SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks (2016)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/sec.1602

RESEARCH ARTICLE

An enhanced and secure trust-extended authentication mechanism for vehicular ad-hoc networks

Saru Kumari[1] *, Marimuthu Karuppiah[2], Xiong Li[3], Fan Wu[4], Ashok Kumar Das[5] and Vanga Odelu[6,7]

[1] Department of Mathematics, Chaudhary Charan Singh University, Meerut, 250 005 Uttar Pradesh, India
[2] School of Computing Science and Engineering, VIT University, Vellore, Tamilnadu 632 014, India
[3] School of Computer Science and Engineering, Hunan University of Science and Technology, Xiangtan 411201, China
[4] Department of Computer Science and Engineering, Xiamen Institute of Technology, Xiamen, 361021, China
[5] Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad, Andhra Pradesh 500 032, India
[6] Department of Mathematics, Indian Institute of Technology, Kharagpur 721 302, India
[7] Department of Computer Science and Engineering, Indian Institute of Information Technology, Sri City, Chittoor, Andhra Pradesh 361021, India

ABSTRACT

Vehicular Ad-hoc Networks (VANETs) are a move towards regulating safe traffic and intelligent transportation system. A VANETs is characterized by extremely dynamic topographical conditions owing to speedily moving vehicles. In VANETs, vehicles can transmit messages within a pre-defined area to achieve safety and efficiency of the system. Then ensuring authenticity of origin of messages to the receiver in such a dynamic environment is a crucial challenge. Another concern in VANET is preservation of privacy of user/vehicle. Recently, Chuang and Lee proposed a trust-extended authentication mechanism (TEAM) for vehicle-to-vehicle communications in VANETs. TEAM not only satisfies various security features but also enhances the performance of the authentication process using transitive trust relationship among vehicles. Nonetheless, our analysis shows that TEAM is vulnerable to insider attack, privacy breach, impersonation attacks and some other problems. In this paper, to eradicate the vulnerabilities found in Chuang-Lee's scheme, an enhanced trust-extended authentication scheme for VANET is proposed. We display the efficiency of our scheme through security analysis and comparison. Through simulation results using widely accepted NS-2 simulator, we show that our scheme authenticates vehicles faster than Chuang-Lee's scheme. Copyright © 2016 John Wiley & Sons, Ltd.

KEYWORDS
VANETs; security; authentication; impersonation attack; privacy preservation; NS-2 simulation

***Correspondence**
Saru Kumari, Department of Mathematics, Ch. Charan Singh University, Meerut 250 005, Uttar Pradesh, India.
E-mail: saryusiirohi@gmail.com

1. INTRODUCTION

In recent years, Vehicular Ad-hoc Networks (VANETs) have received significant attention from researchers, automobile industry personnel, and government [1]. A VANET [2] is a wireless network of moving vehicles communicating and sharing information among themselves resulting in improved traffic conditions in terms of safety, efficiency, and comfort. Moving vehicles in VANET behave like sensing nodes to discover and connect with each other within an approximate range of 300 m [3,4] with frequently changing communication relations. Proper deployment of VANETs can improve road safety, driving experiences, and traffic management in less time and low expenditure. A vehicle may take critical decisions during an emergency situation on the basis of the received information and transmit emergency messages to safeguard the vehicular network. For example, in case of a critical situation like road accident or a bomb threat in a certain area of a city, police headquarters will instantly broadcast alert messages to the vehicles of this area to save the life of the people and for timely evacuation of the area or for rescue operations. VANETs are also helpful in conserving clean-green environment and fuel reservation. Besides, drivers and passengers can also avail value-added applications that are non-safety applications [5–9] through VANETs.

Like any other wireless network [10–13], security of communications in VANETs is also an important aspect. There may arise many undesirable scenarios as an outcome of the propagation of false messages communicated by an adversary. It can lead to wrong traffic diversion causing traffic jams, faulty decision making by drivers leading to road accidents, wastage of time and fuel, vehicle-theft, and others. Another concern in VANETs is safety against privacy breach [14–16], otherwise an adversary can track the location history of vehicles and can misuse driver's private information for crimes like robbery, theft, and kidnapping. Managing proper functioning of VANETs is a challenge because of aforementioned factors and its characteristics like lack of fixed infrastructure, rapidly changing scenarios ranging from moderate rural traffic to heavy urban traffic. In addition to ensure confidentiality and privacy of the transmitted messages, authentication of messages in VANETs is also necessary to prevent attackers from injecting, altering, and replaying messages, as well as to prevent eavesdropping and network controlling by attackers.

A glance of VANET is shown in Figure 1. A VANETs basically consists of three network units, namely, vehicles (users), fixed roadside units (RSUs), and the authentication server (AS) [17]. A user can be a vehicle, its driver or its passengers. Each vehicle in VANETs is fitted with a wireless on-board unit (OBU) embedded with tamper-resistant device [18] that provides secure storage space and communication capability for the vehicle. Vehicles are moving components, but RSUs are stationary acting as gateways to access Internet and to assist vehicles in establishing connections with the outside networks within their radio coverage. The AS is responsible for registration of vehicles and computation of secret parameters required for the purpose of authentication. According to IEEE 802.11p, there are two types of communication environments, vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) or vehicle-to-roadside unit (V2R) communications.

1.1. Related work

To meet the challenges of VANETs, a considerable amount of work [19–38] has been performed, and most [19–30] of which are focused on the issue of privacy-protection. Raya and Hubaux [5] presented an authentication scheme for VANETs based on the concept of anonymous certificates to conceal the original identity of users. They provisioned the storage of a number of anonymous certificates in each vehicle so that the vehicle can use different public or private key pairs in each authentication process so as to prohibit traceability. But in pursuit of changing key every time, a vehicle needs to store a large number of key pairs. In a large network, key-distribution and key-management are a complex issue. Lu et al. [29] proposed an alternative method to avoid the pre-storage of a large number of anonymous certificates. They provisioned that each vehicle would request for the issue of a short-time anonymous certificate from the RSU where the vehicle is near. A vehicle performs this process frequently in order to change the anonymous certificate to avoid message-linkability. It results in frequent vehicle-RSU interaction, affecting the performance of the VANETs. In [30], Freudiger et al. used the method of mix-zones for anonymity of vehicles. Scheme in [30] pre-loads a large number of anonymous certificates in each vehicle. Zhang et al. [20] proposed a scheme for secure vehicle communications with low communication overhead. Their scheme employs a key agreement protocol via which a vehicle obtains a symmetric key from a RSU. Besides, the vehicle has to use different public keys to communicate with RSUs for the sake of privacy protection. As a result, a vehicle pre-loads a fixed number of anonymous certificates. Thus, schemes in [20,30] are completely dependent on RSUs and failure of RSU results into collapse of the schemes. Studer et al. [31] presented a key management scheme using public

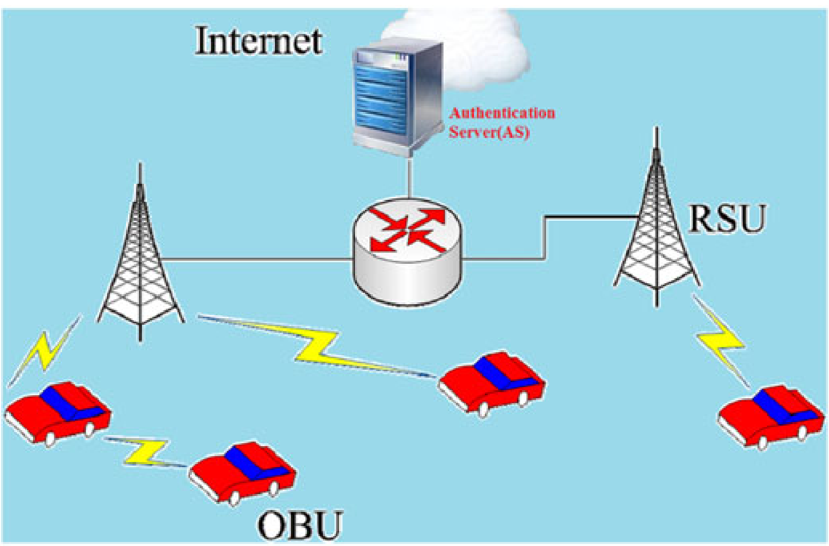

Figure 1. Architecture of VANET.

