JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 33, 1025-1040 (2017)
DOI: 10.6688/JISE.2017.33.4.11

A Privacy-Preserving V2I Authentication Scheme Without Certificates*

HAO-HAO NIE[1], YAN-PING LI[1,+] AND QIAN-HONG WU[2]
[1]*School of Mathematics and Information Science*
Shaanxi Normal University
Xi'an, 710119 P.R. China
[2]*School of Computer Science*
Beihang University
Beijing, 100191 P.R. China
E-mail: 1376134798@qq.com; lyp@snnu.edu.cn[+]; *qianhongwu@buaa.edu.cn*

When new vehicles dynamically join in vehicular ad hoc networks (VANETs), there will be hundreds of messages with signatures need to be authenticated by road side units (RSU) in a very short time. If those signatures can be batch verified, the verification efficiency will be greatly improved. The aggregate signature technology is the desired technique towards addressing such problem. It can greatly reduce the total signature length and verification cost and is very efficient and useful in VANETs. In this paper, a novel security-enhanced certificateless aggregate signature scheme for VANETs (SCLAS) is proposed. Our SCLAS scheme can resist the existing powerful attacks and have a higher efficiency than the existing related schemes. The SCLAS scheme also can provide controlled privacy-preserving, which ensures both authentication security and privacy protection simultaneously. Besides, in the random model, it is proven existentially unforgeable against adaptive chosen message and identity attacks under the hardness assumption of the computational Diffie-Hellman problem. The performance evaluation shows our proposed scheme has little storage space and low computation cost compared to prior related work. Hence the SCLAS scheme is very suitable for the VANETs safety-related applications.

Keywords: certificateless public key cryptosystem, aggregate signature, computational Diffie-Hellman problem (CDHP), vehicle-to-infrastructure (V2I), vehicular ad hoc networks (VANETs)

1. INTRODUCTION

With the massive development of wireless communication technologies, vehicular ad hoc networks (VANETs) have become a significant research area for its specific applications such as road safety and traffic management. A VANET consists of trusted authorities (TAs), road side units (RSUs) and on board units (OBUs) installed in the vehicles, see Fig. 1. In VANETs, there mainly are two challenges. On the one hand, vehicles communicate with each other, as well as with RSUs through an open wireless channel, attackers can easily get users' private information, such as identity, track, hobbies, *etc.*, if they are not properly protected. On the other hand, high-speed mobility leads to

---

Received August 16, 2016; revised September 26, 2016; accepted November 5, 2016.
Communicated by Zhe Liu.
* Foundation: The NFSC China (61672083,61402275, 61402015, 61572246, 61272436, 61370190), the Natural Science Foundation of Shaanxi Province (2016JM6069), the Fundamental Research Funds for the Central Universities (GK201603012), the scientific research foundation for the returned overseas Chinese scholars of MOHRSS, the Innovation Fund Designated for Graduate Students of Shaanxi Normal University (2015CXS023).
+ Corresponding author

limited communication time among RSUs and vehicles. As a result, it is crucial to design an efficient authentication scheme with privacy preserving for VANETs.

Privacy protection is an important factor for the public acceptance and successful deployment of VANETs and V2I technology. In general, the users do not want their sensitive information such as real identities to be exposed. The pseudonym is widely used in the communication among entities to provide users' anonymity, such as schemes in [1, 2]. However, when a traffic collision or crime occurs, the legal authorities should be able to retrieve or trace vehicle message by revealing their identities. So many security frameworks based on cryptographic techniques have been proposed so far to achieve the users' privacy preserving in VANETs. Gamage *et al.* [3] put forward an ID-based ring signature scheme with enhanced privacy. And some schemes based on group signature are proposed in schemes [4-7]. However, Malhi *et al.* [8] point out that both ring signature and group signature are not practical in context of VANETs applications because of their complex structure and high computation cost. More efficient and lightweight schemes are urgently needed for mobile OBUs with limited storage, computation and communication capabilities in VANETs.

Nowadays authentication is the most crucial security issue for VANETs. And authentication with conditional privacy-preserving becomes a public challenge in security field. Generally digital signatures are widely used to authenticate messages senders or provide integrity of messages. According to the dedicated short range communications protocol [9], each vehicle broadcasts traffic related messages every 100-300 ms, an RSU usually is needed to oversee 180 vehicles in a high density traffic scenario. Hence one RSU needs to complete at least about 600 authentications per second. A lot of time will be consumed if the message signatures are verified independently (such as one by one), which may cause the important messages without being verified to be lost. In this case, it would be preferable to introduce a new technique which can verify all the signatures received from vehicles in a batch manner and greatly saves the signature verification time. The aggregate signature is an ideal technique towards solving these problems.

1.1 Related Work

In 2003, Boneh *et al.* [10] put forward the concept of aggregate signature that it allows an efficient algorithm to aggregate n signatures on n distinct messages from n distinct users into one signature. An aggregated signature allows the verifier to authenticate the n signatures simultaneously by one verification equation (in a batch manner). Due to the characteristics of aggregate signature, the workload of signature verifier is greatly reduced and the authentication efficiency is improved simultaneously. Both memory space and communication cost will also be saved at the same time, and the loss of authenticated messages because of congestion also is reduced. Hence, aggregate signature schemes are attractive to applications in environments with low bandwidth communications, low storage and low computability such as mobile authentication.

Several PKI (Public Key Infrastructure)-based and ID-based aggregate signature schemes have been proposed [11-17] in the last few years. However, PKI-based authentication mechanisms require a certificate authority to maintain a huge pool of certificates for users, and users need additional computation to verify the validity of other users' certificates. Although ID-based authentication mechanisms alleviate the certificate management and solve the problems of dynamic certificate revocation, they are considered suitable only for private network because of the inherent key escrow problem. In addition, some of ID-based aggregate signature schemes also have different security flaws. For example, Song *et al.* first proposed an ID-based aggregate signature scheme [13], but it is proved to be universally forgeable later. The scheme in [14] is inefficient since each signer's private key is composed of two group elements, which will bring the security storage problem for each signer. Wen *et al.* [15] pointed out a security drawback of the scheme in [16] and proposed a new one, but the signature length of the new scheme [15] linearly increases with the number of signers, resulting a huge verification cost. Yu *et al.* [17] presented a new ID-based aggregate signature which is easy vulnerable to parameter substitution attack. An adversary can forge anyone's signature if he replace Q with $Q'=xP$, $\forall x\in Z_q^*$. Hence, those schemes are still infeasible in practice.

In order to overcome the key escrow problem of ID-based public key cryptosystem (ID-PKC), Al-Riyami and Peterson [18] proposed certificateless public key cryptography (CL-PKC) in 2003. In CL-PKC the Key Generation Center (KGC) does not know the user's full private key. Thus, CL-PKC systems solve the key escrow problem inherent in ID-PKC, as well as, the certificate management problems in traditional PKI cryptosystem. Some other mechanisms may also be used to eliminate the key escrow problem. For instances, in [19], a new aggregate signature scheme and distributed mechanism are applied to solve this problem. But in this paper, we mainly investigate the aggregate signatures schemes in the CL-PKC for VANETs.

