Efficient Privacy-Preserving Authentication for Vehicular Ad Hoc Networks

Xiaoyan Zhu, Shunrong Jiang, Liangmin Wang, and Hui Li

***Abstract*—In this paper, we present an efficient privacy-preserving authentication scheme based on group signature for vehicular ad hoc networks (VANETs). Although group signature is widely used in VANETs to realize anonymous authentication, the existing schemes based on group signatures suffer from long computation delay in the certificate revocation list (CRL) checking and in the signature verification process, leading to high message loss. As a result, they cannot meet the requirement of verifying hundreds of messages per second in VANETs. In our scheme, we first divide the precinct into several domains, in which roadside units (RSUs) are responsible for distributing group private keys and managing vehicles in a localized manner. Then, we use a hash message authentication code (HMAC) to avoid time-consuming CRL checking and to ensure the integrity of messages before batch group authentication. Finally, we adopt cooperative message authentication among entities, in which each vehicle only needs to verify a small number of messages, thus greatly alleviating the authentication burden. The security and performance analysis show that our scheme is more efficient in terms of authentication speed, while keeping conditional privacy in VANETs.**

***Index Terms*—Batch group signature, cooperation, hash message authentication code (HMAC), privacy-preserving authentication, vehicular ad hoc networks (VANETs).**

I. Introduction

With the massive development of wireless communications, ad hoc networking, and Internet of Things, vehicular ad hoc networks (VANETs) have attracted extensive attention and research efforts from academia, industry, and governments in recent years. In a general setting, a VANET is composed of three components: onboard units (OBUs) equipped in mobile vehicles, fixed roadside units (RSUs), and a central trust authority (TA). Being aware of the traffic condition, such as vehicles' position, speed, direction, etc., VANETs are expected to improve the driving experience, traffic safety, and multimedia infotainment dissemination for drivers and passengers [1]. In VANETs, vehicles communicate with each other, as well as with RSUs, through an open wireless channel, in which attackers can easily get users' private information, such as identity, tracing, preference, etc., if they are not properly protected [2]. Another characteristic of VANETs is high-speed mobility, leading to limited communication time among RSUs and vehicles. As a result, we need to design an efficient authentication scheme with privacy preservation for VANETs.

In VANETs, group signature [3] is widely used for vehicles to achieve anonymous authentication [4]–[7] since it allows any group member to sign a message on behalf of the group without revealing its real identity. When receiving a message from an unknown entity, a vehicle has to check the certificate revocation list (CRL) to avoid communicating with revoked vehicles and then verify the sender's group signature to check the validity of the received message. Unfortunately, it needs 11 ms [8] to verify a message attached with a group signature and 9 ms [1] to check one identity in the CRL. If there are n revoked identities in the CRL, the number of messages that can be verified in one second is $1000/(9n+1)$, which is far smaller than the requirement of 600 [1]. Furthermore, if we consider value-added services [9]–[11], extra time for verification and decryption is needed. Therefore, we should try to reduce the delay caused by the CRL checking and group signature verification to achieve the rapid authentication.

To address the CRL checking problem, Wasef and Shen [12] and Jiang *et al.* [13] use hash message authentication coding (HMAC) to replace the CRL, greatly reducing the checking time. However, both of them are based on pseudonyms, which may not fit group signature based schemes directly. To reduce the signature verification time, Wasef and Shen [6] and Zhang *et al.* [7] employ batch group signature verification, in which a large number of messages can be authenticated in a timely manner. However, if there exist a few invalid messages, they may introduce additional verification delay for a rebatch and then lose their efficiency. Even without considering the rebatch time, the authentication schemes based on batch group signatures [6], [7] can only verify 127 and 274 messages per second, respectively, which still cannot satisfy the requirement of verifying 600 messages per second.

Moreover, the aforementioned schemes focus on rapid authentication in a single vehicle. By observing the fact that each vehicle in the same area verifies almost the same set of messages, Zhang *et al.* [8] and Hao *et al.* [14] propose

Manuscript received July 19, 2013; revised November 12, 2013; accepted November 22, 2013. Date of publication December 5, 2013; date of current version February 12, 2014. This work was supported in part by the National Natural Science Foundation of China under Grant 61003300, by the Fundamental Research Funds for the Central Universities under Grant K5051201041, and by the China 111 Project under Grant B08038. The work of H. Li was supported in part by the National Natural Science Foundation of China under Grant 61272457. The review of this paper was coordinated by Prof. Y. Zhang.

X. Zhu, S. Jiang, and H. Li are with the National Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China (e-mail: xyzhu@mail.xidian.edu.cn; jsywow@gmail.com; lihui@mail.xidian.edu.cn).

L. Wang is with the School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang 212013, China (e-mail: wanglm@ujs.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2013.2294032

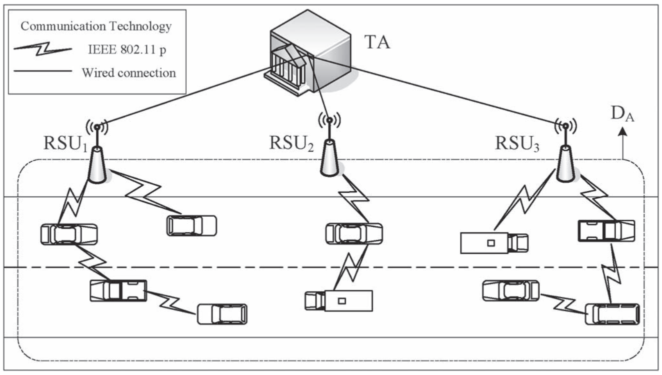


Fig. 1. System model of VANETs.

their schemes based on cooperation among vehicles. Although the Hao *et al.* scheme can achieve the verification speed of 600 messages per second, it does not take account of the CRL checking before signature verification. Therefore, there exists performance degradation in a practical setting.

In this paper, we propose an efficient conditional privacy-preserving authentication scheme for VANETs under the semi-trust model of RSU, by jointly using the techniques of distributed management, HMAC, batch group signature verification, and cooperative authentication. We first divide the precinct into several domains so that the system can run in a localized manner. Then, we calculate HMAC with the group key generated by the self-healing group-key generation algorithm [15], which can replace the time-consuming CRL checking and ensure the integrity of messages before batch verification. We also give a practical setting of the Hao *et al.* cooperative message authentication scheme [14] to improve the efficiency of authentication. The security and performance analysis show that the proposed scheme can achieve more efficient group signature based authentication while keeping conditional privacy for VANETs.

The remainder of this paper is organized as follows. Section II presents the system model and preliminaries. We describe our scheme in Section III, and the cooperative authentication is given in Section IV. We give the security analysis in Section V. Section VI provides performance evaluation of our scheme. Section VII summarizes related works in the literature. Finally, Section VIII concludes this paper.

II. System Model and Preliminaries

A. System Model

The system model of VANETs in this paper consists of a TA, fixed RSUs at the road side, and mobile OBUs equipped in vehicles, as shown in Fig. 1.

- TA is a trusted management center of the network. It provides registration and certification for RSUs and OBUs when they join the network. It also divides the whole precinct into several domains, generates the group key and group signature materials for every domain, and then sends these materials to the RSUs in the domain. As usual, we assume that TA is powerful enough in terms of communication, computation, and storage capability, and it is infeasible for any adversary to compromise.
- RSUs manage and communicate with vehicles in their communication range. They are bridges between TA and end users, which connect with TA by wire and OBUs by a wireless channel. In this paper, we assume RSU to be semi-trust [16], [17], i.e., they can operate as expected but may reveal data to an adversary. RSUs are also responsible for issuing the group key materials and group signature related keys to validate OBUs when OBUs join the domain.
- OBUs periodically broadcast traffic-related status information containing its location, speed, and direction to improve the road environment, traffic safety, and multimedia infotainment dissemination for drivers and passengers. Each vehicle has a tamper-proof device (TPD) to store security-related materials.

$$SD \xrightarrow{h(\bullet)} S_1 \xrightarrow{h(\bullet)} S_2 \cdots\cdots S_{i\text{-}2} \xrightarrow{h(\bullet)} S_{i\text{-}1} \xrightarrow{h(\bullet)} S_i$$

Fig. 2. Hash chain.

Without loss of generality, we do not consider the scenario of vehicles' sharing secret with others in this paper since this type of active attacks cannot be prevented in almost all security systems.

B. Hash Function, Hash Chain, and HMAC

A one-way hash function $h(\cdot)$ is said to be secure if the following properties are satisfied [18].

1) $h(\cdot)$ can take a message of arbitrary length as input and produce a message digest of a fixed-length output.
2) Given x, it is easy to compute $h(x) = y$. However, it is hard to compute $h^{-1}(y) = x$ given y.
3) Given x, it is computationally infeasible to find $x' \neq x$ such that $h(x') = h(x)$.

Furthermore, a hash chain is defined as in Fig. 2, where $S_k = h(S_{k-1})$, $k = 1, 2, \ldots, i$, and $S_0 = SD$, where SD is the initial seed value. According to the definition of the hash function, it is obvious that, given S_k, it is easy to compute $S_{k+1}, S_{k+2}, \ldots, S_i$ but infeasible to compute any one of SD, $S_1, \ldots, S_{k-1}$.

HMAC is used to authenticate the source of a message and its integrity by attaching a message authentication code (MAC) to the message, which is accomplished by a cryptographic keyed hash function (such as MD5, SHA-256). In this paper, we use HMAC for two purposes: 1) ensuring the validity of senders' identities, since only valid users can generate correct HMACs; and 2) checking the integrity of messages before batch verification, thus achieving the efficiency of batch verification. The detailed algorithm process of HMAC can be found in [19].

