Journal of Information Security, 2012, 3, 59-67
http://dx.doi.org/10.4236/jis.2012.31007 Published Online January 2012 (http://www.SciRP.org/journal/jis)

SPCS: Secure and Privacy-Preserving Charging-Station Searching Using VANET

Tat Wing Chim[1*], Jeanno Chin Long Cheung[1], Siu Ming Yiu[1], Lucas Chi Kwong Hui[1], Victor On Kwok Li[2]
[1]Department of Computer Science, The University of Hong Kong, Hong Kong, China
[2]Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong, China
Email: {*twchim, ljcheung, smyiu, hui}@cs.hku.hk, vli@eee.hku.hk

Received November 5, 2011; revised December 24, 2011; accepted January 3, 2012

ABSTRACT

Electric vehicle has attracted more and more attention all around the world in recent years because of its many advantages such as low pollution to the environment. However, due to the limitation of current technology, charging remains an important issue. In this paper, we study the problem of finding and making reservation on charging stations via a vehicular ad hoc network (VANET). Our focus is on the privacy concern as drivers would not like to be traced by knowing which charging stations they have visited. Technically, we make use of the property of blind signature to achieve this goal. In brief, an electric vehicle first generates a set of anonymous credentials on its own. A trusted authority then blindly signs on them after verifying the identity of the vehicle. After that, the vehicle can make charging station searching queries and reservations by presenting those signed anonymous credentials. We implemented the scheme and show that the credential signing process (expected to be the most time consuming step) can be completed within reasonable time when the parameters are properly set. In particular, the process can be completed in 5 minutes when 1024 bits of RSA signing key is used. Moreover, we show that our scheme is secure in terms of authentication and privacy-preserving.

Keywords: Electric Vehicle; Vehicular Ad Hoc Network; Blind Signature; Anonymous Credential

1. Introduction

Electric Vehicle (EV) has attracted a lot of attention all around the world in recent years due to its many advantages such as low noise, low energy consumption and low pollution. However, the distance that can be travelled by an electric vehicle with onboard battery fully charged is usually shorter than that can be travelled by a conventional fossil fuel powered vehicle with the gas tank fully filled. Thus a driver has to search in advance the charging stations it has to visit in order for it to have enough power to complete the whole journey. On the other hand, since charging up a battery usually take much longer time than filling up a fossil fuel tank, long queues at charging stations may be resulted. Thus a reservation mechanism is desirable.

Before continuing our discussion, let us briefly talk about what a VANET is. VANET is an important element of the Intelligent Transportation Systems (ITSs) [1]. In a typical VANET, each vehicle is assumed to have an on-board unit (OBU) and there are road-side units (RSU) installed along the roads. A trusted authority (TA) and maybe some other application servers are installed in the backend. The OBUs and RSUs communicate using the Dedicated Short Range Communications (DSRC) protocol [2] over the wireless channel while the RSUs, TA, and the application servers communicate using a secure fixed network (e.g. the Internet). The basic application of a VANET is to allow arbitrary vehicles to broadcast safety messages (e.g. about vehicle speed, turning direction, road condition, traffic accident information) to other nearby vehicles (denoted as vehicle-vehicle or V2V communications) and to RSU (denoted as vehicle-infrastructure or V2I communications) regularly such that other vehicles may adjust their travelling routes and RSUs may inform the traffic control center to adjust traffic lights for avoiding possible traffic congestion. As such, a VANET can also be interpreted as a sensor network because the traffic control center or some other central servers can collect lots of useful information about road conditions from vehicles.

In this paper, we study the online charging station searching and reservation problem. We mainly focus on the security and privacy issue of the problem as most drivers may not want other to track his route down by

*Corresponding author.

learning which charging stations he visited. Based on the source and destination of a journey, the system can automatically search and suggest a set of charging stations for the driver. Reservations can also be made accordingly. Our scheme addresses two major security problems-authentication and privacy-preservation. In terms of authentication, a driver has to be properly authenticcated before it can use the searching and reservation service. This can help to avoid an attacker from making too many requests and unnecessary reservations, which in turn blocks normal users from using the service. On the other hand, it makes more sense for the searching and reservation service to be paid ones. Otherwise, a driver may simply make reservations at all charging stations around the city so that he/she can change his/her mind about where to go at the very last minute. Thus authenticcation is essential. In terms of privacy-preservation, a driver may not want others to know his/her travelling source, destination and even the route. We adopt the technique of blind signature [3] to facilitate a driver to obtain a valid anonymous credential. Then the driver can make searching and reservation queries as well as get his/her vehicle charged up at charging stations by presenting the anonymous credential. We use the term "anonymous" here because the credential can prove that the holder has been properly authenticated but no one can deduce the identity of the driver based only on the credential itself.

Privacy-preservation can then be achieved.

We provide a security analysis and we implemented the scheme to evaluate our scheme. Throughout our scheme, the most time-consuming part should be the credential signing process by the trusted authority. It may even become a bottleneck if lots of vehicles request it for signing operation at about the same time. Through our study, we show that such a credential signing process can be completed in reasonable time when the parameters are properly set. The process takes about 5 minutes when 1024 bits of RSA signing key is used.