DOI: 10.1002/sec

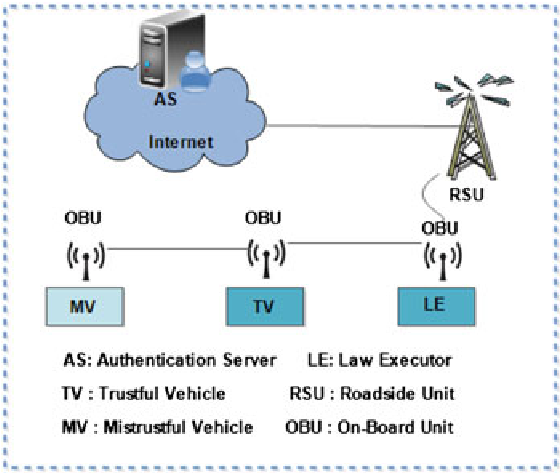

Figure 2. VANET architecture and vehicle category in TEAM (Source: [38]).

key infrastructure for VANETs to identify genuine vehicles. But, due to the use of public key infrastructure, the scheme suffers from problems like certification of public keys. Hsiao et al. [32] analyzed excessive collisions in the network because of message-flooding resulting in steep downfall of performance. Yeh et al. [33] presented a portable privacy-preserving authentication and access control protocol for non-safety applications in VANET. Horng et al. [34] showed that privacy-preserving authentication and access control protocol suffers from privilege elevation attack; that is, two or more vehicles can conspire to increase access privileges for preferred Internet services.

Recently, Chuang and Lee [38] proposed a trust-extended authentication mechanism (TEAM) for (V2V) communications in VANETs. In TEAM, vehicles are of three categorie: first are law executors (LE), which are authorized vehicles such as police car and public buses; second are trustful vehicles (*TV*s); and third are mistrustful vehicles (*MV*s), as shown in Figure 2. There are two states that any vehicle of VANETs is assumed to have, trustful state and mistrustful state. A normal vehicle remains in mistrustful state before passing the authentication process. As soon as a vehicle successfully authenticates itself in VANET, it attains the trustful state. An LE plays the role of a mobile *AS* and is always in trustful state. However, state of trust and mistrust is changeable in normal vehicles. Initially, only LEs are trustful, and all other vehicles are mistrustful. Thus, in starting, vehicles get authenticated and become trustful only with the help of LEs. After some time, some normal vehicles are also in trustful state along with *LE*; these vehicles are then called TVs. From then onwards, *LE* and these *TV*s help the mistrustful vehicles (*MV*s) to become trustful. In other words, TVs behave temporarily like LEs. In this way, trust relationship propagates from *LE*s to *TV*s and in turn to MVs, called as transitive-trust-relationship, in TEAM, as demonstrated in Figure 3. The need of such a provision is that in V2V communication networks, LEs are finite in number, and an LE cannot always move in the vicinity of an authentication-seeker *MV*. In the absence of transitive-trust-relationship, even with a legitimate user, the vehicle has to wait for the nearby LE to undergo authentication process. After successful authentication, a vehicle obtains its specific secret parameter using the secret key it acquires. A vehicle remains trustful as long as the lifetime of the acquired secret key is below the threshold limit beyond which the lifetime of the key is over, and the vehicle again reaches the mistrustful state. In order to continue the trustful state, a vehicle has to undergo a process, namely, key update process when the lifetime of the secret key is about to finish, to obtain a new secret key. It is noticeable that, a normal vehicle as a *TV* can assist the other *MV*s only in authentication process; for key update process, a vehicle has to interact with *LE*s.

In this study, we analyze TEAM proposed by Chuang and Lee for its merits and demerits. TEAM is a decentralized scheme because the authentication process of vehicles is not performed by any centralized authority. The scheme uses only lightweight operations such as hash operation and XOR operation. TEAM fulfills many security attributes, such as free from clock synchronization problem, fast error detection, resistance to replay, modification, key lifetime self-extension, and stolen-verifier attacks, and establishes session key. The storage space requirement of their scheme is minimal as vehicles do not need to store any authentication information of the other vehicles like public key. As our results, we show that TEAM has some weak points. The insider working at the AS has direct access to the user's password, and so their scheme is an easy target of password-misuse. Although, original identity of the user is not transmitted in any message over open network but an adversary can gain a user's identity through guessing attack. Further, guessing attack leads to LE/TV impersonation attacks, gives a way to make a niche in secure communication process, and permits an adversary to acquire a new secret key from LE. Secure communication process can be initiated or responded by an adversary. Moreover, an adversary can compute the session key which has to be established between two participants and hence can read the confidential communication exchanged between them. With a view to overcome the demerits of TEAM, we propose a secure trust-extended authentication scheme for VANET by introducing least possible changes in Chuang-Lee's scheme. In addition, we present the analysis of security features and computational cost of proposed scheme and then we use widely accepted NS-2 simulator to evaluate the performance of the proposed scheme.

1.2. Threat model

We use the Dolev–Yao threat model [39], in which two communicating parties communicate over a public channel [40]. The similar threat model is applicable in this paper, where the channel is public, and the end-points are not in general trustworthy. An adversary (either external or privileged-insider user of the server) can eavesdrop the messages and perform different attacks.

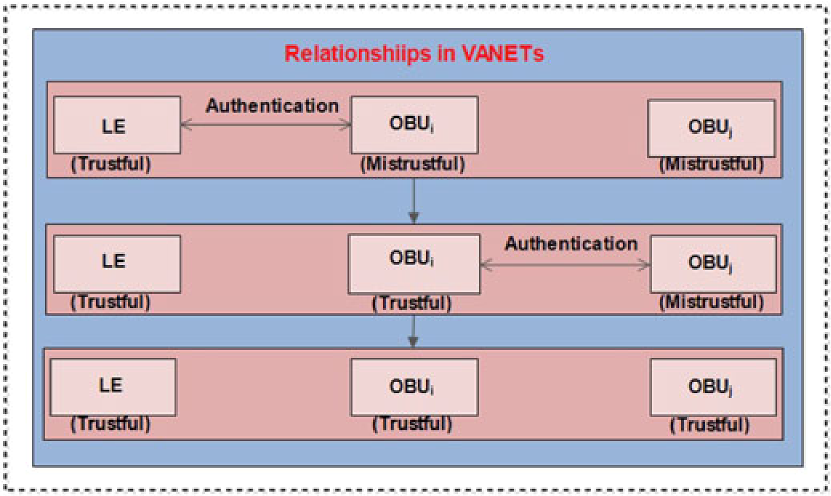

Figure 3. Transitive-trust relationship in TEAM (Source: [38]).

1.3. Organization of the paper

Section 2 reviews Chuang-Lee's scheme, and its cryptanalysis is discussed in Section 3. In Section 4, we propose our secure trust-extended authentication scheme for VANETs. Security of the proposed scheme is analyzed in Section 5. A comparative performance analysis of the proposed scheme is discussed in Section 6. Finally, we give our concluding remarks in Section 7.

2. REVIEW OF CHUANG-LEE'S SCHEME

Here we give description of Chuang and Lee's trust-extended authentication mechanism (TEAM) [38]. TEAM involves eight phases: registration, login, general authentication, trust-extended authentication, password change, secure communication, key revocation and key update. Prior to join the VANET, OBU of a vehicle undergoes registration with the AS. The login phase is initiated by a vehicle to access service from VANET. The OBU checks the authentication state, if the lifetime of the key is reduced to zero; the vehicle reaches the mistrustful state. Then MV undergoes either general or trust-extended authentication process to attain the trustful state. The TVs help other MVs to reach the trustful state by completing the authentication process. Any two TVs can indulge in secure communication to access the Internet. The TVs undergo the key update process with the LE before the key lifetime reaches the predefined threshold. Password change phase helps the user to change its password whenever needed. The state of the LE never changes as the LE is ever trustful. The OBU of each vehicle is assumed to be equipped with security hardware (such as trusted platform module), consisting of a tamper-resistant device (TPD), and an event data recorder (EDR) [41–43]. Because of tamper resistant property of OBU, an attacker cannot gain information stored in it. The EDR records data such as the time, location, preload secret key, and login history of the vehicle. The time of every vehicle is assumed to be synchronous via GPS device. The vehicles in a VANET broadcast the hello message periodically with the authentication state (trust state or mistrust state). Table I gives the list of notations with their corresponding description, which are used in the paper.

