sensors

MDPI

Article

Privacy-Preserving Authentication Using a Double Pseudonym for Internet of Vehicles

Jie Cui [1,2], Wenyu Xu [1,2], Hong Zhong [1,2,*], Jing Zhang [1,2], Yan Xu [1,2] and Lu Liu [3]

[1] School of Computer Science and Technology, Anhui University, Hefei 230039, China; cuijie@mail.ustc.edu.cn (J.C.); xwysml@163.com (W.X.); root_zj@163.com (J.Z.); xuyan@ahu.edu.cn (Y.X.)
[2] Anhui Engineering Laboratory of IoT Security Technologies, Anhui University, Hefei 230039, China;
[3] Department of Computing and Mathematics, University of Derby, Derby DE22 1GB, UK; l.liu@derby.ac.uk
* Correspondence: zhongh@ahu.edu.cn; Tel.: +86-0551-6386-1263

Received: 14 March 2018; Accepted: 1 May 2018; Published: 7 May 2018

Abstract: The Internet of Vehicles (IoV) plays an important role in smart transportation to reduce the drivers's risk of having an accident and help them manage small emergencies. Therefore, security and privacy issues of the message in the tamper proof device (TPD) broadcasted to other vehicles and roadside units (RSUs) have become an important research subject in the field of smart transportation. Many authentication schemes are proposed to tackle the challenges above and 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 vehicles 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 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 the Internet of Vehicles.

Keywords: batch verification; IoV; TPD; privacy-preserving authentication

1. Introduction

In recent years, the Internet of Vehicles (IoV) [1–4] has aimed to enhance driving safety through inter-vehicle communications and communications between vehicles and roadside infrastructure [5–7]. Both academia and industry show great interests in developing a secure and efficient IoV. A typical IoV consists of a trusted third party (TA), a set of Roadside Units (RSUs) distributed along the roads, and many vehicles driving on the road. In IoV, the RSUs communicate with the TA via wired connections, and communicate with the vehicles via wireless channels. A vehicle periodically broadcasts traffic safety related messages such as the speed of the vehicle, the road condition, etc., to nearby vehicles and RSUs using the Dedicated Short Range Communications (DSRC) [8] protocol. These messages can be helpful to deal with emergency road conditions and reduce the risk of accidents.

After receiving messages sent by a vehicle, the RSU or vehicle needs to verify the integrity of the message to ensure that it is not modified by the attacker during the transmission. Meanwhile, the real identity of the vehicle should not be known by a malicious attacker during the transmission to preserve the identity privacy of the sender. However, false messages from attackers may cause significant damages; therefore, for security concerns, a trusted third party is needed to retrieve the real identify and locate the attackers who send the false messages. Many efforts have been made to tackle the above challenge, and many authentication schemes including Chim [9] have been proposed. Most of them are heavy in computation and communication.

To reduce the computation and communication overhead of the existing authentication scheme, in this paper, we propose the novel privacy-preserving authentication using a double pseudonym for the Internet of Vehicles. 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.

The remainder of this paper is organized as follows. Section 2 shows the related work about the identity-based scheme for IoV. The system model and security requirements are presented in Section 3. We describe the design of our scheme in Section 4 and the security analysis of the proposed scheme is indicated in Section 5. Section 6 analyzes the computational overhead and the communication overhead of the proposed scheme. Finally, conclusions and future work are presented in Section 7.

2. Related Work

Security and Privacy issues have attracted wide attention in IoV. Based on our knowledge, there are three types of authentication methods, namely, an anonymous certificate authentication scheme based on Public Key Infrastructure (PKI) [10], a group authentication scheme based on group signature, and a signature verification scheme based on identity. In 2006, Gamage et al. [11] used an identity-based ring signature scheme to protect the true identity of the signer. However, it was not possible to retrieve the true identity of the sender when the message was disputed. Later, Raya and Hubaux [12] proposed a PKI-based authentication scheme to achieve privacy preserving. Firstly, in order to protect the real identity of the vehicle, every vehicle needs to pre-load a large number of public and private key pairs and the corresponding certificate, which caused a serious storage management burden for a vehicle. Secondly, the trusted third party (TA) also suffers from a heavy certificate management burden to maintain all the anonymous certificates of all the vehicles. Furthermore, when the RSU or vehicle checks the validity of the signature, it is necessary to check the validity of the corresponding certificate, which also causes additional overhead to the system.

Some group-based signature schemes [13–15] were also proposed in the same year, where the group manager holds the private key of the group and can restore the true identity of the message signed by any vehicle in the group. In Lin's scheme [14], many vehicles form a group in which each vehicle has its own private key and shares a group of public keys. They use a group signature to implement anonymous authentication of messages sent from a vehicle, and to use identity-based signatures to ensure the integrity of the messages sent from the RSU. A vehicle generates a signature of the corresponding message with its own private key, and the adversary could not link two anonymous identities or two signatures generated by the same vehicle. Hence, the proposed scheme provides unlinkability. Although a traditional certificate management problem is avoided in the group signature-based authentication scheme, the size of the CRL (Certificate Revocation List) grows as the number of recovered signers increases. Each CRL checking operation involves two pairing operations, which results in serious computational overhead for signature verification.

In order to neutralize the above two schemes,in 2008, Zhang et al. [16] first proposed an identity-based batch authentication scheme using a bilinear mapping. Firstly, in Zhang's scheme [16], a large number of public and private key pairs and the corresponding certificate do not need to pre-load into a vehicle, which greatly reduces the overhead of transmitting and verifying the public key certificate. Secondly, the scheme uses the batch authentication method to verify the many messages at the same time, which can reduce the computation overhead. Thirdly, since 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. Finally, when a false message is found, a trusted third party has the ability to reveal the true identity of a vehicle. Therefore, the conditional privacy protection could be achieved.

However, in 2013, Lee and Lai [17] pointed out that Zhang's scheme [16] had some flaws. First of all, Zhang's scheme [16] 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 [16] could not achieve non-repudiation. Although a trusted third party (TA) could recover the real identity of a false message that is sent by an adversary, the attacker could also deny sending the corresponding message. Hence, Lee and Lai [17] proposed an improved scheme to achieve better privacy preserving.