The rest of the paper is organized as follows: related work is reviewed in Section 2. The system model, charging model, assumptions and security requirements are listed in Section 3. Our schemes are described in details in Section 4. The analysis and evaluation of our schemes are given in Sections 5-7. Finally, Section 8 concludes the paper.

2. Related Work

The idea of searching charging stations using VANET is new. However, there are some existing similar VANET applications published in recent years. For example, in [4], a VANET-based smart parking scheme for large parking lots was proposed. Though they also consider security, there are a number of differences between their scheme and ours. First, their scheme is of a small scale which covers a car park while ours is large scale to cover the whole city and beyond. Second, in their scheme a car park is monitored by three RSUs which take up the roles of determining a vehicle's location, searching for a vacant parking space and providing navigation service to guide the vehicle to go from the car park entrance to the selected parking space. In our scheme, the road system is covered by a large number of charging stations which report their status in a distributed manner. Third, our scheme allows one's identity and charging station searching query to be delinked. This feature is only interesting for wide area searching like ours. Forth, our scheme allows a driver to make a reservation at a charging station. Again, this feature is only interesting for a wide area setting like ours. Thus, the scheme provided in [4] cannot be used to solve the charging station searching problem discussed in this paper.

Other interesting VANET application recently published include [5,6]. In [5], an interesting problem about providing querying service using VANET while ensuring that queries will not be linkable to the queriers was discussed. A solution for solving the problem by using techniques of pseudo-identity, indistinguishable credentials, and oblivious transfer was then proposed. Although it also provides a way for a vehicle to anonymously obtain credentials, all credentials are assumed to be identical and so the approach cannot be adopted into our SPCS scheme in which credentials carry different times of issuance and values. In [6], a new real-time urban monitoring system using the Localizing and Handling Network Event Systems (LocHNESs) platform developed by Telecom Italia was developed. Information about urban mobility in real time, ranging from traffic conditions to the movements of pedestrians throughout the city, can be collected efficiently. However, security issues were not mentioned in this work.

In terms of security, recent efforts related to the security issues in VANET include [7-13]. In [7], a batch verification scheme known as IBV was proposed for an RSU to verify a large number of signatures at the same time using only three pairing operations. In [8], an RSU-aided inter-vehicle communications scheme was proposed. A vehicle relies on an RSU to verify the signature of another vehicle. In [9], group communications in VANETs are considered and a group key update protocol was proposed. In [10], some security and privacy-enhancing communications schemes were proposed. Of particular interest, a group communications protocol was defined. After a simple handshaking with any RSU, a group of known vehicles can verify the signature of each other without any further support from RSUs. A common group secret is also developed for secure communica-

tions among group members. [11,12] also target at driver privacy preservation but instead of using pseudo identities, the concept of group signature is adopted. The signature of any vehicle can be verified by the same group key but the actual signer can only be traced by a trusted party. Though privacy can be preserved, these schemes are rather complicated and may not be practical. In [13], a threshold anonymous announcement service using direct anonymous attestation and one-time anonymous authentication to simultaneously achieve the goals of reliability, privacy and auditability was proposed. However, it focuses on inter-vehicle message authentication in general and is not related to querying.

Our scheme is based on the idea of indistinguishable (anonymous) credential. Such a credential system was introduced by Chaum [14]. The system allows a user to obtain a credential from one organization and later show the possession of the credential to another organization while the transactions at the two organizations are not linkable. The idea of anonymous credential has been adopted in different applications. For example, [15] proposes a credential-based privacy-preserving e-learning system under which a student can show his/her progress in e-learning without leaking his/her identity information.

3. Problem Statement

3.1. System Model

Our vehicular network consists of on-board units (OBUs) installed on vehicles, road-side units (RSUs) and charging stations (CSs) along the roads, a trusted authority (TA) and a querying server (QS) in the backend for key management and for answering charging station queries and making reservations at charging stations, respectively.

3.2. Charging Model

We assume that a client will be charged for the following instances:

- Making each charging station searching query: A driver will be charged when he/she makes charging station searching queries to the querying server. This is because the querying server operator needs to invest on facilities for the searching and scheduling service.
- Making reservation at a charging station: A driver will be charged when he/she makes reservations to charging stations and the charge is proportional to the length of the duration reserved. This is necessary because otherwise, a driver may tend to reserve a duration which is longer than necessary to ensure that he/she does not need to wait when he/she arrives at the charging station concerned.

Note that our system also supports different charging models. For example, an operator may want to charge a driver more if he/she makes reservations during peak hours.

