

RESEARCH

Open Access



SD2PA: a fully safe driving and privacy-preserving authentication scheme for VANETs

Saad Ali Alfadhli^{1,2*} , Songfeng Lu^{3,6*}, Abdulaziz Fatani^{1,4}, Haider Al-Fedhly⁵ and Mahmut Ince¹

*Correspondence:

Saad_alfadhli@hust.edu.cn;
lusongfeng@hust.edu.cn

¹ School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China

³ Hubei Engineering Research Center on Big Data Security, School of Cyber Science & Engineering, Huazhong University of Science and Technology, Wuhan 430074, China

Full list of author information is available at the end of the article

Abstract

The basic idea behind the vehicular ad-hoc network (VANET) is the exchange of traffic information between vehicles and the surrounding environment to offer a better driving experience. Privacy and security are the main concerns for meeting the safety aims of the VANET system. In this paper, we analyse recent VANET schemes that utilise a group authentication technique and found important vulnerabilities in terms of driving safety. These systems also suffer from vulnerabilities in terms of management efficiency and computational complexity. To defeat these problems, we propose a lightweight scheme, SD2PA, based on a general hash function for VANET. The proposed scheme overcomes the non-safe driving problem that resulted from the critical driving area. Moreover, the vehicle authentication is only done once by the VANET system administrator during the vehicle's moving, so the authentication redundancy for the entire system is reduced and system management efficiency is enhanced. The SD2PA scheme also provides anonymity to protect the vehicle's privacy, unless an important action needs to be taken against a malicious vehicle. A deep computational cost and communicational overhead analysis indicates that SD2PA is better than related schemes, as well as efficiently meeting VANET's security and privacy needs.

Keywords: VANET, Authentication, Privacy, Anonymity, Hash function, DSRC

Introduction

The Vehicular Ad-Hoc Network (VANET) is a subset of the wireless network. It is made from the Mobile Ad-Hoc Network (MANET) principle, through which vehicles within communication range can wirelessly exchange traffic-related data and other supplemental information under a transportation system [1]. The aim of VANET is to enhance navigation and transportation systems to increase trustworthiness and safety in the transportation environment. VANETs are efficient as long as they deliver travel efficiency and safety through real-time information assistance by launching connections vehicle to vehicle (V2V) and vehicle to road-side infrastructure (V2I) that can significantly enhance the driving experience through smart controls and offer higher relaxation and travel experience for travelers [2].

A VANET mainly consists of a trusted authority (TA), roadside units (RSUs) and vehicles equipped with on-board units (OBUs) for V2V and V2I communications duty. OBUs communicate with each other and with RSUs through a wireless public channel using the dedicated short-range communication (DSRC) protocol that applies the IEEE 802.11p standard for wireless communication, and RSUs connect to TA via a wired channel [3, 4]. The TA is a large storage capacity and high computational power trusted third party; it is responsible for generating and managing the system parameters and issuing secret tackles. A RSU is a communication bridge party that has better computation abilities and memory capacity than OBUs; it is deployed as road-side infrastructure to play specific management and coordination roles. Some VANETs use a Tamper-Proof Device (TPD) attached to OBUs or RSUs. A TPD is fully secured and used to store and calculate sensitive data. [5]. The general architecture of a VANET scheme is shown in Fig. 1. According to the DSRC protocol, OBUs broadcast messages each 100–300 ms, traffic-related messages consist of vehicle speed, position, congestion state, current time, track, and so on. With the help of this information, the system can offer an ideal solution for vehicle route and safety [6, 7].

As a result of the worthy traffic safety and efficiency solutions enabled by launching a VANET, and due to the nature of the open wireless communication used in VANETs, the message exchange in VANETs may be subject to the security risk of data interception, detection, modification and replication by malicious challengers. Thus, a strong mechanism for identity authentication and message integrity is the success key to certify the security of VANETs [8, 9]. Any defect in the authentication mechanism, a malicious vehicle may cause serious disturbance for the traffic through impersonating another true\ valid vehicle or even alter true messages to broadcast fake messages for the nearby vehicles to gain illegal benefits [10].

However, securing strong privacy is another important issue for VANETs. The real identity, location and route of a particular vehicle should not be attainable by the malicious vehicle [11]. Any specific leakage in the vehicle’s information, like the route information, may cause Serious consequences where that information may be used for traffic or criminal accidents by malicious vehicles. Although the vehicle’s privacy should be completely protected in the VANETs, these systems should consider conditional privacy, in which the message sender vehicle usually should involve a piece of

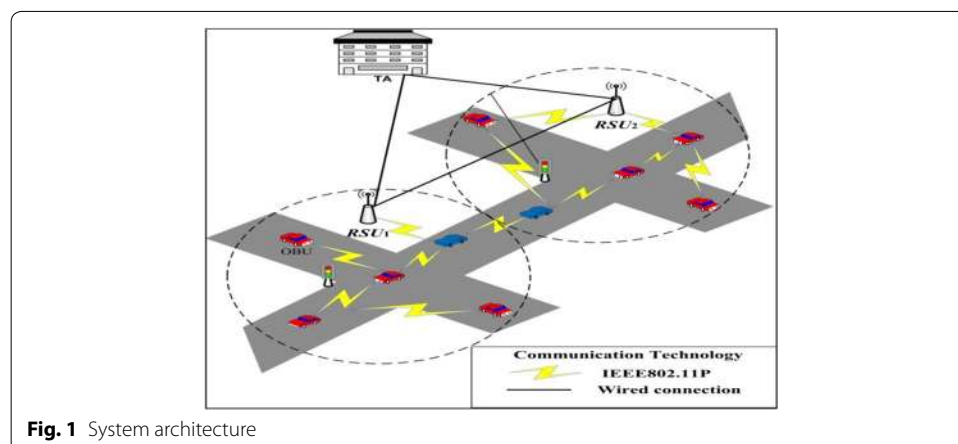


Fig. 1 System architecture

non-linkable information such that only TA can identify the message sender vehicle whenever it required [12–15]. For example, if malicious vehicles start to disturb the system (e.g., sending malicious messages), then the TA should be able to trace this vehicle and take a proper revocation action required. Hence, the conditional privacy-preserving authentication (CPPA) mechanism [8], which can offer both conditional privacy and message authentication is properly able to satisfy the VANETs security and privacy requirements. In summary, three main issues must be well considered in VANET schemes: security, efficiency, and preservation of conditional privacy before such systems are deployed in practice.

Over the past decade, VANET-related systems have attracted massive interest from academia, governmental organizations and industry. Industrial and academic groups have done many VANET-related studies, which have yielded many valuable accomplishments in different related aspects [16–20]. However, although earlier offered authentication schemes could solve some security and privacy issues in VANETs, additional studies are still needed to solve other important issues such as driving safety, system performance, security and privacy. Lately, interesting CPPA VANET schemes have been proposed by various groups [21–23]. However, we have discovered important vulnerabilities in these schemes in terms of driving safety, efficiency and performance. We, therefore, propose an efficient CPPA scheme for VANET that successfully handles the aforementioned issues and satisfies the needs and goals of VANETs.

The main contributions of our SD2PA scheme are summarized below.

- We propose a novel efficient and lightweight VANET scheme that overcomes a critical driving area problem found in existing group-based authentication schemes. A general hash function has been adapted during data transitions for the entire system without requiring heavy-weight bilinear pairings or Elliptic-curve cryptography (ECC).
- We made a detailed security analysis and demonstrate that our scheme meets the security requirements for VANETs.
- We have enhanced the vehicle authentication process so that it does not cause a bottleneck for the TA.
- We have mitigated the communication and computation weights for the available schemes.

The motivation of this paper is to give a comprehensive view of the VANET system and its components. Due to the heterogeneous nature and the dynamic topology of this network caused by the fast-moving of the vehicles, the need for a fast-real-time messages exchange and response, and the power/memory limitation especially in the OBUs, make the VANETs under different security and privacy issues. Therefore, in this paper, we highlight the major security and privacy challenges and concerns of VANETs, also, we give a review of some security, privacy, and efficiency vulnerabilities in some existing schemes proposed in related literature. However, the main motivation of this paper is to identify an important weakness in terms of safe driving—within the scope of security mechanism related to real-time V2V traffic-related beacons exchange—in some of the existing group signature/verification based VANET schemes. Finally, we present

our lightweight SD2PA scheme that can satisfy the security and privacy needs of the VANETs and overcome the aforementioned issues.

The remainder of this paper is ordered as follows: “[Related works](#)” section, summarizes some previous related works. “[Preliminaries](#)” section describes the Preliminaries of our proposed scheme, while the “[Review of the critical driving area problem](#)” section identifies and discusses a critical driving area problem in recent related VANET schemes. “[The proposed scheme](#)” section provides a detailed explanation of our proposed scheme. “[Security analysis](#)” gives a detailed security analysis, while the “[Performance analysis](#)” discuss the performance of our approach and show its enhancements with compare to other related schemes, while the “[Conclusion](#)” section presents the conclusion of this paper.

Related works

Extensive studies have been proposed to improve the driving safety and system efficiency of VANETs. In general, three main categories of VANET schemes have been proposed in the literature: schemes based on either public key infrastructure (PKI) or group signature and identity-based schemes.

The authors in [24, 25] have proposed CPPA schemes based on PKI and used pairs of public/private keys and matching certificates to protect the identity of the vehicle. However, this has two evident defects: first, the vehicle’s OBU requires a big storage capacity to store the pairs of certificates and private/public keys; second, the TA needs to perform a full cross-check in its storage area while searching for the real identity of the challenger, which is a time-consuming process and increases memory overheads. Such VANET schemes suffer from a storage and certificate management bottleneck problem.

