Conditional Privacy-Preserving Authentication Using Registration List in Vehicular Ad Hoc Networks

Hong Zhong, Bo Huang, Jie Cui, Yan Xu and Lu Liu

Abstract—Vehicular ad hoc networks (VANETs) have increased in popularity in recent years and play an extremely important role in the intelligent transportation field. However, the demands of larger communication networks and the integrated message verification process for ensuring security incur more communication and computation overheads, and directly affect the efficiency of existing VANET schemes. To address this issue, this paper proposes a novel and practical conditional privacy-preserving authentication (CPPA) scheme, which uses the registration list instead of the revocation list to reduce the communication overhead. Specifically, our scheme can prevent malicious vehicles from disrupting the security features of VANETs. Moreover, we do not use the bilinear pairing operation, which is the most complicated operation in modern cryptography, thus significantly reducing the computation overhead and communication overhead. Security and performance analyses demonstrate that our proposed scheme is more secure and efficient than current schemes, and that the proposed scheme is more suitable for VANET deployments.

Index Terms—Registration list; Resistance to continuous disruption; Modification of passwords; Conditional privacy; Elliptic curve.

I. INTRODUCTION

A vehicular ad hoc network (VANET) is a special type of mobile ad hoc network that enables communication through multi-hops among the nodes in a wireless network. VANETs, which are increasing in popularity, can provide real-time information exchange by establishing connections among vehicles on roads and road-side infrastructures [1]. VANETs are efficient in providing travel assistance and can effectively reduce the driver’s workload through automatic and intelligent driving controls, and facilitate higher comfort and rich travel experience for passengers. VANETs can also provide personal entertainment and in-car office functions by facilitating Internet access to drivers and passengers [2][3]. In short, VANETs extend the functionalities and facilities of vehicles as mobile information platforms rather than simple transport entities, which significantly enrich the functions and applications of vehicle systems.

The structure of VANETs generally consists of three modules, namely a Trusted Authority (TA), a Road Side Unit (RSU), and an On-Board Unit (OBU) [4]. TA is a trusted third party with high computing power and large storage capacity that is responsible for generating the system parameters and distributing secret materials. RSU is a communication node deployed as a road-side infrastructure that can communicate with vehicles on a section of a road through wireless channels. OBU, which is placed in vehicles, is responsible for issuing and receiving traffic-related messages so that drivers can obtain a better driving experience. In addition, a Tamper-Proof Device (TPD) is used in some VANETs, which is equated to a black box that prevents access to attackers. However, manufacturing overheads of TPD are relatively expensive, so it has not generally been deployed on VANETs on a large scale.

However, security remains one of the prevailing concerns in VANET applications. Adversaries can control communication with ease because communication is often enabled through a wireless channel. There are two immediate implications; first, it is fairly straightforward for attackers to issue illegal messages in order to affect traffic patterns, or to forge malicious information for the purpose of causing traffic accidents. Second, attackers can easily track a vehicle, thus violating user privacy. These two malicious behaviors have had serious impacts upon the efficiency of VANETs in the recent past and have threatened the safety of passenger lives and property in some cases [5].

Also, VANET requires that the sender of the message needs to be tracked quickly[6], and at the same time it should has the ability to disqualify malicious vehicles from sending messages quickly. Nowadays, many people make the same password in different agencies for convenience, for example, the bank card password and the vehicle password are the same. Then the owner will not convenient tell the password to other people directly while many drivers share a vehicle. At this point, the function of modification of the password quickly and convenient is very important and necessary.

In summary, VANETs require further study to enhance security features, robustness, and reliability. To this end, this paper proposes a novel and practical conditional privacy-preserving authentication (CPPA) scheme, which uses the registration list instead of the revocation list for reducing the communication overhead. Important contributions of this paper include the following:

1) Reduction in the retrieval time of the revocation list when it is represented by the registration list in order to reduce the think time available to attackers. Moreover, the proposed scheme can prevent the attacker from con-
The proposed scheme allows the owner of a vehicle to modify passwords anytime, anywhere, which provides more flexibility and privileges to the VANET user.

3) The proposed scheme does not use bilinear pairing, which is the most complicated operation in modern cryptography, and additionally reduces the message length to minimize both the computation and communication overheads in VANETs.

The remainder of the paper is organized as follows. Section II introduces the related research of CPPA schemes in VANETs. The background knowledge and system model are introduced in Section III. Section IV describes our proposed scheme in detail and Section V presents the security analysis of the proposed scheme. Section VI presents the performance evaluation, including calculation overhead and communication overhead comparisons. Finally, Section VII discusses the conclusion and future research.

II. RELATED WORK

A wide range of research has focused on enhancing the safety and efficiencies of VANETs in the recent past. Raya et al. [7] proposed a CPPA scheme based on the public key infrastructure (PKI), which used public/private key pairs and corresponding certificates to hide a vehicle’s true identity. However, there are two obvious shortcomings: first, the OBU of each vehicle requires large storage space to save the public/private key pairs and the corresponding certificates; second, the TA must carry out a complete traversal in its storage space while seeking the true identity of the attacker, thus resulting in larger time and memory overheads.

C. Zhang et al. [8] highlighted that the computation power of the OBU in vehicles is not capable of performing complex computational operations within a short time when the number of vehicles in the VANET is relatively large. Allowing the nearby RSU to verify the message can assist the OBU in the computation, but a more effective method of reducing the OBUs computation and communication overheads in VANETs is an urgent problem.

Wu et al. [9] and L. Zhang et al. [10] proposed a CPPA scheme based on a group signature in which the OBU no longer needs to store more private data and the TA can effectively track the true identity of an attacker based on the revocation list without incurring the overheads caused by the retrieval of the revocation list. However, because the speed of the vehicle is fast and network topology changes quickly as the vehicle progresses, it is difficult to update and select the group managers and group members dynamically.

Chim et al. [11] proposed a scheme using a software-based bilinear pairing operation in which the RSU uses a pseudo identity to protect its true identity during message communication by establishing a shared key in the handshaking phase between the RSU and the TA, where the TA can also track the true identity behind the pseudo identity. In the certification phase, the RSU issues a notification message with a Bloom filter to reduce the OBUs computation overhead. But Horng et al. [12] later pointed out that the scheme proposed by Chim et al. [11] cannot resist an impersonation attack; that is, malicious vehicles can disguise themselves as legitimate vehicles to send a malicious message after intercepting a legal message.