A certificateless aggregate signature (CLAS, for short) scheme combines advantages of aggregate signatures and CL-PKC. In a well-designed CLAS scheme, the signature sizes and verifications costs are independent of the number of the original signing messages. Due to this advantage, much attention has been paid to CLAS schemes in recent years. In 2007, Gong *et al.* firstly present two CLAS schemes and define the security model of CLAS schemes [20]. However, Zhang *et al.* pointed out some drawbacks of the security model in [20] and proposed a new scheme [21]. Xiong *et al.* recently proposed a certificateless aggregate signature (CLSA) scheme which is suitable for vehicular ad hoc networks [22]. However, in Xiong *et al.*'s new scheme, the aggregate signature length linearly increases with the number of signers, the same problem cannot be solved in schemes [23, 24]. Recently, in the up-to-date CLAS schemes proposed [25-29], the length of the aggregate signature becomes constant and the efficiency of CLAS schemes is greatly improved. Chen *et al.* [29] and Cheng *et al.* [24] point out that Xiong *et al.*'s scheme [22] is insecure against the type II adversary and then propose two improved schemes, respectively. Unfortunately, Zhang *et al.* [30] gives a more powerful attack on Chen *et al.*'s scheme. Although Cheng *et al.*'s scheme can resist Zhang *et al.*'s attack, the final aggregate signature of their scheme linearly increases with the number of original signers, as well as the scheme in [31], which leads to lower efficiency. So there is still a space to improve the efficiency of the aggregate signature scheme. And few studies combine CLAS and privacy-preserving. We are trying to design a new securely lightweight CLAS scheme with conditional privacy-preserving for VANETs.

1.2 Our Contributions

In this paper, a new certificateless aggregate signature scheme with conditional privacy-preserving for V2I authentication in VANETs is presented. Compared with the existing batch verification schemes, our SCLAS scheme has the following advantages:

- Firstly, our proposed scheme can provide controlled anonymity, *i.e.*, each vehicle is distributed a pseudonym (*i.e.* an assumed identity) to ensure the private communication, meanwhile, a legal trace authority (TRA) can retrieve the real identity from any pseudo identity for any dispute event;
- Secondly, the final aggregate signature of our proposed SCLAS scheme only consists of two group elements, which have a lower storage and communication cost than Cheng *et al.*'s scheme. In addition, in our SCLAS scheme, the verification algorithm needs only four pairing computations, which does not linearly increase with the number of signatures being aggregated. So the scheme is highly efficient in computation;
- Thirdly, our SCLAS scheme can resist Zhang *et al.*'s powerful attack [30]. It also is proven existentially unforgeable against adaptive chosen-message and chosen-identity attacks in the random oracle model under the CDHP assumption over an additive group.

The rest of the paper is organized as follows. Section 2 gives some preliminaries and the generic security model of CLAS schemes. The new SCLAS scheme is presented in Section 3 and its security is proven in Section 4. In Section 5, the performance of our scheme and some existing CLAS schemes for VANETs is compared. Finally, Section 6 concludes our paper.

2. PRELIMINARIES

2.1 Bilinear Pairing

Let G_1 be an additive group of prime order q, and G_2 be a multiplicative group with the same order. A map e: $G_1 \times G_1 \rightarrow G_2$ is called a bilinear map if it satisfies three properties: Bilinearity (*i.e.* $\forall P, Q \in G_1, a, b \in Z_q^*$, $e(aP, bQ) = e(P, Q)^{ab}$), non-degeneracy and computability. Details please see [1, 10].

2.2 Computation Assumptions

The security of our scheme is based on the assumption of intractability of the CDHP and DLP. **DLP assumption** means it is computationally infeasible to obtain integers a, r from given $U \in G_1$, $V \in G_2$ such that $U=aP$ and $V=e(P, Q)^r$. **CDHP assumption** refers to that it is hard to compute abP given (P, aP, bP) for unknown $a, b \in Z_q^*$ where P is a generator of a cyclic group G_1 with order q. In other words, there is no algorithm solve it in polynomial time with non-ignorable probability.

2.3 Framework of Certificateless Aggregate Signature Scheme

Definition 1: Our SCLAS scheme involves n vehicles $V_1, V_2, \ldots, V_n$, some road side units (RSUs) and a trusted authority TA. TA is composed of a trace authority (TRA) and a key generation center (KGC). It is composed of six polynomial time-bound algorithms:

Setup, Pseudo-Identity-Generate (PIG), Vehicle-Key-Generate, Partial-Private-Key-Extract (PPKE), Sign, Aggregate-Sign-Verify (ASV). Six algorithms are described as follows.

Setup: This algorithm is performed by TRA and KGC. Input a security parameter 1^{λ} to the algorithm, output master public secret key pair (mpk_K, msk_K) for KGC, (mpk_T, msk_T) for TRA, and a list of system parameters params.

PIG: This algorithm is run by TRA that accepts a vehicle V_i's real identity RID_i to calculate the corresponding pseudo identity ID_i.

VKG: This algorithm is run by each vehicle that takes a vehicle V_i's pseudo identity ID_i, selects a random value x_i and outputs the vehicle's secret value/public key x_i/pk_i.

PPKE: This algorithm is performed by KGC. Input msk_K, system parameters params, a vehicle V_i's pseudo identity ID_i and his public key pk_i, output the V_i's partial private key D_i. KGC sends it to V_i by a secure channel. The secret key of V_i is (D_i, x_i).

Sign: This algorithm is run by each vehicle. A vehicle V_i inputs system parameters params, a message m_i, and pseudo identity ID_i and his private key (D_i, x_i) and public key pk_i, and outputs a signature σ_i on message m_i.

ASV: This algorithm has two steps and is performed by one of RSUs. First, it takes n vehicles' signature σ_i on message m_i as input and outputs an aggregate signature σ on message (m_1, m_2, ..., m_n). Second, it inputs system parameters params, n vehicle's pseudo identity (ID_1, ID_2, ..., ID_n) and corresponding public keys (pk_1, pk_2, ..., pk_n), message (m_1, m_2, ..., m_n), and an aggregate signature σ. It outputs *true* if the aggregate signature is valid, or *false* otherwise.

2.4 Security Models of Certificateless Aggregate Signature Scheme

Generally, two types of adversaries are considered in **CL-PKC-**type I adversaries and type II adversaries. A type I adversary $\mathcal{A}_1$ does not have access to the master key, but he has the ability to replace the public key of any vehicle with a value of his choice. While a type II adversary $\mathcal{A}_2$ has the ability to obtain the msk_K, but cannot perform public key replacement. The security of our SCLAS scheme is modeled via the following two games between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}_1$ or $\mathcal{A}_2$. BRoth **Game 1** and **Game 2** are composed of three phase: **Setup, Attack, Forgery,** which are described as follows.

Game 1 (For type I adversary)
Setup: The challenger $\mathcal{C}$ run the **Setup** algorithm that takes a security parameter 1^{λ} as input to obtain the system parameters params and a msk_K ***s***. $\mathcal{C}$ then sends params to the adversary $\mathcal{A}_1$ while keeps the msk_K ***s*** secret.

Attack: $\mathcal{A}_1$ can perform a polynomially bounded number of the following types of queries in an adaptive way.

- Hash queries: $\mathcal{A}_1$ can request any hash value, $\mathcal{C}$ return the corresponding value.
- PPKE queries: When $\mathcal{A}_1$ requests the partial private key of a vehicle V_i with pseudo identity ID_i, $\mathcal{C}$ responds V_i's partial private key D_i by running PPKE algorithm.
- Public-Key queries: When $\mathcal{A}_1$ requests the public key of V_i with pseudo identity ID_i, $\mathcal{C}$

answers the corresponding public key pk_i by running Vehicle-Key-Generate algorithm.

