Wireless Netw (2013) 19:1441–1449
DOI 10.1007/s11276-013-0543-7

Toward a secure batch verification with group testing for VANET

Cheng-Chi Lee · Yan-Ming Lai

Published online: 20 January 2013

Abstract Vehicular Ad-Hoc Network (VANET) is an application of Ad-Hoc Network, which can significantly improve the efficiency of transportation systems. The authentication of information is particularly important in the VANET system, because of its significant impact, and the transportation systems may be paralyzed as a result of receiving the wrong traffic information. Hence, a lot of schemes have been proposed to verify the information of VANET. However, most of currently known schemes verify the information on a one by one basis. In real situation, the large amount of traffic flow will generate a lot of information at the same time. If the authentication method is authenticating one by one, it is bound to lead to information delays, and the system will have difficulty to achieve real-time performance. Therefore, we shall propose an improved authentication of the batch scheme based on bilinear pairing to make VANET more secure, efficient, and more suitable for practical use.

Keywords Ad-Hoc · VANET · Anonymous · Batch authentication · Bilinear pairing

1 Introduction

Ad-Hoc Network is a representative of today's advanced wireless application. It has some advantages, such as having fewer infrastructures, arranging a Local Area Network (LAN) quickly, and allowing its members to join and leave easily. Because of these reasons, Ad-Hoc Network has become the first of choice network model for a real-time LAN. This network model is especially suitable for an environment that changes frequently or that does not have enough infrastructure, e.g. in a disaster area or a transportation system [1–3].

Vehicular Ad-Hoc Network (VANET) is an application of Ad-Hoc Network for vehicle communication. Each vehicle uses a device, called on-board units (OBUs), to communicate with one another. The same device can also be used to communicate with the roadside unit (RSU) or other infrastructures [4–6]. To support this, there are two types of VANET: Vehicle-to-Vehicle (V2V) communication and Vehicle to RSU (V2R) communication [4, 6–12]. With the help of V2V, people can obtain more information and use the information to achieve road safety, such as maintaining a distance from other vehicles. Furthermore, a group can establish simple communication networks and allow members to communicate with one another. People can also communicate with RSU by V2R to download files from the Internet or inquire neighborhood location information, such as the closest gas station and restaurant. In addition, users can query RSU about the local situation to avoid traffic jams. Because RSU is an infrastructure, it can be an Internet node. Hence, people can use Internet services to upload or download files through RSU. On the other hand, traffic management could be done easily by combining the traffic system and the VANET system. Because RSU can collect and monitor traffic flow information, the traffic system can predict the traffic flow and control traffic signals to regulate the flow in real time. If necessary, traffic system can also cooperate with the public affair vehicles, such as ambulances or fire engines, to improve the efficiency of performing any urgent task.

C.-C. Lee (✉) · Y.-M. Lai
Department of Library and Information Science, Fu Jen Catholic University, 510 Jhongjheng Rd., Sinjhuang City, Taipei County 24205, Taiwan, ROC
e-mail: cclee@mail.fju.edu.tw

The security issues in VANET are particularly important, because VANET provides people with many applications and traffic experience for their daily life. The applications of VANET are in general grouped into two categories: safety and non-safety applications [11, 13–15]. The non-safety applications are usually related to local information and traffic information. One of VANET's security challenges is avoiding wrong messages, such as falsified messages, replayed messages, or malicious messages. The wrong messages maybe cause some poor situations such as the following:

1. Wrong traffic flow messages: The wrong traffic flow message may result in the traffic management system making wrong decisions. The wrong decision will cause the traffic lights of the heavy side to stay red and the other side to stay green.
2. Wrong traffic stat messages: The wrong traffic stat message may mislead driver into a traffic jam, and the traffic will be more heavier.
3. Wrong vehicles messages: The wrong vehicles message may make the driver misread the safe distance, and crash into other vehicles.
4. Falsified messages: If an adversary falsifies a public affair vehicle signal, such as an ambulance's signal, he/she may compel the traffic light to cooperate with him/her and harm the driving right of other drivers.

Because VANET improves the traffic experience substantially, any secure leak of VANET may cause inestimable harms to the traffic system. To ensure both the integrity of the messages and non-repudiation is indispensable. A simple solution is to sign each message with a digital signature before the message is sent. In 1976, Diffie and Hellman proposed an idea about public-key cryptography [16]. Two years later, Rivest et al. [17] proposed a novel scheme to accomplish Diffie–Hellman's idea, called RSA algorithm. In 2007, Raya and Hubaux [9] proposed appropriate security architecture for VANET. There is a PKI (Public Key Infrastructure) certificate issues in their scheme. The RSU and the OBU can mutually authenticate by means of the other's public key and establish a session key for communication. However, most of the traditional signature schemes verify the received signatures one by one. When the traffic is heavy, the verifier will receive a lot of signatures. Verifying a large number of signatures sequentially will take a long time, and the information with the signature will be delayed. Because the traffic situations are always changing, real time response is a very important issue for the traffic information [7, 8, 18]. If the information is obsolete, it cannot explain the real traffic situation and help people or traffic management system make decisions, and the information will lose its value [4, 6, 11, 12].

