An Efficient Identity-Based Privacy-Preserving Authentication Scheme for VANETs

Jie Cui[1], Wenyu Xu[1], Kewei Sha[2], and Hong Zhong[1](✉)

[1] School of Computer Science and Technology, Anhui University, Hefei 230039, China
cujie@mail.ustc.edu.cn, zhongh@ahu.edu.cn

[2] Department of Computing Sciences, University of Houston - Clear Lake,
Houston, TX 77058, USA

Abstract. Vehicular ad hoc networks (VANETs) are proposed to improve the traffic safety and efficiency through communications among vehicles and between vehicle and roadside units (RSUs). When a vehicle broadcasts messages to nearby vehicles and roadside units (RSUs), it needs to resist attacks and to preserve the privacy of the message senders. Therefore, security and privacy issues are of great interests and remain challenging. Many authentication schemes are proposed to tackle above challenges while most of them are heavy in computation and communication. In this paper, we propose a novel authentication scheme that utilizes the double pseudonym method to hide the real identity of vehicle and adopts the dynamic update technology to periodically update the information (such as member secret, authentication key, internal pseudo-identity) stored in the tamper-proof device (TPD) to prevent the side-channel attack. Because of not using bilinear pairing, our scheme yields a better performance in terms of computation overhead and communication overhead and is more suitable to be applied in VANETs.

Keywords: Batch verification · VANETs · TPD · Bilinear pairing

1 Introduction

In recent years, the proposal of VANET [1–4] aims to enhance driving safety through inter-vehicle communications and communications between vehicles and roadside infrastructure. Both academia and industry show great interests in developing a secure and efficient VANET. A typical VANET consists of a trusted third party (TA), a set of RSUs distributed along the roads, and many vehicles driving on the road. When the RSUs and vehicles receive the information from vehicle, they need to verify the integrity of the message to ensure that it is not modified by the attacker during the transmission. Moreover, for security concerns, the real identity of vehicle should not be known by a malicious attacker during the transmission to preserve the identity privacy of the sender.

I. Romdhani et al. (Eds.): CollaborateCom 2017, LNICST 252, pp. 508–518, 2018.
https://doi.org/10.1007/978-3-030-00916-8_47

Many efforts have been made to tackle the above challenge, and many authentication schemes have been proposed. Most of them are heavy in computation and communication.

In 2008, Zhang et al. [5] first proposed an identity-based batch authentication scheme using a bilinear mapping. Firstly, in Zhang's scheme [5], they use the batch authentication method to verify the many messages at the same time which can reduce the computation overhead. Secondly, because a vehicle uses a pseudonym identity attached to the message during the transmission process, some untrustworthy parties and malicious attackers could not know the real identity of the vehicle. However, in 2013, Lee and Lai [6] pointed out that Zhang's scheme [5] had some flaws. First of all, Zhang's scheme [5] cannot resist replay attack. In the absence of the corresponding inspection device, the receiver maybe receive a valid signature that has been verified before. Secondly, Zhang's scheme [5] could not achieve non-repudiation. Although a trusted third party (TA) could recover the real identity of false message which is sent by an adversary, the attacker also could deny that sending the corresponding message. Hence, Lee and Lai [6] proposed an improved scheme to achieve better privacy preserving.

Recently, Zhang et al. [7] and Bayat et al. [8] found that Lee and Lai's scheme [6] was not able to resist impersonation attacks, that is, malicious attackers could simulate a legal vehicle to send false messages. Therefore, Zhang et al. [7] and Bayat et al. [8] proposed two improved schemes to address the problem in Lee and Lai's [6] scheme. However, as pointed out in He et al.'s scheme [9], the above two schemes have flaws that they cannot prevent the modification attack, in which the signature of message could be modified by the malicious attacker. Therefore, He et al. [9] proposed a conditional privacy protection scheme that does not use bilinear paring.

