Robust Conditional Privacy-Preserving Authentication based on Pseudonym Root with Cuckoo Filter in Vehicular Ad Hoc Networks

Murtadha A. Alazzawi[1,2], Hongwei Lu[1] , Ali A.Yassin[3] and Kai Chen[1*]
[1]School of Computer Science and Technology, Huazhong University of Science and Technology
Wuhan, 430074, China.
[luhw@hust.edu.cn, kchen@hust.edu.cn]
[2]Department of Computer Techniques Engineering, Imam Al-Kadhum College (IKC)
Baghdad, 10001, Iraq.
[murtadhaali@alkadhum-col.edu.iq]
[3]Computer Science Dept., Education College for Pure Science, University of Basrah
Basrah, 61004, Iraq.
[aliadel79yassin@gmail.com]
*****Corresponding author: Kai Chen

Received January 8, 2019; revised April 30, 2019; accepted May 13, 2019; published December 31, 2019

Abstract

Numerous privacy-preserving authentication schemes have been proposed but vehicular ad hoc networks (VANETs) still suffer from security and privacy issues as well as computation and communication overheads. In this paper, we proposed a robust conditional privacy-preserving authentication scheme based on pseudonym root with cuckoo filter to meet security and privacy requirements and reduce computation and communication overheads. In our proposed scheme, we used a new idea to generate pseudonyms for vehicles where each on-board unit (OBU) saves one pseudonym, named as "pseudonym root," and generates all pseudonyms from the same pseudonym. Therefore, OBU does not need to enlarge its storage. In addition, the scheme does not use bilinear pairing operation that causes computation overhead and has no certification revocation list that leads to computation and communication overheads. The proposed scheme has lightweight mutual authentication among all parties and just for once. Moreover, it provides strong anonymity to preserve privacy and resists ordinary attacks. We analyzed our proposed scheme and showed that it meets security and privacy requirements of VANETs and is more efficient than traditional schemes. The communication and computation overheads were also discussed to show the cost-effectiveness of the proposed scheme.

Keywords: Vehicular ad hoc network (VANET), security, privacy-preserving authentication, cuckoo filter, computation and communication overheads

http://doi.org/10.3837/tiis.2019.12.018 **ISSN : 1976-7277**

1. Introduction

Over the recent years, new network technologies have been presented. One of them is known as vehicular ad hoc network (VANET), which is a subset of mobile ad hoc network and aims to provide intelligent transportation systems [1]. In VANETs, vehicles are considered mobile nodes, and each one has an on-board unit (OBU) that communicates with other OBUs through vehicle-to-vehicle communication and to a roadside unit (RSU) through vehicle-to-infrastructure communication [2]. The IEEE 802.11p technology is specially designed for VANETs and denoted as dedicated short-range communication (DSRC) [3]. By using the DSRC protocol, OBU periodically broadcasts beacons, including the vehicle's current information, such as location, velocity, heading, and traffic events [4]. Beacons broadcasting in an open-access network may make the system susceptible to security and privacy threats. Mismanagement on these beacons may lead to traffic accidents and loss of human lives. Security and privacy issues are challenging areas that impede the progress of VANETs. Therefore, these issues must be satisfied to facilitate the deployment of VANET technology.

According to VANETs nature, adversaries can launch several types of attacks by replaying, intercepting, altering, deleting, or forging beacons transmitted among participants. For example, an adversary wants to trouble a vehicle driver, so he/she may eavesdrop communication and collect information about the driver through beacon-based vehicle tracking. Another adversary may also broadcast fake information on a traffic jam or road accident to mislead other drivers for malicious purposes. The solution for the above problems is the provision of important security requirements, such as authentication, message integrity, non-repudiation, un-likability, traceability, and resistance to attacks. The authentication requirement involves some identification information that may threaten the privacy of users. Therefore, achieving a privacy-preserving authentication scheme has become an essential requirement for securing VANETs. To handle privacy issues, VANETs must provide anonymity and a trusted authority (TA) should be the only component which knows the real identity based on the sender's beacons. For example, when a malicious vehicle is detected, the TA should identify the malicious vehicle driver and revoke him.

Many academic studies have been proposed to handle problems in VANETs, but each has its own flaws. In the present paper, we used privacy-preserving authentication schemes to provide conditional privacy with authentication in VANETs without depending on the certification revocation list (CRL) that cause communication and computation overheads. The main contributions of this paper are as follows:

1. We proposed a new idea that deriving other pseudonyms based on a pseudonym root, in which each vehicle saves just one pseudonym (pseudonym root) to conceal its real identity. Therefore, a vehicle does not need to store thousands of pseudonyms within their certificates, thus mitigating the storage capacity of the TA and OBU.
2. Our scheme has no CRL that is used to check revoked vehicles. Rather, it uses a cuckoo filter to save authentic information of vehicles within the RSU's range. Therefore, we mitigate communication overhead.
3. The vehicles in our scheme need to perform mutual authentication just once when they have already reached the first RSU.
4. RSU cannot reveal the real identity of vehicles; it only knows the pseudonym root. Therefore, in the case of RSU compromise, an adversary cannot reveal valuable information.

5. Our proposed scheme uses a cryptographic hash function to produce the beacons by the OBU. Therefore, we mitigate the communication and computation overhead.

The rest of this paper is organized as follows. Section 2 presents some of the existing related works along with their limitations. Section 3 illustrates the preliminaries of the proposed scheme. In section 4, we describe the proposed scheme in detail that is followed by security analysis in section 5. Section 6 presents the performance analysis, including communication overhead and computation overhead comparisons. The proposed scheme is concluded in Section 7.

2. Related Work

Several schemes have been suggested to meet the security and privacy requirements of VANETs. These schemes include group signature-based, pseudonym-based, ID-based, and symmetric cryptography-based schemes [5]. All proposed schemes aim to resolve several security and privacy issues in VANETs. Raya *et al.* [6] used public/private key pairs and corresponding certificates to design conditional privacy-preserving authentication (CPPA) model based on public key infrastructure (PKI). However, each OBU requires storing a big number of key pairs and corresponding certificates. These keys are changed each period so that the tracker cannot track a vehicle. This work considers one of the early pseudonym-based schemes, and other authors have followed this work and provided a lot of new schemes. The previous work has some drawbacks. First, the OBU of each vehicle needs large storage to save public/private key pairs and corresponding certificates. Second, this work used CRL. Thus, when revoking a vehicle, all certificates supplied to a vehicle must be included in the CRL. As a result, the size of CRL will exponentially grow. Accordingly, this case causes an OBU to check CRL before checking a beacon, communication overhead to publish CRL, and large storage to save CRL on an OBU. Sun *et al.* [7] proposed a new scheme, named PASS, which is different from traditional pseudonym schemes as it uses hash chains to reduce the size of CRL. In addition, they proposed proxy re-signature schemes to reduce the time of updating certificates. Lu *et al.* [8] proposed a scheme that depends on the RSU to provide a short-time pseudonym for each OBU, so the pervasive deployment of RSU is significant in this scheme. Wasef *et al.* [9] proposed a scheme, called EMAP, to adopt PKI to provide more comfort to VANETs. The traditional PKI uses a revocation checking process, which takes a long time due to the large size of CRL. EMAP scheme overcomes this limitation by using a keyed-hash message authentication code instead of the CRL checking process. This scheme decreases the message loss ratio caused by traditional authentication schemes. Rajput *et al.* [10] suggest scheme-based pseudonym without using CRL. However, in this scheme, a vehicle acquires two pseudonyms: primary pseudonym from the Certification Authority (CA) that is used for a long time and secondary pseudonyms from the RSU that is used for a short time. This scheme does not meet the unlink ability requirement as the adversary can link two beacons for the same vehicle.

Lin *et al.*, Calandriello *et al.*, and Zhang *et al.* [11-13] proposed group signature(GS)-based schemes that define a set of vehicles to provide privacy preservation by hiding the real identity of the vehicle among other group members. However, this scheme has some flaws [10]. A group manager can track the members because he/she has full knowledge of the group members, the choice of group manager is also complicated, and a vehicle may connect or leave the group at any time in a dynamic environment. In addition, the verification process

between the signature and identity needs to use a pairing operation, which leads to a significant computation overhead on an OBU.

Shim [14] proposed a scheme to provide CPPA named CPAS based on pseudo identity-based signature, which uses batch authentication of beacons on RSU to mitigate the computation overhead of RSU whenever the number of beacons is larger. However, when TA retrieves a complete revocation list, it consumes additional time. TA cannot also process additional authentication overheads produced by the illegitimate part. Zhang *et al.* [15] proposed another CPPA scheme based on pseudo identity-based signature. This scheme supports batch authentication to increase the throughput of identity authentication and enhances computation overheads in the message signature. However, this scheme cannot provide the non-repudiation requirement as pointed out by Lee *et al.* [16]. It cannot also resist a modification attack as pointed by Liu *et al.* [17]. He *et al.* and Zhong *et al.* [18-19] proposed an ID-based CPPA scheme that does not use bilinear pairing operation to improve the computation process and mitigate the significant computation overhead in previous schemes. However, Zhong *et al.* [20] pointed out a flaw in [18-19]. In the case of an attack, TA can track the real identity of an attacker but cannot block it from continuously sending malicious messages. Zhong *et al.* [20] proposed a CPPA scheme that depends on a registration list by using a bloom filter rather than a CRL to mitigate communication overheads. This scheme meets most security and privacy requirements, but it has some limitations. For instance, in the case of RSU compromise, the adversary will acquire all information of all vehicles in RSU's range because RSU has the real identity of vehicles. Moreover, the verification of beacons in OBU depends on the bloom filter that is issued by RSU and it should be updated in each notification message. This process consumes a lot of time. In addition, an OBU repeats the mutual authentication with each new RSU. This process also needs a TA to verify the identity of a vehicle, leading to computation overhead on TA.