In the group-based signature VANET schemes, a number of vehicles comprise one group, and each vehicle within a specific group must have its own private key and a public key shared with the other group members. Such authentication schemes will face critical driving problems. Interesting schemes have been presented in [26–29]; however, in this model, the sender vehicles sign their messages with their private keys, and the vehicles receiving the messages use the corresponding public keys to verify and validate the message. However, high-speed vehicle movement, as well as the rapid changes in VANET topology, creates many management challenges in group manager election and group member management [30]. The group signature is also heavier than a simple signature, which makes the communication cost, computation weight and signature verification not efficient for VANETs.

To deal with the above-mentioned problems, ID-based authentication schemes have been proposed for VANET systems. Zhang et al. [31] proposed an ID-based CPPA VANET scheme based on bilinear pairing. In this scheme, RSUs and vehicles use a pseudo-identity as a public key, while the private key generator (PKG) generates the private keys. Unlike the PKI schemes, it avoids the need to generate, manage and store a large number of certificates in the entities. Chim et al. [32] proved that the scheme proposed by [31] is vulnerable to anti-traceability resistance and impersonation attack and proposed a new secured VANET scheme. Horng et al. [33] showed that the scheme in [32] is vulnerable to impersonation attacks and proposed a new secured scheme to overcome the problem in [32]. Shim [34] suggested a security ID-based CPPA scheme,

wherein batch message verification is supported by the RSU to enhance its computation overhead in case the number of messages is high. However, to retrieve the entire revocation list, the TA needs to consume more time; it also does not consider the authentication overheads affected by illegal materials. Moreover, Liu et al. [35] showed that the security level in the ID-based signature scheme in [34] does not satisfy the security requirements and is vulnerable to modification attacks.

Numerous other ID-based CPPA schemes such as [35–39] have been proposed that claim to guarantee the privacy-preserving and security requirements. However, the designs of these schemes are based on bilinear pairings, which, due to their heavy computational cost, are not efficient enough for VANETs. Based on the ECC, numerous ID-based CPPA schemes such as [40–42] have been proposed. Although these schemes offer better performance than those that use the bilinear pairing technique, due to the nature of the VANET nodes and system efficiency requirements, these schemes do not satisfy the ideal VANET performance effectiveness. However, although ID-based CPPA schemes could overcome the PKI's computational, communication and management issues, they are vulnerable to system key escrow, insider attacks and batch verification challenges. In such schemes, the private system key is known to all vehicles, so any insider attack could broadcast a fake beacon on behalf of any other vehicle, and any violation in the system key would also damage the entire system.

Recent attention-grabbing studies have been proposed to deal with the VANET authentication issues. Jie Cui et al. [21] proposed a novel edge-computing concept for a CPPA VANET scheme. Jie Cui et al. [22] proposed a secure hash function-based and group-key agreement CPPA VANET authentication scheme. Jie Cui et al. [23] proposed a VANET pseudonym-based authentication CPPA Scheme with cuckoo filter. Unfortunately, we found that these schemes [21–23] were vulnerable to important problems in terms of safety, efficiency and computational cost.

In [21], each RSU has to choose the higher computational resources and closer location of one or more vehicles from among the in-range vehicles to play the role of an edge layer between the RSU and other ordinary vehicles. We know that in VANET networks we cannot guarantee the availability of those kinds of high computational vehicles, and the high speeds of moving vehicles and the fast-changing VANET topology also create difficulties for the RSU to frequently choose special vehicles for edge computing cooperation.

In [22], a scheme based on a group key agreement mechanism is proposed for vehicle authentication, but again, the high vehicle speed and fast-changing network topology create difficulties in group key regeneration, in which the TA needs to regenerate a secret group key each time a vehicle joins or leaves the group, causing a bottleneck for the TA.

In [23], the proposed scheme uses heavy and ineffective signature verification procedures. We also found that the presented schemes [21–23] have a serious critical driving area between every two RSUs that may create disastrous accidents and risks.

In this paper, we identify the critical driving area problem available in aforementioned schemes. Besides, we propose a novel and efficient lightweight pseudo-identity-based CPPA VANET solution that overcomes the critical driving area and system key escrow problems, as well as offering better performance in terms of computation cost

and communications overhead for the entire VANET system. In addition, our scheme can easily prevent an attacker or a trusted vehicle from continuing to send malicious or fake beacons. Finally, our scheme overcomes the batch verification problems and TA bottlenecking.

Preliminaries

In this section, we give a brief overview of the system model, assumptions, and goals, in addition to the hash function and the cuckoo filter that we used in our proposed scheme. Some notation definitions are shown in Table 1.

System model and components description

The main VANET architecture components in our scheme consist of three items: TA, RSU and OBU, as shown in Fig. 1.

1. TA: The TA is a trusted third-party centre that is responsible for registering and managing all of the RSUs and OBUs on the network, and it never gets compromised [43]. We assume that the TA and RSUs use secured wired channels and secured transmission protocol, like the wired Transport Layer Security (TLS) protocol. However, redundant TAs can be installed to avoid a failure point or bottleneck [44].

Table 1 Notations

Notation	Description
RID	Vehicle real identity
$h(.)$	General hash function
TA	Trusted authority
RSU	Roadside unit
OBU	On-board unit
L_{TA-OBU}	List of OBUs information saved in the TA
$L_{RSU-OBU}$	List of RSU–OBU authentication data saved in the RSU
PWD_1	OBU login password
PWD_2	OBU – TA verification password
C_1, C_2	Critical areas between two RSUs
V_i	The i th vehicle
CR_V	V_i 's conventional public authentication key
$Tlist$	A temporary list in the RSUs for saving vehicle information, which is in the critical area
ID_R	The identity of the RSU
ID_{R+1}	The identity of the neighbouring RSU, which is next to the vehicle's current RSU
ID_{R-1}	The identity of the neighbouring RSU, which is prior to the vehicle's current RSU
S_{R-V}	OBU – RSU authentication secret key
SK_{R-TA}	RSU – TA authentication secret key
T	The timestamp
$indx$	The vehicle's group index number in the $L_{RSU-OBU}$
PID_i	V_i 's pseudonym
\parallel	Concatenation operation
\oplus	Exclusive-OR operation
int	Integer number
M_i	V_i 's traffic-related message
L	Hash signature calculated by the RSU during the authentication phase

2. **RSUs:** The RSU is trusted and difficult to compromise. RSUs enjoy better computational power than OBUs. They act as an intermediate communication and management bridge between TAs and OBUs. The RSU–OBU communication range is not less than double the inter-vehicle broadcasting range so that the RSU can ensure that, when it receives messages, all vehicles that received the message will be within its communication and notification range. RSUs can cooperate using a secured communication channel [45]. We assume that RSUs are regionally well distributed according to the real need in a way that guarantees the full interaction between RSUs.
3. **Vehicles:** The vehicles are equipped with an OBU that supports the DSRC protocol; they communicate with each other, as well as with the RSU wirelessly using the OBU. Note: With the help of DSRC, the inter-vehicle broadcasting range can extend across a few hundred meters [46].

Security objectives

Recent works of interest [47–49] presented necessary security goals in related systems. However, due to the unique characteristics of VANET environment [15, 24], a well-designed CCPA scheme should satisfy the security goals below:

1. **Message integrity and authentication:** A system of nodes or vehicles has to be able to confirm that the received messages are signed by the sender vehicle and have not been modified.
2. **Identity privacy preserving:** An unauthorised malicious object should not be able to recognise or determine the vehicle's real identity by considering several messages broadcast by the same sender vehicle.
3. **Traceability and revocability:** Although the vehicle's identity must be hidden from ordinary message receivers to keep the sender's privacy and security safe, the TA has to be able to find the real identity in case tracing or revocation action is required.
4. **Non-repudiation:** The vehicle cannot repudiate a message that it sent.
5. **Un-link-ability:** The attacker must not be able to recognise the sender vehicle from the content of the message sent by the same vehicle.
6. **Resistance to attacks:** VANET system design should consider resistance to attacks, like impersonation, replay and modification attacks.
7. **Lightweight:** The nature of VANET scheme topology requires taking into account huge data exchange, fast vehicle mobility and the ordinary processing abilities of the OBUs to create lightweight computation costs and communication overhead, which are the keys success for any VANET scheme.

The one-way hash function

$h(.)$ is said to be secure when it satisfies the following properties [43]:

1. $h(.)$ can take a random-length message as input and give a fixed-length message output.

2. Finding $k=h(n)$ from a certain x is easy. However, it is difficult to find $n=h^{-1}(k)$ from a certain k .
3. Computationally, for a given n , it is impractical to detect $n' \neq n$ such that $h(n') \neq h(n)$.

Cuckoo filter

The cuckoo filter is a new and efficient data structure that offers ideal performance, more accurateness, and lesser false positives than the other available filters. The cuckoo filter supports dynamic addition and removal, which results in high performance [50]. It is fundamentally a hash table that contains a series of cells, and each cell has a fixed number of entries –the hash function along with a lower-bit output. For a piece of data, d , the hashing function finds the index of two candidate buckets, Ind_1 and Ind_2 , according to the following:

$$Ind_1 = h(d) \bmod N \tag{1}$$

$$Ind_2 = (Ind_1 \oplus h(Fingerprint(d))) \bmod N \tag{2}$$

where N is the size of the cuckoo filter. If there is a free bucket from the existing candidate buckets, then we store the fingerprint in it. Otherwise, we pick an existing item from a selected candidate bucket and re-insert it into its alternating buckets; this process is continued until finding a free bucket, as shown in Fig. 2, which shows $h_1(d)$ and $h_2(d)$ assigned to buckets that allocated already. To check whether item d is in the cuckoo filter, we compute $Fingerprint(d)$ and the two linked buckets using (1) and (2). If the result of $Fingerprint(d)$ matches any of the two linked buckets, then the cuckoo filter returns true, otherwise, it returns false. Therefore, utilizing this filter in VANETs can play an efficient notification or verification role through the fast saving and retrieving hashed signatures during the real-time message exchange. Thus, it can enhance the overall system performance efficiency.