3.3. Assumptions

We further assume the followings:

- The TA is trusted while the QS is only semi-trusted. To avoid being a single point of failure, redundant TAs and QSs which have identical functionalities are installed.
- TA, QS and RSUs communicate through a secure fixed network (e.g. Internet). Thus RSUs can help to relate messages from vehicles on the road to TA and QS.
- There exists a conventional public key infrastructure (PKI) for initial vehicle authentication. Each vehicle V_i having license plate number LP_i has a conventional public key VPK_i and a conventional private key VSK_i and is given a TA-signed certificate $VCert_i$ which contains CPK_i and LP_i. We will discuss details about the generation and verification of $Cert_i$ in Section 4.
- There also exists a conventional identity-based public key infrastructure (PKI) for authenticating TA, QS and CSs. The public keys of TA and QS are the same as their real identities, *TRID* and *QRID* respectively, and are known by *everyone*. Also any charging station CS_i broadcasts its public key which is the same as its real identity $CRID_i$ with hello messages periodically to vehicles that are travelling close to it. Thus $CRID_i$ is known by all vehicles nearby. The validity $CRID_i$ can be ensured using a certificate $CCert_i$ issued by the TA. We will discuss the details in Section 4.
- The real identity of any vehicle is only known by the TA and itself but not by others.
- We assume that there is a reasonably large number of charging station searching queries issued to QS. Otherwise, if there is only one query, the sender can be linked up with the query easily.
- Each vehicle has a tamper-proof device which is responsible for all cryptographic-related functions such as storage of keys, generation of pseudo identities, signing messages and encryption of messages (details will be given one by one in the next section). Also its output interface is limited and we will specify that in the appropriate places in the next section. Finally, it is assumed to have its own clock for generating correct time stamps and be able to run on its own battery. Note that a vehicle can also have a conventional computer device for performing the verification of RSUs' hop information (to speed up the process and details will be given in the next section). Since a tamper-proof device is always associated with a dedi-

cated vehicle, we will use these two terms interchangeably without prior notice throughout the paper.

3.4. Security Requirements

We aim at designing a scheme to provide VANET-based charging station searching and reservation service for electric vehicles. The scheme has to satisfy the following two security requirements:

- Message integrity and authentication: A vehicle should be authenticated before it can request for credential signing, issue a charging station searching query, make reservations to charging stations and receive charging service at charging stations. On the other hand, TA, QS and vehicles are able to verify that a message is indeed sent and signed by a certain party without being modified by anyone.
- Privacy-preserving: In this paper, the privacy of a driver is defined as his driving habit and travelling route. It can be preserved by two means. First, even if TA and QS collude, they cannot link up a vehicle's query (which contains the travelling source and destination) with its real identity. Thus no one can trace a driver's travelling route and driving habit easily. Second, a vehicle's query (which contains the travelling source and destination) as well as the list of charging stations returned by QS can not be eavesdropped by neighboring vehicles easily.

4. Our Solutions—SPCS

This section presents our Secure and Privacy-preserving Charging-station Searching (SPCS) scheme using VANET. Our scheme consists of five phases—setup phase, credential signature requesting phase, charging station searching and reservation phase, charging phase and reconciliation phase. Next we explain each of these phases in details.

4.1. Setup Phase

During system startup, the following steps will be carried out:

- TA assigns itself an identity *TRID* and a secret key *TSK* such that *TRID* and *TSK* form a public and private key pair. *TRID* is assumed to be known by everyone in the system.
- TA assigns QS an identity *QRID* and a secret key *QSK* such that *QRID* and *QSK* form a public and private key pair.
- For each charging station CS_i locating at CL_i an identity $CRID_i$ and a secret key CSK_i such that $CRID_i$ and CSK_i form a public and private key pair. To facilitate others to verify CS_i's identity, TA generates its certificate as $CCerti = \langle CRID_i,\ CL_i, TSIG_{TSK}(CRID_i \| CL_i) \rangle$ where $TSIG_{TSK}(CRID_i \| CL_i)$ is TA's signature on the concatenation of $CRID_i$ and CL_i. Note that any asymmetric signature scheme can be used in the above steps.
- During system setup or first registration of an electric vehicle V_i, TA assigns it a real identity $VRID_i$ and the tamper-proof device activation password $VPWD_i$. TA also assigns V_i a license plate number LP_i and generates a pair of conventional public and private keys, VPK_i and VSK_i respectively. TA then signs a certificate $VCert_i = \langle LP_i,\ VPK_i,\ TSIG_{TSK}(LP_i \| VPK_i) \rangle$ where $TSIG_{TSK}(LP_i \| VPK_i)$. TA preloads $VRID_i$, $VPWD_i$, LP_i, VSK_i and $VCert_i$ into the tamper-proof device on V_i.

Throughout the paper, conventional asymmetric and symmetric encryptions and signatures are used at some locations. To make the context concise, let us use the notations $AS\_ENC_x(M)$ and $S\_ENC_x(M), SIG_x(M)$ to denote asymmetrically encrypting, symmetrically encrypting and signing message M using the key x based on any asymmetric encryption, symmetric encryption and signature algorithms, respectively.

