional step
5. $K_{session}$(Gpub(x), T_2, Pri_s(H), spInfo)
6. $K_{session}$(x, Req, T_3)

Figure 2. Group-based authentication protocol in VANET.

(5) RSU→OBU:
$K_{session}(GPub(x), T_2, Pri_s(H), spInfo)$.

The fifth step is for an RSU to give back a common shared secret encrypted with a session key. An RSU constructs the verifiable common secret for the group members with a random value x, its current time T_2, its current group version V_G, the signature $Pri_s(H)$ obtained through encrypting the digest message (MAC) of $GPub$(x) using hash function H with its private key Pri_s, and server side information $spInfo$.

(6) OBU→RSU: $K_{session}(x, Req, T_3)$.

An OBU decrypts x from the verifiable common secret and verifies its anonymity. Upon successful decryption and verification, it constructs a reply message with x, service request Req, and its current time T_3. The message is encrypted with the session key $K_{session}$.

Figure 2 shows group-based authentication protocol.

3. Challenges and solutions in a VANET

In this section, we present anonymity authentication challenges in a VANET. Meanwhile, we further discuss how we tackle challenging problems through detail of the protocol from the previous section and some techniques even outside the protocol design.

Previous work [2][3] of anonymity authentication in VANET is based on a zero trust model in which an OBU does not trust an RSU. Therefore, an OBU will give up on the anonymity authentication process if an RSU is found to be making server probing attacks. In detail, an RSU tries to reduce anonymity group size by breaking the integrity of

VCS. An RSU acts maliciously to break the privacy of an OBU.

3.1. Cooperative mechanism

The cooperative mechanism is the most important design work we have for anonymity authentication in a VANET. How can we efficiently bring the mechanism into protocol design and force an OBU to cooperate with a RSU rationally and help to sustain anonymity authentication?

Cooperative designs have two stages. An RSU should first give a reasonable Maximum anonymity Group SIZE (MGSIZE) to limit any OBU's choice of adaptive anonymity group size in a VANET. Otherwise, an RSU will defend its decision by sending an OBU a Cryptographic Client PUZZLE [4] (CCPUZZLE) if an OBU's anonymity group size is bigger than MGSIZE. CCPUZZLE is an easy and quick way for an RSU to provide a fairness mechanism in the competition among OBUs. Therefore, CCPUZZLE brings cooperation among OBUs.

DoS attacks from a malicious OBU is possible in VANET. A malicious OBU can continuously send anonymity authentication requests in a limited time period and block an RSU's service to other OBUs, similar to the TCP SYN flooding attack. Second, without a maximum limit of anonymity group size, an OBU will not hesitate to seek a big anonymity group. Without commitment, after an RSU processes the request, an OBU may choose a small sample to verify or even abandon the response.

Other mechanisms can also bring an OBU's cooperation, but their implementation may not be as convenient as a CCPUZZLE implementation. An RSU can make an inference from an authentication request and its location, even though the requestor is anonymous. However, an OBU's mobile status makes it difficult to associate with one OBU with different positions.

We propose MGSIZE to bring cooperation of OBUs. Under MGSIZE, an OBU can adaptively choose an anonymity group size. The idea is to let an OBU configure a terminal client application, an RSU configure server application. Both applications can boot up with a default configuration and also have a configuration interface for a user to manage the settings.

3.2. Data privacy in a VANET

Data privacy in a VANET varies. There are numerous privacy concerns, such as legal enforcement or insurance policy. Moreover, different users (OBUs) may have varying privacy needs in different contexts, and the same user may require different levels of privacy at different times.

Using location privacy threats as an example, an RSU may be an adversary. An RSU obtains an OBU's location information by authentication. An RSU draws inferences from the OBU's time and frequency of passing certain RSUs. Thus, we have to help a user to find a comfortable balance, which is between the extreme of fully disclosed and completely withheld driving history through an anonymity authentication process.

