Message Authentication Using Proxy Vehicles in Vehicular Ad Hoc Networks

Yiliang Liu, Liangmin Wang, *Member, IEEE*, and Hsiao-Hwa Chen, *Fellow, IEEE*

***Abstract*—Normally, authentication in vehicular ad hoc networks (VANETs) uses public key infrastructure to verify the integrity of messages and the identity of message senders. The issues considered in the authentication schemes include the level of security and computational efficiency in the verification processes. Most existing schemes mainly focus on assuring the security and privacy of VANET information. However, these schemes may not work well in VANET scenarios. For instance, it is difficult for a roadside unit (RSU) to verify each vehicle's signature sequentially when a large number of vehicles emerge in the coverage areas of an RSU. To reduce the computational overhead of RSUs, we propose a proxy-based authentication scheme (PBAS) using distributed computing. In the PBAS, proxy vehicles are used to authenticate multiple messages with a verification function at the same time. In addition, the RSU is able to independently verify the outputs from the verification function of the proxy vehicles. We also design an expedite key negotiation scheme for transmitting sensitive messages. It is shown from the analysis and simulations that an RSU can verify 26 500 signatures per second simultaneously with the help of the proxy vehicles. The time needed to verify 3000 signatures in the PBAS can be reduced by 88% compared with existing batch-based authentication schemes.**

***Index Terms*—Key negotiation, privacy preservation, proxy-based authentication, proxy vehicle, vehicular ad hoc network (VANET).**

I. Introduction

VEHICULAR ad hoc networks (VANETs) have attracted a lot of attention due to their potential to offer a better driving experience and road safety, as well as many other value-added services [1], [2]. Security issues [3], [4] are critical in VANETs because many different forms of attacks [3] against VANETs may emerge due to the use of wireless devices in VANET communications. Such security attacks may lead to a bad user experience (thus causing the loss of revenue for those value-added service providers) or create even more catastrophic consequences, such as the loss of lives due to traffic accidents caused by the failure of VANET communications.

Some sophisticated security schemes have been proposed in the literature as an effort to ensure that all information exchanged in VANETs is authenticated and, thus, can be fully trusted. In particular, Raya and Hubaux presented a public key infrastructure (PKI)-based scheme for vehicular signature applications [1], where a roadside unit (RSU) verifies received messages one after another. Because vehicles normally forward messages on the fly at any time, it may not be predicted and known by the RSU. Moreover, those PKI-based schemes [1], [5], [6] are time-consuming processes and may fail to satisfy the computational efficiency requirement under dynamic traffic patterns, where the computational complexity and transmission overhead of RSUs linearly increase with the number of vehicles that need to be authenticated.

Zhang *et al.* in [7] introduced an efficient batch signature verification scheme for the communications between vehicles and RSUs, in which an RSU can verify multiple received signatures at the same time, such that the total verification time required can be significantly reduced. In their proposed scheme, an RSU can simultaneously verify approximately 1600 messages per second, which is not bad but is still not fast enough to meet the requirement of VANET authentication speed. According to the dedicated short-range communications (DSRC) protocol [8], [9], each vehicle broadcasts a traffic safety message every 100–300 ms. This implies that an RSU must verify around 2500–5000 messages per second when there are 500 vehicles within the coverage of an RSU, which is indeed a great challenge for any current batch-based digital signature scheme reported in the literature [10]–[13].

In this paper, our goal is to tackle the aforementioned efficiency problem of the existing authentication schemes. In particular, we will propose a proxy-based authentication scheme (PBAS) for this purpose. In this proposed scheme, each proxy vehicle plays an important role, which is adopted to authenticate multiple messages with the help of a verification function at the same time. This way, distributed computing can be used to shed the time-consuming centralized computing loads at RSUs. We also design a systematic and independent mechanism for RSUs to verify the output of the verification function from different proxy vehicles, by which an RSU can evaluate the validity levels of different messages in the same way as done in separate verification schemes. In addition, batch key negotiations can also be accomplished in the proposed scheme, in which an RSU can complete the batch process of vehicles' key negotiations by broadcasting a single message. Fig. 1 shows the main characteristic features of the proposed PBAS scheme.

Manuscript received May 20, 2014; revised August 31, 2014; accepted September 6, 2014. Date of publication September 17, 2014; date of current version August 11, 2015. This work was supported in part by the National Natural Science Foundation of China under Grant 61272074 and Grant 61472001 and in part by the Taiwan Ministry of Science and Technology under Grant 102-2221-E-006-008-MY3. The review of this paper was coordinated by Prof. Y. Qian.

Y. Liu is with the Department of Internet of Things Engineering, Jiangsu University, Zhenjiang 212013, China, and also with the Department of Engineering Science, National Cheng Kung University, Tainan 70101, Taiwan (e-mail: alanliuyiliang@gmail.com).

L. Wang is with the Department of Internet of Things Engineering, Jiangsu University, Zhenjiang 212013, China (e-mail: Jasonwanglm@gmail.com).

H.-H. Chen is with the Department of Engineering Science, National Cheng Kung University, Tainan 70101, Taiwan (e-mail: hshwchen@mail.ncku.edu.tw).

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.2014.2358633

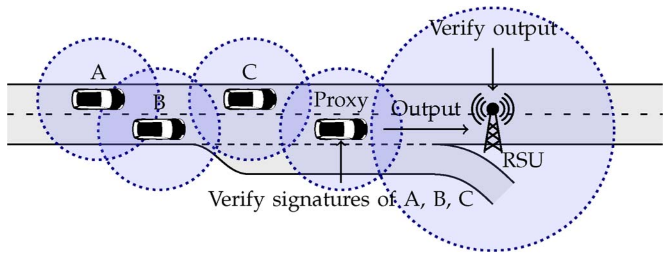


Fig. 1. PBAS reduces the computation load of RSUs via the cooperation among proxy vehicles, where a proxy vehicle verifies the signatures of A, B, and C with a verification function, and then, it transmits its output to a nearby RSU. The RSU verifies the output only, thus consuming fewer computing resources. Note that the verification functions perform cryptographic operations in an authentication scheme, and these operations are executed in RSUs using traditional authentication schemes.

Specifically, the design requirements of the proposed PBAS can be summarized as follows.

1) The scheme should be designed to meet the computational efficiency requirements of VANETs (see Section IV).
2) The scheme should be designed to meet the general security requirements of VANETs, such as message integrity and authentication, privacy preservation, etc. (see Section V).
3) The scheme has the property that enables the verification process to continue even in the event that a small number of proxy vehicles have been compromised in VANETs (see Section V).

The remainder of this paper is outlined as follows. Section II surveys the related works in the literature. Section III introduces the system and security models, together with the related preliminaries. Section IV discusses the issues on the proxy-based batch authentication scheme (PBAS). Section V conducts security performance analysis. Section VI is dedicated for complexity evaluation and simulations for the PBAS and the other existing authentication schemes, followed by the conclusions made in Section VII.

II. Related Works

In the current IEEE 1609.2 standard [9], vehicular communication messages should be authenticated using the elliptic curve digital signature algorithm (ECDSA) [14]. Each message also includes a certificate. As shown by the analytical study conducted in [15], a major challenge that remains to be tackled is to find a way to reduce the resource consumptions in computation and transmission. In the following, we will discuss the two authentication schemes that have been proposed in the literature.

A. Conventional Authentication

Let us start with the discussions about prior works on the ECDSA-based schemes and will take this scheme as an example to explain the important relationship between the integrity of messages and the validity of the sender's identities. Studer *et al.* pointed out in [5] that a VANET user needs to verify the validity of the identity of a message sender before verifying the integrity of the messages it sends out. If the system designers focus only on the mechanisms to verify messages and ignore the importance associated with the verification of valid entities, a malicious participant could exploit many forged identities to disable VANET communications. Therefore, they particularly proposed TESLA++ [5] as a modified version of TESLA [16], which combines the advantages of ECDSA signatures and TESLA. Compared with TESLA, TESLA++ has the advantage of having a relatively shorter hash message authentication code (HMAC) to verify the integrity of messages, which helps cut down the transmission overhead of RSUs. If compared with TESLA, TESLA++ signs on each message before its transmission, which is to perform the identity authentication that provides nonrepudiation of attribution in multihop communications. Any receiver can use the signer's public key to verify the identity of the message. To verify the messages from the vehicles outside the coverage of an RSU, Zhang *et al.* in [6] suggested that the neighboring vehicles could cooperatively work to probabilistically verify only a small percentage of these message signatures.

B. Batch Authentication

On the other hand, batch verification offers an efficient way for verifying signatures in VANETs. Zhang *et al.* in [7] introduced an identity-based batch signature verification (IBV) scheme for vehicular-to-infrastructure (V2I) communications, which works based on identity-based encryption algorithms [17], [18] proposed by Boneh *et al.* In the IBV scheme, an RSU can also verify multiple received signatures at the same time such that the computation time can be significantly reduced. Meanwhile, the certificates are not needed in the verification processes, and thus, the transmission overhead can be substantially reduced. The IBV scheme can achieve conditional privacy preservation using pseudo identities, and a trust authority (TA) is capable of tracing a vehicle's real identity from its pseudo identity. In [19], Zhang *et al.* made their effort to enhance the IBV scheme via adopting a group testing technique. The objective of the group testing is to find invalid signatures with a minimal batch verification workload. In [10], Huang *et al.* proposed an anonymous batch authenticated and key agreement (ABAKA) scheme for different value-added services, which can authenticate multiple messages sent from different vehicles and establish different session keys for different vehicles at the same time. The security of the ABAKA scheme is ensured based also on ECDSA. Compared with the basic ECDSA scheme, relatively short signatures are adopted by the ABAKA scheme to reduce the computation and transmission overheads of RSUs. In [20], Chim *et al.* introduced a secure and privacy enhancing communications scheme (SPECS), where any vehicle can form a group with the other vehicles after batch authentication and can communicate with one another securely without RSUs. However, in [11], Horng found out that the SPECS is vulnerable to impersonation attacks, and a malicious vehicle can act as an arbitrary vehicle to broadcast fake messages or even counterfeits another group member to send fake messages securely among themselves. To deal with this issue, they proposed b-SPECS+ to overcome the weaknesses of the SPECS. In [12], Shim proposed a conditional privacy preserving authentication scheme (CPAS), which is based on computational Diffie–Hellman (CDH), to bridge the gap between the privacy and nonrepudiation requirements. In [13], Li and Wang proposed a rapid certification scheme (RCS), in which a VANET leader is responsible for collecting the messages of n distinct vehicles and then sending them to an RSU. The RSU verifies the batch of messages. The RCS is able to reduce the transmission overhead of RSUs by integrating messages into batches.