- Secret-Value queries: When $\mathcal{A}_1$ requests the secret value of a vehicle whose pseudo identity is ID_i. In respond, $\mathcal{C}$ outputs the secret value x_i ($\mathcal{C}$ outputs $\perp$, if the signer's public key has been replaced).
- Public-Key-replacement queries: For any vehicle V_i with pseudo identity ID_i, $\mathcal{A}_1$ can choose a random value pk_i' as the new public key of V_i. $\mathcal{C}$ will record this replacement.
- Sign queries: When $\mathcal{A}_1$ requests V_i's signature on a message m_i in the region of RSU_j, $\mathcal{C}$ responds the corresponding signature σ_i by running Sign algorithm.

Forgery: Finally, $\mathcal{A}_1$ outputs a tuple (m^*, ID^*, RSU^*, σ^*) in which $m^* = (m_1^*, m_2^*, \ldots, m_n^*)$, $ID^* = (ID_1^*, ID_2^*, \ldots, ID_n^*)$, and σ^* is an aggregate signature. $\mathcal{A}_1$ wins **Game 1** if and only if: (1) σ^* is a valid aggregate signature on messages m^* under identities (ID_1^*, ID_2^*, …, ID_n^*) and the corresponding public keys (pk_1^*, pk_2^*, …, pk_n^*); (2) At least one of the identities, without loss of generality, say $ID_1^* \in ID^*$ has not been queried during the PPKE queries and the (m_1^*, ID_1^*) has never been queried during the Sign queries.

Game 2 (For type II adversary)

In this game, $\mathcal{C}$ works similarly in **Game 1** and interacts with adversary $\mathcal{A}_2$ in almost the same way, except the flowing differences.

- In **Setup** phase, $\mathcal{C}$ will send the msk_K to the adversary $\mathcal{A}_2$ since $\mathcal{A}_2$ simulates a malicious KGC.
- In Attack phase, $\mathcal{A}_2$ has never performed PPKE queries to get V_i's partial private key D_i, nor performed Public-Key-replacement queries to replace V_i's public key pk_i.

Remark 1: In two above games, the responses from the random oracle to $\mathcal{A}_1$'s and $\mathcal{A}_2$'s are uniformly random and independently distributed in G_1. From the $\mathcal{A}$'s view, all responses are valid and random, which are indistinguishable from the real life.

3. OUR EFFICIENT SCLAS SCHEME FOR VANETS

In our scheme, it involves n vehicles V_1, V_2, ..., V_n, some road side units (RSUs) and a trusted authority (TA), which is composed of a trace authority (TRA) and a key generation center (KGC). And TRA is only responsible for V_i's pseudonym generation and tracing the dishonest V_i's real identity. KGC is only charge of the generation of V_i's partial private key D_i. Since we assume that the KGC may be dishonest and TRA must be fully trusted, so KGC and TRA need be separated and each performs its own functions. RSUs are the aggregator and verifier of n signatures from n mobile vehicles V_i (Fig. 2).

A novel and efficient SCLAS scheme for VANETs is stated as follows.

Setup: Both the KGC and TRA input a security parameter 1^λ, the algorithm outputs a cyclic additive group G_1 on elliptic curve which is generated by P with prime order $q \geq 2^\lambda$, a cyclic multiplicative group G_2 with the same order, a bilinear map e: $G_1 \times G_1 \to G_2$, four cryptographically secure hash functions H_0: $G_1 \to \{0,1\}^n$, H_1: $\{0,1\}^* \times G_2^2 \to G_1$, H_2: $G_1^2 \to G_1$, H_3: $\{0, 1\}^* \times G_1^3 \to Z_q^*$.

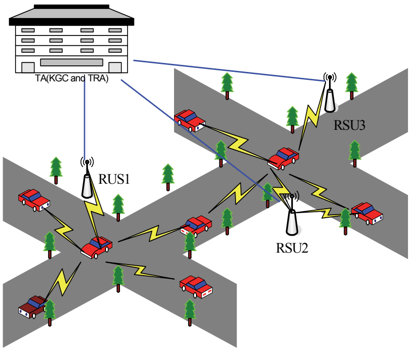

Fig. 1. Generic Framework of VANETs.

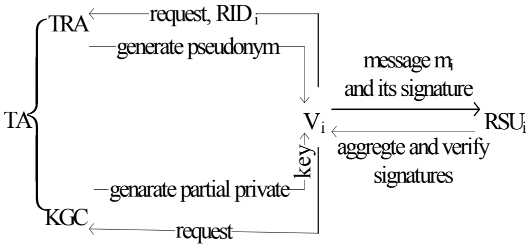

Fig. 2. Function of each participator in our SCLAS scheme.

The KGC selects a random $s \in Z_q^*$ as the system master secret key msk_K and sets $P_{pub}=sP$ as master public key mpk_K. And the TRA chooses a random $t \in Z_q^*$ as msk_T and sets $T_{pub}=tP$ as mpk_T. Both of them keep their msk_K, msk_T secret. Each RSU_i (i = 1, 2, ..., n) have a different identity number $N_{rsu_i} \in \{0, 1\}^*$, which is public. The KGC and TRA publish the system parameters params $=\{q, G_1, G_2, e(.,.), P, P_{pub}, T_{pub}, H_0, H_1, H_2, H_3\}$.

PIG: A vehicle V_i computes $ID_{i1}=k_iP$ for a randomness $k_i \in Z_q^*$ and $V=k_iT_{pub} \oplus RID_i$, then V_i transmits (ID_{i1}, V) to the TRA. RID_i is the real identity of the vehicle V_i and $|RID_i|=n$. After receiving (ID_{i1}, V), the TRA calculates $RID_i=V \oplus tID_{i1}$ and verifies RID_i's validity, then computes $ID_{i2}=RID_i \oplus tH_0(ID_{i1})$ then sends a pseudo identity $ID_i=(ID_{i1}, ID_{i2})$ to vehicle V_i. It's worth mentioning that the TRA does not need to record anything in its secret database, because TRA can get any V_i's real identity from its pseudo identity $ID_i=(ID_{i1}, ID_{i2})$ by calculating $RID_i=ID_{i2} \oplus tH_0(ID_{i1})$.

VKG: A vehicle V_i selects a random $x_i \in Z_q^*$ as his secret value and computes $pk_i=x_iP$. V_i publicly released his public key pk_i.

PPKE: KGC inputs params and msk_K s, V_i's identity ID_i and his public key pk_i, and computes $Q_i=H_1(ID_{i1}||ID_{i2}||pk_i)$, sends partial private key $D_i=sQ_i$ to V_i via a secure channel. The V_i's full private key $sk_i=(x_i, D_i)$.

Sign: Input params, a signed message $m_i \in \mathcal{M}$ ($\mathcal{M}=\{0,1\}^*$), the vehicle V_i's identity ID_i and his private and public key sk_i and pk_i, the vehicle V_i in the region of RSU_j (see Fig. 3 which show the RSUs' regional regulation), performs the following steps.

1. Chooses $r_i \in Z_q^*$ and computes $R_i=r_iP$, $h_i=H_3(m_i||Q_i||pk_i||R_i)$;
2. Computes $W=H_2(P_{pub}, N_{rsu_j})$, $T=H_2(T_{pub}, N_{rsu_j})$, $S_i=D_i+x_iW+h_ir_iT$;
3. Outputs $\sigma_i=(R_i, S_i)$ as the signature on m_i.