Recently, Zhang et al. [18] and Bayat et al. [19] found that Lee and Lai's scheme [17] was not able to resist impersonation attacks, that is, malicious attackers could simulate a legal vehicle to send false messages. Therefore, Zhang et al. [18] and Bayat et al. [19] proposed two improved schemes to address the problem in Lee and Lai's scheme [17]. However, as pointed out in He et al.'s scheme [20], the two schemes above have flaws in that they cannot prevent the modification attack in which the signature of message could be modified by the malicious attacker. Therefore, He et al. [20] proposed a conditional privacy protection scheme that does not use bilinear paring.

In He et al.'s scheme [20], since the system's master private key is stored in a tamper-proof device (TPD), which is a device from which that no attacker can extract any stored data, the attacker could not acquire the system's master private key to control the whole system. However, in a side-channel attack, the adversary collects a 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. [21] proposed a novel privacy-preserving authentication scheme. Instead of storing the master private key in the TPD that cannot be updated, their scheme stores 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, and thus leads to a heavy computational overhead.

3. System Model and Design Goals

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

Table 1. List of notations and definitions.

Notation	Definitions
TA	A trusted authority
R_j	The j-th RSU
V_i	The i-th vehicle
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 $IPID$
$IPID_{V_i}$	An internal pseudonym identity of V_i, generated by the TA based on ID_{V_i}
$PPID_{i,t}$	The public pseudonym identity of V_i, generated from $IPID_{V_i}$ of V_i
$h_{\{R_j,\mathrm{TA}\}}$	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

3.1. Network Model

As shown in Figure 1, an IoV 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 IoV, with sufficient storage and computing power, and is considered impossible to compromise by an adversary. When an attacker simulates that a 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) protocol.

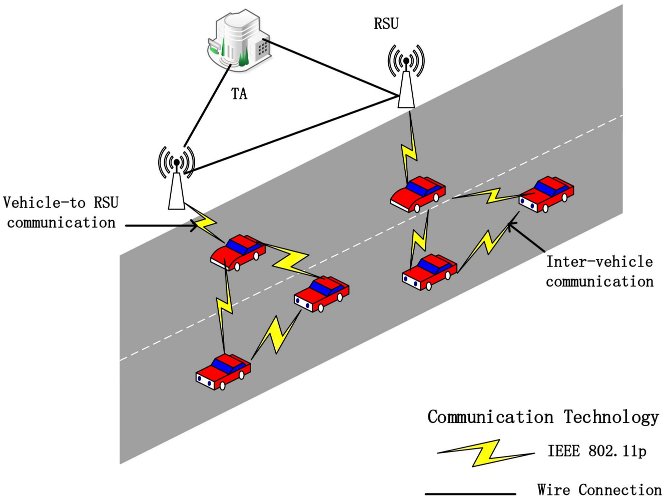

Figure 1. System model.

3.2. Security Requirements

A security scheme for IoV should meet some of the following features:

1. Message integrity: In IoV, 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. Traceability: When an attacker presents as being a legal vehicle and sends a false message that may cause damage, the TA can reveal the real identity of the false message.
4. Non-repudiation: When the trusted third party (TA) retrieves the real identity of the false message, the sender of the message could not deny the attack.
5. Resistance against side-channel attack: The attacker should not be able to obtain any information stored in the tamper-proof device through the side-channel attack.

4. The Proposed Scheme

Recently, some safety-related authentication schemes for IoV have been proposed. However, most of them are heavy in computation and communication, and could not resist some attack existing in IoV. In order to deal with the security problem existing in IoV, we proposed the privacy-preserving authentication using a double pseudonym that does not use bilinear paring, which can be used in the inter-vehicle communications and vehicle to RSU communications. Figure 2 graphically describes the details of our scheme. TA generates the private key and system parameters in our scheme. Each RSU has its own public-private key pairs and the corresponding certification from the TA. When a vehicle enters the range of RSU, then it will request the shares (member secret) of RSU, after authenticating the identity of vehicle via the TA, the shares and the corresponding authenticated period will be sent to the vehicle. This authenticated period is valid for a short time. Once it expires, it should be executed once again. Upon the vehicle receiving the shares and authenticated period, it generates a one-time use private key and signature. This signature could be verified by other vehicles and RSUs. If a vehicle sends a false message, the TA could trace the real identity of the vehicle.

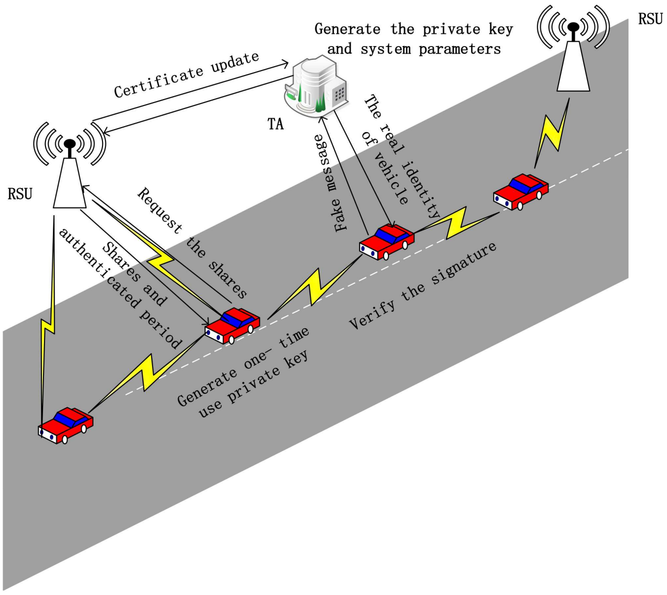

Figure 2. Graphical representation of our scheme.