C. Certificate Revocation

Based on the previous discussions, we understand that the optimized certificate update schemes [21]–[25] are promising approaches for efficient authentication in VANETs, but the revocation list will get very long when it is needed to check the time-consuming certificate revocation lists (CRLs). In [25], Wasef and Shen introduced a protocol for vehicle-to-vehicle (V2V) communications, which is called the expedite message authentication protocol, which uses a keyed-HMAC technique to replace the CRL checking process. It can help reduce the computation overhead compared with the conventional schemes employing the CRL.

D. Tradeoff Between Privacy and Nonrepudiation

The authentication schemes require that vehicles in VANETs should publish their certificates or public keys. Even in identity-based signature algorithms, their identifications should be sent to the destination together with their messages. The privacy issues have attracted much attention because these identity materials are revealed in VANETs. Lu *et al.*, in [26], and Sun *et al.*, in [27], proposed pseudonym-changing-based authentication schemes to achieve conditional privacy. The term "conditional" here means that when car attacks occur, the identity information has to be revealed by the TA to establish the liability of the attacks. In [28], Choi and Jung considered that the TA has all cryptographic materials and may abuse its access ability, and thus, they proposed a security framework with nonrepudiation and conditional privacy, in which the TA never knows the user's private key. Lu *et al.*, in [29], and Zhu *et al.*, in [30], employed a lightweight conditional privacy-preservation scheme with a simple hash-chain technique, which attributes to the reduction in the computational overhead while achieving conditional privacy.

E. MAC

Future vehicles will have the functions of medium access control (MAC) protocols so that the passengers can surf the Internet in the vehicles. In [31], Qian *et al.* provided a secure MAC protocol to access DSRC channels. The secure communication protocol is designed based on an authentication scheme to satisfy the requirements of message authentication and integrity, together with nonrepudiation and privacy of senders. The protocol takes advantage of the time-stamp mechanism to guarantee the freshness of messages.

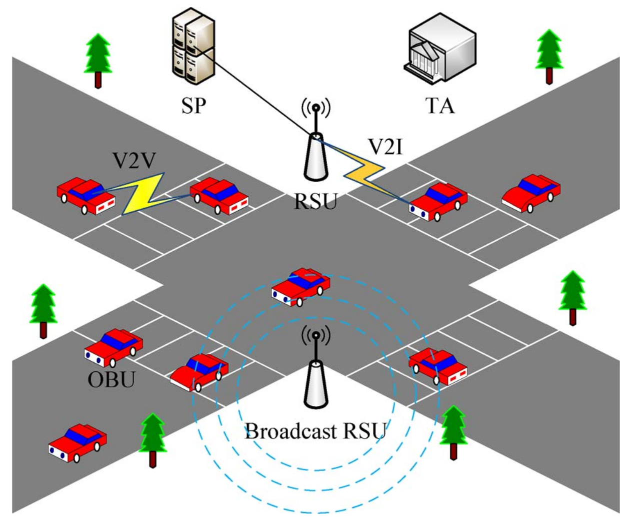


Fig. 2. VANET communication system is supported by DSRC, which offers V2V and V2I communications. The system should meet the security requirements to ensure that all information data exchanged are authenticated and can be trusted [9].

III. Preliminaries

Before introducing the PBAS proposed in this paper, we would like to offer a brief review of the preliminary knowledge on VANET security in the following sections, to facilitate the discussions and performance analysis on the proposed scheme, which will be presented in Section IV.

A. Bilinear Pairing

Let $\mathbb{G}_1$ denote an additive group of prime order q and $\mathbb{G}_2$ denote a multiplicative group of the same prime order. Let P be a generator of $\mathbb{G}_1$ and $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be a bilinear mapping with the following properties.

- Bilinear: For all P, R, $Q \in \mathbb{G}_1$, and $a, b \in \mathbb{Z}_q^*$, we have $e(Q, P + R) = e(P + R, Q) = e(Q, P) \times e(Q, R)$. In particular, $e(aP, bQ) = e(P, Q)^{ab}$.
- Nondegeneracy: $e(P, Q) \neq 1$.
- Computability: The map e is efficiently computable.

Next, we state the following two underlying problems as the basis for our proposed scheme.

- CDH problem: For unknown $a, b \in Z_q^*$ and for the given $aP, bP \in \mathbb{G}_1$, compute P^{ab}.
- Decisional Diffie–Hellman (DDH) problem: For unknown $a, b \in Z_q^*$ and for the given $aP, bP, abP \in \mathbb{G}_1$, check if $e(aP, bP) \stackrel{?}{=} e(abP, P)$.

It is easy to show that the DDH problem is easy to solve, whereas the CDH problem is extremely hard to solve.

B. System Model

Fig. 2 introduces a two-layer network model of VANETs with its underlined security layer and communication layer. The security layer is comprised of a TA and tamper-proof devices. The TA is trusted by all entities in the system, it is in charge of distributing the secret keys to all entities, and it has an ability for tracing back to the real identity of a vehicle whenever any uncertainty occurs. According to the VANET standard [8], [9], a tamper-proof device installed in the onboard unit of a vehicle is responsible for storing security materials and implementing all crypto operations. On the other hand, the communication layer is comprised of V2I and V2V modules. The V2V communication system provides a 360° view of all its peer vehicles within the communication range. The V2I communication and broadcast systems provide traffic and entertainment information for the drivers.

C. Security Model

In [1], Raya and Hubaux defined five basic attacks, including bogus information, cheating with sensor information, ID disclosure of the other vehicles to track their locations, denial of service, and masquerading. Samara *et al.* in [32] and Papadimitratos and Hubaux in [33] extended the attack types by introducing a replay attack. Here, we take all those basic attacks into consideration in a VANET of interest, except for "cheating by sensor information" because the research on this particular topic belongs to data-centric trust establishment [34]–[36].

The work reported in [37] indicated that security mechanisms of the VANET framework should support different applications and services. Hence, before discussing the security requirements of our scheme, we first consider two application scenarios, namely, safety-related applications and value-added applications. For the safety-related applications, vehicles in danger will send (broadcast or unicast) safety-related messages to other entities in VANETs. The entities need to authenticate these messages before utilizing them. In the safety-related applications, there are typically no confidentiality requirements on these safety-related messages. For the value-added applications, confidentiality is required. RSUs are registered as the gateways for Internet access, via which the vehicles that request for the services can establish secrecy links with the Internet service provider (ISP) because most of the services levy charges. Hence, the messages from the ISP can satisfy confidentiality through the key generation process between vehicles and RSUs. In summary, the following four security requirements are needed in the PBAS.

1) *Message integrity and authentication:* Messages sent by vehicles can be authenticated to prove that they are indeed sent by authorized entities without being modified or forged. Moreover, RSUs should have an ability to authenticate a large amount of signatures for many vehicles.
2) *Identity privacy preserving and traceability:* The real identity of a vehicle should be kept anonymous, which is heterogeneous with the other pseudo identities. Any third party should not be able to reveal the real identity of a vehicle by analyzing multiple messages sent from it. However, when the vehicles send malicious information, the TA has an ability to reveal the real identities from the pseudo identities of the misbehaved vehicles.
3) *Resisting signature replay attacks:* Signature replay attacks can be prevented by such a carefully designed scheme. The definition of a signature replay attack can be generalized as an attack that replays the signatures from a different vehicle for the intended or expected RSUs, thereby to fool the RSUs to believe that they have successfully completed the verification of the owner of these signatures.
4) *Confidentiality:* A server can establish a secure communication link with a requesting vehicle for subsequent communications. For instance, ISP and parking payment systems require that the session key negotiation process generates the keys for confidentiality of their transmitted messages.

TABLE I
NOTATIONS AND THEIR DEFINITIONS

Notation	Definition
V_i	The ith vehicle.
s_j	The jth private master key of the tamper proof device, where $j \in \{1, 2, 3\}$.
s_r	The private master key of the RSU.
PK_k	The public key of the system, where $k \in \{1, 2, r\}$.
ID_i	The pseudo identity of the vehicle V_i, where $ID_i = (ID_i^1, ID_i^2)$.
SK_i	The private key of the vehicle V_i, where $SK_i = (SK_i^1, SK_i^2)$.
RID_i	The real identity of the vehicle V_i, where $RID_i \in \mathbb{G}$.
R_r	The real identity of the RSU, where $R_r \in \mathbb{G}$.
SK_r	The private key of the RSU, where $SK_r = (SK_r^1, SK_r^2)$.
T	The time stamp of messages.
M_i	A message from vehicle V_i.
$h(\cdot)$	A one-way hash function such as SHA-1, SHA-2.
$H(\cdot)$	A MapToPoint hash function such as $H : \{0,1\}^* \to \mathbb{G}$.

IV. Proxy-Based Batch Authentication

Here, we introduce the PBAS, whose algorithm consists of the following four phases: 1) system initialization phase; 2) message signing phase; 3) batch verification by proxy vehicles; and 4) verification by an RSU at the outputs from proxy vehicles. In addition, a key negotiation phase is included if confidentiality is required. The notations used here are listed and defined in Table I.

A. System Initialization

The TA (as shown in Fig. 2) initializes the system parameters for each registered VANET member. Each vehicle generates its pseudo identity and the corresponding key. According to the IEEE standard for VANETs [9], each vehicle should be equipped with a tamper proof device, and no adversary can attain any data stored in the tamper-proof device. The system initialization phase can be mathematically modeled as follows.

- *System parameter generation*:
    1) The TA stores $U_V = \{RID_i | 1 \leq i \leq n\}$.
    2) Given the bilinear parameters $(P, q, \mathbb{G}_1, \mathbb{G}_2, e)$, the TA chooses four random numbers, i.e., $s_1, s_2, s_3, s_r \in \mathbb{Z}_q^*$.
    3) The TA computes $PK_1 = s_1 P$, $PK_2 = s_2 P$, and $SK_r^2 = s_3 P$.
    4) The tamper-proof device of each vehicle is secretly preloaded with the parameters $\{s_1, s_2, s_3\}$.
    5) The RSUs are secretly preloaded with the parameters $\{SK_r^2, s_r\}$.

- *Pseudonym and key generation*:
  1) The RSU computes $SK_r^1 = s_r R_r$ and $PK_r = s_r P$. Therefore, the private key of the RSU can be modeled as (SK_r^1, SK_r^2).
  2) A vehicle, which is denoted by V_i, chooses a random number $r_i \in \mathbb{Z}_q^*$.
  3) V_i computes $ID_i^1 = r_i P$ and $ID_i^2 = RID_i \oplus H(r_i PK_1)$.
  4) V_i computes $SK_i^1 = s_1 ID_i^1$ and $SK_i^2 = s_2 H(ID_i^1 \| ID_i^2)$.
- *Publishing the system parameters*:
  1) The system parameters $(P, q, \mathbb{G}_1, \mathbb{G}_2, e, PK_1, PK_2, PK_r)$ are preloaded by each VANET member.

