Computer Communications 34 (2011) 447–456

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

PAACP: A portable privacy-preserving authentication and access control protocol in vehicular ad hoc networks

Lo-Yao Yeh [a], Yen-Cheng Chen [b,*], Jiun-Long Huang [a]

[a] Department of Computer Science, National Chiao Tung University, Hsinchu 300, Taiwan
[b] Department of Information Management, National Chi Nan University, Puli, Nantou 545, Taiwan

ARTICLE INFO

Article history:
Available online 25 May 2010

Keywords:
Access control
Key establishment
Mutual authentication
Privacy
Vehicular ad hoc networks

ABSTRACT

Recently, several studies addressed security and privacy issues in vehicular ad hoc networks (VANETs). Most of them focused on safety applications. As VANETs will be available widely, it is anticipated that Internet services could be accessed through VANETs in the near future. Thus, non-safety applications for VANETs would rise in popularity. This paper proposes a novel portable privacy-preserving authentication and access control protocol, named PAACP, for non-safety applications in VANETs. In addition to the essential support of authentication, key establishment, and privacy preservation, PAACP is developed to provide sophisticated differentiated service access control, which will facilitate the deployment of a variety of non-safety applications. Besides, the portability feature of PAACP can eliminate the backend communications with service providers. Therefore, better performance and scalability can be achieved in PAACP.

1. Introduction

With the rapid growth of wireless communication technology, each car with wireless communicating capability can be envisioned. A vehicle will be allowed to communicate with roadside infrastructure or other vehicles. Vehicular ad hoc networks (VANETs) are emerging to improve road safety and traffic management. Recently, several communities, industries and academic institutions [1–4], have embarked on investigating many aspects of VANETs. It is estimated that the market of VANETs will bring billions of dollars by 2012.

In VANETs, there are two components: onboard units (OBUs) and roadside units (RSUs). OBUs represent the wireless communication devices equipped in vehicles, and RSUs are wireless access devices located at critical points or intersections on the road. There are two kinds of communications: roadside-to-vehicle communications (RVCs) and inter-vehicle communications (IVCs). The birth of VANETs comes from improving the road safety. Therefore, safety-related applications are developed over VANETs. In addition to safety-related applications, VANETs also provide non-safety applications [5,6] to offer maps, advertisements, and entertainment information [7]. For example, Microsoft Corp.'s MSN TV and KVH industries Inc. [8,9], have introduced an automotive vehicle Internet access system, called TracNet, bringing Internet services to in-car video screen.

In the recent years, several researches on VANETs have been investigated by academic or industries, such as IEEE P1609.2 working group [1], Car-2-Car. [2] consortium. Most studies were interested in the performance of medium access control (MAC) layer or the routing issues inherent in VANETs. Recently, some works addressed the security issues. As a special case of mobile ad hoc networks (MANET), VANETs may suffer any malicious user's behaviors, such as bogus information and replay attacks on the disseminated messages. Among various security threats, privacy preservation in VANETs is one of the new challenges to protect users' private information including the driver's name, license plate, model, and traveling route. In 2005, Raya and Hubaux [10] first proposed a solution to tackle both security and privacy issues for safety-related applications. However, their solution is not complete and sound [11]. In 2007, Lin et al. [12] proposed a secure and privacy-preserving protocol, called GSIS, for VANET communications. GSIS adopted a group signature scheme in IVCs and ID-based cryptography (IBC) in RVCs to protect communication messages. All the above protocols were developed especially for safety-related applications. Similar to safety applications, non-safety applications in VANETs have to take both security and privacy issues into consideration [12–14]. In addition, designing a practical non-safety application for VANETs should take into consideration the following characteristics in VANETs [11,15,16].

1. Stringent time constraint in communication: The speed of a vehicle could be more than 140 km/h. The communication delay in IVCs or RVCs should be short enough to meet stringent time requirement [1,7,15].

* Corresponding author. Tel.: +886 49 2910960x4654; fax: +886 49 2915205.
E-mail addresses: lyyeh@cs.nctu.edu.tw (L.-Y. Yeh), ycchen@ncnu.edu.tw (Y.-C. Chen), jlhuang@cs.nctu.edu.tw (J.-L. Huang).

0140-3664/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2010.05.010

2. Large scale networks: In general, with an inter-vehicle distance of 70 m, there are some 70 vehicles within a radius of 1 km around a given car. During a traffic jam, with an inter-vehicle instance of 5 m, there can be more than 1000 vehicles within the same region. Therefore, VANETs will be large scale networks [15,16].

Both characteristics introduce performance and scalability issues in VANETs. In 2008, Zhang et al. [7,17] proposed two schemes to deal with the scalability problem in VANETs. Wang et al. [11] proposed an enhanced communication protocol based on the infrastructure of Raya and Hubaux [10,14] to support non-safety applications with confidentiality and non-repudiation property. However, Wang et al.'s scheme did not address the scalability issue. Li et al. [15] also proposed a secure and efficient communication scheme with privacy preservation, called SECSPP, for non-safety applications in VANETs. Moreover, SECSPP discussed the security issue among service providers, roadside units and vehicles. In SECSPP, a vehicle needs to acquire a blind signature for privacy preservation before the vehicle accesses the desired services from its neighboring RSU. A service provider (SP) is responsible for signing and verifying the validity of signatures, and also involves in session key establishment between the RSUs and requesting vehicles. There are some drawbacks in SECSPP:

1. Deficient in meeting stringent time requirement: When a vehicle tries to access a non-safety service via an RSU, the RSU must pass the signature sent from the requesting vehicle to the proper SP for verification, whereas the SP may be located in a distant network. The speed of a vehicle may be extremely high. It is possible that the response sent from the SP has not arrived yet, but the requesting vehicle had passed the transmission range of the RSU.
2. Lack of scalability in SP: All requests of non-safety applications must be first verified by the proper SP, which will become the bottleneck of SECSPP. The scalability issue rises in a popular SP if a large number of requests pours out.
3. Short of differentiated service access control: In SECSPP, when a vehicle sends the *Access_Service_Request* to an SP via an RSU, the SP only responds the accept/reject permission. However, in modern commerce model, an SP may provide several services with different access privileges for different users' requirements, named differentiated service access control [18].

The lack of scalability and access control in SECSPP will limit the development of non-safety applications. In this paper, we propose a Portable privacy-preserving Authentication and Access Control Protocol, named PAACP, with the support of differentiated service access control. In addition, considering stringent time requirement in transmission delay, PAACP eliminates the communications between the roadside units (RSUs) and service providers (SPs). In a conventional access control scheme, SPs are usually responsible for determining the validity of the access requests. To get rid of the communication with SPs, we propose a novel portable access control method to store a portable service right list (*SRL*) into each vehicle, instead of keeping the *SRL*s in the SPs. In order to assure the validity and privacy of an *SRL*, we also propose a novel attachable blind signature. Based on the attachable blind signature, vehicles (OBUs) cannot tamper the *SRL*. Therefore, PAACP can prevent privilege elevation attacks [19]. As for privacy protection of users, the SP cannot trace the current location of the requesting vehicle, due to the attachable blind signature and the no need of any verification by SP. In addition, PAACP is more efficient than conventional access control schemes since RSUs can verify the correctness of an *SRL* without backend communications with SPs. As a result, PAACP is desirable for large scale VANETs. To the best of our knowledge, PAACP is the first study supporting sophisticated service access control without the scalability problem in VANETs. In summary, PAACP achieves the following properties: (1) mutual authentication between the requesting vehicle and RSU, (2) dynamic session key establishment for the subsequent communications, (3) privacy preservation of the vehicle's information, (4) data confidentiality and integrity, (5) differentiated service access control, and (6) better scalability.

The remainder of this paper is organized as follows. The related work is introduced in Section 2. The proposed PAACP scheme including system architecture and preliminary cryptography is presented in Section 3. Section 4 presents the security analysis and the correctness analysis proven by BAN logic model. The comparison of security features and performance evaluation are given in Section 5. Finally, we conclude this article in Section 6.

2. Related work

2.1. Li et al.'s work

Recently, Li et al. [15] proposed a secure and efficient communication scheme, named SECSPP, with authenticated key establishment for non-safety applications in VANETs. SECSPP is the first security scheme addressing non-safety applications with explicit authentication procedures [15]. In this section, we briefly introduce the procedures of SECSPP. The notations throughout Li et al.'s protocol are summarized in Table 1.