ASV: After receiving n signatures $\{m_i, \sigma_i=(R_i, S_i)\}_{i=1}^n$ from n distinct vehicles, RSU_j first aggregate n signatures into one. The RSU_j inputs params and computes $h_i=H_3(m_i||Q_i||pk_i||R_i)$, $1 \le i \le n$. Then RSU_j computes $R=\sum_{i=1}^n h_iR_i$, $S=\sum_{i=1}^n R_i$. Finally, it outputs the aggregate signature $\sigma = (R, S)$. Secondly, the RSU_i verify this aggregate signature $\sigma=(R, S)$ is signed by n vehicles with pseudo identities $\{ID_i\}_{i=1}^{n}$ and corresponding public key $\{pk_i\}_{i=1}^{n}$ on message $\{m_i\}_{i=1}^{n}$, he will do the following steps:

1. Computes $W=H_2(P_{pub}, N_{rsu_j})$, $T=H_2(T_{pub}, N_{rsu_j})$, $Q_i=H_1(ID_{i1}||ID_{i2}||pk_i)$, $1\leq i \leq n$;
2. Verifies $e(S, P) \stackrel{?}{=} e(\sum_{i=1}^{n} Q_i, P_{pub})e(\sum_{i=1}^{n} pk_i, W)e(R, T)$; (1)

If the Eq. (1) holds, the algorithm outputs *true*. Otherwise, it outputs *false*.

Remark 2: Many schemes in [20, 21, 23, 26, 27, 29] require certain synchronization like state information so that all vehicles must share the same state information to generate an aggregate signature. As Horng *et al.* [1] said it is not easy to achieve synchronized in many computing scenarios. Clearly, our scheme does not require the synchronization of aggregated state information. But in [32], a recent new scheme shows a way to achieve such synchronization in VANETs.

4. SECURITY PROOF OF OUR PROPOSED SCHEME

4.1 Correctness

This equation shows that our SCLAS scheme satisfies correctness,

$$\begin{aligned} e(S,P) &= e(\sum_{i=1}^{n} D_i + x_i W + h_i r_i T, P) = e(\sum_{i=1}^{n} D_i, P) e(\sum_{i=1}^{n} x_i W, P) e(\sum_{i=1}^{n} h_i r_i T, P) \\ &= e(\sum_{i=1}^{n} Q_i, P_{pub}) e(\sum_{i=1}^{n} pk_i, W) e(R,T). \end{aligned} \tag{2}$$

4.2 Security Proof

Theorem 1: In the random oracle model, if there exists a type I adversary $\mathcal{A}_1$ who has an advantage ε in forging a valid aggregate signature of our scheme in an attack modeled by Game 1 within a time t for a security parameter k, after $\mathcal{A}_1$ asking at most q_{H_i} times H_i (i = 1, 2, 3) queries, q_p times Partial-Private-Key-Extract queries, q_{pk} times Public-Key queries, q_{sv} times Secret-Value queries, q_s times Sign queries, then the CDHP can be solved within time $t' \leq t+O(2q_{H_1}+2q_{H_2}+2q_p+q_{pk}+q_{sv}+5q_s+2n+1)t_{sm}$ and with the probability $\varepsilon' \geq \frac{1}{(q_p+n)\, e}\varepsilon$ (e is the natural base), where t_{sm} is the time to compute a scalar multiplication in G_1, n is the size of the aggregating set.

Proof: We will describe how $\mathcal{C}$ can use $\mathcal{A}_1$ as a subroutine to solve a given instance (P, aP, bP) of CDHP in G_1 in the following.

Setup: Firstly, $\mathcal{C}$ sets $P_{pub}=aP$ and selects params$=\{q, G_1, G_2, e(.,.), P, P_{pub}, T_{pub}, H_0, H_1, H_2, H_3\}$, then he sends params to $\mathcal{A}_1$.

Attack: The adversary $\mathcal{A}_1$ can perform a polynomially bounded number of the following types of queries in an adaptive manner. $\mathcal{C}$ maintains four lists L_{H_1}, L_{H_2}, L_{H_3}, L_{pk} that are initially empty, $\mathcal{C}$ simulates three Hash oracles and VKG oracle.

***H_1* queries:** $\mathcal{C}$ maintains a list L_{H_1} of tuples (ID_k, pk_k, α_k, Q_k, D_k, c_k). When $\mathcal{A}_1$ initiates a H_1 query on ID_i, if the request has been asked before, $\mathcal{C}$ returns the same answer from the list L_{H_1}. Otherwise, $\mathcal{C}$ first randomly picks $\alpha_i \in Z_q^*$, then flips a coin $c_k \in \{0,1\}$ that yields 0 with probability δ and 1 with probability $1-\delta$ (δ will be determined later). If c_i=0, $\mathcal{C}$ sets $Q_i=\alpha_i bP$, $D_i=\perp$; Otherwise, sets $Q_i=\alpha_i P$, $D_i=\alpha_i aP$; Finally, $\mathcal{C}$ adds (ID_k, pk_k, α_k, Q_k, D_k, c_k) to L_{H_1}, returns Q_i as answer.

***H_2* queries:** $\mathcal{C}$ maintains a list L_{H_2} of tuples $\{(P_{pub}, RSU_k, \beta_k, W_k), (T_{pub}, RSU_k, \gamma_k, T_k)\}$. On receiving a query $H_2(P_{pub}, N_{rsu_i})$ or $H_2(T_{pub}, N_{rsu_i})$, the same answer from the list L_{H_2} will be given if the request has been asked before. Otherwise, $\mathcal{C}$ selects randomly β_i, $\gamma_k \in Z_q^*$, computes $W_i=\beta_i aP$ or $T_i=\gamma_i P$, adds (P_{pub}, RSU_i, β_i, W_i) or (T_{pub}, RSU_i, γ_i, T_i)} to L_{H_2} and returns W_i or T_i as answer.

***H_3* queries:** $\mathcal{C}$ maintains a list L_{H_3} of tuples (m_k, Q_k, pk_k, R_k, h_k). When $\mathcal{A}_1$ launches a query (m_i, Q_i, pk_i, R_i) to H_3, if the request has been asked before, $\mathcal{C}$ returns the same answer from the list L_{H_3}. Otherwise, $\mathcal{C}$ selects randomly $h_i \in Z_q^*$, adds (m_i, Q_i, pk_i, R_i, h_i) to L_{H_3} and returns h_i as answer.

PPKE queries: If $\mathcal{A}_1$ issues a PPKE query on ID_i, $\mathcal{C}$ makes an H_1 query on ID_i and finds the tuple (ID_i, pk_i, α_i, Q_i, D_i, c_i) on L_{H_3}. If c_i=0, $\mathcal{C}$ returns $\perp$. Or else, $\mathcal{C}$ returns D_i.

Public-Key queries: $\mathcal{C}$ maintains a list L_{pk} of tuples (ID_k, x_k, pk_k, d_k). Whenever $\mathcal{A}_1$ issues a Public-Key query on ID_i the same answer from the list L_{pk} will be given if the request has been asked before. Otherwise, $\mathcal{C}$ selects randomly $x_i \in Z_q^*$, computes $pk_i=x_iP$, sets d_i:= 0 (d_i denotes the times of public key replacement), adds (ID_i, x_i, pk_i, d_i) to L_{pk} and returns pk_i as answer.