To solve the verification bottleneck problem, a lot of related schemes have been proposed. In 1990, Fiat proposed the first batch cryptography scheme based on RSA [19]. In 2007, Lin et al. [18] proposed a group signature scheme based on bilinear pairing to improve the authentication efficiency. Because the verifier can verify multiple signatures simultaneously in Lin et al.'s scheme, the cost of computation time will not grow linearly with the amount of the signature. Unfortunately, Lin et al.'s scheme uses a lot of exponent operations, and it has complex computing process. In 2011, Zhang et al. [11] and Huang et al. [20] proposed a new scheme respectively. Both of their schemes are based on bilinear pairing and use addition operations to batch verify multiple signatures simultaneously. As an addition operation is simpler than any exponent operations, both of the two schemes are more efficient. Hence, batch verifying is more efficient than single verifying when the verifier has to verify a large number of signatures.

However, Zhang et al.'s scheme has some weaknesses. First, Zhang et al.'s scheme is vulnerable to the replaying-attack. Because of this weakness, an adversary can simulate a fake situation, such as a traffic jam, by collecting the vehicle messages and signatures in the corresponding situation and replaying them. Second, Zhang et al.'s scheme doesn't achieve the signature non-repudiation. A malicious driver can broadcast wrong information to mislead other drivers and repudiate the behavior when the traffic manager traces him/her by his/her signature. In Huang et al.'s scheme, which is known as ABAKA, the scheme also doesn't achieve the signature non-repudiation. Wang and Zhang [21] pointed out this weakness in 2012. Hence, ABAKA is not suitable for VANET. The details of ABAKA can refer to [20]. For this reason, we want to propose an improved scheme to enhance the security and keep the efficiency of Zhang et al.'s scheme. The improved scheme can make the VANET information verification be more suitable.

In this paper, we will describe the weaknesses of Zhang et al.'s scheme, and propose an improved scheme. The paper is organized as follows. In Sect. 2, we present the background and preliminaries, which includes the network model and equipment, security requirements, and the bilinear maps. After that, we describe the Zhang et al.'s scheme in Sect. 3 and provide our analysis in Sect. 4. In Sect. 5, we will propose an improved scheme and present an analysis of the proposed scheme in Sect. 6. Finally, we conclude the paper in Sect. 7.

2 Background and preliminaries

2.1 Network model and equipment

A two-layer vehicular network model was introduced in some recent articles [11, 20, 22]. The top layer consists of a trust authority (TA) and application servers. We assume

that TA can be completely trusted, and it is responsible for pre-assigning secure information for each vehicle. Most of the time, TA is off-line with other vehicles, and responsible for tracing the real identity of vehicles in case that disputes happens. The application servers for non-safety applications, such as traffic management center, communicate with RSUs and provide services or information. In the lower layer, vehicles and RSUs can communicate with one another based on DSRC protocol [23]. Each vehicle has its own public and private key-pairs for signing each message before the message is sent. Messages and signatures will be sent to the sender's neighboring RSU, and the RSUs will verify the digital signatures after receiving those information. Each vehicle has to be equipped with a tamper-proof device, which is a secure storage for secrets. We assume that the tamper-proof device is always credible and its information is never been disclosed. The device will preload some secure values, such as real identity of vehicle and secret key of system. The computing process of vehicle is also included in this device and the value is never disclosed.

2.2 Security requirements

Communication security is crucial to protect the privacy of the users. In VANET communication, security issues are also very important. In this field, we can generalize three security requirements as follows [12, 22].

1. Message authentication: Ensuring that a message was sent from a legitimate user and the integrity of message wasn't broken is a primary issue.
2. User privacy preserving: In VANET, communications are always transmitted via a wireless network. Compared to a wire network, wireless is easier intercepted, overheard, and traced. The system has to protect the privacy of a legitimate user, including the user's real identity or other individual information.
3. Audit-ability: To avoid the inside user using the user privacy preserving to broadcast malicious message which maybe mislead other legitimate user, systems should have a mechanism for retrieving the real identity of a malicious user.

2.3 Bilinear maps

Our proposed scheme in this paper is based on bilinear pairing, which is briefly introduced in this section [11, 18, 24, 25].

- Let G be a cyclic additive group generated by P, and G_T be a cyclic multiplicative group. G and G_T have the same prime order q, that means $|G| = |G_T|$.
- Let $\hat{e} : G \times G \rightarrow G_T$ be a bilinear map.

The bilinear map satisfies the following properties:

1. Bilinear: For all $P, Q, R \in G$, $\hat{e}(Q, P + R) = \hat{e}(P + R, Q) = \hat{e}(Q, P) \cdot \hat{e}(Q, R)$. Let $a, b \in Z^*q, \hat{e}(aQ, bP) = \hat{e}(bQ, aP) = \hat{e}(Q, P)^{ab}$.
2. Non-degenerate: There exist $P, Q \in G$ such that $\hat{e}(P, Q) \neq 1_{GT}$. where 1_{G_T} is the identity element of G_T.
3. Computable: There is an efficient algorithm to compute $\hat{e}(Q, P)$ for any $P, Q \in G$.