SECSPP consists of two phases: access authorization phase and access service phase. There are three participants: the vehicular node V_i, the service provider S_i, and the roadside device R_j. In the access authorization phase, V_i gets an authorized credential AC_i^* from S_i. Then, in the access service phase, V_i presents the authorized credential AC_i^* to access the desired services via R_j without disclosing any sensitive information.

2.1.1. Access authorization phase

- Step 1: $V_i \rightarrow S_i :< VID_i, SID_i, T_{V_i}, C \oplus (VID_i \| SID_i \| AC_i' \| M_i \| T_{V_i}) >$

First, V_i selects a random number a_1 and computes the authorized credential AC_i, where $AC_i = H(M_i \| VID_i \| a_1)$. Next, V_i chooses a blind factor a_2 to blind AC_i, and makes $AC_i' = a_2^{PK_{S_i}} \cdot AC_i$. Finally, V_i sends $< VID_i, SID_i, T_{V_i}, C \oplus (VID_i \| SID_i \| AC_i' \| M_i \| T_{V_i}) >$ to S_i, where $C = (SID_i^2)^{H(T_{V_i}) \cdot VK_i} \pmod N$ and VK_i is V_i's secret key, which is based on non-interactive ID-based public key cryptography [15].

Table 1
Notations of SECSPP.

VID_i	The identity of vehicular node i
RID_j	The identity of roadside device node j
S_i	The identity of service provider i
VK_i	The secret key of V_i, based on non-interactive ID-based public key cryptography
RK_j	The secret key of R_j, based on non-interactive ID-based public key cryptography
SPK_i	The secret key of S_i, based on non-interactive ID-based public key cryptography
(PK_{S_i}, SK_{S_i})	The public key and private key of service provider S_i
MAC	The message authentication code $MAC = H(K; m)$, where $m$denotes message under the protection key of K.
M_i	The receipt of the service access sent from S_i for a user i to register as a legal user
$H(\cdot)$	A collision-free and public one-way hash function
$\oplus$	Exclusive OR operation
T_x	A timestamp, which node x attaches
$a \| b$	Concatenation of message a and b
$E_{PK_{S_i}}\{\cdot\}$	The asymmetric encryption function with service provider's PK_{S_i}
$D_{SK_{S_i}}\{\cdot\}$	The asymmetric decryption function with service provider's SK_{S_i}

- Step 2: $S_i \rightarrow V_i :< C' \oplus (SID_i \| VID_i \| AC_i'' \| T_{S_i}) >$

After receiving $< VID_i, SID_i, T_{V_i}, C \oplus (VID_i \| SID_i \| AC_i' \| M_i \| T_{V_i}) >, S_i$ reveals $(VID_i \| SID_i \| AC_i' \| M_i \| T_{V_i})$ by computing $C \oplus (VID_i \| SID_i \| AC_i' \| M_i \| T_{V_i}) \oplus C'$ and verifies the validity of M_i, where $C' = (VID_i^2)^{H(T_{V_i}) \cdot SPK_{S_i}} \pmod N$,[1] and SPK_{S_i} is the secret key of S_i. If is M_i valid, then S_i records (VID_i, M_i, T_{V_i}) in its database and marks M_i as non-fresh. In addition, S_i signs AC_i' with its private key SK_{S_i} by computing $AC_i'' = AC_i'^{SK_{S_i}} = a_2 \cdot AC_i^{SK_{S_i}}$. Finally, S_i delivers $< C' \oplus (SID_i \| VID_i \| AC_i'' \| T_{S_i}) >$ to V_i.

Once getting $C' \oplus (SID_i \| VID_i \| AC_i'' \| T_{S_i}), V_i$ extracts $(SID_i \| VID_i \| AC_i'' \| T_{S_i})$ by calculating $C' \oplus (SID_i \| VID_i \| AC_i'' \| T_{S_i}) \oplus C$. Then, AC_i'' is unblinded by computing $AC_i'' \cdot (a_2)^{-1}$ and then V_i obtains $AC_i^* = AC_i^{SK_{S_i}}$. Moreover, AC_i^* is confirmed by checking whether $AC_i = (AC_i^*)^{PK_{S_i}}$. If yes, V_i believes AC_i^* is the signature of AC_i; otherwise, V_i drops it and stops this session.

2.1.2. Access service phase

- Step 1: $V_i \rightarrow R_j :< Access\_Service\_Request, E_{PK_{S_i}}\{Access\_Service\_Request, RID_j, AC_i, AC_i^*, T_{V_i}, a_3\} >$

When a legal V_i wants to access the pay-service from the roadside unit R_j, V_i computes $E_{PK_{S_i}}$ {*Access_Service_Request*, RID_j, AC_i, AC_i^*, T_{V_i}, a_3}, where a_3 is a random number generated by V_i. Then, V_i sends it with an *Access_Service_Request* request to R_j.

- Step 2: $R_j \rightarrow S_i$: <RID_j, T_{R_j}, $C \oplus (E_{PK_{S_i}}$ {*Access_Service_Request*, RID_j, AC_i, AC_i^*, T_{V_i}, a_3})>

Once receiving the *Access_Service_Request* request sent from V_i, R_j computes $C \oplus (E_{PK_{S_i}}\{Access\_Service\_Request, RID_j, AC_i, AC_i^*, T_{V_i}, a_3\})$, where $C = (SID_i^2)^{H(T_{R_j}) \cdot RK_j}$, and delivers $(RID_j, T_{R_j}, C \oplus (E_{PK_{S_i}}\{Access\_Service\_Request, RID_j, AC_i, AC_i^*, T_{V_i}, a_3\}))$ to its back-end service provider S_i.

- Step 3: $S_i \rightarrow R_j$: <SID, $C' \oplus$(*Access_Permission*, a_3, b_1, AC_i, T_{S_i})>

After receiving the message from R_j, S_i first extracts $E_{PK_{S_i}}\{Access\_Service\_Request, RID_j, AC_i, AC_i^*, T_{V_i}, a_3\}$ by computing $C \oplus (E_{PK_{S_i}}\{Access\_Service\_Request, RID_j, AC_i, AC_i^*, T_{V_i}, a_3\}) \oplus C'$, where $C' = (RID_j^2)^{H(T_{R_j}) \cdot SPK_{S_i}} \pmod N$, and computes $D_{SK_{S_i}}\{E_{PK_{S_i}}\{Access\_Service\_Request, RID_j, AC_i, AC_i^*, T_{V_i}, a_3\}\}$. Next, S_i will confirm the validity of the authorized credential AC_i^* by checking whether $AC_i = (AC_i^*)^{PK_{S_i}}$ holds or not. If yes, V_i is granted the access privilege from R_j and then S_i generates a random number b_1 and computes the temporary service key $TSK_i = H(a_3 \| b_1 \| AC_i \| 0)$; otherwise, this access request is denied. Last, S_i sends < SID_i, $C' \oplus$(*Access_Permission*, a_3, b_1, AC_i, T_{S_i})> to R_j.

- Step 4: $R_j \rightarrow V_i$:<b_1, $TSK_i \oplus (RID_j, b_2, T_{R_j})$>

Upon receiving the message from S_i, R_j acquires (*Access_Permission*, a_3, b_1, AC_i, T_{S_i}) by computing $C' \oplus$(*Access_Permission*, a_3, b_1, AC_i, T_{S_i})$\oplus C$. Based on a_3, b_1, AC_i, R_j can compute $TSK_i = H(a_3 \| b_1 \| AC_i \| 0)$ for the subsequent data encryption for accessing pay-services between V_i and R_j. Next, R_j generates a random number b_2 for mutual authentication and sends the message $< b_1, TSK_i \oplus (RID_j, b_2, T_{R_j})>$ to V_i.

- Step 5: $V_i \rightarrow R_j$: <*MAC*>

After receiving the message from R_j, V_i calculates temporary service key $TSK_i = H(a_3 \| b_1 \| AC_i \| 0)$ by the received b_1, his own a_3 and AC_i, and then reveals (RID_j, b_2, T_{R_j}) by TSK_i. Next, V_i sends back $MAC = H(TSK_i'', b_2 + 1)$, where $TSK_i'' = H(a_3 \| b_1 \| AC_i \| 1)$, for mutual authentication.

Finally, R_j verifies V_i by checking whether *MAC* is correct or not. If yes, V_i is convinced; otherwise, this session is dropped. In the end, both R_j and V_i take $TSK_i = H(a_3 \| b_1 \| AC_i \| k)$ for data encryption of the kth session in the access service phase, where $k = 2, 3, 4, \ldots$, and so on.