K. A. Shim. [13] proposed a security ID-based CPPA scheme in which the RSU supports batch authentication of messages to reduce the computation overhead of the RSU when the number of messages is large. However, the TA must consume more time in retrieving the entire revocation list, and furthermore it does not address the additional authentication overheads caused by illegal information.

J. Zhang et al. [14] proposed another ID-based CPPA scheme to optimize the computation overheads in the message signature and authentication process, while the scheme also supports batch authentication in order to improve the efficiency of identity authentication. However, Lee et al. [15] later pointed out that this scheme cannot achieve the function of non-repudiation, and Liu et al. [16] pointed out that the scheme cannot resist a modification attack.

In order to improve the communication efficiency while ensuring the conditional privacy protection of the vehicles in VANETs, He et al. [17] proposed an efficient and fast signature scheme without using the bilinear pairing operation. This scheme reduces the computation overhead significantly while meeting security requirements.

Zhong et al. [18] proposed a CPPA scheme to optimize the computation process and to reduce the computation overhead based on the scheme proposed by He et al. [17]. However, the schemes proposed by He et al. [17] and Zhong et al. [18] include several security assumptions, as it is difficult to equip each vehicle with a TPD in practice. During an attack, TA can track the true identity of the attacker but cannot prevent it from sending additional malicious messages.

In order to solve the above problems, this paper proposes a novel and practical conditional privacy protection scheme based on the scheme of He et al. [19], which improves communication efficiency under the premise of reducing the demands in the security hypothesis. Additionally, our scheme can effectively prevent the attacker from continually sending malicious information because of the presence of the registration list, which improves the security features of VANETs.

III. BACKGROUND

In this section, we introduce the system model of our scheme and a background on the security requirements in VANETs.

A. System Model

The structure of VANETs consists of three parts in general: Trusted Authority (TA), Road Side Unit (RSU) and On Board Unit (OBU), as showed in Fig. 1.

Information among OBUs or between OBU and RSU is transmitted over the wireless channel, and information between TA and RSU is transmitted over the wired channel [20].

TA, a trusted third party with very high computing power and storage capacity, is responsible for generating the system parameters and distributing the secret material. It is also
responsible for the offline registration of OBU and RSU and stores the registration list of RSU and OBU.

RSU, a trusted roadside node with high computing power and storage capacity, verifies the validity and integrity of the message and then broadcasts the relevant message to the surrounding vehicles by the notification message. It can also identify the real identity of the attacker if necessary, and then notify the real identity to the TA. Because the number of RSU is less than OBU, and RSU is easier to be maintained than the OBU in VANETs, RSUs are more equipped with TPD than OBU in practice.

OBU is a semi-trusted computing unit with lower computing power and storage capacity load on the vehicle. It is responsible for calculating and issuing traffic-related messages and storage capacity load on the vehicle. It is responsible for the offline registration of OBU and stores the registration list of RSU and OBU.

RSU, a trusted roadside node with high computing power and storage capacity, verifies the validity and integrity of the message and then broadcasts the relevant message to the surrounding vehicles by the notification message. It can also identify the real identity of the attacker if necessary, and then notify the real identity to the TA. Because the number of RSU is less than OBU, and RSU is easier to be maintained than the OBU in VANETs, RSUs are more equipped with TPD than OBU in practice.

OBU is a semi-trusted computing unit with lower computing power and storage capacity load on the vehicle. It is responsible for calculating and issuing traffic-related messages and receiving notification messages from the RSU.

System assumptions in our scheme are as follows:
1) TA is completely trustworthy and will not be compromised by attackers in anytime.
2) The time in various parts of the entire VANET is synchronized.
3) RSU’s computing power and storage capacity are lower than TA and higher than OBUs.

B. Security Requirements

1) Identity privacy preservation: The attacker should not be able to obtain the real identity of the vehicles through the messages sent by the vehicles. Only the TA can track the real identity of the sender of a given message.
2) Traceability: The TA should be able to track the real identity of the attacker through malicious messages and counteract if necessary.
3) Non-repudiation: When the vehicle sends a message, it cannot deny it.
4) Un-linkability: The attacker should not be able to determine whether the messages are issued by the same vehicle through the message content.
5) Resistant to continuous disruption: As a kind of real-time network, VANETs should not only be able to trace the real identity when the attacker appears, but should also possess the ability to cease the continuous malicious behaviors.
6) Modification of passwords: The owner of a vehicle should be able to modify the passwords anywhere.
7) Resistance to ordinary attacks: The CPPA scheme in VANETs should have the ability to resist some ordinary attacks, such as replay attack, modification attack and impersonation attack.

C. Preliminary Knowledge

1) The one-way hash function \( h(\cdot) \) is said to be secure if the following properties are satisfied [21]:
   - \( h \) can take a message of arbitrary length as input and produces a message digest of a fixed-length output;
   - Given \( x \), it is easy to compute \( y = h(x) \). However, given \( y \), it is hard to compute \( x = h^{-1}(y) \);
   - Given \( x \), it is computationally infeasible to find \( x' \neq x \) that \( h(x') \neq h(x) \).
2) Elliptic Curve Discrete Logarithm Problem (ECDLP):
   The ECDLP problem [22] is to determine the integer \( x \), \( 0 \leq x \leq q - 1 \), such as \( Q = xP \), while two points \( P, Q \) of order \( q \) are on a given elliptic curve.
3) Computation Diffie-Hellman problem (CDH):
   The CDH problem [22] is to compute \( abP \in G \), while \( P, aP, bP \in G \) is given and \( a, b \in \mathbb{Z}_q^* \) is unknown.
4) Bloom Filter:
   A bloom filter [11] is an algorithm for representing a set \( A = a_1, a_2, a_3, \ldots, a_n \) of \( n \) elements to support membership queries. The idea is to allocate a vector \( v_n \) with \( n \) bits, initially all set to 0, and a secure hash functions \( h(\cdot) \), and used to compute the hash value of element, if the element in the set \( A \), and then the bit of the corresponding position is set to 1. To determine whether a given value \( b \) is in \( A \), we can check the bits at positions \( h(b) \). If this position is set as 1, then \( b \) is definitely in the set \( A \).