C. Bilinear Pairing

The properties of the bilinear operation used in this paper are defined as follows [20]: Let $\mathbb{G}_1$ and $\mathbb{G}_2$ denote additive cyclic

TABLE I
NOTATION

Notation	Description
$\|\|$	Message concatenation
R_x	xth RSU
V_i	Real identity of ith vehicle
D_A	Ath domain
H, H_1, H_2, H_3	Hash functions like SHA-1
SK_{TA}, PK_{TA}	Private key and corresponding public key of TA
SK_{R_x}, PK_{R_x}	Private key and corresponding public key of R_x
SK_{V_i}, PK_{V_i}	Private key and corresponding public key of V_i
$Cert_{TA,R_x}$	Certificate of R_x generated by TA
$Cert_{TA,V_i}$	Certificate of V_i generated by TA
GPK_{D_A}	Group public key of domain D_A
GSK_{D_A,V_i}	Group private key of V_i in domain D_A
$\{M\}_K$	Encrypt the plaint message M by key K
$Sig_{SK}(M)$	Sign the message M by private key SK
$Verify(PK, M, \sigma)$	Verify signature σ of M by public key PK

groups, and $\mathbb{G}_T$ denote a multiplicative cyclic group of the same prime order p. Let g_1 be a generator of $\mathbb{G}_1$, g_2 be a generator of $\mathbb{G}_2$, and ψ be an isomorphism from $\mathbb{G}_2$ to $\mathbb{G}_1$ such that $\psi(g_2) = g_1$. $e : \mathbb{G}_1 \times \mathbb{G}_2 \longrightarrow \mathbb{G}_T$ is a bilinear map, which satisfies the following.

1) Bilinear: $e(u^a, v^b) = e(u, v)^{ab}$ for all $u \in \mathbb{G}_1,\ v \in \mathbb{G}_2$ and $a, b \in \mathbb{Z}_p$.
2) Nondegeneracy: $e(g_1,\ g_2) \neq 1_{\mathbb{G}_T}$.
3) Admissible: Map e and isomorphism ψ are efficiently computable.

III. Proposed Scheme

Here, we describe our scheme with the following process: system initiation, RSU's certificate issuing, vehicle's certificate issuing, secure group key distribution and batch authentication, and a periodic update of the group key. The notations used in this paper are listed in Table I.

A. System Initialization

We employ the Schnorr signature algorithm [21] as the underlying signature algorithm employed by TA, RSUs, and OSUs. Here, we use it for its efficiency in the VANETs scenario [2]. In fact, our scheme can be easily changed to use other underlying signature schemes.

According to the Schnorr signature algorithm, TA chooses

- primes p and q such that $q|p-1$, $q \geq 2^{140}$, and $p \geq 2^{512}$;
- $\alpha \in \mathbb{Z}_p$ with order q, i.e., $\alpha^q = 1 \pmod p$, and $\alpha \neq 1$;
- a one-way hash function h: $(0, 1)^* \rightarrow (0, 1)^l$;
- a random number $s \in \mathbb{Z}_q^*$ as its own private key so that $\mathrm{SK_{TA}} = s$.

Then, TA computes its public key $\mathrm{PK_{TA}} = p^s$ and publishes the tuple $(p, q, \alpha, h, \mathrm{PK_{TA}})$ as the system parameters.

B. RSU's Certificate Issuing

TA divides its precinct into a few domains, each of which includes several RSUs. For RSU R_x in domain D_A, TA verifies its identity and issues the certificate $\mathrm{Cert}_{\mathrm{TA},R_x}$ as follows.

1) TA chooses a random number $\mathrm{SK}_{R_x} \in \mathbb{Z}_q^*$ as the private key of R_x and computes the public key $\mathrm{PK}_{R_x} = p^{\mathrm{SK}_{R_x}}$ for R_x.
2) TA generates the signature σ_{TA,R_x}, where $\sigma_{\mathrm{TA},R_x} = \mathrm{Sig}_{\mathrm{SK_{TA}}}(\mathrm{PK}_{R_x} \| D_A)$.
3) TA delivers SK_{R_x} and $\mathrm{Cert}_{\mathrm{TA},R_x}$ to R_x, where $\mathrm{Cert}_{\mathrm{TA},R_x} = (\mathrm{PK}_{R_x} \| D_A, \sigma_{\mathrm{TA},R_x})$. The delivery of SK_{R_x} must be via a secure channel, such as Secure Sockets Layer.

C. Vehicle's Certificate Issuing

For vehicle V_i, TA issues certificate $\mathrm{Cert}_{\mathrm{TA},V_i}$ after verifying its identity as follows.

1) TA chooses a random number $\mathrm{SK}_{V_i} \in \mathbb{Z}_q^*$ as the private key of V_i and computes public key $\mathrm{PK}_{V_i} = p^{\mathrm{SK}_{V_i}}$ for V_i.
2) TA generates the certificate $\mathrm{Cert}_{\mathrm{TA},V_i}$ of V_i, where $\mathrm{Cert}_{\mathrm{TA},V_i} = \mathrm{Sig}_{\mathrm{SK_{TA}}}(\mathrm{PK}_{V_i})$.
3) TA securely delivers SK_{V_i} and $\mathrm{Cert}_{\mathrm{TA},V_i}$ to V_i offline during the vehicle inspection.

D. Secure Group Key Distribution and Batch Authentication

For the domain D_A, TA generates group signature keys, containing the public materials and group public key $(G\mathrm{PK}_{D_A})$. Here, we use Wasef and Shen's scheme [6] for the reality of batch group signature verification. Ferrara *et al.*'s algorithm [22] can be also used.

Given bilinear parameters $(p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e)$, TA generates the group public key as follows.

1) TA selects random generator $g_2 \in \mathbb{G}_2$ and computes $g_1 = \psi(g_2)$, where g_1 is the generator of $\mathbb{G}_1$, and ψ is an isomorphism from $\mathbb{G}_2$ to $\mathbb{G}_1$ such as $g_1 = \psi(g_2)$.
2) TA selects random numbers $h,\ u,\ v \in \mathbb{G}_1$, and selects numbers $s_1,\ s_2 \in \mathbb{Z}_p$, such that $u^{s_1} = v^{s_2} = h$.
3) TA selects random numbers $\gamma \in \mathbb{Z}_p$ and $\lambda \in \mathbb{Z}_p^*$, and sets $\omega = g_2^{\gamma}$.

Here, s_1 and s_2 are master secret keys of domain D_A, which are managed by TA. The public system parameters of domain D_A are $(g_1, g_2, u, v, h, \lambda)$, and its group public key is $G\mathrm{PK}_{D_A} = \omega$. TA sends the public system parameters and the group public key to all RSUs in D_A. Then, the vehicle and the RSU can realize mutual authentication by using these prestored materials. When a vehicle V_i joins a new domain D_A, it registers at the RSU that it first meets, which can prevent illegal vehicles from joining domain D_A.

1) Registration: When a vehicle V_i joins a new domain, a mutual authentication process between the vehicle and the RSU it first meets should start, as described in Fig. 3. Notice that, in our protocol, if an RSU is compromised, TA will revoke it by broadcasting the information of the domain it belongs to and its identity, i.e., every vehicle can get information of revoked RSUs.

First, every RSU in the system periodically broadcasts its certificate, the domain it belongs to, and the group public key. For the RSU R_x in domain D_A, it broadcasts Message 1: $(\mathrm{PK}_{R_x},$

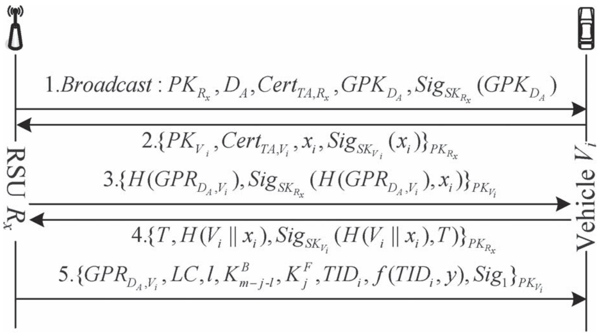

$1.Broadcast : PK_{R_x}, D_A, Cert_{TA,R_x}, GPK_{D_A}, Sig_{SK_{R_x}}(GPK_{D_A})$
$2.\{PK_{V_i}, Cert_{TA,V_i}, x_i, Sig_{SK_{V_i}}(x_i)\}_{PK_{R_x}}$
$3.\{H(GPR_{D_A,V_i}), Sig_{SK_{R_x}}(H(GPR_{D_A,V_i}), x_i)\}_{PK_{V_i}}$
$4.\{T, H(V_i \| x_i), Sig_{SK_{V_i}}(H(V_i \| x_i), T)\}_{PK_{R_x}}$
$5.\{GPR_{D_A,V_i}, LC, l, K^B_{m-j-l}, K^F_j, TID_i, f(TID_i, y), Sig_1\}_{PK_{V_i}}$
RSU R_x
Vehicle V_i

Fig. 3. Mutual authentication protocol.

D_A, $\text{Cert}_{\text{TA},R_x}$, $G\text{PK}_{D_A}$, $\text{Sig}_{\text{SK}_{R_x}}(G\text{PK}_{D_A})$), e.g., every 5 s. When a vehicle V_i gets this message, it first checks whether D_A is a new domain. If it is, this vehicle begins the registration process. By running Verify($\text{PK}_{\text{TA}}, \text{PK}_{R_x} \| D_A, \sigma_{\text{TA},R_x}$), V_i can authenticate the validity of R_x. If $\text{Cert}_{\text{TA},R_x}$ is valid, V_i verifies $\text{Sig}_{\text{SK}_{R_x}}(G\text{PK}_{D_A})$ by PK_{R_x}.