2.2. Comments on SECSPP

SECSPP gives a security solution for non-safety applications in VANETs. Both security and privacy issues were considered in the protocol design. However, the scalability issue is not addressed in SECSPP. As mentioned above, VANETs should be regarded as large scale networks. In SECSPP, only a single SP takes charge of checking the validity of authorized credential AC_i. This may lead to a bottleneck problem, or may introduce the threat of potential Distributed/Denial-of Service (D/DoS) attacks. In addition, SECSPP does not support differentiated service access control, which allows a variety of non-safety services with different privileges. It is believed that if non-safety applications try to achieve the success in VANETs, a sophisticated access control scheme [18] is required to meet a variety of users' demands.

In SECSPP, each SP_i needs two secret keys, SPK_{S_i} and SK_{S_i}. The former is used for non-interactive ID-based public key, and the latter is used for signing and decrypting the messages sent from vehicles or RSUs. This may cause inconvenience for SPs. In terms of security, SECSPP adopted a conventional blind signature to prevent the vehicle's privacy from tracing by SPs. However, the conventional blind signature is not designed for access control. If SECSPP is adopted to provide the differentiated service access control, SECSPP could not withstand privilege elevation attacks [19], since SECSPP cannot examine whether the access privileges are valid or not. To deal with this weakness, we will first devise a novel attachable blind signature, and then develop a portable access control scheme based on the attachable blind signature. In addition, performance and scalability issues will be carefully examined in the design of our protocol.

3. The Portable privacy-preserving Authentication and Access Control Protocol (PAACP)

3.1. System architecture

A system architecture of non-safety applications in VANETs is given in Fig. 1. In general, a non-safety application of VANETs is composed of three types of entities, (1) onboard units (OBUs), (2) roadside units (RSUs), and (3) service providers (SPs). The SPs are responsible for providing various non-safety services with the differentiated access privileges. For example, a travel company serves as the service provider to provide a travel guide service with two classes of customers, VIP and non-VIP customers. While a VIP customer uses the travel guide service, the travel company automatically pushes a bunch of coupons of local hotels or restaurants, which are only available for a VIP-exclusive service. In practice, an SP may deploy devices or databases in networks near RSUs for offering various non-safety services in a distributed fashion. Thus, the access of non-safety services can be fulfilled locally.

Initially, each OBU must send a *Register_Service_Request* message to the SP to request the authorization of the desired services in the access authorization phase. In the access service phase, when an OBU wants to access some services, the OBU delivers the *Access_Service_Request* message to its neighboring RSU. If the requesting OBU is authorized, then the neighboring RSU sends

[1] Based on non-interactive ID-based public key cryptography, $C = (SID_i^2)^{H(T_{V_i}) \cdot VK_i} = (VID_i^2)^{H(T_{V_i}) \cdot SPK_{S_i}} \pmod N = C'$.

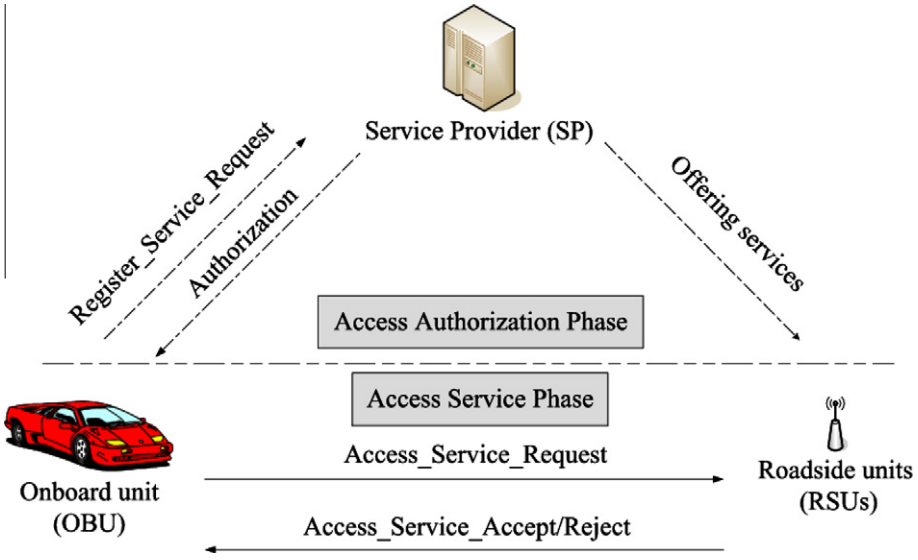


Fig. 1. System architecture of a non-safety application.

back the *Access_Service_Accept* message and allows the requesting OBU to access the desired services; otherwise, the OBU receives the *Access_Service_Reject* message without any access permission.

3.2. The proposed attachable blind signature

Generally, blind signatures could be implemented by different cryptosystems, such as RSA and ElGamal. We adopt RSA-based blind signature in the proposed blind signature scheme. First, we briefly introduce the conventional RSA-based blind signature. A user U_A blinds a message m with a random blind factor r and computes the blind document

$$BD = r^e m,$$

where e is the public key of the signer. The blind document is then sent to the signer. Once receiving BD, the signer signs BD by his/her private key d as

$$BD' = BD^d = rm^d.$$

Then the signer sends BD' back to U_A. Upon receiving BD', U_A unblinds BD' by the blind factor r to obtain the signer's signature

$$BD'' = m^d = BD'/r.$$

Finally, U_A confirms the integrity of BD'' by checking

$$(BD'')^e = m.$$

In a conventional blind signature, the signer does nothing but signs the blind document BD sent from the user. Such a conventional blind signature is not designed for access control in origin. In terms of access control, the service provider (SP) plays the role of the signer and also confirms whether the requested access privileges for a user are legal. Since the blind document containing the requested access privileges is blinded by a random number r, it is infeasible for the SP to check whether the requested access privileges are legal. To ensure the genuineness of the requested access privileges, we propose an attachable blind signature as follows.

First, a user U_A chooses random blind factors r_1, r_2 and a, and then computes

$$BD_1 = (r_1)^e m^a (modN).$$
$$BD_2 = (r_2)^e m^{(1-a)} (modN).$$

Then, U_A sends BD_1 and BD_2 to the signer. Once receiving BD_1 and BD_2, the signer first attaches a message m' into BD_2 as

$$BD_2^{\#} = (r_2)^e m^{(1-a)} m' (modN)$$

and signs $BD_1, BD_2^{\#}$ by his/her own private key d as

$$BD_1' = (BD_1)^d = r_1 (m^a)^d (modN),$$
$$BD_2' = (BD_2^{\#})^d = r_2 (m^{(1-a)} m')^d (modN).$$

Then, the signer sends BD_1', BD_2' back to U_A. Upon receiving BD_1' and BD_2', U_A first unblinds two messages as

$$BD_1^U = BD_1'/r_1 = (m^a)^d (modN),$$
$$BD_2^U = BD_2'/r_2 = (m^{(1-a)} m')^d = (m^{(1-a)d})(m'd)(modN)$$

and generates the signer's signature by

$$BD'' = BD_1^U \cdot BD_2^U = m^d \cdot m'^d (modN).$$

Note that the proposed attachable blind signature scheme attaches a message m' into the signature and still keeps the privacy of user's message m. To withstand the privileges elevation attack, PAACP takes the advantage of m' to ensure the validity of m.

3.3. Portable privacy-preserving Authentication and Access Control Protocol (PAACP)

In this section, we propose a novel Portable privacy-preserving Authentication and Access Control Protocol (PAACP) for non-safety applications in VANETs. Since the stringent time requirement is regarded as an important property of VANETs [7,11,17], PAACP gets rid of the backend communication between roadside units and service providers. In PAACP, SPs do not involve in the access service phase. That is, the verification of vehicles and their access privileges can be accomplished in RSUs themselves. Thus, it is not required to take a long round trip of communication between RSUs and SPs for access request verifications. In the access authorization phase of PAACP, the SP authorizes the access privileges for a legitimate vehicle, and stores a service right list in a portable authorized credential carried by the vehicle. The portable authorized credential is protected using the proposed attachable blind signature to withstand privilege elevation attacks.