Recently, some works proposed to treat the previous problems. Yang et al [21] and Ismaila et al [22] propose schemes based the certificateless cryptography [23] with the elliptic curve cryptography (ECC). These schemes do not need the certificate management found in the traditional public key cryptography (PKC). In a certificateless scheme, a secret key is generated by the vehicle itself depending on the partial secret key produced by a trusted party named the key generator center (KGC). Thus, the KGC does not know the secret key of all vehicles. Besides, in the certificateless scheme, public key certificates are not required to assure the validity of public keys. The schemes in [21,22] satisfy the security and privacy requirements in VANET but still suffer the computation and communication cost inefficiency. To mitigate the computation cost, Jie et al [24] suggest a new scheme using the Chinese remainder theory to share the symmetric cryptography key. This scheme is prone to insider attackers and do not meet the non-repudiation requirement. Using the attribute-based signature (ABS), Hui et al [25] proposed a scheme to address the security in VANET. Since ABS has a similar property to that of GS-based schemes but without needing group administrator to manage the members in the group. In ABS scheme, the TA issues a secret attribute key related with a set of attributes (vehicle's type, color, city, ...) for every vehicle. The validity of the ABS message's signature is achieved by a claim-predicate over these attributes. Moreover, Hui et al [25]'s scheme used the binary structure tree to satisfy the traceability and revocation requirements because the ABS does not support these requirements. In this paper, we propose a new scheme based on CPPA, which uses a new idea to generate pseudonyms for vehicles. Here, each OBU saves one pseudonym (pseudonym root) and generates all pseudonyms from pseudonym root. Therefore, OBU

does not need large storage. It also uses the cuckoo filter in the authentication and verification processes, and the proposed scheme has no CRL that causes computation and communication overheads. Moreover, the proposed scheme satisfies the security and privacy requirements of VANETs.

3. Preliminaries

In this section, we present the system model of the proposed scheme, design objectives, desired requirements, and auxiliary tools.

3.1. System model

VANET structure generally comprises three components, namely, TA, RSU and OBU which are installed in each vehicle as shown in **Fig. 1**. The connection among OBUs or between OBU and RSU uses a wireless channel, and that between TA and RSU uses a wired channel [26].

1. **TA** is a third party that is accountable for generating essential system parameters for the OBU and RSUs. Furthermore, TA is supposed to know the presence of all RSUs and has a secure connection to them.
2. **RSU** is a stationary device located beside the roads and at signal crossroads. It is used to manage the communication of all vehicles inside its range and broadcast notification traffic messages. It also communicates with other RSUs and the TA to send and receive information related to road traffic over a secure channel of the wired network. Furthermore, each RSU has a unique real identity. We refer to this real identity as ID_R in the rest of the paper.
3. **OBU** is a device installed in each vehicle and is used to connect other vehicles and periodically publish beacons. Furthermore, each OBU has a tamper-proof device (TPD) that is used to save secure information. Each vehicle also has a unique real identity, which is referred as ID_v in the rest of the paper

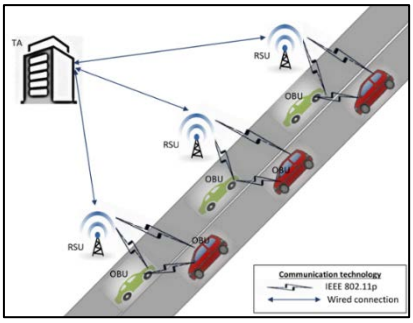

Fig. 1. System model

3.2. Design objectives

The design objectives of the proposed scheme are as follows:

1. **Authentication**: The most important aim is to guarantee the legitimacy of the user. The beacon receiver should be able to authenticate the beacon sender.
2. **Identity and privacy preservation**: Privacy preservation of vehicle information and its owner is also an important aim in VANETs. If this technology is used the people, the

adversary will not be able to acquire the real identity of vehicles according to beacons sent by the vehicles. Only the TA will know the real identity of the beacon sender.

3. **Message integrity**: Message integrity should be ensured, where, the content of the beacon will be transported unaltered to the receiver.
4. **Non-repudiation**: It is one of the significant requirements in VANETs, it means, the sender will not deny sending the beacon.
5. **Traceability and revocability**: TA will trace the real identity of any vehicle and revoke this vehicle from continue sending authentic beacons in VANETs.
6. **Un-linkability**: The adversary will not be able to detect the two beacons sent by the same vehicle.
7. **Resistance to attacks**: A good privacy-preserving authentication scheme in VANETs should be able to resist ordinary attacks, such as:
   - Replay attack which is illegal or malicious users attempt to impersonate an authentic node by using previously generated messages in new connections.
   - Modification attack which is illegal or malicious users attempt to alter or modify messages among all participants in VANETs.
   - Impersonation attack which is illegal or malicious users attempt to impersonate an authentic node, either to disturb the normal working of the network or to use network resources that might not be accessible to it.
8. **Self-verification**: OBU should make a mutual authentication with an RSU without the need for TA intervention. This feature mitigates the time consumed in the communication between RSU and TA.
9. **Pseudonym prediction**: RSU should predict a new pseudonym for any vehicle in its range to make the scheme more secure and cancel the time consumed in the process of exchanging pseudonyms.

3.3. Auxiliary tools

In this paper, we used the following auxiliary tools to complete the proposed scheme:

1. **Cryptographic hash function h**: It is a one-way function that uses to produce fixed length data from arbitrary length data. The following properties should be satisfied to make h secured [27]:
   - Given x, $y = h(x)$ can be easily calculated, whereas the inverse of function $x = h^{-1}(y)$ can be hardly calculated.
   - Given x and y, finding $h(x) = h(y)$ can be computationally infeasible. This property is named as a strong collision resistance.
2. **Cuckoo filter**: It is a new form of probabilistic data structure that is used to test a membership of an item among the set. It gives good search accuracy and time than bloom filters corresponding in a storage size [28]. It is involved in an array of buckets where each bucket comprises several entries. It reduces its space by only computing a fingerprint f of the item's value to be stored in the array. It uses a small f bit fingerprint to symbolize data. Cuckoo filter is used as a cuckoo hashing function to get rid of collisions and is basically a compact cuckoo hash table. Cuckoo hashing function is a form of collision management in hash-based data structures. In cuckoo hashing function, each data item is hashed by two dissimilar hash functions to calculate the indices of two candidate buckets i_1 and i_2 as $i_1 = h(item) mod\ M$ and $i_2 = i \oplus h(f(item)) mod\ M$, where M is the size of cuckoo filter. Value f can be allocated to one of the two candidate buckets where candidate bucket i_1 is tried first. If the bucket i_1 is empty, then the value is located in i_1. If it is allocated, then bucket i_2 is tried. If the

bucket i_2 is empty, then the value is located there. If i_2 is allocated, then the occupant of i_2 is evicted and the value of f is located there. **Fig. 2** illustrates how item w is inserted to cuckoo filter, where the two hashed values i_1, i_2 are mapped to place that are already allocated. To test the membership of any item in the cuckoo filter, we first compute the fingerprint of item $f(item)$ and compute i_1, i_2. Then, if $f(item)$ can be found i_1 or i_2, then the cuckoo filter is proven true; otherwise, the cuckoo filter is proven false

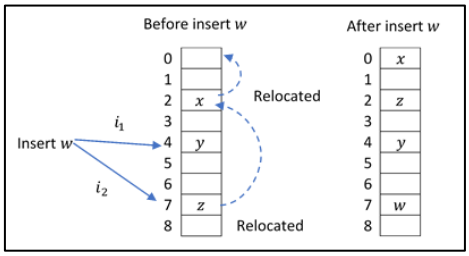

Fig. 2. Insertion operation in cuckoo filter

3. **Homomorphic encryption**: It is defined as a method of encryption that performs a specific algebraic operation on ciphertext and generates an encrypted result that matches the result of the same algebraic operation performed on its plaintext. In mathematics, this method is called mappings or functions [29].

3.4. Assumptions

The following are some assumptions in the proposed scheme:

1. The clock of all participants is synchronized.
2. TA is wholly trustworthy and will not be compromised.
3. Storage capacity and computing power of TA are higher than those of RSU, and those in RSU are higher than those in OBUs.