Table I. The notations and their meanings.

Notation	Description
AS	Authentication server
LE	Law executor
MV	Mistrustful vehicle
TV	Trustful vehicle
RSU	Roadside unit
OBU	On-board unit
$User_i$	ith user
E	An adversary
id_i	Identity of ith entity
pw_i	Password of ith entity
sk_{ij}	Session key between ith and jth entities
Msg_{KU}	Key update message
r_i	A random number/nonce
Psk	A secret key pre-shared between LEs and AS
T_i	Current timestamp of ith entity
$\oplus$	Bitwise exclusive-OR operator
$h(\cdot)$	A cryptographic one-way hash function
$\|\|$	String concatenation operator

LEs, law executors; AS, authentication server.

2.1. Registration phase

2.1.1. Law executor registration.

LE registers itself with the *AS* through the manufacturer or a secure channel. In this phase, *AS* computes a secret

Psk_n	Psk_{n-1}..............	Psk_2	Psk_1
$h(nonce) \longrightarrow$	$h^2(nonce) \longrightarrow$	$h^{n-1}(nonce) \longrightarrow$	$h^n(nonce)$

Figure 4. Generation of secret key-set using the hash-chain method.

key-set $\{Psk_i, i = 1, \ldots, n\}$ using the hash-chain method ($h^2(x) = h(h(x))$) as given in Figure 4.

AS provides this key-set to the *LE*. *LE* keeps this key-set stored in its security hardware. For security purpose, the lifetime of each Psk_i is short. Because of the one-way property of hash function, the new *Psk*(say Psk_2) cannot be derived from the old *Psk*(say Psk_1).

2.1.2. Vehicle registration.

Vehicles other than *LE*s undergo registration process with *AS* through the manufacturer or in a secure manner when the vehicle leaves the car factory. The process is described in the following steps:

(1) $User_i$ chooses its identity id_i and password pw_i, submits $\{id_i, pw_i\}$ to the *AS* via the manufacturer or a secure channel.
(2) On receiving $\{id_i, pw_i\}$, the *AS* computes $a_i = h(id_i \| x)$, $b_i = h^2(id_i \| x) = h(a_i)$, $c_i = h(pw_i) \oplus b_i$, and $D_i = Psk \oplus a_i$.
(3) The *AS* stores $\{id_i, b_i, c_i, D_i, h(\cdot)\}$ in the OBU's security hardware via the manufacturer or a secure channel.

2.2. Login phase

When $User_i$ wishes to access the service from VANET, he/she initiates the login process as the steps follows:

(1) $User_i$ inputs id_i and pw_i to the OBU_i.
(2) The OBU_i checks id_i and verifies if $h(pw_i) \oplus c_i$ and b_i are equal. The equality, guarantees the correctness of the inputted id_i and pw_i. Otherwise, the login request is rejected.

2.3. General authentication phase

As soon as the login process is complete, OBU_i carry out the general authentication process with some law executor vehicle, say LE_j as the steps follows:

(1) The OBU_i generates a random number r_1 to compute $m_1 = h(b_i) \oplus r_1$. It also computes $aid_i = h(r_1) \oplus id_i$, $m_2 = h(r_1 \| aid_i \| D_i)$. OBU_i transmits the authentication request $\{aid_i, m_1, m_2, D_i\}$ to LE_j.
(2) On receiving $\{aid_i, m_1, m_2, D_i\}$, the LE_j uses *Psk* to retrieve $a_i = D_i \oplus Psk$, $r_1 = m_1 \oplus h^2(a_i)$. Checks if $h(r_1 \| aid_i \| D_i)$ and m_2 are equal. The equality confirms the legality of OBU_i. Otherwise, the authentication request is rejected, thereby believing the breach of integrity of the request. LE_j computes $id_i = aid_i \oplus h(r_1)$ and generates a random number r_2 to compute $aid_j = id_j \oplus r_2$ and a session key $sk_{ij} = h(r_1 \| r_2)$. Further, LE_j computes $m_3 = r_2 \oplus h^2(r_1)$, $m_4 = a_i \oplus h(id_i)$, and $m_5 = h(m_4 \| r_2 \| aid_j)$. LE_j transmits the authentication response message $\{aid_j, m_3, m_4, m_5\}$ to OBU_i.
(3) OBU_i retrieves $r_2 = m_3 \oplus h^2(r_1)$ and checks if $h(m_4 \| r_2 \| aid_j)$ and m_5 are equal. The equality confirms the trustfulness of LE_j. Otherwise OBU_i terminates the process. Further, OBU_i retrieves $a_i = m_4 \oplus h(id_i)$, computes the session key $sk_{ij} = h(r_1 \| r_2)$, and the value $sk_{ij} \oplus h(r_2)$. OBU_i stores a_i in the security hardware and sends $sk_{ij} \oplus h(r_2)$ to LE_j.
(4) LE_j retrieves $h(r_2)$ from $sk_{ij} \oplus h(r_2)$ using sk_{ij} and checks it to detect an invalid *OBU* mounting a replay attack.

At the end of this process, OBU_i becomes trustful as it obtains the parameter *Psk* by computing $Psk = a_i \oplus D_i$. From then on, OBU_i can help other mistrustful *OBU*s to get authenticated without necessarily requiring an *LE*.

2.4. Trust-extended authentication phase

Trust-extended authentication process is based on the notion of transitive trust relationships which facilitates more and more *OBU*s to become trustful in VANET. As soon as a mistrustful *OBU* is authenticated successfully, it becomes trustful and obtains the authorized parameter *Psk*. Afterwards, this trustful *OBU* acts as a temporary *LE* and helps the other mistrustful *OBU*s in authentication. The procedure of the general authentication and the trust-extended authentication are the same. In this way, all vehicles in the VANET rapidly authenticate and attain the trustful state.

2.5. Password change phase

A user initiates this phase if he/she wishes to change his/her password. This phase is free from any involvement of *AS*. The steps required for this phase are as follows:

(1) $User_i$ inputs id_i and pw_i to the OBU_i.
(2) The OBU_i checks id_i and verifies if $h(pw_i) \oplus c_i$ and b_i are equal. The equality confirms the correctness of the inputted id_i and pw_i, and $User_i$ is asked to input the new password pw_{inew}. The OBU_i computes $c_{inew} = c_i \oplus h(pw_i) \oplus h(pw_{inew}) = b_i \oplus h(pw_{inew})$ and replaces c_i with c_{inew}.

2.6. Secure communication phase

When two trustful vehicles wish to communicate with each other, they can indulge in secure communication phase. Once the login phase is computed successfully by OBU_i, it can establish secure communication with OBU_j as the steps follows:

(1) OBU_i generates a random number r_3, computes $aid_i = r_3 \oplus id_i$, $m_1 = Psk \oplus r_3$, and $m_2 = Psk \oplus h(aid_i \| r_3)$, where OBU_i possesses Psk from the general/trust-extended authentication process. OBU_i sends $\{aid_i, m_1, m_2\}$ to the OBU_j.
(2) On receiving $\{aid_i, m_1, m_2\}$, OBU_j uses Psk to retrieve $r_3 = m_1 \oplus Psk$ and also retrieves $h(aid_i \| r_3)$ from m_2. OBU_j itself computes the value $h(aid_i \| r_3)$ and compares it with the value retrieved from m_2. The equality of these values confirms the trustfulness of OBU_i. OBU_j generates a random number r_4 to compute $aid_j = r_4 \oplus id_j$ and computes the session key $sk_{ij} = h(r_3 \| r_4 \| Psk)$ for secure communication. OBU_j also computes $m_3 = Psk \oplus r_4$ and $m_4 = Psk \oplus h(aid_j \| r_4 \| h(r_3))$. OBU_j sends the message $\{aid_j, m_3, m_4\}$ to the OBU_i.
(3) OBU_i retrieves $r_4 = m_3 \oplus Psk$ and also retrieves $h(aid_j \| r_4 \| h(r_3))$ from m_4. OBU_i itself computes the value $(aid_j \| r_4 \| h(r_3))$ and compares it with the value retrieved from m_4. The equality of these values confirms the trustfulness of OBU_j. Then OBU_i computes the session key $sk_{ij} = h(r_3 \| r_4 \| Psk)$ for the secure communication and also computes $sk_{ij} \oplus h(r_4)$. OBU_i sends $sk_{ij} \oplus h(r_4)$ to OBU_j.
(4) OBU_j retrieves $h(r_4)$ from $sk_{ij} \oplus h(r_4)$ using sk_{ij} and checks it to detect an invalid OBU mounting a replay attack. From then on, these two trustful vehicles can communicate securely using the established session key.