Another merit of PAACP is the support of differentiated access privileges for each service. A service may provide different access

privileges to satisfy distinct requirements of the users. For this, the access privileges for the service i are represented by a bit string AR_i of k_i bits. Each bit of AR_i represents a distinct access privilege of the service i. In a travel guide service, for instance, we may use one bit to indicate the permission of viewing detailed maps, and one bit to indicate the permission of downloading coupons, and another bit to denote the capability of watching a particular video program. Therefore, k_i distinct access privileges can be specified in AR_i. Assume an SP provides n services with access privileges AR_i, $1 \leqslant i \leqslant n$. Suppose a vehicle V is granted to access m services, $1 \leqslant m \leqslant n$, with index $\{SVID_1, SVID_2, \ldots, SVID_m\}$. Let AR'_j, $1 \leqslant j \leqslant m$, be the granted value of AR_j for V. Then, the service right list SRL for V can be represented by a bit string

$$SRL = (SVID_1 \| AR'_1) \| (SVID_2 \| AR'_2) \| \cdots \| (SVID_m \| AR'_m)$$

with length $\sum_{i=1}^{m} (logn + k_i)$. For example, we assume an SP provides 16 services and the travel guide is the 12th service with three different access privileges: viewing maps, downloading coupons, watching videos, then $n = 16$ and $k_{12} = 3$ for AR_{12}. If V_i applies for the travel guide service with the access privileges of viewing maps and downloading coupons, then V will set $SRL = (1100 \| 110)$ [19].

The proposed scheme consists of two phases: access authorization phase and access service phase, as illustrated in Figs. 2 and 3. The notations of PAACP are summarized in Table 2.

According to the purchased services and granted access privileges, in the access authorization phase, a vehicle V_i creates a service right list $SRL_i^{V_i}$ in $AC_i^{V_i}$ and blinds $AC_i^{V_i}$ into blind documents $BD1_i$, $BD2_i$. To obtain the corresponding portable authorized credential for later use, V_i sends the blind documents with its certificate $Cert_i$ to the service provider S_t. After checking the validity of $Cert_i$, S_t generates the service right list $SRL_i^{S_t}$ based on the sold contract, stores $SRL_i^{S_t}$ in $AC_i^{S_t}$ and attaches $AC_i^{S_t}$ into blind documents $BD1_i$, $BD2_i$ based on the proposed attachable blind signature. Then, S_t delivers the blind documents back to V_i. At the end of the access authorization phase, V_i will obtain the portable authorized credential AC_i^* , where AC_i^* consists of both $AC_i^{V_i}$ and $AC_i^{S_t}$. AC_i^* is stored in V_i's tamper-proof device [10,11,17]. In the access service phase, V_i sends an *Access_Service_Request* to its neighboring RSU R_j, and then R_j verifies the authorized credential AC_i^* by itself without further communication with S_t. According to the access privileges stored in the authorized credential $AC_i^{S_t}$, R_j could decide whether V_i's request is accepted or not. Furthermore, R_j could detect whether V_i is launching a privilege elevation attack.

We explain the details of each phase as follows.

3.4. Access authorization phase

- Step 1: $V_i \rightarrow S_t$: $<VID_i, \sigma_i, BD1_i, BD2_i>$

In the access authorization phase, according to the purchase receipt from the service provider S_t, a vehicle V_i creates its service right list $SRL_i^{V_i} = \{SVID_1 \| AR_1 \| \cdots \| SVID_k \| AR_k\}$, where $SVID_k$ denotes the index of the kth service, and AR_k represents the granted access privileges of $SVID_k$. The service right list will be signed by S_t as part of an authorized credential. First, V_i chooses random numbers RN_1, RN_2 and a, and then sets $AC_i^{V_i} = \{SID_t \| T_{expired} \| SRL_i^{V_i}\}$. These random numbers are used as blind factors. Then, V_i computes blind documents

$$BD1_i = (RN_1)^{PK_{S_t}} \cdot (AC_i^{V_i})^a (modN),$$
$$BD2_i = (RN_2)^{PK_{S_t}} \cdot (AC_i^{V_i})^{1-a} (modN).$$

Finally, V_i sends its identity VID_i, signature $\sigma_i = \{BD1_i, BD2_i\}^{SK_{V_i}}$, and the blinded documents $BD1_i$, $BD2_i$ to S_t.

- Step 2: $S_t \rightarrow V_i$: $<BD1'_i, BD2'_i>$

Upon receiving message $< VID_i, \sigma_i, BD1_i, BD2_i >$ sent from V_i, S_t first confirms whether the σ_i is valid by V_i's public key. If valid, V_i is successfully authenticated; otherwise, this session is dropped. S_t then generates the authorized credential $AC_i^{S_t} = \{SID_t \| T_{expired} \| SRL_i^{S_t}\}$ according to the selling contract for V_i and attaches it into $BD2_i^{\#}$ as

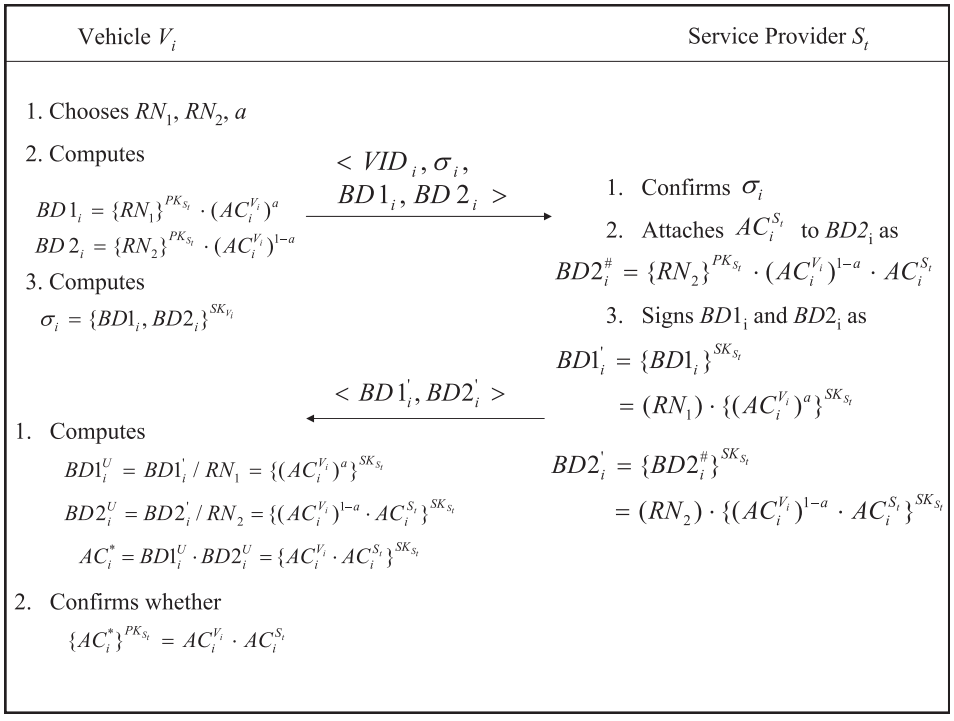


Fig. 2. Access authorization phase of the proposed scheme.

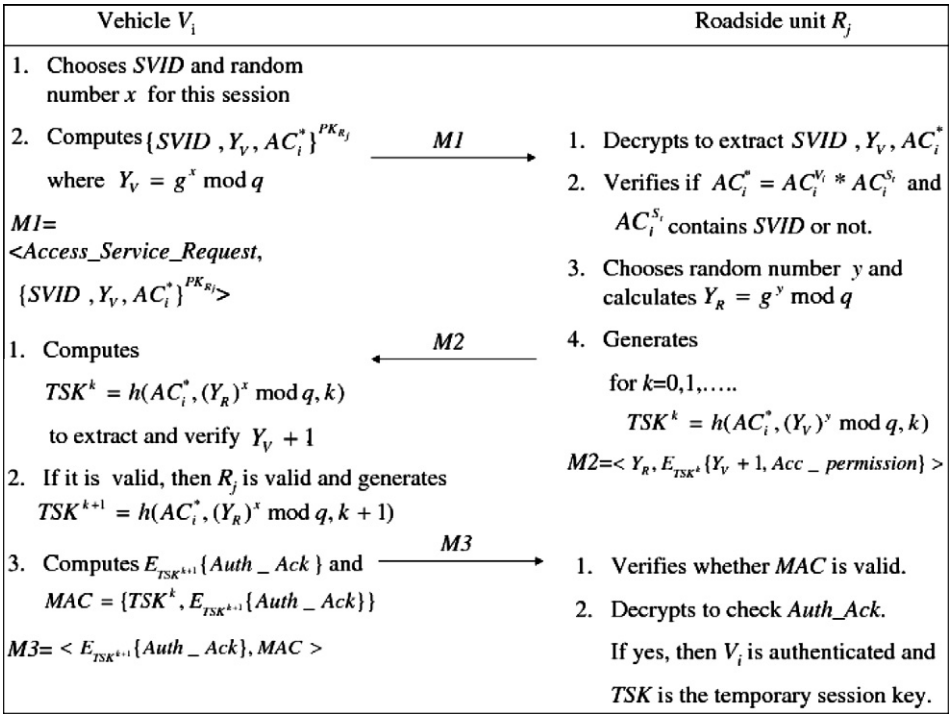


Fig. 3. Access service phase of the proposed scheme.