Second, after authenticating R_x, and if D_A is a new domain, V_i sends Message 2: $\{\text{PK}_{V_i}, \text{Cert}_{\text{TA},V_i}, x_i, \text{Sig}_{\text{SK}_{V_i}}(x_i)\}_{\text{PK}_{R_x}}$ to R_x, where x_i is the random number used for computing the group private key $G\text{SK}_{D_A,V_i}$.

Notice that the public key PK_{V_i} with certificate $\text{Cert}_{\text{TA},V_i}$ is unique in the system; as a result, it is also an identifier of V_i. In our scheme, the public key and the certificate of V_i are encrypted by the public key PK_{R_x} of R_x, and only R_x can get the plaintext, which protects the privacy of V_i from revealing its identity.

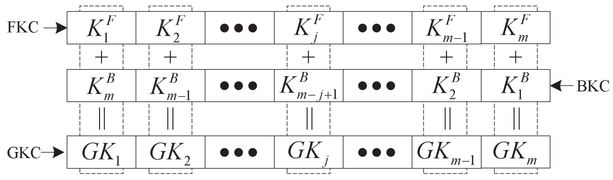
When R_x obtains Message 2, it authenticates V_i by using Algorithm 1.

Algorithm 1: The process of R_x verifying Message 2

```
Require: $\{\text{PK}_{V_i}, \text{Cert}_{\text{TA},V_i}, x_i, \text{Sig}_{\text{SK}_{V_i}}(x_i)\}_{\text{PK}_{R_x}}$
Decrypt $\{\text{PK}_{V_i}, \text{Cert}_{\text{TA},V_i}, x_i, \text{Sig}_{\text{SK}_{V_i}}(x_i)\}_{\text{PK}_{R_x}}$ by its private key $\text{SK}_{R_x}$.
Get $\text{PK}_{V_i}$, $\text{Cert}_{\text{TA},V_i}$, $x_i$, $\text{Sig}_{\text{SK}_{V_i}}(x_i)$.
Check $\text{Cert}_{\text{TA},V_i}$ against the CRL.
if $\text{Cert}_{\text{TA},V_i}$ is not in the CRL then
    Verify $\text{Cert}_{\text{TA},V_i}$ by $\text{PK}_{\text{TA}}$.
    if $\text{Cert}_{\text{TA},V_i}$ is valid then
        Do $Verify(\text{PK}_{V_i}, x_i, \text{Sig}_{\text{SK}_{V_i}}(x_i))$.
        if $\text{Sig}_{\text{SK}_{V_i}}(x_i)$ is valid then
            $R_x$ computes $A_i = g_1^{(1/x_i+\gamma)}$.
            Set $G\text{SK}_{D_A,V_i} = (x_i, A_i)$.
        end if
    end if
end if
```

Third, after getting $G\text{SK}_{D_A,V_i}$, R_x sends to V_i Message 3: $\{H(G\text{SK}_{D_A,V_i}), \text{Sig}_{\text{SK}_{R_x}}(H(G\text{SK}_{D_A,V_i}), x_i)\}_{\text{PK}_{V_i}}$. When V_i receives Message 3, it first decrypts the message by its private key SK_{V_i} and verifies the signature.

Fourth, if the signature is valid, V_i sends Message 4: $\{T, H(V_i \| x_i), \text{Sig}_{\text{SK}_{V_i}}(H(V_i \| x_i), T)\}_{\text{PK}_{R_x}}$ to R_x, in which T is a timestamp. When R_x receives Message 4 at time T^*, it executes Algorithm 2.



FKC: Forward Key Chain BKC: Backward Key Chain GKC: Group Key Chain

Fig. 4. Hash key chain used to generate group keys.

x_i	GSK_{D_A,V_i}	$H(V_i \| x_i)$	PK_{V_i}	T

Fig. 5. Record stores in R_x.

x_i	GSK_{D_A,V_i}	$Sig_{SK_{R_x}}(H(GSK_{D_A,V_i}), x_i)$

Fig. 6. Record stores in V_i.

In this algorithm, $f(\text{TID}_i, y)$ is a bivariate polynomial such as $f(x, y) = s_{0,0} + s_{1,0} \cdot x + s_{0,1} \cdot y + s_{1,1} \cdot xy + \cdots + s_{t,t} \cdot x^t y^t$ for $F_q[x, y]$, where x and y are variables, and $s_{i,j}$ is the constant coefficient. K^B_{m-j-l} and K^F_j are group key seeds used to calculate the group keys, as Fig. 4 shows, l is the length of the backward hash chain, and LC is the life cycle of the group key.

Algorithm 2: The process of R_x verifying Message 4

```
Require: $\{T, H(V_i \| x_i), \text{Sig}_{\text{SK}_{V_i}}(H(V_i \| x_i), T)\}_{\text{PK}_{R_x}}$.
Decrypt $\{T, H(V_i \| x_i), \text{Sig}_{\text{SK}_{V_i}}(H(V_i \| x_i), T)\}_{\text{PK}_{R_x}}$ by its private key $\text{SK}_{R_x}$.
Verify $(T^* - T) \leq \triangle T$, $\triangle T$ is the limit for time-difference.
if $T$ is valid then
    Do Verify($\text{PK}_{V_i}, T \| H(V_i \| x_i), \text{Sig}_{\text{SK}_{V_i}}(H(V_i \| x_i), T)$).
    if $\text{Sig}_{\text{SK}_{V_i}}(H(V_i \| x_i), T)$ is valid then
        $R_x$ gives a temporary identity $\text{TID}_i$ to $V_i$, and computes $f(\text{TID}_i, y)$.
        Compute $\text{Sig}_1 = \text{Sig}_{\text{SK}_{R_x}}(H(G\text{SK}_{D_A,V_i}) \| \text{LC} \| l \| K^B_{m-j-l} \| K^F_j \| \text{TID}_i \| f(\text{TID}_i, y))$.
    end if
end if
```

Fifth, R_x sends to V_i Message 5: $\{G\text{SK}_{D_A,V_i}, \text{LC}, l, K^B_{m-j-l}, K^F_j, \text{TID}_i, f(\text{TID}_i, y), \text{Sig}_1\}_{\text{PK}_{V_i}}$. After receiving Message 5 from R_x, V_i executes Algorithm 3 to get the group key for the HMAC computation. The current group key GK_j is computed as

$$\text{GK}_j = H(K^F_j + K^B_{m-j+1}) \tag{1}$$

where K^F_j is the forward key chain, and K^B_{m-j+1} is the backward key chain in Fig. 4.

Finally, R_x stores the information as in Fig. 5. V_i also stores the information as in Fig. 6.

Group ID	Message ID	Timestamp	Location	Signature	HMAC
2 bytes	2 bytes	4 bytes	4 bytes	181 bytes	4 bytes

Fig. 7. Message type sent by vehicles.

Algorithm 3: The process of V_i verifying Message 5

```
Require: ${GSK_{D_A,V_i}, l, K^B_{m-j+1}, K^F_j, \text{TID}_i, f(\text{TID}_i, y), \text{Sig}_1}_{\text{PK}_{V_i}}$.
1: Decrypt ${GSK_{D_A,V_i}, l, K^B_{m-j+1}, K^F_j, \text{TID}_i, f(\text{TID}_i, y), \text{Sig}_1}_{\text{PK}_{V_i}}$ by $\text{SK}_{V_i}$.
2: Compute $H(G\text{SK}_{D_A,V_i})$.
3: Do Verify($\text{PK}_{R_x}, H(G\text{SK}_{D_A,V_i})\|l\|K^B_{m-j+1}\|K^F_j\|\text{TID}_i\| f(\text{TID}_i, y), \text{Sig}_1$)
4: if $\text{Sig}_1$ is valid then
5: Get $K^B_{m-j+1}$, $K^F_j$.
6: Set the update time as $hboxLC/2$, compute the group key $\text{GK}_j$.
7: end if
```

2) Batch Authentication: According to the dedicated short-range communication (DSRC), each vehicle has to broadcast a security-related message every 300 ms. To ensure the validity of the message source and integrity of these messages, the receiver should verify them. The CRL checking is widely used to exclude invalid vehicles before authentication; however, it needs 9 ms to check one identity in the CRL in a group signature based scheme [1]. Therefore, if a vehicle receives n messages, and the number of revoked vehicles is m, this vehicle needs 9 mn ms to check the source of these messages. Obviously, the CRL checking introduces too much computation delay, greatly degrading the system performance.

To improve the efficiency of authenticating the message source, we employ the HMAC checking to replace the time-consuming CRL checking. In addition, our method of HMAC checking cannot only authenticate the message source but also check the integrity of messages. Combining with the distributed management, we make valid vehicles in the same domain have the same group key seeds ($K^B_{m-j-l}\|K^F_j$) during the registration phase. With the group key seeds, valid vehicles can calculate the group key. Once a vehicle receives the group key, each message sent by this vehicle will attach a HMAC value, as shown in Fig. 7. The signature of this message is generated by $G\text{SK}_{D_A,V_i}$, as shown in Algorithm 4.