Table 1. Notations used in the proposed scheme

Notation	Descriptions
Pk, Sk	Public & Private key for TA
p, q	Two big prime numbers
ID_v, ID_R	Real identity for OBU & RSU
PID_v, PID_R	Pseudonyms for hiding the real identity of vehicle and RSU in mutual authentication process
PW	Password of TPD on vehicle
h	Secure hash function
enc_{Pk} , dec_{Sk}	Encryption & decryption operation by TA keys
$\oplus$, $\|$	Exclusive-OR operation and message concatenation operation
T_{Reg_v}, T_{Reg_R}	Registration time for OBU & RSU
P_{root}	Pseudonym root for OBU
Ps	Pseudonym for hiding the real identity of vehicle in beacons
Lev	Level of pseudonym for each vehicle
m_R , m_v, y, z , s, g	Random integer numbers
$RegL_R$, $RegL_v$	Registration lists for RSU & OBU in TA
TL	Temporary list in RSU
PsL	List of authentic vehicles to RSU
msg	Traffic-related message
$T_{1,2,3,4,5,6}$, T_{rec} , ΔT	Timestamps, receiving time, and time delay value

4. Proposed scheme

The proposed scheme is comprised of five phases, namely, initialization, registration, mutual authentication, broadcast and verification, and vehicle revocation phases. The main notations used in our scheme and their descriptions are illustrated in **Table 1**.

4.1. Initialization phase

In this phase, the TA works in producing the essential system parameters. These parameters are published to the participants of VANETs to facilitate the registration and other processes for OBU and RSU:

1- Key generation: By using homomorphic encryption, TA generates public key Pk and private key Sk:
   - First step: TA randomly selects two big prime numbers p and q. These numbers should be independent of each other, such that $\gcd(pq, (p-1)(q-1)) = 1$.
   - Second step: TA computes $n = pq$ and $\lambda = lcm(p-1, q-1)$. lcm means least common multiple.
   - Third step: TA selects random integer g, where $g \in Z^*_{n^2}$
   - Last step: n divides the order of g by checking the existence of the following modular multiplicative inverse $\mu = (L(g^{\lambda} \bmod n^2))^{-1} \bmod n$, where function L is defined as $L(x) = \frac{x-1}{n}$.
2- Public key Pk is (n, g) and private key Sk is (λ, μ)
3- TA selects the cryptographic hash function h.
4- TA generates big integer number $s \in Z^*$. TA periodically updates s each time period.
5- All vehicles can get $\{Pk, h\}$, and all RSUs can get $\{s, h\}$ from TA

4.2. Registration phase

The new participant should undergo a registration process to be verified as authentic. This phase comprises two registration processes, one for RSU registration and the other for vehicle registration:

1- **RSU registration:** TA chooses the real identity of RSU ID_R according to its position. Then, it generates random integer number $m_R \in Z^*$ and finds corresponding registration time T_{Reg_R}. Finally, TA saves $< ID_R, m_R, T_{Reg_R} >$ to registration list $RegL_R$ and sends the same information with number s to RSU.
2- **OBU registration:** In this process, OBU will use 4G/5G communication to send registration request to TA. First, vehicle's driver chooses password PW, and then OBU will send message $\{enc_{Pk}(ID_v, PW)\}$. TA decrypts receiving message $\{dec_{Sk}(ID_v, PW)\}$, and then it will validate real identity ID_v, generate random integer number $m_v \in Z^*$, find corresponding registration time T_{Reg_V}, and compute $m_v^* = m_v \oplus h(PW)$ and $\sigma_{TA} = h(T_{Reg_v} \| ID_v \| m_v)$. Finally, it will save $< ID_R, PW, m_R, T_{Reg_R} >$ to registration list $RegL_v$ and sends $< m_v^*, T_{Reg_v}, \sigma_{TA} >$ to OBU. After receiving the message, OBU computes $m_v = m_v^* \oplus h(PW)$ and checks if $\sigma_{TA} = ? h(T_{Reg_v} \| ID_v \| m_v)$. If equivalent, then it saves $< ID_R, PW, m_R, T_{Reg_R} >$ to its TPD. **Fig. 3** illustrates this process.

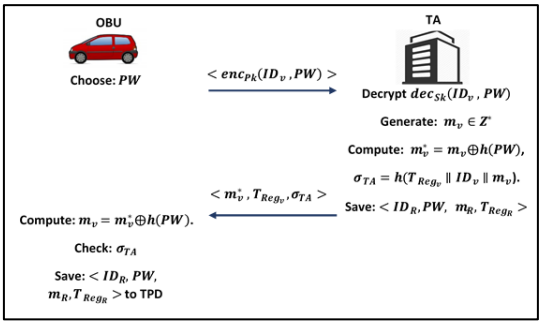

Fig. 3. OBU registeration process

4.3. Mutual authentication phase

This important phase creates a robust authentication among all parts of VANETs (TA, RSU, and OBU). A vehicle driver must input ID_v and PW to TPD to start OBU, and then TPD checks whether $\{ID_v, PW\}$ are matching to the stored ones. If not matching, then OBU does not work; otherwise, it generates random integer number $y \in Z^*$ and compute $PID_v = ID_v \oplus h(y)$, $x = y \oplus h(m_v)$, and $P_{root} = h(ID_v \| y)$. The previous calculation can be completed before a vehicle enters to RSU's range. When the vehicle reaches to the first RSU's range, it performs steps (1–5). When it reaches to other RSUs, it only performs steps (6–8) to mitigate the computation and communication overheads on TA and RSU.

1- OBU computes hash function $\sigma_{OBU} = h(T_{Reg_v} \| T_1 \| PID_v \| x \| y \| ID_v \| ID_R)$. Finally, it sends $\{T_{Reg_v}, T_1, PID_v, x, \sigma_{OBU}\}$ to the RSU.

2- After receiving message $\{T_{Reg_v}, T_1, PID_v, x, \sigma_{OBU}\}$, RSU first checks timestamp T_1 if it is the latest or not. (All the timestamps are confirmed in the following method: Suppose T_{rec} is the receiving time of message, and ΔT is the predefined time delay value. If $(\Delta T > T_{rec} - T)$, then the timestamp is valid. Otherwise, the message will drop). If so, then RSU generates random integer number $z \in Z^*$ and computes $PID_R = ID_R \oplus h(z)$, $w = z \oplus h(m_R)$, and $\sigma_{RSU} = h(T_{Reg_R} \| T_2 \| w \| m_R \| ID_R \| \sigma_{OBU} \| PID_v)$ and then saves information $\{T_2, T_{Reg_v}, PID_v, z\}$ to temporary handshaking list TL. Finally, RSU sends $\{T_{Reg_R}, T_2, w, PID_R, \sigma_{RSU}, T_{Reg_v}, T_1, x, PID_v, \sigma_{OBU}\}$ to TA.

3- After receiving message $\{T_{Reg_R}, T_2, w, PID_R, \sigma_{RSU}, T_{Reg_v}, T_1, x, PID_v, \sigma_{OBU}\}$, TA first checks timestamp T_2 if it is the latest or not. If so, then TA retrieves information $\{m_R, ID_R\}$ and $\{m_v, ID_v\}$ from $RegL_R$ and $RegL_v$ according to T_{Reg_R} and T_{Reg_v}, respectively. Then, TA computes $z' = w \oplus h(m_R)$ and $ID_R{}' = PID_R \oplus h(z')$ and then checks if $(ID_R =? ID'_R)$. If equivalent, then TA checks if $\sigma_{RSU} =? h(T_{Reg_R} \| T_2 \| w \| m_R \| ID_R \| \sigma_{OBU} \| PID_v)$. If yes, then TA computes $y' = x \oplus h(m_v)$ and $ID_v{}' = PID_v \oplus h(y')$ and checks if $(ID_v =? ID'_v)$. If equivalent, then TA ensures the integrity of OBU message by checking if $\sigma_{OBU} =? h(T_{Reg_v} \| T_1 \| PID_v \| x \| y' \| ID_v \| ID_R)$. If equivalent, then it produces $P_{root} = h(ID_v \| y)$. TA saves P_{root} to $RegL_v$ by using it later in the revocation process and hiding it by computing $P^*_{root} = P_{root} \oplus h(m_R)$. Before TA sends the responding message, it should calculate two hash functions, one for RSU and the other for OBU, to prove its validity. The functions are $\sigma_{TA_v} = h(P_{root} \| ID_v \| m_v)$ and $\sigma_{TA_R} = h(T_3 \| T_2 \| P_{root} \| \sigma_{TA_v} \| m_R)$. Lastly, TA sends $\{T_3, T_2, P^*_{root}, \sigma_{TA_R}, \sigma_{TA_v}\}$ to RSU.

4- After receiving message $\{T_3, T_2, P^*_{root}, \sigma_{TA_R}, \sigma_{TA_v}\}$, RSU first checks timestamp T_3 if it is the latest or not. If so, then RSU retrieves the information in TL according to T_2. Then, it computes $P_{root} = P^*_{root} \oplus h(m_R)$ and checks if $\sigma_{TA_R} =? h(T_3 \| T_2 \| P_{root} \| \sigma_{TA_v} \| m_R)$. If equivalent, then RSU derives the first pseudonym level Ps when $Lev = 1$ (we will explain Ps, lev in the next phase). Then, it saves the information of vehicle $< P_{root}, T_{Reg_v}, Ps, Lev >$ with the pseudonym list PsL that contains information of all vehicles in its range. Then, RSU computes $\sigma_{RSU} = h(T_4 \| s \| P_{root} \| \sigma_{TA_v})$ and $s^* = s \oplus h(P_{root})$. Finally, it sends $\{T_4, s^*, \sigma_{RSU}, \sigma_{TA_v}\}$ to OBU.