Secret-Value queries: When $\mathcal{A}_1$ issues a Secret-Value query on ID_i, $\mathcal{C}$ first makes a Public-Key query on ID_i and finds the tuple (ID_i, x_i, pk_i, d_i) on L_{pk}. If d_i=0, $\mathcal{C}$ returns x_i, otherwise, $\mathcal{C}$ returns $\perp$.

Public-Key-Replacement queries: When $\mathcal{A}_1$ issues a Public-Key-Replacement query on ID_i, $\mathcal{C}$ first makes a Public-Key query on ID_i and finds the tuple (ID_i, x_i, pk_i, d_i) on L_{pk}, then $\mathcal{C}$ replaces pk_i with pk_i' chosen by $\mathcal{A}_1$ and puts $d:=d+1$. $\mathcal{C}$ returns pk_i'.

Sign queries: When $\mathcal{A}_1$ issues a Sign query on tuple (m_i, ID_i, pk_i, RSU_i), $\mathcal{C}$ finds tuple (ID_i, pk_i, α_i, Q_i, D_i, c_i), (P_{pub}, RSU_i, β_i, W_i) and (T_{pub}, RSU_i, γ_i, T_i) from L_{H_1} and L_{H_2}.

1. If c_i=0, $\mathcal{C}$ selects a random $r_i \in Z_q^*$ computes $R_i=r_iP-Q_i$, sets $h_i=\gamma_i^{-1}$, adds (m_i, Q_i, pk_i, R_i, h_i) to L_{H_3}. Then $\mathcal{C}$ sets $T_i=\gamma_i P_{pub}$, computes $S_i=\beta_i pk_i+r_iP_{pub}$;
2. If c_i=1, $\mathcal{C}$ executes Sign algorithm in the normal way and returns what the Sign algorithm returns. Finally, $\mathcal{C}$ returns $\sigma_i=(R_i, S_i)$.

Forgery: Eventually, $\mathcal{A}_1$ outputs a tuple (m^*, ID^*, RSU^*, σ^*) in which $m^* = (m_1^*, m_2^*, \ldots, m_n^*)$, $ID^* = (ID_1^*, ID_2^*, \ldots, ID_n^*)$, and σ^* is an aggregate signature . It satisfies that: (1) σ^* is a valid aggregate signature on messages m^* under identities (ID_1^*, ID_2^*, …, ID_n^*) and the corresponding public keys $(pk_1^*, pk_2^*, \ldots, pk_n^*)$; (2) At least one of the identities, say $ID_1^* \in ID^*$ has not been queried during the PPKE queries. And the (m_1^*, ID_1^*) has never been queried during the Sign queries.

For all $1 \le i \le n$, C finds tuple $(ID_i^*, pk_i^*, \alpha_i^*, Q_i^*, D_i^*, c_i^*)$ from L_{H_1}, $(P_{pub}, RSU^*, \beta^*, W^*)$ and $(T_{pub}, RSU^*, \gamma^*, T^*)$ from L_{H_2}. If $c_1 = 0$ and $c_i = 1$, $i = 2, 3, \ldots, n$, C continues. Otherwise, C aborts. The forged signature $\sigma^* = (R^*, S^*)$ must satisfy Eq. (1).

$$e(Q_1^*, P_{pub}) = e(S^*, P)e(\textstyle\sum_{i=2}^{n} Q_i^*, -P_{pub})e(\sum_{i=1}^{n} pk_i^*, -W^*)e(R^*, -T^*) \tag{3}$$

By our setting, $Q_1^* = \alpha_1^* bP$, $W^* = \beta^* P$, $T = \gamma^* P$ and for all $2 \le i \le n$, $Q_i^* = \alpha_i^* P$, hence, C can compute $abP = (\alpha_i^*)^{-1}(S^* - \sum_{i=2}^{n} \alpha_i^* P_{pub} - \sum_{i=2}^{n} \beta^* pk_i - \gamma^* R^*)$. To complete the proof, we shall show that C solve the given instance of CDHP with probability $\varepsilon' \ge \frac{1}{(q_p+n)e}\varepsilon$. First we analyze the three events for C to succeed:

E_1: C does not abort any $\mathcal{A}_1$'s PPKE queries;
E_2: $\mathcal{A}_1$ generates a valid and nontrivial forged aggregate signature.
E_3: Event E_2 occurs, and $c_1^* = 0$ and $c_i^* = 1$ for all i, $2 \le i \le n$.

C succeeds if all the above events happen. The probability $\Pr[E_1 \wedge E_2 \wedge E_3]$ can be decomposed as $\Pr[E_1 \wedge E_2 \wedge E_3] = \Pr[E_1]\Pr[E_2|E_1]\Pr[E_3|E_1 \wedge E_2]$.

Claim 1: As $\Pr[c_i=1]=1-\delta$, the probability that C doesn't abort for a PPKE query is $1-\delta$. Since $\mathcal{A}_1$ makes at most q_p times to the PPKE oracle, the probability that C does not abort any $\mathcal{A}_1$'s PPKE queries is at least $(1-\delta)^{q_p}$. Hence we have $\Pr[E_1] \ge (1-\delta)^{q_p}$.

Claim 2: Suppose C does not abort any $\mathcal{A}_1$'s signature queries and PPKE extraction queries, then $\mathcal{A}_1$'s view is the same as the view in the real attack. Hence, $\Pr[E_2|E_1]=\varepsilon$.

Claim 3: Suppose Event E_2 occurs, C will abort unless $\mathcal{A}_1$ generates a forgery such that $c_1=0$ and $c_i=1$, $i = 2, 3, \ldots, n$. Hence we have $\Pr[E_3|E_1 \wedge E_2] \ge \delta(1-\delta)^{n-1}$. Totally, we have $\varepsilon = \Pr[E_1 \wedge E_2 \wedge E_3] \ge (1-\delta)^{q_p}\varepsilon\delta(1-\delta)^{n-1} = \delta(1-\delta)^{q_p+n-1}\varepsilon$ when $\delta = \frac{1}{q_p+n}$, $\delta(1-\delta)^{q_p+n-1}$ is maximized at $\frac{1}{q_p+n}(1-\frac{1}{q_p+n})^{q_p+n-1}$. When q_p is sufficient large, this probability approaches $\frac{1}{(q_p+n)e} \cdot \varepsilon$. Hence we have $\varepsilon \ge \frac{1}{(q_p+n)e} \cdot \varepsilon$.

The running time t for C is the sum of $\mathcal{A}_1$'s running time, the time that C answers queries and C computes the CDHP instance. During each H_1 query, H_2 query, PPKE query, Public-Key query, Secret-Value query and Sign query, it needs 2,2,2,1,1,5 scalar multiplications respectively. And during C computing the CDHP instance, it needs $2n+1$ scalar multiplication. So $t' \le t + O(2q_{H_1} + 2q_{H_2} + 2q_p + q_{pk} + q_{sv} + 5q_s + 2n + 1)t_{sm}$.