4.2. Credential Signature Requesting Phase

At the beginning of a charging period (say a month) or when a driver uses up all the credentials in hand, the driver triggers the tamper-proof device on his/her electric vehicle V_i to send the following credential signing request message to the TA:
$\langle Cred\_Sig\_Req, VCert_i, SIG_{VSKi}(Cred\_Sig\_Req) \rangle$.
TA verifies the signature $SIG_{VSKi}(Cred\_Sig\_Req)$ using V_i's public key VPK_i as stored in $VCert_i$. If successful, TA generate a random session key *SESS_KEY* and securely transmits it to V_i's tamper-proof device. That is the transmitting message becomes

$$\langle AS\_ENC_{VPKi}(SESS\_KEY),\ TSIG_{TSK}(AS\_ENC_{VPKi}(SESS\_KEY)) \rangle$$

Upon finishing verifying TA's signature with *TRID*, the tamperproof device stores *SESS_KEY* locally for ongoing communi- cations.

Next the tamper-proof device generates a set of credentials for the TA to sign. Each credential $Cred_k$ is of the form $\langle CN_k, TOI_k, Val_k, Sig_k \rangle$ where CN_k is a unique credential number (*i.e.* different credentials have different credential numbers), TOI_k indicates the time that the credential is issued, Val_k is the duration that the electric vehicle can get charged up by presenting that credential. Sample values include 5, 10 and 15 (minutes) and the value depends on the kind of electric vehicle as well as the capacity of the onboard battery. Since a driver also needs a credential for the charging station searching ser-

vice, such a credential carries a special value of 0. Sig_k represents the TA's signature on the first three fields and will be filled up later. For the credential of a certain value, the tamper-proof device generates N times more credentials than necessary (*i.e.* among every N credentials, only one credential will be used while the other (N – 1) credentials are only for checking purpose which we will describe in later context). For each credential, the tamper-proof device transforms the bit pattern of the concatenation $CN_k \| TOI_k \| Val_k$ into $CTV'_k = Trans(CN_k \| TOI_k \| Val_k)$. It then sends the transformed credentials altogether to the TA for blind signature. To avoid the credentials being stolen or tampered by eavesdroppers, all of them are encrypted symmetrically using *SESS_KEY* and signed using VSK_i before they are sent out.

For every N credentials with the same value, TA randomly challenges the tamper-proof device to open $(N-1)$ of them and checks whether they are of proper format (including whether any two of them have the same credential number, whether TOI_k is really close to the current time, whether all the $(N-1)$ credentials have the same value). If yes, TA blindly signs the remaining credential (*i.e.* the one that was not opened) and sends it back to the temper-proof device. Similar to the above, all communications are symmetrically encrypted using the session key *SESS_KEY* to resist against eavesdroppers. Also the messages from V_i and TA are signed using VSK_i and *TSK* respectively by the corresponding party.

After that, the tamper-proof device retrieves the actual signed credentials by removing the hiding factors. That is, for each credential, it de-transforms $SIG_{TSK}(CTV'_k)$ back into $SIG_{TSK}(CN_k \| TOI_k \| Val_k)$ by performing $SIG_{TSK}(CN_k \| TOI_k \| Val_k) = DTrans(SIG_{TSK}(CTV'_k))$. At this moment, the tamper-proof device can fill up all TA signatures in all credentials. The tamper-proof device stored all credentials locally.

Finally, TA records the total number of credentials which it has blindly signed and their values into its local database (to be used in reconciliation phase).

4.3. Charging Station Searching and Reservation Phase

When a driver starts up his/her electric vehicle V_i for a journey, he/she first enters the real identity $VRID_i$ and password $VPWD_i$ (assigned by TA in Section 4.2 above) into the tamper-proof device to activate it. Here only simple hardware checking is involved. If either the real identity or the password is, or both are incorrect, the tamper-proof device refuses to perform further operations. Otherwise, the tamper-proof device prompts the driver to enter the travelling source Src_i and destination $Dest_i$. The tamper-proof device then randomly picks up a credential $Cred_k$ with value 0 (*i.e.* those for searching purpose) from its pool, composes the message $AS\_ENC_{QRID}(Cred_k \| Src \| Dest)$ and sends it to the querying server (QS). Note that no signature is required here because the tamper-proof device will hide its real identity from now on.

Upon receiving the message from V_i, QS first decrypts the message using its private key *QSK*. It then checks whether the attached credential $Cred_k$ is a valid one by checking the validity of TA's signature using *TRID*. If it is valid, it also checks whether the same credential (same credential number and same time of issurance) has been used before. To achieve this goal, QS maintains a local database for storing the pair (CN_k, TOI_k) of all received credentials and the checking becomes a simple linear scanning. If so unluckily the credential has been used, the QS asks the driver to present another credential by sending it the signed message

$$\langle Cred\_Used, CN_k, TOI_k, SIG_{QSK}(Cred\_Used \| CN_k \| TOI_k) \rangle.$$

The above validation process then repeats. The driver records the previous credential as unused but will not use it any more in the future. He/she also records the QS's signed message so that he/she could later present it to TA (to avoid unnecessary charging).