Review of the critical driving area problem in the [21–23] CCPA schemes

In the schemes proposed by Jie Cui et al. [21–23], the vehicle needs to be frequently authenticated by the TA at each time it leaves one RSU\group to join the next RSU\group. In this situation, we found that a critical driving area appears in those systems;

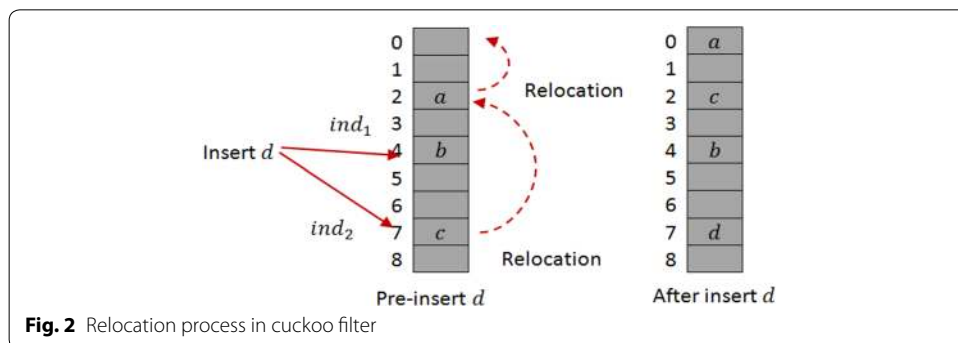


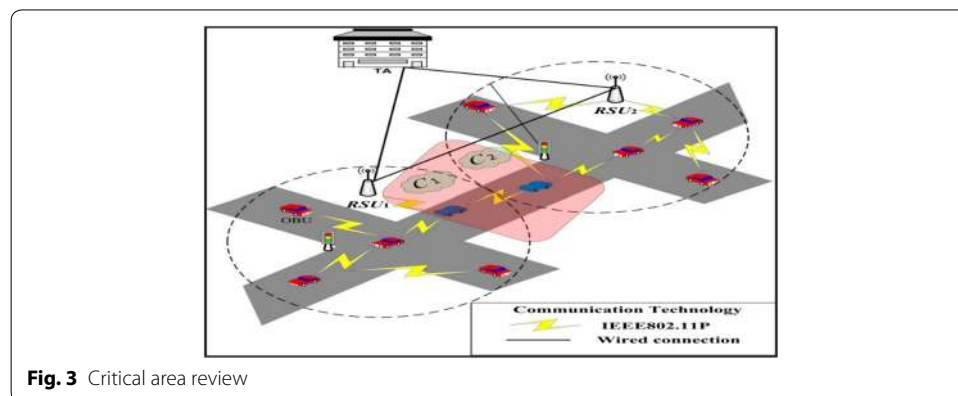
Fig. 2 Relocation process in cuckoo filter

that is, when a vehicle A moves from the range of one RSU group to the range of a new RSU group, it can be authenticated and exchange beacons only when it has already entered a new RSU group. At the same time, vehicle A will not be authenticated and cannot exchange beacons with nearby vehicles in the old RSU group once it joins the new RSU group. This creates a risky, critical and non-safe driving area, where valid beacons from trusted vehicles will not be accepted by nearby vehicles in the neighbouring RSU , which may result in tragic accidents. As shown in Fig. 3, in schemes [21–23] the vehicles within the range of RSU_1 are only authenticated in RSU_1 , and the vehicles within the range of RSU_2 are only authenticated in RSU_2 , and so forth. The vehicles in the area C_1 and the vehicles in the area C_2 will, therefore, ignore each other's beacons and cannot exchange beacons, which makes those areas (C_1, C_2) critical driving areas that could cause VANET safety failures. Therefore, in this paper, driving safety-related term issues fall within the scope of the security-related issues (C_1, C_2). Where the safe driving aim for VANETs can only be achieved through considering all the valid V2V real-time traffic-related beacon.

The proposed scheme

In this section, we introduce SD2PA. Our scheme consists of two main parts. The first part is the system initialisation and setup, which is offline and done only once unless there is a need for a system update. Whereas, the second part is an online vehicle authentication procedure and navigation management.

In our SD2PA scheme, the TA arranges the system materials and assigns the setup parameters and data to the system members (RSUs and OBUs). It also can allow valid vehicles to join the VANET, trace, and revoke any misbehavior vehicle. To join the VANET, each OBU needs to trigger a mutual authentication handshake with the TA. Thereafter, using a predefined OBU-RSU secret key, a joined OBU can broadcast signed traffic-related beacons. From a receiving nodes side: the concerning RSU will be in charge of verifying the signatures. Besides, utilising the Cuckoo filter, it stores the positive and negative fingerprints of the valid and non-valid OBUs signatures. Each RSU broadcasts the Cuckoo filter with a notification message for all the in-range OBUs. In which the receiving OBUs can, efficiently, validate the received traffic-related beacons by considering the positive and negative fingerprints in the Cuckoo filter obtained from



the notification message received from the RSU. A detailed explanation in the following subsections.

System initialization and setup phase

This subsection demonstrates the initialisation setup process, carried out by the TA.

1. TA initialization

In SD2PA scheme, the following initialisation procedures will be done once during the system life cycle, unless there is a need for an update.

- The TA selects the hash function $h(\cdot)$.
- The TA shares the system hash function to all RSUs and OBUs during their registration to the VANET.
- The TA assigns a unique identity of RSU ID_R and secret key SK_{R-TA} for every RSU, as well as sending $\{ID_{R-1}, ID_{R+1}\}$, where (ID_{R-1}, ID_{R+1}) represent the identities of the previous and next neighbour for the current RSU.

2. Vehicle registration

In this phase, the TA assigns $\{RID_i, PWD_1, PWD_2, CR_i, S_{R-V}\}$ to the OBU for each vehicle V_i , where RID_i is the V_i 's real identity, PWD_1 is the V_i 's password, $PWD_2 \in Z^*$ is a secret key known only to the OBU and the TA, $CR_i \in Z^*$ is a conventional public key and $S_{R-V} \in Z^*$ is a unique authentication secret key to be used for verification purposes between the RSU and OBU during the V_i broadcasting phase. After that, the TA saves $\langle RID_i, PWD_1, PWD_2, CR_i, S_{R-V} \rangle$ into the registration list L_{TA-OBU} and provides $\langle RID_i, PWD_1 \rangle$ to the vehicle's owner. Those procedures are to be done once and offline.

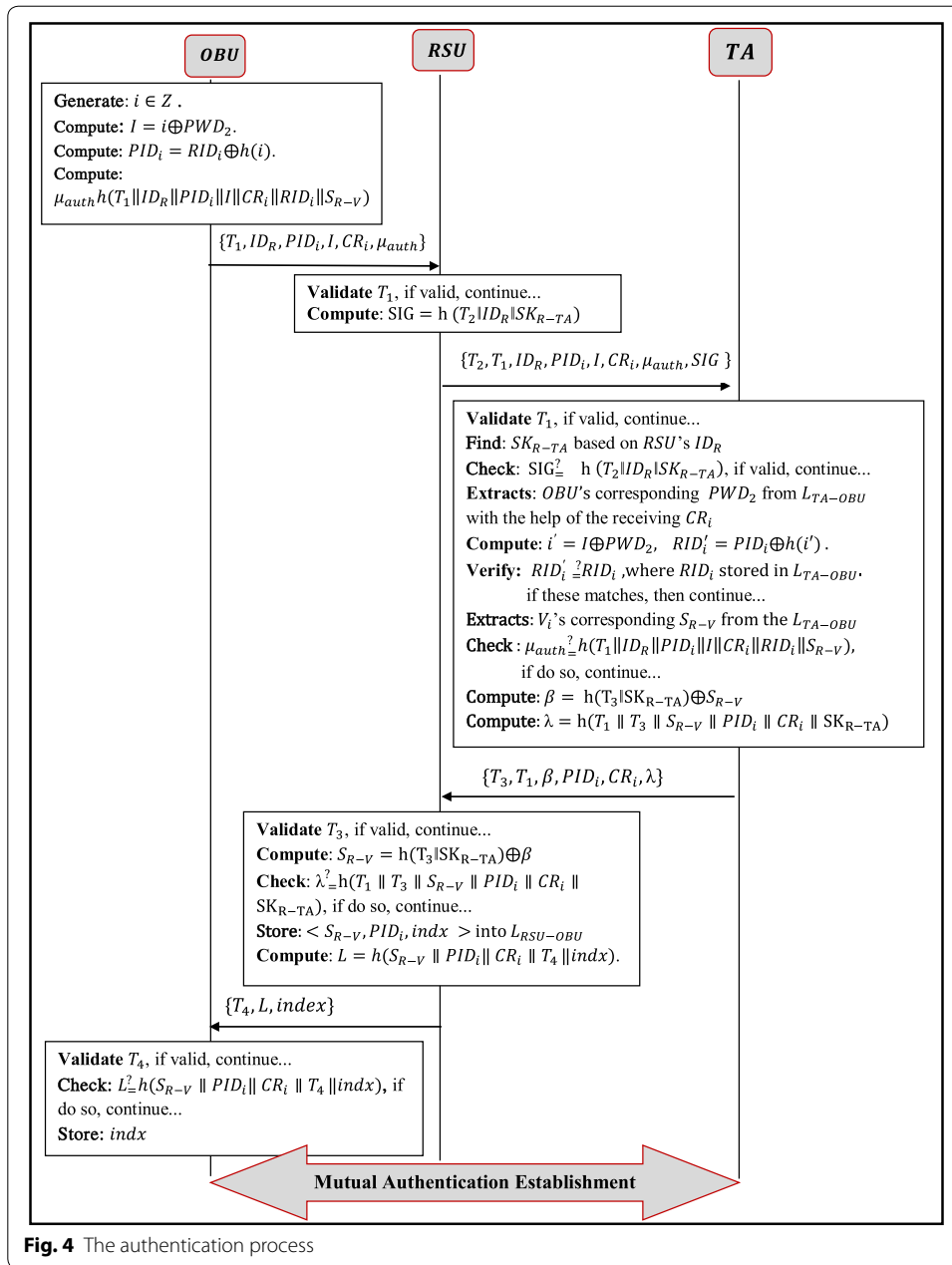
Authentication phase

All of the RSUs in the VANET broadcast the ID_R periodically; whenever a vehicle starts working, it needs to be authenticated to join the system through the following handshake procedure. Figure 4 illustrates the procedure of this phase.