Theorem 2: In the random oracle model, if there exists a type II adversary $\mathcal{A}_2$ who has an advantage ε in forging a signature of our scheme in an attack modeled by Game 2 within a time span t for a security parameter k, after asking at most q_{H_i} times $H_i (i = 2, 3)$ queries, q_{pk} times Public-Key queries, q_{sv} times Secret-Value queries, q_s times Sign queries. Then the CDHP can be solved within time $t' \le t + O(2q_2 + q_{pk} + q_{sv} + 5q_s + 2n + 1)t_{sm}$ and with the probability $\varepsilon' \ge \frac{1}{(q_p+n)e} \cdot \varepsilon$ (e is the natural base).

Proof: The following will show how $\mathcal{C}$ can use $\mathcal{A}_2$ as a subroutine to solve a given instance (P, aP, bP) of CDHP in G_1.

Setup: Firstly, C chooses a random $s \in Z_q^*$ as the msk_K, sets $P_{pub} = sP$ and selects params $= \{q, G_1, G_2, e, P, P_{pub}, T_{pub}, H_0, H_1, H_2, H_3\}$, then he sends params and s to $\mathcal{A}_2$. Since $\mathcal{A}_2$ gets $\boldsymbol{s}$ and can do PPKE by himself and it don't need H_1 queries.

Attack: The adversary $\mathcal{A}_2$ can perform a polynomially bounded number of the following types of queries in an adaptive manner. $\mathcal{C}$ keeps three lists L_{H_2}, L_{H_3} and L_{pk} to simulate hash oracles H_2, H_3 and VKG oracle, they are initially empty.

***H*₂ queries**: $\mathcal{C}$ maintains a list L_{H_2} of tuples $\{(P_{pub}, RSU_k, \beta_k, W_k), (T_{pub}, RSU_k, \gamma_k, T_k)\}$. On receiving a query $H_2(P_{pub}, N_{rsu_i})$ or $H_2(T_{pub}, N_{rsu_i})$, the same answer from the list L_{H_2} will be given if the request has been asked before. Otherwise, $\mathcal{C}$ selects randomly β_i, $\gamma_i \in Z_q^*$, computes $W_i = \beta_i aP$ or $T_i = \gamma_i P$, adds $(P_{pub}, RSU_i, \beta_i, W_i)$ or $(T_{pub}, RSU_i, \gamma_i, T_i)$ to L_{H_2} and returns W_i or T_i as answer.

***H*₃ queries:** C maintains a list L_{H_3} of tuples $(m_k, Q_k, pk_k, R_k, h_k)$. When $\mathcal{A}_2$ issues a query (m_i, Q_i, pk_i, R_i) to H_3, if the request has been asked before, $\mathcal{C}$ returns the same answer from the list L_{H_3}. Otherwise, $\mathcal{C}$ selects randomly $h_i \in Z_q^*$, adds $(m_i, Q_i, pk_i, R_i, h_i)$ to L_{H_3} and returns h_i as answer.

Public-Key queries: C maintains a list L_{pk} of tuples (ID_k, x_k, pk_k, c_k). Whenever $\mathcal{A}_2$ issues a Public-Key query on ID_i, the same answer from the list L_{pk} will be given if the request has been asked before. Otherwise, $\mathcal{C}$ first selects random $x_i \in Z_q^*$, then flips a coin $c_i \in \{0, 1\}$ that yields 0 with probability δ and 1 with probability $1-\delta$. If $c_i=0$, $\mathcal{C}$ sets $pk_i=x_i bP$; if $c_i=1$, $\mathcal{C}$ sets $pk_i=x_iP$. Finally $\mathcal{C}$ adds (ID_i, x_i, pk_i, c_i) to L_{pk} and returns pk_i.

Secret-Value queries: When $\mathcal{A}_2$ issues a Secret-Value query on ID_i, $\mathcal{C}$ first makes a Public-Key query on ID_i and finds the tuple (ID_i, x_i, pk_i, c_i) on L_{pk}. If $c_i=1$, $\mathcal{C}$ returns x_i. Otherwise, $\mathcal{C}$ returns $\perp$.

Sign queries: When $\mathcal{A}_2$ issues a Sign query on tuple (m_i, ID_i, pk_i, RSU_i), $\mathcal{C}$ finds tuple $(P_{pub}, RSU_i, \beta_i, W_i)$ and $(T_{pub}, RSU_i, \gamma_i, T_i)$ from L_{H_2}. Then $\mathcal{C}$ performs as follows:

1. If $c_i=0$, $\mathcal{C}$ selects a random $r_i \in Z_q^*$; computes $R_i=r_iP-\beta_i pk_i$, sets $h_i=\gamma_i^{-1}$, adds $(m_i, Q_i, pk_i, R_i, h_i)$ to L_{H_3}. Then $\mathcal{C}$ sets $T_i=\gamma_i aP$ and computes $S_i=D_i+r_i\, aP$;
2. If $c_i=1$, $\mathcal{C}$ executes **Sign** algorithm in the normal way and returns what the **Sign** algorithm returns. At last, $\mathcal{C}$ returns $\sigma_i=(R_i, S_i)$.

Forgery: $\mathcal{A}_2$ outputs a four-tuple $(m^*, ID^*, RSU^*, \sigma^*)$ in which $m^* = (m_1^*, m_2^*, \ldots, m_n^*)$, $ID^* = (ID_1^*, ID_2^*, \ldots, ID_n^*)$ and σ^* is a valid aggregate signature. $\mathcal{A}_2$ wins the Game.

For all $1 \le i \le n$, $\mathcal{C}$ finds tuple $(P_{pub}, RSU^*, \beta^*, W^*)$ and $(T_{pub}, RSU^*, \gamma^*, T^*)$ from L_{H_2}, and $(m_i^*, ID_i^*, pk_i^*, h_i^*)$ from L_{H_3}. If $c_1=0$ and $c_i=1$, $2 \le i \le n$, $\mathcal{C}$ continues; otherwise, $\mathcal{C}$ aborts. Since the forged $\sigma^*=(R^*, S^*)$ must satisfies Eq. (1), we have

$$e(pk_1^*, W^*) = e(S^*, P)e(\sum_{i=1}^{n} D_i^*, -P)e(\sum_{i=2}^{n} pk_i^*, -W^*)e(R^*, -T^*). \quad (4)$$

By our setting, $pk_1^* = x_2^* bP$, $T^* = \gamma^* P$ and for $2 \le i \le n$, $pk_i^* = x_i^* P$. Hence, $\mathcal{C}$ can compute $abP = (x_i^* \beta)^{-1} \cdot (S^* - \sum_{i=1}^{n} D_i^* - \sum_{i=2}^{n} x_i^* W^* - \gamma^* R^*)$.

Three events needed for $\mathcal{C}$ to succeed are following:

E_1: $\mathcal{C}$ does not abort as a result of any of $\mathcal{A}_2$'s Secret-Value queries.
E_2: $\mathcal{A}_2$ generates a valid and nontrivial forged aggregate signature.
E_3: Event E_2 occurs, and $c_1^* = 0$ and $c_i^* = 1$ for all i, $2 \le i \le n$.

The rest proof is very similar with Game 1. $\mathcal{C}$ can solve the given instance of CDHP with probability $\varepsilon' \ge \frac{1}{(q_p+n)e} \cdot \varepsilon$ within time $t' \le t + O(2q_2 + q_{pk} + q_{sv} + 5q_s + 2n + 1)t_{sm}$. Due to the limited space, we omit it here.