IV. THE PROPOSED SCHEME

The proposed scheme in our paper mainly includes two main phases such as the offline registration and the driving stage. Offline registration includes the initialization of TA and registration of RSUs and vehicles. The driving stage includes five phases including mutual authentication, release of traffic information, message verification, release of notification message and receiving messages. The main symbols used in our scheme and their definitions are illustrated in Table I.

A. Offline Registration

In this section, we introduce the system initialization phase. RSU and OBU get registered offline while in the factory or annual inspection. TA is responsible for the corresponding identity (ID) distribution and management.

- a. Initialization of TA

TA is a trusted third party with a high computing power and storage capacity that coordinates and controls the operation of the entire VANETs. Details of TAs Initialization are as follows:
The article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2017.2782672, IEEE Access

<table>
<thead>
<tr>
<th>Notation</th>
<th>Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>p, q</td>
<td>Two large prime numbers.</td>
</tr>
<tr>
<td>E</td>
<td>An elliptic curve defined by the equation ( y^2 = x^3 + ax + b \mod p ), where ( a, b \in \mathbb{F}_p ).</td>
</tr>
<tr>
<td>G</td>
<td>An additive group which consists of all points on the elliptic curve ( E ).</td>
</tr>
<tr>
<td>ID(<em>R), T(</em>{reg})(_R)SU</td>
<td>The real identity is associated with the location and its corresponding registration time of RSU.</td>
</tr>
<tr>
<td>ID(<em>V), T(</em>{reg})(_V)</td>
<td>The real identity and its corresponding registration time of Vehicle.</td>
</tr>
<tr>
<td>PW(_1), PW(_2)</td>
<td>Two passwords of vehicle.</td>
</tr>
<tr>
<td>PK(_RSU), SK(_RSU)</td>
<td>The public key and corresponding private key of RSU.</td>
</tr>
<tr>
<td>L(_RSU)</td>
<td>The registration lists of RSU that saved in TA.</td>
</tr>
<tr>
<td>L(_OBU)</td>
<td>The registration lists of vehicle that saved in TA.</td>
</tr>
<tr>
<td>t(_o)</td>
<td>The message list that saved in RSU.</td>
</tr>
<tr>
<td>h((\cdot))</td>
<td>A secure hash function.</td>
</tr>
</tbody>
</table>

1) TA chooses two large prime numbers \( p, q \) and an additive group \( G \) with the order \( q \) and its generator is \( P \), which consists of all points on the elliptic curve \( E \) defined by the equation \( y^2 = x^3 + ax + b \mod p \), where \( a, b \in \mathbb{F}_p \).

2) TA chooses a random number \( s \in \mathbb{Z}_q^* \) as the master private key, and computes \( P_{pub} = s \cdot P \) as the master public key.

3) TA chooses a secure hash function \( h(\cdot) \).

4) TA broadcasts the system parameter \( \{ p, q, a, b, P, P_{pub}, h(\cdot) \} \) periodicity.

- b. Registration of RSU

TA chooses the identity \( ID_{R} \) of RSU according to its location, and computes \( K_{R} = h(ID_{R}|s) \), where \( T_{reg}^{RSU} \) is the corresponding registration time. After that, TA saves \( \{ K_{R}, ID_{R} \} \) to RSU.

- c. Registration of Vehicle

TA chooses the identity \( ID_{V} \) and two passwords \( PW_{1}, PW_{2} \) and then calculates \( K_{V} = h(ID_{V}|s) \) and \( Z_{V} = K_{V} \oplus h(PW_{1}|PW_{2}) \), where \( T_{OBU} \) is the corresponding registration time, \( K_{V} \) and \( Z_{V} \) will be used in the modification of passwords if needed. After that, TA saves \( \{ ID_{V}, PW_{1}, PW_{2}, Z_{V}, T_{reg}^{OBU} \} \) to OBU and the owner of the vehicle respectively.

### B. Driving Stage

At this phase, RSU periodicity broadcasts message \( \{ ID_{R}, PK_{RSU} \} \) to all vehicles in its range, where \( PK_{RSU} \) is the public key of RSU. Whenever a given vehicle enters the range of a new RSU, OBU generates a pseudo identity and sends it to the corresponding RSU. After that, RSU sends the message related to the vehicles pseudo identity to the TA after receiving and processing the message from the vehicle. TA returns the message to the RSU after confirming the legal presence of RSU and OBU in the registration list based on the timestamp. RSU broadcasts the corresponding message after the computation process. When the vehicle receives the message and confirms its legitimacy, OBU, RSU and TA should have completed the mutual authentication process. After the authentication process, OBU can issue the traffic information with the help of RSU. The details are as follows:

- a. Mutual Authentication

1) Driver needs to input the identity and two passwords \( ID_{V}, PW_{1}, PW_{2} \) to start OBU, and then OBU will check whether \( ID_{V} \) and \( PW_{1}, PW_{2} \) are identical to the stored ones. If so, OBU computes \( K_{V} = Z_{V} \oplus h(PW_{1}|PW_{2}) \). After that, OBU chooses a random number \( x \in Z_{q}^* \) and computes \( X = x \cdot P, X^* = x \cdot P_{pub} \) and \( PID_{V} = ID_{V} \oplus h(X^*) \). Obviously, the above calculation process can be done off-line in advance.

When a vehicle enters the range of a new RSU, it computes the two signature hash equations \( \sigma_{OBU} = h(T_{1}|ID_{V}|ID_{R}|K_{V}|X|X^*) \) and \( \sigma_{check} = h(T_{OBU}^{reg}|T_{1}|X|PID_{V}|\sigma_{OBU}) \).

Finally, OBU sends \( \{ T_{OBU}^{reg}, T_{1}, X, PID_{V}, \sigma_{OBU}, \sigma_{check} \} \) to RSU.