The system initialization phase should keep private material confidentiality. First, the private master keys $\{s_1, s_2, s_3\}$ are loaded into the vehicle's tamper-proof device in the system parameter generation process. Any adversary cannot extract any data stored in the device. Second, the tamper-proof device is responsible for generating the identity (ID_i^1, ID_i^2) and the corresponding privacy key (SK_i^1, SK_i^2) in the pseudonym and key generation process. Their security is ensured based on the cyclic group discrete logarithm problem, so that none can get s_1 and s_2 from the private key.

The identity information (ID_i^1, ID_i^2) is a pseudonym that can achieve privacy preservation, because vehicle V_i will generate a new pseudo identity when entering into the communication range of another RSU, where $ID_i^1 = r_i \cdot P$, $ID_i^2 = RID_i \oplus H(r_i \cdot PK_1)$, and r_i should be different in different areas. The private keys (SK_r^1, SK_r^2) are used as the verifiers by the RSU, which are calculated without r_i. Hence, when a vehicle leaves the coverage area of an RSU and enters the coverage area of another RSU, the new RSU can continue to verify their messages with the primary system parameters.

B. Message Signing

The vehicles in a VANET will periodically broadcast messages. To ensure the integrity of messages and the validity of the originators, each message sent by a vehicle should be signed with its private key. The message signing phase can be modeled as follows.

1) Vehicle V_i, where $i \in (1, 2, 3, \ldots, n)$, generates related information M_i, where $M_i = \mathcal{M} \| T$.
2) V_i picks up a pseudo identity ID_i and the corresponding private key SK_i from the tamper-proof device. Then, V_i signs on message M_i, where $\sigma_i^1 = SK_i^1 + h(M_i) SK_1^2$.
3) The tamper-proof device of V_i generates σ_i^2 with s_3, where $\sigma_i^2 = (r_i + s_3(h(M_i) + \sigma_i^1)) PK_r$.
4) Then, V_i sends the message $\{ID_i, M_i, \sigma_i^1, \sigma_i^2\}$ to the other participants in the vicinity.

From the given discussions, one can see that, compared with conventional signatures generated by private keys, we combine PK_r and the private keys of vehicles to sign vehicular messages.

Given a message from a vehicle, the signature attached within the message is shorter than the current standard ECDSA of IEEE1609.2 [9]. With a 160-bit q cyclic group $\mathbb{G}$,[1] the length of a signature in the PBAS is only half that of the ECDSA, i.e., $|\sigma_i^1| = 21$ bytes.[2] Similarly, σ_i^2 has the same length, i.e., $|\sigma_i^2| = 21$ bytes. In addition, our signature scheme is an identity-based encryption algorithm, which makes the mapping between identities publicly available. Therefore, the certificate is unnecessary when verifying messages. In other words, only a short-length pseudo identity is sent, i.e., $|ID_i| = |ID_i^1| + |ID_i^2| = 42$ bytes. Conclusively, the signature size of a vehicular message is 84 bytes, i.e., $|ID_i| + |\sigma_i^1| + |\sigma_i^2| = 84$ bytes. Nevertheless, the current standard ECDSA uses 256-byte signature.

C. Batch Verification by Proxy Vehicles

Proxy vehicles can efficiently authenticate multiple messages sent from the other vehicles and then output the result of their authentication process and send it to the entities that have relatively low computing capabilities.

First of all, we propose an efficient proxy vehicle selection strategy. It is crucial to make sure that vehicles have extra computation resources to serve for the others. We consider that u vehicles in the area can communicate with each other directly. Each of them needs to sign and send a messages. We assume that C_v is the cost of authenticating one signature in the PBAS, which is undertaken by the proxy vehicles. C_s is the cost of generating one signature. The total computation load of each vehicle V_i is $C_i, i \in \{0, u\}$. The proxy vehicle selection strategy is explained as follows.

1) When extra resource $C_r^i = C_i - aC_s$ satisfies $C_r^i > 0$, V_i is qualified to be a candidate of a proxy vehicle.
2) According to C_r^i, denote these extra resources by $\{c_1, c_2, \ldots, c_{v_{r,0}}\}$ in a descending order and the corresponding vehicles by $\{p_1, p_2, \ldots, p_{v_{r,0}}\}$, where $0 \leq v_{r,0} \leq u$.
3) Use the median C_{me} of $\{c_1, c_2, \ldots, c_{r,0}\}$ as a threshold and select the proxy vehicles based on $C_r^i > C_{\text{me}}$. The proxy vehicles are defined as $\{p_1, p_2, \ldots, p_v\}$, where v is the number of proxy vehicles.
4) Each proxy vehicle authenticates the same number of signatures based on the threshold C_{me}, which is defined as $(C_{\text{me}} - aC_s)/C_v$.

The given selection strategy can be implemented in every vehicle without the TA. Based on (C_v, C_s, C_i), vehicles can become proxy vehicles spontaneously to authenticate received signatures. When there are no computation resources in the vehicles, the PBAS degenerates to normal authentication schemes. The signature scheme is based on ID cryptography, and RSUs are not required to prestore the certificates of proxy vehicles. Hence, there is no upper limit in the number of proxy vehicles that could be governed by an RSU.

The verification phase in a proxy vehicle can be described as follows.

1) The messages $\{ID_i, M_i, \sigma_i^1, \sigma_i^2\}$ sent by V_i, $i \in (1, 2, 3, \ldots, n)$, are received by a proxy vehicle V_{proxy}.

[1]Every finite cyclic group $\mathbb{G}$ is isomorphic to a group $\mathbb{Z}_q^*$, where q is the order of the group, and the security of a signature is based on $\mathbb{Z}_q^*$.

[2]We use an MNT curve with 160-bit q, which has the same security level with IBV, b-SPECS+, and CPAS.

Then, V_{proxy} verifies that the signatures in batch, i.e., $\sigma_i^1, i \in (1, 2, 3, \ldots, n)$, are valid if the following equation holds:

$$e\left(\sum_{i=1}^{n}\sigma_i^1, P\right) = e\left(\sum_{i=1}^{n} ID_i^1, PK_1\right) \times e\left(\sum_{i=1}^{n} h(M_i)H(ID_i^1 \| ID_i^2), PK_2\right). \quad (1)$$

Before the verification process, the proxy vehicle has obtained the public key (PK_1, PK_2), received the message M_i, the signature σ_i of M_i, and the pseudo identity (ID_i^1, ID_i^1) from each surrounding vehicle V_i. Then, $e(\sum_{i=1}^{n}\sigma_i^1, P)$ and $e(\sum_{i=1}^{n} ID_i^1, PK_1)$ $e(\sum_{i=1}^{n} h(M_i)H(ID_i^1 \| ID_i^2), PK_2)$ can be calculated by the proxy vehicle, respectively. If these two terms are indeed identical, the integrity of all messages and the identities of senders of these messages are verified. The validity of (1) can be verified as follows:

$$\begin{aligned}
e&\left(\sum_{i=1}^{n}\sigma_i^1, P\right) \\
&= e\left(\sum_{i=1}^{n}\left(SK_i^1 + h(M_i)SK_i^2\right), P\right) \\
&= e\left(\sum_{i=1}^{n} SK_i^1, P\right) e\left(\sum_{i=1}^{n} h(M_i)SK_i^2, P\right) \\
&= e\left(\sum_{i=1}^{n} s_1 ID_i^1, P\right) e\left(\sum_{i=1}^{n} s_2 h(M_i)H\left(ID_i^1 \| ID_i^2\right), P\right) \\
&= e\left(\sum_{i=1}^{n} ID_i^1, s_1 P\right) e\left(\sum_{i=1}^{n} h(M_i)H\left(ID_i^1 \| ID_i^2\right), s_2 P\right) \\
&= e\left(\sum_{i=1}^{n} ID_i^1, PK_1\right) e\left(\sum_{i=1}^{n} h(M_i)H\left(ID_i^1 \| ID_i^2\right), PK_2\right).
\end{aligned} \quad (2)$$

2) Then, V_{proxy} computes $\Sigma_{i=1}^{n}\sigma_i^1 \in \mathbb{Z}_q^*$, $\Pi_{i=1}^{n}\sigma_i^2 \in \mathbb{Z}_q^*$ and sends $\{M_{\text{proxy}}, ID_{\text{proxy}}, \sigma_{\text{proxy}}^1\}$ to an RSU, where the output denotes $M_{\text{proxy}} = \mathcal{M} \| \Sigma_{i=1}^{n}\sigma_i^1 \| \Pi_{i=1}^{n}\sigma_i^2 \| T \| ID_i$, $i \in (1, 2, 3, \ldots, n)$, and the verification result is generated by the proxy vehicle and included in $\mathcal{M}$. Here, $\{\mathcal{M} = a\}$ indicates that the batch of messages is valid, and $\{\mathcal{M} = b\}$ indicates that the batch of messages is invalid. Signature σ_{proxy}^1 is generated with V_{proxy}'s privacy key $(SK_{\text{proxy}}^1, SK_{\text{proxy}}^2)$.

Given n distinct messages authenticated by a proxy vehicle, an RSU does not need to receive all the signatures because these signatures have been calculated as $\Sigma_{i=1}^{n}\sigma_i^1$ and $\Pi_{i=1}^{n}\sigma_i^2$. Each proxy vehicle sends the message $\{M_{\text{proxy}}, ID_{\text{proxy}}, \sigma_{\text{proxy}}^1\}$ to the RSU. The length of the packet is $126 + 42n$ bytes, where $|ID_{\text{proxy}}| = 42$ bytes, $|\sigma_{\text{proxy}}^1| = 21$ bytes, and $|M_{\text{proxy}}| = |\Sigma_{i=1}^{n}\sigma_i^1| + |\Pi_{i=1}^{n}\sigma_i^2| + |ID_i| = 42 + 42n$ bytes. Thus, the signature size sent to the RSU can be significantly reduced compared with IBV, whose signature size is $63n$ when sending n messages [7].