Table 2
Notations of the proposed scheme.

V_i	The ith vehicle
VID_i	The identification of the ith vehicle
R_j	the jth roadside unit
RID_j	The identification of the jth roadside unit
S_t	the tth service provider
SID_t	The identification of the tth service provider
$SVID_i$	The identification of the ith service
AR_i	The access privilege of $SVID_i$
(PK_{V_i}, SK_{V_i})	A public key and private key of vehicle V_i
(PK_{R_j}, SK_{R_j})	A public key and private key of roadside unit R_j
(PK_{S_t}, SK_{S_t})	A public key and private key of service provider S_t
$Cert_i$	The certificate of vehicle V_i
TSK	A temporary session key between the vehicle and roadside unit
a, RN_j	Random numbers, where j = 1, 2.
AC_i	Authorized credential for vehicle V_i
AC^{S_t}, AC^{V_i}	Authorized credential made by S_t and V_i, respectively
AC_i^*	Portable authorized credential for vehicle V_i
SRL	The service right list
SRL^{S_t}, SRL^{V_i}	Service right list made by S_t, and V_i, respectively
$BD1, BD2$	The blind documents used in the proposed attachable blind signature
$E_{K_{AB}}\{\cdot\}$	The encryption function with shared key K_{AB}
$D_{K_{AB}}\{\cdot\}$	The decryption function with shared key K_{AB}
MAC	The message authentication code
$h(\,)$	A collision-free and public one-way hash function
σ_i	A signature signed by secret key SK_{V_i}
q	A large prime number
g	A generator of a finite cyclic group with order q.

$$BD2_i^{\#} = BD2_i \cdot AC_i^{S_t} = \left(RN_2^{PK_{S_t}} \cdot (AC_i^{V_i})^{1-a} \cdot AC_i^{S_t}\right)(mod N).$$

Then, S_t signs them as follows:

$$BD1_i' = BD1_i^{SK_{S_t}} = \left((RN_1)^{PK_{S_t}} \cdot \left(AC_i^{V_i}\right)^a\right)^{SK_{S_t}} = (RN_1) \cdot (AC_i^{V_i})^{aSK_{S_t}},$$

$$BD2_i' = BD2_i^{\#SK_{S_t}} = \left((RN_2)^{PK_{S_t}} \cdot \left(AC_i^{V_i}\right)^{1-a} \cdot AC_i^{S_t}\right)^{SK_{S_t}} = (RN_2) \cdot \left(\left(AC_i^{V_i}\right)^{1-a} \cdot AC_i^{S_t}\right)^{SK_{S_t}}.$$

Next, $BD1_i', BD2_i'$ are sent back to V_i. After obtaining $< BD1_i', BD2_i' >$ from S_t, V_i unblinds them as follows:

$$BD1_i^U = BD1_i'/RN_1 = (AC_i^{V_i})^{aSK_{S_t}},,$$

$$BD2_i^U = BD2_i'/RN_2 = \left((AC_i^{V_i})^{1-a} \cdot AC_i^{S_t}\right)^{SK_{S_t}}.$$

In order to get the portable authorized credential $AC_i^* = \{AC_i^{V_i} \cdot AC_i^{S_t}\}^{SK_{S_t}}$, V_i computes

$$BD1_i^U \cdot BD2_i^U = \left(AC_i^{V_i}\right)^{aSK_{S_t}} \cdot \left(\left(AC_i^{V_i}\right)^{1-a} \cdot AC_i^{S_t}\right)^{SK_{S_t}} = \left(AC_i^{V_i} \cdot AC_i^{S_t}\right)^{SK_{S_t}}.$$

To confirm the AC_i^* is certified, V_i could verify the correctness of AC_i^* by checking whether $\{AC_i^*\}^{PK_{S_t}}$ is equal to $AC_i^{V_i} \cdot AC_i^{S_t}$.[2] If it holds, V_i keeps AC_i^* for the subsequent service requests; otherwise, V_i will stop this session. Note that, we assume V_i could protect AC_i^* in secret by tamper-proof device after obtaining AC_i^*.

3.5. Access service phase

- Step 1: $V_i \rightarrow R_j :< Access\_Service\_Request, \{SVID, Y_V, AC_i^*\}^{PK_{R_j}} >$

In the access service phase, when a vehicle V_i wants to access the desired services from its neighboring roadside unit R_j, V_i will transmit an *Access_Service_Request* with $\{SVID, Y_V, AC_i^*\}^{PK_{R_j}}$, where *SVID* is the identification of the desired services, and $Y_V = g^x \bmod q$, where x is a random number in $\mathbb{Z}_q^*$, to R_j.

- Step 2: $R_j \rightarrow V_i$:<Y_R, E_{TSK^0} {Y_V+1, *Access_Permission*}>

Upon receiving $\{SVID, Y_V, AC_i^*\}^{PK_{R_j}}$, R_j decrypts it by his own private key SK_{R_j} to acquire ($SVID$, Y_V, AC_i^*). First, R_j calculates

$$AC_i^{S_t} = \left(AC_i^{*PK_{S_t}}\right)^{1/2}$$

[2] Note that if both V_i and S_t are legal, $AC_i^{V_i}$ and $AC_i^{S_t}$ should be the same, which means V_i or R_j could confirm whether $AC_i^{V_i} \cdot AC_i^{S_t}$ is expected or not.

to extract the access credential $AC_i^{S_t}$, which is authorized by S_t. Then, R_j examines whether SID_t as well as $SVID$ is included in $AC_i^{S_t}$, and checks the validity of the authorized credential by $T_{expired}$. If the verification succeeds, AC_i^* is legitimate and V_i is authorized; otherwise, R_j terminates this session. After AC_i^* is verified, R_j calculates

$$Y_R = g^y \bmod q,$$

where y is a random number in $\mathbb{Z}_q^*$, and generates a temporary session key

$$TSK^0 = h(AC_i^*, (Y_V)^y \bmod q, 0)$$

for protecting the subsequent communications. Finally, R_j delivers $< Y_R, E_{TSK^0}\{Y_V+1, Access\_Permission\}>$ to V_i.

- Step 3: $V_i \rightarrow R_j$: $<E_{TSK^1}\{Auth\_Ack\}, MAC>$

After receiving $< Y_R, E_{TSK^0}\{Y_V+1, Access\_Permission\}>$, V_i computes a temporary session key

$$TSK^0 = h(AC_i^*, (Y_R)^x \bmod q, 0)$$

and decrypts $E_{TSK^0}\{Y_V+1, Access\_Permission\}$ using TSK^0 to check the validity of $Y_V + 1$. If valid, R_j is successfully authenticated; otherwise, V_i ceases this connection. Then, V_i generates an $Auth\_Ack$ encrypted by

$$TSK^1 = h(AC_i^*, (Y_R)^x \bmod q, 1)$$

and computes the message authentication code

$$MAC = (TSK^0, E_{TSK^1} Auth\_Ack).$$

Finally, V_i sends $<E_{TSK^1}\{Auth\_Ack\}, MAC>$ to R_j.

Upon receiving the message, R_j verifies the MAC to ensure the integrity, and calculates

$$TSK^1 = h(AC_i^*, (Y_V)^y \bmod q, 1)$$

to decrypt $E_{TSK^1}\{Auth\_Ack\}$. If R_j could recognize $Auth\_Ack$, it is implied that V_i indeed holds the corresponding TSK^1. Finally, the subsequent communications can be encrypted by the session key TSK^k, where

$$TSK^k = h(AC_i^*, (Y_V)^y \bmod q, k).$$

4. Security and correctness analysis

4.1. Security properties

Based on the security of asymmetric and symmetric cryptosystems, PAACP preserves several security properties, as discussed below.