Algorithm 4: Message signed by V_i

```
Require: $g_1, g_2, u, v, h, G\text{PK}_{D_A}, G\text{SK}_{D_A,V_i}$.
1: Select random numbers $\alpha, \beta \in \mathbb{Z}_p$.
2: Set $t_{1,i} = \alpha u, t_{2,i} = \beta v, t_{3,i} = A_i + (\alpha + \beta)h$.
3:Set $\delta = \alpha x_i$ and $\mu = \beta x_i$.
4: Select random number $r_\alpha, r_\beta, r_x, r_\delta, r_\mu \in \mathbb{Z}_p$.
5: Set $\begin{cases} S_1 = r_\alpha u \\ S_2 = r_\beta v \\ S_3 = e(t_{3,i}, g_2)^{r_x} e(h, (-r_\alpha - r_\beta)w + (-r_\delta - r_\mu)g_2) \\ S_4 = r_x t_{1,i} - r_\delta u \\ S_5 = r_x t_{2,i} - r_\mu v \end{cases}$
6: Set $c = (S_3 \lambda^{H(M\|T_{stamp}) + t_{1,i} + t_{2,i} + t_{3,i} + S_1 + S_2 + S_3 + S_4 + S_5})$ mod $p$.
7: Set $\begin{cases} s_\alpha = r_\alpha + c\alpha \\ s_\beta = r_\beta + c\beta \\ s_x = r_x + cx_i \\ s_\delta = r_\delta + c\delta \\ s_\mu = r_\mu + c\mu \end{cases}$
8: $\sigma = (t_{1,i}, t_{2,i}, t_{3,i}, c, s_\alpha, s_\beta, s_x, s_\delta, s_\mu)$.
9: return msg $= (M, T_{\text{stamp}}, \sigma)$.
```

When other OBUs receive the message, they can validate the legitimacy of the messages by HMAC checking, thus avoiding the communication overhead and storage overhead caused by the CRL. According to [12], the time for HMAC checking is 0.23 μs for Cipher Block Chaining Advanced Encryption Standard (CBC-HMAC AES) [23] and 0.42 μs for Secure Hash Algorithm SHA-1 [24].[1]

As we state earlier, HMAC works not only to authenticate the message source but also to ensure the integrity of the messages. However, if we only use HMAC, it is impossible for us to know who will be responsible for a message. As a result, before sending a message, the sender has to generate a group signature on this message, which can be verified by batch group verification later.

The batch verification of traffic messages is shown in Algorithm 5.

Algorithm 5: Batch group verification

```
Require: $\text{msg}_1, \text{msg}_2, \ldots, \text{msg}_n, g_1, g_2, u, v, h, G\text{PK}_{D_A}$.
1: Set $\sum_{i=1}^n \widetilde{S}_{1,i} = -\sum_{i=1}^n c_i t_{1,i} + \sum_{i=1}^n s_{\alpha i} u$.
2: Set $\sum_{i=1}^n \widetilde{S}_{2,i} = -\sum_{i=1}^n c_i t_{2,i} + \sum_{i=1}^n s_{\beta i} v$.
3: Set $\sum_{i=1}^n \widetilde{S}_{4,i} = \sum_{i=1}^n s_{xi} t_{1,i} - \sum_{i=1}^n s_{\delta i} u$.
4: Set $\sum_{i=1}^n \widetilde{S}_{5,i} = -\sum_{i=1}^n s_{xi} t_{2,i} - \sum_{i=1}^n s_{\mu i} v$.
5: Set $\prod_{i=1}^n \widetilde{S}_{3,i} \overset{?}{=} e(\sum_{i=1}^n s_{xi} t_{3,i} - \sum_{i=1}^n (s_{\delta i} + s_{\mu i})h - \sum_{i=1}^n c_i g_1, g_2) e(-\sum_{i=1}^n (s_{\alpha i} + s_{\beta i})h + \sum_{i=1}^n c_i t_{3,i}, w)$.
6: if $\prod_{i=1}^n c_i \bmod p \overset{?}{=} ((\prod_{i=1}^n \widetilde{S}_{3,i}) \times \lambda^{\sum_{i=1}^n H(M_i\|T_{stamp_i}) + t_{1,i} + t_{2,i} + t_{3,i} + \widetilde{S}_{1,i} + \widetilde{S}_{2,i} + \widetilde{S}_{4,i} + \widetilde{S}_{5,i}})$ mod $p$
7: then
8: Accept $\text{msg}_1, \text{msg}_2, \ldots, \text{msg}_n$.
```

E. Periodic Update of Group Key

When V_i is verified by one RSU in the domain D_A, it will receive the group key update messages regularly from the RSUs in D_A. The message of the $(j + 1)$th updating round is B_{j+1}, as shown in the following:

$$\begin{cases} B_{j+1} = \{r_{j+1}(x)\} \cup \{p_{j+1}(x)\} \\ r_{j+1}(x) = (x - \text{TID}_{r_1})(x - \text{TID}_{r_2}) \cdots (x - \text{TID}_{r_w}) \\ p_{j+1}(x) = r_{j+1}(x) K^B_{m-j} + f(x, K^F_{j+1}) \end{cases} \quad (2)$$

[1] Both HMAC algorithms are executed on an Intel Core2Duo 2-GHz machine.

where $\mathrm{TID}_{r_1}, \mathrm{TID}_{r_2}, \ldots, \mathrm{TID}_{r_w}$ are temporary identities of the revoked vehicles, which have gotten the group key materials $f(\mathrm{TID}_i, y)$, K^B_{m-j+1}, and K^F_j of D_A before session $(j+1)$ and were revoked during session $(j+1)$; $r_{j+1}(x)$ is the revocation polynomial in session $(j+1)$; and $p_{j+1}(x)$ is the masking polynomial in session $(j+1)$.

Notice that only the vehicles that have passed the legitimacy authentication of D_A could obtain the group key materials, and RSUs only manage vehicles in their domain. Therefore, the number of revoked vehicles is limited, and only a temporary identity of each revoked vehicle is needed to compute $f(\mathrm{TID}_i, y)$; therefore, the size of $p_{j+1}(x)$ is very small.

When V_i receives the broadcast message B_{j+1}, it uses K^F_j to calculate $K^F_{j+1} = H(K^F_j)$ and $f(\mathrm{TID}_i, K^F_{j+1})$. Then, V_i calculates $p_{j+1}(\mathrm{TID}_i)$, and obtains K^B_{m-j} using the following:

$$K^B_{m-j} = \frac{p_{j+1}(\mathrm{TID}_i) - f(\mathrm{TID}_i, K^F_{j+1})}{r_{j+1}(\mathrm{TID}_i)}. \tag{3}$$

After getting K^B_{m-j}, V_i computes whether $H^l(K^B_{m-j-l}) = K^B_{m-j}$ holds or not. If it holds, V_i computes the new group key according to (1).

IV. Cooperative Authentication

In our basic scheme, even if we ensure that only legal vehicles join the domain and there are no invalid signatures in the batch, the scheme can only verify 274 messages at most per second, which still cannot meet the requirement of authentication speed. Therefore, measures must be taken to solve this problem. According to Zhang *et al.* [8] and Hao *et al.* [14], the authentication efficiency can be improved by utilizing cooperation. By making the neighboring vehicles work cooperatively, their schemes can ensure that a vehicle knows the authenticity of all received messages without verifying all the message signatures. The basic requirements of cooperative authentication scheme are listed in the following.

- One verifier should physically be in front of V, whereas the others should be behind V, which means that the verifiers that are used to cooperate with are better to be a pair and can broadcast verification result messages to others.
- Verifiers should be away from each other as far as possible.
- The number of verifiers should be neither too small nor too large.

We assume that each security-related message carries the location information of the sender vehicle.[2] When the vehicle V_i receives messages from different senders at the same time, it first extracts location information of senders and then runs the cooperation choice process according to the given requirements to decide which messages should be used for batch verification.

V_i checks the received messages every 300 ms and computes the distance between the message senders and itself according to location information. Then, it can build a table similar to Table II, where the message ID is a random order index, direction means that the sender is in front of or behind the receiver, and distance is the distance between the senders and the receiver.

TABLE II
Messages V_i Received

Message ID	Direction	Distance (m)
1	front	50
2	front	115
3	behind	45
4	behind	85
5	front	130
. . .	. . .	. . .

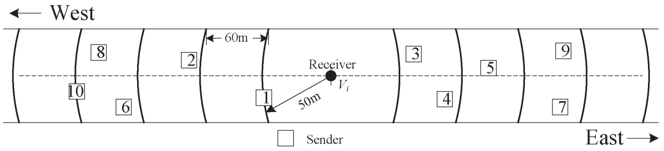


Fig. 8. Cooperation sample.

Assuming vehicles are uniformly distributed, we divide the communication range by 60 m as shown in Fig. 8, according to the basic requirements of cooperation scheme and the number of authentication messages. We define the vehicles (50 ± 5), (110 ± 5), (170 ± 5), (230 ± 5), (290 ± 5) m away from the sender are cooperative vehicles. In Fig. 8, V_i receives ten messages sent by senders 1–10 at the same time, then computes the distance between each sender and itself, and obtains Table II. V_i should let messages 1, 2, and 3 in the batch to be verified. Since the cooperation scheme can reduce the number of verification messages, the authentication speed is increased. The performance analysis shows that our scheme can satisfy the requirement of authenticating 600 messages per second in VANETs.

V. Security Analysis

A. Against RSUs' Compromission

Considering the problem of RSUs' compromission, in the communication process of mutual authentication and group key generation, V_i can get services without revealing its real identity to RSUs. Therefore, if there exist some RSUs compromised, our protocol can still preserve the privacy of vehicles' identities.

1) Against False Charge: If a vehicle is investigated, TA will start an accusation procedure and ask the corresponding RSUs for some information about the investigated vehicle. However, the RSU may be compromised and give TA another vehicle's information to protect the investigated vehicle, which we call false charge. Here, we show our scheme can prevent this type of attacks.

In our protocol, each message sent by V_i is signed by its private key SK_{V_i}, and the group private key and private key of V_i are bound together. Since R_x does not have SK_{V_i}, it cannot forge V_i's signature. Moreover, the group private key and the private key of V_i are bound together, which are hard for R_x to forge. We also store the information about the mutual authentication, as shown in Figs. 5 and 6. When a dispute happens, the TA can ask the vehicle and the RSU to show the information.