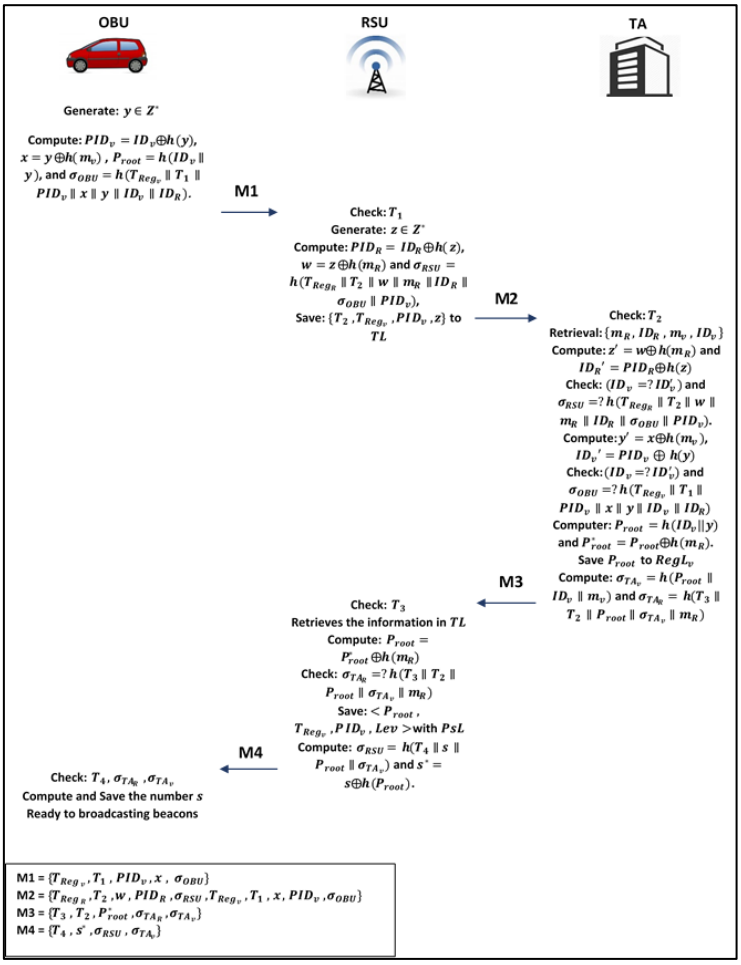

Fig. 4. Mutual authentication process with first RSU

5- After receiving message $\{T_4, s^*, \sigma_{RSU}, \sigma_{TA_v}\}$, OBU first checks timestamp T_4 if it is the latest or not. If yes, OBU computes $s = s^* \oplus h(P_{root})$, it checks if $\sigma_{RSU} =? h(T_4 \| s \| P_{root} \| \sigma_{TA_v})$ and $\sigma_{TA_v} =? h(P_{root} \| ID_v \| m_v)$. If they are equivalent, that means all the three parts have completed the mutual authentication process. Moreover, RSU and OBU have agreement to use the same P_{root} to generate all other pseudonyms. **Fig. 4** illustrates all the processes in this phase. After the end of this step, an OBU acquires number s. Number s is a shared secure number among all RSUs in VANETs that is

randomly generated by TA and periodically updated. An OBU acquires this number after finishing the five steps with the first RSU, and OBU will use the number to create mutual authentication with the rest of the RSUs. Therefore, it only needs to perform steps (6–8) to create the mutual authentication with the rest of the RSUs. In addition, OBU uses number s with each beacon to facilitate the verification process on the beacon receiver. (When TA updates number s, all OBUs should send a request message to the nearest RSU to acquire the new number s).

6- When OBU gets out from the first RSU's range and gets into the new RSU's range, it sends a joint request message to the new RSU. This message is $\{T_5, T_{Reg_v}, P^*_{root}, Lev, \sigma_{OBU}\}$, where $\sigma_{OBU} = h(T_5 \parallel T_{Reg_v} \parallel P^*_{root} \parallel Lev \parallel s)$ and $P^*_{root} = P_{root} \oplus h(s)$.

7- After receiving message $\{T_5, T_{Reg_v}, P^*_{root}, Lev, \sigma_{OBU}\}$, RSU first checks timestamp T_5 if it is the latest or not. If so, then it checks if $\sigma_{OBU} =? h(T_5 \parallel T_{Reg_v} \parallel P^*_{root} \parallel Lev \parallel s)$. If not equivalent, then the message will drop and assign $check = 0$. Otherwise, RSU computes $P_{root} = P^*_{root} \oplus h(s)$, increases $Lev + 1$, derives new Ps, inserts $f(Ps)$ to the cuckoo filter, saves all the information of OBU to PsL, and assigns $check = 1$. Finally, it computes $\sigma_{RSU} = h(T_6 || check || P_{root})$ and sends the message $\{T_6, check, \sigma_{RSU}\}$ to OBU.

8- After receiving message $\{T_6, check, \sigma_{RSU}\}$, OBU first checks timestamp T_6 if it is the latest or not. If so, then it checks if $\sigma_{RSU} =? h(T_6 || check || P_{root})$. If equivalent, then it checks if the value of $check$ is equal to one, and then it starts broadcasting beacons. Otherwise, it should perform the first five steps **Fig. 5** illustrates the mutual authentication with the rest RSUs after getting out from the first RSU's range.

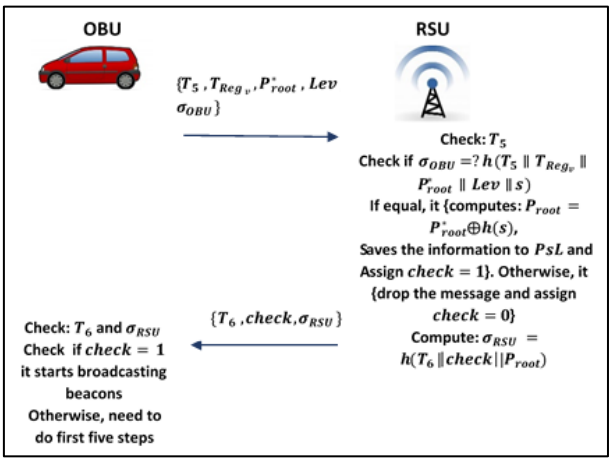

Fig. 5. Mutual authentication process with the rest RSUs

4.4. Broadcasting and verification phase

RSU periodically broadcasts notification messages that are contained on the cuckoo filter. The filter is used to store the fingerprint of legitimate pseudonym $f(Ps)$. RSU also updates the cuckoo filter after the time period or when a vehicle leaves or joins.

1. **Broadcasting process**: After the mutual authentication process is completed, OBU starts broadcasting the beacons. Prior to that, RSU and OBU perform the following:

   - RSU derives the first pseudonym level for the new vehicle from its P_{root} as $Ps = h(P_{root} \parallel Lev)$, where $Lev = 1$.
   - RSU inserts $\{Ps, Lev\}$ to PsL.
   - RSU inserts $f(Ps)$ to the cuckoo filter by cuckoo hashing that is explained in section (3.4), and publishes it with a notification message. (After this step, all vehicles in RSU's range acquires the cuckoo, so the beacons for the new vehicle will be proven authentic.)
   - OBU derives the first pseudonym level Ps from P_{root} and $Lev = 1$. Therefore, the beacon will be known to all participants and the beacon's sender will be confirmed as an authentic sender. The beacon is derived from $\{T, msg, \sigma_{msg}\}$, where $\sigma_{msg} = Ps \oplus h(T \parallel msg \parallel s)$. (RSU increases Lev by one for all OBUs in its range to derive the new Ps with each updating process for the cuckoo filter. After updating the cuckoo filter, OBU also increases Lev by one and derives the same new Ps.)

2. **Verification process**: When a vehicle receives beacon $\{T, msg, \sigma_{msg}\}$, it performs the following steps:

   - First step: Check timestamp T if it is the latest or not. If so, then it continues the verification process. Otherwise, it drops the beacon.
   - Second step: Compute $Ps = \sigma_{msg} \oplus h(T \parallel msg \parallel s)$.
   - Third step: Check $f(Ps)$ in the two hashed i_1, i_2 in the cuckoo filter. If unoccupied, then it drops the beacon.

4.5. Vehicle revocation phase

This phase explains how a TA revokes any vehicle that broadcasts false information. However, each RSU has all the information about OBUs inside its range in PsL, so if a culprit vehicle is found, the RSU acquires the information of this vehicle from its beacon. Then, it sends the information $\{P_{root}, T_{Reg_v}\}$ to TA. TA retrieves real identity ID_v from $RegL_v$ according to the information in the received message. Next, it removes the vehicle from $RegL_v$, inserts it to the revocation list, and updates number s. Lastly, TA notices RSU to remove the vehicle from its PsL and revoke it from continuous broadcasting. **Fig. 6** illustrates this phase.