2) Upon receiving the message \( \{ T_{OBU}^{reg}, T_{1}, X, PID_{V}, \sigma_{OBU}, \sigma_{check} \} \) sent by the vehicle, RSU checks whether the timestamp \( T_{1} \) is the latest or not. All the timestamps are tested in the following way: \( t_{1} \) is the value that the current time value minus the time value contained in the received timestamp, \( t_{2} \) is the value that the clock difference value plus the time delay value, and then judge that whether \( t_{1} \) is less than \( t_{2} \). If yes, that means it is the latest, than RSU checks whether the equation \( \sigma_{check} = h(T_{OBU}^{reg}|T_{1}|X|PID_{V}|\sigma_{OBU}) \) exists. If so, RSU chooses a random number \( y \in Z_{q}^* \) and computes \( Y = y \cdot P, Y^* = y \cdot P_{pub}, PID_{R} = ID_{R} \oplus h(Y^*) \) and \( \sigma_{RSU} = h(T_{2}|PID_{V}|X|\sigma_{OBU}|ID_{R}|K_{V}|Y|Y^*) \).

Now, RSU saves the item \( < T_{2}, X, Y, Y^* > \) to the handshaking list \( L_{rs} \) stored in TPD, and then sends \( \{ T_{RSU}^{reg}, T_{2}, Y, PID_{R}, \sigma_{RSU}, T_{OBU}^{reg}, T_{1}, X, PID_{V}, \sigma_{OBU} \} \) to TA. Data stored in TPD are periodically deleted in order to reduce its storage burden.

3) Upon receiving the message \( \{ T_{OBU}^{reg}, T_{2}, Y, PID_{R}, \sigma_{RSU}, T_{OBU}^{reg}, T_{1}, X, PID_{V}, \sigma_{OBU} \} \), TA checks whether the timestamp \( T_{2} \) is the latest. If it is the latest, TA computes \( Y^* = Y \cdot s \) and \( ID_{R} = PID_{R} \oplus h(Y^*) \).

Now TA checks whether \( ID_{R} \) is contained in the registration list \( L_{RSU} \) according to the timestamp \( T_{reg}^{RSU} \). If so, TA computes \( K_{R} = h(ID_{R}|s) \) and checks whether the equation \( \sigma_{RSU} = h(T_{2}|PID_{V}|X|\sigma_{OBU}|ID_{R}|K_{V}|Y|Y^*) \) holds. If so, TA computes \( X^* = X \cdot s \) and \( ID_{V} = PID_{V} \oplus h(X^*) \).

Third, TA checks whether \( ID_{V} \) is contained in the registration list \( L_{OBU} \) according to timestamp \( T_{reg}^{OBU} \). If yes, TA computes \( K_{V} = h(ID_{V}|s) \) and checks whether the equation \( \sigma_{OBU} = h(T_{2}|PID_{V}|X|\sigma_{OBU}|ID_{R}|K_{V}|X|X^*) \) exists. If so, TA computes \( TAID_{V} = ID_{V} \oplus h(Y|Y^*|K_{R}), \sigma_{TA-RSU} = h(T_{3}|ID_{V}|TAID_{V}|X|ID_{R}|Y|K_{R}), TAID_{R} = ID_{R} \oplus h(X|X^*|K_{V}) \) and \( ID_{V} = PID_{V} \oplus h(X^*) \).

Finally, TA sends the message \( \{ T_{3}, T_{2}, T_{OBU}^{reg}, \sigma_{TA-OBU}, \sigma_{TA-RSU}, TAID_{V}, TAID_{R} \} \) to RSU.
4) Upon receiving the message \( \{T_3, T_2, T^{reg}_{OBU}, \sigma_{TA\rightarrow OBU}, \sigma_{TA\rightarrow RSU}, T_{AIDV}, T_{AIDR} \} \), RSU checks whether the timestamp \( T_3 \) is the latest. If it is the latest, the adversary computes \( ID_V = T_{AIDV} \oplus h(y)[y^*][K_R] \) and further checks whether the equation \( \sigma_{TA\rightarrow RSU} = h(T_3)[ID_V][TAIDV][X][ID_R][Y][SK] \) exists. If so, RSU computes \( SK = y \cdot X \) and checks whether the equation \( \sigma_{RSU\rightarrow OBU} = h(T_3)[ID_V][ID_R][X][Y][SK] \) exists. If yes, OBU calculates \( SK = y \cdot Y \) and checks whether the equation \( \sigma_{RSU\rightarrow OBU} = h(T_4)[ID_V][ID_R][X][Y][SK] \) exists. At this stage, the message is issued by the RSU.

5) Upon receiving the message \( \{T_4, T_{AIDR}, \sigma_{TA\rightarrow OBU}, \sigma_{RSU\rightarrow OBU} \} \), OBU checks whether the timestamp \( T_4 \) is the latest. If it is the latest, OBU computes \( ID_R = T_{AIDR} \oplus h(X)[X^*][K_V] \) and checks whether the equation \( \sigma_{TA\rightarrow OBU} = h(ID_V)[X][X^*][ID_R][Y][SK] \) exists. If yes, OBU calculates \( SK = y \cdot Y \) and checks whether the equation \( \sigma_{RSU\rightarrow OBU} = h(T_4)[ID_V][ID_R][X][Y][SK] \) exists. By now, OBU, RSU and TA should have completed the mutual certification process, therefore, the vehicle is legal and RSU has not been compromised.

b. Release of Traffic Information

If a vehicle in travel wants to issue traffic information, OBU sends \( \{T_5, m, \sigma_m\} \) to RSU and to other vehicles, where \( \sigma_m = h(T_5)[m][ID_R][ID_V][X][Y][SK][\sigma_{TA\rightarrow OBU}] \).

c. Message Verification

Upon receiving the message \( \{T_5, m, \sigma_m\} \), RSU checks whether the timestamp \( T_5 \) is the latest. If the adversary computes \( ID_V = T_{AIDV} \oplus h(Y)[Y^*][K_R] \) and checks whether the equation \( \sigma_{TA\rightarrow RSU} = h(T_5)[ID_V][TAIDV][X][ID_R][Y][SK] \) exists. If so, RSU computes \( SK = y \cdot X \) and checks whether the equation \( \sigma_{RSU\rightarrow OBU} = h(T_5)[ID_V][ID_R][X][Y][SK] \) exists. If yes, OBU calculates \( SK = y \cdot Y \) and checks whether the equation \( \sigma_{RSU\rightarrow OBU} = h(T_4)[ID_V][ID_R][X][Y][SK] \) exists.