[2] This can be generated by a Global Positioning System device [14].

2) Nonrepudiation of Giving the Group Private Key to a Vehicle: Once R_x gives the group private key to V_i, it cannot repudiate. In Message 3, R_x sends the hash value of $G\mathrm{SK}_{D_A,V_i}$ and the signature of the group private key. When V_i receives Message 5, and obtains $G\mathrm{SK}_{D_A,V_i}$, it can verify the validity of GPR_{D_A,V_i} by the hash function, thus protecting the failure caused by the wireless channel interface. To ensure that the group private key is generated by x_i, V_i stores the signature $\mathrm{Sig}_{\mathrm{SK}_{R_x}}(H(G\mathrm{SK}_{D_A,V_i}), x_i)$ sent by R_x, whereas R_x also stores x_i and $H(V_i \| x_i)$. When a dispute happens, R_x can show the information to TA. Since the public parameters of the group signature are generated by TA, it can compute the group private key of V_i. From PK_{V_i}, TA can get V_i, and then it can verify $H(V_i \| x_i)$. If $H(V_i \| x_i)$ passes the validity verification, the group private key $G\mathrm{SK}_{D_A,V_i}$ is valid; otherwise, $G\mathrm{SK}_{D_A,V_i}$ is not valid. For V_i, V_i first shows x_i to TA, then TA can compute the group private key $G\mathrm{SK}_{D_A,V_i}$ for V_i. If $G\mathrm{SK}_{D_A,V_i}$ is correct, TA verifies the signature to ensure $G\mathrm{SK}_{D_A,V_i}$ is generated by R_x.

3) Preventing Colluding With Vehicles: A compromised RSU may collude with a malicious vehicle and then send other legal vehicles' group private key to its accomplice. Then, the malicious vehicle can broadcast messages on behalf of other vehicles.

To prevent this kind of attack, in the protocol, the message signatures contain the information of identity; R_x and V_i also store these information of authentication after finishing the mutual authentication. If there is a dispute, V_i can show TA the stored information. By computing the group private key $G\mathrm{SK}_{D_A,V_i}$ and verifying the signature $\mathrm{Sig}_{\mathrm{SK}_{R_x}}(H(G\mathrm{SK}_{D_A,V_i}), x_i)$, TA can affirm to which vehicle $G\mathrm{SK}_{D_A,V_i}$ belongs.

B. Conditional Privacy

As in the framework of VANETs that we give, the RSUs are responsible for issuing the private key of the group signature. When V_i wants to join domain D_A, V_i sends its public key PK_{V_i} and certificate $\mathrm{Cert}_{\mathrm{TA},V_i}$ to R_x since the public key PK_{V_i} is unique in the VANETs, which will reveal the identity-related information of V_i. To deal with this issue, in our scheme V_i encrypts this information with the public key PK_{R_x} of R_x; therefore, only R_x can get the identity-related information. After getting the private key $G\mathrm{SK}_{D_A,V_i}$ of the group signature, the messages sent by V_i are signed by $G\mathrm{SK}_{D_A,V_i}$. Due to the anonymous character of the group signature, other vehicles can authenticate V_i without learning anything about its identity. Therefore, the identity of a message sender can be protected from other vehicles, whereas RSUs can distinguish whether two messages are from the same vehicle, and TA and RSUs can cooperate to trace the real identity of a message sender.

C. Message Integrity and Source Authentication

In general, the integrity of messages can be ensured by the signature or HMAC. In the registration stage of V_i joining the domain D_A, the messages delivered between R_x and V_i contain the signature, which can ensure the message integrity and source authentication. After obtaining the private group-signature key $G\mathrm{SK}_{D_A,V_i}$, V_i can send messages signed with $G\mathrm{SK}_{D_A,V_i}$ to realize message integrity. Since the communication between vehicles and RSUs is via a wireless channel, packet loss or bogus injection is more likely to happen. The signature verification may introduce extra computation overhead; therefore, we employ the lightweight HMAC to check the integrity of messages. By verifying HMAC before batch verification, the integrity of messages can be ensured, which the existing batch group signature schemes [6], [7] do not consider. Due to the nature of message integrity and source authentication, typical attacks, such as bogus attack and impersonation attack, can be prevented.

VI. Performance Evaluation

As we described in Section III, when a vehicle comes to a new domain, it should mutually authenticate with the RSU that it first meets in this domain and then get the key materials. After finishing this phase, the vehicle will broadcast the security-related messages every 300 ms according to the DSRC. For a better performance evaluation comparison, we divide the performance analysis into two phases: 1) key distribution and 2) periodic security-related message broadcasting.

A. Key Distribution

In our scheme, we adopt a distributed management scheme, in which the whole precinct is divided into a few different domains. When a vehicle joins a new domain, it authenticates with the first RSU that it meets, obtains the group signature key, and then stores the key materials for computing HMAC. To reflect the efficiency of this phase, we analyze the communication and computation overhead analysis here. We compare our work with the Hao *et al.* scheme (named CMAP) [14] since this work has also considered a distributed manner for a group signature based scheme, and RSUs are also assumed to be semi-trustworthy.

In CMAP, messages are signed with an elliptical-curve digital signature algorithm (ECDSA) and encrypted with an elliptical-curve integrated encryption scheme (ECIES). For a better comparison, we also employ ECDSA for signature and ECIES for encryption on the sides of RSUs and OBUs, whereas we employ the Schnorr signature algorithm [21] for TA to sign certifications.

The communication overhead: Comparing with CMAP, during the key distribution phase, the mutual authentication process is composed of five messages. The total size of these messages in our scheme is 501 B and that in CMAP is 497 B, in which the length of several parameters is given in Table III. We assume that the plaintext and ciphertext of a message have the same length.

We divide the HMAC generation materials in this phase, which leads to addition communication overhead. Fortunately, the additional delay for this phase is limited. To reflect the efficiency, we focus on the average end-to-end transmission delay with different number of vehicles in the

TABLE III
LENGTH OF PARAMETERS

Notation	Length (Byte)
PK_{R_x}	21
x_i	8
GSK_{D_A,V_i}	22
D_A	4

TABLE IV
SIMULATION PARAMETERS

PARAMETER	VALUE
Simulation area	$1000m \times 1000m$
Simulation time	$30s$
Speed of vehicle	$10 - 30m/s$
Wireless protocol	802.11p
Agent	PCB
Mobility generation tool	VanetMobiSim
Network simulation tool	NS2
Channel bandwidth	6 Mbs
Radio propagation model	TwoRayGround

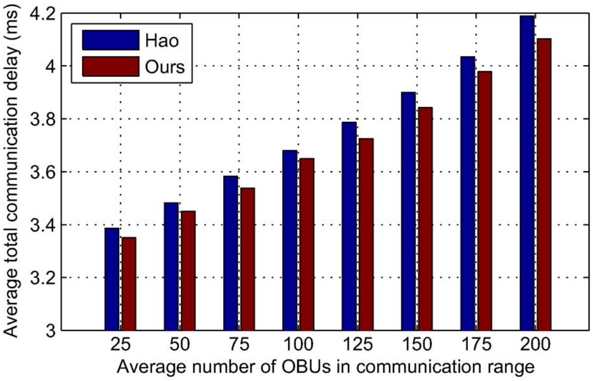


Fig. 9. Average communication delay.

communication range. We give the simulation using NS2.34 under 802.11p. The simulation parameters are listed in Table IV. Notice that we do not consider the computation delay in the simulation.

In Fig. 9, we can see that the total transmission delay slowly increases with the number of vehicles in the communication range. The average transmission delay is 3.75 ms in our scheme and 3.7 ms in the Hao *et al.* scheme. It is because that we also send the group key materials to vehicles, which makes the message size in our scheme larger than that in the Hao *et al.* scheme. However, the size of additional messages is very small; therefore, the communication delays of both schemes are very close.

We do not give the detail analysis of computation overhead since both schemes have almost the same computation overhead.

B. Periodic Security-Related Message Broadcast

In VANETs, each OBU has to broadcast a security-related message every 300 ms. To ensure the validity of the message source and the integrity of messages, the receiver has to verify the certification of the message source and the signature of this message. Before verifying those, the receiver may need to check whether the sender has been revoked, e.g., through CRL checking or other technique such as HMAC, which we propose. Obviously, the total time needed for verifying one received message is composed of three parts: certification verification time, signature verification time, and revocation check time. If we do not consider any countermeasure, the total time will linearly increase with the number of verified messages, in which the revocation check time for one message also linearly increases with the number of revoked vehicles. As a result, these may greatly degrade the system performance and may even paralyze the whole system.

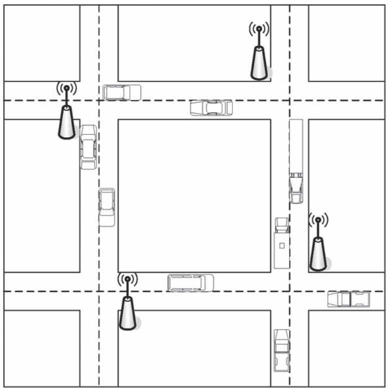


Fig. 10. Road scenario for simulation.

There are a few schemes proposed to decrease the verification time of messages [6], [7], [12], [14]. In the Zhang *et al.* scheme [7] and Wasef and Shen's scheme [6], batch group signature verification is employed to reduce the signature verification time, in which messages in a batch can be verified at the same time. The Hao *et al.* scheme [14] uses cooperative authentication to reduce the number of messages that each vehicle needs to verify. However, the given three schemes do not consider the revocation check time, which consumes much time if the CRL checking is used as we described earlier. Wasef and Shen's scheme [12] considers the problem caused by CRL checking; however, this scheme achieves conditional privacy based on pseudonyms, which cannot suit for a group signature based scenario.