As another example in a VANET, an OBU's route may be a privacy concern. An OBU chooses to use an m-invariant model [7] to choose k users instead of a k-anonymity model. In detail, given an example of m-invariant model and location privacy in a VANET [8], in order to control the number of alternative routes, a mobile user (OBU) would have difficulty maintaining anonymity in situations where all k users are traveling along the same route segments which pass through the same set of identifiable RSUs. It passes a k-anonymity model test, but has a low m value compared to the value k.

Therefore, we will consider various anonymity models.

3.3. Short review of anonymity models

In protocol extension to a VANET, a k-anonymity model was selected in protocols [2][3], in which an OBU ensures that it forms a group with a size k big enough for the privacy concern. The extended protocol mainly focuses on an RSU server probing attack, which is to break anonymity group size k. In this protocol, we extend the protocol in a VANET considering that an RSU can launch more efficient data analysis instead of a k-anonymity group size as seen in all data privacy models analysis. Since more data is available to public or anonymity preserving published in a VANET, an RSU can launch more efficient attacks than server probing attacks. Here we go through several models and discuss the techniques that can be extended in this protocol. In general, we advise a technique to let an OBU look into anonymity-preserving publishing data (seen in Table 1) and carefully build a SQL query to choose anonymity group members according to its privacy concern.

l-diversity model [7]: The k-anonymity can create groups that leak information due to absence of diversity in the sensitive attribute. The l-diversity model is dealing with sensitive attributes diversity.

t-closeness model [7] : The t-closeness model, as a privacy notion, requires that the distribution of a sensitive attribute in any equivalence class be close to the distribution of the attribute in the overall table i.e., the distance between the two distributions should be no more than a threshold t.

Personalized privacy preservation model [7]: The problem in anonymity design is that we may be offering insufficient protection to a subset of people, while applying excessive privacy control to another subset.

Background knowledge attack model [7]: Several anonymity models are related to background knowledge and inference attacks.

4. Analysis and evaluation

In this section we analyze the anonymity authentication protocols. Our test computer was a $2.0GHz$ Intel Core Duo CPU with $1.50GB$ of RAM running Windows Vista. Using Java platform standard edition 6.0's cryptography library RSA algorithm, the computer is able to complete $1,698$ public key encryptions per second. In other words, the test computer takes $0.58ms$ to complete an RSA public key encryption.

4.1. Experiment with client puzzles

Client puzzles are a viable method for protecting SSL servers from SSL based DoS attacks in a client puzzle extension to the TLS protocol against DoS attacks through the work showed in paper [9]; and that work showed that a client puzzle can efficiently prevent a DoS attack if the puzzle size is selected properly.

Client puzzles in our protocol play a defensive role to countermeasure against DoS attacks, and force an OBU to select the anonymity group size properly. As seen in work [9], we omit the detail of client puzzle states enter/exit and experiment with/without client puzzle during the attack.

To support our anonymity authentication design, an RSU and an OBU have to configure parameters accordingly. Several parameters have to be decided: (1) a hash function and (2) a client puzzle size.

We choose to use a hash function MD5 to evaluate the protocol related to client puzzles feature. For hash function $h(y)$, a client puzzle is the triple $(n, y', h(y))$, where y' is y with its n lowest bits set to 0. The solution to the puzzle is the full value of y. The best way for a client to generate y is to exhaust the possible values of the n lowest bits. This should take 2^{n-1} calculation of $h(y)$ on average. The test computer spends $629ms$ to complete 1 million calculations of the MD5 hash function. The RSU server, on the other hand, needs to generate a random block (for MD5, 512 bits) of data, and evaluates the hash function twice.