If the above validation is successful, based on the reservation status of the charging stations and the real-time road conditions (reported by OBUs and RSUs as discussed in [10]), QS suggests the locations of one or more charging stations together with the charging durations at each charging station to the driver. Note that the searching of charging stations is actually another non-trivial scheduling problem and we will leave it as our future work. To summarize, the searching and reservation process should guarantee that:

- The charging stations have enough power when the electric vehicle V_i arrives (since under smart grid framework, power is supplied on an on-demand manner); and
- The total delay experienced by V_i is minimized. The total delay here includes travelling delay (to be reflected by the fact that whether the chosen charging stations are close enough to the original travelling route) and queuing delay at charging stations (since an electric vehicle usually takes much longer time for charging up than for filling up the fossil fuel tank).

Since the credential number CN_k of the credential $Cred_k$ is only known by the tamper-proof device and QS, we makes use of it to develop a secure channel (*i.e.* symmetric encryption using the key CN_k) for ongoing communications between V_i and QS from now on. QS replies the tamper-proof device the set of charging sta-

tions (with identities and locations) and charging durations at each charging station via the secure channel and requests it to submit the appropriate number of credentials for reservation purpose. The tamper-proof device submits the credentials via the secure channel in return. In some cases, a driver may dislike one or more charging stations suggested. For example, the driver may have arguments with the staff at a certain charging station before. In this case, he/she can request QS to provide a replacement set of charging stations by eliminating the charging stations he/she dislikes. The above process then repeats. Similar to the above, any message sent by QS to V_i is signed using *QSK* to avoid being tampered by eavesdroppers on its way.

For simplicity, we assume that the credentials will be used in sequence. That is, the first credential in the submitted list of credentials will be used at the charging station closest to Src_i while the last credential in the submitted list of credentials will be used at the charging station closest to $Dest_i$. After validating the submitted credentials, QS distributes them to the charging stations in order. Note that QS and charging stations are connected via a fixed infrastructure and so we assume that the communications there are secure. The charging stations then record the reservations accordingly by storing the corresponding credentials. Having the credentials, a charging station can double-verify the reservation status upon V_i arrives.

4.4. Charging Phase

The driver then drives his/her electric vehicle V_i to the suggested (and reserved) charging stations in sequence. Since the credentials are being used in sequence, based on QS's suggestion (charging stations and charging durations at them), he/she knows which credential(s) should be used at which charging station and can present the credential(s) appropriately. To resist against eavesdroppers, all credentials presented are encrypted using the identity $CRID_j$ of the corresponding charging station (say CS_j).

Upon receiving the message, the charging station CS_j first decrypts using its private key CSK_j to obtain the credentials. It then checks whether it has received the same credential from QS (during reservation phase) before. If yes, it serves V_i by charging it up. For commercial purpose, an electric vehicle which has not made reservation should also be served but at a lower priority.

4.5. Reconciliation Phase

This phase is usually done at the end of each charging period (say a month). The QS sends all credentials stored in its local database to TA. Each driver also sends the credentials that have not been used (as well as QS's signed *Cred_Used* messages) to TA. For authentication and integrity purpose, a driver's tamper-proof device signs the message using VSK_i.

Since TA has recorded the number of signed credentials and their values during the credential signature requesting phase, based on the received information, it can now calculate how many times the driver has used the searching service and how many charging durations the driver has reserved. The driver is then charged accordingly.

Note that our system also supports different charging models. For example, an operator may want to charge a driver more if he/she makes reservations during peak hours. In our system, this can be done easily by requiring the driver to submit more credentials during reservation phase.

5. Security Analysis

In this section, we briefly analyze our scheme with respect to the security requirements listed in Section 3.4.

- Message integrity and authentication:

Before a vehicle can request for credential signing from the TA, it needs to authenticate itself using its signature and certificate. Thus each properly-signed credential represents the valid identity of a vehicle. Before a vehicle can issue a charging station searching query, make reservations to charging stations and receive charging service at charging stations, it needs to present one or more properly-signed credentials. Hence a vehicle is authenticated everywhere in our scheme.

Recall that in our scheme, all messages sent by TA and QS are properly signed by their private keys, *TSK* and *QSK*, respectively and their identity, *TRID* and *QRID*, are assumed to be known by everyone. Thus the actual sender of the messages concerned can be easily guaranteed.

Since our scheme relies heavily on credentials signed by the TA, next we formally prove that signatures generated by the TA is secure under the RSA assumption—given a public key (N, e) and a message m where N is a RSA modulus and e is a random number less than $\varphi(N)$, it is hard to evaluate $m^d \bmod N$ where $ed \equiv 1 \bmod \varphi(N)$.

On the other hand, since we focus on the security of TA's signature, we ignore the blind signature mechanism (in fact, a signature blindly generated by TA is equivalent to a conventional signature) and assume that all credential numbers are randomly generated by TA.

***Theorem*:** TA's signature on any credential is secure under the RSA assumption. If there exists an adversary A who makes at most q_s signature queries and qc credential number generation queries to a random oracle and can successfully generate an existential forgery with an advantage $Adv_A \geq \varepsilon$ in time τ, we can construct a reduction

R which can make use of A to solve the RSA problem and break the RSA assumption with an advantage $Adv_R \geq \varepsilon'$ in time τ' such that $\varepsilon' \approx \varepsilon / q_c$ and $\tau' \approx \tau$.