4.1.1. Mutual authentication

In PAACP, vehicle V_i and roadside unit R_j are mutually authenticated based on the secret authorized credential AC_i^* and the public key cryptosystem. Only an authorized V_i could own AC_i^*, and only legitimate R_j has the capability of decrypting messages to extract Y_V. Mutual authentication is an essential property to prevent malicious attacks from outsiders. This property will be formally proven by BAN logic proof in Section 4.2.

4.1.2. Context privacy

Based on the proposed attachable blind signature, no one could comprehend the access privileges in $AC_i^{V_i}$. Note that even if service provider S_t could realize V_i's access privileges in the access authorization phase, the non-linkability discussed in next subsection is also guaranteed. In the access service phase, all messages are well protected by asymmetric and symmetric cryptographic primitives without disclosing any information to outsiders. On the other hand, although the roadside unit R_j can confirm the validity of the authorized credential AC_i^* and the desired services $SVID$, R_j cannot realize who is accessing those services.

4.1.3. Non-linkability

In general, the non-linkability means both insiders and outsiders could neither realize any session to a particular user nor link any two different sessions to the same user. First, PAACP ensures that outsiders cannot attain any information in the communications between V_i and R_j. Therefore, the non-linkability for outsiders is guaranteed under the security of asymmetric and symmetric cryptosystems. On the other hand, service provider S_t cannot link any sessions to a particular user since S_t is not involved in the access service phase. Moreover, even if S_t obtains the authorized credential AC_i^*, the non-linkability is still ensured by the proposed attachable blind signature since S_t cannot link this AC_i^* to the exact vehicle, unless the service right list itself is distinct for a certain vehicle. It is possible that R_j could link the authorized credential AC_i^* to the same vehicle, but R_j cannot derive any additional information about the vehicle.

4.1.4. Data traffic protection

After the execution of PAACP, all messages between V_i and R_j are encrypted by the session key TSK. Under the security of symmetric cryptographic primitive such as AES, the data confidentiality and integrity are guaranteed as well.

4.1.5. Differentiated service access control

Different from the previous work [18] adopting several public/private key pairs to achieve the differentiated service access control, PAACP only requires a single public/private key pair and uses an SRL [19] to encode the access privileges of each services. As a result, PAACP also keeps the privacy of the service request in the access service phase.

4.1.6. Forward secrecy

Different from the previous works [15,18], PAACP applies the concept of Diffie-Hellman exchange protocol using $Y_V = g^x \bmod q$ and $Y_R = g^y \bmod q$ to establish the session key $TSK^i = h(AC_i^*, g^{xy} = (Y_V)^y = (Y_R)^x, i)$. This implies that PAACP preserves the forward secrecy property even though a long-term secret key is compromised.

4.2. Correctness verification

We formally verify the correctness of PAACP based on well-known model BAN logic [20]. BAN logic is a famous logic model widely used to reason about beliefs, encryptions and protocols. Protocol correctness means both communication parties ascertain that they have shared a fresh session key and ensure that the same belief is held by the other party. Recently, several authentication schemes [18,21] have applied BAN logic to prove the validity of an authentication and key distribution protocol.

In Table 3, we briefly introduce some notations used in BAN logic. Following the beginning procedures of BAN logic, we first list the verification goals in Table 4. To reduce the expression complexity, we provide a generic type and then transform it to an idealized protocol, as shown in Table 5. We highlight the messages exchanged between vehicle V and roadside unit R, and verify whether the two communicating parties could ascertain that they have shared a fresh session key TSK with each other.

According to BAN logic, some assumptions are made in Table 6. The first two assumptions mean both V and R believe that R holds a public key K_b. Assumptions (A.3) and (A.4) tell that V and R, respec-

Table 3
Notations used in BAN logic.

$P\| \equiv X$	Principal P believes X or P would be entitled to believe X. In other words, P may act as though X is true. The construct is central to the logic.
$P \triangleleft X$	P sees formula X Some has sent a message containing X to P.
$P\| \sim X$	P once said X. P at some time sent a message including X.
$P \Rightarrow X$	P has jurisdiction over X. P is an authority on X and should be trusted on the matter.
$<X>_Y$	Formula X combined with a secret parameter Y.
$\{X\}_Y$	Formula X encrypted by key Y.
$P \overset{K}{\leftrightarrow} Q$	Principals P and Q may use the shared key K to communicate. Here K will never be discovered by any principals expect for P and Q.
$\overset{K_b}{\mapsto} R$	R has K_b as public key.
$P \overset{AC^*}{\rightleftharpoons} Q$	The formula AC^* is a secret known only to P and Q, and possibly to principals trusted by them.
$\#(X)$	The formula X is fresh. X has not been sent in a message at any time before.
TSK	A temporary session key negotiated in each session.

Table 4
Goals of correctness verification.

Verification goals:	
$G1.V\| \equiv V \overset{TSK}{\leftrightarrow} R$	$G2.V\| \equiv R\| \equiv V \overset{TSK}{\leftrightarrow} R$
$G3.R\| \equiv V \overset{TSK}{\leftrightarrow} R$	$G4.R\| \equiv V\| \equiv V \overset{TSK}{\leftrightarrow} R$

Table 5
Generic and idealized type of the access service phase.

Protocol generic type	
Message 1	$V \rightarrow R : \{SVID, Y_V, AC^*\}^{PK_R}$
Message 2	$R \rightarrow V: Y_R, E_{TSK}\{Y_V + 1, Access\_Permission\}$
Message 3	$V \rightarrow R: E_{TSK}\{Y_V + 1, Auth\_Ack\}, MAC$
Idealized protocol type	
Message 1	$V \rightarrow R : \{Y_V, V \overset{TSK}{\leftrightarrow} R\}^{K_R}$
Message 2	$R \rightarrow V : Y_R, E_{TSK}\{Y_V, V \overset{AC^*}{\rightleftharpoons} R, V \overset{TSK}{\leftrightarrow} R\}$
Message 3	$V \rightarrow R : E_{TSK}\{V \overset{TSK}{\leftrightarrow} R\}$
Session key	$TSK = h(AC^*, (Y_V)^y = (Y_R)^x = g^{xy})$

Table 6
The assumptions based on BAN logic.

Assumptions	
(A.1) $V\| \equiv \overset{K_b}{\mapsto} R$	(A.2) $R\| \equiv \overset{K_b}{\mapsto} R$
(A.3) $V\| \equiv \#(Y_V)$	(A.4) $V\| \equiv \#(Y_R)$
(A.5) $V\| \equiv V \overset{AC^*}{\rightleftharpoons} R$	(A.6) $R\| \equiv V \overset{AC^*}{\rightleftharpoons} R$
(A.7) $V\| \equiv R\| \equiv V \overset{AC^*}{\rightleftharpoons} R$	(A.8) $R\| \equiv V\| \equiv V \overset{AC^*}{\rightleftharpoons} R$
(A.9) $V\| \equiv R\| \Rightarrow V \overset{TSK}{\leftrightarrow} R$	(A.10) $R\| \equiv V \overset{TSK}{\leftrightarrow} R$

tively, generates Y_V and Y_R regarded as two fresh nonces, which are to ensure the freshness property. Assumptions (A.5) to (A.8) are related to the authorized credential AC^* shared between R and V. In short, R believes that AC^* is the secret shared between an authorized vehicle and itself since R can easily verify the validity of AC^* based on the proposed attachable blind signature. Note that, although R cannot actually realize the exact identification of V, R still believes that AC^* is an authentic secret shared between them. This is the reason why we adopt blind signatures. In assumption (A.9), V believes that R has jurisdiction right over TSK, since R will generate Y_R and send it to V together with V's challenge Y_V. In the view of V, the TSK is determined by the Y_R. The assumption (A.10) holds since R invents the fresh session key TSK with a shared secret AC^* between V and R, and one fresh nonce, Y_R, is chosen by itself. The details of verification procedures are outlined in Table 7. PAACP achieves the verification goals by equations (S.5), (S,6), (S.10) and (A.10). That is, the correctness of PAACP is guaranteed based on BAN logic.

Table 7
Verification procedures of access service phase.