According to Theorems 1 and 2, we can get the conclusion: Our SCLAS scheme is $(t, \varepsilon, n, q_H, q_p, q_{pk}, q_{sv}, q_s)$-secure against existential forgery under adversaries $\mathcal{A}_1$ and $\mathcal{A}_2$ adaptively choosing message and identity attack.

5. PERFORMANCE EVALUATION

In this section, we first compare the computational costs of our scheme with some existing CLAS scheme [23, 24, 26-29] in Table 1, where we omit the computations which take little time such as Hash algorithm *etc.* From Table 1, the whole computation is mainly composed by A-V cost and Sign cost. From overall perspective, our scheme has little computation and better efficiency than other schemes in Table 1. And the partial private key of a signer is only one element in G_1, whose length is shorter than that of scheme in [23, 26]. As for the final aggregate signature length, our new scheme only requires two elements in G_1, far shorter than [24, 29], and approximately 320 bits if G_1 is an additive group on elliptic curve with 160 bits. Therefore, our scheme is the most bandwidth-saving and storage-saving simultaneously.

Table 1. Comparisons of computation cost and signature length for six CLAS schemes.

Scheme	Sign cost	A-V cost	A-S size	*sk* size	Security
Scheme in [26]	5 *sm*	$5p$+$2n$ *sm*	$2\|G_1\|$	$2\|G_1\|+q$	×
Scheme in [27]	3 *sm*	$(n+3)p$	$2\|G_1\|$	$\|G_1\|+q$	√
Scheme in [28]	4 *sm*	$4p$+$2n$ *sm*	$2\|G_1\|$	$\|G_1\|+q$	×
Scheme in [29]	4 *sm*	$4p$+$2n$ *sm*	$(n+1)\|G_1\|$	$\|G_1\|+q$	√
Scheme in [23]	4 *sm*	$4p$+$2n$ *sm*	$2\|G_1\|$	$2\|G_1\|+2q$	×
Scheme in [24]	4 *sm*	$3p$+$2n$ *sm*	$(n+1)\|G_1\|$	$\|G_1\|+q$	√
Our Scheme	3 *sm*	$4p$	$2\|G_1\|$	$\|G_1\|+q$	√

1. ***p***: Computation cost of pairing operation $e(\cdot, \cdot)$; 2. ***sm***: Computation cost of scalar multiplications in $G1$; 3. **A-V cost**: Computation cost of aggregate signature verification; 4. **sign cost**: Computation cost of signature generation; 5. $|G_1|$: Size of one element in G_1; 6. ***q***: Size of one element in Z_q^*; 7. **A-S size**: Size of an aggregate signature; 8. ***sk* size**: Size of user's secret key.

Next, we evaluate the efficiency on applying the proposed schemes **for VANETs** [1, 8, 33, 34] in Table 2. We adopt the experiment in [35, 36], which observes processing time for Tate pairing on a 159-bit subgroup of an MNT cure with an embedding degree 6 at an 80-bit security level, running on an Intel i7 3.07 GHz machine, the following result is obtained: p is 3.21 ms and sm is 0.39 ms. The computational cost of our scheme is dominant to pairing operation, which does not linearly increase with the number of aggregated signatures. Then, with the increasing of the vehicles numbers within a RSU's radiation range, the RSU's verification cost is constant. We show our advantage in Fig. 4.

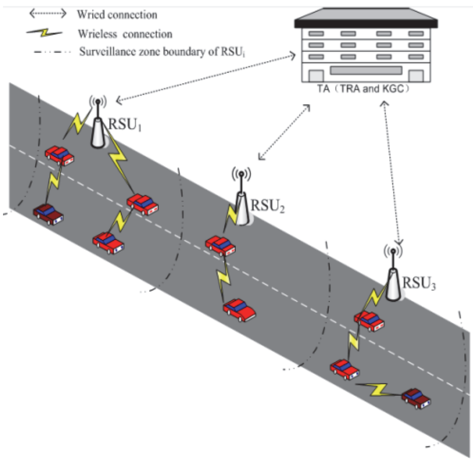

Fig. 3. A diagram of RSUs regional regulation.

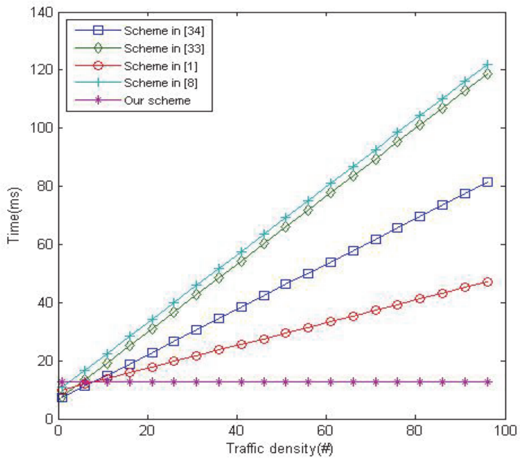

Fig. 4. Verification delay vs. traffic density in VANET-based schemes.

Table 2. The comparison of verification overhead of related schemes in VANETs.

Schemes	Scheme in [34]	Scheme in [33]	Scheme in [1]	Scheme in [8]	Our SCLAS
Verify a signature	$2p$+2 sm	$2p$+sm	$3p$+ sm	$3p$+ $3n$ sm	$4p$
Verify n signatures	$2p$+$2n$ sm	$2p$+$3n$ sm	$3p$+ n sm	$3p$+ $3n$ sm	$4p$

6. CONCLUSIONS

In this paper, a novel and efficient certificateless aggregate signature scheme is presented for vehicle communications. The proposed scheme is proven existentially unforgeable against adaptive chosen-message attacks and chosen-identity attacks in the random model assuming that the DLP and CDHP are hard. The new aggregate signature consists of only two group elements which significantly saves storage space. Our scheme is designed specifically for securing vehicle communication in VANETs by reducing the signature verification time drastically and verifying more messages in specific stipulated time, thus increasing the efficiency of communication. So it is more suitable for the applications in bandwidth-limited, computing-limited and storage-limited mobile devices and scenarios such as VANETs.