2.7. Key revocation phase

In Chuang–Lee's scheme, key revocation is based on timer that is regarded as the lifetime of the key. The authentication state of a mistrust vehicle changes to trustful when it obtains the key Psk after successful completion of the authentication process. At this stage the secure hardware starts counting down in a timer. When the lifetime of the key is finished, the state of the vehicle becomes mistrustful. Actually, the system can ask the trustful vehicle to undergo the key update phase.

2.8. Key update phase

Every trustful vehicle undergoes the key update process with LE when the key lifetime is about to over. After the completion of this phase, the trust state of TV gets extended. The process is as in the following steps:

(1) OBU_i generates a random number r_5 to compute $m_1 = Psk_{old} \oplus r_5$, $m_2 = Psk_{old} \oplus Msg_{KU}$, and $m_3 = h(r_5 \| Msg_{KU})$. OBU_i sends $\{m_1, m_2, m_3\}$ as a key update request to LE_j.
(2) LE_j retrieves $r_5 = m_1 \oplus Psk_{old}$ and $Msg_{KU} = m_2 \oplus Psk_{old}$. LE_j itself computes the value $h(r_5 \| Msg_{KU})$ and compares it with the obtained value m_3. The equality of these two values confirms the trustfulness of OBU_i. Then, LE_j generates a random number r_6 to compute $m_4 = r_6 \oplus h(r_5)$, $m_5 = Psk_{new} \oplus r_6$, and also computes $m_6 = h(r_6 \| Psk_{new})$. Further, LE_j computes the session key $sk_{ij} = h(r_5 \| r_6 \| Psk_{new})$. LE_j sends the reply message $\{m_4, m_5, m_6\}$ to the OBU_i.
(3) On receiving $\{m_4, m_5, m_6\}$, OBU_i retrieves $r_6 = m_4 \oplus h(r_5)$, and acquires $Psk_{new} = m_5 \oplus r_6$. OBU_i itself computes the value $h(r_6 \| Psk_{new})$ and compares it with the obtained value m_6. The equality of these two values confirms the trustfulness of LE_j. OBU_i updates the Psk and computes the session key $sk_{ij} = h(r_5 \| r_6 \| Psk_{new})$ for the secure communication and also computes $sk_{ij} \oplus h(r_6)$. OBU_i sends $sk_{ij} \oplus h(r_6)$ to LE_j.
(4) LE_j retrieves $h(r_6)$ from $sk_{ij} \oplus h(r_6)$ using sk_{ij} and checks it to detect an invalid OBU_i mounting a replay attack.

3. CRYPTANALYSIS OF CHUANG–LEE'S SCHEME

Cryptanalysis of Chuang–Lee's scheme is based on the fact that messages transmitted over open network can be intercepted. Messages exchanged during general/trust-extended authentication phase, secure communication phase, and key update phase can be intercepted by an adversary. We also point out an attack during the registration phase.

3.1. Insider attack

During registration phase, user sends its plaintext password pw_i to AS. It is very risky because the insider working at AS simply gets access to user's password. The insider can misuse pw_i. Thus, users in the scheme are victims of the insider attack.

3.2. Lacks user anonymity

Suppose an adversary E intercepts the messages $\{aid_i, m_1, m_2, D_i\}$ and $\{aid_j, m_3, m_4, m_5\}$, pertaining to a general authentication process, from the network. E guesses a value id_i^* as identity of $User_i$ of OBU_i, computes $(h(r_1))^* = aid_i \oplus id_i^*$, $(r_2)^* = m_3 \oplus h\left((h(r_1))^*\right)$, and $h\left(m_4 \| (r_2)^* \| aid_j\right)$. E checks if the computed value $h\left(m_4 \| (r_2)^* \| aid_j\right)$ and the received value m_5 are equal. The equality confirms that the guess id_i^* is correct, and in this way, E obtains the identity id_i of $User_i$. In case the equality does not hold, E repeats the process with another guess and keep on doing this till achieves success. In case of success, E also acquires the correct random number r_2. Similarly, E can gain the identity of a user from trust-extended authentication. Thus, an adversary E can reveal the identity of a user, and the scheme fails to provide user anonymity.

3.3. Session key breach

Suppose E intercepts the messages $\{aid_i, m_1, m_2, D_i\}$ and $\{aid_j, m_3, m_4, m_5\}$, exchanged during general authentication process, from the network. E can obtain the identity id_i of $User_i$ of OBU_i and LE_j's random number r_2 as discussed in Section 3.2. E can compute the secret value $a_i = m_4 \oplus h(id_i)$. Then E computes $b_i = h(a_i)$ and $h(b_i)$ to obtain $r_1 = m_1 \oplus h(b_i)$. Having random numbers r_1 and r_2, E can compute the session key $sk_{ij} = h(r_1 \| r_2)$. Therefore, E can read the confidential messages encrypted with this session key. Similarly, E can target the trust-extended authentication to breach the session key being agreed between the participants. Thus, session key is under breach in the scheme.

3.4. Impersonation attack

Here, we show that an adversary E can act as *LE/TV* to deceive the unauthenticated *OBU*'s as follows:

(1) First of all, E watches a general/trust-extended authentication process and intercepts the messages $\{aid_i, m_1, m_2, D_i\}$ and $\{aid_j, m_3, m_4, m_5\}$ from the network.
(2) E acquires the identity id_i of $User_i$ of OBU_i and LE_j's random number r_2 as explained in Section 3.2.
(3) E computes the secret value $a_i = m_4 \oplus h(id_i)$ pertaining to OBU_i and derives the secret key $Psk = D_i \oplus a_i$.
(4) E intercepts and blocks the authentication message $\{aid_k, m_{1k}, m_{2k}, D_k\}$ sent by OBU_k to an LE.
(5) E computes $a_k = D_k \oplus Psk$, $r_{1k} = m_{1k} \oplus h^2(a_k)$ and checks if $h(r_{1k} \| aid_k \| D_k)$ and m_{2k} are equal. The equality confirms the legality of OBU_k. Then E computes $id_k = aid_k \oplus h(r_{1k})$ and generates a random number r_{2E} to compute $aid_E = id_E \oplus r_{2E}$ and a session key $sk_{kE} = h(r_{1k} \| r_{2E})$, where id_E is an identity chosen by E. Next, E computes $m_{3E} = r_{2E} \oplus h^2(r_{1k})$, $m_{4Ew} = a_{Ew} \oplus h(id_k)$, and $m_{5E} = h(m_{4E} \| r_{2E} \| aid_E)$, where a_{Ew} is an arbitrary value chosen by E and noticeably $a_{Ew} \neq a_k$. E transmits message $\{aid_E, m_{3E}, m_{4Ew}, m_{5E}\}$ to OBU_k.
(6) OBU_k retrieves $r_{2E} = m_{3E} \oplus h^2(r_{1k})$ and checks if $h(m_{4wE} \| r_{2E} \| aid_E)$ and m_{5E} are equal. Clearly, the equality will hold because m_{4wE} and aid_E are received from the response message $\{aid_E, m_{3E}, m_{4Ew}, m_{5E}\}$, and r_{2E} is the correct random number that is chosen by E in Step 3.4. Hence, OBU_k believes that it is connected with a valid *LE*/trustful *OBU*. Thus, OBU_k retrieves $a_{Ew} = m_{4Ew} \oplus h(id_k)$, where $a_{Ew} \neq a_k$. Further, OBU_k computes the session key $sk_{kE} = h(r_{1k} \| r_{2E})$ and the value $sk_{kE} \oplus h(r_{2E})$; noticeably, the session key computed by OBU_k is exactly the same to the value of session key computed by E in Step 3.4. OBU_k stores a_{Ew} in its security hardware and sends $sk_{kE} \oplus h(r_2)$ in reply.
(7) E retrieves $h(r_{2E})$ from $sk_{kE} \oplus h(r_2)$ using sk_{kE} to check the equality of the session key computed by it and OBU_k.