Verification	
Message 1 $V \rightarrow R : \{Y_V, V \overset{TSK}{\rightarrow} R\}^{K_R}$	
(S.1) $R \triangleleft (Y_V, V \overset{AC^*}{\rightleftharpoons} R)$	By seeing rule
Message 2	
$R \rightarrow V : Y_R, E_{TSK}\{Y_V, V \overset{AC^*}{\rightleftharpoons} R, V \overset{TSK}{\leftrightarrow} R\}$	
(S.2) $V \triangleleft (Y_R, Y_V, V \overset{AC^*}{\rightleftharpoons} R, V \overset{TSK}{\leftrightarrow} R)$	By seeing rule
(S.3) $V\| \equiv R\| \sim (Y_R, Y_V, V \overset{AC^*}{\rightleftharpoons} R, V \overset{TSK}{\leftrightarrow} R)$	By (A.5), (S.2), msg-meaning rule
(S.4) $V\| \equiv R\| \equiv (Y_R, Y_V, V \overset{AC^*}{\rightleftharpoons} R, V \overset{TSK}{\leftrightarrow} R)$	By (A.3), (S.3), nonce-verification, freshness rule
(S.5) $V\| \equiv R\| \equiv V \overset{TSK}{\leftrightarrow} R$	By (S.4), belief rule
(S.6) $V\| \equiv V \overset{TSK}{\leftrightarrow} R$	By (S.5), (A.9), jurisdication rule
Message 3 $V \rightarrow R : E_{TSK}\{V \overset{TSK}{\leftrightarrow} R\}$	
(S.7) $R \triangleleft \{V \overset{TSK}{\leftrightarrow} R\}$	By seeing rule
(S.8) $R\| \equiv \#(V \overset{TSK}{\leftrightarrow} R)$	By (A.4), (A.9)
(S.9) $R\| \equiv V\| \sim V \overset{TSK}{\leftrightarrow} R$	By (S.7), (A. 10), msg-meaning rule
(S.10) $R\| \equiv V\| \equiv V \overset{TSK}{\leftrightarrow} R$	By (S.8), (S.9), nonce-verification rule

5. Discussion

5.1. Comparsion

In this section, we compare PAACP with the related works [15,11] in terms of security properties and performance evaluation. First, we compare the security features of PAACP with SECSPP and Wang et al. [11], which are typical authentication schemes for non-safety applications in VANETs. Table 8 lists important security properties in VANETs. The comparison shows that PAACP provides more merits, including differentiated service access control, privilege elevation attack resistance, and better scalability.

Table 8
The comparison of security features.

	Ours	SECSPP [15]	Wang et al. [11]
Mutual authentication	Yes	Yes	Yes
Context privacy	Yes	Yes	Yes
Session key agreement	Yes	Partially yes[a]	Partially yes[b]
Differentiated service access control	Yes	No	No
Privilege elevation attack resistance	Yes	N/A	N/A
Scalability	Fully distributed	Bottleneck at service provider	N/A
Formal correctness proof	Yes	No	No

[a] In SECSPP, the session key TSK is determined by V and S, not V and R.
[b] In Wang et al.'s scheme, the session key TSK is built for inter-vehicle communication (IVC), not V and R.

5.2. Performance evaluation

Next, we evaluate the performance of SECSPP and PAACP in Table 9. For time complexity estimation, we define some computational parameters as follows:

T_{Asym}: the time for the asymmetric encryptions/decryptions.
T_{sym}: the time for the symmetric encryptions/decryptions.
$T_{ID_{exp}}$: the time for the modular exponentiation of the ID-based cryptography.
T_{hash}: the time for the one-way hash function operation.
T_{xor}: the time for the XOR operation.

Based on the computation method in Li et al. [15] and Wang et al. [11], PAACP takes 2.0885 s to compute the necessary operations and SECSPP spends 2.0895 s in the authorization phase. In the access service phase, the verification time $T_{verification}$ of PAACP is 1.5839 s/time and that of SECSPP is 2.613 s/time. Note that the time spent in the access service phase is the major concern in terms of performance, since the access service phase will be executed frequently, whereas the authorization phase is executed only once. In addition to the required computation time in the access service phase, the overall elapsed time can be evaluated by the communication rounds needed and the waiting time for each vehicle when there are a number of service requests simultaneously. In general, the service provider is far away from RSUs, but vehicles are in the neighborhood of RSUs. Let $T_{trans-delay}$ be the transmission delay in seconds to deliver a message from a vehicle, forwarded by an RSU, to the SP. It is reasonable to assume $1 < T_{trans-delay} < 2$. The transmission delay in seconds to deliver a message from a vehicle to its neighboring RSU is less than 0.01 s [4], which can be neglected. Considering the scalability issue, we further assume that n vehicles in the VANET request the services of the same SP at the same time and the locations where these service requests are invoked are uniformly distributed within m RSUs [22]. In SECSPP, the average waiting time $T_{waiting}$ for a requesting vehicle can be estimated as

$$T_{waiting_{SECSPP}} = 2T_{trans-delay} + (n+1)/2 * T_{verification},$$

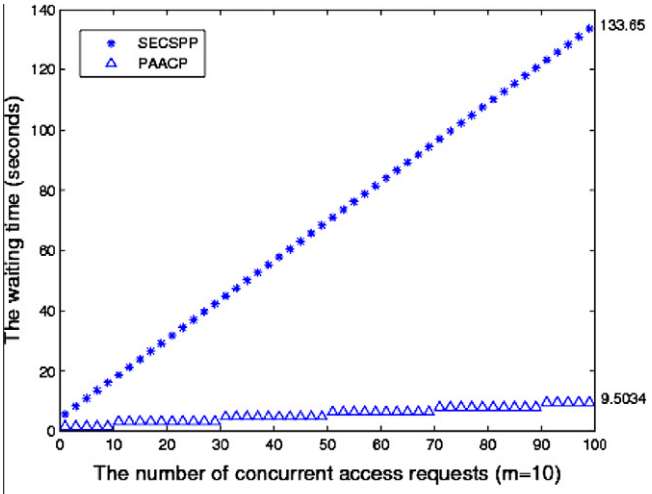
The waiting time consists of round-trip transmission delay and the time spent in the SP for verification. Since there are n requests pending for verification, the average time spent in SP will be $(n+1)/2 * T_{verification}$. On the other hand, in PAACP, the average waiting time $T_{waiting}$ for a requesting vehicle can be estimated as

$$T_{waiting_{PAACP}} = \begin{cases} (n/m+1)/2 * T_{verification}, & \text{if} \quad n > m \\ T_{verification}, & \text{otherwise.} \end{cases}$$

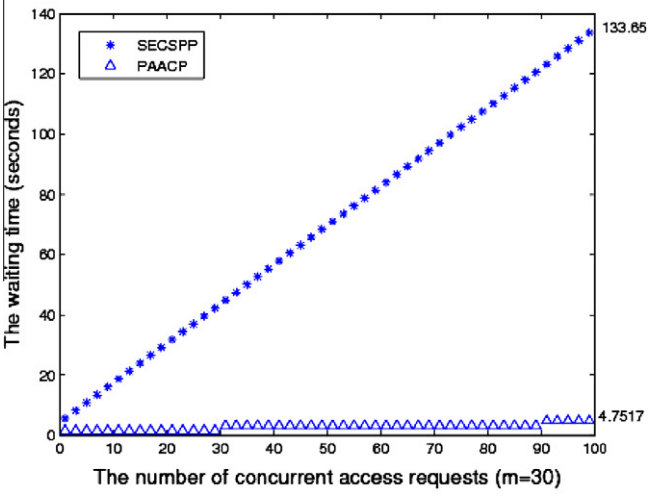
In a uniform distribution of locations, the avergae number of requests pending in each RSU will be n/m. Therefore, the average time spent for request verification in a RSU is $(n/m+1)/2 * T_{verification}$. Fig. 4

Table 9
The comparison of efficiency.