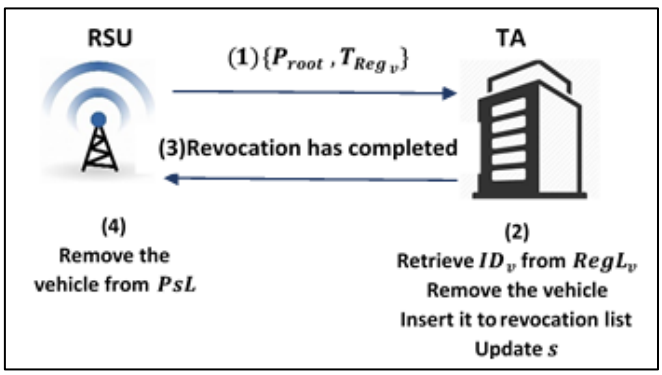

Fig. 6. Revocation process

5. Analysis of the proposed scheme

In this section, we analyze the proposed scheme to confirm that the requirements of security and privacy in VANETs have been met in this paper. In addition, we analyze the resistance of our scheme to some ordinary attacks. **Table 2** explains the comparisons between our scheme and other related schemes in terms of the security and privacy requirements and ordinary attacks.

5.1. Security and privacy requirements

1. **Authentication**: In the proposed scheme, all vehicles broadcast safety beacons, where the content of each beacon is $\{T, msg, \sigma_{msg}\}$. The vehicle must have number s to compute $\sigma_{msg} = Ps \oplus h(T \parallel msg \parallel s)$, and only the authentic vehicle that completed the mutual authentication phase has this number. Therefore, the adversary cannot easily broadcast authentic beacons without number s. Moreover, the receiver vehicle must have the same number to implement $Ps = \sigma_{msg} \oplus h(T \parallel msg \parallel s)$ and acquire the true pseudonym of sender Ps to check it in the cuckoo filter. Thus, our scheme has met the authentication requirement.

Table 2. Security comparisons our proposed scheme and previous schemes

Security features	[14]	[15]	[18]	[19]	[20]	[21]	[22]	Our scheme
Authentication	✓	✓	✓	✓	✓	✓	✓	✓
Identity privacy preserving	✓	✓	✓	✓	✓	✓	✓	✓
Message integrity	✓	✓	✓	x	✓	✓	✓	✓
Non-repudiation	✓	✓	x	✓	✓	✓	✓	✓
Traceability	✓	✓	✓	✓	✓	✓	✓	✓
Revocability	X	x	x	x	✓	x	x	✓
Un-linkability	✓	✓	✓	✓	✓	✓	✓	✓
Resistance to replay attack	X	x	✓	✓	✓	✓	✓	✓
Resistance to impersonation attack	✓	x	✓	x	✓	✓	✓	✓
Resistance to modification attack	✓	✓	✓	x	✓	✓	✓	✓
Self-verification	✓	✓	✓	✓	x	✓	✓	✓
Pseudonym prediction	X	x	x	x	x	x	x	✓

2. **Identity and privacy preservation**: The content of the beacon $\{T, msg, \sigma_{msg}\}$ has no information about the vehicle's real identity. Therefore, no adversary can acquire real identity ID_v. Thus, our scheme has met the identity and privacy preservation requirement.
3. **Message integrity**: In our proposed scheme, we use hash function to check message integrity, in which each beacon has $\sigma_{msg} = Ps \oplus h(T \parallel msg \parallel s)$ that ensures that the beacon is received without alterations. Thus, our scheme has met the message integrity requirement.

4. **Non-repudiation**: According to the value of σ_{msg} in the beacon $\{T, msg, \sigma_{msg}\}$, which is computed as $\sigma_{msg} = Ps \oplus h(T \| msg \| s)$, where $Ps = h(P_{root} \| Lev)$, vehicles cannot broadcast the same beacon because each vehicle has unique P_{root}. Therefore, any vehicle cannot deny sending its beacons. Thus, our scheme has met the non-repudiation requirement.
5. **Traceability and revocability**: RSU can trace OBU in accordance with σ_{msg} in the beacon, where $\sigma_{msg} = Ps \oplus h(T \| msg \| s)$, and RSU has number s, so it can compute $Ps = \sigma_{msg} \oplus h(T \| msg \| s)$ and acquire all vehicle's information in PsL in accordance with Ps. Then, it sends $\{P_{root}, T_{Reg_v}\}$ to TA to reveal the real identity of OBU in accordance with $\{P_{root}, T_{Reg_v}\}$ in $RegL_R$. Finally, it revokes the vehicle from continuously broadcasting beacons by sending a notice message to RSU to remove it from the cuckoo filter and update number s to all RSUs. Thus, our scheme has met the traceability and revocability requirement.
6. **Un-linkability**: The format of beacon in our scheme is $\{T, msg, \sigma_{msg}\}$, where $\sigma_{msg} = Ps \oplus h(T \| msg \| s)$. Therefore, all beacons for the same vehicle are different and the adversary cannot determine whether the two given beacons are generated by the same OBU. Thus, our scheme has met the Un-linkability requirement.
7. **Self-verification**: In our proposed scheme, the mutual authentication needs TA intervention just for once with the first RSU. After that, the mutual authentication process with other RSUs does not need the TA. The OBU sends a request message to the new RSU. The content of request message is $\{T_5, T_{Reg_v}, P^*_{root}, Lev, \sigma_{OBU}\}$, where $\sigma_{OBU} = h(T_5 \| T_{Reg_v} \| P^*_{root} \| Lev \| s)$ and $P^*_{root} = P_{root} \oplus h(s)$. Therefore, if number s in the request message matches to that saved in the RSU, then an RSU will accept the request message and an OBU will become a legitimate node. This procedure continues until TA updates number s. Thus, after each updating process for number s, OBU needs to renew the mutual authentication from the first.
8. **Pseudonym prediction**: In the proposed scheme, OBU uses the pseudonym root to derive any new Ps by $Ps = h(P_{root} \| Lev)$, where $P_{root} = h(ID_v \| y)$ and $y \in Z^*$ random integer number. Through the mutual authentication phase, TA computes the same P_{root} and sends it to RSU, so RSU has the ability to predict the new pseudonym for any vehicle by computing $Ps = h(P_{root} \| Lev)$. Thus, our scheme has met the pseudonym prediction requirement.

5.2. Attack scenarios

Theorem 1. *The proposed scheme resists the replay attack.*

Proof Theorem 1. In accordance with the content of beacon $\{T, msg, \sigma_{msg}\}$, where $\sigma_{msg} = Ps \oplus h(T||msg||s)$, the attack cannot possibly use another timestamp because it leads to different values of σ_{msg}. In addition, the beacon receiver first checks the timestamp. If it is not the latest, then it drops the beacon, so the replay attack fails in our scheme.

Theorem 2. *The proposed scheme resists the modification attack.*

Proof Theorem 2. In accordance with σ_{msg} that is embedded in each beacon, the modification attack cannot fully modify a beacon because $\sigma_{msg} = Ps \oplus h(T||msg||s)$ and the attacker has no Ps, s. Thus, the modification attack fails in our scheme.

Theorem 3. *The proposed scheme resists the impersonation attacks.*
Proof Theorem 3. The attacker must acquire P_{root} of the vehicle if it desires to broadcast an authorized beacon by impersonating the legal vehicle. In accordance with $P_{root} = h(ID_v||y)$, where $y \in Z^*$ is a random integer number, the attacker cannot easily acquire ID_v and y to compute P_{root} of the vehicle. Thus, the impersonation attack fails in our scheme.

Theorem 4. *In the proposed scheme, if an adversary succeeds in compromising RSU, then it cannot reveal the real identity of any vehicle.*
Proof Theorem 4. In the proposed scheme, RSU only has P_{root} that is acquired from the TA through the mutual authentication phase, where $P_{root} = h(ID_v||y)$ and $y \in Z^*$ is a random integer number. Therefore, an adversary cannot compute ID_v from P_{root} or acquire valuable information by compromising any RSU.

Theorem 5. *In the proposed scheme, an adversary cannot acquire the secure random numbers* m_v , s.
Proof Theorem 5. An adversary cannot acquire number m_v because TA computes $m_v^* = m_v \oplus h(PW)$ and sends m_v^* to OBU. OBU retrieves m_v by implementing $m_v = m_v^* \oplus h(PW)$, so an adversary should know PW to get m_v. In addition, OBU uses $x = y \oplus h(m_v)$ to send a request message to the first RSU, where $y \in Z^*$ is a random integer number. Similarly, an adversary cannot acquire number s because RSU computes $s^* = s \oplus h(P_{root})$ and sends s^* to an OBU. OBU retrieves s by implementing $s = s^* \oplus h(P_{root})$. Thus, an adversary cannot easily acquire number s unless it has P_{root}. OBU also uses the value of hashed number s in beacons and request joint messages with the rest of RSUs:

1. The beacon message is $\{T, msg, \sigma_{msg}\}$, where $\sigma_{msg} = Ps \oplus h(T \parallel msg \parallel s)$.
2. The request join message is $\{T_5, T_{Reg_v}, P_{root}^*, Lev, \sigma_{OBU}\}$, where $\sigma_{OBU} = h(T_5 \parallel T_{Reg_v} \parallel P_{root}^* \parallel Lev \parallel s)$ and $P_{root}^* = P_{root} \oplus h(s)$.

Furthermore, the TA will update number s periodically. Therefore, the adversary cannot easily acquire secure random numbers m_v , s.

6. Performance analysis

In this section, we discuss the performance of our proposed scheme in terms of communication and computation overheads. We also compare the performance of our proposed scheme with Shim et al. [14], Zhang et al. [15], He et al. [18], Zhong et al. [19], Zhong et al. [20], Yang et al. [25], and Ismaila et al.'s [26] schemes.