In He et al.'s scheme [9], because the system's master private key is stored in a tamper-proof device (TPD) which is a device that no attacker can extract any stored data. However, because of side-channel attack, the assumption of TPD is shown to be too strong in practice. In side-channel attack, the adversary collects side channel information leak from some cryptographic operations. Once the TPD is compromised, the attacker could acquire the system's master private key so that the whole system will be compromised. In order to prevent side-channel attacks, Zhang et al. [10] proposed a novel privacy-preserving authentication scheme. Instead of storing the mast private key in the TPD that cannot be updated, their scheme store security-related information in the TPD, which can be periodically updated. This approach can get rid of the ideal TPD, so it is more practical. However, this scheme uses bilinear mapping and multiple Map-To-Point operations, thus leads to a heavy computational overhead.

To reduce the computation and communication overhead of the existing authentication scheme, in this paper, we propose an efficient identity-based privacy-preserving authentication scheme for VANETs. Our scheme makes use of the double pseudonym method and dynamic update technology. The computation and communication overhead are reduced because no bilinear paring is needed in the signature generation and verification. In addition, we show that the proposed scheme is secure via comprehensive security analysis. Finally, we periodically update the informations (e.g., member secret, authentication key, IPID) stored in the tamper-proof device, therefore, our scheme can resist the side-channel attack.

2 System Model and Design Goals

In this chapter, we briefly introduce the network model, security requirements. Some notations are defined as shown in Table 1.

Table 1. List of notations and definitions

Notation	Definitions
TA	A trusted authority
s, P_{pub}	The private key and public key of TA
$cert_{R_j}$	A certificate of R_j issued by TA
$ID_{\{R_j,V_i\}}$	The real identity of R_j or V_i
VP_i	The validity period of $IPID_{V_i}$
$IPID_{V_i}$	An internal pseudonym identity of V_i, generated by the TA
$PPID_{i,t}$	The public pseudonym identity of V_i, generated from $IPID_{V_i}$ of V_i
$h_{\{R_j,\text{TA}\}}$	A hash-based message authentication code generated by R_j or the TA
$E_\pi(.)/D_\pi(.)$	A symmetric encryption scheme, where π is the key

2.1 Network Model

As shown in Fig. 1, a VANET consists of a third-party trusted authority (TA), some RSUs distributed on the roadside and multiple vehicles.

- **TA:** TA is a trusted third party in VANET, with sufficient storage and computing power, and is considered impossible to compromise with an adversary. When an attacker simulates a legal vehicle sends a false message, the TA can resume the true identity of the sent message.
- **RSUs:** The RSU is an infrastructure that is distributed on the roadside and communicates with the TA via a wired connection, and communicates with vehicles over a wireless connection to verify the validity of the received message.
- **Vehicles:** Each vehicle is equipped with TPD, and communicates with other vehicles and RSUs through wireless connections. The vehicle periodically broadcasts security-related messages to nearby vehicles and RSUs through the Dedicated Short Range Communications (DSRC) [11] protocol.

2.2 Security Requirements

A security scheme for VANETs should meet some of the following features.

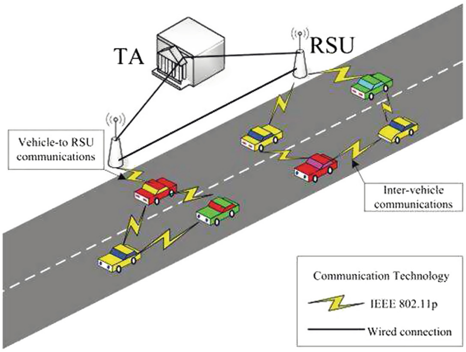

Fig. 1. System model.

1. Message integrity: In VANETs, we need to ensure that the recipient received the message from the sender, and the message during the sending process has not been modified by the attacker to maintain integrity.
2. Non-forgery: The attacker should not generate a valid signature on behalf of any vehicle under the randomly selected message attack in the random oracle model.
3. Resistance against side-channel attack: The attacker should not be able to obtain any informations stored in the TPD through the side-channel attack.