The lengths of the group signatures in [7] and [14] and our proposed scheme are 192, 368, and 181 B, respectively. The broadcast security-related message format in our scheme is given in Fig. 7. The computation overhead for a vehicle to verify n broadcast messages is $2nT_{\text{par}} + 10nT_{\text{mul}} + 3nT_{\text{exp}}$ in [14], $2T_{\text{par}} + 13nT_{\text{mul}}$ in [7], and $2T_{\text{pai}} + (6n + 7)T_{\text{mul}}$ in our scheme, respectively, where T_{par} means the time for executing a pairing operation and T_{mul} means the computing time of a point multiplication [6]. According to [1], T_{pai} is 4.5 ms, and T_{mul} is 0.6 ms on an Intel Pentium IV 3-GHz machine, respectively. The road scenario for simulation is given in Fig. 10.

In the following, we analyze the additional communication and computation overhead brought by the broadcast message.

1) Without Considering Revocation Check: We use the average end-to-end delay (AEED) to reflect the efficiency. The AEED is defined as the average time difference between the verified time in the receiver side and the broadcast time in

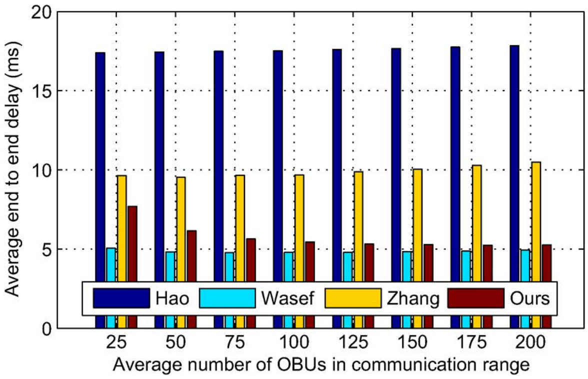


Fig. 11. AEED without considering CRL checking.

the sender side of a message. For the vehicle V_j, the AEED can be computed as follows:

$$\text{AEED} = \frac{\sum_{i=1,i\neq j}^{N}\sum_{l=1}^{M_i}\left(T_{V_i\to V_j,m_l}^{\text{recv}} - T_{V_i\to V_j,m_l}^{\text{send}}\right)}{\sum_{i=1}^{N} M_i} + T_{\text{ave}}^{\text{veri}} \tag{4}$$

where N represents the total number of vehicles in the simulation, and M_i is the number of messages sent by the V_i. $T_{V_i\to V_j,m_l}^{\text{recv}}$ means the moment of V_j receiving the message m_l from V_i, and $T_{V_i\to V_j,m_l}^{\text{send}}$ is the sending time of m_l. $T_{\text{ave}}^{\text{veri}}$ is the average verification time for each message, which may vary in different schemes. The first part of (4) represents the average communication delay, and its second part gives the average computation delay.

In Fig. 11, we give the AEED with different number of vehicles in the communication range, where the speed of vehicles is from 10 to 30 m/s. We can see that the average AEED of Hao *et al.*'s scheme is about 17.6 ms, which increases very slowly with the number of OBUs in the communication range. It is the largest among that of four schemes.[3] The average AEED is 4.86 ms in Wasef and Shen's scheme, 9.9 ms in the Zhang *et al.* scheme, and 5.76 ms in our scheme, which is 27.6%, 56.3%, 32.7% of the Hao *et al.* average AEED, respectively. Since the last three schemes employ the batch group signature algorithm, the average verification delay for each message decreases. Moreover, in the last three schemes, AEEDs keep a small fluctuation. It is because the average number of OBUs in the communication range increases, and the transmission delay is increasing, whereas the average verification delay is decreasing. In Fig. 11, it is shown that the AEED in our scheme is larger than that in Wasef and Shen's scheme since cooperative authentication makes the number of messages in the batch verification to be reduced. With the increase in OBUs' number, the performance of our scheme is better.

However, in the aforementioned AEED comparison, we do not consider the time constraint. To further show our scheme's performance, we use the average message loss ratio as another measurement in our evaluation. The average message loss ratio is defined as the ratio of the number of messages dropped to the total number of messages received in every 300 ms. Through simulation, we obtain a comparison of the three schemes (see Fig. 12). As Fig. 12 shows, in Wasef and Shen's scheme, when the number of vehicles in the communication range is larger than 62 (and 31 in the Zhang *et al.* scheme), the average message loss ratio increases with the number of vehicles, whereas that in our scheme remains zero. We give a brief explanation as follows.

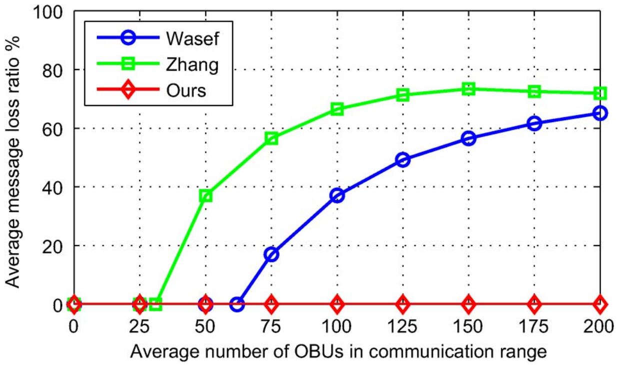


Fig. 12. Message loss ratio with the number of vehicles.

Although the AEED in Wasef and Shen's scheme is lower than that in our scheme, each vehicle has to authenticate all the messages it received, and the total verification delay increases with the number of vehicles. However, in our scheme, because we adopt cooperative authentication, the number of messages that need to be verified is less than that in Wasef and Shen's scheme. In addition, an interesting appearance in Fig. 12 is that the average message loss ratio of the Zhang *et al.* scheme declines when the number of vehicles in the communication range increases to 150. It is because the packet loss ratio[4] increases with the number of vehicles in the communication range. It also happens to Wasef and Shen's scheme. Here, we do not give the performance of the Hao *et al.* scheme in Fig. 12, since without considering the revocation check, the Hao *et al.* scheme can also verify more than 200 messages per 300 ms and keep the same performance as our scheme.

In Fig. 12, it is clearly shown that our scheme is the most efficient among the group signature based authentication schemes. Note that we do not take account of the revocation check and invalid messages in this analysis. If these two factors are considered, the performance of the Zhang *et al.* scheme and Wasef and Shen's scheme will further degrade. However, even considering the revocation check, our scheme still have a good performance since we use HMAC to avoid the time-consuming CRL checking and to ensure the integrity of messages.

2) Considering the Revocation Check: Our scheme uses HMAC instead of the CRL checking to check and exclude the revoked vehicles. If we consider this process, the corresponding AEED in the Zhang *et al.* scheme is given as follows:

$$\text{AEED}_{\text{Zhang}} = \frac{1}{\sum_{i=1}^{N} M_i} \times \sum_{i=1,i\neq j}^{N}\left\{\sum_{l=1}^{M_i}\left(T_{V_i\to V_j,m_l}^{\text{recv}} - T_{V_i\to V_j,m_l}^{\text{send}} + T_{m_l}^{\text{CRL}}\right) + T_{V_i}^{\text{batch}}\right\} \tag{5}$$

[3] The best number of verifiers is 8 in [14].

[4] The packet loss ratio is defined as the ratio between the number of messages V_i receives and the number of messages sent to V_i.

where $T^{\rm CRL}_{m_l}$ is the CRL checking time for message m_l, and $T^{\rm batch}_{V_i}$ is the batch verification time for all the messages from V_i.

In Wasef and Shen's scheme, after executing the CRL checking process, the receiver batch verifies the received messages. The corresponding AEED is given as follows:

$$\mathrm{AEED}_{\rm Wasef} = \frac{1}{\sum_{i=1}^{N} M_i} \times \sum_{i=1,i\neq j}^{N} \left\{ \sum_{l=1}^{M_i} \left(T^{\rm recv}_{V_i\to V_j,m_l} - T^{\rm send}_{V_i\to V_j,m_l} + T^{\rm CRL}_{m_l} \right) + T\prime^{\rm batch}_{V_i} \right\} \tag{6}$$

where $T\prime^{\rm batch}_{V_i}$ is the batch verification time for all messages from V_i.

In the Hao *et al.* scheme, after receiving messages, it first executes the cooperative choice algorithm, and then carries out the CRL checking process and verifies the signatures of the chosen messages. Therefore, the AEED for the Hao *et al.* scheme is given as

$$\mathrm{AEED}_{\rm Hao} = \frac{\sum_{i=1,i\neq j}^{N} \sum_{l=1}^{M_i} \left(T^{\rm recv}_{V_i\to V_j,m_l} - T^{\rm send}_{V_i\to V_j,m_l} \right)}{\sum_{i=1}^{N} M_i} + \frac{\sum_{i=1,i\neq j}^{N} \sum_{k=1}^{K_i} \left(T^{\rm CRL}_{k} + T^{\rm veri}_{k} \right)}{\sum_{i=1}^{N} K_i} \tag{7}$$

where K_i means the number of messages chosen with the cooperative authentication by vehicle V_j, $T^{\rm CRL}_k$ is the CRL checking time for message k and $T^{\rm veri}_k$ is the verification time for message k.

In our scheme, the receiver should choose a few messages from the received messages to execute HMAC checking and then verify the signatures in a batch. The AEED is given as formula

$$\mathrm{AEED}_{Our} = \frac{\sum_{i=1,i\neq j}^{N} \sum_{l=1}^{M_i} \left(T^{\rm recv}_{V_i\to V_j,m_l} - T^{\rm send}_{V_i\to V_j,m_l} \right)}{\sum_{i=1}^{N} M_i} + \frac{\sum_{i=1,i\neq j}^{N} \left(\sum_{k=1}^{K_i} T^{\rm HMAC}_{k} + T\prime^{\rm batch}_{K_i} \right)}{\sum_{i=1}^{N} K_i} \tag{8}$$

where $T^{\rm HMAC}_k$ is the HMAC checking time for message k, and $T\prime^{\rm batch}_{K_i}$ is the batch verification time for K_i messages chosen by V_j by using the cooperative algorithm.