Table 3. The execution time and definition of related operations [20].

Operation	The time	Definition
T_{sm-bp}	0.694 ms	The time consumed to execute the scale multiplication operation in a group based on bilinear pairing
$T_{sm-bp-s}$	0.0736 ms	The time consumed to execute the small scalar point multiplication operation in a group based on bilinear pairing
T_{pa-bp}	0.0018 ms	The time consumed to execute the point addition operation in a group based on bilinear pairing
T_{sm-ecc}	0.3218 ms	The time consumed to execute the scale multiplication operation in a group based on ECC
$T_{sm-ecc-s}$	0.0246 ms	The time consumed to execute the small scalar point multiplication operation in a group based on ECC

T_{pa-ecc}	0.0024 ms	The time consumed to execute the point addition operation in a group based on ECC
T_{bp}	5.086 ms	The time consumed to execute the bilinear pairing operation
T_h	0.001 ms	The time consumed to execute the general hash function operation
T_{mtp}	0.0992 ms	The time consumed to execute the map-to-point hash function operation

6.1. Computation overhead

Shim et al.'s [14] and Zhang et al.'s [15] schemes are based on bilinear pairing. In the bilinear pairing, the the additive group $\bar{G}$ is generated based on elliptic curve $y^2 = x^3 + x \ mod \ p$, where P is a 512-bit prime number. He et al. [18], Zhong et al. [19], and Zhong et al. [20], Yang et al [21], and Ismaila et al.'s [22] scheme are based on ECC. In the ECC, the additive group G is generated based on elliptic curvev $y^2 = x^3 + ax + b \ mod \ p$, where p is a 160-bit prime number. Our proposed scheme generates and verifies beacons based on the hash function. The cryptography operations that adopted in our paper present in **Table 3** according to Zhong et al. [20]. Where Zhong et al. used the MIRACL library in their experiment, which MIRACL is widely utilized in computing different cryptography operations.

We compare our proposed scheme with the above-mentioned schemes in terms of computation costs, in two processes, namely, authentication beacon generation and authentication beacon verification. **Table 4** and **Fig. 7** lists the comparison results of the beacon generation process, and **Table 5** and **Fig. 8** lists the comparison results of the beacon verification process.

Table 4. The comparison of the execution time for beacon generation

Scheme	Execution time for single beacon generation	Execution time for n beacons generation
Shim. [14]	2.0866 ms	2.0866n ms
Zhang *et al.* [15]	4.2708 ms	4.2708n ms
He *et al.* [18]	0.9684 ms	0.9684n ms
Zhong *et al.* [19]	0.6456 ms	0.6456n ms
Zhong *et al.* [20]	0.3278 ms	$[0.3278n$, $0.001n + 0.3218]ms$
Yang et al. [21]	0.3218 ms	0.3218n ms
Ismaila et al [22]	0.9732 ms	0.9732n ms
Our proposed scheme (with first RSU)	0.008 ms	$0.001n + 0.007$ ms
Our proposed scheme (with rest of RSUs)	0.004 ms	$0.001n + 0.003$ ms

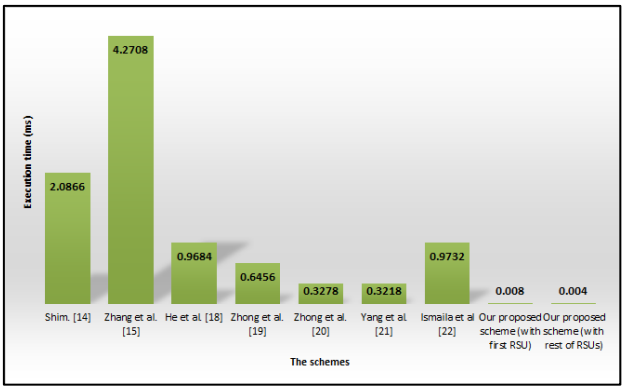

Fig. 7. The comparison of the execution time for beacon generation

In Shim's scheme [14], the execution time for generating a single beacon is $3T_{sm-bp} + 2T_{pa-bp} + 1T_h \approx 2.0866\,ms$.

In Zhang et al.'s scheme [15], the execution time for generating a single beacon is $6T_{sm-bp} + 2T_{pa-bp} + 1T_{mtp} + 4T_h \approx 4.2708\,ms$.

In He et al.'s scheme [18], the execution time for generating a single beacon is $3T_{sm-ecc} + 3T_h \approx 0.9684\,ms$.

In Zhong et al.'s scheme [19], the execution time for generating a single beacon is $2T_{sm-ecc} + 2T_h \approx 0.6456\,ms$.

In Zhong et al.'s scheme [20], the execution time for generating a single beacon is $1T_{sm-ecc} + 6T_h \approx 0.3278\,ms$.

In Yang et al.'s scheme [21], the execution time for generating a single beacon is $1T_{sm-ecc} \approx 0.3218\,ms$.

In Ismaila et al.'s scheme [22], the execution time for generating a single beacon is $3T_{sm-ecc} + 2T_{pa-ecc} + 3T_h \approx 0.9732\,ms$.

In our proposed scheme, when the vehicle joins the new RSU's range, it needs to perform mutual authentication by sending a request message to RSU. Through the mutual authentication process, OBU consumes $7T_h$ with the first RSU and $3T_h$ with the rest of RSUs. After that, it only consumes $1T_h$ with each beacon generation process in the broadcasting process. Therefore, the execution time for generating a single beacon in our proposed scheme is $8T_h = 0.008\,ms$ with the first RSU and $4T_h = 0.004\,ms$ with the rest of RSUs. For broadcasting *n* beacons, the execution time with the first RSU is $nT_h + 7T_h = 0.001n + 0.007$ and the execution time with the rest of RSUs is $nT_h + 3T_h = 0.001n + 0.003$.

Table 5. The comparison of the execution time for beacon verification

Scheme	Execution time for single beacon verification	Execution time for n beacons verification
Shim. [14]	$16.6498\ ms$	$0.7014n + 15.9466\ ms$,
Zhang *et al.* [15]	$16.6508\ ms$	$0.8496n + 15.9484\ ms$
He *et al.* [18]	$0.9722\ ms$	$0.3802n + 0.6412\ ms$
Zhong *et al.* [19]	$0.9698\ ms$	$0.3778n + 0.6412\ ms$
Zhong *et al.* [20]	$0.5\ ms$	$0.5n\ ms$
Yang et al. [21]	$1.2872\ ms$	$1.2872n\ ms$
Ismaila et al [22]	$0.6470\ ms$	$0.0772n + 0.6436\ ms$
Our proposed scheme	$0.004\ ms$	$0.004n\ ms$.

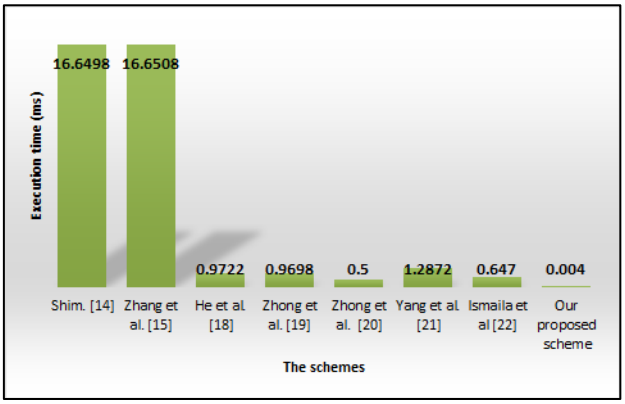

Fig. 8. The comparison of the execution time for single beacon verification

In Shim's scheme [14], the execution time for verifying a single beacon is $3T_{bp} + 2T_{sm-bp} + 1T_{pa-bp} + 2T_h \approx 16.6498\ ms$ and that for verifying n beacons is $3T_{bp} + (1 + \text{n})T_{sm-bp} + (3n - 3)T_{pa-bp} + 2nT_h \approx 0.7014n + 15.9466\ ms$.

In Zhang et al.'s scheme [15], the execution time for verifying a single beacon is $3T_{bp} + 2T_{sm-bp} + 1T_{pa-bp} + 3T_h \approx 16.6508\ ms$ and that for verifying n beacons is $3T_{bp} + (n + 1)T_{sm-bp} + (2n)T_{sm-bp-s} + (3n - 2)T_{pa-bp} + (3n)T_h \approx 0.8496n + 15.9484\ ms$.

In He et al.'s scheme [18], the execution time for verifying a single beacon is $3T_{sm-ecc} + 2T_h + 2T_{pa-ecc} \approx 0.9722\ ms$ and that for verifying n beacons is $(n + 2)T_{sm-ecc} + (2n)T_{sm-ecc-s} + (3n - 1)T_{pa-ecc} + (2n)T_h \approx 0.3802n + 0.6412\ ms$.

In Zhong et al.'s scheme [19], the execution time for verifying a single beacon is $3T_{sm-ecc} + 2T_h + 1T_{pa-ecc} \approx 0.9698\ ms$ and that for verifying n beacons is $(n + 2)T_{sm-ecc} + (2n)T_{sm-ecc-s} + (2n - 1)T_{pa-ecc} + (2n)T_h \approx 0.3778n + 0.6412\ ms$.