This scheme can be divided into the following modules:

- **System Setup**: In this phase, the TA generates the private key and system parameters.
- **RSU Setup**: In this phase, the RSU can generate its own public-private key and the corresponding certification $cert_{R_j}$ from the TA.
- **Vehicle Setup**: In this phase, when the vehicle joins into the IoV, the TA generates the inter-pseudonym identity (IPID). The vehicle chooses the authentication key λ_i, and puts the IPID and the λ_i into the tamper-proof device.
- **Member key generation**: In this phase, when the vehicle enters the range of the RSU, the vehicle will request to acquire the member secret of the RSU. After the RSU authenticates the identity of the vehicle, it sends the member secret (β_j, γ_j) and the corresponding valid period (VP_i) to the vehicle.
- **Vehicle Sign**: In this phase, if the vehicle wants to send a message, it will first generate its own external pseudonym identity and one-time signing key, and then generate the signature of the corresponding message.
- **Message Verification**: In this phase, we will use batch authentication to verify the validity of signatures without bilinear pairing, which greatly reduced the computation overhead.
- **IPID and authentication key updated**: In this phase, vehicles can use the online mode to update their own inter-pseudonym identity and authentication key and to prevent the attacker from tracking the true identity of the vehicle.

4.1. System Setup Phase

In this phase, there are some initialization parameters that preload into the vehicles and RSUs generated by the TA using the following steps. This can be done once, unless the private key of the system is compromised by an attacker, or the system wants to periodically update the system parameters and private key to enhance the system security level:

1. TA selects two large prime p and q as well as a non-singular elliptic curve E defined by the equation $y^2 = x^3 + ax + b \ mod \ n$, where a, b $\in F_p$.
2. TA selects the cyclic addition group G, where the P is the generator and q is the order of group.

3. TA selects a random number s $\in Z_q^*$ as the secret key of the TA, and calculates $P_{pub} = \text{s}{\cdot}\text{P}$ as the public key of the TA.
4. TA selects $E_\pi(\cdot)/D_\pi(\cdot)$ and some hash functions: $h_1 : G \rightarrow Z_q, h_2 : \{0,1\}^* \rightarrow Z_q$, $H_{1_{key}}(\cdot) : \{0,1\}^* \rightarrow \{0,1\}^l$, $H_2(\cdot) : \{0,1\}^* \rightarrow \Gamma$, $H_3(\cdot) : \{0,1\}^* \rightarrow \{0,1\}^{l'}$, where $H_{1_{key}}(\cdot)$ as a keyed hash.
5. The system parameters are $\psi = (\text{p}, \text{q}, \text{a}, \text{b}, \text{P}, P_{pub}, h_1, h_2, H_{1_{key}}(\cdot), H_2(\cdot), H_3(\cdot), E_\pi(.)/D_\pi(.))$, putting the system parameters ψ into the vehicles and RSUs in advance.

4.2. 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 a 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, and then the RSU broadcasts the $cert_{R_j}$ within its own range.

4.3. Vehicle Setup Phase

In this phase, when the vehicle joins the range of the IoV, 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 2 March 2017–3 April 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.

4.4. Member Key Generation Phase

In this phase, a vehicle can obtain the member secrets and the corresponding valid period from the nearest RSU. This process among the vehicle, the RSU and the TA should be confidential. When a vehicle enters the communication range of RSU, the vehicle will receive the certification from RSU, and send the request of acquiring the member secrets and the corresponding valid period of RSU. After the RSU authenticates the identity of the vehicle, it sends the member secret and the valid period to the vehicle. The details are as the following steps:

1. When the vehicle enters the communication range of RSU, the vehicle will receive the certification from RSU and first check the validity of the $cert_{R_j}$ that has the format $(ID_{R_j}, (PK_{R_{j1}}, PK_{R_{j2}}, sig_j))$, where sig_j is a signature on $(ID_{R_j}, (PK_{R_{j1}}, PK_{R_{j2}}))$ issued by the TA. If the certification is valid under the public key of the system, extract the public key and identity of RSU from the certification $cert_{R_j}$.
2. 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, and π_{i1},π_{i2} are used as the keys of the symmetric encryption scheme $(E_\pi(.)/D_\pi(.))$. Finally, the vehicle computes $p_j = E_{\pi_{i2}}(\lambda_i, T_i)$ and sends $\text{s} = \left(\text{f}, ID_{R_j}, p_j, T_i\right)$ to RSU.
3. The RSU receives s from vehicle, if T_i is invalid, then it aborts; otherwise, it sends s to the TA through the secure channel. When the TA receives s and first computes $\pi_{i2} = H_2\left(\text{f}, P_{pub}, \text{sf}, ID_{R_j}, T_i\right)$, $D_{\pi_{i2}}(p_j)$ to get (λ_i', T_i'). If the equation $\lambda_i' \neq \lambda_i$ does not appear in a tuple of the 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, the TA authenticates the vehicle and sends the authenticated message to RSU.

4. When 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 $h_{R_j} = H1_{\pi_{i1}}(\beta_j, \gamma_j, \tau_p)$, and $p_j' = E_{\pi_{i1}}\left(\beta_j, \gamma_j, \tau_p, h_{R_j}\right)$; and sends t $= (H_3\,(\mathrm{f})\,, p_j')$ to the vehicle.
5. 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 $h_{R_j} = H_{1_{\pi_{i1}}}(\beta_j, \gamma_j, \tau_p)$ 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.

4.5. Vehicle Signature Phase

In this phase, when a vehicle obtains the member secret (β_j, γ_j) from the RSU and the validity period of member secret is within the authorized period, the vehicle will generate the external pseudonym identity of the vehicle and the digital signature of the message. Finally, a vehicle broadcasts the external pseudonym identity, the message as well as the digital signature to other vehicles and the RSU. The details are as the following steps:

1. Vehicle computes the external pseudonym identity $PPID_i = H_3\left(IPID_{V_i}, T_i\right)$ and the one time signature key $sk_i = \left(\beta_j \cdot \gamma_j\right) \cdot h_1\left(PPID_i\right) modn$.
2. The vehicle chooses a random number $r_i \in Z_q^*$, and computes $R_i = r_i \cdot P$, $\beta_i = h_2\left(PPID_i\,||R_i||\,M_i\right)$, $S_i = sk_i + \beta_i \cdot r_i$. Then the vehicle sends $(M_i, PPID_i, R_i, S_i)$ to nearby vehicles and RSUs.
3. 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 $\beta_j = r \cdot \beta_j$, $\gamma_j = r \cdot \gamma_j$ as the new member secret.