3 The Proposed Scheme

In this section, we proposed an efficient identity-based privacy-preserving authentication scheme that does not use bilinear paring to address the security problem existing in VANET. They are some initialization parameters that pre-load into the vehicles and RSUs generated by the TA using the following steps.

1. TA selects a random number s $\in Z_q^*$ as the secret key of the TA and calculates $P_{pub} = \mathrm{s} \cdot \mathrm{P}$ as the public key of the TA, where the P is the generator of G.
2. TA selects $E_\pi(\cdot)/D_\pi(\cdot)$ and some hash functions: $h_1 : G \to Z_q, h_2 : \{0,1\}^* \to Z_q$, $H_{1_{key}}(\cdot) : \{0,1\}^* \to \{0,1\}^l$, $H_2(\cdot) : \{0,1\}^* \to \varGamma$, $H_3(\cdot) : \{0,1\}^* \to \{0,1\}^{l'}$, where $H_{1_{key}}(\cdot)$ as a keyed hash.
3. The system parameters $\psi = (\mathrm{P}, P_{pub}, h_1, h_2, H_{1_{key}}(\cdot), H_2(\cdot), H_3(\cdot), E_\pi(.)/D_\pi(.))$. Pre-load the system parameters ψ into the vehicles and RSUs.

This scheme consists of six phases: RSU setup phase, vehicle setup phase, member key generation phase, vehicle signature phase, message verification phase, IPID and authentication key update phase.

3.1 RSU Setup Phase

In this phase, the RSU generates its own public-private key pairs and the corresponding certification from the TA. This certification can be used only in the short time. Once the period is over, the RSU should execute the step once again. To generate its own public-private key pairs, the RSU randomly chooses two numbers $k_j, \eta_j \in Z_q^*$ and computes $PK_{R_{j1}} = k_j P, PK_{R_{j2}} = \eta_j P$. The private key is (k_j, η_j) and the public key is $\left(PK_{R_{j1}}, PK_{R_{j2}}\right)$, where k_j is used to generate the shares of vehicle, and η_j is used to generate the secure channel between the RSU and vehicle. After generating its public key, the RSU sends the public key $\left(PK_{R_{j1}}, PK_{R_{j2}}\right)$ and its own identity information to the TA through the secure channel. When the TA receives the messages, it generates the certification of RSU. Then the $cert_{R_j}$ is broadcasted within RSU range.

3.2 Vehicle Setup Phase

In this phase, when the vehicle joins the range of the VANETs, the information stored in the TPD should be initialized. Assuming the real identity of vehicle is RID, the TA can compute the inter-pseudonym identity $IPID_{V_i} = H_{1_{\Lambda}}(RID||VP_i)$, where the VP_i is the valid period of the inter-pseudonym identity like 02.03.2017–03.04.2017. The vehicle chooses the authentication key λ_i, putting the $\psi, IPID_{V_i}, \lambda_i$ into the TPD. $(\text{RID}, VP_i, IPID_{V_i}, \lambda_i)$ is stored into the member list ML in the TA.

3.3 Member Key Generation Phase

In this phase, when the vehicle enters the communication range of RSU, it will receive the certification from RSU and first check the validity of the $cert_{R_j}$. If the certification is valid under the public key of system, extracting the public key and identity of RSU from the certification $cert_{R_j}$. Then, the vehicle chooses a random number r $\in Z_q^*$, and computes $f = rP$, $\pi_{i1} = H_2(f, PK_{R_{j2}}, rPK_{R_{j2}}, ID_{R_j}, T_i)$, $\pi_{i2} = H_2\left(f, P_{pub}, rP_{pub}, ID_{R_j}, T_i\right)$. Where T_i is a timestamp, π_{i1},π_{i2} are used as the keys of the symmetric encryption scheme $\left(E_\pi\left(.\right)/D_\pi\left(.\right)\right)$. Finally the vehicle computes $p_j = E_{\pi_{i2}}\left(\lambda_i, T_i\right)$ and sends s $= \left(\text{f}, ID_{R_j}, p_j, T_i\right)$ to RSU.