The bilinear map can be constructed by the modification on elliptic curves [18, 24, 25]. It also possesses the characteristic of elliptic curves as follows. Let $P, Q \in G$ and $a \in Z^*q$, $Q = aP$, and $\{P, Q\}$ are known. Finding the integer a from Q and P is the elliptic curve discrete logarithm problem (ECDLP).

3 Review of Zhang et al.'s scheme

There are three subsections in Zhang et al.'s scheme [11], including (a) key generation and pre-distribution, (b) pseudo identity generation and message signing, and (c) message verification. The notation is shown in Table 1. We briefly describe them as follows.

3.1 Key generation and pre-distribution

In Zhang et al.'s scheme, TA is responsible for setting up the system parameters for each vehicle and RSU as follows.

Table 1 Notation of this paper

V_i	The ith vehicle
RSU	A roadside unit
TA	A trust authority
TPD	A tamper-proof device
s_1, s_2	The private master key of the system
P_{pub1}, P_{pub2}	The public key of the TA
RID_i	The real identity of $V_i, RID_i \in G$
PWD_i	A password of V_i
ID^i	A pseudo identity of the vehicle V_i, $ID^i = (ID^i_1, ID^i_2)$
SK^i	A private key of the vehicle V_i, $SK^i = (SK^i_1, SK^i_2)$
M_i	A message sent by the vehicle V_i
$h()$, $h_2()$	A one-way hash function
$H()$	A map to point hash function, H: $\{0, 1\}^* \rightarrow G$
$\|\|$	Message concatenation operation
T_i	A timestamp generated by V_i
Vec_i	A vector used to distinguish signatures, $i = 1, 2, \ldots n$

1. Let G be a cyclic additive group generated by P, and G_T be a cyclic multiplicative group and G and G_T have the same prime order q. After that, let $\hat{e} : G \times G \rightarrow G_T$ be a bilinear map.
2. Choose two random numbers $\{s_1, s_2\} \in Z^*q$ as its two master keys, and compute $P_{pub1} = s_1P, P_{pub2} = s_2P$ as its public keys. These two master keys $\{s_1, s_2\}$ of the TA are pre-loaded in each vehicle's tamper-proof device.
3. The public parameters $\{G, G_T, q, P, P_{pub1}, P_{pub2}\}$ are pre-loaded in each RSU and vehicle.
4. Each vehicle is assigned its real identity, denoted as $RID \in G$, and password, denoted as PWD. Both RID and PWD are stored in the tamper-proof device.

3.2 Pseudo identity generation and message signing

To achieve user anonymity, each vehicle has to generate a pseudonym before commutation. The details of this phase are shown as follows.

1. The vehicle V_i inputs its unique real identity RID_i and the password PWD_i to initiate a pseudo identity generation process.
2. After verifying RID_i and PWD_i, TPD chooses a random number r and computes pseudo $ID^i = \{ID_1^i, ID_2^i\}$ and $SK^i = \{SK_1^i, SK_2^i\}$.

$$
\begin{aligned}
ID_1^i &= rP \\
ID_2^i &= RID_i \oplus H(rP_{pub1}) \\
SK_1^i &= s_1 ID_1^i \\
SK_2^i &= s_2 H(ID_1^i \parallel ID_2^i)
\end{aligned}
$$

3. After that, TPD outputs ID^i and SK^i, and V_i can sign messages by using those values.
4. Each message M_i has to be signed before sent. V_i signs M_i as $\sigma_i = SK_1^i + h(M_i)SK_2^i$. Subsequently, V_i sends the final message $\{ID^i, M_i, \sigma_i\}$ to its neighboring RSU.

If V_i broadcasts a malicious message, TA can trace the RID_i of V_i by computing $RID_i = ID_2^i \oplus H(s_1 ID_1^i)$. Therefore, once a signature is in dispute, the TA has the tracing ability to find the RID of vehicle from the disputed message.

3.3 Message verification

The message verification process of Zhang et al.'s scheme has two versions: single message verification and batch message verification. We briefly describe them as follows.

3.3.1 Single message verification

When each RSU receives any final message, such as $\{ID^i, M_i, \sigma_i\}$ from a vehicle, it will verify the message's validity. If $\hat{e}(\sigma_i, P) = \hat{e}(ID_1^i, P_{pub1}) \cdot \hat{e}(h(M_i)H(ID_1^i \parallel ID_2^i), P_{pub2})$, the message is legal and unaltered. The proof is shown as follows.

$$
\begin{aligned}
\hat{e}(\sigma_i, P) &= \hat{e}(SK_1^i + h(M_i)SK_2^i, P) \\
&= \hat{e}(SK_1^i, P) \cdot \hat{e}(h(M_i)SK_2^i, P) \\
&= \hat{e}(s_1 ID_1^i, P) \cdot \hat{e}(h(M_i)s_2 H(ID_1^i \parallel ID_2^i), P) \\
&= \hat{e}(ID_1^i, s_1 P) \cdot \hat{e}(h(M_i)H(ID_1^i \parallel ID_2^i), s_2 P) \\
&= \hat{e}(ID_1^i, P_{pub1}) \cdot \hat{e}(h(M_i)H(ID_i^1 \parallel ID_2^i), P_{pub2})
\end{aligned}
$$