Since the CRL checking needs 9 ms [1] for one identity in the CRL, from (5)–(7), with different number of revoked identities in their scheme, the AEED is different. We give the AEED with two identities in Fig. 13 and that with four identities in Fig. 14 in the CRL, respectively. In Fig. 13, the average AEED is 22.8 ms in Wasef and Shen' scheme, 27.9 ms in the Zhang *et al.* scheme, 35.6 ms in the Hao *et al.* scheme, and 5.76 ms in our scheme. In Fig. 14, the AEED is 40.8 ms in Wasef and Shen' scheme, 45.9 ms in the Zhang *et al.* scheme, 53.6 ms in the Hao *et al.* scheme, and 5.76 ms in our scheme. Compared with that in Fig. 11, considering the CRL checking process, the schemes (the Hao *et al.* scheme, Wasef and Shen's scheme, and Zhang's scheme) have an obviously increasing AEED with the number of identities in the CRL. In their schemes, when a vehicle receives messages at the same time, the cooperation chosen process runs first (although this process only happens to the Hao *et al.* scheme), and then the CRL checking should be run before the group signature verification. One message checking relates to two pair calculations for one identity in the CRL, which needs 9 ms [1]. Assuming there are m identities in the CRL and n messages are selected, the entire time for CRL checking is 9 mn ms. However, in our scheme, since we avoid the CRL checking, the performance improvement is remarkable.

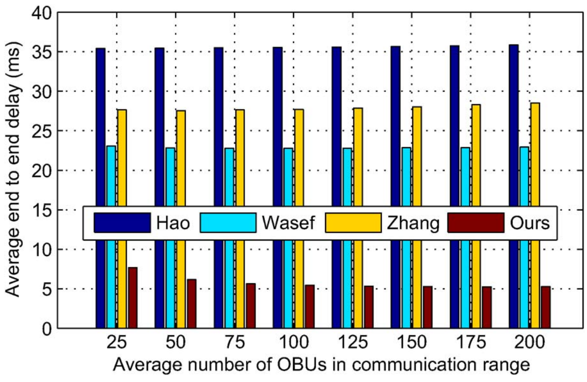


Fig. 13. AEED with two identities in the CRL.

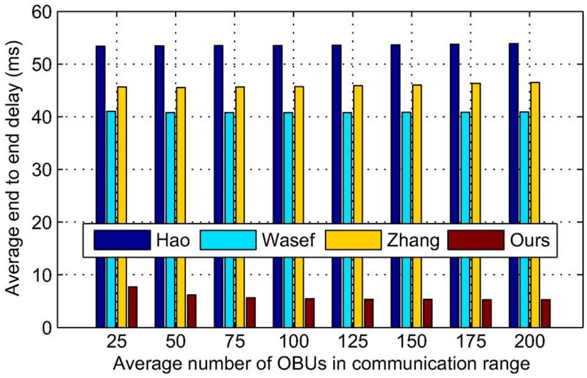
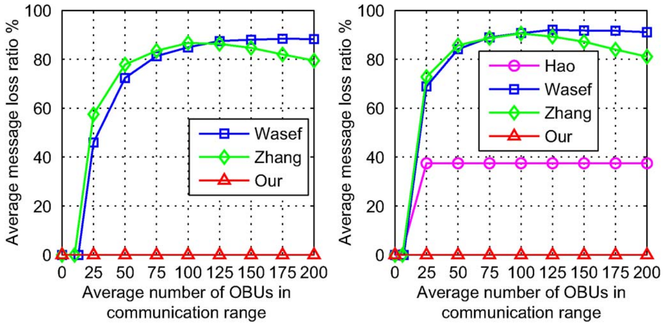


Fig. 14. AEED with four identities in the CRL.



(a)

(b)

Fig. 15. Average messages loss ratio with $n_{\rm id}$ identities in the CRL. (s) $n_{\rm id} = 2$. (b) $n_{\rm id} = 4$.

We also give the corresponding message loss ratio in Fig. 15(a) and (b) according to Figs. 13 and 14, respectively.

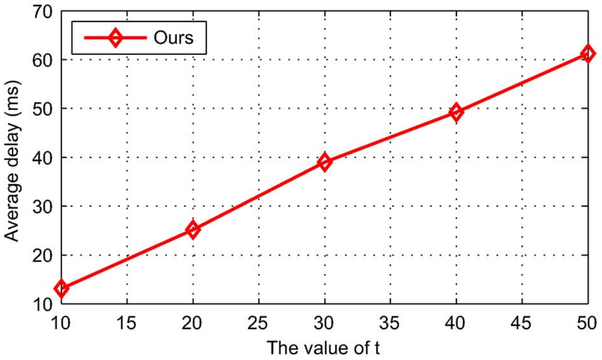


Fig. 16. Delay for generating the key for HMAC.

The average message loss ratio is greater than zero when the average number of OBUs is more than 13 in the communication range in Wasef and Shen's scheme (10 in Zhang's scheme) in Fig. 15(a), and 5 in the Hao *et al.* scheme, 6 in Zhang's scheme, and 7 in Wasef and Shen's scheme in Fig. 15(b). Although the AEED of the Hao *et al.* scheme is the largest, by employing cooperative authentication, it only needs to verify eight messages among all the messages it receives. As in Fig. 15(a), the average message loss ratio of Hao *et al.* is also zero; even in Fig. 15(b), the average message loss ratio is smaller than that in Wasef and Shen's scheme and Zhang's scheme. It is obvious that the average message loss ratio of their schemes (the Hao *et al.* scheme, Wasef and Shen's scheme, and Zhang's scheme) increases with the number of identities in the CRL. In our scheme, since adopting the HMAC checking process to replace the CRL checking process, the message loss ratio still remains zero.

C. HMAC Periodic Update

After key distribution, the group key for computing HMAC will periodically update or update when there is a revoked vehicle. In our scheme, since we use the distributed manner, the periodic update or revoking update is executed according to different domains. The communication overhead of our scheme is given as (2), whereas the message length is $(t+1)\log q \approx (t+1)$ B and the corresponding computation overhead is $(2t+1)$ multiplication operations [15].

We are also interested in the average delay for vehicles' getting the new group key. The simulation is given in Fig. 16, where we assume that there are 200 vehicles in the communication range of the RSU. Fig. 16 shows the influencing factor to the delay, which means that our scheme can realize seamless application.

VII. Related Work

According to the DSRC, a vehicle should broadcast security-related messages every 300 ms. In other words, a vehicle has to verify 600 security-related messages per second if there are about 180 vehicles in the communication range. To achieve this objective, the verification process of the group signature attached to the security-related messages has to be efficient enough. To reduce the signature verification time, Wasef and Shen [6] and Zhang *et al.* [7] employ batch group signature verification based on the properties of bilinear pairing operation, in which a large number of messages can be authenticated in a timely manner. The question is that they do not check the integrity of messages before running batch verification. If there exists a few invalid messages caused by wireless interference, packet loss, or bogus injection, they may introduce additional verification delay for rebatch and then lose their efficiency. Even if we do not count the rebatch time, the computation overhead of batch group signature verification in [7] is $2T_{\rm par} + 13nT_{\rm mul}$[5] and that in [6] is $3T_{\rm pai} + (6n+7)T_{\rm mul}$, which still cannot satisfy the requirement of verifying 600 messages per second.

In VANETs, the CRL is employed to efficiently manage the revoked vehicles. However, the CRL checking process is time-consuming [12], [13], [25]. To address the CRL checking problem, HMAC is adopted to replace the CRL, in [12], [13], and [25], greatly reducing the checking time. In Wasef and Shen's [12] scheme, the key for HMAC computation is in a global manner. Once an illegal vehicle is found, the global key update process starts, which is another form of CRL and is difficult to implement. The Jiang *et al.* [13] scheme runs in a distributed manner, further improving the efficiency of the HMAC checking. However, both of their schemes are based on pseudonyms, which may not fit group signature based schemes directly. Based on the group signature, our previous work [25] uses HMAC to avoid the CRL checking in a distributed manner, where we assume that RSUs are entirely trusted. However, RSUs may want to look for users' privacy information, although they may perform according to the protocol. Therefore, the semi-trust model of RSU is introduced in [16] and [17] to get a more practical setting in VANETs.

By using the aforementioned schemes, the time in CRL checking and group signature verification processes can be considerably reduced. However, it is still infeasible for a single vehicle to accomplish the requirement of verifying 600 messages per second. By observing the fact that each vehicle in the same area verifies almost the same set of messages, Zhang *et al.* [8] and Hao *et al.* [14] propose their schemes based on cooperation among vehicles. By allowing the neighboring vehicles to cooperatively authenticate messages, their schemes can ensure that a vehicle knows the authenticity of all received messages without verifying all the signatures it receives. Although the Hao *et al.* scheme can achieve the verification speed of 600 messages per second, it does not take account of the CRL checking before signature verification. Therefore, there exists performance degradation in a practical setting. In this paper, we jointly use the techniques of distributed management, HMAC, batch group signature verification, and practical cooperative authentication to achieve efficient conditional privacy-preserving authentication under the semi-trust model of the RSU.