4.6. Message Verification Phase

This phase allows the verifier to check the validity of the received message without bilinear pairing, which greatly reduces the computation overhead. Moreover, our scheme can support the batch verification function, which can verify many messages at the same time to improve performance. Then, we will show the details of verifying a single message and many messages.

- **Single message verification**: When the verifier receives the safety-related message $(M_i, PPID_i, R_i, S_i)$ broadcasted from the vehicle, it could use the system parameters ψ to verify the validity of the message. The details are as following:

  1. The verifier first checks the validity of timestamp T_i. If T_i is invalid, it aborts; otherwise, it executes the next step.
  2. The verifier checks whether the equation $S_i{\cdot}P = h_1\left(PPID_i\right){\cdot}PK_{R_{j1}} + \beta_i \cdot R_i$ holds. If it holds, the verifier receives the message; otherwise, it rejects the message.

  Due to $sk_i = (\beta_j{\cdot}\gamma_j){\cdot}h_1\left(PPID_i\right) modn$, $\beta_j{\cdot}\gamma_j = k_j$, $PK_{R_{j1}} = k_j{\cdot}P$, $R_i = r_i{\cdot}P$, $\beta_i = h_2\left(PPID_i\,||R_i||\,M_i\right)$ and $S_i = sk_i + \beta_i{\cdot}r_i$, we can check the equation through the following steps:

$$\begin{aligned} S_i \cdot P &= (sk_i + \beta_i r_i) \cdot P \\ &= \left(\left(\beta_j \cdot \gamma_j\right) \cdot h_1\left(PPID_i\right) + \beta_i \cdot r_i\right) \cdot P \\ &= \left(\beta_j \cdot \gamma_j\right) \cdot h_1\left(PPID_i\right) \cdot P + \beta_i \cdot r_i \cdot P \\ &= h_1\left(PPID_i\right) \cdot PK_{R_{j1}} + \beta_i \cdot R_i. \end{aligned} \tag{1}$$

  Hence, the correctness of the single message verification is verified.
- **Multiple messages batch verification**: We used a small index test technique during the batch verification procedure to ensure the non-repudiation of the signature. A vector, including the

small random integer, is used to detect the modification of the batch signature in the small index test technique. After receiving multiple messages $(M_1, PPID_1, R_1, S_1)$, $(M_2, PPID_2, R_2, S_2)$, ...,$(M_n, PPID_n, R_n, S_n)$, a verifier uses the system parameter to verify the validity of the many messages at the same time. The details are as the following steps:

1. 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 executes the next step.
2. Verifier chooses a random vector v = $\{v_1, v_2, \ldots, v_n\}$, where v_i is a small random integer in $\left[1, 2^t\right]$ and t is a small integer with low overhead. Then, the verifier checks the correctness of the equation $(\sum_{i=1}^{n} v_i{\cdot}S_i){\cdot}P = \sum_{i=1}^{n} (V_i \cdot h_1\,(PPID_i)){\cdot}PK_{R_{j1}} + \sum_{i=1}^{n} (v_i{\cdot}\ \beta_i{\cdot}R_i)$. If it does not hold, the verifier rejects the messages; otherwise, the verifier receives the messages.

Due to $sk_i = \left(\alpha_j \cdot \beta_j\right) \cdot h_1\,(PPID_i)\, modn$, $\beta_j \cdot \gamma_j = k_j$, $PK_{R_{j1}} = k_j \cdot P$, $R_i = r_i \cdot P$, $\beta_i = h_2\,(PPID_i\,||R_i||\,M_i)$ and $S_i = sk_i + \beta_i \cdot r_i$, we can check the equation through the following steps:

$$\begin{aligned}
\left(\sum_{i=1}^{n} v_i \cdot S_i\right) \cdot P &= (\sum_{i=1}^{n} v_i\,(sk_i + \beta_i \cdot r_i)) \cdot P \\
&= \left(\sum_{i=1}^{n} v_i\,\left(\left(\beta_j \cdot \gamma_j\right) \cdot h_1\,(PPID_i) + \beta_i \cdot r_i\right)\right) \cdot P \\
&= \sum_{i=1}^{n} (v_i \cdot \left(\left(\beta_j \cdot \gamma_j\right) \cdot h_1\,(PPID_i) \cdot P + \beta_i \cdot r_i \cdot P\right)) \\
&= \sum_{i=1}^{n} (v_i \cdot h_1\,(PPID_i) \cdot PK_{R_{j1}} + v_i \cdot \beta_i\ \cdot R_i) \\
&= \sum_{i=1}^{n} (v_i \cdot h_1\,(PPID_i)) \cdot PK_{R_{j1}} + \sum_{i=1}^{n} (v_i \cdot \beta_i \cdot R_i) \\
&= \sum_{i=1}^{n}\,(v_i \cdot h_1\,(PPID_i)) \cdot PK_{R_{j1}} + \sum_{i=1}^{n}\,(v_i \cdot \beta_i \cdot R_i)\,.
\end{aligned} \tag{2}$$

Hence, the correctness of the multiple messages verification is verified.