After receiving s from vehicle, the RSU checks the validity of T_i, if it is invalid, then it aborts; otherwise it sends s to the TA through the secure channel. When TA receives s and computes $\pi_{i2} = H_2\left(\text{f}, P_{pub}, \text{sf}, ID_{R_j}, T_i\right)$, $D_{\pi_{i2}}\left(p_j\right)$ to get $(\lambda_i^{'}, T_i^{'})$. If it does not appear the equation $\lambda_i^{'} \neq \lambda_i$ in a tuple of member list $(\text{RID}, VP_i, IPID_{V_i}, \lambda_i)$ of the TA or $T_i \neq T_i^{'}$ or VP_i is invalid, it aborts; otherwise TA authenticates the vehicle and sends authenticated message to RSU.

Upon the RSU receives the authenticated message from the TA, it means the vehicle is legal. RSU first computes $\pi_{i1} = H_2\left(f, PK_{R_{j2}}, f\eta_j, ID_{R_j}, T_i\right)$; and chooses an authenticated period τ_p and member secret (β_j, γ_j), where β_j and γ_j satisfy $k_j = \beta_j \cdot \gamma_j$; it computes $h_{R_j} = H_{1_{\pi_{i1}}}\left(\beta_j, \gamma_j, \tau_p\right)$, and $p_j^{'} = E_{\pi_{i1}}\left(\beta_j, \gamma_j, \tau_p, h_{R_j}\right)$; and sends t $= (H_3\left(\text{f}\right), p_j^{'})$ to the vehicle.

When the vehicle receives the t and $D_{\pi i1}\left(p_j^{'}\right)$ to get $\left(\beta_j, \gamma_j, \tau_p, h_{R_j}\right)$, it verifies whether the equation $h_{R_j} = H_{1_{\pi_{i1}}}\left(\beta_j, \gamma_j, \tau_p\right)$ holds. If so, it lets the member secret and authenticated period in the TPD; otherwise, it aborts. This member key can only be used under the authenticated period, and once it expires, the member key stored in the TPD is deleted.

3.4 Vehicle Signature Phase

In this phase, when a vehicle obtains the member secret (β_j, γ_j) and the corresponding validity period from the RSU, the vehicle first computes the external pseudonym identity $PPID_i = H_3\left(IPID_{V_i}, T_i\right)$ and the one time signature key $sk_i = \left(\beta_j \cdot \gamma_j\right) \cdot h_1\left(PPID_i\right) \; mod \; n$. Then, the vehicle chooses a random number $r_i \in Z_q^*$, and computes $R_i = r_i \cdot P$, $\beta_i = h_2\left(PPID_i \left|\right|R_i\left|\right| M_i\right)$, $S_i = sk_i + \beta_i \cdot r_i$. Finally, it sends $(M_i, PPID_i, R_i, S_i)$ to nearby vehicles and RSUs.

The member secret (β_j, γ_j) stored in the TPD needs to be periodically updated. Choose a random number r $\in Z_q^*$, and set the $\beta_j = r \cdot \beta_j$, $\gamma_j = r \cdot \gamma_j$ as the new member secret.

3.5 Message Verification Phase

In this phase, after receiving multiple message $(M_1, PPID_1, R_1, S_1)$, $(M_2, PPID_2, R_2, S_2), ..., (M_n, PPID_n, R_n, S_n)$ from the vehicle, verifier first checks the validity of T_i, where $i = 1, 2, \ldots, n$. If T_i is invalid, the verifier rejects the messages; otherwise, it chooses a random vector v $= \{v_1, v_2, \ldots, v_n\}$, where v_i is a small random integer in $[1, 2^t]$ and t is a small integer with low overhead. Then, the verifier checks the correctness of the equation $(\sum_{i=1}^{n} v_i \cdot S_i) \cdot P = \sum_{i=1}^{n}(V_i \cdot h_1\left(PPID_i\right)) \cdot PK_{R_{j1}} + \sum_{i=1}^{n}(v_i \cdot \beta_i \cdot R_i)$. If it does not hold, the verifier rejects the messages; otherwise, the verifier receives the messages.