It takes the test computer $629ms$ to complete 1 million MD5 hash function calculations. One million is in the range between 2^{19} and 2^{20}. While $1,000$ RSA public key encryptions require a test computer $580ms$ to complete. For example, given MGSIZE value of 100, and that an OBU sends out anonymity authentication with a group size more than 100, the RSU can give the OBU a client puzzle with $n = 10$ to solve. Therefore, the RSU forces the OBU to spend time solving a client puzzle, giving the RSU time to serve other OBUs. After the RSU verifies the OBU's client puzzle solution, the RSU calculates 100 RSA public key encryptions. Also the RSU could easily avoid continuously processing requests from any malicious OBU, since any malicious OBU has to finish client puzzles accordingly. Based on the fact that the client puzzle size n is linearly proportional to the RSA group size, we conclude that the client puzzles could stop attackers but will not disrupt an OBU's client operations.

The protocol can also be extended in different ways. One idea is to allow any OBU to bid for resources by tuning the difficulty of puzzles it solves called *Puzzle Auctions* [10].

4.2. Group selection and anonymity-preserving publishing

The existing protocol design [2] uses a complete binary tree over the ordered list, in which the public keys are stored only in leaves of the tree and internal nodes are used to identify each subtree. The anonymity group can be identified using a subtree root and the protocol did provide flexible subgroup organization. However, two nodes widely separated in the ordered list cannot belong to the same group unless the group includes all the nodes in the connecting path. We choose to keep the ordered list, but also we avoid the slow search speed a list. The double linked list of HashMap G_{LHM} also provides quick search of a node using constant time.

In a large scale VANET, subscriber's information may be public or anonymity-preserving published [11]. It may cost time and storage for RSU/OBU to pre-install all members information. An OBU chooses to select multiple result sets through queries on a corporate server database, and later it can use those result sets to form a different anonymity group for authentication purposes.

Example 1: An OBU is registered as Andy who has access to the following information:

Table 1. partial privacy-preserving publication OBU's User table

Public Key	Age	Zip	Color	Model	Year	Seq
***	17	12k	white	Ford	1996	1
***	19	13k	blue	Ford	2008	2
***	20	14k	red	GM	2007	3
***	24	12k	green	Chrysler	2006	4
***	29	12k	black	Toyota	2003	5
***	34	12k	purple	Hyundai	2007	6
***	45	39k	white	Ford	2009	7

In this example, Andy has several privacy preferences for preparing anonymity group data sets. Zipcode is already generalized and he is OK with releasing his zipcode. Therefore, he may build a SQL statement "**Select** * **from** User **where** Zip = '12k' **order by** Auto Year **desc** ". The SQL query is to select anonymity group members. The query returns members with the same zip code. The result set with zip code $12k$ satisfies a k-anonymity model with ($k \geq 4$). He also avoids an old car which is sensitive since an RSU predicts possible breakdown on the road. Result set is 6, 4, 5, 1 expressed in Seq column value. If k-anonymity group size is 3, then the first three 6, 4, 5 are selected. The node chooses $k-1$ node instead of k nodes including itself in a k-anonymity model. On a large scale, an OBU may customize multiple queries for its diversified privacy concerns to select hundreds of records from millions of records for group formation.

In terms of RSU's searching for a public key, the previous protocol [2] has actually $O(n)$ computation cost, but our mechanism provides $O(1)$ time to search for a member and also has much more flexibility in group formation.

4.3. Randomly selected group members and various anonymity models

In a large scale VANET, an APS may have over one million OBUs. Search and maintainance computation cost is an important concern in an anonymity group using random selection of group members. In our protocol, we use a HashMap to provide constant time to search for a public key. Therefore, the time to form a randomly selected group is linear to the size of an anonymity group. Since members are chosen with queries using customized criteria, we could accommodate various models.

5. Related Work

The design of anonymity authentication protocol in a VANET is challenging work. There are different focuses. Related work is widely spread: anonymity in network [12], data privacy and models, anonymity authentication, wireless ad hoc network (e.g. VANET), and cryptographic techniques etc.