1. Vehicle V_i 's owner has to enter the identity RID_i and passwords PWD_1 to start the OBU, then the OBU checks whether RID_i and PWD_1 match the stored verification data; if they do, it then proceeds to the following steps:

- The OBU generates a random number $i \in Z$ and then computes I and PID_i , where $I = i \oplus PWD_2$ and $PID_i = RID_i \oplus h(i)$.
- The OBU computes $\mu_{auth} = h(T_1 ID_R PID_i I CR_i RID_i S_{R-V})$, then sends $\{T_1, ID_R, PID_i, I, CR_i, \mu_{auth}\}$ to the current RSU.

2. Once the current RSU receives the message $\{T_1, ID_R, PID_i, I, CR_i, \mu_{auth}\}$ from the OBU, it checks the validity of the timestamp T_1 . The timestamp T is said to be valid if the $\Delta T > T_{rs} - T$, where T_{rs} is the receiving timestamp and ΔT is a predefined time difference. If the timestamp T_1 is valid, then the RSU will store the data



$\langle T_1, PID_i, CR_i \rangle$ into the temporary list L_{tmp} and calculates $SIG = h(T_2 || ID_R || SK_{R-TA})$ forward the message $\{T_2, T_1, ID_R, PID_i, I, CR_i, \mu_{auth}, SIG\}$ to the TA, where T_2 is the timestamp for the RSU.

- Once the TA receives the message $\{T_2, T_1, ID_R, PID_i, I, CR_i, \mu_{auth}, SIG\}$ from the RSU, it checks the validity of the timestamp T_2 ; if it is valid, it then proceeds to the steps below.

- According to the receiving *RSU*'s ID_R , the TA finds the corresponding SK_{R-TA} from its repository and examine the *RSU*'s signature $SIG_{=}^2 h(T_2 || ID_R || SK_{R-TA})$. If invalid, it drops the message, otherwise, it performs the coming steps.
 - Based on the receiving *OBU*'s key, CR_i , the TA extracts the corresponding secret key PWD_2 from the registration list L_{TA-OBU} and calculates $i' = I \oplus PWD_2$ and $RID'_i = PID_i \oplus h(i')$.
 - The TA matches RID'_i with the supposed V_i 's RID_i that is already stored in L_{TA-OBU} ; if these match, then the TA extracts the V_i 's corresponding S_{R-V} and checks the equation $[\mu_{auth}^2 = h(T_1 || ID_R || PID_i || I || CR_i || RID_i || S_{R-V})]$. If this is valid and the V_i is not in the revocation list, the TA calculates $\beta = h(T_3 || SK_{R-TA}) \oplus S_{R-V}$ then sends the message $\{T_3, T_1, \beta, PID_i, CR_i, \lambda\}$ to the *RSU*, where $\lambda = h(T_1 || T_3 || S_{R-V} || PID_i || CR_i || SK_{R-TA})$.
4. Once the *RSU* receives the message $\{T_3, T_1, \beta, PID_i, CR_i, \lambda\}$, it checks the timestamp T_3 ; if it is valid, then, the *RSU* extracts the *RSU-OBU* authentication key S_{R-V} from the equation $S_{R-V} = h(T_3 || SK_{R-TA}) \oplus \beta$, and checks if $\lambda_{=}^2 = h(T_1 || T_3 || S_{R-V} || PID_i || CR_i || SK_{R-TA})$. If valid, it retrieves the corresponding data $\langle T_1, PID_i, CR_i \rangle$ from L_{tmp} . It then stores $\langle S_{R-V}, PID_i, indx \rangle$ into its *RSU-OBU* authentication list $L_{RSU-OBU}$. Note: we assume that the coverage capacity for each *RSU* is 400 to 600 vehicles. For efficient performance, we propose adding a group index ($indx$) for every 30 *OBUs* on the $L_{RSU-OBU}$. That is, we add one group index $indx$ for every 30 *OBUs* within the *RSU*, although the group amount is adjustable by *RSU*. In another word, we propose to repeat the same $indx$ (row) number for the true vehicles on the $L_{RSU-OBU}$ in our case, we repeat the same $indx$ number for every 30 rows on the $L_{RSU-OBU}$ so that, we can achieve an efficient performance as well as avoiding any traceability or likability attempt of an adversary vehicle or attacker. Finally, the *RSU* sends the message $\{T_4, L, indx\}$ to the *OBU*, where $L = h(S_{R-V} || PID_i || CR_i || T_4 || indx)$.
 5. Once the *OBU* receives the message $\{T_4, L, indx\}$, it checks the timestamp T_4 , if it is valid, it then checks $L_{=}^2 = h(S_{R-V} || PID_i || CR_i || T_4 || indx)$. If this holds, it then stores the $indx$ into its repository. This completes the authentication handshake process, and the vehicle can broadcast beacons.

Broadcasting phase

When a vehicle, V_i , wants to broadcast a beacon, the *OBU* calculates a signing key $\gamma = h(T_5 || PID_i || S_{R-V} || M_i || ID_R || indx)$ and broadcasts the beacon $\{\gamma, T_5, M_i, ID_R, indx\}$.

Verification phase

According to the location of the sender vehicle, V_i , there are two possible verification cases, as shown below.

Case 1: If the vehicle V_i broadcasts beacons while it is not inside the critical area, then the *RSU* does the following steps.

1. When the RSU receives the beacon $\{\gamma, T_5, M_i, ID_R, indx\}$, it checks the timestamp T_5 and the ID_R . If the ID_R is for itself not for any neighbouring RSUs and the timestamp is valid, it then examines the legitimacy of the OBU by verifying the receiving signed key $\gamma = h(T_5 \| PID_i \| S_{R-V} \| M_i \| ID_R \| indx)$ with the help of the authentication data $\langle S_{R-V}, PID_i, indx \rangle$ already stored in $L_{RSU-OBU}$. According to the $indx$, in the worst case, the RSU needs to identify 30 items in the $L_{RSU-OBU}$. If valid, the RSU then stores $fingerprint(\gamma \| T_5)$ in the positive cuckoo filter; otherwise, it stores it in the negative filter. The positive filter holds the fingerprints of the legal vehicles and the negative filter holds the fingerprints of the illegal vehicles. Finally, the RSU broadcasts the filters with each notification message.
2. When a vehicle V_j receives the beacon $\{\gamma, T_5, M_i, ID_R, indx\}$ from vehicle V_i , it checks the ID_R . If this is valid for its current RSU and the timestamp is also valid, it then verifies V_i 's signature by computing its fingerprint $f_i = fingerprint(\gamma \| T_5)$. It then gets two locations, $Ind_{.1} = h(\gamma \| T_5) \bmod M$ and $Ind_{.2} = Ind_{.1} \oplus h(f_i) \bmod M$. Finally, it matches the result with the positive and negative filters received from the RSU broadcast. Four possible actions will be taken according to matching results shown in Table 2. Note: there is a very small probability of a false positive may occur in the cuckoo filter report [21, 47]. However, case 4 in Table 2 handles this issue.

Case 2: Sometimes vehicle V_i broadcasts beacons while it is within the range of a critical area C_1 , and the broadcasting range exceeds the range of vehicle V_j in critical area C_2 of the neighbouring RSU. The broadcasting range of vehicle V_j in the critical area C_2 will also exceed the range for vehicle V_i in the critical area C_1 . The vehicles on both sides of C_1, C_2 need to consider and verify each other's beacons for the safety reasons mentioned earlier. However, in our SD2PA scheme, when a vehicle V_i approaches critical area C_1 , its current RSU sends the information $\langle S_{R-V}, PID_i, ID_R, indx \rangle$ through a secured channel to the upcoming neighbour RSU, while the neighbouring RSU will do the same for vehicle V_j in critical area C_2 . The RSUs will temporarily store this information into the temporary list $Tlist$. According to the location of the beacon's recipient, there are two possible verification procedures, as shown below:

1. If the recipient is the vehicle in the same RSU as V_i , steps 1 and 2 in case 1 are sufficient for the verification.
2. If the recipient is a vehicle V_j in critical area C_2 , then V_j will follow the steps shown in the critical area verification phase.

Table 2 Validation status and results on cuckoo filter

Result case	Positive filter	Negative filter	Validity of γ
1	Yes	No	Valid
2	No	Yes	Invalid
3	No	No	Waiting mode
4	Yes	Yes	False query

Critical area verification phase

When a vehicle V_j in the area C_2 receives a beacon from vehicle V_i , it first checks the ID_R attached to the received beacon as well as the timestamp. Once it finds that the ID_R is not its current RSU and the timestamp is valid, it then verifies V_i 's signature by computing its fingerprint and checks the result with positive and negative filters already received from its RSU. In case of positive or negative matches, it follows the procedures from the first and second result cases mentioned in Table 2. Otherwise, it forwards the entire beacon to the current RSU to help the sender vehicle V_i be authenticated in the area C_2 while it is in C_1 . As soon as this RSU receives the forwarded message, it will go through the following steps:

1. The RSU checks the ID_R belonging to the neighbouring RSU. After that, it examines the legitimacy of the OBU by verifying the received signed key $\gamma \stackrel{?}{=} h(T_5 \| PID_i \| S_{R-V} \| M_i \| ID_R \| indx)$ with the help of the information (PID_i, S_{R-V}) from the vehicle V_i that is already stored in $Tlist$.
2. If the verification check is valid, then the RSU stores $fingerprint(\gamma \| T_5)$ in the positive cuckoo filter; otherwise, it stores it in the negative filter. Finally, the RSU broadcasts the filters with the notification message. The $Tlist$ will be vacated whenever the vehicle V_i enters this new RSU coverage area or after a specified period of time.