REFERENCES

1. S. J. Horng, S. F. Tzeng, P. H. Huang, *et al.*, "An efficient certificateless aggregate sig nature with conditional privacy-preserving for vehicular sensor networks," *Information Sciences*, Vol. 317, 2015, pp. 48-66.
2. Q. Wu, J. Domingo-Ferrer, and U. Gonzalez-Nicolas, "Balanced trustworthiness, safety, and privacy in vehicle-to-vehicle communications," *IEEE Transactions on Vehicular Technology*, Vol. 59, 2010, pp. 559-573.
3. C. Gamage, B. Gras, B. Crispo, *et al.*, "An identity-based ring signature scheme with enhanced privacy," *IEEE Securecomm and Workshops*, 2006, pp. 1-5.
4. X. Lin, X. Sun, P. Ho, *et al.*, "GSIS: a secure and privacy-preserving protocol for vehicular communications," *IEEE Transactions on Vehicular Technology*, 2007, Vol. 56, pp. 3442-3456.
5. R. Lu, X. Lin, H. Zhu, *et al.*, "ECPP: Efficient conditional privacy preservation protocol for secure vehicular communications," in *Proceedings of the 27th IEEE Conference on Computer Communications*, 2008, pp. 1229-1237.
6. A. Studer, E. Shi, F. Bai, *et al.*, "Tacking together efficient authentication, revocation, and privacy in VANETs," in *Proceedings of the 6th Annual IEEE Communications Society Conference on Sensor*, *Mesh and Ad Hoc Communications and Networks*, 2009, pp. 1-9.
7. X. Zhu, S. Jiang, L. Wang, *et al.*, "Efficient privacy-preserving authentication for vehicular ad hoc networks," *IEEE Transactions on Vehicular Technology*, 2014, Vol. 63, pp. 907-919.
8. A. Malhi and S. Batra, "An efficient certificateless aggregate signature scheme for vehicular ad-hoc networks," *Discrete Mathematics and Theoretical Computer Science*, Vol. 17, 2015, pp. 317-338.
9. T. Zhang and L. Delgrossi, "Vehicle safety communications: protocols, security, and privacy," Wiley online library, September, 2012.
10. D. Boneh, C. Gentry, B. Lynn, *et al.*, "Aggregate and verifiably encrypted signatures from bilinear maps," *Advances in Cryptology*, Springer, Berlin, Heidelberg, 2003, pp. 416-432.
11. J. Chen, H. Yue, and Z. Huang, "Secure certificate-based aggregate signature scheme," *Computer Engineering and Applications*, 2013, Vol. 49, pp. 60-64.
12. J. Li, X. Zhao, Y. Zhang, *et al.*, "Provably secure certificate-based conditional proxy re-encryption," *Journal of Information Science and Engineering*, Vol. 32, 2016, pp. 813-830 .
13. J. Song, H. Kim, S. Lee, *et al.*, "Security enhancement in ad hoc network with ID-based cryptosystem," in *Proceedings of the 7th IEEE International Conference on Advanced Communication Technology*, Vol. 1, 2005, pp. 372-376.
14. C. Gentry and Z. Ramzan, "Identity-based aggregate signatures," in *Proceedings of PKC*, LNCS 3958, 2006, pp. 257-273.
15. Y. Wen, J. Ma, and C. Wang, "New ID-based aggregate signature scheme," *Computer Science*, Vol. 6, 2011, p. 014.
16. X. Cheng, J. Liu, and X. Wang, "An ID-based aggregate signature scheme from *m*-torsion groups," *Journal of Xidian University*, Vol. 32, 2005, pp. 427-431.

17. Y. Yu, X. Zheng, and H. Sun, "An identity based aggregate signature scheme from pairing," *Journal of Networks*, Vol. 6, 2011, pp. 631-637.
18. S. Al-Riyami and K. Paterson, "Certificateless public key cryptography," *Advances* in *Cryptology-ASIACRYPT*, 2003, pp. 452-473.
19. L. Zhang, Q. Wu, *et al*., "Distributed aggregate privacy-preserving authentication in VANETs," *IEEE Transactions on Intelligent Transportation Systems*, DOI: 10.1109/ TITS.2016.2579162.
20. Z. Gong, Y. Long, X. Hong, *et al*., "Two certificateless aggregate signatures from bilinear maps," in *Proceedings of the 8th IEEE ACIS International Conference on Software Engineering*, *Artificial Intelligence*, *Networking*, *and Parallel/Distributed Computing*, 2007, Vol. 3, pp. 188-193.
21. L. Zhang and F. Zhang, "A new certificateless aggregate signature scheme," *Computer Communications*, Vol. 39, 2009, pp. 1079-1085.
22. H. Xiong, Z. Guan, Z. Chen, *et al.*, "An efficient certificateless aggregate signature with constant pairing computations," *Information Sciences*, Vol. 219, 2013, pp. 225-235.
23. Y. Chen, R. Tso, M. Mambo, *et al.*, "Certificateless aggregate signature with efficient verification". *Security and Communication Networks*, Vol. 8, 2015, pp. 2232-2243.
24. L. Cheng, Q. Wen, *et al.*, "Cryptanalysis and improvement of a certificateless aggregate signature scheme," *Information Science*, Vol. 295, 2015, pp. 337-346.
25. H. Lu, X. Yu, and Q. Xie, "Provably secure certificateless aggregate signature with constant length," *Journal of Shanghai Jiaotong University*, 2012, Vol. 46, p. 016.
26. L. Zhang, B. Qin, Q. Wu, *et al.*, "Efficient many-to-one authentication with certificateless aggregate signature," *Computer Network*, Vol. 54, 2010, pp. 2482-2491.
27. M. Zhou, M. Zhang, C. Wang, *et al.*, "CCLAS: A pPractical and compact certificateless aggregate signature with share extraction," *International Journal of Network Security*, Vol. 16, 2014, pp. 174-181.
28. H. Du, M. Huang, and Q. Wen, "Efficient and provably-secure certificateless aggregate signature scheme," *Acta Electronica Sinica*, Vol. 54, 2013, pp. 2482-2491.
29. H. Chen, S. Wei, C. Zhu, *et al.*, "Secure certificateless aggregate signature scheme," *Journal of Software*, Vol. 26, 2015, pp. 1173-1180.
30. J. Zhang, X. Zhao, and J. Mao, "Attack on Chen *et al.*'s certificateless aggregate signature scheme," *Security and Communication Networks*, Vol. 9, 2016, pp. pp. 54-59.
31. M. Bellare and P. Rogawaym, "Random oracles are practical: a paradigm for designing efficient protocol," in *Proceedings of the 1st ACM Conference on Computer and Communications Security*, 1993, pp. 62-73.
32. L. Zhang, C. Hu, Q. Wu, *et al.*, "Privacy-preserving vehicular communication authentication with hierarchical aggregation and fast response," *IEEE Transactions on Computers*, Vol. 65, 2016, pp. 2562-2574.
33. H. Wang, B. Qin, and J. Domingo-Ferrer, "An improved binary authentication tree algorithm for vehicular networks," in *Proceedings of the 4th IEEE International Conference on Intelligent Networking and Collaborative Systems*, 2012, pp. 206-213.
34. S. Horng, S. Tzeng, Y. Pan, *et al.*, "*b*-SPECS+: batch verification for secure pseudonymous authentication in VANET," *IEEE Transactions on Information Forensics and Security*, Vol. 8, 2013, pp. 1860-1875.

35. K. A. Shim, "CPAS: an efficient conditional privacy-preserving authentication scheme for vehicular sensor networks," *IEEE Transactions on Vehicular Technology*, Vol. 61, 2012, pp. 1874-1883.
36. Z. Liu, H. Seo, J. Großschädl, *et al.*, "Efficient implementation of NIST-compliant elliptic curve cryptography for 8-bit AVR-based sensor nodes," *IEEE Transactions on Information Forensics and Security*, Vol. 11, 2016, pp. 1385-1397.

Hao-Hao Nie (聂好好) received her B.S. degree from Yuncheng University in 2014. She now is an M.S. degree candidate in Applied Mathematics with the School of Mathematics and Information Science, Shaanxi Normal University, Xi'an, China. Her research interests include aggregate signature scheme and its applications.

Yan-Ping Li (李艳平) received her M.S. degree from Shaanxi Normal University in 2004 and Ph.D. degree from Xidian University in 2009, Xian, China. She now is an Associate Professor with the School of Mathematics and Information Science, Shaanxi Normal University. Her research interests include public key cryptography and its applications.

Qian-Hong Wu (伍前红) received the M.S. degree in Applied Mathematics from Si Chuan University, Sichuan, China, and the Ph.D. degree in Cryptography from Xidian University, Xi'an, China, in 2001 and 2004, respectively. His research interests include cryptography, information security and privacy, and ad hoc network security.