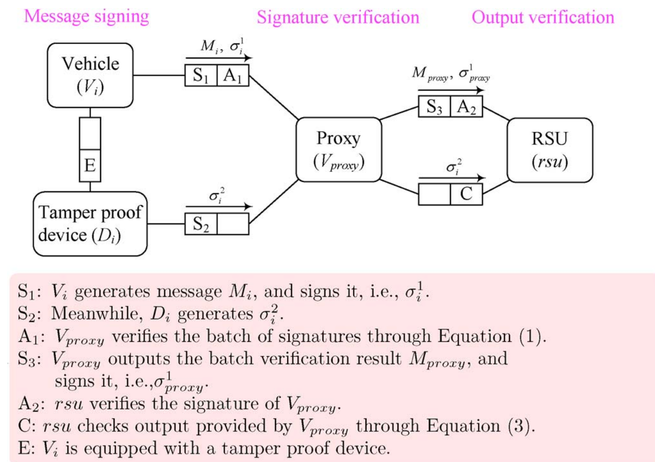


Fig. 3. PBAS, where the proxy vehicles are used for verifying the messages of nearby vehicles to replace time-consuming centralized verification in one RSU.

D. Verification by an RSU at Outputs From Proxy Vehicles

RSUs can independently verify the results from the previous verification processes of the proxy vehicles, and then, the system can exclude false results and revoke malicious proxy vehicles. The verification in an RSU at the outputs from the proxy vehicles includes the following three tasks. Task (1) ensures that the originators of the messages is indeed the real proxy vehicle and that there are no forwarding nodes actively modifying messages; Task (2) guarantees that the result from a proxy vehicle contains correct verification output through their batch verification phase; and Task (3) revokes the proxy vehicle when the RSU finds that it fails the process.

This process can be described as follows.

1) When receiving $\{M_{\text{proxy}}, ID_{\text{proxy}}, \sigma_{\text{proxy}}^1\}$, the RSU initiates Task (1) to verify if single signature σ_{proxy}^1 is valid. The single signature verification process has been proposed and proved in [7]. If it is valid, then the TA traces the real identities of this batch of vehicles by computing $RID_i = ID_i^2 \oplus H(s_1 \cdot ID_i^1)$, where $i \in (1, 2, 3, \ldots, n)$.
2) If Task (1) is passed, the RSU goes to Task (2) to check the authentication result sent by a proxy vehicle. The result is valid, and the batch of messages is authenticated if the following equation holds:

$$e\left(\prod_{i=1}^{n}\sigma_i^2, R_r\right) = e\left\{\prod_{i=1}^{n} ID_i^1 \left[\sum_{i=1}^{n}\left(h(M_i) + \sigma_i^1\right)\right] SK_r^2, SK_r^1\right\}. \quad (3)$$

As the RSU has already obtained its private key (SK_r^1, SK_r^2) and extracted $\Sigma_{i=1}^{n}\sigma_i^1$, $\Pi_{i=1}^{n}\sigma_i^2$ from the message M_{proxy} before, in the verification process, the RSU calculates $e(\prod_{i=1}^{n}\sigma_i^2, R_r)$ with R_r, and $e\{\prod_{i=1}^{n} ID_i^1$ $[\sum_{i=1}^{n}(h(M_i) + \sigma_i^1)]SK_r^2, SK_r^1\}$, respectively. If these two terms are identical, the result is valid, and the batch of messages is correctly authenticated by the proxy vehicle. The related details of the PBAS are shown in Fig. 3.

3) If (3) is not held, the proxy vehicle is considered malicious by the RSU. The TA receiving the feedback from the RSU will revoke the malicious proxy, which can prevent it from disturbing the authentication processes later. The algorithm to identify malicious proxy vehicles is Algorithm 1. The security explanation of Steps 8–13 is given in Section V-A.

Algorithm 1 The algorithm to identify malicious proxy vehicles.

1: The batch of messages is marked valid by $\{\mathcal{M} = a\}$.
2: The batch of messages is marked invalid by $\{\mathcal{M} = b\}$.
3: Task (1): verify the message M_{proxy} from the proxy vehicle:
4: **if** M_{proxy} is valid **then**
5:     Task (2): verify the result of the proxy vehicle:
6:     **if** $\{\mathcal{M} = a\|$ (3) is held **then**
7:         The batch of messages is valid, and the proxy vehicle is trusted.
8:     **else if** $\mathcal{M} = a\|$ (3) is not held **then**
9:         The batch of messages is invalid, and the proxy vehicle is untrusted.
10:         TA revokes the proxy vehicle.
11:     **else if** $\mathcal{M} = b\|$ (3) is held **then**
12:         The batch of messages is valid, and the proxy vehicle is untrusted.
13:         TA revokes the proxy vehicle.
14:     **else**
15:         The validity of the batch of messages is hard to determine, and the proxy vehicle is untrusted.
16:         TA revokes the proxy vehicle.
17: **else**
18:     The result message is not from the authentic proxy vehicle.

The validity of (3) can be verified as follows:

$$
\begin{aligned}
&e\left(\prod_{i=1}^{n}\sigma_i^2, R_r\right)\\
&= e\left\{\left[\sum_{i=1}^{n} r_i + s_3\left(\sum_{i=1}^{n} h(M_i) + \sum_{i=1}^{n}\sigma_i^1\right)\right] PK_r, R_r\right\}\\
&= e\left\{\left[\sum_{i=1}^{n} r_i + s_3\left(\sum_{i=1}^{n} h(M_i) + \sum_{i=1}^{n}\sigma_i^1\right)\right] s_r P, R_r\right\}\\
&= e\left\{\left[\sum_{i=1}^{n} r_i + s_3\left(\sum_{i=1}^{n} h(M_i) + \sum_{i=1}^{n}\sigma_i^1\right)\right] P, s_r R_r\right\}\\
&= e\left\{\left(\sum_{i=1}^{n} r_i\right) P \cdot s_3\left[\sum_{i=1}^{n}\left(h(M_i) + \sigma_i^1\right)\right] P, s_r R_r\right\}\\
&= e\left(\prod_{i=1}^{n} ID_i^1\left[\sum_{i=1}^{n}\left(h(M_i) + \sigma_i^1\right)\right] SK_r^2, SK_r^1\right).
\end{aligned}
\tag{4}
$$

The computation cost that an RSU spends on verifying n signatures is equivalent to that spent on checking a proxy vehicle's operation. From the given discussions, the total cost consists of two pairing operations and one multiplication. However, in IBV [7], the cost that an RSU spends on verifying n signatures is comprised of n multiplications and three pairing operations.

The batch key generation process is used when some of the vehicles want to establish secrecy links with an RSU. This process can be described as follows.

1) If confidentiality is required, the RSU chooses a random number $z \in \mathbb{Z}_q^*$ and then computes $Pub_r = zP$ and $\sigma_r = sig_{SK_r^1}(Pub_r\|T)$. The RSU calculates the session key, i.e., $K_{r_i} = z \cdot ID_i^1, i \in (1, 2, 3, \ldots, n)$, for each vehicle.
2) The RSU broadcasts a single message, i.e., $\{Pub_r, T, \sigma_r\}$. Vehicles that apply for confidentially establishing communications will verify the signature sent by the RSU with PK_r to ensure the validity of the RSU and the integrity of the broadcast messages.
3) Finally, V_i calculates the session key, i.e., $K_{r_i} = r_i \cdot Pub_r$.

The RSU just needs to generate only one single message to broadcast for a batch of key negotiations. Note that the session keys are distinct because of different vehicles' contributions on r_i. The broadcast message $\{Pub_r, T, \sigma_r\}$ by the RSU consists of a 21-byte public parameter, 21-byte signature, i.e., $|Pub_r| + |\sigma_r| = 42$ bytes. It is noted that the key negotiations and the traceability process can be generated offline in a server. Therefore, the key negotiation process will not impose much computational burden on the RSU.

V. SECURITY PERFORMANCE

Here, we analyze the security and fault tolerance performance of the PBAS. The security analysis of the PBAS includes the following four aspects, i.e., message integrity and authentication, replay attack resistance, nonrepudiation, and privacy preservation. Particularly, the message integrity and authentication is one of the basic security requirements in VANETs. The fault tolerance of the PBAS is defined as the property that enables the verification process to continue operating properly even in the presence of a small number of compromised proxy vehicles in VANETs. If its operational quality degrades, the degradation is proportional to the number of compromised proxy vehicles, as compared with a naively designed system, in which even a very small failure can cause the breakdown of an entire system.

A. Security Analysis

1) Message Integrity and Mutual Identity Authentication: The PBAS achieves mutual identity authentication between RSUs and vehicles. To be authenticated by RSUs, V_i generates signature σ_i^1 of message M_i with its privacy key SK_i. Another signature σ_i^2 is generated by a tamper-proof device of V_i. Without knowing SK_i and s_3, no attacker can forge neither a message nor the corresponding signature. Similarly, without knowing the RSU's privacy key SK_r^1, it is computationally infeasible to forge a valid pair $(\sigma_r, (Pub_r, T))$.

Let us consider a scenario where the attackers are divided into external and internal attackers. The external attackers are nonauthorized entities and can only attain public parameters and public keys. The internal attackers are authorized vehicles (such as V_i), each of which knows its own privacy key (SK_i^1, SK_i^2), but it cannot extract s_1, s_2, s_3 stored in the tamper-proof device.

In the PBAS, (SK_i^1, SK_i^2) of V_i is changed when the vehicle enters into another coverage area. Without knowing (SK_i^1, SK_i^2), it is impossible to forge a valid signature $\sigma_i = SK_i^1 + h(M_i)SK_i^2$. Due to the CDH problem in $\mathbb{G}$, it is infeasible to obtain s_1 and s_2 from PK_1 and PK_2. Therefore, any attacker cannot obtain the privacy keys of the others.

If malicious proxy vehicles send bogus results to an RSU, the PBAS is secure against these additional attacks as listed in *Challenge 1* and *Challenge 2*.

Challenge 1: A proxy vehicle may fool RSUs using the following two possible ways: 1) All messages in a batch are valid, but a proxy vehicle claims that there are invalid messages in the batch; 2) there are some invalid messages in a batch, but a proxy vehicle claims that they are all valid.

Resistance: In both cases, $\{M_{\text{proxy}}, ID_{\text{proxy}}, \sigma^1_{\text{proxy}}\}$ is still sent to an RSU under the mechanism of the PBAS, where the output denotes $M_{\text{proxy}} = \mathcal{M}\|\Sigma_{i=1}^n \sigma_i^1\|\Pi_{i=1}^n \sigma_i^2\|T\|ID_i$, where $i \in (1, 2, 3, \ldots, n)$. It is difficult to forge a valid signature σ_i^1 and its corresponding σ_i^2 by cryptography analysis. In addition, without s_3, the attacker is impossible to calculate $\Pi_{i=1}^n \sigma_i^2$. It is also infeasible to obtain s_3 from the formula $\sigma_i^2 = (r_i + s_3(h(M_i) + \sigma_i^1))PK_r$ because of the CDH problem in $\mathbb{G}$. On the other hand, without (SK_r^1, SK_r^2), a malicious proxy vehicle cannot calculate $\Sigma_{i=1}^n \sigma_i^1$ and $\Pi_{i=1}^n \sigma_i^2$ from (2) directly, as it is an NP-hard problem. Therefore, the PBAS is secure against the attacks of forging bogus results and the corresponding signatures, i.e., $\Sigma_{i=1}^n \sigma_i^1$ and $\Pi_{i=1}^n \sigma_i^2$. The RSU verifies whether the output of the batch verification by a proxy vehicle is valid with (2).