3.3.2 Batch message verification

If a RSU receives a number of large messages, denoted as $\{ID^1, M_1, \sigma_1\}$, $\{ID^2, M_2, \sigma_2\}$, $\{ID^3, M_3, \sigma_3\} \ldots \{ID^n, M_n, \sigma_n\}$, in a short span, the RSU can verify the messages' validity simultaneously by means of batch message verification. If $\hat{e}\left(\sum_{i=1}^{n} \sigma_i, P\right) = \hat{e}\left(\sum_{i=1}^{n} ID_1^i, P_{pub1}\right) \cdot \hat{e}\left(\sum_{i=1}^{n} (h(M_i)H(ID_1^i \parallel ID_2^i), P_{pub2}\right)$, the batch of messages is legal and unaltered. The proof of this equation is as follows:

$$
\begin{aligned}
&\hat{e}\left(\sum_{i=1}^{n} \sigma_i, P\right) \\
&= \hat{e}\left(\sum_{i=1}^{n} (SK_i^1 + h(M_i)SK_i^2), P\right) \\
&= \hat{e}\left(\sum_{i=1}^{n} (SK_i^1), P\right) \cdot \hat{e}\left(\sum_{i=1}^{n} (h(M_i)SK_i^2), P\right) \\
&= \hat{e}\left(\sum_{i=1}^{n} (SK_i^1), P\right) \cdot \hat{e}\left(\sum_{i=1}^{n} (h(M_i)s_2 H(ID_1^i \parallel ID_2^i)), P\right) \\
&= \hat{e}\left(\sum_{i=1}^{n} s_1 ID_1^i, P\right) \cdot \hat{e}\left(\sum_{i=1}^{n} (h(M_i)H(ID_1^i \parallel ID_2^i)), s_2 P\right) \\
&= \hat{e}\left(\sum_{i=1}^{n} ID_1^i, s_1 P\right) \cdot \hat{e}\left(\sum_{i=1}^{n} (h(M_i)H(ID_1^i \parallel ID_2^i)), P_{pub2}\right) \\
&= \hat{e}\left(\sum_{i=1}^{n} ID_1^i, P_{pub1}\right) \cdot \hat{e}\left(\sum_{i=1}^{n} (h(M_i)H(ID_1^i \parallel ID_2^i)), P_{pub2}\right)
\end{aligned}
$$

4 Cryptanalysis of Zhang et al.'s scheme

Zhang et al. proposed an efficient batch message verification to solve the verification bottleneck problem. However, the Zhang et al. scheme has two weaknesses, i.e. (a) it is

vulnerable to the replaying attack and (b) it fails to achieve non-repudiation. The details of the two weaknesses of Zhang et al.'s scheme are shown as follows.

4.1 Replaying attack

Zhang et al.'s scheme is vulnerable to the replaying attack. We assume an adversary can intercept a public affair vehicle message and signature. He/she can replay the information to mislead the traffic management system when he/she needs. On the other situation, an adversary can intercept a lot of signatures from different vehicles when those vehicles are in a traffic jam, and replay those signatures to invent a fake traffic jam and mislead other vehicles in order to avoid the jammed sections.

4.2 Not achieving non-repudiation

Zhang et al.'s batch message verification is very efficient. However, the batch verification scheme has a leak, which allows the malicious user to deny his/her signatures. Assume a malicious user generates several different messages and signatures, such as $\{ID^1, M_1, \sigma_1\}$, $\{ID^2, M_2, \sigma_2\}$, $\{ID^3, M_3, \sigma_3\}$, and swaps their contents to become $\{ID^1, M_1, \sigma_3\}$, $\{ID^2, \mathrm{M}_2, \sigma_1\}$, $\{ID^3, M_3, \sigma_2\}$. After that, the malicious user sent those changed messages and signatures to its neighboring RSU. If the RSU uses a batch message verification process to verify those signatures, it will consider that those changed messages and signatures are legal. The proof is shown as follows.

$$\begin{aligned}\hat{e}\left(\sum_{i=1}^{3}\sigma_i, P\right) &= \hat{e}(\sigma_1+\sigma_2+\sigma_3, P)\\ &= \hat{e}(\sigma_3+\sigma_1+\sigma_2, P)\\ &= \hat{e}\left(\sum_{i=1}^{3} ID_1^i, P_{pub1}\right)\\ &\quad\cdot \hat{e}\left(\sum_{i=1}^{3} h(M_i)H(ID_1^i \parallel ID_2^i), P_{pub2}\right)\end{aligned}$$

Although the orders of those signatures have been changed, their sum remains the same. However, those messages and signatures cannot match each other obviously. These signatures can't be passed if the RSU uses single message verification process to verify them one by one. For this reason, the malicious user can deny his/her signatures.

5 The proposed scheme

To overcome those weaknesses of Zhang et al.'s scheme, we propose an improved scheme. In our scheme, we extend the framework of Zhang et al.'s scheme. We also use a two-layer vehicular network model, and we require each vehicle to have a tamper-proof device. The notation of our scheme is also shown in Table 1.