Since $sk_i = \left(\alpha_j \cdot \beta_j\right) \cdot h_1\left(PPID_i\right) \; mod \; n$, $\beta_j \cdot \gamma_j = k_j$, $PK_{R_{j1}} = k_j \cdot P$, $R_i = r_i \cdot P$, $\beta_i = h_2\left(PPID_i \left|\right|R_i\left|\right| M_i\right)$ and $S_i = sk_i + \beta_i \cdot r_i$, we can get the equation hold. Hence, the correctness of the multiple messages verification is verified.

3.6 IPID and Authentication Key Update Phase

At this phase, when a vehicle wants to update the internal pseudo-identity and authentication key, it first chooses a random number $t \in Z_q^*$, and computes $g = t \cdot$ P, $\pi_i = H_2\left(g, P_{pub}, tP_{pub}, T_i\right)$, $p_i = E_{\pi_i}\left(\lambda_i, T_i\right)$. Then, it sends z $= (g, T_i, p_i)$ to the TA through the nearby RSU.

After the TA receives z, if T_i is invalid, it aborts; otherwise, it first computes $\pi_i = H_2(g, P_{pub}, s \cdot g, T_i)$, $D_{\pi_i}\left(p_i\right)$ to get $(\lambda_i^{'}, T_i^{'})$ and checks the validity of $T_i^{'}$. If $T_i^{'}$ is invalid, it aborts; otherwise, it searches the member list for a tuple (RID, $VP_i, IPID_{V_i}, \lambda_i$) such as $\lambda_i = \lambda_i^{'}$. If such a tuple does not exist, it aborts; otherwise, TA checks the validity of VP_i. If it is invalid, choose a new valid period $VP_i^{'}$. Then, it computes $IPID_{V_i}^{'} = H_{1_\Lambda}(RID||VP_i^{'})$ and chooses a new authentication key $\widehat{\lambda_i}$; otherwise, it aborts. Finally, TA computes $p_i = E_{\pi_i}(IPID_{V_i}^{'}, \widehat{\lambda_i}, T_i^{'}, h_{TA})$. If $h_{TA} = H_{1_{\lambda_i^{'}}}\left(IPID_{V_i}^{'}, \widehat{\lambda_i}, T_i\right)$ is an $HAMC$, sends $(H_3\left(g\right), p_i^{'})$ to the vehicle and put (RID, $VP_i^{'}, IPID_{V_i}^{'}, \widehat{\lambda_i}$) into ML.

Upon receiving $(H_3\left(g\right), p_i^{'})$, the vehicle first computes $D_{\pi_i}\left(p_i^{'}\right)$ to get $(IPID_{V_i}^{'}, \lambda_i^{'}, T_i^{'}, h_{TA}^{'}$. Then, it checks the validity of $T_i^{'}$ and $h_{TA}^{'}$. If it is invalid, set the $\left(IPID_{V_i}^{'}, \lambda_i^{'}\right)$ as the new internal pseudo-identity and authentication key.

4 Security Proof and Analysis

Because the computational Elliptic Curve Discrete Logarithm (ECDL) problem is hard to address, so any attacker could not generate a valid signature on behalf of any vehicle through the game that is made up of a challenger C and an adversary A.

Theorem 1: Our scheme for VANETs is secure existential forgery under the randomly selected message attack in the random oracle model.

Proof: Assuming there is an adversary could forge message $(M_i, PPID_i, R_i, S_i)$, then we construct a challenger C, which could address the ECDL problem through running A as a subroutine. The details are as the following steps:

Setup stage: Challenger C first sets Q $= PK_{R_{j1}}$, then it sends the system parameters ψ to an adversary A.