After completing the aforementioned process, OBU_k believes itself to be trustful and computes $D_k \oplus a_{Ew}$ to obtain the secret parameter Psk. On the contrary, $D_k \oplus a_{Ew} = (Psk \oplus a_i) \oplus a_{Ew} \neq Psk$; therefore, OBU_k still remains mistrustful and cannot help the other *OBU*s to reach the trustful state. E can perform this process with other *OBU*s, thereby, rendering them mistrustful. Consequently, the general/trust-extended authentication process in the VANET gets hindered. Therefore, E can create such nuisance till the lifetime of Psk is over. On the other hand, E establishes the session key sk_{kE} with OBU_k for secure communication.

3.5. Inefficient secure communication between TVs

E can intercept the messages $\{aid_i, m_1, m_2\}$ and $\{aid_j, m_3, m_4\}$, exchanged during a secure communication process, from the network. E guesses a value id_i^* as identity of $User_i$ of OBU_i, computes $(r_3)^* = aid_i \oplus id_i^*$, $Psk^* = m_1 \oplus (r_3)^*$, and $l^* = m_2 \oplus Psk^*$. Then E computes $h(aid_i \| (r_3)^*)$ and checks if it is equal to l^*. For $l^* = h(aid_i \| (r_3)^*)$, E owns the correct identity id_i and correct random number r_3 pertaining to OBU_i and the correct parameter Psk. In case the equivalence does not holds, E repeats this process with some other guess and keeps on doing so till the success is achieved. Having Psk in hand, E retrieves $r_4 = m_3 \oplus Psk$ and computes the session key $sk_{ij} = h(r_3 \| r_4 \| Psk)$ to communicate with OBU_j and deceive it. Knowing the value of Psk, E can attack the scheme to hinder the smooth procedures in VANET in the following ways:

(1) E can act as *LE*/*OBU* to deceive the *OBU*s in seeking general/trust-extended authentication.
(2) E can initiate or reciprocate the secure communication process like a trustful *OBU*.

Thus, the secure communication phase of Chuang–Lee's scheme is inefficient as it invites an adversary E to enter the VANET and deteriorate the normal functioning.

3.6. Disclosure of new key

Once an adversary E obtains the correct existent secret parameter Psk(say Psk_{old}), as demonstrated in Sections 3.4 and 3.5, it can obtain the new key Psk_{new} as the steps follows:

(1) E generates a random number r_{5E} to compute $m_{1E} = Psk_{old} \oplus r_{5E}$, $m_{2E} = Psk_{old} \oplus Msg_{KU}$, and $m_{3E} = h(r_{5E} \| Msg_{KUE})$, where Msg_{KUE} is the key update message written by E. E initiates the key update phase by sending $\{m_{1E}, m_{2E}, m_{3E}\}$ as a key update request to LE_j.
(2) LE_j retrieves $r_{5E} = m_{1E} \oplus Psk_{old}$ and $Msg_{KUE} = m_{2E} \oplus Psk_{old}$. LE_j itself computes the value $h(r_{5E} \| Msg_{KUE})$ and compares it with the obtained value m_{3E}. Clearly, these two values will be equal by virtue of similar values of r_{5E} and Msg_{KUE} used by E and LE_j, respectively. Thus, LE_j believes that the key update request is sent by some trustfulness OBU. Then, LE_j generates a random number r_6 to compute $m_4 = r_6 \oplus h(r_{5E})$, $m_5 = Psk_{new} \oplus r_6$, and also computes $m_6 = h(r_6 \| Psk_{new})$. Then, LE_j computes the session key $sk_{Ej} = h(r_{5E} \| r_6 \| Psk_{new})$. LE_j sends message $\{m_4, m_5, m_6\}$ in reply.
(3) On receiving $\{m_4, m_5, m_6\}$, E retrieves $r_6 = m_4 \oplus h(r_{5E})$, and acquires $Psk_{new} = m_5 \oplus r_6$. E also computes the value $h(r_6 \| Psk_{new})$ and compares it with the obtained value m_6 to check the legality of LE_j. Equality of these two values authenticates LE_j. Next, E computes the session key $sk_{Ej} = h(r_{5E} \| r_6 \| Psk_{new})$ to establish a confidential communication channel with LE_j and sends $sk_{Ej} \oplus h(r_6)$ to LE_j so as to complete the key update process.
(4) LE_j retrieves $h(r_6)$ from $sk_{Ej} \oplus h(r_6)$ using sk_{Ej} and checks it. Obviously, E passes this test and successfully completes the key update process.

Adversary's capability to acquire the new key from an LE makes him/her the controller of the VANET as he/she can continuously deceive the OBUs in seeking authentication even after their key update.

3.7. User traceability attack

Whenever $User_i$ wishes to access services from the VANET, he/she sends the authentication request $\{m_1, m_2, D_i\}$, where the value $D_i = Psk \oplus a_i$ is the same value every time. Thus, an adversary E can trace the $User_i$ by intercepting the messages from open network. Hence, the scheme does not provide location privacy to users.

3.8. Lacks mutual authentication

An adversary E can act as LE/TV to deceive the unauthenticated OBU's as during general/trust-extended authentication as discussed in Section 3.4. Moreover, E can initiate or reciprocate the secure communication process like a trustful OBU as discussed in Section 3.5. Therefore, the scheme fails to provide mutual authentication.

4. THE PROPOSED ENHANCED SCHEME

In this section, to eradicate the vulnerabilities found in Chuang–Lee's scheme, an enhanced trust-extended authentication scheme for VANET is proposed. Our proposed scheme consists of eight phases: registration, login, general authentication, trust-extended authentication, password change, secure communication, key revocation, and key update.

4.1. Registration phase

4.1.1. LE registration.

This phase is same as in Chuang–Lee's scheme

4.1.2. Vehicle registration.

All vehicles except LEs have to register with AS through the manufacturer/a secure channel before leaving the car factory. Here are the required steps:

(1) $User_i$ of OBU_i chooses its identity id_i, password pw_i, and a random number u_i. Computes $h(pw_i) \oplus u_i$ submits $\{id_i, h(pw_i) \oplus u_i\}$ to the AS via the manufacturer or a secure channel.
(2) On receiving $\{id_i, h(pw_i) \oplus u_i\}$, the AS computes $a_i = h(id_i \| x)$, $b_i = h^2(id_i \| x) = h(a_i)$, $c_i = h(pw_i) \oplus u_i \oplus b_i$, $k_i = Psk \oplus a_i$ and $D_i = h(Psk \| T_r) \oplus id_i$, where T_r is the current timestamp acquired by AS. AS stores $\{c_i, D_i, b_i, k_i, T_r, h(\cdot)\}$ in the OBU's security hardware via the manufacturer or a secure channel.
(3) $User_i$ inputs its id_i and pw_i to the OBU_i. OBU_i computes $C_i = c_i \oplus u_i = h(pw_i) \oplus b_i$, $e_i = h(b_i \| id_i \| pw_i)$ and stores $\{C_i, D_i, e_i, k_i, T_r, h(\cdot)\}$ in its security hardware while discards c_i and b_i.

4.2. Login phase

To access the service from VANET, $User_i$ initiates the login process as the following steps and also in Figure 5:

(1) $User_i$ inputs its id_i and pw_i to the OBU_i.
(2) The OBU_i retrieves $b_i = C_i \oplus h(pw_i)$ and verifies if $h(b_i \| id_i \| pw_i)$ and e_i are equal. If $h(b_i \| id_i \| pw_i) = e_i$, it guarantees the correctness of the inputted id_i and pw_i. Otherwise, the login request is rejected.

4.3. General authentication phase

Here, OBU_i involves in authentication process with some law executor vehicle, say LE_j as the following steps and also in Figure 5:

(1) The OBU_i generates a random number r_1 to compute $m_1 = b_i \oplus r_1$, $m_2 = h(r_1 \| id_i \| b_i \| T_o \| T_r)$. OBU_i transmits the authentication request $\{m_1, m_2, D_i, T_o, T_r\}$ to LE_j. T_o is the current timestamp acquired by OBU_i.
(2) On receiving $\{m_1, m_2, D_i, T_o, T_r\}$, the LE_j first checks the freshness of timestamp T_o. If T_o is fresh then the LE_j uses Psk to retrieve $id_i = D_i \oplus h(Psk \| Tr)$, $r_1 = m_1 \oplus h^2(id_i \| x)$. Checks if $h(r_1 \| id_i \| b_i \| T_o \| T_r)$

$User_i$		OBU_i

Login phase

Inputs its id_i and pw_i.