Our scheme also includes key generation and pre-distribution, pseudo identity generation and message signing, and message verification. The differences between Zhang et al.'s scheme and our scheme are pseudo identity generation and message signing, and message verification. We explain them as follows.

5.1 Key generation and pre-distribution

In our scheme, *TA* is also responsible for setting up the system parameters for each vehicle and RSU. The process of this phase is the same as Zhang et al.'s scheme, and the difference is easily discerned in the subsequent subsections.

5.2 Pseudo identity generation and message signing

To achieve user anonymity, each vehicle has to generate a pseudonym before commutation. In this subsection, we add a timestamp T_i to overcome the replaying attack and use a one-way hash function $h_2()$ instead of the map to point function $H()$. The details of this phase are shown as follows.

1. The vehicle V_i inputs its unique real identity RID_i and the password PWD_i to initiate pseudo identity generation process.
2. After verifying RID_i and PWD_i, *TPD* chooses a random number r, sets a current timestamp T_i, and computes pseudo $ID^i = \{ID_1^i, ID_2^i\}$ and $SK^i = \{SK_1^i, SK_2^i\}$.
$$\begin{aligned}ID_1^i &= rP\\ ID_2^i &= RID_i \oplus H(rP_{pub1})\\ SK_1^i &= s_1 ID_1^i\\ SK_2^i &= s_2 h_2(ID_1^i \parallel ID_2^i \parallel T_i)P\end{aligned}$$
3. After that, *TPD* outputs ID^i and SK^i, and V_i can sign messages using ID^i and SK^i.
4. Each message M_i has to be signed before sent. V_i signs M_i as $\sigma_i = SK_1^i + h(M_i)SK_2^i$. Subsequently, V_i sends the final message $\{ID^i, M_i, \sigma_i, T_i\}$ to its neighboring *RSU*.

If V_i broadcasts a malicious message, *TA* can trace the RID_i of V_i by computing $RID_i = ID_2^i \oplus H(s_1 ID_1^i)$. Therefore, once a signature is in dispute, the *TA* has the tracing ability to find the *RID* of vehicle from the disputed message.

5.3 Message verification

When each RSU receives any final message, such as $\{ID^i, M_i, \sigma_i, T_i\}$ from a vehicle, it will check the message's T_i. If $T_r - T_i < T_\Delta$, RSU continues the verification process, or else rejects the final message. T_r denotes the received-time of the message and T_Δ denotes the predefined endurable transmission delay. The message verification process of our scheme also has two versions: single message verification and batch message verification. The details of these two versions are described as follows.

5.3.1 Single message verification

If the RSU just receives a few final messages in a span, it can verify the message's validity one by one. For each signature, if $\hat{e}(\sigma_i, P) = \hat{e}(ID_1^i, P_{pub1}) \cdot \hat{e}(h(M_i)h_2(ID_1^i \parallel ID_2^i \parallel T_i)P, P_{pub2})$, the message is legal and unaltered. The proof of this equation is as follows.

$$\begin{aligned}\hat{e}(\sigma_i, P) &= \hat{e}(SK_1^i + h(M_i)SK_2^i, P)\\ &= \hat{e}(SK_1^i, P) \cdot \hat{e}(h(M_i)SK_2^i, P)\\ &= \hat{e}(s_1 ID_1^i, P) \cdot \hat{e}(h(M_i)s_2 h_2(ID_1^i \parallel ID_2^i \parallel T_i)P, P)\\ &= \hat{e}(ID_1^i, s_1 P) \cdot \hat{e}(h(M_i)h_2(ID_1^i \parallel ID_2^i \parallel T_i)P, s_2 P)\\ &= \hat{e}(ID_1^i, P_{pub1}) \cdot \hat{e}(h(M_i)h_2(ID_1^i \parallel ID_2^i \parallel T_i)P, P_{pub2})\end{aligned}$$

5.3.2 Batch message verification

If a RSU receives a number of messages, denoted as $\{ID^1, M_1, \sigma_1, T_1\}$, $\{ID^2, M_2, \sigma_2, T_2\}$, $\{ID^3, M_3, \sigma_3, T_3\}\ldots\{ID^n, M_n, \sigma_n, T_n\}$, within a short span, the RSU can verify the messages' validity simultaneously by batch message verification. In this subsection, we add a vector parameter Vec_i to overcome the weaknesses of Zhang et al.'s scheme. Before batch message verification, the RSU distributes Vec_i to each message and signature. The Vec_i's value is a random number and ranges between 1 and x, where x is a small value and doesn't make the overhead of computation. After that, the RSU starts the batch message verification. If

$$\hat{e}\left(\sum_{i=1}^{n} Vec_i\sigma_i, P\right) = \hat{e}\left(\sum_{i=1}^{n} Vec_i ID_1^i, P_{pub1}\right) \cdot \hat{e}\left(\left(\sum_{i=1}^{n} Vec_i h(M_i)h_2(ID_1^i \parallel ID_2^i \parallel T_i)\right)P, P_{pub2}\right),$$

the batch of messages are legal and unaltered. The proof of this equation is as follows.