d. Release of Notification Message

At this stage, the notification message is issued by the RSU, consisting of the bloom filters (a positive filter and a negative filter). The positive filter stores the hash value of legitimate traffic message and their timestamp, and the negative filter stores the hash value of illegitimate traffic information and their timestamp. [11]. It is encrypted with the private key \( SK_{RSU} \) of the RSU which can prevent an attacker from modifying or forging the notification message.

e. Receiving Messages

Upon receiving the notification message from RSU, OBU decrypts it using the public key \( PK_{RSU} \) of RSU. If a vehicle wants to verify the validity of the message \( \{T_5, m, \sigma_m\} \) sent by the other vehicles, OBU will compute \( h(T_5, m) \) and check whether this value is in the notification message. There are three cases of the results, as showed in Table II.

<table>
<thead>
<tr>
<th>Case</th>
<th>Positive Filter</th>
<th>Negative Filter</th>
<th>Result of the Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>True</td>
<td>False</td>
<td>Valid</td>
</tr>
<tr>
<td>2</td>
<td>False</td>
<td>True</td>
<td>Invalid</td>
</tr>
<tr>
<td>3</td>
<td>False</td>
<td>False</td>
<td>Wait for next broadcast</td>
</tr>
</tbody>
</table>

Case 1 means that the message is legitimate, and case 2 indicates that the message is illegitimate. Case 3 depicts that the message has not been authenticated by RSU, therefore, the vehicle just needs to wait for the next notification message from RSU.

V. SECURITY ANALYSIS AND COMPARISONS

Security is one of the basic requirements and core elements of VANETs. In this section, the security features of the proposed scheme are proven to ensure that VANETs security requirements have been met, and further the proposed scheme has been evaluated against a few existing security schemes.

A. Security Proof

The security model of our scheme is to design construct a game between challenger \( C \) and adversary \( A \) that is, whether the adversary \( A \) can win the game of overcoming the challenge given by the challenger \( C \) in the polynomial time with a non-negligible probability.

Definition 1. In the game constructed by the security model of the CPPA scheme in VANETs, the scheme is secure if the advantage of the adversary \( A \) is negligible in polynomial time.

Theorem 1. The registration of RSU in the proposed scheme is secure in the random oracle model.

Proof. Suppose there is an adversary \( A \) who can forge a legitimate message \( \{ID_R, K_R\} \), we construct a challenger \( C \) that can solve the ECDLP problem with a non-negligible probability by running \( A \) as a subroutine.

Setup – Oracle: \( C \) chooses a random number \( s \in \mathbb{Z}_q^* \) as the master private key, and computes \( P_{pub} = s \cdot P \) as the master public key and generates public parameters \( \{p, q, a, b, P, P_{pub}, h\} \).

h – Oracle: \( C \) keeps the list \( L_h \) which maintains the item of query from \( A \) along with its corresponding answer \( \{ID_R, \tau\} \), while the list is initialized to be empty. Upon receiving a query \( ID_R \) from \( A \), \( C \) checks whether the item \( \{ID_R, \tau\} \) is in the list or not. If yes, \( C \) sends \( \tau \) to \( A \). Otherwise \( C \) computes \( \tau = h(ID_R)[s] \), saves \( \{ID_R, \tau\} \) to \( L_h \), and sends \( \tau \) to \( A \).

Sign – Oracle: Upon receiving a query \( ID_R \) from \( A \), \( A \) computes \( K_R = h(ID_R)[s] \) and sends \( \{ID_R, K_R\} \) to \( A \). We can know that \( K_R \) is the signature of \( ID_R \) that is calculated by TA in our scheme.

Output: At last, \( A \) outputs \( \{ID'_R, K'_R\} \), and then \( C \) checks whether the equation \( K'_R = h(ID'_R)[s] \) is satisfied. If not, the game is over and \( A \) fails in the game. If yes, according to the forgery lemma [23], \( A \) will output another valid signature \( \{ID''_R, K''_R\} \) when the equation \( K''_R = h(ID''_R)[s] \) is satisfied.
It means that $A$ can work out $K''_R - K'_R = h(ID''_R||s) - h(ID'_R||s)$. However, the result is contradictory with the unidirectionality of the secure hash function and the ECDLP is a difficult problem, which means $A$ cannot work out the above equation. Therefore, theorem 1 is proved.

**Theorem 2.** The registration of a vehicle in the proposed scheme is secure in the random oracle model.

**Proof.** Suppose there is an adversary $A$ who can forge a legitimate message $\{ID_V, K_V, Z_V\}$, we construct a challenger $C$ that can solve the ECDLP problem with a non-negligible probability by running $A$ as a subroutine.

**Setup – Oracle:** $C$ chooses a random number $s \in Z_q^*$ as the master private key, computes $P_{pub} = s \cdot P$ as the master public key and generates public parameters $\{p, q, a, b, P, P_{pub}, h\}$.

$h – Oracle: C$ keeps the list $L_h$ which maintains the item of query from $A$ and its corresponding answer $\{ID_V, \tau\}$, while the list is initialized to be empty. Upon receiving a query $ID_V$ from $A$, $C$ checks whether the item $\{ID_V, \tau\}$ is in the list or not. If yes, $C$ sends $\tau$ to $A$. Otherwise $C$ computes $\tau = h(ID_V||s)$, saves $\{ID_V, \tau\}$ to , and sends $\tau$ to $A$.

$\oplus – Oracle: C$ keeps the list $L_\oplus$ which maintains the item of query from $A$ and its corresponding answer $\{h(PW_1||PW_2), \tau'\}$, while the list is initialized to be empty. Upon receiving a query $h(PW_1||PW_2)$ from $A$, $C$ checks whether the item $\{h(PW_1||PW_2), \tau'\}$ is in the list or not. If yes, $C$ sends $\tau'$ to $A$. Otherwise $C$ computes $\tau' = K_V \oplus h(PW_1||PW_2)$, saves $\{h(PW_1||PW_2), \tau'\}$ to $L_\oplus$, and sends $\tau'$ to $A$.