Challenge 2: Given the pseudo identity of V_i, i.e., (ID_i^1, ID_i^2), and the pseudo identity of V_j, i.e., (ID_j^1, ID_j^2), a malicious attacker attempts to confuse the sequence of these authorized vehicles' identities, i.e., (ID_i^1, ID_j^2) and (ID_j^1, ID_i^2), to prevent the TA from tracing the vehicle's real identity.

Resistance: In this case, (2) still holds, but the TA should compute $RID_i = ID_i^2 \oplus H(s_1 \cdot ID_i^1)$, $i \in (1, 2, 3, \ldots, n)$, to trace the real identities in the server. The TA will not attain a valid RID_i if the relationship between ID_i^1 and ID_i^2 is confused.

2) Replay Attack Resistance: To guarantee the freshness of messages, T_R denotes the arrival time of a received message, and T denotes the message departure time. Δt_1 denotes the time difference between the vehicle's clock and local clock, and Δt_2 denotes the expected network delay. Upon receiving a message, the PBAS first checks whether the following inequality is valid:

$$|T_R - T| < \Delta t_1 + \Delta t_2. \tag{5}$$

If T is lapsed, then the receivers drop the message.

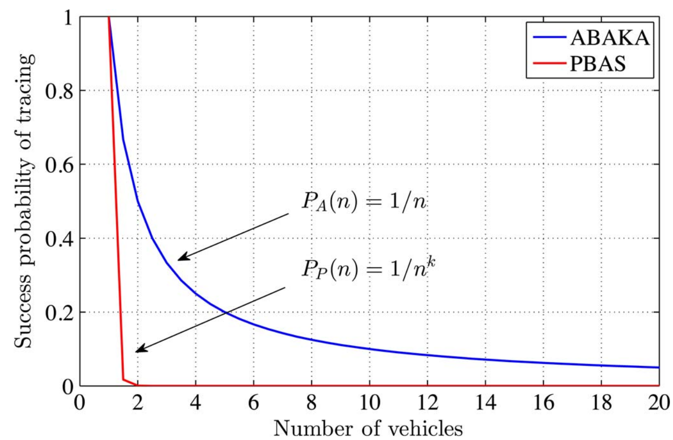


Fig. 4. Comparison of the authentication schemes (ABAKA and PBAS) in terms of the probabilities for an attacker to successfully trace a vehicle from the pseudo identities of n vehicles. The pseudo identity of a vehicle in the PBAS will dynamically change k times in a specified period, whereas the pseudo identity of a vehicle in ABAKA is static. Here, $k = 10$.

3) Nonrepudiation: Given the pseudo identity (ID_i^1, ID_i^2), only the TA can trace the real identity of a vehicle by s_1, or

$$\begin{aligned} ID_i^2 \oplus H\left(s_1 \cdot ID_i^1\right) &= RID_i \oplus H\left(r_i \cdot PK_1\right) \oplus H\left(s_1 r_i \cdot P\right) \\ &= RID_i. \end{aligned} \tag{6}$$

Finally, the real identity RID_i of a vehicle is obtained. The TA will store (RID_i, K_{r_i}) to encrypt the subsequent sensitive messages of services. When a traffic accident occurs, the law enforcement departments can punish the driver of the vehicle.

4) Privacy Preservation: In the PBAS, a vehicle takes advantage of its identity as the public key to reduce the size of signatures, but the disclosure of identity may cause privacy violations. Thus, the real identity RID_i of vehicle V_i is converted into two area-sensitive pseudo identity (ID_i^1, ID_i^2) for privacy preservation, where $ID_i^1 = r_i \cdot P$, and $ID_i^2 = RID_i \oplus H(r_i \cdot PK_1)$. Because V_i will regenerate the secret key r_i when entering into a new communication area of another RSU, the pseudo identity and signature, σ_i^1, σ_i^2, will dynamically change with the secret key. Without knowing s_1, it is impossible to calculate the real identity.

We use a probability model to analyze the relationship between the probability that an attacker can successfully trace a vehicle and the number of vehicles in the range of an RSU. The successful traceability probability denotes the capability to distinguish one vehicle from the pseudo identities of vehicles in a given period. As shown in Fig. 4, we assume that the number of vehicles is n. In ABAKA [10], the attacker should trace a vehicle by selecting the static pseudo identities, where the probability is $P_A(n) = 1/n$. In our scheme, the probability to trace a certain vehicle is $P_P(n) = 1/n^k$, where k is the frequency of changing pseudo identities. If a vehicle passes ten different coverage areas, then we have $k = 10$.

B. Fault Tolerance Analysis

The PBAS takes advantage of the proxy vehicles to realize efficient verification. However, the limitation of the proxy-based verification is that once a proxy vehicle is compromised, its security performance decreases such that the entire verification process in a batch through the compromised proxy vehicles may lose its efficiency.

The compromised proxy vehicle may come from a variety of ways, such as a loss or misconfigured device, as well as attackers. According to the simulations conducted in Section IV, the average packet loss ratio is generally lower than 0.1%. If a proxy vehicle sends an inaccurate message through a misconfigured device, an RSU can detect it when the verification process fails and then revoke the vehicle's certificate. Another situation is that a malicious proxy vehicle forges or tampers several verification messages to pass the batch authentication. This behavior can also be detected with the help of our scheme, as mentioned in Section V. Once a malicious proxy vehicle is detected by the RSU, the TA will also revoke the malicious proxy vehicle's certificate, and this can prevent the malicious proxy vehicle from disturbing the authentication processes later.

Although a compromised vehicle cannot obtain any sensitive information, such as others' secret keys and secret parameters, the server will spend some amount of time in locating the misbehaving nodes, which leads to performance degradation for our scheme. Considering the given three cases, we assume that, at most, r percent of proxy vehicles are compromised and send invalid messages. N_v denotes the number of vehicles, and N_p denotes the number of proxy vehicles. Thus, $N_c = N_p \times r$ denotes the largest number of compromised vehicles in a batch period. We also assume that a proxy vehicle can verify, at most, n messages, and each vehicle sends only one message in a batch period. To generate an appropriate formula, the number of proxy vehicles that verify more than n messages is denoted as i. The number of cases that N_v vehicles are authenticated through N_p proxy vehicles is equal to

$$(N_p)^{N_v} - \sum_{i=1}^{\lfloor N_v/n \rfloor} \left[\sum_{x=0}^{i-1} (-1)^x P_i^x \right] \binom{N_p}{i} (N_p)^{N_v - i(n+1)}. \quad (7)$$

When $N_v \gg n$, (5) approaches $(N_p)^{N_v}$. Similarly, the number of cases that vehicles are authenticated through compromised vehicles is equal to $(N_c)^k$. Moreover, the number of cases that vehicles are authenticated through trusted proxy vehicles is equal to $(N_p - N_c)^{N_v - k}$, where k denotes the number of vehicles that are authenticated by a compromised proxy vehicle. $P\{X = k\}$ represents the probability that the verification processes of k vehicles fail. The probability distribution function conforms to the following formula:

$$P\{X = k\} \approx \frac{\binom{N_v}{k} (N_c)^k (N_p - N_c)^{N_v - k}}{(N_p)^{N_v}}. \quad (8)$$

Fig. 5 shows an example of fault tolerance that there are 100 vehicles in the coverage area of an RSU in a period, the number of proxy vehicles is 20 in this area, and each proxy vehicle can verify 30 messages simultaneously. When the number of compromised vehicles is two, where ($r = 0.1$), the probability that the verification processes of ten vehicles fail is approximately 0.132. A backup server must immediately take over these ten failed vehicles. The failure probability of 15 failed vehicles dramatically drops to 0.06. We also observe that there are more failed vehicles as r increases. Fortunately, the worst probability of this case is much lower. For instance, the probability that the verification processes of 20 vehicles fail is approximately 0.1 when $r = 0.2$.

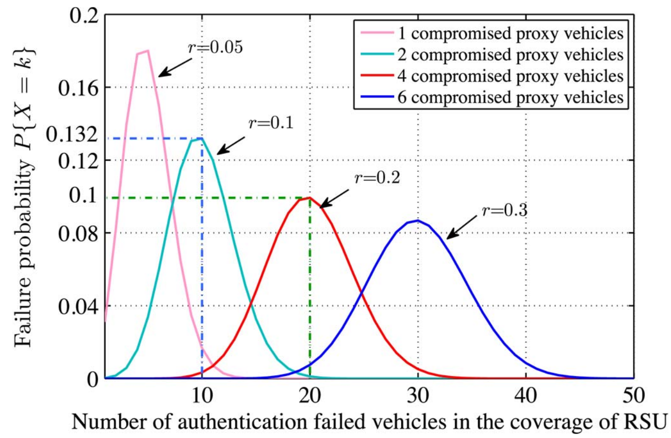


Fig. 5. Probability that the verification processes of k vehicles fail in the event that a small number of compromised proxy vehicles exist in VANETs. Each curve has a peak of lowest probability. For instance, the lowest probability of $r = 0.1$ is 0.132 when $k = 10$, which means that the authentication failure of ten vehicles is most likely to occur when there are two compromised proxy vehicles in the coverage area, where $N_v = 100$, and $N_p = 20$.

The result shows that the PBAS can achieve fault tolerance and continue its verification operation, possibly with a bit reduced performance, rather than failing completely. With the protection mechanisms of the PBAS, RSUs can detect that there are some invalid messages when some compromised proxy vehicles exist. If so, the TA will revoke the compromised proxy vehicle's certificate. This way, the number of compromised proxy vehicles is well controlled below 10%.[3] They can only make up to 20 failed vehicles, and the failure probability is low enough to be negligible.

VI. Performance Evaluation and Simulations

Here, we evaluate the performance of the proposed PBAS and compare it with the related schemes, such as TESLA++ [5], IBV [6], ABAKA [10], b-SPECS+ [11], and CPAS [12], in terms of computation and transmission overheads. It is noted that TESLA++ uses the standard signature algorithm ECDSA adopted by IEEE 1609.2 [9]. In the simulations, we used ns-2 [39] and a mobility model generation tool called VanetMobiSim [40] to estimate the average message delays and the average loss ratios of these schemes in a real environment.