Input $\{id_i, pw_i\}$ $\longrightarrow$

Compute $b_i \leftarrow C_i \oplus h(pw_i)$.
Verify $h(b_i||id_i||pw_i) \stackrel{?}{=} e_i$.
If true, it guarantees the correctness of the $\{id_i, pw_i\}$;
else, it rejects the login request.

OBU_i		LE_i

General authentication phase

Generate a random number r_1.
Compute $m_1 = b_i \oplus r_1$,
$m_2 = h(r_1||id_i||b_i||T_o||T_r)$.

$\{m_1, m_2, D_i, T_o, T_r\}$ $\longrightarrow$

For fresh T_o, compute $id_i = D_i \oplus h(Psk||T_r)$,
$r_1 = m_1 \oplus h^2(id_i||x)$.
Verify $h(r_1||id_i||b_i||T_o||T_r) \stackrel{?}{=} m_2$.
Compute $D_{inew} = h(Psk||T_l) \oplus id_i$,
$d_i = D_{inew} \oplus r_2$,
$m_3 = b_i \oplus r_2$,
$sk_{il} = h(r_1||r_2||id_i||b_i||T_l)$,
$m_4 = h(id_i||b_i||D_{inew}||sk_{ij})$,
$m_5 = a_i \oplus D_{inew}$.

$\longleftarrow$ $\{d_j, m_3, m_4, m_5, T_l\}$

For fresh T_l, compute $r_2 = m_3 \oplus b_i$,
$D_{inew} \leftarrow d_i \oplus r_2$,
$sk_{ij} = h(r_1||r_2||id_i||b_i)$.
Verify $h(id_i||b_i||D_{inew}||sk_{ij}) \stackrel{?}{=} m_4$.
Compute $a_i = m_5 \oplus D_{inew}$.
Keep a_i.

Figure 5. Login and general authentication phases of the proposed scheme.

and m_2 are equal. The equality confirms the legality of OBU_i. Otherwise, the authentication request is rejected believing the integrity breach of the request. LE_j acquires the current timestamp T_l and generates a random number r_2 to compute $D_{inew} = h(Psk||T_l) \oplus id_i$, $d_i = D_{inew} \oplus r_2$, and $m_3 = b_i \oplus r_2$. Next, LE_j computes the session key $sk_{il} = h(r_1||r_2||id_i||b_i||T_l)$ for secure communication, $m_4 = h(id_i||b_i||D_{inew}||sk_{il})$, and $m_5 = a_i \oplus D_{inew}$. LE_j transmits the authentication response message $\{d_j, m_3, m_4, m_5, T_l\}$ to OBU_i.

(3) On receiving $\{d_j, m_3, m_4, m_5, T_l\}$, for fresh T_l, OBU_i retrieves $r_2 = m_3 \oplus b_i$, $D_{inew} = d_i \oplus r_2$, computes the session key $sk_{il} = h(r_1||r_2||id_i||b_i||T_l)$ for secure communication, and checks if $h(id_i||b_i||D_{inew}||sk_{il})$ and m_4 are equal. The equality confirms the trustfulness of LE_j. Otherwise OBU_i terminates the process. OBU_i replaces D_i and T_r with D_{inew} and T_l, respectively. Further, OBU_i retrieves $a_i = m_5 \oplus D_{inew}$ and replaces D_i with D_{inew} and stores a_i in the security hardware.

Now, OBU_i is trustful as it can obtain Psk by computing $Psk = k_i \oplus a_i$. Afterwards, OBU_i helps other mistrustful OBUs to get authenticated.

4.4. Trust-extended authentication phase

This phase is same as in Chuang–Lee's scheme.

4.5. Password change phase

In this phase, $User_i$ can change his/her password without assistance of AS.

(1) $User_i$ inputs id_i and pw_i to the OBU_i.
(2) The OBU_i retrieves $b_i = C_i \oplus h(pw_i)$ and verifies if $h(b_i \| id_i \| pw_i)$ and e_i are equal. If $h(b_i \| id_i \| pw_i) = e_i$, it guarantees the correctness of the inputted id_i and pw_i. Then $User_i$ inputs a new password pw_{inew}. The OBU_i computes $C_{inew} = C_i \oplus h(pw_i) \oplus h(pw_{inew}) = b_i \oplus h(pw_{inew})$, $e_{inew} = h(b_i \| id_i \| pw_{inew})$, and replaces C_i and e_i with replaces C_{inew} with e_{inew}, respectively.

4.6. Secure communication phase

This phase is meant for two trustful vehicles OBU_i and OBU_j to indulge in secure communication with each other as the steps follows and also in Figure 6:

(1) OBU_i acquires the current timestamp T_{oi}, computes $m_6 = h(Psk \| T_{oi}) \oplus id_i$ and $m_7 = h(T_{oi} \| Psk \| id_i)$, and sends $\{m_6, m_7, T_{oi}\}$ to the OBU_j.
(2) On receiving $\{m_6, m_7, T_{oi}\}$, OBU_j retrieves $id_i = m_6 \oplus h(Psk \| T_{oi})$. OBU_j itself computes the value $h(T_{oi} \| Psk \| id_i)$ and compares it with the value retrieved from m_7. The equality of these values confirms the trustfulness of OBU_i. OBU_j acquires the current timestamp T_{oj} to compute $m_8 = h\left(Psk \| T_{oj}\right) \oplus id_j$ and computes the session key $sk_{ij} = h\left(T_{oi} \| T_{oj} \| id_i \| id_j \| Psk\right)$ for secure communication. OBU_j also computes $m_9 = h(id_i \| id_j \| sk_{ij})$ and sends the message $\{m_8, m_9, T_{oj}\}$ to the OBU_i.
(3) On receiving $\{m_8, m_9, T_{oj}\}$, OBU_i first checks the freshness of timestamp T_{oj}. If T_{oj} is fresh then OBU_i retrieves $id_j = m_8 \oplus h(Psk \| T_{oj})$ and computes the session key $sk_{ij} = h\left(T_{oi} \| T_{oj} \| id_i \| id_j \| Psk\right)$ for the secure communication. OBU_i itself computes the value $h(id_i \| id_j \| sk_{ij})$ and compares it with the value retrieved from m_9. The equality of these values confirms the trustfulness of OBU_j.

4.7. Key revocation phase

Key revocation phase is similar to that in Chuang–Lee's scheme.

4.8. Key update phase

This phase is mandatory for every TV for extending their trust state before the key lifetime is about to over. For this, TV seeks the assistance of an LE. The process is described in the following steps and also in Figure 6:

(1) OBU_i generates a random number r_3 to compute $m_{10} = Psk_{old} \oplus r_3$, $m_{11} = Psk_{old} \oplus Msg_{KU}$, and $m_{12} = h(r_3 \| Msg_{KU} \| T_{oii})$. OBU_i sends $\{m_{10}, m_{11}, m_{12}, T_{oii}\}$ as a key update request to LE_j. T_{oii} is the current timestamp acquired by OBU_i.
(2) On receiving $\{m_{10}, m_{11}, m_{12}, T_{oii}\}$, LE_j first checks the freshness of timestamp T_{oii}. If T_{oii} is fresh then LE_j retrieves $r_3 = m_{10} \oplus Psk_{old}$ and $Msg_{KU} = m_{11} \oplus Psk_{old}$. LE_j itself computes the value $h(r_3 \| Msg_{KU} \| T_{oii})$ and compares it with the obtained value m_{12}. The equality of these two values confirms the trustfulness of OBU_i. Then, LE_j generates a random number r_4 to compute $m_{13} = r_4 \oplus h(r_3)$, $m_{14} = Psk_{new} \oplus r_4$, and also computes $m_{15} = h(r_4 \| Psk_{new} \| T_{ll})$. T_{ll} is the current timestamp acquired by LE_j. Further, LE_j computes the session key $sk_{ij} = h(r_3 \| r_4 \| Psk_{new} \| T_{oii} \| T_{ll})$. LE_j sends the reply message $\{m_{13}, m_{14}, m_{15}, T_{ll}\}$ to the OBU_i.
(3) On receiving $\{m_{13}, m_{14}, m_{15}, T_{ll}\}$, OBU_i first checks the freshness of timestamp T_{ll}. If T_{ll} is fresh, then OBU_i retrieves $r_4 = m_{13} \oplus h(r_3)$ and acquires $Psk_{new} = m_{14} \oplus r_4$. OBU_i itself computes the value $h(r_4 \| Psk_{new} \| T_{ll})$ and compares it with the obtained value m_{15}. The equality of these two values confirms the trustfulness of LE_i. OBU_i updates the Psk and computes the session key $sk_{ij} = h(r_3 \| r_4 \| Psk_{new} \| T_{oii} \| T_{ll})$ for the secure communication.