$h_1 - oracle$: Challenger C first initializes the list L_{h_1} with the form of $(\langle PPID_i, \tau_{h_1}\rangle)$. When receiving the query of the message with the form of $<PPID_i>$ from the adversary A, the challenger C checks a tuple of the $<PPID_i>$ to find out whether it appears in the list L_{h_1}. If the tuple exists in the list L_{h_1}, then send $\tau_{h_1} = h_1\left(PPID_i\right)$ to the adversary A; otherwise, C chooses a random number $\tau_{h_1} \in Z_q^*$ and sets the tuple $\langle PPID_i, \tau_{h_1}\rangle$ into the L_{h_1}, finally sends the $\tau_{h_1} = h_1\left(PPID_i\right)$ to A.

$h_2 - oracle$: Challenger C first initializes the list L_{h_2} with the form of L_{h_2} $(\langle PPID_i, R_i, \mathrm{M}_i, \tau_{h_2}\rangle)$. When receiving the query of the message with the form of $\langle PPID_i, R_i, \mathrm{M}_i\rangle$ from the adversary A, the challenger C checks a tuple of the $\langle PPID_i, R_i, \mathrm{M}_i\rangle$ whether it appears in the list L_{h_2}. If the tuple exists in in the list L_{h_2}, then send $\tau_{h_2} = h_2\left(PPID_i\,||R_i||\,\mathrm{M}_i\right)$ to the adversary A; otherwise, C chooses a random number $\tau_{h_2} \in Z_q^*$ and sets the tuple $(\langle PPID_i, R_i, \mathrm{M}_i, \tau_{h_2}\rangle)$ into the L_{h_2}, finally sends the $\tau_{h_2} = h_2\left(PPID_i\,||R_i||\,\mathrm{M}_i\right)$ to A.

$sign - oracle$: Upon receiving the message M_i from adversary A, challenger C generates random numbers S_i, $h_{i,1}, \beta_i \in Z_q^*$ and $PPID_i$. Challenger C puts $\langle PPID_i, h_{i,1}\rangle$ and $(M_i, PPID_i, R_i, S_i)$ to adversary A, which is easy to verify equation $S_i \cdot P = h_1\left(PPID_i\right) \cdot PK_{R_{j1}} + \beta_i \cdot R_i$ hold. thus, the message and signature $(M_i, PPID_i, R_i, S_i)$, which A acquired from the inquiry from C, is valid.

Output: Finally, A outputs the message $(M_i, PPID_i, R_i, S_i)$. C checks whether the equation holds.

$$S_i \cdot P = h_1\left(PPID_i\right) \cdot PK_{R_{j1}} + \beta_i \cdot R_i \tag{1}$$

If it does not hold, C aborts the process; otherwise, because of the forged lemma, if A executes $h_1-oracle$ once again, a valid message $(M_i, PPID_i, R_i, S_i^{'})$ will be generated. It can also conclude the similar equation.

$$S_i^{'} \cdot P = \left(h_1\left(PPID_i\right)\right)^{'} \cdot PK_{R_{j1}} + \beta_i \cdot R_i \tag{2}$$

According to the Eqs. (1) and (2), we could get

$$\left(S_i - {S_i}^{'}\right) = \left(h_1\left(PPID_i\right) - \left(h_1\left(PPID_i\right)\right)^{'}\right) \cdot k_j \tag{3}$$

Therefore, C output the $\left(h_1\left(PPID_i\right) - \left(h_1\left(PPID_i\right)\right)^{'}\right)^{-1} \cdot \left(S_i - S_i^{'}\right)$. However, it is difficult to address the ECDL problem, so our scheme is secure against forgery under the randomly selected message attack in the random oracle model.

We will introduce the security requirement as described in Subsect. 2.2.