A special anonymity authentication mechanism only works for the first k times of authentication [13], which may be used to prevent user misbehavior, such as DoS attacks if k is small enough. Besides [2][3], a k-anonymity model is used for location privacy [14]. Ren proposed a privacy preserving authentication in [15] that uses blind signature and a one-way hash chain cryptographic technique to keep privacy. Calandriello [16] proposed a pseudonym-based protocol that uses group signature as the mechanism for a vehicle to generate its own pseudonyms. Sun [17] used a group signature and identity-based signature scheme, which guarantees security and anonymity. In detail, he applied group signature to vehicle-to-vehicle authentication and identity-based signature to vehicle-to-infrastructure authentication.

Different models and algorithms in the privacy-preserving data mining area [7] and anonymity in network [12] both deal with anonymity, but from different angles. A k-anonymity model is widely used in anonymity networks. DC-Net, Mix-Net, and other numerous anonymity network schemes are based on a k-anonymity model. Anonymity in the network communication context is to hide sender and receiver, and message-among-hops correlation. In order to reach target k-anonymity group size, common techniques include using time delays when sending packets, flushing using time or threshold of k messages or both, and using dummy message to prevent traffic analysis.

6. Conclusion

In this paper, cooperative consideration in anonymity authentication in a VANET leads us to a new design. An RSU chooses a cryptographic client puzzle to force OBUs to work cooperatively. CCPUZZLE also help an RSU to countermeasure attacks from malicious OBUs. Our cooperative design can accommodate various anonymity models in anonymity authentication and deal with the scalability problem of large VANETs.

References

[1] S. Schecheter, T. Parnell, and A. Hartemink. Anonymous authentication of membership in dyanmic group. January 1999.

[2] K. Sha, Y. Xi, W. Shi, L. Schwiebert, and T. Zhang. Adaptive privacy-preserving authentication in vehicular networks. In *Proceedings of IEEE International Workshop on Vehicle Communication and Applications, pp.1-8*, 2006.

[3] Y. Xi, K. Sha, W. Shi, L. Schwiebert, and T. Zhang. Probabilistic adaptive anonymous authentication in vehicular networks. In *Journal of Computer Science and Technology*, Nov. 2008.

[4] A. Juels and J. Brainard. Client puzzles: A cryptographic countermeasure against connection depletion attacks. In *6th NDSS*, 1999.

[5] Sun developer network. http://java.sun.com/javase/index.jsp.

[6] V. Srinivasan, P. Nuggehalli, C. Chiasserini, and R. Rao. Cooperation in wireless ad hoc networks. 2003.

[7] C. Aggarwal and P. Yu. In *Privacy-Preserving Data Ming Models and Algorithms*, Springer LLC, 2008.

[8] L. Liu. From data privacy to location privacy: Models and algorithms. In *VLDB, pages 1429-1430. ACM.*, 2007.

[9] D. Dean and A. Stubblefield. Using clients puzzles to protect tls. In *Proceedings of 10th Annual USENIX Security Symposium*, 2001.

[10] X. Wang and M. Reiter. Defending against denial-of-service attacks with puzzle auctions. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy (SP'03)*, 2003.

[11] Y. Tao J. Li and X. Xiao. Preservation of proximity privacy in publishing numerical sensitive data. In *SIGMOD'08*, 2008.

[12] Free haven's selected papers in anonymity. In *http://www.freehaven.net/anonbib*.

[13] L. Nguyen and R. Safavi-Naini. Dynamic k-times anonymous authentication. In *ACNS, pages 318-333*, 2005.

[14] B. Gedik and L. liu. Location privacy in mobile systems: A personalized anonymization model. In *Proc. the 25th International Conference on Distributed Computing Systems*, 2005.

[15] Ren K etc al. A novel privacy preserving authentication and access control scheme for pervasive computing environments. In *IEEE Transaction on Vehicular Technology*, 2006.

[16] G. Calandriello, P. Papadimitratos, A. Lloy, and J. Hubaux. Efficient and robust pseudonymous authentication in vanet. In *Proc. the Fourth ACM International Workshop on Vehicular Ad Hoc Networks*, 2007.

[17] X. Sun, X. Lin, and P. Ho. Secure vehicular communications based on group signature and id-based signature scheme. 2007.