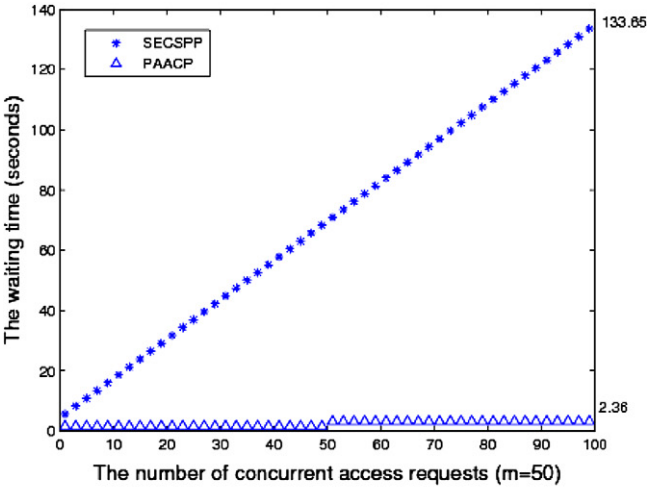
	Ours	SECSPP [15]
Authorization phase	$4T_{Asym} + 1T_{hash}$	$2T_{Asym} + 2T_{ID_e}xp + 3T_{hash} + 4T_{xor}$
Access service phase	$3T_{Asym} + 2T_{sym} + 1T_{hash}$	$3T_{Asym} + 2T_{IDexp} + 6T_{hash} + 5T_{xor}$
Computation time (s)	Authorization phase ≈ 2.0885 (s) Access service phase ≈ 1.5839 (s)	Authorization phase ≈ 2.0895 (s) Access service phase ≈ 2.613 (s)
Communication rounds	2 + 3	2 + 5



(a) The average waiting time v.s. the number of concurrent access requests (m=10)



(b) The average waiting time v.s. the number of concurrent access requests (m=30)



(c) The average waiting time v.s. the number of concurrent access requests (m=50)

Fig. 4. Average wating time vs. concurrent access requests.

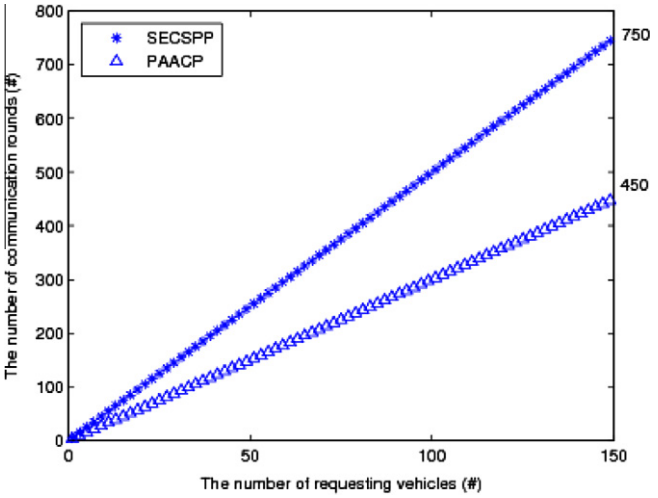


Fig. 5. Communication rounds vs. requesting vehicles.

shows the average waiting time for a service request as n increases with different values of m (10, 30, and 50). As Fig. 5a–c shows, when 100 vehicles are requesting the desired services, the average waiting time to finish the authentication in SECSPP is 134 seconds. As for PAACP, the waiting times for m = 10, 30, and 50 take about 10, 5, and 3 s, respectively. The waiting time for PAACP is short since the verifications of access requests can be performed locally because of the distributed nature of PAACP. Moreover, the more RSUs are installed, the less waiting time in PAACP is required. In terms of communication rounds, PAACP eliminates the transmission overhead between RSUs and SPs. Hence, the total number of communication rounds required in PAACP is lower than that of SECSPP, as shown in Fig. 5. Obviously, the number of communication rounds of PAACP is 60% fewer thant that of SECSPP. In summary, PAACP outperforms SECSPP significantly.

5.3. Implementation issue

In terms of implementation issue, the proposed scheme can be developed by the existing IEEE 1363 standard IEEE1363-2000-Std. [23] for public key cryptography and NIST standards NIST-standard [24] for AES cryptosystem, respectively. Only the proposed attachable blind signature is required to be specifically implemented, and the details of attachable blind signature is introduced in Section 3.2. Moreover, the system parameters of the attachable blind signature can be found in IEEE 1363 standard IEEE1363-2000-Std. [23].

6. Conclusion

In the near future, it is anticipated that various services would be available in VANETs to bring more convenient services to drivers and passengers. Therefore, access control will be an essential security issue in VANETs. In this paper, we have proposed a Portable privacy-preserving Authentication and Access Control Protocol (PAACP) for non-safety applications in VANETs. Considering the stringent time requirement in VANETs, we devised a portable access control protocol to get rid of the involvement of service providers in the access service phase. Due to the portability of authorized service right lists, roadside units can verify the validity of access privileges without the aid of service providers. Moreover, we proposed an attachable blind signature to keep the privacy of the requested services and to withstand the privilege elevation attack. The performance evaluations also show that PAACP is efficient and suitable for large scale VANETs.

References

[1] IEEE1609.2-Std., Ieee trial-use standard for wireless access in vehicular environments-security services for applications and management messages, 2006.
[2] Car-2-Car., [online]. available: <http://www.car-2-car.org>, 2007.
[3] European project react, [online]. available: <http://www.react-project.org>, 2006.
[4] DSRC., Dedicated short range communications, 2006. [Online]. Available: URL <http://www.leearmstrong.com/dsrc/dsrchomeset.htm>.
[5] S. Yousefi, M. Mousavi, M. Fathy, Vehicular ad hoc networks (vanets): challenge and persepectives, in: International Conference on ITS Telecommunications, 2006.
[6] J.T. Isaac, J.S. Camara, S. Zeadally, J.T. Marquez, A secure vehicle-to-roadside communication payment protocol in vehicular ad hoc networks, Comput. Commun. 31 (2008) 2478–2484.
[7] C. Zhang, X. Lin, R. Lu, P.H. Ho, X. Shen, An efficient message authentication scheme for vehicular communications, IEEE Trans. Vehicular Technol. 57 (6) (2008) 3357–3368.
[8] Kvh industries inc. [online]. available: <http://www.kvh.com/>, 2006.
[9] Msntv. [online]. available: <http://www.msntv.com/>, 2006.
[10] M. Raya, J.-P. Hubaux, The security of vehicular ad hoc networks, in: SASN, 2005.
[11] N.-W. Wang, Y.-M. Huwang, W.-M. Chen, A novel sercure communication scheme in vehicular ad hoc networks, Comput. Commun. 31 (2008) 2827–2837.
[12] X. Lin, X. Sun, P.-H. Ho, X. Shen, Gsis: a secure and privacy-preserving protocol for vehicular communications, IEEE Trans. Vehicular Technol. 56 (2007) 3442–3456.
[13] X. Lin, R. Lu, C. Zhang, H. Zhu, P.-H. Ho, X. Shen, Security in vehicular ad hoc networks, IEEE Commun. Mag. 46 (4) (2008) 88–95.
[14] M. Raya, J.-P. Hubaux, Securing vehicular ad hoc networks, J. Comput. Security 15 (2007) 39–68.
[15] C.-T. Li, M.-S. Hwang, Y.-P. Chu, A secure and efficient communication scheme with authenticated key establishment and privacy preserving for vehicular ad hoc networks, Comput. Commun. 31 (2008) 2803–2814.
[16] Y. Toor, P. Muhlethaler, A. Laouiti, Vehicle ad hoc networks: applications and related technical issues, IEEE Commun. Surveys Tutorials 10 (2008) 74–87.
[17] C. Zhang, R. Lu, X. Lin, P.-H. Ho, X.S. Shen, An efficient identity-based batch verification scheme for vehicular sensor networks, in: The 27th Conference on Computer Communications. IEEE INFOCOM, 2008.
[18] K. Ren, W. Lout, K. Kim, R. Deng, A novel privacy preserving authentication and access control scheme for pervasive computing environments, IEEE Trans. Vehicular Technol. 55 (4) (2006) 1373–1384.
[19] Y.-C. Chen, L.-Y. Yeh, An efficient authentication and access control scheme using smart cards, in: Proceedings of International Conference on Parallel and Distributed Systems (ICPADS), 2005.
[20] M. Burrows, M. Abadi, R. Needham, A logic of authentication, ACM Trans. Comput. Syst. 8 (1) (1990) 18–36.
[21] C.-C. Chang, C.-Y. Lee, Y.-C. Chiu, Enhanced authentication scheme with anonymity for roaming service in global mobility networks, Comput. Commun. 32 (4) (2009) 611–618.
[22] X. Lin, X. Sun, X. Wang, C. Zhang, P.-H. Ho, X. Shen, Tsvc: timed efficient and secure vehicular communications with privacy preserving, IEEE Trans. Wireless Commun. 7 (12) (2008) 4987–4998.
[23] IEEE1363-2000-Std., Standard specifications for public key cryptography, 2000.
[24] NIST-standard, Fips publication 197: The advanced encryption standard (aes), 2001.