4.7. IPID and Authentication Key Update Phase

At this phase, the vehicle can use the online model to update the internal pseudo-identity and authentication key stored in the TPD. The details are as following:

1. 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, $\pi_i = H_2\left(g, P_{pub}, tP_{pub}, T_i\right)$, $p_i = E_{\pi_i}\,(\lambda_i, T_i)$. Then, it sends z = (g, T_i, p_i) to the TA through the nearby RSU.
2. The TA receives z and checks the validity of T_i. If T_i is invalid, it aborts; otherwise, it executes the following steps:
    - TA computes $\pi_i = H_2(g, P_{pub}, s{\cdot}g, T_i)$ and $D_{\pi_i}(p_i)$ to get $(\lambda_i^{'}, T_i^{'})$.
    - TA checks the validity of $T_i^{'}$, if $T_i^{'}$ is invalid, it aborts; otherwise, it executes the next step.
    - TA 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, it executes the next step.
    - TA checks the validity of VP_i. If it is invalid, choose a new valid period $VP_i^{'}$. TA computes $IPID_{V_i}^{'} = H_{1_\Lambda}(RID||VP_i^{'})$ and chooses a new authentication key $\widehat{\lambda}_i$; otherwise, it aborts.

- TA computes $p_i = E_{\pi_i}(IPID'_{V_i}, \widehat{\lambda}_i, T'_i, h_{TA})$. If $h_{TA} = H1_{\lambda'_i}\left(IPID'_{V_i}, \widehat{\lambda}_i, T_i\right)$ 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.

3. After a vehicle receives $(H_3\left(g\right), p'_i)$, it first computes $D_{\pi_i}\left(p'_i\right)$ to get $\left(IPID'_{V_i}, \lambda'_i, T'_i, h_{TA}{}'\right)$. Then, the vehicle 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.

5. Security Proof and Analysis

In this section, because it is difficult to address the computational Elliptic Curve Discrete Logarithm (ECDL) problem, we prove that the proposed identity-based scheme has the feature of non-forgery. In addition, we also show that our scheme can satisfy the security requirement and illustrate the differences between our scheme and others.

5.1. Security Analysis

In this sub-section, we will show that an 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.

Definition 1. *Since it is difficult to address the computational Elliptic Curve Discrete Logarithm (ECDL) problem, the proposed scheme is security existential forgery under the randomly selected message attack in the random oracle model. The proof is as follows.*

Theorem 1. *Our scheme for IoV is secure in the random oracle.*

Assuming there is an adversary that could forge message $(M_i, PPID_i, R_i, S_i)$, then we construct a challenger C, which could solve 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 $\psi = (\mathrm{p}, \mathrm{q}, \mathrm{a}, \mathrm{b}, \mathrm{P}, P_{pub}, h_1, h_2\ , H1_{key}\left(\cdot\right), H_2\left(\cdot\right), H_3\left(\cdot\right), E_\pi\left(.\right)/D_\pi\left(.\right))$ to an adversary A.

$h_1 - oracle$: Challenger C first initializes the list L_{h_1} with the form of $\left(\left\langle PPID_i, \tau_{h_1}\right\rangle\right)$. 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 $\left\langle PPID_i, \tau_{h_1}\right\rangle$ into the L_{h_1}, finally sending 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 $L_{h_2}\ \left(\left\langle PPID_i, R_i, \mathrm{M}_i, \tau_{h_2}\right\rangle\right)$. 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 $\langle PPID_i, R_i, \mathrm{M}_i\rangle$ for whether it appears in the list L_{h_2}. If the tuple exists in in the list L_{h_2}, then sends $\tau_{h_2} = h_2\left(PPID_i\,||R_i||\,\mathrm{M}_i\right)$ to the adversary A; otherwise, C chooses a random number $\tau_{h_2} \in Z_q^*$ and sets the tuple $\left(\left\langle PPID_i, R_i, \mathrm{M}_i, \tau_{h_2}\right\rangle\right)$ into the L_{h_2}, and 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 an 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 the adversary A, and it is easy to verify that the equation $S_i{\cdot}P = h_1\left(PPID_i\right){\cdot}PK_{R_{j1}} + \beta_i{\cdot}R_i$ holds. Therefore, the message and signature $(M_i, PPID_i, R_i, S_i)$, which A acquired from the inquiry from C, are valid.

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

$$S_i \cdot P = h_1\left(PPID_i\right) \cdot PK_{R_{j1}} + \beta_i \cdot R_i. \tag{3}$$

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:

$$S_i^{'} \cdot P = (h_1 (PPID_i))^{'} \cdot PK_{R_{j1}} + \beta_i \cdot R_i. \tag{4}$$

According to Equations (3) and (4), we could get

$$\begin{aligned} \left(S_i - S_i^{'}\right) \cdot P &= S_i \cdot P - S_i^{'} \cdot P \\ &= h_1 (PPID_i) \cdot PK_{R_{j1}} + \beta_i \ \cdot R_i \\ &- \left((h_1 (PPID_i))^{'} \cdot PK_{R_{j1}} + \beta_i \cdot R_i\right) \\ &= \left(h_1 (PPID_i) - (h_1 (PPID_i))^{'}\right) \cdot PK_{R_{j1}} \\ &= \left(h_1 (PPID_i) - (h_1 (PPID_i))^{'}\right) \cdot k_j \cdot P \end{aligned} \tag{5}$$

and

$$\left(S_i - S_i{}^{'}\right) = \left(h_1 (PPID_i) - (h_1 (PPID_i))^{'}\right) \cdot k_j. \tag{6}$$

Therefore, C outputs the $\left(h_1 (PPID_i) - (h_1 (PPID_i))^{'}\right)^{-1} \cdot \left(S_i - S_i^{'}\right)$. However, it is difficult to address the computational Elliptic Curve Discrete Logarithm (ECDL) problem, and the security of the proposed scheme is secure against forgery under the randomly selected message attack in the random oracle model.