In Zhong et al.'s scheme [20], the execution time for verifying a single beacon is $500\text{T}_\text{h} \approx 0.5$ ms and that for verifying n beacons is $(500\text{n})\text{T}_\text{h} \approx 0.5\text{n}$ ms.

In Yang et al.'s scheme [21], the execution time for verifying a single beacon is $4T_{sm-ecc} \approx 1.2872\ ms$ and that for verifying n beacons is $(4n)T_{sm-ecc} \approx 1.2872n\ ms$.

In Ismaila et al.'s scheme [22], the execution time for verifying a single beacon is $2T_{sm-ecc} + 1T_{pa-ecc} + T_h \approx 0.6470\,ms$ and that for verifying n beacons is $2T_{sm-ecc} + (3n)T_{sm-ecc-s} + nT_{pa-ecc} + nT_h \approx 0.0772n + 0.6436\,ms$.

In our proposed scheme, the execution time for verifying a single beacon is $4T_h \approx 0.004$ ms and that for verifying n beacons is $(4n)T_h \approx 0.004n$ ms.

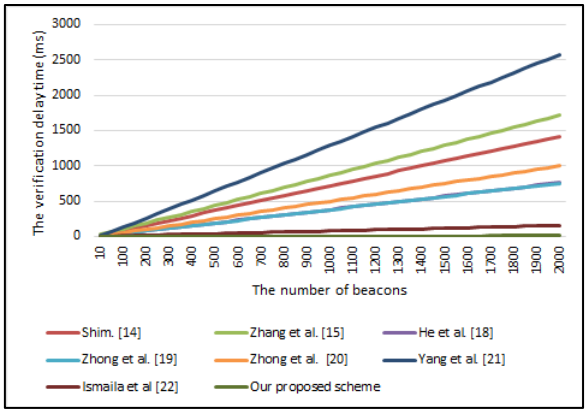

Fig. 9. Verification delay versus a number of received beacons

Fig. 9 presents the verification delay according to the number of beacons that is received by any trusted vehicle. The vehicle in VANET requites 100-300 ms to broadcast one beacon [3]. In case of high density, each vehicle will receive beacons from about 180 vehicles every 100-300 ms, that means, the vehicle should verify 600-2000 beacons per one second [14]. To verify 2000 beacons, the vehicle in our scheme only needs to 8.00 ms, whereas the schemes in [14,15,18,19,20,21,22] need to 1418,74 ms, 1715.14 ms, 761.04 ms, 756.24 ms, 1000 ms, 2574.40 ms, and 155.04 ms Respectively.

Consequently, we can deduce that our proposed scheme is more efficient compared with other related schemes in terms of reducing computation overhead and thus can be effectively used in VANETs.

6.2. Communication overhead

In the elliptic curve $y^2 = x^3 + x \bmod p$, where p is a 512-bit prime number, the size of each element in the bilinear pairing-based group $\bar{G}$ is 128 bytes. Whereas in the elliptic curve $y^2 = x^3 + ax + b \bmod p$, where p is a 160-bit prime number, the size of each element in the ECC-based group G is 40 bytes [30]. The size of timestamp is 4 bytes and the size of output of the secure hash function is 20 bytes [31]. The size of element in integer group Z_q^* is also 20 bytes. The length of the traffic information message is not computed in the comparisons listed in **Table 6** and **Fig. 10**.

Table 6. Communication cost of different schemes

Scheme	Communication cost for single beacon (byte)	Communication cost for n beacons (byte)
Shim. [14]	644	644n
Zhang *et al.* [15]	388	388n
He *et al.* [18]	124	124n
Zhong *et al.* [19]	84	84n
Zhong *et al.* [20]	24	24n
Yang et al. [21]	148	148n
Ismaila et al [22]	144	144n
Our scheme	24	24n

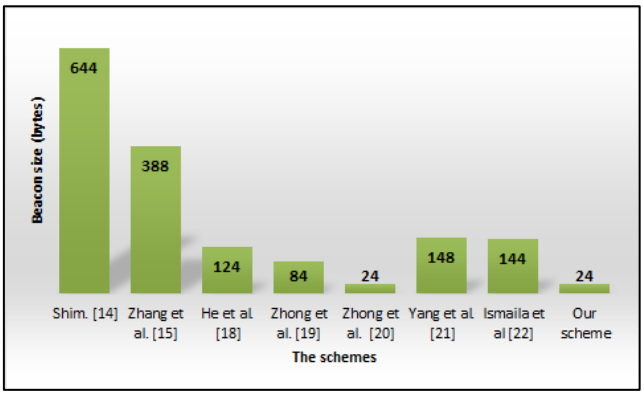

Fig. 10. Communication cost of different schemes

In Shim's scheme [14], the content of the broadcasted message is $\{AID_i, T_i, U_i, V_i, W_i\}$, where $AID_1 = \{AID_i^1, AID_i^2\}, AID_i^1, AID_i^2, U_i, V_i, W_i \in \bar{G}$ and T_i is the timestamp. Therefore, the communication cost for Shim's scheme [14] is $128 \times 5 + 4 = 644$ bytes.

In Zhang et al.'s scheme [15], the content of the broadcasted message is $\{ID, M, \sigma, T\}$, where $ID = \{ID_1, ID_2\}$, $ID_1, ID_2, \sigma \in \bar{G}$ and T is the timestamp. Therefore, the communication cost for Zhang et al.'s scheme [15] is $128 \times 3 + 4 = 388$ bytes.

In He et al.'s scheme [18], the content of the broadcasted message is $\{M, AID, T, R, \sigma\}$, where $AID = \{AID_1, AID_2\}$, $AID_1, R \in G$ and $AID_2, \sigma \in Z_q^*$ and T is the timestamp. Therefore, the communication cost for He et al.'s scheme [18] is $40 \times 2 + 20 \times 2 + 4 = 124$ bytes.

In Zhong et al.'s scheme [19], the content of the broadcasted message is $\{AID, M, \sigma, T\}$, where $AID = \{AID_1, AID_2\}$, $AID_1 \in G$ and $AID_2, \sigma \in Z_q^*$ and T is the timestamp. Therefore, the communication cost for Zhong et al.'s scheme [19] is $40 + 20 \times 2 + 4 = 84$ bytes.

In Zhong et al.'s scheme [20], the content of the broadcasted message is $\{T, m, \sigma\}$, where $\sigma \in Z_q^*$ and T is the timestamp. Therefore, the communication cost for Zhong et al.'s scheme [20] is $20 + 4 = 24$ bytes.

In Yang et al.'s scheme [21], the content of the broadcasted message is $\{PID_i, PK_i, ct_i, u_i, v_i\}$, where $PID_i = \{PID_{i,1}, PID_{i,2}, T_i\}$, $PID_{i,1}, PK_i \in G$, $PID_{i,2}, u_i, v_i \in Z_q^*$ and T_i, ct_i are timestamps. Therefore, the communication cost for Yang et al.'s scheme [21] is $40 \times 2 + 20 \times 3 + 4 \times 2 = 148$ bytes.

In Ismaila et al.'s scheme [22], the content of the broadcasted message is $\{PID_{y,k}, PK_k, \omega_k, \sigma_k = (R_k, \vartheta_k), T_k\}$, where $PK_k, R_k \in G$, $PID_{y,k},, \omega_k, \vartheta_k \in Z_q^*$ and T is the timestamp. Therefore, the communication cost for Ismaila et al.'s scheme [22] is $40 \times 2 + 20 \times 3 + 4 = 144$ bytes.

In our proposed scheme, the content of the broadcasted message is $\{T, msg, \sigma\}$, where $\sigma \in Z_q^*$ and T is the timestamp. Therefore, the communication cost for our proposed scheme is $20 + 4 = 24$ bytes.

In accordance with the comparative evaluations of our proposed scheme with previous schemes in terms of reducing the communication overhead, we can deduce that the proposed scheme can be efficiently used in VANETs.

7. Conclusions

This paper proposed a robust CPPA scheme in VANETs that depends on a new idea of acquiring pseudonym by using a pseudonym root. It utilizes the cuckoo filter to mitigate the verification process. The proposed scheme does not use bilinear pairing and has no CRL. Therefore, the proposed scheme can effectively mitigate the computation and communication overheads in VANETs. Furthermore, the proposed scheme can ensure the security and privacy requirements such as privacy-preserving authentication, integrity, non-repudiation, traceability, revocability, Un-linkability, self-verification, pseudonym prediction and resistance to the ordinary attacks. Moreover, the proposed scheme can provide conditional anonymity to participants in a network thereby preventing malicious vehicles from disrupting VANETs. In future works, we plan to mitigate the dependence of vehicles on the RSU to resolve authentication problems in non-RSU.