A. Computation Overhead Analysis

Here, we evaluate the performance of the PBAS and the other schemes in terms of the computation overhead in an RSU. T_{mtp} denotes the time needed to perform a MapToPoint hash operation, T_{mul} denotes the time for performing one point multiplication, and T_{par} denotes the time to perform a

[3] In [10], it was even believed that the attacker can compromise, at most, 1% entities subordinated by a TA.

TABLE II
COMPARISON OF COMPUTATION OVERHEADS IN AN RSU

Scheme	Verify a single message	Verify n messages
TESLA++	$4T_{mul}$	$4T_{mul}$
IBV	$2T_{mul} + 3T_{par} + T_{mtp}$	$nT_{mul} + 3T_{par} + nT_{mtp}$
CPAS	$3T_{par} + T_{mul}$	$(n+1)T_{mul} + 3T_{par}$
b-SPECS+	$2T_{mul} + 2T_{par} + T_{mtp}$	$(2n+1)T_{mul} + 2T_{par} + nT_{mtp}$
ABAKA	$3T_{mul}$	$(2n+1)T_{mul}$
Our scheme	$4T_{mul} + 5T_{par} + T_{mtp}$	$2mT_{mul} + (2m+3)T_{par} + T_{mtp}, m = \lfloor \frac{n}{300} \rfloor$

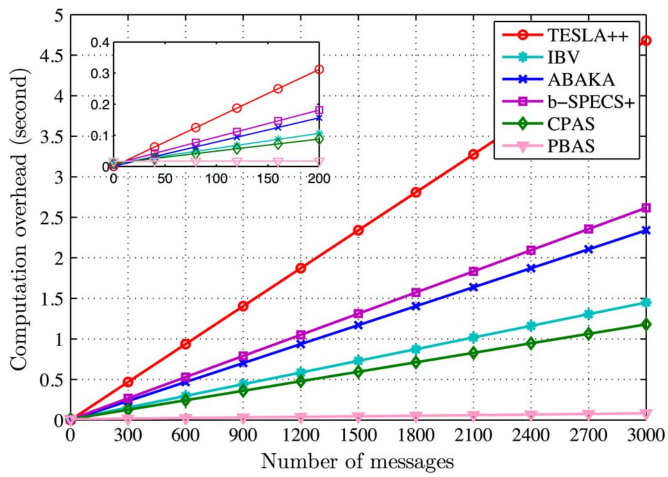


Fig. 6. Performance comparison of these schemes in terms of the computation overhead in an RSU. The computation overhead is defined as the computation time spent on verifying signatures, which are signed by 300 vehicles, and each vehicle periodically broadcasts a traffic-related message and its signature every 300 ms.

pairing operation. The experiments run on an Intel i7 3.07-GHZ machine. The computation times of the following parameters in [11], i.e., T_{mul}, T_{mtp}, and T_{par}, are 0.39, 0.09, and 3.21 ms, respectively. Therefore, we can know that the operation times of T_{mul} and T_{mtp} are, in general, much lower than T_{par}. For the other operations, such as one-way hash function calculation, the operation time is negligible because its computation time is only 0.23 μs [25]. Thus, we consider the aforementioned three parameters as the main computing costs.

Table II shows the comparison of all schemes for the computation overhead of an RSU in terms of signing a single message and n messages. TESLA++ uses the current standard signature scheme ECDSA, and the total computational time in terms of authenticating n messages is $4nT_{\mathrm{mul}}$. Since IBV, CPAS, and b-SPECS+ are used for authenticating safety-related messages, the key negotiation session is excluded in these schemes. To be fair, the computing time in ABAKA spent on key negotiation was not considered.

First, we assume that the traffic density is equal to the number of signatures in a verification period, and each vehicle periodically broadcasts a traffic-related message every 300 ms. At least m vehicles should work as the proxy vehicles to verify the messages, and a proxy vehicle can act on, at most, 300 messages. Thus, $m = \lfloor n/300 \rfloor$. In addition, we assume that the communication coverage of an RSU is 1 km^2.

Fig. 6 shows the relationship between the number of messages within an RSU's coverage area and the computation overhead of the RSU. We can see in the figure that the computation overhead increases as the number of messages increases. In addition, we can also see that the computational overhead of TESLA++ is highest when the number of messages is larger than ten. In other words, the current standard ECDSA scheme is incompatible with the dynamic traffic patterns. The PBAS is more efficient when verifying a large number of signatures: When there are more than 40 messages, the computation overhead of the PBAS in an RSU is much lower than that of the others. For instance, in 1 s, the maximum number of signatures that can be verified by the RSU is approximately 2450, 1000, 1100, and 2000 for CPAS, b-SPECS+, ABAKA, and IBV, respectively. In the PBAS, this number reaches 26 500.

B. Transmission Overhead Analysis

Next, let us analyze the transmission overhead of the PBAS when compared with IBV, ABAKA, and CPAS. The comparison is made in terms of two aspects: the transmission overhead from vehicles to RSUs and the transmission overhead from RSUs to vehicles. We exclude b-SPECS+ and TESLA++ because the transmission overhead of TESLA++ with 125-byte certificates is intolerable when the number of vehicles is relatively large, and b-SPECS+ needs large overhead in initial handshaking of the scheme. PBAS, IBV, ABAKA, and CPAS work based on identity-based cryptography, in which only a short 42-byte pseudo identity is transmitted along with an original message. The transmission overhead only considers a pseudo identity and a signature appended to the original message, whereas the message itself is not considered.

According to the analysis in Section IV, the packet size of $\{M_{\mathrm{proxy}}, ID_{\mathrm{proxy}}, \sigma^1_{\mathrm{proxy}}\}$ sent by the proxy vehicles to an RSU is $126 + 42n$ bytes, whereas the packet size of $\{Pub_{sp}, T, \sigma_{sp}\}$ sent by an RSU is 42 bytes. The packet size from vehicles to RSU (RSU to vehicles) of IBV, ABAKA, and CPAS costs 63 bytes, 84 bytes, and 101 bytes (N/A, 80 bytes, 70 bytes), respectively. Table III shows the comparison of transmission overhead.

Fig. 7 shows the relationship between the transmission overhead and the number of messages received by an RSU in 3 s. Obviously, because each signature of CPAS has three parts, a 42-byte pseudo identity and the other necessary parameters, and each part uses a 160-bit cyclic group (21 bytes), the total signature size of CPAS is 174 bytes. The transmission overhead of CPAS is largest among these schemes as the number of message increases, whereas the transmission overhead of ABAKA is smaller because each signature has only a 21-byte parameter and a 42-byte pseudo identity. Moreover, the total signature size of ABAKA is 63 bytes. The transmission overhead of IBV is the same as ABAKA, i.e., 63 bytes.

TABLE III
COMPARISON OF TRANSMISSION OVERHEAD

Scheme	A single message		n messages	
	OBU→ RSU	RSU→ OBU	OBU→ RSU	RSU→ OBU
CPAS	174 bytes	143 bytes	$174n$ bytes	$143n$ bytes
IBV	63 bytes	N/A	$63n$ bytes	N/A
ABAKA	63 bytes	80 bytes	$63n$ bytes	80 bytes
Our scheme	84 bytes	105 bytes	$126m$+$42n$ bytes	42 bytes

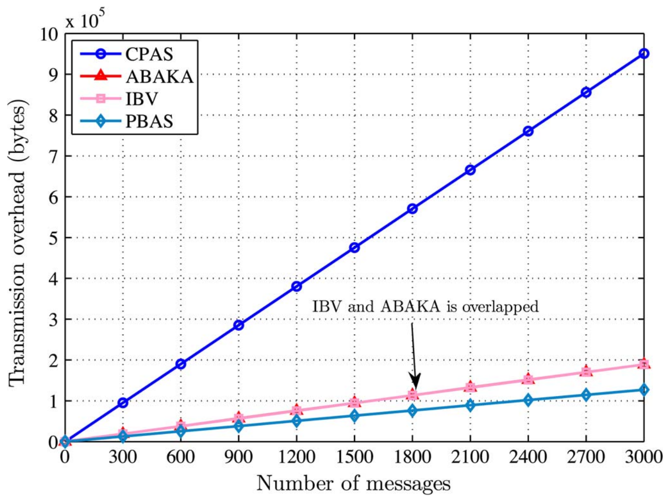


Fig. 7. Performance comparison of these schemes in terms of transmission overhead in an RSU. Transmission overhead is produced mainly by the size of signatures, which are signed by 300 vehicles, and each vehicle periodically broadcasts a traffic-related message, and its signature every 300 ms.

Through the aggregation operation, i.e., $(\sigma_1^1+\sigma_2^1+\cdots+\sigma_n^1)$ and $(\sigma_1^2 \cdot \sigma_2^2 \cdot \cdots \cdot \sigma_n^2)$, in the proxy vehicles, each proxy vehicle sends only a 126-byte packet to the RSU to authenticate a batch of messages. The transmission overhead is smallest if compared with CPAS and ABAKA.

Obviously, the figure shows that the transmission overhead linearly increases with an increasing number of messages. The transmission overhead of CPAS is largest among these schemes, and that of the PBAS is much smaller than the others. When the number of messages increases up to 1000, the PBAS saves 241 and 48 MB of bandwidth compared with CPAS and ABAKA (IBV), respectively. Here, 1000 is the number of messages sent by 300 vehicles in 1 s.

C. Simulations

To perform a more realistic performance evaluation in simulations, the mobility traces adopted in the simulations were generated using VanetMobiSim [40]. The road scenario of the mobility model for simulations is shown in Fig. 8.

The ns-2.35 [39] was used to simulate the average messages delays and the average loss ratios in RSUs to assess the performance of the PBAS. The adopted simulation parameters of DSRC are given in Table IV. The hidden terminal problem is naturally reflected in these two performance parameters in the simulation processes. In particular, the first phase of the PBAS is that vehicles broadcast messages in the area. The hidden terminal problem in the broadcast scenario is more severe than that in the second phase of the PBAS, in which the proxy vehicles communicate with the RSU. To observe and discuss

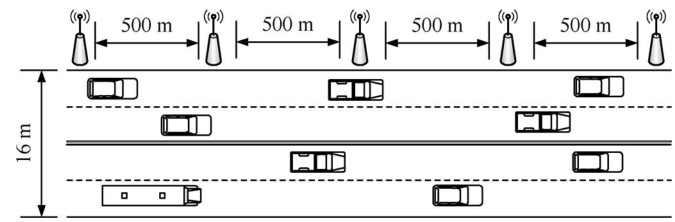


Fig. 8. Road scenario for simulations. The simulation scenario area length is 8000 m, which includes four lanes, and each lane is 2000 m long and 4 m wide. The road deploys five RSUs because the transmission range of each vehicle is only 300 m.