**Sign – Oracle:** Upon receiving a query $ID_V$ and $h(PW_1||PW_2)$ from $A$, $C$ computes $K_V = h(ID_V||s)$ and $Z_V = K_V \oplus h(PW_1||PW_2)$, sends $\{ID_V, h(PW_1||PW_2), K_V, Z_V\}$ to $A$. We can know that $K_V$ and $Z_V$ is the signature of $ID_V$ and $h(PW_1||PW_2)$ those are calculated by $TA$ in our scheme.

**Output:** At last, $A$ outputs $\{ID_V, h'(PW_1||PW_2), K'_V, Z'_V\}$, and then $C$ checks whether the equation $K'_V = h(ID'_V||s)$ and $Z'_V = K'_V \oplus h'(PW_1||PW_2)$ is satisfied. If not, the game is over and $A$ fails in the game. If yes, according to the forgery lemma [23], $A$ will output another valid signature $\{ID''_V, h''(PW_1||PW_2), K''_V, Z''_V\}$ which the equation $K''_V = h(ID''_V||s)$ and $Z''_V = K''_V \oplus h''(PW_1||PW_2)$ is satisfied.

It means that $A$ can work out $K''_R - K'_R = h(ID''_R||s) - h(ID'_R||s)$. However, the result is contradictory with the unidirectionality of the secure hash function and the ECDLP is a difficult problem, which means $A$ cannot work out the above equation. Therefore, theorem 2 is proved.

**Theorem 3.** The process of sending an authentication message by OBU in the proposed scheme is secure in the random oracle model.

**Theorem 4.** The process of calculating an authentication message from OBU by RSU in the proposed scheme is secure in the random oracle model.

Theorem 3 and Theorem 4 can be proved by the same way. Therefore, the proposed scheme is secure in the random oracle model.

The next section analyzes the security requirements of the CPPA scheme in VANETs.

1) **Identity privacy preservation:** Normally, the vehicle only sends the pseudo identity once it comes within the range of RSU. Pseudo identity is computed by the equation $X^* = x \cdot P_{pub}$ and $PID_V = ID_V \oplus h(X^*)$, where $x \in Z_q^*$ is a random number. Therefore, no attacker can obtain the real identity $ID_V$ of the vehicle through the pseudo identity $PID_V$. It means that our proposed scheme has met the requirements of identity privacy preservation.

2) **Traceability:** RSU can search the item $< T_{OBU}^{reg}, T_5, m, ID_V >$ according to $T_5$ in the message list $L_m$ when it encounters malicious messages, and then sends $< T_{OBU}^{reg}, ID_V >$ to TA. TA can search the item $< T_{OBU}^{reg}, ID_V >$ according to $T_{OBU}$ in the registration list $L_{RSU}$ while RSU is comprised.

3) **Non-repudiation:** RSU can search out the item $< T_{OBU}^{reg}, T_5, m, ID_V >$ according to the timestamp $T_5$ in the message list $L_m$ quickly, which includes the real identity of the vehicle and its registration time. Therefore, our scheme has met the requirements of Non-repudiation.

4) **Un-linkability:** The format of message related to traffic information in our proposed scheme is $\{T_5, m, s_m\}$, where $s_m = h(ID_R||ID_V||X||Y||SK||s_{TA-OBU}||m||T_5)$, therefore, the attacker cannot determine whether the two given messages are issued by the same vehicle using the message content, which achieves the security requirements of un-linkability.

5) **Resistant to continuous disruption:** TA will delete the registration form in the corresponding registration list when either the real identity of a malicious vehicle or a compromised RSU is detected. Therefore, when either a malicious vehicle is authenticated by a valid RSU or a legitimate vehicle is authenticated by a compromised RSU, TA will immediately stop the certification process to prevent continuous damage.

6) **Modification of passwords:** The owner of a vehicle can change the passwords anytime anywhere whenever he considers the passwords is not secure and the details are as follows. Owner inputs $ID_V, PW_{old}^1, PW_2, PW_{new}^1$ to start OBU. OBU will check whether $ID_V$ and $PW_{old}^1, PW_2$ are identical to the stored ones. If yes, OBU computes $K_V = Z_V \oplus h(PW_{old}^1||PW_2)$ and $Z_{new}^V = K_V \oplus h(PW_{new}||PW_2)$. At last, OBU only needs to replace $Z_{old}^V$ with $Z_{new}^V$.

7) **Resistance to ordinary attacks:**

- **Replay attack:** RSU will check the timestamp while receiving a message, once it is found not to be the latest, RSU will drop it immediately.
- **Modification attack:** It is impossible that the attacker can modify a legal message $\{T_5, m, s_m\}$ to $\{T_5, m', s'_m\}$ where $s'_m = h(ID_R||ID_V||X||Y||SK||s_{TA-OBU}||m'||T_5)$, while the sent real identity of a vehicle $\{T_5, m, s_m\}$ is unknown.
- **Impersonation attack:** If the attacker wants to send a legal message by impersonating the legal vehicle,
it must obtain the real identity of the vehicle. However the attacker cannot obtain the real identity of the vehicle according to the preceding knowledge. Therefore, our proposed scheme can resist the impersonation attack.

**B. Security Comparisons**

In general, the security requirements of VANETs mainly span across message authentication, preservation of identity privacy, traceability, un-linkability, resistant to continuous disruption, modification of passwords, and resistance to ordinary attacks. We evaluate the performance of our scheme against four existing schemes in terms of the security requirements of VANETs. The results are presented in Table III.

**TABLE III**  
**SECURITY COMPARISONS**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Preservation of identity privacy</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Traceability</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
</tr>
<tr>
<td>Un-linkability</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>Resistant to continuous disruption</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>Modification of passwords</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>×</td>
<td>√</td>
</tr>
</tbody>
</table>

Among the evaluated schemes, the other four schemes are not resistant to continuous disruption and modification of passwords. Though, our proposed scheme effectively satisfies all the security requirements of VANETs.