1. **Message integrity**: According to Theorem 1, because it is difficult to solve 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 $S_i \cdot P = h_1 (PPID_i) \cdot PK_{R_{j1}} + \beta_i \cdot R_i$ 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**: Since it is difficult to address the computational Elliptic Curve Discrete Logarithm (ECDL) problem, 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. **Traceability**: During this stage, when the adversary sends false messages which cause damage, the TA can trace the real identity of the corresponding message. Assuming the public pseudonym identity of the vehicle is $PPID_i$, the TA can extract the timestamp from the message M_i, which can find the valid period VP_i of the internal pseudo-identity of vehicle. Then, the TA can verify whether the equation $H_3 \left(IPID_{V_i}, T_i\right) = PPID_i$ holds, where the $IPID_{V_i}$ is in the tuple of member list $\left(\text{RID}, VP_i, IPID_{V_i}, \lambda_i\right)$. If it holds, the TA outputs the real identity of vehicle.
4. **Non-repudiation**: Once the TA traces the real identity of false message, the message sender could not deny that he has sent this false message. To achieve this goal, in our scheme, we use a random vector v $= \{v_1, v_2, \ldots, v_n\}$ to ensure an attacker cannot deny its signature in a message sent by exchanging signatures among several different messages.
5. **Resistance side channel attack**: In this paper, we use the more realistic TPD to resist side channel attack. There are three types of related information (IPID, authentication key, and member secret) stored in the TPD of our scheme. Due to the first type of secret often being 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 the vehicle in the nearby RSU can be influenced. Furthermore, because the RSU can periodically update its public-private key pairs, the attacker could not acquire enough information through the side channel to resume the member key stored in the TPD.

5.2. Security Comparison

In this sub-section, we compare the proposed scheme with other existing schemes in terms of satisfactory security requirements. The comparison results are summarized in Table 2, where sr1, sr2, sr3, sr4, sr5 denote the message integrity, non-forgery, traceability, non-repudiation and resistance side channel attack, respectively.

Table 2. The security comparisons of each scheme.

	sr1	sr2	sr3	sr4	sr5
Zhang et al. [18]	√	×	√	√	×
Bayat et al. [19]	√	×	√	√	×
Zhang et al. [21]	√	√	√	√	√
He et al. [20]	√	√	√	√	×
Our Scheme	√	√	√	√	√

As shown in Table 2, we can conclude that the schemes of Zhang et al. [18], Bayat et al. [19], and He et al. [20] could not satisfy all five of the security requirements. However, our scheme could satisfy all security requirements.

6. Performance Analysis and Comparison

In this section, we will analyze the proposed identity-based scheme compared with other related schemes in terms of the computation overhead and communication overhead.

6.1. Computation Overhead Analysis

In the scheme-based bilinear pairing proposed by Zhang et al. [18] and Bayat et al. [19], the order q of group G in the bilinear pairing $e : \mathrm{G} \times \mathrm{G} \to G_T$, generated by the Elliptic Curve $y^2 = x^3 + x \ mod$ n to achieve the security level of 80 bits, where n is the 512-bit prime number and the order q of the group G is a 160-bit prime number. However, among the schemes based on the Elliptic Curve, such as the scheme proposed by He et al. [20], the order q of group G is generated by the Elliptic Curve $y^2 = x^3 + ax + b \ mod \ n$ to achieve the same security level compared with the scheme based on the bilinear pairing, where n and q are the 160-bit prime numbers. For convenience, some time-consuming cryptographic operations [22] are defined as follows: T_p denotes the execution time of the bilinear pairing operation; T_{mp-p} denotes the execution time of the small scale multiplication operation; T_{mtp} denotes the execution time of a Map-to-Point operation; and T_{mp-ECC} denotes the execution time of the small scale multiplication operation based on the Elliptic Curve. Table 3 lists the execution time required for these operations [20].

The computation overhead of our scheme can be compared with other schemes in three aspects, letting PSGH and SMVH and MMVH denote the pseudonym and signature generation phase, signal message verification phase and multiple messages verification phase, respectively. Details are only shown in Zhang et al.'s scheme [21] and our scheme. The other schemes can be analyzed by the same method. Table 4 lists the computation overhead of our scheme compared with the schemes of Zhang et al. [18], Bayat et al. [19], Zhang et al. [21] and He et al. [20].

Table 3. Different execution time of each cryptographic operations.

Cryptographic Operation	Execution Time
T_p	4.211 ms
T_{mp-p}	1.709 ms
T_{mtp}	4.406 ms
T_{mp-ECC}	0.442 ms

Table 4. The computation overhead of each scheme.

Scheme	PSGH	SMVH	MMVH
Zhang et al. [18]	$6T_{mp-tp}$ $+T_{mtp}\approx$ 14.66 ms	$3T_{bp}+$ $2T_{mp-bp}$ $\approx$16.051 ms	$(n+1)\,T_{mp-bp}+$ $3T_{bp}\approx 12.633+$ 1.709 (n + 1) ms
Bayat et al. [19]	$5T_{mp-bp}$ $+T_{mtp}\approx$ 12.951 ms	$3T_{bp}+T_{mtp}$ $+T_{mp-bp}\approx$ 18.748 ms	$3T_{bp}+nT_{mp-bp}$ $+nT_{mtp}\approx 6.115n$ $+12.633$ ms
Zhang et al. [21]	$2T_{mtp}\approx$ 8.812 ms	$2T_{bp}+2T_{mtp}$ $\approx$17.234 ms	$2T_{bp}+2nT_{mtp}\approx$ $8.812n+8.422$ ms
He et al. [20]	$3T_{mp-ECC}$ $\approx$ 1.326 ms	$3T_{mp-ECC}$ $\approx$ 1.326 ms	$(n+2)\,T_{mp-ECC}$ $\approx 0.442\,(n+2)$ ms
Our Scheme	T_{mp-ECC} $\approx$ 0.442 ms	$3T_{mp-ECC}$ $\approx$ 1.326 ms	$(n+2)\,T_{mp-ECC}$ $\approx 0.442\,(n+2)$ ms

In the pseudonym and signature generation phase of Zhang et al's scheme [21], which needs to execute the two Map-to-Point operations, the whole computation time of this phase is $2T_{mtp} = 8.812$ ms. During the signal message verification phase, there are two bilinear pairing operations and two Map-to-Point operations that need to be executed. Thus, the whole computation time of this phase is $2T_p + 2T_{mtp} = 17.234$ ms. In the multiple messages verification phase, there are two bilinear pairing operations and $2n$ Map-to-Point operations that need to be executed. Thus, the whole computation time of this phase is $2\mathrm{T}_p + 2T_{mtp} = 8.812n + 8.422$ ms.