Notice that many pseudonym based anonymous authentication schemes for VANETs have been proposed [1], [2], [26], [27]. Here, we give a brief comparison between group signature

[5] Note that the batch group signature scheme used in the Zhang *et al.* scheme is defined by Ferrara *et al.* [22].

based schemes and pseudonym based ones. It is clear that the purpose of employing these two underlying techniques is to realize anonymity of attendees. The remarkable advantage of pseudonym based schemes is that they can authenticate many messages in a short time, which is suitable for the periodic broadcasting of security-related information according to DSRC. However, to realize privacy, TA has to generate a large number of pseudonyms by hash chain or pseudonym pool, which may result in pseudonym collision. Moreover, TA needs to assign these pseudonyms to vehicles, as well as manage and store related messages for the accusation procedure. Furthermore, the pseudonym collision means that different vehicles may have the same pseudonyms; therefore, the identity-based signature schemes cannot work. As a result, it is hard to distinguish who signs the messages. While in the group signature based schemes, the heavy load of pseudonym management can be eliminated. However, it is at a price of high message loss ratio since verifying a group signature consumes more time than authenticating a pseudonym. Fortunately, we can minimize the disadvantage of a group signature based authentication scheme by using the batch verification and cooperation, while keeping the advantage of easy management.

VIII. Conclusion

We have proposed an efficient privacy-preserving group signature based authentication scheme for VANETs in this paper. We have jointly used the techniques of distributed management, HMAC, batch group signature verification, and cooperative authentication to achieve the design goal. First, we divide the whole network into several domains, which allows localized management. HMAC is used in our scheme to replace the time-consuming CRL checking and to ensure the integrity of messages before batch verification, reducing the number of invalid messages in the batch. We also use cooperative authentication to further improve the efficiency of our scheme. By employing the given methods, our scheme can meet the requirement of verifying 600 messages per second. The security and performance analysis show that our scheme can achieve efficient group signature based authentication while keeping conditional privacy for VANETs.

References

[1] C. Zhang, R. Lu, X. Lin, P.-H. Ho, and X. Shen, "An efficient identity-based batch verification scheme for vehicular sensor networks," in *Proc. IEEE INFOCOM*, Phoenix, AZ, USA, Apr. 2008, pp. 246–250.

[2] Y. Sun, R. Lu, X. Lin, X. Shen, and J. Su, "An efficient pseudonymous authentication scheme with strong privacy preservation for vehicular communications," *IEEE Trans. Veh. Technol.*, vol. 59, no. 7, pp. 3589–3603, Sep. 2010.

[3] D. Chaum and E. van Heyst, "Group signatures," *Adv. Cryptol.—Eurocrypt*, vol. 547, pp. 257–265, 1991.

[4] J. Guo, J. P. Baugh, and S. Wang, "A group signature based secure and privacy-preserving vehicular communication framework," in *Proc. Mobile Netw. Veh. Environ.*, Anchorage, AK, USA, May 2007, pp. 103–108.

[5] X. Lin, X. Sun, P.-H. Ho, and X. Shen, "GSIS: A secure and privacy-preserving protocol for vehicular communications," *IEEE Trans. Veh. Technol.*, vol. 56, no. 6, pp. 3442–3456, Nov. 2007.

[6] A. Wasef and X. Shen, "Efficient group signature scheme supporting batch verification for securing vehicular networks," in *Proc. IEEE ICC*, Cape Town, South Africa, May 2010, pp. 1–5.

[7] L. Zhang, Q. Wu, A. Solanas, and J. Domingo-Ferrer, "A scalable robust authentication protocol for secure vehicular communications," *IEEE Trans. Veh. Technol.*, vol. 59, no. 4, pp. 1606–1617, May 2010.

[8] C. Zhang, X. Lin, R. Lu, P.-H. Ho, and X. Shen, "An efficient message authentication scheme for vehicular communications," *IEEE Trans. Veh. Technol.*, vol. 57, no. 6, pp. 3357–3368, Nov. 2008.

[9] S.-B. Lee, G. Pan, J.-S. Park, M. Gerla, and S. Lu, "Secure incentives for commercial ad dissemination in vehicular networks," in *Proc. 8th ACM Int. Symp. MobiHoc*, Montreal, QC, Canada, Sep. 2007, pp. 150–159.

[10] M. Raya and J.-P. Hubaux, "Securing vehicular ad hoc networks," *J. Comput. Security*, vol. 15, no. 1, pp. 39–68, Jan. 2007.

[11] K. Mershad and H. Artail, "A framework for secure and efficient data acquisition in vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 62, no. 2, pp. 536–551, Feb. 2013.

[12] A. Wasef and X. Shen, "EMAP: Expedite message authentication protocol for vehicular ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 1, pp. 78–89, Jan. 2013.

[13] S. Jiang, X. Zhu, and L. Wang, "A conditional privacy scheme based on anonymized batch authentication in vehicular ad hoc networks," in *Proc. IEEE WCNC*, Shanghai, China, Apr. 2013, pp. 2375–2380.

[14] Y. Hao, Y. Chen, C. Zhou, and S. Wei, "A distributed key management framework with cooperative message authentication in VANETs," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 3, pp. 616–629, Mar. 2011.

[15] R. Dutta, S. Mukhopadhyay, and M. Collier, "Computationally secure self-healing key distribution with revocation in wireless ad hoc networks," *Ad Hoc Netw.*, vol. 8, no. 6, pp. 597–613, Aug. 2010.

[16] K. Sampigethaya, M. Li, L. Huang, and R. Poovendran, "AMOEBA: Robust location privacy scheme for VANET," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 8, pp. 1569–1589, Oct. 2007.

[17] Y. Hao, Y. Cheng, and K. Ren, "Distributed key management with protection against RSU compromise in group signature based VANETs," in *Proc. IEEE GLOBECOM*, New Orleans, LA, USA, Dec. 2008, pp. 1–5.

[18] W. Mao, *Modern Cryptography: Theory and Practice*. Englewood Cliffs, NJ, USA: Prentice-Hall, 2003.

[19] H. Krawczyk, R. Canetti, and M. Bellare, "HMAC: Keyed-hashing for message authentication," RFC 2104, Feb. 1997.

[20] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Proc. Adv. Cryptol._CRYPTO*, vol. 2139, *Lecture Notes in Computer Science*, 2001, no. 2001, pp. 213–229.

[21] C. P. Schnorr, "Efficient signature generation by smart cards," *J. Cryptol.*, vol. 4, no. 3, pp. 161–174, 1991.

[22] A. L. Ferrara, M. Green, S. Hohenberger, and M. Pedersen, "Practical short signature batch verification," in *Proc. Top. Cryptol.—CT-RSA*, vol. 5473, *Lecture Notes in Computer Science*, 2009, no. 2009, pp. 309–324.

[23] S. Frankel, R. Glenn, and S. Kelly, The AES-CBC cipher algorithm and its use with IPsec, RFC 3602, Sep. 2003.

[24] D. Eastlake and P. Jones, US secure hash algorithm 1 (SHA1), RFC 3174, Sep. 2001.

[25] X. Zhu, S. Jiang, L. Wang, H. Li, W. Zhang, and Z. Li, "Privacy-preserving authentication based on group signature for VANETs," presented at the IEEE Global Telecommunications Conf., Atlanta, GA, USA, Dec. 2013, Paper WN-23.

[26] K. A. Shim, "An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks," *IEEE Trans. Veh. Technol.*, vol. 61, no. 4, pp. 1874–1883, May 2012.

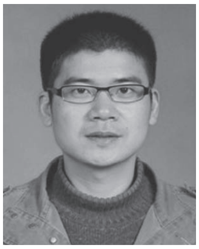
[27] J. L. Huang, L. Y. Yeh, and H. Y. Chien, "AKABA: An anonymous batch authenticated and key agreement scheme for value-added services in vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 1, pp. 248–262, Jan. 2011.

Xiaoyan Zhu received the B.E., M.E., and Ph.D. degrees in 2000, 2004, and 2009, respectively, from Xidian University, Xi'an, China.

She is currently an Associate Professor with the School of Telecommunications Engineering, Xidian University. Her research interests include security and privacy for wireless networks, cloud computing, big data, etc.

Shunrong Jiang received the B.E. degree in information engineering from Chongqing University, Chongqing, China, in 2008 and the M.E. degree in computer science from Jiangsu University, Zhenjiang, China, in 2012. He is currently working toward the Ph.D. degree in communication and information systems with the School of Telecommunications Engineering, Xidian University, Xi'an, China.

His research interests include security and privacy for wireless networks, cloud computing, etc.



Liangmin Wang received the B.S. degree in computational mathematics from Jilin University, Changchun, China, in 1999 and the Ph.D. degree in cryptology from Xidian University, Xi'an, China, in 2007.

He is currently a Full Professor with the School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China. His current research interests include security protocols and Internet of Things.

Dr. Wang has been honored as a Wanjiang Scholar of Anhui Province since November 2013.

Hui Li received the B.Sc. degree from Fudan University, Shanghai, China, in 1990 and the M.Sc. and Ph.D. degrees from Xidian University, Xi'an, China, in 1993 and 1998, respectively.

In 2009, he was a Visiting Scholar with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. Since 2005, he has been a Professor with the School of Telecommunications Engineering, Xidian University. He is the author of over 130 papers in academic journals and conferences and the coauthor of two books. His research interests include cryptography, wireless network security, security and privacy in cloud computing, and information theory.

Dr. Li served as the Technical Program Committee Cochair for the Fifth International Conference on Information Security Practice and Experience and the IEEE Industrial Applications Society Meeting in 2009. He served as the General Cochair for the Third International ICST Conference on Forensic Applications and Techniques in Telecommunications, Information and Multimedia in 2010 (E-Forensic 2010), the Fifth International Conference on Provable Security (ProvSec), and the Information Security Conference (ISC) in 2011.