5. SECURITY ANALYSIS OF THE PROPOSED SCHEME

This section discusses the security features of the proposed scheme under the same scenario for which Chuang–Lee's scheme is susceptible. Besides, we also demonstrate other security features.

5.1. Insider attack

During registration phase, $User_i$ submits $\{id_i, h(pw_i) \oplus u_i\}$ to AS, where u_i is a random number. Because hash of $User_i$'s password pw_i is combined with the random number u_i, an insider of AS cannot reveal the value pw_i. Thus, $User_i$'s password is not available for misuse to the insider, and the scheme resists insider attack.

5.2. Provides user anonymity

The plaintext identity of user is not transmitted over open network. An adversary E can intercept the

OBU_i		OBU_j

Secure communication phase

Compute $m_6 = h(Psk||T_{oi}) \oplus id_i$,
$m_7 = h(T_{oi}||Psk||id_i)$.

$\xrightarrow{\{m_6, m_7, T_{oi}\}}$

For fresh T_{oi}, compute $id_i = m_6 \oplus h(Psk||T_{oi})$.
Verify $h(T_{oi}||psk||id_i) \stackrel{?}{=} m_7$.
Compute $m_8 = h(Psk||T_{oj}) \oplus id_j$,
$sk_{ij} = h(T_{oi}||T_{oj}||id_i||id_j||Psk)$,
$m_9 = h(id_i||id_j||sk_{ij})$.

$\xleftarrow{\{m_8, m_9, T_{oj}\}}$

For fresh T_{oj}, compute $id_j = m_8 \oplus h(Psk||T_{oj})$,
$sk_{ij} = h(T_{oi}||T_{oj}||id_i||id_j||Psk)$.
Verify $h(id_i||id_j||sk_{ij}) \stackrel{?}{=} m_9$.

OBU_i		LE_j

Key update phase

Compute $m_{10} = Psk_{old} \oplus r_3$,
$m_{11} = Psk_{old} \oplus Msg_{KU}$,
$m_{12} = h(r_3||Msg_{KU}||T_{oii})$.

$\xrightarrow{\{m_{10}, m_{11}, m_{12}, T_{oii}\}}$

For fresh T_{oi}, compute $r_3 = m_{10} \oplus Psk_{old}$,
$Msg_{KU} = m_{11} \oplus Psk_{old}$.
Verify $h(r_3||Msg_{KU}||T_{oii}) \stackrel{?}{=} m_{12}$.
Compute $m_{13} = r_4 \oplus h(r_3)$,
$m_{14} = Psk_{new} \oplus r_4$,
$m_{15} = h(r_4||Psk_{new}||T_{ll})$,
$sk_{ij} = h(r_3||r_4||Psk_{new}||T_{oii}||T_{ll})$.

$\xleftarrow{\{m_{13}, m_{14}, m_{15}, T_{ll}\}}$

For fresh T_{oi}, compute $r_4 = m_{13} \oplus h(r_3)$,
$Psk_{new} = m_{14} \oplus r_4$.
Verify $h(r_4||Psk_{new}||T_{ll}) \stackrel{?}{=} m_{15}$.
Compute $sk_{ij} = h(r_3||r_4||Psk_{new}||T_{oii}||T_{ll})$.

Figure 6. Secure communication and key update phases of the proposed scheme.

authentication request $\{m_1, m_2, D_i, T_o, T_r\}$ and response $\{d_j, m_3, m_4, m_5, T_l\}$ from the network. To obtain the identity id_i of $User_i$ from $D_i = h(Psk \| T_r) \oplus id_i$, knowledge of the secret parameter *Psk* is necessary. The identity id_i cannot be gained from $m_2 = h(r_1 \| id_i \| b_i \| T_o \| T_r)$ because of one-way property of hash function. To guess id_i from $m_2 = h(r_1 \| id_i \| b_i \| T_o \| T_r)$, knowledge of random number r_1 and user specific value b_i is needed. However, r_1 is a random value that cannot be guessed and $b_i = h^2(id_i \| x)$ involves the secret key x of *AS*. Therefore, the value $m_1 = b_i \oplus r_1$ is of no help in revealing the identity id_i of $User_i$. Hence, the proposed scheme provides user anonymity.

5.3. Secure session key

The session key established between OBU_i and LE_j during general authentication process is given by $sk_{ij} = h(r_1 \| r_2 \| id_i \| b_i \| T_l)$. Assume that E intercepts the authentication request $\{m_1, m_2, D_i, T_o, T_r\}$ and response $\{d_j, m_3, m_4, m_5, T_l\}$ from the network. But E is not able to gain the identity id_i of $User_i$ as explained in Section 5.2. To obtain the random number r_1 from $m_1 = b_i \oplus r_1$, the knowledge of b_i is required and to obtain the secret b_i from $m_1 = b_i \oplus r_1$, the knowledge of random number r_1 is required. Further, r_2 is not retrievable from $d_i = D_{inew} \oplus r_2 = h(Psk \oplus T_l) \oplus id_i \oplus r_2$ and $m_3 = b_i \oplus r_2$ because of being combined with secrets Psk and b_i, respectively. In the absence of values r_1, r_2, id_i, and b_i, adversary E cannot compute the session key sk_{ij}. Besides, the session key sk_{ij} cannot be revealed using $m_4 = h(id_i \| b_i \| D_{inew} \| sk_{ij})$ because of the one-way property of hash function.

The session key established between two trustful vehicles OBU_i and OBU_j during secure communication process is given by $sk_{ij} = h(T_{oi} \| T_{oj} \| id_i \| id_j \| Psk)$. Assume that E intercepts the authentication request $\{m_6, m_7, T_{oi}\}$ and response $\{m_8, m_9, T_{oj}\}$ from the network. To retrieve $User_i$'s identity id_i from $m_6 = h(Psk \| T_{oi} \oplus id_i)$, E requires the knowledge of the secret key Psk. Besides, no constituent value can be retrieved from $m_7 = h(T_{oi} \| Psk \| id_i)$ because of one-way property of hash function. To obtain $User_j$'s identity id_j from $m_8 = h(Psk \| T_{oj}) \oplus id_j$, E again needs the secret key Psk. Further, secret key Psk is stored inside the security hardware of law executor LEs and the trustful vehicles TVs. The value $m_9 = h(id_i \| id_j \| sk_{ij})$ does not allow the session key to be revealed because of one-way property of hash function. Thus, the scheme provides secure session key.

5.4. Resistance to impersonation attack

To impersonate as OBU_i, an adversary E should have access to the $User_i$ related value b_i and $User_i$'s identity id_i, else he cannot compute a valid authentication request. For E can choose two random numbers r_{1E} and b_{iE}, he can compute $m_1 = b_{iE} \oplus r_{1E}$, and take D_i from the network, but he cannot compute the correct $m_2 = h(r_{1E} \| id_i \| b_{iE} \| T_o \| T_r)$ necessary for authentication without having id_i.

To impersonate as an LE in general authentication or as a trusted OBU in trust-extended authentication, E requires access to the secret key Psk. Otherwise, he cannot retrieve the correct value of $User_i$'s identity id_i from D_i, and hence he cannot retrieve the correct value of r_1 from m_1. Without possessing $User_i$'s identity id_i, E cannot compute m_3 and m_4 as both of these values involve either id_i and/or b_i. As a result, in the absence of Psk, E cannot compute a valid response message. Therefore, the proposed scheme resists impersonation attacks.