In the pseudonym and signature generation phase of our scheme, which needs to execute a small scale multiplication operation based on the Elliptic Curve, the whole computation time of this phase is $T_{mp-ECC} = 0.442$ ms. During the signal message verification phase, there are three small scale multiplication operations based on the Elliptic Curve that need to be executed. Thus, the whole computation time of this phase is $3T_{mp-ECC} = 1.326$ ms. In the multiple message verification phase, there are $(n+2)$ small scale multiplication operation based on the Elliptic Curve that need to be executed. Thus, the whole computation time of this phase is $(\mathrm{n}+2)\,T_{mp-ECC} = 0.442\,(n+2)$ ms. Figure 3 shows the computation overhead to sign and verify the single message in each scheme. Compared with the schemes of Bayat et al. [19] and Zhang et al. [21] using the bilinear pairing, our scheme's computation is lower. At the same time, our scheme is also lower than the scheme of He et al. [20] in terms of computation. Figure 4 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 IoV.

6.2. Communication Overhead Analysis

In this section, the communication overheads of our scheme compared with other schemes will be analyzed. In the group G_1 based on bilinear mapping, the size of the elements in G_1 is 64 × 2 = 128 byte [23]. However, in the group G based on the Elliptic Curve, the size of the elements in G_1 is 20 × 2 = 40 byte. Furthermore, we assume that the size of result of the general hash function is 20 bytes and the size of the timestamp is 4 bytes [24]. In addition, we do not consider the size of the message that is transmitted by the vehicle in this phase [25]. Table 5 lists the computation overheads of our scheme compared with other schemes.

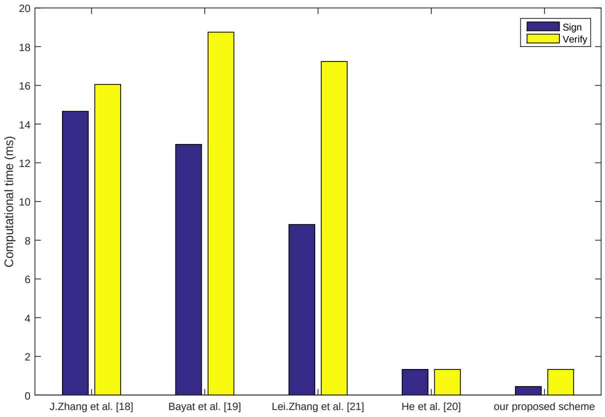

Figure 3. Computation overhead comparison of signing and verifying a single message.

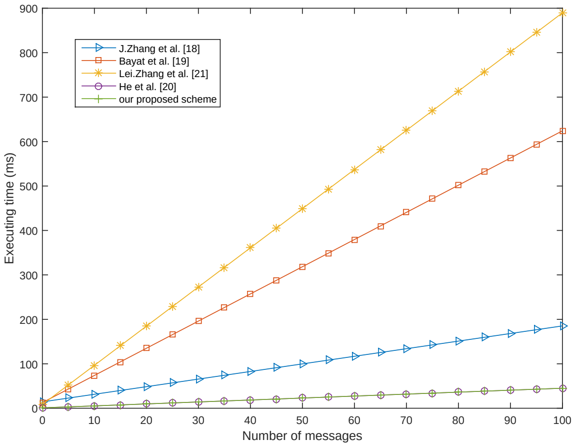

Figure 4. Computation overhead comparison of verifying multiple message.

Table 5. The communication overhead of each scheme.

Scheme	Sending a Single Message	Sending n Messages
Zhang et al. [18]	388 bytes	388 n bytes
Bayat et al. [19]	388 bytes	388 n bytes
Zhang et al. [21]	148 bytes	148 n bytes
He et al. [20]	144 bytes	144 n bytes
Our Scheme	80 bytes	80 n bytes

In Table 5, the communication message is $\{AID_i, M_i, S_i, T_i\}$ in Zhang et al.'s scheme [18], where $AID_i = \{AID_1, AID_2\}$, $AID_1, AID_2, S_i \in G_1$, hence the communication overhead of sending single message is 128 × 3 + 4 = 388 bytes. When multiple messages are broadcasted, which needs n pseudonym, signature and timestamp, the total communication overhead of sending multiple messages is 388n bytes. In addition, in He et al.'s scheme [20], due to the AID_1, $AID_2, R_i \in G, \sigma_i \in Z_q^*$, T_i is a timestamp, and the communication overhead of sending a single message and multiple messages are 40 × 3 + 20 + 4 = 144 byte and 144 n bytes, respectively. The other scheme's communication overhead can be concluded by the same method. In our scheme, the communication messages include

$PPID_i \in Z_q^*$, $R_i \in G$, $S_i \in Z_q^*$, whose overhead is 20 × 2 + 40 = 80 byte. The communication overhead of sending multiple messages is 80n bytes.

7. Conclusions

In this paper, we propose a privacy-preserving authentication scheme using a double pseudonym that supports both Vehicle to Vehicle (V2V) communication and Vehicle to Infrastructure (V2I) communication in IoV. Firstly, unlike other schemes, which store the system master secret (that cannot be updated) in the TPD, in our scheme, the information stored in the TPD is 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 IoV. Furthermore, performance analysis and comparison show that our scheme is better than other schemes in terms of computation overhead and communication overhead. This shows that our scheme is more suitable used in the IoV.

As for future work, we will pay more attention to addressing the extreme environment in which the system suffers Denial of Service (DoS) attack during the messages broadcast. Since the Dos attack is hard to defend and causes huge damage in the batch verification, addressing the DoS attack has become an urgent task in future research.