***Proof*:** Assume that the challenge to the reduction R is that given a public key (N, e) and a complete set of credential contents (credential value, time of issuance and value where the latter two are denoted as TV^* while all three are denoted as CTV^*, to compute CTV^* mod N where d is the private key corresponding to (N, e). R sends (N, e) to A and A can make q_s signature queries and q_c credential number generation queries (*i.e.* given time of issuance and value, to generate a credential number to complete the credential content) to R. To facilitate the execution of the game, R creates a variable i (initialized to 1) to record the number of credential number generation queries received so far. It maintains a table with entries (TV_i, CTV_i, CTV_i^d) and simulates the signature and credential number generation queries as described below. It also chooses a random $i^* \epsilon [1 \cdots q_c]$, sets $TV_{i^*} = TV^*$ and sets $CTV_{i^*} = CTV^*$.

We first talk about the simulation of signature queries upon given time of issuance and value (TV). If there already exists an entry in the table $(TV, *, y)$, then it returns y as the signature. If there does not exist an entry with message TV, then it picks a random number y and sets CTV to y^e and adds the entry (TV, y^e, y) to the table. Note that y is a valid signature for TV since $(y^e)^d$ mod $N = y$. It then sends y to A as a response to its signature query. If the queried time of issuance and value are TV^* (*i.e.* the i^{*th} entry), R quits since it does not know how to answer.

Next we talk about the simulation of credential number generation queries upon given time of issuance and value TV. If $i \neq i^*$, then R picks a random number y and sets CTV to y^e and adds the entry (TV, y^e, y) to the table. Note that y is a valid signature for TV since $(y^e)^d$ mod $N = y$. It then sends y^e to A as a response to its credential number generation query. But if this is the i^{*th} credential number generation query, R updates $TV_{i^*} = TV$ and returns CTV^* as a response.

Finally, A is allowed to forge a signature and send to R. Since we assume that A needs R's help in generating a credential number, the signature must correspond to one of the CTV values queried before. With a probability $1/q_c$, A will forge a signature corresponding to the i^{*th} entry. R can then resolve the challenge of computing CTV^* mod N.

Therefore, $Adv_R \geq \epsilon / q_c$. This leads to a contradiction to the RSA assumption and so we can conclude that TA's signature on any credential is secure.

- Privacy preserving:

In our scheme, the only place that a vehicle shows its real identity is when it presents its signature and certificate during credential signature requesting phase and reconciliation phase. For all other phases including charging station searching and reservation phase and charging phase, a vehicle only presents an anonymous credential signed by the TA. Since the anonymous credential does not carry any information about the vehicle's identity, even TA and QS collude, no one knows who is performing which charging station query and who will visit which charging station.

During charging station searching and reservation phase, the travelling source and destination of a driver is first encrypted using QS's public key before sending out. Thus no third party can eavesdrop them. For the searching results (*i.e.* the set of suggested charging stations and charging durations) by QS, they are encrypted using the unique credential number of the credential presented by the driver. Since that credential number is only known by the driver and QS (even TA does not know it due to the property of blind signature), no third party can obtain the results as well.

Therefore, the driver's privacy is preserved.

6. Analysis on Time Complexity

In this section, we briefly analyze the time complexity of our SPCS scheme. Note that we ignore the time complexity involved in setup phase since it can be done offline and is only done once occasionally (e.g. when TA wants to update the parameters). It is not critical to the efficiency of our scheme.

We let T_{senc}, T_{sdec}, T_{aenc}, T_{adec}, T_{sig} and T_{ver} denote the time required to perform symmetric encryption, symmetric decryption, asymmetric encryption, asymmetric decryption, signature generation and signature verification operations respectively. These operations dominate the efficiency of our scheme, so we only consider the time taken by these operations and neglect all others.

According to Section 4.2, during the credential signature requesting phase, the tamper-proof device on vehicle V_i takes T_{sig} of time to produce $SIG_{VSKi}(Cred\_Sig\_Req)$. TA then takes T_{ver} of time to verify this signature. Next TA takes T_{aenc} of time to produce $AS\_ENC_{VPKi}(SESS\_KEY)$ and T_{sig} of time to produce the signature $TSIG_{TSK}(AS\_ENC_{VPKi}(SESS\_KEY))$. Upon receiving the message, the tamper-proof device on vehicle V_i takes T_{adec} of time to decrypt $AS\_ENC_{VPKi}(SESS\_KEY)$ and takes T_{ver} of time to verify the signature $TSIG_{TSK}(AS\_ENC_{VPKi}(SESS\_KEY))$.

For each credential, the tamper-proof device on vehicle V_i takes T_{senc} of time to symmetrically encrypt it and takes T_{sig} of time to sign it. For each unopened credential, TA takes T_{sig} of time to blindly sign it. Since the credential signing process should be the most time-consuming part in our whole SPCS scheme, we will analyze its per-

formance further using implementation in Section 7.