**VI. PERFORMANCE ANALYSIS**

The performance of VANETs is susceptible to computation and communication overheads due to the rapid speed of the vehicles and the rapid changes in the network topology.

**A. Computation Overhead Analysis**

The CPPA schemes proposed by Shim et al. [13] and Zhang et al. [14] are based on bilinear pairing, where the additive group \( G \) with the order \( q \) and its generator \( P \) constitutes all points on the elliptic curve \( E \) defined by the equation \( y^2 = x^3 + x \mod p \), where \( p \) is a 512-bit prime number and \( q \) is a 160-bit prime number. The schemes proposed by He et al. [17] and Zhong et al. [18] are based on Elliptic Curve Cryptography (ECC) to achieve the same level of security, where the additive group \( G \) with the order \( q \) and its generator \( P \) constitutes all points on the elliptic curve \( E' \) defined by the equation \( y^2 = x^3 + ax + b \mod p \), where \( a, b \in F_p, p \) and \( q \) is a 160-bit prime number. The cryptography library used in our experiment is MIRACL[30], which is a well-known and widely used cryptographic library in computing the time required for various cryptographic operations. And our hardware platform consists of an Intel I7-6700 processor 8 gigabytes memory and runs Windows 7 operating system. The definition and execution time of related operations in cryptography are shown in Table IV.

**TABLE IV**  
**THE DEFINITION AND EXECUTION TIME OF RELATED OPERATIONS**

<table>
<thead>
<tr>
<th>Operation</th>
<th>Definition</th>
<th>Execution time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( T_{sm-bp} )</td>
<td>The time of a scalar multiplication operation in a group based on bilinear pairing</td>
<td>0.694</td>
</tr>
<tr>
<td>( T_{sm-ec} )</td>
<td>The time required for performing a scalar point multiplication in a group based on ECC</td>
<td>0.3218</td>
</tr>
<tr>
<td>( T_{sm-bp-s} )</td>
<td>The time required for performing a small scalar point multiplication in a group based on ECC</td>
<td>0.0736</td>
</tr>
<tr>
<td>( T_{sm-ec-s} )</td>
<td>The time of a scalar multiplication operation in a group based on bilinear pairing</td>
<td>0.0246</td>
</tr>
<tr>
<td>( T_{pa-bp} )</td>
<td>The time of a point addition operation in a group based on bilinear pairing</td>
<td>0.0018</td>
</tr>
<tr>
<td>( T_{pa-ec} )</td>
<td>The time required to execute a bilinear pairing operation.</td>
<td>5.086</td>
</tr>
<tr>
<td>( T_h )</td>
<td>The execution time of a general hash function operation</td>
<td>0.001</td>
</tr>
<tr>
<td>( T_{mtp} )</td>
<td>The time required for executing a hash function that maps a string to a point in group</td>
<td>0.0992</td>
</tr>
</tbody>
</table>

The calculation and storage capacity of OBU are low and the number of vehicles is more than RSU, therefore one of the purposes of our scheme is to reduce the OBUs computation overhead. The performance evaluation of the schemes in terms of the execution time consumed to send traffic-related message (that is generating the pseudo identity and message signature) by OBU is shown in Table V.

**TABLE V**  
**THE COMPARISON OF THE EXECUTION TIME**

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Single traffic-related message ( 3T_{sm-bp} + 2T_{pa-bp} + 1T_h \approx 2.0866\text{ms} )</th>
<th>( n ) traffic-related messages ( (3n)T_{sm-bp} + (2n)T_{pa-bp} + nT_h \approx 2.0866\text{ms} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shim et al. [13]</td>
<td>( 3T_{sm-bp} + 2T_{pa-bp} + 1T_h \approx 2.0866\text{ms} )</td>
<td>( (3n)T_{sm-bp} + (2n)T_{pa-bp} + nT_h \approx 2.0866\text{ms} )</td>
</tr>
<tr>
<td>Zhang et al. [14]</td>
<td>( 6T_{sm-bp} + 2T_{pa-bp} + 1T_{mtp} + 4T_h \approx 4.2708\text{ms} )</td>
<td>( (6n)T_{sm-bp} + (2n)T_{pa-bp} + (4n)T_h \approx 4.2708\text{ms} )</td>
</tr>
<tr>
<td>He et al. [17]</td>
<td>( 3T_{sm-ec} + 3T_h \approx 0.9684\text{ms} )</td>
<td>( (3n)T_{sm-ec} + (3n)T_h \approx 0.9684\text{ms} )</td>
</tr>
<tr>
<td>Zhong et al. [18]</td>
<td>( 2T_{sm-ec} + 2T_h \approx 0.6456\text{ms} )</td>
<td>( (2n)T_{sm-ec} + (2n)T_h \approx 0.6456\text{ms} )</td>
</tr>
<tr>
<td>The proposed scheme</td>
<td>( T_{sm-ec} + 6T_h \approx 0.3278\text{ms} )</td>
<td>The worst case: ( nT_{sm-ec} + (6n)T_h \approx 0.3278\text{ms} )</td>
</tr>
<tr>
<td></td>
<td>In the scheme of Zhang et al. [14], the execution time of issuing single traffic-related messages is ( 6T_{sm-bp} + 2T_{pa-bp} + 1T_{mtp} + 4T_h \approx 14.6746\text{ms} ) and the execution time of...</td>
<td>The best case: ( T_{sm-ec} + (n + 5)T_h \approx 0.3218\text{ms} )</td>
</tr>
</tbody>
</table>
issue \( n \) traffic-related message is \((6n)T_{sm-bp} + (2n)T_{pa-bp} + (4n)T_h \approx 14.6746n \) ms, while the execution time in the scheme proposed by Zhong et al. [18] is \( 2T_{sm-ec} + 2T_h \approx 0.8842n \) ms and \((2n)T_{sm-ec} + (2n)T_h \approx 0.8842n \) ms respectively.