References

[1] Mu Han, Lei Hua and Shidian Ma, "A Self-Authentication and Deniable Efficient Group Key Agreement Protocol for VANET," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 7, pp. 3678-3698, 2017. Article (CrossRef Link)

[2] Hamssa Hasrouny, Abed Ellatif Samhat, Carole Bassil and Anis Laouiti, "VANet security challenges and solutions: A survey," *Vehicular Communications*, vol. 7, pp. 7-20, 2017. Article (CrossRef Link)

[3] John B. Kenney, "Dedicated short-range communication (DSRC) standards in the United States," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1162-1182, 2011. Article (CrossRef Link)

[4] Shihan B., Waleed H., Haitham C., Zhili S., Phillip A. and Ao, L., "A lightweight authentication and privacy-preserving scheme for VANETs using TESLA and Bloom Filters," *ICT Express*, vol. 4, no. 4, pp. 221-227, 2018. Article (CrossRef Link)

[5] Sunilkumar S. Manvia and Shrikant Tangade, "A survey on authentication schemes in VANETs for secured communication," *Vehicular Communications*, vol. 9, pp. 19-30, 2017. Article (CrossRef Link)

[6] Maxim Raya and Jean-Pierre Hubaux, "Securing vehicular ad hoc networks," *Journal of Computer Security*, vol. 15, no. 1, pp. 39-68, 2007. Article (CrossRef Link)

[7] Yipin S., Rongxing L., Xiaodong L., Xuemin (Sherman) S. and Jinshu S., "An Efficient Pseudonymous Authentication Scheme with Strong Privacy Preservation for Vehicular Communications," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 7, pp. 3589 – 3603, 2010. Article (CrossRef Link)

[8] Rongxing Lu, Xiaodong Lin, Haojin Zhu, Pin-Han Ho and Xuemin (Sherman) Shen, "ECPP: Efficient Conditional Privacy Preservation Protocol for Secure Vehicular Communications," in *Proc. of the IEEE INFOCOM, the 27th Conference on Computer Communications, Phoenix, AZ, USA*, pp. 1903-1911, April 13-18, 2008. Article (CrossRef Link)

[9] Albert Wasef and Xuemin (Sherman) Shen, "EMAP: Expedite Message Authentication Protocol for Vehicular Ad Hoc Networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 1, pp. 78-89, 2013. Article (CrossRef Link)

[10] Ubaidullah Rajput, Fizza Abbas and Heekuck Oh, "A Hierarchical Privacy Preserving Pseudonymous Authentication Protocol for VANET," *IEEE Access*, vol. 4, pp. 7770 – 7784, 2016. Article (CrossRef Link)

[11] Xiaodong Lin, Xiaoting Sun, Pin-Han Ho and Xuemin Shen, "GSIS: A Secure and Privacy-Preserving Protocol for Vehicular Communications," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6, pp. 3442 – 3456, 2007. Article (CrossRef Link)

[12] Giorgio Calandriello, Panos Papadimitratos, Jean-Pierre Hubaux and Antonio Lioy "Efficient and Robust Pseudonymous Authentication in VANET," in *Proc. of the fourth ACM international workshop Montreal, Quebec, Canada*, pp. 19-28, September 10 – 10, 2007. Article (CrossRef Link)

[13] Lei Zhang, Qianhong Wu, Agusti Solanas and Josep Domingo-Ferrer, "A Scalable Robust Authentication Protocol for Secure Vehicular Communications," *IEEE Transactions on Vehicular Technology*, vol. 59, no. 4 pp. 1606-1617, 2010. Article (CrossRef Link)

[14] Kyung-Ah Shim, "CPAS: An Efficient Conditional Privacy-Preserving Authentication Scheme for Vehicular Sensor Networks," *IEEE transactions on vehicular technology*, vol 61, no. 4, pp. 1874-1883, 2012. Article (CrossRef Link)

[15] Zhang Jianhong, Xu Min and Liu Liying, "On the Security of a Secure Batch Verification with Group Testing for VANET," *International Journal of Network Security*, vol. 16, no. 5, pp. 355-362, 2014. Article (CrossRef Link)

[16] Cheng-Chi Lee and Yan-Ming Lai, "Toward a secure batch verification with group testing for VANET," *Wireless Networks*, vol. 19, no. 6, pp. 1441-1449, 2013. Article (CrossRef Link)

[17] Joseph K. Liu, Tsz Hon Yuen, Man Ho Au and Willy Susilo, "Improvements on an authentication scheme for vehicular sensor networks," *Expert Systems with Applications*, vol. 41, no. 5, pp. 2559-2564, 2014. Article (CrossRef Link)

[18] Debiao He, Sherali Zeadally, Baowen Xu and Xinyi Huang "An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2681-2691, 2015. Article (CrossRef Link)

[19] Hong Zhong, Jingyu Wen, Jie Cui and Shun Zhang, "Efficient Conditional Privacy-Preserving and Authentication Scheme for Secure Service Provision in VANET," *Tsinghua Science and Technology*, vol. 21, no. 6, pp. 620-629, 2016. Article (CrossRef Link)

[20] Hong Zhong, Bo Huang, Jie Cui, Yan Xu and Lu L.iu, "Conditional Privacy-Preserving Authentication Using Registration List in Vehicular Ad Hoc Networks," *IEEE Access*, vol 6, pp. 2241-2250, 2017. Article (CrossRef Link)

[21] Yang Ming and Xiaoqin Shen, "PCPA: A practical certificateless conditional privacy preserving authentication scheme for vehicular ad hoc networks," *Sensors*, vol. 18, no. 5, pp. 1573. 2018. Article (CrossRef Link)

[22] Ismaila Adeniyi Kamil and Sunday Oyinlola Ogundoyin "An improved certificateless aggregate signature scheme without bilinear pairings for vehicular ad hoc networks," *Journal of information security and applications*, vol. 44, pp. 184-200, 2019. Article (CrossRef Link)

[23] Sattam S. Al-Riyami and Kenneth G. Paterson, "Certificateless public key cryptography," in *Proc. of Springer International conference on the theory and application of cryptology and information security, Berlin, Heidelberg*, pp.452-473, November, 2003. Article (CrossRef Link)
[24] Jie Cui, Xuefei Tao, Jing Zhang, Yan Xu and Hong Zhong, "HCPA-GKA: A hash function-based conditional privacy-preserving authentication and group-key agreement scheme for VANETs," *Vehicular communications*, vol. 14, pp.15-25, 2018. Article (CrossRef Link)
[25] Hui Cui, Robert H. Deng and Guilin Wang, "An Attribute-Based Framework for Secure Communications in Vehicular Ad Hoc Networks," *IEEE/ACM Transactions on Networking*, vol 27, no. 2, pp. 721-733, 2019. Article (CrossRef Link)
[26] Shunrong Jiang, Xiaoyan Zhu and Liangmin Wang, "An Efficient Anonymous Batch Authentication Scheme Based on HMAC for VANETs," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 8, pp. 2193-2204, 2016. Article (CrossRef Link)
[27] Chuanhua Zhou, Gemei Zhu, Baohua Zhao and Wei Wei, "Study of One-way Hash Function to Digital Signature Technology," in *Proc. of the IEEE International Conference on Computational Intelligence and Security, Guangzhou, China*, pp. 1503-1506, November 03-06, 2006. Article (CrossRef Link)
[28] Bin Fan, David G. Andersen, Michael Kaminsky and Michael D. Mitzenmacher, "Cuckoo Filter: Practically Better Than Bloom," in *Proc. of the 10th ACM International on Conference Sydney, Australia*, pp. 75-88, December 02 – 05, 2014. Article (CrossRef Link)
[29] Tan Soo Fun and Azman Samsudin, "A Survey of Homomorphic Encryption for Outsourced Big Data Computation," *KSII Transactions on Internet and Information Systems*, vol. 10, no. 8, pp. 3826-3851, 2016. Article (CrossRef Link)
[30] Martin and Martin, "Identity-Based Cryptography Standard (IBCS) #1: Supersingular Curve Implementations of the BF and BB1 Cryptosystems," *document RFC 5091*, 2007. Article (CrossRef Link)
[31] Adams, Cain, Pinkas and Zuccherato, "Internet x. 509 Public Key Infrastructure Time Stamp Protocol (TSP)," *document RFC 3161*, 2001. Article (CrossRef Link)

Murtadha A. Alazzawi, received the bachelor's degree and master's degree in Computer Science from Science College, University of Basrah, Iraq, in 2010, 2013 respectively. He is currently pursuing the Ph.D. degree in computer Science and Technology with Huazhong University of Science and Technology, China. He has five years of teaching experience and was working with Imam Al-Kadhum College (IKC) before taking study leave and coming to China. His research interests are security and privacy issues in VANET.

Hongwei Lu, received the B.Sc., M.Sc. and Ph.D. degrees from HUST, Wuhan, China. Currently, he is a professor at the School of Computer Science and Technology, HUST. His research interests are in security and privacy in ubiquitous computing and electronic commerce, with a focus on security protocol analysis, access control, and trust negotiation.

Ali A. Yassin, received his Bachelor and Master degrees from University of Basrah, Basrah, Iraq and his Ph.D. from Huazhong University for Science and Technology, Wuhan, Hubei, China. He is currently working as an assistant professor with the Computer Science Dept., Education College for Pure Science, University of Basrah, Iraq. His research interest includes Security of Cloud computing, Image processing, Pattern Recognition, Biometric, Data Integrity, DNA Cryptography, Steganography, Sharing Data, Graphical Password, QR Code, and Soft computing.

Kai Chen, received his M.S. degree in 2008 and PhD degree in 2012, in the School of Computer Science & Technology at Huazhong University of Science and Technology. His current research interests include computer network application, computer network security and computer network protocol analysis