According to Section 4.3, during the charging station searching and reservation phase, the tamper-proof device on vehicle V_i takes T_{aenc} of time to produce $AS\_ENC_{QRID}\left(Cred_k \| Src \| Dest\right)$. QS then takes T_{adec} of time to decrypt the message and takes T_{ver} of time to check the validity of $Cred_k$ by verifying TA's signature on it. If so unluckily the credential has been used, QS takes T_{sig} of time to generate the signature $SIG_{QSK}\left(Cred\_Used \| CN_k \| TOI_k\right)$. For all ongoing communications, the tamper-proof device on V_i and QS takes T_{senc} for symmetric encryption or takes T_{sdec} of time for symmetric decryption. For any message sent by QS to V_i, QS takes T_{sig} of time to sign it while the tamper-proof device on V_i takes T_{ver} of time to verify the signature.

According to Section 4.4, during the charging phase, the tamper-proof device on vehicle V_i takes T_{aenc} of time to encrypt each credential. The charging station CS_j then takes T_{adec} to decrypt it.

According to Section 4.5, during the reconciliation phase, the tamper-proof device on vehicle V_i takes T_{sig} of time to sign all unused credentials. Accordingly, TA takes T_{ver} of time to verify the signature.

Next let us summarize the time complexity of secure taxi service scheme in **Table 1**.

7. Implementation Results

Throughout our scheme, the most time-consuming part should be the credential signing process by the trusted authority. It may even become a bottleneck if lots of vehicles request it for signing operation at about the same time. In this section, we present our implementation results and show that the credential signing process can be completed in reasonable time when the parameters are properly set.

We have written a test program in Java to measure the actual time required for the TA to sign the credentials. The TA is implemented on a laptop computer with an Intel Core 2 Duo CPU, T5870@2 GHz and all signing processes run on a single core. We assume that the TA has to sign 10,000 credentials at a certain instance. The results are shown in **Figure 1**. From the figure, we can see that longer signing time is required when the length of TA's private key (*i.e. TSK*) gets longer.

Since 512 bits RSA keys are proved to be insecure nowadays, let us compare the cases with 1024 bits and 2048 bits RSA key for *TSK* in depth. Assume that on average, an electric vehicle uses 150 credentials per day. Then for each month, each vehicle uses 4500 credentials per month. For our scheme, we set N to 10 (*i.e.* for every 10 credentials submitted by a vehicle, the TA only signs 1 of them and the other 9 are for checking purpose). This means that the TA needs to sign at least 45,000 credentials for each client. From the data we obtained from the experiment, signing 45,000 tickets using a consumer PC with 1024 bits RSA key for *TSK* needs about 5 minutes. If 2048 bits RSA key is used, it needs about 37 minutes. Assume that the TA server can run for 24 hours a day so that it can keep on signing credentials for different vehicles, about 43,200 vehicles can be supported by the system if 1024 bits RSA key is used and about 1168 vehicles can be supported if 2048 bits RSA key is used. Nevertheless, recall that the TA server is usually more powerful than a conventional laptop computer. This means that the signing process can be done much faster. It is also straight forward to apply multi-threading technique for the credential signing process and multiple TA servers can be used for this purpose.

Besides credential signing, vehicles need to prepare the binding factors which also require modular exponenttiation. However, we suggest that the exponent part of the public key can be assigned a small number. In this way, the exponentiation can be done in a much shorter time. This also allows us to use a cheaper device (*i.e.* a conventional computer or even a smart card) on the client side.

Table 1. Time complexity of spcs scheme.

Phase	Time Complexity
Credential signature requesting	$T_{sig} + T_{adec} + T_{ver}\left(V_i\right)$ $T_{ver} + T_{aenc} + T_{sig}\left(\mathrm{TA}\right)$
Credential signing	$T_{senc} + T_{sig}\left(V_i\right)$ (for each credential) $T_{sig}\left(\mathrm{TA}\right)$ (for each unopened credential)
Charging station searching and reservation	$T_{aenc}\left(V_i\right)$ $T_{adec} + T_{ver}\left(\mathrm{QS}\right)$ $T_{sig}\left(\mathrm{QS}\right)$ (if credential has been used) $T_{senc} / \left(T_{sdec} + T_{ver}\right)$ $\left(V_i\right)$ (for each ongoing communication) $T_{senc} + T_{sig} / T_{sdec}\left(\mathrm{QS}\right)$ (for each ongoing communication)
Charging phase	$T_{aenc}\left(V_i\right)$ $T_{adec}\left(CS_j\right)$
Reconciliation	$T_{sig}\left(V_i\right)$ $T_{ver}\left(\mathrm{TA}\right)$

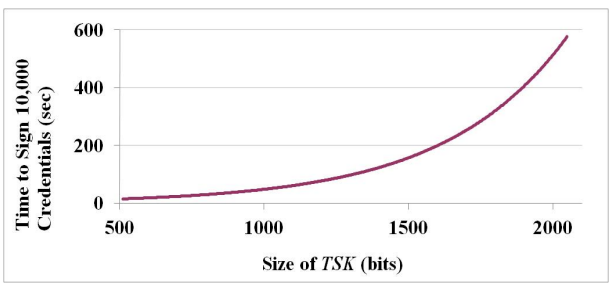

Figure 1. Time to sign 10,000 Credentials vs Size of *TSK*.