Author Contributions: This paper is completed by all authors. W.X. is responsible for proposing the idea of the paper and J.C. checks whether the idea is feasible and give some suggestions to improve this idea. J.Z. is responsible for writing this paper and the security analysis is completed by H.Z. Y.X. is responsible for the performance analysis and comparison. Finally, the language of the paper is improved by L.L.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grant 61572001, Grant 61502008, and Grant 61702005, in part by the Key Program of National Natural Science Foundation of China under Grant U1405255, in part by the Natural Science Foundation of Anhui Province under Grant 1708085QF136, and in part by the Excellent Talent Project of Anhui University.

Acknowledgments: The authors thank the Associate Editor and the anonymous reviewers for their useful comments and suggestions which helped us improve the quality and presentation of this paper.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Sha, K.; Xi, Y.; Shi, W.; Schwiebert, L.; Zhang, T. Adaptive privacypreserving authentication in vehicular networks. In Proceedings of the 1st International Conference on Communications and Networking in China, Beijing, China, 25–27 October 2006; pp. 1–8.
2. Xi, Y.; Sha, K.; Shi, W.; Schwiebert, L.; Zhang, T. Enforcing privacy using symmetric random key-set in vehicularnetworks. In Proceedings of the Eighth International Symposium on Autonomous Decentralized Systems, Sedona, AZ, USA, 21–23 March 2007; pp. 344–351.
3. Qu, F.; Wu, Z.; Wang, F.-Y.; Cho, W. A security and privacy review of vanets. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 2985–2996. [CrossRef]
4. Cheng, J.J.; Cheng, J.L.; Zhou, M.C.; Liu, F.Q.; Gao, S.C.; Liu, C. Routing in internet of vehicles: A review. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 2339–2352. [CrossRef]
5. Zeadally, S.; Hunt, R.; Chen, Y.-S.; Irwin, A.; Hassan, A. Vehicular ad hoc networks (vanets): Status, results, challenges. *Telecommun. Syst.* **2012**, *50*, 217–241. [CrossRef]
6. Toor, Y.; Muhlethaler, P.; Laouiti, A. Vehicle ad hoc networks: Applications and related technical issues. *IEEE Commun. Surv. Tutor.* **2008**, *10*, 3. [CrossRef]
7. Wang, F.-Y.; Zeng, D.; Yang, L. Smart cars on smart roads: An ieee intelligent transportation systems society update. *IEEE Pervasive Comput.* **2006**, *5*, 68–69. [CrossRef]
8. Oh, H.; Yae, C.; Ahn, D.; Cho, H. 5.8 ghz dsrc packet communication system for its services. In Proceedings of the Vehicular Technology Conference, Amsterdam, The Netherlands, 19–22 Septemper 1999; Volume 4, pp. 2223–2227.
9. Chim, T.W.; Yiu, S.-M.; Hui, L.C.; Li, V.O. Specs: Secure and privacy enhancing communications schemes for vanets. *Ad Hoc Netw.* **2011**, *9*, 189–203. [CrossRef]

10. Internet X.509 Public Key Infrastructure: Logotypes in X.509 Certificates. Available online: https://tools.ietf.org/html/rfc3709 (accessed on 14 March 2018).
11. Gamage, C.; Gras, B.; Crispo, B.; Tanenbaum, A.S. An identitybased ring signature scheme with enhanced privacy. In Proceedings of the Securecomm and Workshops, Baltimore, MD, USA, 28 August–1 September 2006; pp. 1–5.
12. Raya, M.; Hubaux, J.-P. Securing vehicular ad hoc networks. *J. Comput. Secur.* **2007**, *15*, 39–68. [CrossRef]
13. Lin, X.; Lu, R.; Zhang, C.; Zhu, H.; Ho, P.-H.; Shen, X. Security in vehicular ad hoc networks. *IEEE Commun. Mag.* **2008**, *46*, 4.
14. Lin, X.; Sun, X.; Ho, P.-H.; Shen, X. Gsis: A secure and privacypreserving protocol for vehicular communications. *IEEE Trans. Veh. Technol.* **2007**, *56*, 3442–3456.
15. Studer, A.; Shi, E.; Bai, F.; Perrig, A. Tacking together efficient authentication, revocation, privacy in vanets. In Proceedings of the 6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks. Rome, Italy, 22–26 June 2009; pp. 1–9.
16. Zhang, C.; Lu, R.; Lin, X.; Ho, P.-H.; Shen, X. An efficient identitybased batch verification scheme for vehicular sensor networks. In Proceedings of the 27th Conference on Computer Communications. Phoenix, AZ, USA, 13–18 April 2008; pp. 246–250.
17. Lee, C.-C.; Lai, Y.-M. Toward a secure batch verification with group testing for vanet. *Wirel. Netw.* **2013**, *19*, 1441. [CrossRef]
18. Jianhong, Z.; Min, X.; Liying, L. On the security of a secure batch verification with group testing for vanet. *Int. J. Netw. Secur.* **2014**, *16*, 351–358.
19. Bayat, M.; Barmshoory, M.; Rahimi, M.; Aref, M.R. A secure authentication scheme for vanets with batch verification. *Wirel. Netw.* **2015**, *21*, 1733. [CrossRef]
20. He, D.; Zeadally, S.; Xu, B.; Huang, X. An efficient identitybased conditional privacy-preserving authentication scheme for vehicular ad hoc networks. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 2681–2691. [CrossRef]
21. Zhang, L.; Wu, Q.; Domingo-Ferrer, J.; Qin, B.; Hu, C. Distributed aggregate privacy-preserving authentication in vanets. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 516–526. [CrossRef]
22. 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.* **2016**, *5*, 1–14. [CrossRef]
23. Identity-Based Cryptography Standard (Ibcs)#1: Supersingular Curve Implementations of the bf and bb1 Cryptosystems. Technical Report. 2007. Available online: https://www.rfc-editor.org/rfc/rfc5091.txt (accessed on 14 March 2018).
24. 509 Public Key Infrastructure Time Stamp Protocol (TSP)[J]. 2001. Available online: https://tools.ietf.org/html/rfc3161 (accessed on 14 March 2018).
25. Lo, N.-W.; Tsai, J.-L. An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks without pairings. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 1319–1328. [CrossRef]