TABLE IV
SIMULATION PARAMETERS

Parameter	Value
Simulation area	8000 × 16 m^2
No. of traffic lane	4
No. of RSUs	5
Maximum No. of proxy vehicles	20
Simulation time	100 s
MAC protocol	802.11p
Channel bandwidth	6 Mbps
Transmission range of OBU	300 m
Transmission range of RSU	1000 m
Minimum inter-vehicle distance	40 m
Route protocol	AODV
Slot-time	13 μs
SIFS	32 μs
AIFS (high priority)	58 μs
Contention window size (CW)	15∼1023

these performance parameters, the average message delay of the PBAS is defined as the time to transmit messages from vehicles to an RSU, which can be expressed as

$$AD_{\mathrm{Msg}} = \frac{1}{N_V M_{sent\_m} \cdot RSU^n \cdot pro^n} \sum_{n=1}^{N_V} \sum_{m=1}^{M_{sent\_m}} \sum_{p=1}^{pro^n} \sum_{r=1}^{RSU^n} \times \left(T_{\mathrm{sign}}^{n\_m} + T_{\mathrm{trans}}^{n\_m\_n^{\mathrm{pro}}} + T_{\mathrm{trans}}^{n^{\mathrm{pro}}\_m\_RSU} + T_{\mathrm{verify}}^{n\_m\_n^{\mathrm{pro}}} + T_{\mathrm{verify}}^{n^{\mathrm{pro}}\_n\_RSU} \right) \quad (9)$$

in which AD_{Msg} denotes the average message delay, V denotes the sample area in the simulations, N_V denotes the number of vehicles in V, M_{sent_m} denotes the number of messages sent by vehicle n, RSU^n is the number of RSUs in the area, and pro^n denotes the number of proxy vehicles. In addition, $T_{\mathrm{sign}}^{n\_m}$ denotes the time required for vehicle n to sign message m, $T_{\mathrm{trans}}^{n\_m\_n^{\mathrm{pro}}}$ is the time that vehicle n spends in transmitting message m to the proxy vehicle n^{pro}, whereas $T_{\mathrm{trans}}^{n^{\mathrm{pro}}\_m\_RSU}$ designates the time that the proxy vehicle n^{pro} spends in transmitting message m to the RSU, $T_{\mathrm{verify}}^{n\_m\_n^{\mathrm{pro}}}$ is the time that the proxy vehicle n^{pro} authenticates message m, and $T_{\mathrm{verify}}^{n^{\mathrm{pro}}\_n\_RSU}$ denotes the time that the RSU checks the result from the proxy vehicle n^{pro}.

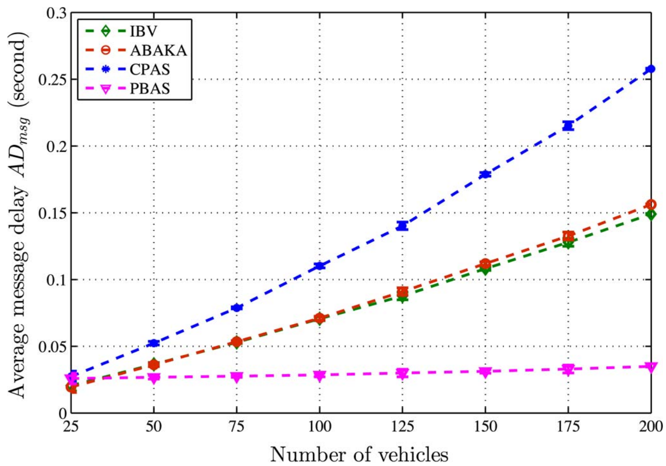


Fig. 9. Performance comparison via simulations for different authentication schemes in terms of the relationship between the average message delays in RSUs and the number of vehicles. Vehicles are evenly distributed over different lanes. The speeds of vehicles in each lane are approximately 10–30 m/s.

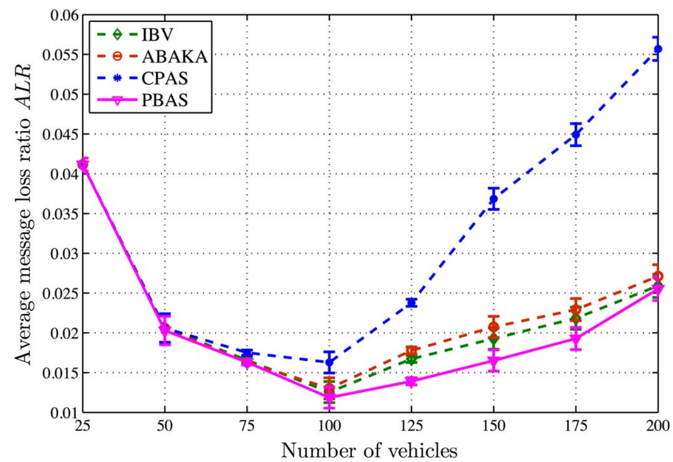


Fig. 10. Performance comparison via simulations for different authentication schemes to show the relationship between the average message loss ratios in RSUs and the number of vehicles. Vehicles are evenly distributed over different lanes. The speeds of vehicles in each lane are approximately 10–30 m/s.

The average loss ratio is defined as the ratio between the number of messages dropped and the total number of messages received in every 100 s by an RSU, which can be expressed as

$$ALR = \frac{1}{100 \cdot RSU^n} \frac{\sum_{r=1}^{RSU^n} M^n_{\text{arrived}}}{\sum_{n=1}^{N_V} M_{sent\_n}} \tag{10}$$

where M^n_{arrived} denotes the number of messages received by RSUs. We run 100 times for each simulation, which lasts 100 s to authenticate messages using the current schemes CPAS, ABAKA, IBV, or our proposed scheme PBAS, respectively. For reliability of the simulations, we set a 0.95 confidence coefficient as an observed interval.

Fig. 9 shows the first set of simulation results to reveal the relationship between the average message delays and the number of vehicles. In general, the more vehicles that appear, the larger the average message delay that appears at RSUs. The PBAS outperforms ABAKA, IBV, and CPAS, because the proxy vehicles in the PBAS reduce the number of handshakes between vehicles and RSUs. In the figure, it is seen that the average message delay of CPAS increases from 50 to 110 ms, and that of ABAKA increases from 31 to 73 ms when the vehicle density increases from 50 to 100. The PBAS performs better than CPAS and ABAKA, whose average message delay increases from 25 to 50 ms. From the simulations, we can see that the performance of the PBAS is slightly affected by vehicle density.

Fig. 10 shows the second set of simulation results to reveal the relationship between the average message loss ratios and the number of vehicles. Note that the transmission range of a vehicle is only 300 m. In an ad hoc on-demand distance vector, the relay vehicles can help the other vehicles forward messages. If a vehicular message cannot find a suitable relay within its range to forward to the destination, it will give up the messages. We can see in Fig. 10 that the message loss ratios of all the schemes decrease at the beginning of the frame when the number of vehicles increases, because the number of relays increases as the number of vehicles increases. After the number of vehicles exceeds 100 and sequentially increases, we observe that the hidden terminal problem degrades the average message loss ratios significantly due to the frequent occurrence of message collisions. In fact, in addition to collisions caused by the hidden vehicle node, frequent transmissions of the RSU synchronized with the vehicles in the same communication area may also cause collisions. Unfortunately, usually, there are more than 100 vehicles in VANETs, which cause the performance degradation of VANETs. However, the PBAS still shows its advantages if compared with ABAKA, IBV, and CPAS.

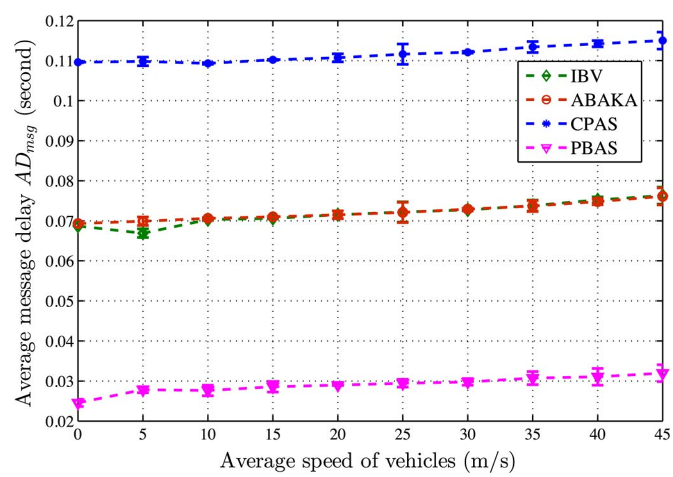


Fig. 11. Performance comparison via simulations for different authentication schemes to reveal the relationship between the average message delays in RSUs and the average speed of vehicles, where the number of vehicles is 100.

Fig. 11 shows the third set of simulations to show the relationship between the average message delays and the average speed of vehicles. In Fig. 11, we can see that the average message delay of each scheme approaches a constant, which is only slightly affected by the speed of the vehicles. However, in Fig. 12, the last simulation result shows that the message

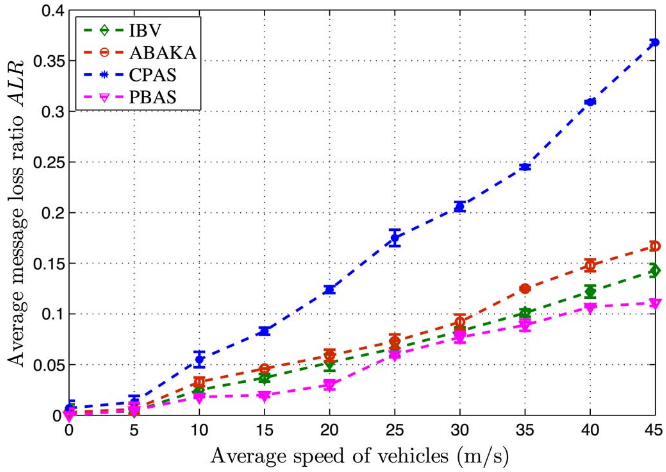


Fig. 12. Performance comparison via simulations for different authentication schemes in RSUs to show the relationship between the average message loss ratios and the average speed of vehicles, where the number of vehicles is 100.

loss ratios for all schemes increase with the increasing number of vehicles, because the transmissions have a higher probability of being interrupted when the vehicles are moving fast. On the other hand, the PBAS has the lowest message loss ratio, even when the speed increases, because the direct transmission time between RSUs and vehicles is shortest in the PBAS with the help of the proxy vehicles.