New RSU joining phase

When the vehicle V_i enters into the range of a new RSU, it will send a joining request message to the new RSU to get a new valid $indx$ in the new RSU authentication list $L_{RSU-OBU}$ as follows:

1. The OBU sends $\{T_1, \gamma\}$ to the RSU, where $\gamma = h(T_1 \| PID_i \| S_{R-V})$.
2. Once the new RSU receives the joining request message $\{T_1, \gamma\}$, it checks the timestamp. If this is valid, it then refers to $Tlist$ to verify the joining signature $\gamma \stackrel{?}{=} h(T_1 \| PID_i \| S_{R-V})$. If the joining signature is valid, it then proceeds to the next steps, otherwise, it ignores the request.
3. The new RSU shifts the relevant authentication information from $Tlist$ to $L_{RSU-OBU}$. Furthermore, it acknowledges the previous RSU of vehicle V_i and forwards the new $indx$ to it, so that the previous RSU shifts the relevant authentication data from $L_{RSU-OBU}$ to $Tlist$ and acts as a neighbouring RSU. At the same time, the new RSU sends the message $\{T_2, indx, L\}$ to V_i , where $L = h(T_2 \| indx \| PID_i \| S_{R-V})$.
4. Once the OBU receives the message $\{T_2, indx, L\}$, it checks the timestamp T_2 ; if this is valid, it verifies the equation $L \stackrel{?}{=} h(T_2 \| indx \| PID_i \| S_{R-V})$. If this, too, is valid, then the joining authentication is complete.

Vehicle revocation phase

When a trusted vehicle V_i broadcasts fake information, the TA in our scheme can efficiently find and revoke the vehicle's real identity as follows:

1. The RSU finds the authentication information $\{PID_i \parallel S_{R-V}\}$ in the $L_{RSU-OBU}$ that meets V_i 's signature $\gamma = h(T_5 \parallel PID_i \parallel S_{R-V} \parallel M_i \parallel ID_R \parallel indx)$ and sends the S_{R-V} to the TA.
2. According to the S_{R-V} , the TA extracts the RID_i corresponding to $V_i S_{R-V}$ from L_{TA-OBU} .
3. The TA adds the V_i authentication $\langle RID_i, S_{R-V} \rangle$ data into the revocation list and reports it to the RSU.
4. Once the RSU receives the revocation report from the TA, it deletes the V_i 's authentication $\langle PID_i, S_{R-V} \rangle$ from $L_{RSU-OBU}$, immediately preventing the malicious V_i from disturbing the system.

Security analysis

This section discusses the security of the SD2PA scheme. It firstly demonstrates that SD2PA scheme can meet all the goals mentioned in the preliminaries section through the informal security analysis. Moreover, it presents the formal security analysis proof between OBU and RSU in SD2PA scheme using BAN Logic [51].

Security discussion

In this subsection, we prove that the SD2PA scheme fulfills all the mentioned security goals and compare it with the other schemes as shown in Table 3.

1. Message integrity and authentication: In our scheme, the hash function $h(.)$ is applied to the message signature. According to the definition of the hash function $h(.)$, it is impossible to fabricate a valid beacon [43]. The secret key S_{R-V} is also attached to the hashed data of the beacons. With the help of the signature $\gamma = h(T_5 \parallel PID_i \parallel S_{R-V} \parallel M_i \parallel ID_R \parallel indx)$, the RSU can efficiently ensure the validity and integrity of the message. The SD2PA scheme, therefore, provides the desired message integrity and authentication properties.
2. Identity privacy preserving: In the SD2PA scheme, the beacon contains $\{\gamma, T_5, M_i, ID_R, indx\}$, in which there is no identity-related information that can be

Table 3 The security and privacy comparison

The security goals	[21]	[22]	[23]	SD2PA
Message integrity and authentication	✓	✓		✓
Identity privacy preserving	✓	✓	✓	✓
Traceability	✓	✓	✓	✓
Revocability	✓	X	✓	✓
Non-repudiation	✓	✓	✓	✓
Unlinkability	✓	✓	✓	✓
Resistance to modification attack	✓	✓	✓	✓
Resistance to replay attack	✓	✓	✓	✓
Resistance to impersonation attack	✓	✓	✓	✓
Lightweight	x	X	X	✓

used by an adversary to retrieve the vehicle's real identity. The scheme, therefore, offers the desired identity privacy-preserving property.

3. Traceability and revocability: According to the vehicle revocation phase mentioned earlier, the SD2PA scheme provides the desired traceability and revocability properties.
4. Non-repudiation: In the SD2PA scheme, the beacon contains $\{\gamma, T_5, M_i, ID_R, indx\}$, where $\gamma = h(T_5 \| PID_i \| S_{R-V} \| M_i \| ID_R \| indx)$. As the vehicle has to use the secret key S_{R-V} that is only known to the RSU, the vehicle cannot deny that it has sent the beacon in question. The SD2PA scheme, therefore, provides the desired non-repudiation property.
5. Unlinkability: In the SD2PA scheme, the vehicle broadcasts the beacon $\{\gamma, T_5, M_i, ID_R, indx\}$, which is different in each broadcasting operation. The ID_R will be the same for all the vehicles within the RSU, and the $indx$ will be the same for every 30 vehicles within the RSU. It is therefore difficult for an adversary to expect that two beacons belong to the same vehicle. The scheme, therefore, provides the desired unlinkability property.
6. Resistance to attacks:
 - Modification attack: In the SD2PA scheme, the beacon contains $\{\gamma, T_5, M_i, ID_R, indx\}$, where $\gamma = h(T_5 \| PID_i \| S_{R-V} \| M_i \| ID_R \| indx)$. The adversary must, therefore, have the secret key S_{R-V} , if he or she wants to modify the beacon, which means this scheme is able to resist this type of attack.
 - Replay attack: In the SD2PA scheme, the timestamp is attached to each beacon and is added to the hashed signature $\gamma = h(T_5 \| PID_i \| S_{R-V} \| M_i \| ID_R \| indx)$. It is therefore impossible for an adversary to replay the beacon, making this scheme resistant to this type of attack.
 - Impersonation attack: In the SD2PA scheme, the beacon contains $\{\gamma, T_5, M_i, ID_R, indx\}$, where $\gamma = h(T_5 \| PID_i \| S_{R-V} \| M_i \| ID_R \| indx)$. The adversary must have the secret key S_{R-V} , if he or she wants to impersonate the vehicle, making this scheme resistant to this type of attack.
7. Lightweight: In the SD2PA scheme, the beacon contains $\{\gamma, T_5, M_i, ID_R, indx\}$, where $\gamma = h(T_5 \| PID_i \| S_{R-V} \| M_i \| ID_R \| indx)$. Only the one-way hash function is used for security, so the computation and communication costs are reduced. The authentication process with the TA is also only needed once during the driving phase. More details will be discussed in the next section.

Mutual authentication proof

In this subsection, we prove the mutual authentication validity between OBU and RSU with the help of the widely used BAN logic technique. The analysis shows that SD2PA scheme can achieve the designed authentication goals. Table 4 explains the relevant notations in the BAN logic analysis.

Rules: The used rules for BAN logic analysis is shown below:

Table 4 The notations of BAN logic

The notations	Meaning
ρ, ϱ	The main participants in the model
X_m	Messages
K	A secret key
$\rho \equiv \varrho$	ρ believes ϱ
$\rho \triangleleft X_m$	ρ sees X_m
$\rho \sim X_m$	ρ sent X_m
$\#(X_m)$	The message X_m is fresh
$\rho \stackrel{K}{\leftrightarrow} \varrho$	ρ and ϱ communicate by K
$\rho \Rightarrow \varrho$	ρ is able to control ϱ
$(X_m)_K$	The message X_m is hashed by K

- **R₁**: Message meaning rule: $\frac{\rho | \equiv \rho \stackrel{K}{\leftrightarrow} \varrho, \rho \triangleleft (X_m)_K}{\rho | \equiv \varrho | \sim X_m}$
- **R₂**: Freshness rule: $\frac{\rho | \equiv \#(X_m)}{\rho | \equiv \#(X_m, Y_m)}$
- **R₃**: Nonce-verification rule: $\frac{\rho | \equiv \#(X_m), \rho | \equiv \varrho | \sim X_m}{\rho | \equiv \varrho | \equiv X_m}$
- **R₄**: Jurisdiction rule: $\frac{\rho | \equiv \varrho | \Rightarrow X_m, \rho | \equiv \varrho | \equiv X_m}{\rho | \equiv X_m}$

Goals: Our scheme fulfills the ultimate requirements of authentication for VANETs if it can achieve the following goals:

- **G₁**: $TA | \equiv OBU | \equiv (\mu_{auth})$
- **G₂**: $TA | \equiv RSU | \equiv (SIG)$
- **G₃**: $RSU | \equiv \left(RSU \stackrel{S_{R-V}}{\leftrightarrow} OBU \right)$
- **G₄**: $OBU | \equiv RSU | \equiv (L)$

The idealized form: The transformation of our proposed scheme is viewed in the following:

1. The protocol messages are:

- **PM₁**: $OBU \rightarrow RSU : \{T_1, ID_R, PID_i, I, CR_i, \mu_{auth}\}$.
- **PM₂**: $RSU \rightarrow TA : \{T_2, T_1, ID_R, PID_i, I, CR_i, \mu_{auth}, SIG\}$
- **PM₃**: $TA \rightarrow RSU : \{T_3, T_1, \beta, PID_i, CR_i, \lambda\}$
- **PM₄**: $RSU \rightarrow OBU : \{T_4, L, index\}$