1. **Message integrity:** According to the Theorem 1, because it is difficult to address the ECDL problem, the signature used in our scheme is not forged under the random oracle model. Therefore, no adversary can simulate a legal vehicle to generate a valid signature or modify a legal signature. We can verify the equation that $S_i \cdot P = h_1\left(PPID_i\right) \cdot PK_{R_{j1}} + \beta_i \cdot R_i$ holds to check the validity and integrity of the message $(M_i, PPID_i, R_i, S_i)$. Thus, the proposed scheme can achieve message integrity.
2. **Non-forgery:** Because it is difficult to address the ECDL problem, so the attacker could not generate a valid signature on behalf of any vehicle under the randomly selected message attack in the random oracle model. Thus, the proposed scheme can achieve non-forgery.
3. **Resistance side channel attack:** Due to the IPID is often used, if the vehicle does not periodically update this information, it will give the attacker a chance to recover the real identity of vehicle. In our scheme, before the attacker can probe the related information to recover the IPID through the side channel attack, the IPIP has already been updated. Secondly, the authenticated key can only be used during the authentication of vehicle. It is much harder for the attacker to resume the authenticated key than recover the IPID. In addition, as for the member secret, even if the adversary could recover the member secret, only vehicle in the nearby RSU can be influenced. Furthermore, because the RSU can periodically update its public-private key pairs, hence, the attacker could not acquire enough information through the side channel to resume the member key stored in the TPD.

5 Performance Analysis and Comparison

5.1 Computation Overhead Analysis

Table 2 shows some time-consuming cryptographic operations [12] need to be executed in each scheme. The processing time for the bilinear pairing operation T_p is 4.211 ms, the time for small scale multiplication operation T_{mp-p} is 1.709 ms and the Map-To-Point operation T_{mtp} is 4.406 ms. The time for small scale multiplication operation based on the Elliptic Curve T_{mp-ECC} is 0.442 ms [9]. Figure 2 shows the total execution time of the batch verification as the amount of the vehicle increasing in each scheme. When the authenticated vehicle is increased to 100, in our scheme, the total execution time is less than 50 ms. Hence, our scheme is more suitable for the scene of multiple vehicles in VANETs.

5.2 Communication Overhead Analysis

In the group G_1 based on bilinear mapping, the size of the elements in G_1 is $64 \times 2 = 128$ byte [13]. However, in the group G based on the Elliptic Curve, the size of the elements in G is $20 \times 2 = 40$ byte [9]. Furthermore, we assume that the size of result of the general hash function is 20 byte and the size of the timestamp is 4 byte [14]. In addition, we do not consider the size of the message which is transmitted by the vehicle in this phase [15]. Table 3 lists the

Table 2. The computation overhead of each scheme

Scheme	Pseudonym and signature generation phase	Multiple messages verification phase
Zhang et al. [7]	$6T_{mp-tp} + T_{mtp}$	$(n+1)\,T_{mp-bp} + 3T_{bp}$
Bayat et al. [8]	$5T_{mp-bp} + T_{mtp}$	$3T_{bp}+nT_{mp-bp}+nT_{mtp}$
Zhang et al. [10]	$2T_{mtp}$	$2T_{bp} + 2nT_{mtp}$
He et al. [9]	$3T_{mp-ECC}$	$(n+2)\,T_{mp-ECC}$
Our Scheme	T_{mp-ECC}	$(n+2)\,T_{mp-ECC}$

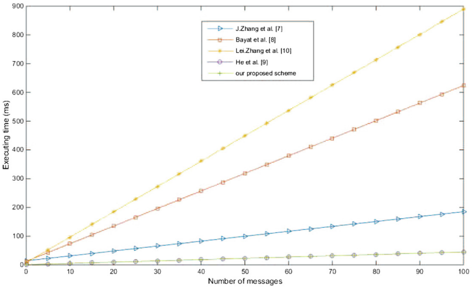

Fig. 2. Computation overhead comparison of verifying multiple message.

Table 3. THE communication overhead of each scheme

Scheme	Sending a single message	Sending n messages
Zhang et al. [7]	388 bytes	388n bytes
Bayat et al. [8]	388 bytes	388n bytes
Zhang et al. [10]	148 bytes	148n bytes
He et al. [9]	144 bytes	144n bytes
Our Scheme	80 bytes	80n bytes

communication overhead of our scheme compared with the schemes of Zhang et al. [7], Bayat et al. [8], Zhang et al. [10] and He et al. [9].