$$\begin{aligned}&\hat{e}\left(\sum_{i=1}^{n} Vec_i\sigma_i, P\right)\\ &= \hat{e}\left(\sum_{i=1}^{n} Vec_i(SK_1^i + h(M_i)SK_2^i), P\right)\\ &= \hat{e}\left(\sum_{i=1}^{n} Vec_i SK_1^i, P\right) \cdot \hat{e}\left(\sum_{i=1}^{n} Vec_i\, h(M_i)SK_2^i, P\right)\\ &= \hat{e}\left(\sum_{i=1}^{n} Vec_i\, s_1 ID_1^i, P\right)\\ &\quad \cdot \hat{e}\left(\sum_{i=1}^{n} Vec_i\, h(M_i)s_2 h_2(ID_1^i \parallel ID_2^i \parallel T_i)P, P\right)\\ &= \hat{e}\left(\sum_{i=1}^{n} Vec_i\, s_1 ID_1^i, P\right)\\ &\quad \cdot \hat{e}\left(\left(\sum_{i=1}^{n} Vec_i\, h(M_i)h_2(ID_1^i \parallel ID_2^i \parallel T_i)\right)s_2 P, P\right)\\ &= \hat{e}\left(\sum_{i=1}^{n} Vec_i\, ID_1^i, s_1 P\right)\\ &\quad \cdot \hat{e}\left(\left(\sum_{i=1}^{n} Vec_i\, h(M_i)h_2(ID_1^i \parallel ID_2^i \parallel T_i)\right)P, s_2 P\right)\\ &= \hat{e}\left(\sum_{i=1}^{n} Vec_i\, ID_1^i, P_{pub1}\right)\\ &\quad \cdot \hat{e}\left(\left(\sum_{i=1}^{n} Vec_i\, h(M_i)h_2(ID_1^i \parallel ID_2^i \parallel T_i)\right)P, P_{pub2}\right)\end{aligned}$$

6 Analysis of our scheme

6.1 Security analysis

In this section, we analyze the security of the proposed batch verification scheme in terms of the security requirements, which includes message authentication, user privacy preserving, and audit-ability, as follows.

(1) Message authentication:

The message authentication is the most basic security requirement to ensure the legality of a message's source and the integrity of a message in any communication. In our scheme, σ_i not only uses a one-way hash function to pack the message M_i, but also uses a current timestamp T_i to generate SK_2^i in order to resist the replaying attack and ensures that the signature σ_i is fresh. Our scheme also inherits the advantage of Zhang et al.'s scheme, includes that it is difficult to derive the private keys SK_1^i and SK_2^i by way of ID^i, P_{pub1}, P_{pub2}, and P [11]. We not only overcome the replaying attack, but also propose a solution to the other

Table 2 Security comparison

	Batch message verification	Avoiding any replaying attack	Avoiding non-repudiation
Our scheme	√	√	√
Zhang et al.'s scheme [11]	√	×	×
ABAKA [20]	√	√	×

Table 3 Comparison of three schemes in term of the computational complexity

	Signal verification	Batch verification
Our scheme	$3T_{par} + T_{mul}$	$3T_{par} + T_{mul}$
Zhang et al.'s scheme [11]	$3T_{par} + T_{mtp} + T_{mul}$	$3T_{par} + nT_{mtp} + nT_{mul}$
ABAKA [20]	$3T_{mul}$	$(2n + 1)T_{mul}$

n number of verifying signatures

problem, non-repudiation. In our scheme, we used a vector parameter Vec_i to avoid user swap of the M_i and σ_i. If a malicious user wants to deny the signatures by swapping M_i and σ_i, his/her signatures will result in the batch message verification failing. Table 2 is a comparison between our scheme and other schemes which in the same field.

(2) User privacy preserving:

If an adversary attempts to use the information, which is intercepted from public communicating environment, to trace a specific user, he/she needs to determine the relation between each communication. In our scheme, all of information sent by a user is changed in each communication. Therefore, a person's ID_i is converted by an unknown random number r. For this reason, we claim our scheme both achieves and preserves the user anonymity and user privacy.

(3) Audit-ability:

To avoid the user privacy preserving abused by the malicious behaviors, the malicious user should have TA traceability, where the traceability is also called conditional privacy [6]. In the proposed scheme, the TA can trace the RID_i of V_i as the Sect. 5.2 explains. When a user attempts to use malicious information to mislead others, the TA can trace the *RID* of the malicious user, and stop the right of the malicious user.

6.2 Performance evaluation

We evaluate the performance of our scheme in this section. Verification delay is the most important issue, which may affect the value of information. The different calculations in our scheme include one point multiplication over an elliptic curve, notated T_{mul}, map to point hash operation, notated T_{mtp}, and pairing operation, notated T_{par}. We adopt the MNT curve [11, 20, 26], which embeds degree $k = 6$ and 160-bit q, running on an Intel Pentium IV 3.0 GHZ machine. The following results are obtained: T_{mul} is 0.6 ms, T_{par} is 4.5 ms, and T_{mtp} is 0.6 ms. We compare the computational complexity of our scheme with Zhang et al.'s scheme and ABAKA in Table 3. Although our scheme has to compute$Vec_i\sigma_i$, $Vec_iID_1^i$, and $Vec_ih(M_i)h_2(ID_1^i \parallel ID_2^i \parallel T_i)$, the range of Vec_i is vary small, such as 1–10, and the cost of Vec_i's computation is negligible. In fact, the real program design can use addition operation instead of multiplication operation, such as letting σ_i plus Vec_i times instead of computing $Vec_i\sigma_i$. On the other hand, we use a one-way hash function $h_2()$ instead of the map to point function $H()$ and reduce point multiplication over an elliptic curve to improve the performance. Hence, the efficiency of our scheme is more efficient than Zhang et al.'s scheme.