2. Idealizing the protocol messages are:

- **IM₁**: $OBU \rightarrow TA : (\mu_{auth})_{h(RID_i || S_{R-V})}$
- **IM₂**: $RSU \rightarrow TA : (SIG)_{h(SK_{R-TA})}$
- **IM₃**: $TA \rightarrow RSU : \left(RSU \stackrel{S_{R-V}}{\leftrightarrow} OBU \right)_{h(SK_{R-TA})}$
- **IM₄**: $RSU \rightarrow OBU : (L)_{h(S_{R-V})}$

Assumptions The proof of our scheme relies on some assumptions as follow:

- $A_1: RSU | \equiv \#(T_1, T_3)$
- $A_2: TA | \equiv \#(T_2)$
- $A_3: OBU | \equiv \#(T_4)$
- $A_4: TA | \equiv OBU \stackrel{RID_i || S_{R-V}}{\leftrightarrow} TA$
- $A_5: RSU | \equiv RSU \stackrel{SK_{R-TA}}{\leftrightarrow} TA$
- $A_6: TA | \equiv RSU \stackrel{SK_{R-TA}}{\leftrightarrow} TA$
- $A_7: RSU | \equiv TA \Rightarrow RSU \stackrel{S_{R-V}}{\leftrightarrow} OBU$
- $A_8: OBU | \equiv RSU \stackrel{S_{R-V}}{\leftrightarrow} OBU$

Proof The proof is shown below:

Based on IM_1 , we obtain

$$S_1: TA \triangleleft (\mu_{auth})_{h(RID_i || S_{R-V})}$$

Based on S_1, A_4 , and by using R_1 , we can obtain

$$S_2: TA | \equiv OBU | \sim (\mu_{auth})$$

Based on S_2, A_2 , and by using R_2 and R_3 , we can obtain

$$S_3: TA | \equiv OBU | \equiv (\mu_{auth}) G_1$$

Based on IM_2 , we get

$$S_4: TA \triangleleft (SIG)_{h(SK_{R-TA})}$$

Based on S_4, A_6 , and by using R_1 , we can obtain

$$S_5: TA | \equiv RSU | \sim (SIG)$$

Based on S_5, A_2 , and by using R_2 and R_3 , we can obtain

$$S_6: TA | \equiv RSU | \equiv (SIG) G_2$$

Based on IM_3 , we get

$$S_7: RSU \triangleleft \left(RSU \stackrel{S_{R-V}}{\leftrightarrow} OBU \right)_{h(SK_{R-TA})}$$

Based on S_7, A_5 , and by using R_1 , we can obtain

$$S_8: RSU | \equiv TA | \sim \left(RSU \stackrel{S_{R-V}}{\leftrightarrow} OBU \right)$$

Based on S_8, A_1 , and by using R_2 and R_3 , we can obtain

Table 5 Different cryptographic symbol descriptions and execution time

Operations	Descriptions	The execution time (ms)
T_{sm-e}	Execution time for calculating the elliptic curve point multiplication	0.3476
T_{sm-e-s}	Execution time for small-scale scalar point multiplication operation based on elliptic curve	0.0246
T_{pa-e}	Execution time for calculating the elliptic curve point addition	0.002
T_h	Execution time for the general hash operation	0.0012
T_{aes-e}	Execution time for the encryption in the AES algorithm	0.183
T_{aes-d}	Execution time for the decryption in AES algorithm	0.157

$$S_9: RSU| \equiv TA| \equiv \left(RSU \overset{S_{R-V}}{\leftrightarrow} OBU \right)$$

Based on S_9 , A_7 , and by using R_4 , we can obtain

$$S_{10}: RSU| \equiv \left(RSU \overset{S_{R-V}}{\leftrightarrow} OBU \right) G_3$$

Based on IM_4 , we get

$$S_{11}: OBU \triangleleft (L)_{h(S_{R-V})}$$

Based on S_{11} , A_8 , and by using R_1 , we can obtain

$$S_{12}: OBU| \equiv RSU| \sim (L)$$

Based on S_{12} , A_3 , and by using R_2 and R_3 , we can obtain

$$S_{13}: OBU| \equiv RSU| \equiv (L) G_4$$

Consequently, it is clear that SD2PA scheme satisfies all the earlier said goals. Thus, our scheme is fully protected.

Performance analysis

Computational overhead

In this subsection, we analyse and compare the computational overhead for three of the recent interesting VANET schemes [21–23], as well as our proposed SD2PA scheme, in terms of computational complexity. The schemes proposed by Jie Cui et al. [21] and Jie Cui et al. [23] have been designed based on ECC, whereas the scheme proposed by Jie Cui et al. [22] has been designed based on the Advanced Encryption Standard algorithm (AES). Our proposed SD2PA scheme is designed based on the hash function. The ECC technology adopted in the schemes [21] and [23] is based on an 80-bit security level and built on the following: G is an additive group of order q , generated by a point P on a non-singular elliptic curve $\bar{E} : Y^2 = x^3 + ax + b \pmod p$, where $a, b \in Z_q^*$, p, q are 160-bit prime numbers.

To assure comparison accuracy, the crypto-operations metrics have to be under the same environments and conditions. The VANET computation evaluation method proposed in [22] is here adopted to guarantee an accurate comparison and results. For simplicity, let BG, SBV and NBV denote the execution times for beacon generation, single beacon verification and n beacon verification, respectively. The cryptographic operation notations, along with their execution time, are shown in Table 5. A detailed computational analysis for [21–23] is discussed below.

In Jie Cui et al's scheme [21], BG consists of two scalar multiplication operations and three hash function operations, so the overall cost of BG is $2T_{sm-e} + 3T_h = 0.6988$. SBV consists of three scalar multiplication operations, one point addition operation and two hash function operations, so the overall cost of SBV is $3T_{sm-e} + T_{pa-e} + 2T_h = 1.0472$. NBV consists of $(n+2)$ scalar multiplication operations, $(2n)$ small-scale scalar point multiplications operations, one point addition operation and $(2n)$ hash function operations, so the overall cost of NBV is $(2+n)T_{sm-e} + (2n)T_{sm-e-s} + T_{pa-e} + (2n)T_h = 0.6972 + 0.3983n$.

In Jie Cui et al's scheme [22], BG consists of one AES encryption operation and two hash function operations, so the overall cost of BG is $T_{aes-e} + 2T_h = 0.1854$. SBV consists of one AES decryption operation and four hash function operations, so the overall cost of SBV is $T_{aes-d} + 4T_h = 0.1618$. NBV consists of (n) decryption operations and four hash function operations, so the overall cost of NBV is $(n)T_{aes-d} + 4T_h = 0.0048 + 0.157n$.

In Jie Cui et al's scheme [23], BG consists of two scalar multiplication operations and two hash function operations, so the overall cost of BG is $2T_{sm-e} + 2T_h = 0.6976$. The verification process consists of two main stages:

- OBU–RSU authentication key verification, in case of SBV, this stage requires one scalar multiplication operation and one hash function operation. For NBV, it requires (n) scalar multiplication operation and (n) hash function operation.
- Message signature validation requires two scalar multiplication operations, one point addition operation and one hash function operation for SBV. NBV consists of $(n+2)$ scalar multiplication operations, $(2n)$ small-scale scalar point multiplications operations, $(n+1)$ point addition operations and (n) hash function operations.

Overall SBV is therefore $(T_{sm-e} + T_h) + (2T_{sm-e} + T_{pa-e} + T_h) = 1.0472$, while overall NBV is $((n)T_{sm-e} + (n)T_h) + ((n+2)T_{sm-e} + (2n)T_{sm-e-s} + (n+1)T_{pa-e} + (n)T_h) = 0.6972 + 0.7488n$.

In our scheme, BG consists of only three hash function operations for the authentication process and one for broadcasting, so the overall cost of BG is $4T_h = 0.0048$. The beacon verification in our scheme is done by the RSU, which has more resources than the OBUs. It broadcasts the verification results in a notification message that is encrypted by its private key and the OBUs only need to decrypt this notification message in a negligible time, using the RSU's public key. However, once the RSU receives the beacon, it needs to find the items $\langle S_{R-V}, PID_i, indx \rangle$ in $L_{RSU-OBU}$ that satisfy the signature. With the help of the $indx$, in the worst case, the RSU needs to identify 30 items in $L_{RSU-OBU}$ as long as each $indx$ refers to information for only 30 vehicles in $L_{RSU-OBU}$.

Table 6 Computation costs for the four schemes

Scheme	BG (ms)	SBV (ms)	NBV (ms)
[21]	0.6988	1.0472	$0.6972 + 0.3983n$
[22]	0.1854	0.1618	$0.0048 + 0.157n$
[23]	0.6976	1.0472	$0.6972 + 0.7488n$
SD2PA	0.0048	0.036	$0.036n$