VII. Conclusion

The PBAS makes use of vehicles' computational capacity to reduce the burden of RSUs, where the proxy vehicles can authenticate multiple messages from the other vehicles. The PBAS also provides RSUs with a systematic and independent mechanism to verify the messages from the proxy vehicles. In addition, the PBAS can negotiate a session key with every other vehicle for the confidentiality of sensitive information. The evaluation model of the PBAS showed that the PBAS offers fault tolerance, which enables the scheme to continue operating properly even if a small number of proxy vehicles are compromised in VANETs. Moreover, we analyzed and compared the performance of the PBAS with the other authentication schemes in terms of their computation and transmission overheads. We also used simulations to verify the efficiency of the PBAS in realistic environments, showing that the PBAS is a promising security scheme for efficient VANET authentication.

In this paper, on the PBAS, we focused on a cryptography algorithm under the assumption that any vehicle having completed system initialization can act as a proxy vehicle. However, it is crucial to make sure that these vehicles have incentives to serve for the others under the condition of efficient message delivery. In the future, we will exploit the game theory to study the incentive mechanism. The redundant authentication is another issue, in which different proxy vehicles may work on the same message. To minimize the redundant authentication events, we should design a selection strategy that combines extra computation resource utilization optimization and redundant authentication reduction.

Acknowledgment

The authors would like to thank the anonymous reviewers for their helpful comments that helped improve this paper.

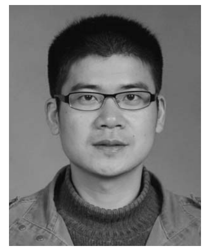
References

[1] M. Raya and J. P. Hubaux, "Securing vehicular *ad hoc* networks," *J. Comput. Security*, vol. 15, no. 1, pp. 39–68, Jan. 2007.
[2] T. W. Chim, S. M. Yiu, C. K. Hui, and O. K. Li, "VSPN: VANET-based secure and privacy-preserving navigation," *IEEE Trans. Comput.*, vol. 63, no. 2, pp. 510–524, Feb. 2014.
[3] J. T. Isaac, S. Zeadally, and J. S. Camara, "Security attacks and solutions for vehicular *ad hoc* networks," *IET Commun.*, vol. 4, no. 7, pp. 894–903, Apr. 2010.
[4] S. Zeadally, R. Hunt, Y. S. Chen, A. Irwin, and A. Hassan, "Vehicular *ad hoc* networks (VANETS): Status, results, challenges," *Telecommun. Syst.*, vol. 50, no. 4, pp. 217–241, Aug. 2012.
[5] A. Studer, F. Bai, B. Bellur, and A. Perrig, "Flexible, extensible, efficient VANET authentication," *J. Commun. Netw.*, vol. 11, no. 6, pp. 574–588, Dec. 2009.
[6] C. Zhang, X. Lin, R. Lu, P. 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.
[7] C. Zhang, R. Lu, X. Lin, P. Ho, and X. Shen, "An efficient identity-based batch verification scheme for vehicular sensor networks," in *Proc. IEEE INFOCOM*, 2008, pp. 246–250.
[8] Dedicated Short Range Commun. (DSRC). [Online]. Available: http://standards.ieee.org/develop/wg/1609_WG.html
[9] *1609.2-2013-IEEE Standard for Wireless Access in Vehicular Environments—Security Services for Applications and Management Messages*, IEEE Std. 1609, 2013.
[10] J. L. Huang, L. Y. Yeh, and H. Y. Chien, "ABAKA: 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.
[11] S. J. Horng *et al.*, "b-SPECS+: Batch verification for secure pseudonymous authentication in VANET," *IEEE Trans. Inf. Forensics Security*, vol. 8, no. 11, pp. 1860–1875, Nov. 2013.
[12] K. A. Shim, "CPAS: An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks," *IEEE Trans. Veh. Technol.*, vol. 61, no. 4, pp. 1874–1883, May 2012.
[13] X. Li and L. Wang, "A rapid certification protocol from bilinear pairing for vehicular *ad hoc* networks," in *Proc. IEEE Conf. TrustCOM*, 2012, pp. 890–895.
[14] D. Johnson and A. Menezes, "The elliptic curve digital signature algorithm (ECDSA)," *Int. J. Inf. Security*, vol. 1, no. 1, pp. 36–63, Aug. 2001.
[15] J. Petit, "Analysis of ECDSA authentication processing in VANETs," in *Proc. IEEE Conf. NTMS*, 2009, vol. 1, pp. 1–5.
[16] A. Perring, R. Canetti, D. Song, and D. Tygar, "Efficient and secure source authentication for multicast," in *Proc. NDSS*, 2001, pp. 35–46.
[17] D. Boneh and M. Franklin, *Identity-Based Encryption from the Weil Pairing*. New York, NY, USA: Springer-Verlag, 2001, pp. 213–229.
[18] H. Yoon, J. H. Cheon, and Y. Kim, "Batch verification with ID-based signatures," in *Proc. Inform. Security Cryptology*, 2004, pp. 233–248.
[19] C. Zhang, P. Ho, and J. Tapolcai, "On batch verification with group testing for vehicular communications," *Wireless Netw.*, vol. 17, no. 8, pp. 1851–1865, Nov. 2011.
[20] T. W. Chim, S. M. Yiu, C. K. Hui, and O. K. Li, "SPECS: Secure and privacy enhancing communications schemes for VANETs," *Ad Hoc Netw.*, vol. 12, pp. 189–203, 2011.
[21] A. Wasef, R. Lu, X. Lin, and X. Shen, "Complementing public key infrastructure to secure vehicular *ad hoc* networks [Security and Privacy in Emerging Wireless Networks]," *IEEE Wireless Commun.*, vol. 17, no. 5, pp. 22–28, Oct. 2010.
[22] A. Boldyreva, V. Goyal, and V. Kumar, "Identity-based encryption with efficient revocation," in *Proc. ACM Conf. Comput. Commun. Security*, 2008, pp. 417–426.
[23] A. Wasef and X. Shen, "EDR: Efficient decentralized revocation protocol for vehicular *ad hoc* networks," *IEEE Trans. Veh. Technol.*, vol. 58, no. 9, pp. 5214–5224, Nov. 2009.
[24] A. Wasef, Y. Jiang, and X. Shen, "DCS: An efficient distributed-certificate-service scheme for vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 59, no. 2, pp. 533–549, Feb. 2010.

[25] 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.
[26] R. Lu, X. Lin, and H. Luan, "Pseudonym changing at social spots: An effective strategy for location privacy in VANETs," *IEEE Trans. Veh. Technol.*, vol. 61, no. 1, pp. 86–96, Jan. 2012.
[27] 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.
[28] J. Choi and S. Jung, "A security framework with strong non-repudiation and privacy in VANETs," in *Proc. IEEE Conf. CCNC*, 2009, pp. 1–5.
[29] R. Lu, X. Lin, Z. Shi, and X. Shen, "A lightweight conditional privacy-preservation protocol for vehicular traffic-monitoring systems," *IEEE Intell. Syst.*, vol. 28, no. 3, pp. 62–65, May/Jun. 2013.
[30] X. Zhu, S. Jiang, L. Wang, and H. Li, "Efficient privacy-preserving authentication for vehicular *ad hoc* networks," *IEEE Trans. Veh. Technol.*, vol. 63, no. 2, pp. 907–919, Feb. 2014.
[31] Y. Qian, K. Lu, and N. Moayeri, "A secure VANET MAC protocol for DSRC applications," in *Proc. IEEE GLOBECOM*, 2008, pp. 1–5.
[32] G. Samara, W. A. H. A. Salihy, and R. Sures, "Security analysis of vehicular *ad hoc* networks (VANET)," in *Proc. IEEE Conf. NETAPPS*, 2010, pp. 55–60.
[33] P. Papadimitratos and J. P. Hubaux, "Report on the "secure vehicular communications: Results and challenges ahead" workshop," *ACM SIGMOBILE Mobile Comput. Commun. Rev.*, vol. 12, no. 2, 2008.
[34] M. Raya, P. Papadimitratos, V. D. Gligory, and J. P. Hubaux, "On data-centric trust establishment in ephemeral *ad hoc* networks," in *Proc. IEEE Annu. Joint Conf. Comput. Commun. Soc.*, 2008, pp. 1238–1246.
[35] S. Dietzel, J. Petit, G. Heijenk, and F. Kargl, "Graph-based metrics for insider attack detection in VANET multihop data dissemination protocols," *IEEE Trans. Veh. Technol.*, vol. 62, no. 4, pp. 1505–1518, May 2013.
[36] R. G. Engoulou, M. Bellaiche, S. Pierre, and A. Quintero, "VANET security surveys," *Comput. Commun.*, vol. 44, pp. 1–3, May 2014.
[37] Y. Qian and N. Moayeri, "Design of secure and application-oriented VANETs," in *Proc. IEEE VTC Spring 2008*, 2008, pp. 2794–2799.
[38] X. Lin and X. Li, "Achieving efficient cooperative message authentication in vehicular *ad hoc* networks," *IEEE Trans. Veh. Technol.*, vol. 62, pp. 3339–3348, Sep. 2013.
[39] The Network Simulator NS-2. [Online]. Available: http://www.isi.edu/nsnam/ns
[40] VanetMobiSim Project Home Page. [Online]. Available: http://vanet.eurecom.fr

Yiliang Liu received the B.E. degree in computer science and communication engineering from Jiangsu University, Zhenjiang, China, in 2012, where he is currently working toward the M.E. degree in computer science and communication engineering.

Since January 2014, he has been a visiting research student with the Department of Engineering Science, National Cheng Kung University, Tainan, Taiwan. His research interests include security for wireless networks and vehicular ad hoc networks.



Liangmin Wang (M'11) 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.

From 2008 to 2013, he was an Associate Professor with the School of Computer Science and Communication Engineering, Jiangsu University, Zhenjiang, China. From 2009 to 2010, he was a Visiting Scholar with Prof. A. Kot with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Currently, he is a specially engaged Professor as a "Wanjiang Scholar" with the School of Computer Science and Technology, Ahui University, Hefei, China. His research interests include Internet of Things and wireless security.

Dr. Wang is a Senior Member of the Chinese Computer Federation and a member of the Chinese Cryptology Federation. He has been a member of the Association for Computing Machinery since 2009.

Hsiao-Hwa Chen (F'10) received the B.Sc. and M.Sc. degrees from Zhejiang University, Hangzhou, China, in 1982 and 1985, respectively, and the Ph.D. degree from the University of Oulu, Oulu, Finland, in 1991.

He is currently a Distinguished Professor with the Department of Engineering Science, National Cheng Kung University, Tainan, Taiwan.

Dr. Chen is the founding Editor-in-Chief of Wiley's *Security and Communication Networks Journal* (http://www.interscience.wiley.com/security). He is also serving as the Editor-in-Chief for the IEEE WIRELESS COMMUNICATIONS. He is a Fellow of the Institution of Engineering and Technology and an elected Member-at-Large of the IEEE Commmunications Society.