In our scheme, whenever the vehicle enters the range of a new RSU, it is necessary to send the identity authentication message to RSU, and the RSU will send messages to TA after processing the identity authentication message which ensures that no malicious vehicle or compromised RSU is involved in the transmission process of traffic-related message. When \( n \) traffic-related messages need to be sent, the worst case is that the OBU sends an identity authentication message to the RSU for each traffic-related message. Therefore the execution time of OBU is \( nT_{sm-ec} + (6n)T_h \approx 0.4426n \) ms. While the best case is when OBU sends the identity authentication message only once in the range of RSU, therefore the execution time of the OBU is \( 1T_{sm-ec} + (n + 5)T_h \approx 0.0001n + 0.4435 \) ms.

\[
T_{sm-bp} + T_{pa-bp} + T_h \approx 16.6098 \text{ ms}
\]

\[
T_{sm-bp} + (2 + 1)T_{sm-bp} + 2T_h + (3n - 3)T_{pa-bp} + 2nT_h \approx 0.748n + 17.3346 \text{ ms}
\]

\[
T_{sm-bp} + 2T_{pa-bp} + 3T \approx 16.6098 \text{ ms}
\]

\[
T_{sm-bp} + (2n)T_{sm-bp} + s + T_h (3n - 2)T_{pa-bp} + (3n)T_h \approx 0.8496n + 15.9484 \text{ ms}
\]

\[
T_{sm-bp} + 2T_h \approx 0.9722 \text{ ms}
\]

\[
(n + 2)T_{sm-ec} + s + T_{pa-bp} \approx 0.500n + 0.6412 \text{ ms}
\]

\[
T_{sm-ec} + (2n)T_{pa-ec} + s + T_h \approx 0.9698 \text{ ms}
\]

\[
T_{sm-ec} + (2n)T_{pa-ec} + s + (3n - 1)T_{pa-bp} + (2n)T_h \approx 0.500n + 0.6412 \text{ ms}
\]

\[
500T_h \approx 0.5 \text{ ms}
\]

\[
(500n)T_h \approx 0.5n \text{ ms}
\]

![Fig. 2. The comparison of the execution time.](image)

![Fig. 3. The comparison of authenticating message time.](image)

Consider, the coverage of the RSU is about 600 m, the vehicle speed is between 0 km/h and 120 km/h [7] and the time taken for sending traffic information message is 100-300 ms [24]. Now, OBU needs to send at least 50 traffic-related messages in the range of RSU. From Fig. 2, the execution time required by OBU to issue traffic-related message in our proposed scheme is much less than that of the other four schemes even in the worst case.

The speed of traffic-related message verification process also determines the computation efficiency of VANETs. Assuming there are 500 vehicles in the range of RSU, the RSU needs to verify 2500-5000 messages per second [25].

Traffic-related message verification process in our scheme is carried out by RSU which usually comprises more storage and computing power. RSU issues the notification information which is encrypted with the private key and the OBU only needs to decrypt the notification information using the public key, and the operation time of decryption is negligible. Upon receiving the traffic information message, the RSU will find out the item \(< T_{ec}^OBU, ID_V, X, Y, SK, \sigma_{TA-OBU} > \) in the list \( L_{auth} \) which satisfies the signature equation. Considering the storage capacity of TPD in the VANETs, TPD will delete the private data before a certain time. Therefore, the data in the list \( L_{auth} \) will be deleted periodically, and the number of item in the list is usually maintained at a level more than that of vehicles in order to continue the process when the vehicle speed is low. Assuming there are 500 vehicles within the RSU coverage, the number of item saved in the list \( L_{auth} \) is 1000, the best case in message authentication process is that only one hash function is performed and the worst case is carried out with 1000 hash functions. The execution time of the message verification process with 500 hash functions by all the evaluated schemes is depicted in Table VI.
This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2017.2782672, IEEE Access

Four schemes in both the aspects. Therefore, our scheme can accommodate more transmission tasks while realizing higher level of security in VANETs, and it is more suitable for VANETs.

VII. CONCLUSION AND FUTURE WORK

This paper proposed a novel and practical ID-based CPA scheme based on invoking the registration list to reflect the role of the revocation list. The proposed scheme greatly reduces the time taken to retrieve the registration list and effectively improves the security of VANETs. Moreover, our scheme does not use bilinear pairing, which is the most complicated operation in modern cryptography, and effectively reduces the length of messages sent by vehicles. Therefore, the proposed scheme also effectively improves the communication performance and efficiency of VANETs.

Security analyses showed that the proposed scheme is not only effective in satisfying the basic security requirements of VANETs but also efficient in preventing the malicious vehicle or the compromised RSU from disrupting VANET security protocols because of the presence of the registration list. Performance analyses demonstrated that our proposed scheme achieves better performance in reducing both the computation overhead and the communication overhead compared to existing schemes, and further exhibits better practical application in current VANETs. As future work, we plan to study means of reducing the operation time of the TA and RSU and the delay time between them, which can obviously increase the security and efficiency of VANETs.

ACKNOWLEDGMENT

The authors are very grateful to the anonymous referees for their detailed comments and suggestions regarding this paper.

REFERENCES


In CPA scheme of VANETs, the value $T$ in the traffic-related message denotes the timestamp. As is shown in Table VII, the traffic-related messages in the scheme of He et al.[17] are \{M, AID, T, R, σ\}, where $AID = \{AID_1, AID_2\}$, $AID_1, R \in \mathbb{G}$ and $AID_2, σ \in Z_q^*$, and the length of a single traffic-related message is 40*2+20*2+4=124 bytes. Moreover, in the CPA scheme of Zhong et al. [18], the vehicle broadcasts the traffic-related messages \{AID, M, T, σ\} to others, where $AID = \{AID_1, AID_2\}$, $AID \in \mathbb{G}$ and $AID_2, σ \in Z_q^*$, and the length of a single traffic-related message is 40+20+4=84 bytes. Traffic-related messages are \{T, m, σ\} in our proposed scheme, where $σ \in Z_q^*$ and the length of a single traffic-related message is 20+4=24 bytes. Therefore, the proposed scheme is superior to the other four schemes in terms of reducing the communication overheads. Thus, from the aforementioned comparative evaluations of the schemes in terms of reducing the computation and communication overheads, we can draw the conclusion that our proposed scheme has obvious advantages than the other schemes.