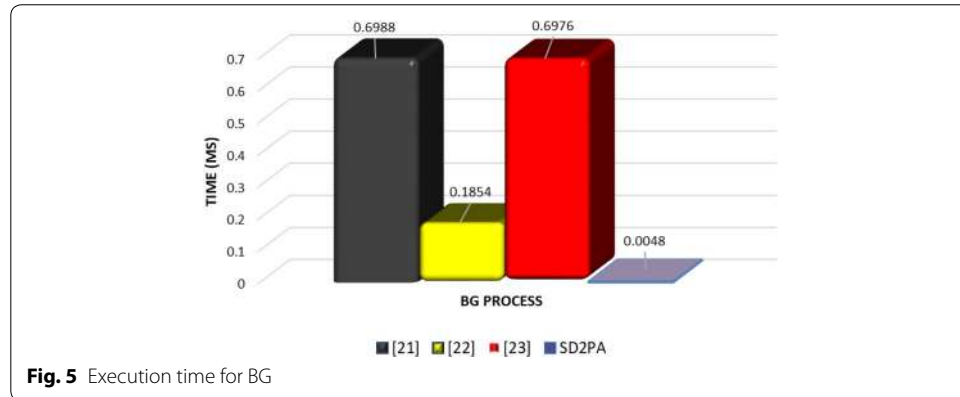


Fig. 5 Execution time for BG

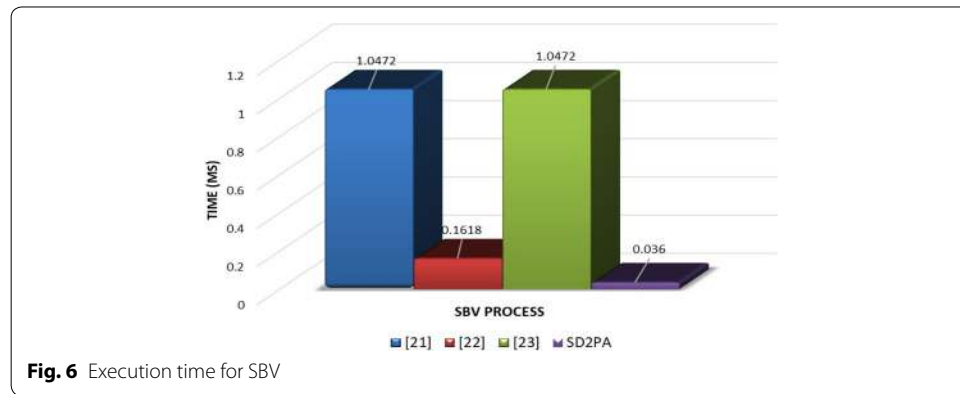


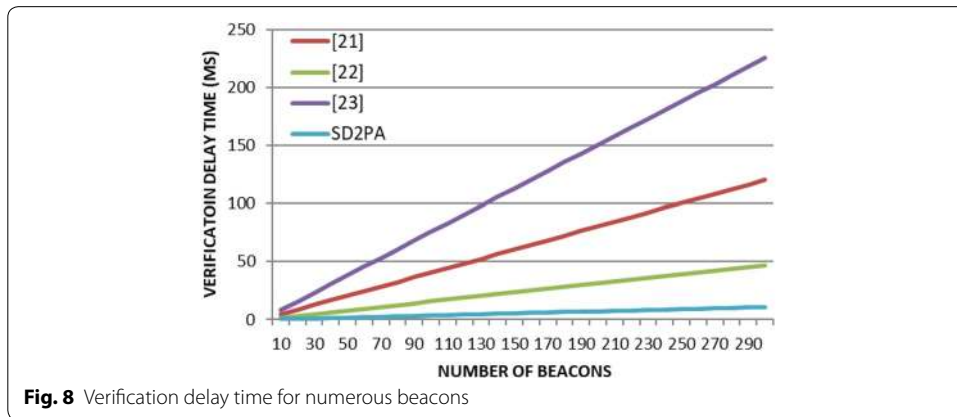
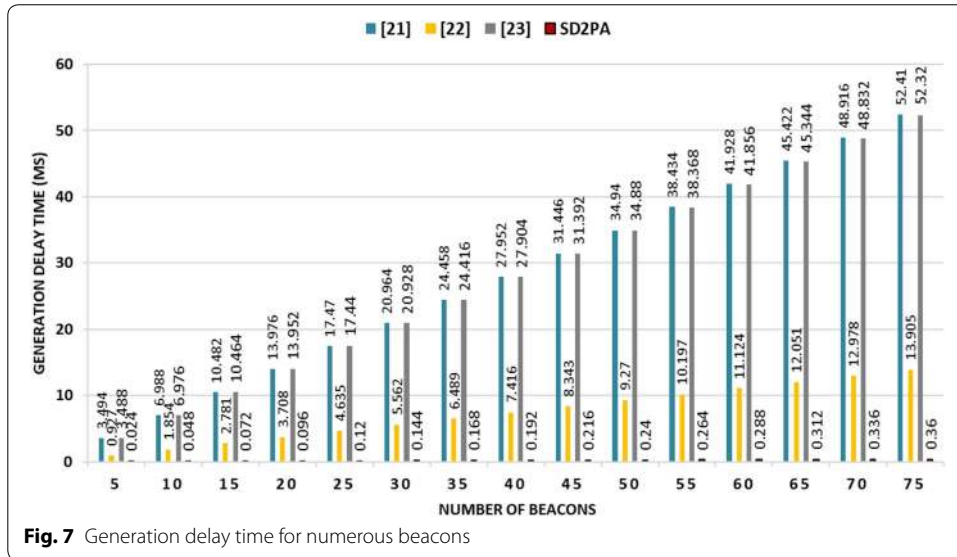
Fig. 6 Execution time for SBV

SBV, therefore, consists of thirty hash function operations, so the overall cost of SBV is $30T_h = 0.036$. NBV consists of $(30n)$ hash function operations, so the overall cost of NBV is $(30n)T_h = 0.036n$. Table 6, Figs. 5 and 6 show the comparative findings. It is obvious that the SD2PA scheme superior the other mentioned VANET solutions in terms of computational costs, and can highly enhances the system performance.

From Table 6, we can find that the SD2PA scheme enhances BG time by 99.3%, 97.4% and 99.3% compared with the Jie Cui et al. [21], Jie Cui et al. [22] and Jie Cui et al. [23] schemes, respectively. It also enhances SBV time by 96.6%, 77.8% and 96.6% compared with the Jie Cui et al. [21], Jie Cui et al. [22] and Jie Cui et al. [23] schemes, respectively, while NBV time is enhanced by 91.3%, 77.1% and 95.3% for 50 beacons compared with the Jie Cui et al. [21], Jie Cui et al. [22] and Jie Cui et al. [23] schemes, respectively. The enhancements of SD2PA over the other schemes are presented in Table 7.

Table 7 (SD2PA) scheme enhancement compared to others

Scheme	BG (%)	SBV (%)	NBV(50 beacons) (%)
[21]	99.3	96.6	91.3
[22]	97.4	77.8	77.1
[23]	99.3	96.6	95.3



The delay time for the generation and verification of numerous beacons is illustrated in Figs. 7 and 8. From these figures, it is clear that the SD2PA scheme is more rapid and more suitable than the other available schemes for VANET systems.

Table 8 Communication costs for the four schemes

Scheme	Beacon size (bytes)
[21]	84
[22]	152
[23]	84
SD2PA	48

Communication overhead

In this subsection, we compare the SD2PA scheme with the schemes of Jie Cui et al. [21], Jie Cui et al. [22] and Jie Cui et al. [23] in terms of communication costs. During the computation overhead subsection, we mentioned that the size of p is 20 bytes. Each element in G , therefore, requires 40 bytes. We also suppose that the size of the timestamp output, the hash function output and the elements in Z_q^* and int . are 4 bytes, 20 bytes, 20 bytes and 4 bytes, respectively. Table 8 shows the comparison of communication overhead regardless of the size of the traffic-related message, which is used in all the compared schemes and is the same size.

In the Jie Cui et al. [21] scheme, the beacon message is $\{PID_i, M_i, \sigma_i, T_i\}$ where $PID_i = \{PID_{i1}PID_{i2}\}$, $\{PID_{i1} \in G\}$, $\{PID_{i2}, \sigma_i \in Z_q^*\}$ and T_i is a timestamp. The beacon size is, therefore, $40 + 2 * 20 + 4 = 84$ bytes. The remaining calculations for [22, 23] were found in the same way. In our scheme, the beacon message is $\{\gamma, T_5, M_i, ID_R, indx\}$, where $\{\gamma, ID_R \in Z^*\}$, $indx \in int$. and T_5 is a timestamp. The beacon size is therefore $2 * 20 + 2 * 4 = 48$ bytes. Our scheme is, therefore, more efficient than the schemes of [21–23] in terms of communication costs.

Conclusion

In this paper, we have shown that recently proposed CPPA schemes by Jie Cui et al. [21–23], have failed to offer sufficient driving safety for vehicles in a critical driving area and that, moreover, they are vulnerable in terms of VANET’s computational, communicational and management efficiency. We, therefore, proposed an efficient, fully safe driving and lightweight CPPA VANET scheme based on a general hash function. For ideal vehicle authentication and message verification, we used the cuckoo filter database in cooperative RSUs such that, when a vehicle leaves one RSU and joins the next, the new RSU can authenticate the vehicle efficiently without burdening the TA or creating a bottleneck. The security and privacy analysis indicates that the proposed scheme satisfies the VANET requirements. Extensive and deep performance analysis directories show that the proposed scheme yields much better performance in terms of computational costs and communication overhead compared with the recently proposed schemes. The proposed scheme is therefore much more suitable for practical use in VANET conditions.

Acknowledgements

Not applicable.

Authors’ contributions

ASS proposed the main conception of the work and designed it. ASS and HA analyzed and discussed the results. AF and MI defined the overall organization of the manuscript. SL performed total supervision of this work. All authors read and approved the final manuscript.

Funding

This work is supported by the Science and Technology Program of Shenzhen, China under Grant Nos. JCYJ20180306124612893, JCYJ20170818160208570 and JCYJ20170307160458368.

Availability of data and materials

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

¹ School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan 430074, China. ² Department of Computer Techniques Engineering, Imam Al-Kadhun College (IKC), Baghdad 10001, Iraq. ³ Hubei Engineering Research Center on Big Data Security, School of Cyber Science & Engineering, Huazhong University of Science and Technology, Wuhan 430074, China. ⁴ Umm Alqura University, Makkah, Saudi Arabia. ⁵ Intelligent Manufacturing Systems (IMS) Centre, University of Windsor, 401 Sunset Ave, Windsor, ON N9B 3P4, Canada. ⁶ Nanjing Souwen Information Technology Co., Ltd., Nanjing 211800, China.