6 Conclusion

In this paper, we propose an efficient identity-based privacy-preserving authentication scheme supports both V2V communication and V2I communication in VANETs. Firstly, unlike other schemes, which stores the system master secret (that cannot be updated) in the TPD, in our scheme, the informations stored in the TPD are regularly updated. Therefore, the proposed scheme can resist side-channel attacks and hence is more practical. Secondly, the security analysis shows that our scheme can satisfy the security requirements for VANETs. Furthermore, performance analysis and comparison shows that our scheme is better than other schemes in terms of computation overhead and communication overhead. This shows our scheme is more suitable used in the VANETs.

References

1. Sha, K., Xi, Y., Shi, W., Schwiebert, L., Zhang, T.: Adaptive privacy-preserving authentication in vehicular networks. In: First International Conference on Communications and Networking in China, ChinaCom 2006, pp. 1–8. IEEE (2006)
2. Xi, Y., Sha, K., Shi, W., Schwiebert, L., Zhang, T.: Enforcing privacy using symmetric random key-set in vehicular networks. In: Eighth International Symposium on Autonomous Decentralized Systems, ISADS 2007, pp. 344–351. IEEE (2007)
3. Qu, F., Wu, Z., Wang, F.-Y., Cho, W.: A security and privacy review of VANETs. IEEE Trans. Intell. Transp. Syst. **16**(6), 2985–2996 (2015)
4. Wen, X., Shao, L., Xue, Y., Fang, W.: A rapid learning algorithm for vehicle classification. Inf. Sci. **295**, 395–406 (2015)
5. Zhang, C., Lu, R., Lin, X., Ho, P.-H., Shen, X.: An efficient identity-based batch verification scheme for vehicular sensor networks. In: The 27th Conference on Computer Communications, INFOCOM 2008, pp. 246–250. IEEE (2008)
6. Lee, C.-C., Lai, Y.-M.: Toward a secure batch verification with group testing for VANET. Wirel. Netw. **19**(6), 1441 (2013)
7. Zhang, J., Xu, M., Liu, L.: On the security of a secure batch verification with group testing for VANET. Int. J. Netw. Secur. **16**(5), 351–358 (2014)

8. Bayat, M., Barmshoory, M., Rahimi, M., Aref, M.R.: A secure authentication scheme for VANETs with batch verification. Wirel. Netw. **21**(5), 1733 (2015)
9. He, D., Zeadally, S., Xu, B., Huang, X.: An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks. IEEE Trans. Inf. Forensics Secur. **10**(12), 2681–2691 (2015)
10. Zhang, L., Wu, Q., Domingo-Ferrer, J., Qin, B., Hu, C.: Distributed aggregate privacy-preserving authentication in VANETs. IEEE Trans. Intell. Transp. Syst. **18**(3), 516–526 (2017)
11. Oh, H., Yae, C., Ahn, D., Cho, H.: 5.8 GHz DSRC packet communication system for its services. In: IEEE VTS 50th Vehicular Technology Conference, VTC 1999-Fall, vol. 4, pp. 2223–2227. IEEE (1999)
12. He, D., Kumar, N., Shen, H., Lee, J.-H.: One-to-many authentication for access control in mobile pay-TV systems. Sci. China Inf. Sci. **5**(59), 1–14 (2016)
13. Boyen, X., Martin, L.: Identity-based cryptography standard (IBCS) 1: supersingular curve implementations of the BF and BB1 cryptosystems. Technical report (2007)
14. Adams, C., Pinkas, D.: Internet X.509 public key infrastructure time stamp protocol (TSP) (2001)
15. Lo, N.-W., Tsai, J.-L.: An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks without pairings. IEEE Trans. Intell. Transp. Syst. **17**(5), 1319–1328 (2016)