We use the results of the MNT curve and the value of performance comparison to forecast the effect on the batch

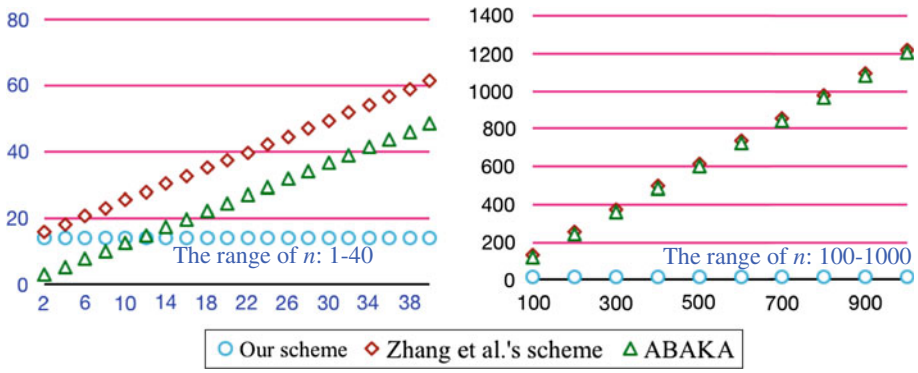

Fig. 1 Effect on the batch verification delay. x-axis: the number of verifying signatures (n), y-axis: the delay time (unit: *ms*)

verification delay of compared schemes in Fig. 1. We let *x*-axis mean the number of verifying signatures (*n*) and *y*-axis mean the delay time (unit: *ms*). The left side of Fig. 1 is the situation while *n* is small (range: 1–40), and the right side is the situation while *n* is large (range: 100–1,000). We can find the slope of our scheme is the lowest. In Fig. 1, although the effect of our scheme isn't the best when *n* is lower than 10, it is faster than others when *n* becomes larger. When *n* is 100, the delay of ABAKS's batch verification is 120.6 ms, Zhang's is 133.5 ms and our scheme's is 14.1 ms. When *n* is 1,000, the delay of ABAKS's batch verification is 1,200.6 ms, Zhang et al.'s scheme is 1,213.5 ms and our scheme's still maintains 14.1 ms; obviously, our scheme is the best. In addition, our scheme is more secure than ABAKA and Zhang et al.'s scheme. For this reason, our scheme is the most suitable for VANET.

7 Conclusions

With the present day communication technology development, VANET can be regarded as the "predictable" technology. In this paper, we proposed an improved batch scheme for VANET, which overcame the weaknesses of Zhang et al.'s scheme and maintain the efficiency. The scheme is designed to improve the quality of traffic. In the future, we would like to further enhance the features of batch scheme for VANET, such as identifying illegal signatures, designing new schemes in order to gain more efficiency.

Acknowledgments The authors would like to express their appreciation to the anonymous referees for their valuable suggestions and comments. This research was partially supported by the National Science Council, Taiwan, R.O.C., under contract no.: 101-2221-E-030-018.