Received: 29 September 2019 Accepted: 11 August 2020

Published online: 02 September 2020

References

- Lee JK, Jeong YS, Park JH (2015) s-ITSF: a service based intelligent transportation system framework for smart accident management. *Hum Cent Comput Inf Sci* 5(1):34
- Papadimitratos P, Fortelle AL, Evenssen K, Brignolo R, Cosenza S (2009) Vehicular communication systems: enabling technologies, applications, and future outlook on intelligent transportation. *IEEE Commun Mag* 47(11):84–95
- Alfadhli SA, Alreshdeedi S, Lu S, Fatani A, Ince M (2019) ELCPH: An efficient lightweight conditional privacy-preserving authentication scheme based on hash function and local group secret key for VANET. In: Proceedings of the 2019 the world symposium on software engineering pp 32–36
- Bajaj K, Limbasiya T, Das D (2020) An efficient message transmission and verification scheme for VANETs. In: International conference on distributed computing and internet technology. Springer, Cham pp 127–143
- Pournaghi SM, Zahednejad B, Bayat M, Farjami Y (2018) NECPPA: a novel and efficient conditional privacy-preserving authentication scheme for VANET. *Comput Netw* 134:78–92
- Grover K, Lim A, Lee S (2015) Efficient authentication approach for highly dynamic vehicular ad hoc networks. *Int J Ad Hoc Ubiquit Comput* 19:193–207
- Guo J, Li X, Liu Z, Ma J, Yang C, Zhang J, Wu D (2020) TROVE: a context awareness trust model for VANETs using reinforcement learning. *IEEE Internet Things J*. <https://doi.org/10.1109/JIOT.2020.2975084>
- Bayat M, Barmshoory M, Rahimi M, Aref MR (2015) A secure authentication scheme for VANETs with batch verification. *Wireless Netw* 21(5):1733–1743
- Sheikh MS, Liang J, Wang W (2020) Security and privacy in vehicular ad hoc network and vehicle cloud computing: a survey. *Wirel Commun Mobile Comput*. <https://doi.org/10.1155/2020/5129620>
- Malhi AK, Batra S, Pannu HS (2019) Security of vehicular ad hoc networks: a comprehensive survey. *Comput Secur*. <https://doi.org/10.1016/j.cose.2019.101664>
- Manivannan D, Moni SS, Zeadally S (2020) Secure authentication and privacy-preserving techniques in Vehicular Ad hoc Networks (VANETs). *Veh Commun*. <https://doi.org/10.1016/j.vehcom.2020.100247>
- Zhang C, Lin X, Lu R, Ho PH, Shen X (2008) An efficient message authentication scheme for vehicular communications. *IEEE Trans Veh Technol* 57(6):3357–3368
- Lee CC, Lai YM (2013) Toward a secure batch verification with group testing for VANET. *Wirel Netw* 19(6):1441–1449
- Alamer A, Basudan S (2020) An efficient truthfulness privacy-preserving tendering framework for vehicular fog computing. *Eng Appl Artif Intell* 91:103583
- Cui J, Xu W, Han Y, Zhang J, Zhong H (2020) Secure mutual authentication with privacy preservation in vehicular ad hoc networks. *Veh Commun* 21:100200
- Kakkasageri MS, Manvi SS (2014) Information management in vehicular ad hoc networks: a review. *J Netw Comput Appl* 39:334–350
- Bitam S, Mellouk A, Zeadally S (2015) VANET-cloud: a generic cloud computing model for vehicular Ad Hoc networks. *IEEE Wirel Commun* 22(1):96–102
- Xia Z, Hu Z, Luo J (2017) UPTP vehicle trajectory prediction based on user preference under complexity environment. *Wirel Pers Commun* 97(3):4651–4665
- Yi K, Du R, Liu L, Chen Q, Gao K (2017) Fast participant recruitment algorithm for large-scale vehicle-based mobile crowd sensing. *Pervasive Mobile Comput* 38:188–199
- Granados JA, Batalla JM, Togay C (2020) Redundant localization system for automatic vehicles. *Mech Syst Signal Process* 136:106433
- Cui J, Wei L, Zhang J, Xu Y, Zhong H (2019) An efficient message-authentication scheme based on edge computing for vehicular ad hoc networks. *IEEE Trans Intell Transp Syst* 20(5):1621–1632
- Cui J, Tao X, Zhang J, Xu Y, Zhong H (2018) HCPA-GKA: a hash function-based conditional privacy-preserving authentication and group-key agreement scheme for VANETs. *Veh Commun* 14:15–25
- Cui J, Zhang J, Zhong H, Xu Y (2017) SPACF: a secure privacy-preserving authentication scheme for VANET with cuckoo filter. *IEEE Trans Veh Technol* 66(11):10283–10295
- Raya M, Hubaux JP (2007) Securing vehicular ad hoc networks. *J Comput Secur* 15(1):39–68

25. Lin X, Lu R, Zhang C, Zhu H, Ho PH, Shen X (2008) Security in vehicular ad hoc networks. *IEEE Commun Mag* 46(4):88–95
26. Lin X, Sun X, Ho PH, Shen X (2007) GSIS: a secure and privacy-preserving protocol for vehicular communications. *IEEE Trans Veh Technol* 56(6):3442–3456
27. Sun X, Lin X, Ho PH (2007) Secure vehicular communications based on group signature and ID-based signature scheme. In: 2007 IEEE international conference on communications pp 1539–1545
28. Guo J, Baugh JP, Wang S (2007) A group signature based secure and privacy-preserving vehicular communication framework. In: 2007 mobile networking for vehicular environments pp 103–108
29. Lu R, Lin X, Liang X, Shen X (2012) A dynamic privacy-preserving key management scheme for location-based services in vanets. *IEEE Trans Intell Transp Syst* 13(1):127–139
30. Rajput U, Abbas F, Oh H (2016) A hierarchical privacy preserving pseudonymous authentication protocol for VANET. *IEEE Access* 4:7770–7784
31. Zhang C, Ho PH, Tapolcai J (2011) On batch verification with group testing for vehicular communications. *Wirel Netw* 17(8):1851–1865
32. Chim TW, Yiu SM, Hui LC, Li VO (2011) SPECS: secure and privacy enhancing communications schemes for VANETS. *Ad Hoc Netw* 9(2):189–203
33. Horng SJ, Tzeng SF, Pan Y, Fan P, Wang X, Li T, Khan MK (2013) b-SPECS+ : batch verification for secure pseudonymous authentication in VANET. *IEEE Trans Inf Forensics Secur* 8(11):1860–1875
34. Shim KA (2012) CPAS: an efficient conditional privacy-preserving authentication scheme for vehicular sensor networks. *IEEE Trans Veh Technol* 61(4):1874–1883
35. Liu JK, Yuen TH, Au MH, Susilo W (2014) Improvements on an authentication scheme for vehicular sensor networks. *Expert Syst Appl* 41(5):2559–2564
36. Jianhong Z, Min X, Liying L (2014) On the security of a secure batch verification with group testing for VANET. *Int J Netw Secur* 16(5):355–362
37. Wang Y, Zhong H, Xu Y, Cui J, Guo F (2016) Efficient extensible conditional privacy-preserving authentication scheme supporting batch verification for VANETS. *Secur Commun Netw* 9(18):5460–5471
38. Zhang L, Hu C, Wu Q, Domingo-Ferrer J, Qin B (2016) Privacy-preserving vehicular communication authentication with hierarchical aggregation and fast response. *IEEE Trans Comput* 65(8):2562–2574
39. Tzeng SF, Horng SJ, Li T, Wang X, Huang PH, Khan MK (2017) Enhancing security and privacy for identity-based batch verification scheme in VANETS. *IEEE Trans Veh Technol* 66(4):3235–3248
40. Xie Y, Wu L, Zhang Y, Shen J (2016) Efficient and secure authentication scheme with conditional privacy-preserving for VANETS. *Chin J Electron* 25(5):950–956
41. Zhong H, Wen J, Cui J, Zhang S (2016) Efficient conditional privacy-preserving and authentication scheme for secure service provision in VANET. *Tsinghua Sci Technol* 21(6):620–629
42. Wu L, Fan J, Xie Y, Wang J, Liu Q (2017) Efficient location-based conditional privacy-preserving authentication scheme for vehicle ad hoc networks. *Int J Distrib Sens Netw* 13(3):1–13
43. Alazzawi MA, Lu H, Yassin AA, Chen K (2019) Efficient conditional anonymity with message integrity and authentication in a vehicular ad-hoc network. *IEEE Access* 7:71424–71435
44. Vijayakumar P, Azees M, Chang V, Deborah J, Balusamy B (2017) Computationally efficient privacy preserving authentication and key distribution techniques for vehicular ad hoc networks. *Clust Comput* 20(3):2439–2450
45. Zhang C, Lu R, Lin X, Ho PH, Shen X (2008) An efficient identity-based batch verification scheme for vehicular sensor networks. In: IEEE INFOCOM 2008—the 27th conference on computer communications pp 246–250
46. Kenney JB (2011) Dedicated short-range communications (DSRC) standards in the United States. *Proc IEEE* 99(7):1162–1182
47. Xie K, Ning X, Wang X, He S, Ning Z, Liu X, Wen J, Qin Z (2017) An efficient privacy-preserving compressive data gathering scheme in WSNs. *Inf Sci* 390:82–94
48. Rathore S, Park JH (2018) Semi-supervised learning based distributed attack detection framework for IoT. *Appl Soft Comput* 72:79–89
49. Singh SK, Rathore S, Park JH (2019) Blockiotintelligence: a blockchain-enabled intelligent IoT architecture with artificial intelligence. *Fut Gener Comput Syst*. <https://doi.org/10.1016/j.future.2019.09.002>
50. Fan B, Andersen DG, Kaminsky M, Mitzenmacher MD (2014) Cuckoo filter: Practically better than bloom. In: Proceedings of the 10th ACM international on conference on emerging networking experiments and technologies. pp 75–88
51. Burrows M, Abadi M, Needham RM (1990) A logic of authentication. *ACM Trans Comput Syst* 8(1):18–36

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.