8. Conclusion

In this paper, we addressed the problem of privacy-preserving charging station searching and reservation for electric vehicles. We made use of the property of blind signature to achieve this goal. In brief, an electric vehicle first generates a set of anonymous credentials on its own. A trusted authority then blindly signs on them after verifying the identity of the vehicle. After that, the vehicle can make charging station searching queries and reservations by presenting those signed anonymous credentials. Throughout our scheme, the most time-consuming part should be the credential signing process by the trusted authority. It may even become a bottleneck if lots of vehicles request it for signing operation at about the same time. Through implementation, we showed that such a credential signing process could be completed in reasonable time when the parameters were properly set. The process could be as quickly as 5 minutes when 1024 bits of RSA signing key was used. Moreover, we showed that our scheme is secure enough in terms of authentication and privacy-preserving. In the future, we will work out an approximate solution for the querying server to optimistically schedule electric vehicles to charging stations around the city.

REFERENCES

[1] F. Wang, D. Zeng and L. Yang, "Smart Cars on Smart Roads: An IEEE Intelligent Transportation Systems Society Update," *IEEE Pervasive Computing*, Vol. 5, No. 4, 2006, pp. 68-69. doi:10.1109/MPRV.2006.84

[2] H. Oh, C. Yae, D. Ahn and H. Cho, "5.8 GHz DSRC Packet Communication System for ITS Services," *Proceedings of the IEEE VTC*'99, Amsterdam, 19-29 September 1999, pp. 2223-2227.

[3] D. Chaum, "Blind Signatures for Untraceable Payments, Advances in Cryptology," *Proceedings of the Springer-Verlag Crypto*'82, Vol. 3, 1983, pp. 199-203.

[4] R. Lu, X. Lin, H. Zhu and X. Shen, "SPARK: A New VANET-Based Smart Parking Scheme for Large Parking Lots," *Proceedings of the IEEE INFOCOM*'09, Rio de Janeiro, 19-25 April 2009, pp. 1413-1421.

[5] T. W. Chim, S. M. Yiu, L. C. K. Hui and V. O. K. Li, "OPQ: OT-Based Private Querying in VANETs," *Transactions on Intelligent Transportation Systems*, Vol. 12, No. 4, 2011, pp. 1413-1422. doi:10.1109/TITS.2011.2158208

[6] F. Calabrese, M. Colonna, P. Lovisolo, D. Parata and C. Ratti, "Real-Time Urban Monitoring Using Cell Phones: A Case Study in Rome," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 12, No. 1, 2011, pp. 141-151. doi:10.1109/TITS.2010.2074196

[7] C. Zhang, R. Lu, X. Lin, P. H. Ho and X. Shen, "An Efficient Identity-Based Batch Verification Scheme for Vehicular Sensor Networks," *Proceedings of the IEEE INFOCOM*'08, Phoenix, 13-18 April 2008, pp. 816-824.

[8] C. Zhang, X. Lin, R. Lu and P. H. Ho, "RAISE: An Efficient RSU-Aided Message Authentication Scheme in Vehicular Communication Networks," *Proceedings of the IEEE ICC*'08, Beijing, 19-23 May 2008, pp. 1451-1457.

[9] A. Wasef and X. Shen, "PPGCV: Privacy Preserving Group Communications Protocol for Vehicular Ad Hoc Networks," *Proceedings of the IEEE ICC*'08, Beijing, 19-23 May 2008, pp. 1458-1463.

[10] T. W. Chim, S. M. Yiu, L. C. K. Hui and V. O. K. Li, "SPECS: Secure and Privacy Enhancing Communications for VANET," *Ad Hoc Networks*, Vol. 9, No. 2, 2011, pp. 189-203.

[11] B. K. Chaurasia, S. Verma and S. M. Bhasker, "Message Broadcast in VANETs Using Group Signature," *Proceedings of the IEEE WCSN*'09, Allahabad, 27-29 December 2008, pp. 131-136.

[12] A. Studer, E. Shi, F. Bai and A. Perrig, "TACKing Together Efficient Authentication, Revocation, and Privacy in VANETs," *Proceedings of the IEEE SECON*'09, Rome, 22-26 June 2009, pp. 1-9.

[13] L. Chan, S. L. Ng and G. Wang, "Threshold Anonymous Announcement in VANETs," *IEEE Journal on Selected Areas in Communications*, Vol. 29, No. 3, 2011, pp. 605-615. doi:10.1109/JSAC.2011.110310

[14] D. Chaum, "Security without Identification: Transaction Systems to Make Big Brother Obsolete," *Communications of the ACM*, Vol. 28, No. 70, 1985, pp. 1030-1044. doi:10.1145/4372.4373

[15] E. Aimeur, H. Hage and F. S. M. Onana, "Anonymous Credentials for Privacy-Preserving E-learning," *Proceedings of the IEEE MCETECH*'08, Washington DC, 23-25 January 2008, pp. 70-80.