References

1. Dahiya, A., & Chauhan, R. K. (2010). A comparative study of MANET and VANET environment. *Journal of Computing, 2*(7), 87–92.
2. Li, C.-T., & Hwang, M.-S. (2011). A lightweight anonymous routing protocol without public key en/decryptions for wireless ad hoc networks. *Information Sciences, 181*(23), 5333–5347.
3. Sivakumar, R., Sinha, P., & Bharghavan, V. (2003). Braving the broadcast storm: Infrastructural support for ad hoc routing. *Computer Networks, 41*(6), 687–706.
4. Chim, T. W., Yiu, S. M., Hui, L. C. K., & Li, V. O. K. (2011). SPECS: Secure and privacy enhancing communications schemes for VANETs. *Ad Hoc Networks, 9*(12), 189–203.
5. Ghosh, M., Varghese, A., Gupta, A., Kherani, A. A., & Muthaiah, S. N. (2010). Detecting misbehaviors in VANET with integrated root-cause analysis. *Ad Hoc Networks, 8*(7), 778–790.
6. Toor, Y., Muhlethaler, P., & Laouiti, A. (2008). Vehicle ad hoc networks: Applications and related technical issues. *IEEE Communications Surveys and Tutorials, 10*(3), 74–87.
7. Boukerche, A., Oliveira, H. A. B. F., Nakamura, E. F., & Loureiro, A. A. F. (2008). Vehicular ad hoc networks: A new challenge for localization-based systems. *Computer Communications, 31*(12), 2838–2849.
8. Palomar, E., de Fuentes, J. M., González-Tablas, A. I., & Alcaide, A. (2012). Hindering false event dissemination in VANETs with proof-of-work mechanisms. *Transportation Research Part C: Emerging Technologies, 23*, 85–97.
9. Raya, M., & Hubaux, J.-P. (2007). Securing vehicular ad hoc networks. *Journal of Computer Security, 15*(1), 39–68.
10. Wu, T.-Y., Guizani, S., Lee, W.-T., & Liao, K.-H. (2012). Improving RSU service time by distributed sorting mechanism. *Ad Hoc Networks, 10*(2), 212–221.
11. Zhang, C., Ho, P.-H., & Tapolcai, J. (2011). On batch verification with group testing for vehicular communications. *Wireless Networks, 17*(8), 1851–1865.
12. Zhang, C., Lin, X., Lu, R., Ho, P.-H., & Shen, X. (2008). An efficient message authentication scheme for vehicular communications. *IEEE Transactions on Vehicular Technology, 57*(6), 3357–3368.
13. Hubaux, J.-P., Capkun, S., & Luo, J. (2004). The security and privacy of smart vehicles. *IEEE Security and Privacy, 2*(3), 49–55.
14. Li, W., Wen, Q., Su, Q., & Jin, Z. (2011). An efficient and secure mobile payment protocol for restricted connectivity scenarios in vehicular ad hoc network. *Computer Communications, 35*(2), 188–195.
15. Antolino Rivas, D., Barceló-Ordinas, J. M., Guerrero Zapata, M., & Morillo-Pozo, J. D. (2011). Security on vanets: Privacy, misbehaving nodes, false information and secure data aggregation. *Journal of Network and Computer Applications, 34*(6), 1942–1955.
16. Diffie, W., & Hellman, M. (1976). New directions in cryptography. *IEEE Transactions on Information Theory, 22*(6), 644–654.
17. Rivest, R. L., Shamir, A., & Adleman, L. M. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM, 21*(2), 120–126.
18. Lin, X., Sun, X., Ho, P.-H., & Shen, X. (2007). GSIS: A secure and privacy-preserving protocol for vehicular communications. *IEEE Transactions on Vehicular Technology, 56*(6), 3442–3456.
19. Fiat, A. (1990). Batch RSA. In *Lecture notes in computer science* Vol. 435(17), pp. 175–185.
20. Huang, J.-L., Yeh, L.-Y., & Chien, H.-Y. (2011). ABAKA: An anonymous batch authenticated and key agreement scheme for value-added services in vehicular ad hoc networks. *IEEE Transaction on Vehicular Technology, 60*(1), 248–262.
21. Wang, H., & Zhang, Y. (2012). On the security of an anonymous batch authenticated and key agreement scheme for value-added services in VANETs. *Procedia Engineering, 29*, 1735–1739.
22. Chen, L., Ng, S.-L., & Wang, G. (2011). Threshold anonymous announcement in VANETs. *IEEE Journal on Selected Areas in Communications, 29*(3), 605–615.
23. ASTM E2213-03 (2010). Standard specification for telecommunications and information exchange between roadside and vehicle systems 8212; 5 GHz Band Dedicated Short Range Communications (DSRC) Medium Access Control (MAC) and Physical Layer (PHY) Specifications. ASTM.org. Accessed: 2011/11/15, form: http://www.astm.org/Standards/E2213.htm.
24. Boneh, D., & Franklin, M. (2001). Identity-based encryption from the Weil pairing. *Lecture Notes in Computer Science, 2139*(13), 213–229.
25. Scott, M. Efficient implementation of cryptographic pairings. [Online]. Available: fttp://fttp.disi.unige.it/pub/person/MoraF/CRYPTO/PARING/mscott-samos07.pdf, accessed: 2012/4/21.

26. Miyaji, A., Nakabayashi, M., & Takano, S. (2001). New explicit conditions of elliptic curve traces for FR-reduction. *IEICE Transaction on Fundamentals of Electronics, E84-A*(5), 1234–1243.

Author Biographies

Cheng-Chi Lee received the B.S. and M.S. in Information Management from Chaoyang University of Technology (CYUT), Taichung, Taiwan, in 1999 and in 2001. He researched in Computer and Information Science from National Chiao Tung University (NCTU), Taiwan, Republic of China, from 2001 to 2003. He received the Ph.D. in Computer Science from National Chung Hsing University (NCHU), Taiwan, in 2007. He was a Lecturer of Computer and Communication, Asia University, from 2004 to 2007. From 2007, he was an assistant professor of Photonics and Communication Engineering, Asia University. From 2010, he was an assistant professor of Library and Information Science, Fu Jen Catholic University. From 2013, he is an associate professor of L.I.S., Fu Jen Catholic University. Now he is an editorial board member of International Journal of Network Security, Journal of Computer Science, and International Journal of Secure Digital Information Age. His current research interests include data security, cryptography, network security, mobile communications and computing, wireless communications. Dr. Lee had published over 90+ articles on the above research fields in international journals.

Yang-Ming Lai received the B.S. in Library and Information Science, Fu Jen Catholic University, New Taipei City 24205, Taiwan, R. O. C, in 2011. He will receive the M.S. in Library and Information Science, Fu Jen Catholic University, New Taipei City 24205, Taiwan, R. O. C., in 2013. His current research interests include information security, VANET, and remote user authentication.