5.5. Efficient secure communication between TVs

In the proposed scheme, $User_i$'s identity id_i is protected as $m_6 = h(Psk \| T_{oi}) \oplus id_i$ with secret key during Psk during secure communication. E can make a guess for the value of identity id_i, and the timestamp T_{oi} is available from the message $\{m_6, m_7, T_{oi}\}$ transmitted over public network. But E cannot make a guess for the probable value of Psk because of its random nature. Besides, E cannot gain Psk from $m_7 = h(T_{oi} \| Psk \| id_i)$ because of one-way property of hash function. So E cannot compute the secret key Psk using m_6. Without knowing Psk, E can neither initiate nor respond to the secure communication process. Thus, the proposed scheme provides secure communication between two trustful OBUs.

5.6. Resistance to disclosure of new key

Because an adversary E cannot obtain the correct existent secret parameter Psk(say Psk_{old}), he cannot initiate or pass the key update process with LE. Therefore, it is not feasible for anyone except a trustful OBU to obtain a new key Psk_{new} from LE.

5.7. Resistance to replay attack

During general/trust-extended authentication phase, secure communication phase and key update phase, all the transmitted messages contain current timestamps as constituent, and these timestamps are also embedded in the verifying equations. Each of these messages have to first pass the timestamp freshness test and then the verification based on the received verifying equation involving the current timestamps. Thus, replay attack is not applicable in the proposed scheme by virtue of current timestamps.

5.8. Resistance to stolen verifier attack

Since none of the entity, TVs, LEs and AS keep any database storing secrets pertaining to other entities. Thus, stolen verifier attack is not applicable on the proposed scheme.

5.9. Resistance to modification attack

During general/trust-extended authentication process, the equivalence $h(r_1 \| id_i \; \| b_i \| T_i \| T_r) = m_2$ verifies the authenticity of OBU_i to LE/an OBU. For the same process, the equivalence $h(id_i \| b_i \| D_{inew} \| sk_{ij}) = m_4$ verifies the authenticity of LE/an OBU to service seeking OBU_i. During secure communication process between two TVs, the equivalences $h(T_{oi} \| Psk \| id_i) = m_7$ and $h(id_i \| id_j \| sk_{ij}) = m_9$ verify the authenticity of OBU_i to OBU_j and OBU_j to OBU_i, respectively. Similarly, in key update process, the authenticity verifying equation is based on the hash function. Because of one-way property of hash function, an adversary E cannot modify the request and reply messages of any phase of the proposed scheme.

5.10. Resistance to key lifetime self-extension attack

During LE registration process, AS generates a key-set based on hash chain method in such a manner that two consecutive keys from this set are related as: $Psk_1 = h^n(nonce)$

and $Psk_2 = h^{(n-1)}(nonce)$. Therefore, even a trustful OBU cannot infer Psk_2 from Psk_1 and hence cannot extend the lifetime of its existent key Psk_1 of its own. Only a registered LE can extend the lifetime of an existent key of a trustful OBU.

5.11. Resistance to user traceability

In our scheme, the real identity of the user is not transmitted over insecure networks. Further, consider three different phases, authentication, secure communication, and key update phases, being initiated by a same OBU, say OBU_i. It is noticeable that the request messages $\{m_1 = b_i \oplus r_1, m_2 = h(r_1\|id_i\|b_i\|T_o\|T_r), D_i = h(Psk\|T_r) \oplus id_i, T_o, T_r\}$, $\{m_6 = h(Psk\|T_{oi}) \oplus id_i, m_7 = h(T_{oi}\|Psk\|id_i), T_{oi}\}$, and $\{m_{10} = Psk_{old} \oplus r_3, m_{11} = Psk_{old} \oplus Msg_{KU}, m_{12} = h(r_3\|Msg_{KU}\|T_{oii}), T_{oii}\}$, respectively of these three phases are independent of each other. Besides, the value D_i is renewed in every authentication attempt of OBU_i as $D_{inew} = h(Psk\|T_l) \oplus id_i$ and is sent to OBU_i in the form of $d_i = D_{inew} \oplus r_2$. Therefore, two or more authentication request messages of a user, say $User_i$, cannot be identified by E as having originated from $User_i$. Thus, E cannot trace the location of a user by intercepting various messages from the open network.

5.12. Provides fast error detection

During login phase, user inputs its identity id_i and password pw_i to OBU_i. The OBU_i retrieves $b_i = C_i \oplus h(pw_i)$ and checks if $h(b_i\|id_i\|pw_i)$ and e_i are equal. The equivalence $h(b_i\|id_i\|pw_i) = e_i$ guarantees the correctness of the inputted id_i and pw_i. Otherwise, the login request is rejected. Moreover, prior to initiate the authentication, password change or secure communication phase, user has to pass the login phase. Thus, a wrong user cannot be the owner of a specific OBU to enter and access the VANET because of fast error detection capability of OBU.

5.13. Provides choose and change password facility

A user can choose and change his/her password at will without any involvement of the AS in the process. This is a user friendly feature of the proposed scheme.

Table II. Comparison of computation cost.

↓ **Phases** & **Schemes** →	↓ **Entities**	**Ours**	**Chuang–Lee** [38]
Registration	$User_i$	$1h(\cdot) + 1\oplus$	—
	OBU_i	$1h(\cdot) + 1\oplus$	—
	AS	$3h(\cdot) + 3\oplus$	$3h(\cdot) + 2\oplus$
Login	OBU_i	$2h(\cdot) + 1\oplus$	$1h(\cdot) + 1\oplus$
General/Trust-extended Authentication	OBU_i	$3h(\cdot) + 4\oplus$	$8h(\cdot) + 5\oplus$
	LE_j	$7h(\cdot) + 6\oplus$	$8h(\cdot) + 7\oplus$
Password Change	OBU_i	$4h(\cdot) + 3\oplus$	$2h(\cdot) + 3\oplus$
Secure Communication	OBU_i	$5h(\cdot) + 2\oplus$	$5h(\cdot) + 5\oplus$
	OBU_j	$5h(\cdot) + 2\oplus$	$5h(\cdot) + 5\oplus$
Key Update	OBU_i	$4h(\cdot) + 4\oplus$	$6h(\cdot) + 5\oplus$
	LE_j	$4h(\cdot) + 4\oplus$	$5h(\cdot) + 6\oplus$

Table III. Comparison of cecurity features.

Security Threats and Schemes	**Ours**	**Chuang–Lee** [38]
Resistance to insider attack	✓	×
Resistance to session key breach	✓	×
Resistance to impersonation attacks	✓	×
Resistance to disclosure of new key	✓	×
Resistance to user traceability attack	✓	×
Resistance to replay attack	✓	✓
Resistance to stolen verifier attack	✓	✓
Resistance to modification attack	✓	✓
Resistance to key lifetime self-extension attack	✓	✓
Provides user anonymity	✓	×
Provides efficient secure communication	✓	×
Provides fast error detection	✓	✓
Provides choose and change password facility	✓	✓
Provides mutual authentication	✓	×

✓: achieved; ×: not achieved.

5.14. Provides mutual authentication

Because our scheme resists impersonation attacks (Section 5.4) and modification attack (Section 5.9), any two willingly connecting entities (OBU_i&LE_j, OBU_i&OBU_j) can authenticate each other. Thus, our scheme provides mutual authentication.

6. PERFORMANCE COMPARISON

In this section, we analyze the performance of our scheme by comparing it with Chuang–Lee's scheme [38].

6.1. Cost and security requirement analysis

Table II compares the computational load/cost of each phase, and Table III compares the security features of these schemes. We consider only hash and *XOR* operations and neglect string concatenation because of its negligible operational cost. It is noticeable that hash operation is slightly more complex than *XOR* operation. During registration phase, there is no computational load on $User_i$ and OBU_i in Chuang–Lee's scheme. However, in our scheme, $User_i$ and OBU_i, each has computational load of one hash operation and one *XOR* operation. This little increase in computational load protects our scheme from insider attack and also from other problems. For the same phase, *AS* in our scheme computes one *XOR* operation more but one hash operation less than Chuang–Lee's scheme. During login phase, our scheme requires OBU_i to compute only one hash operation more than Chuang–Lee's scheme. During general/trust-extended authentication, our scheme requires remarkably less computational load on both the entities, minimum difference is of one *XOR* operation and maximum difference is of five hash operations. Similarly, in secure communication phase, each OBU_i in our scheme needs to compute three *XOR* operations less than Chuang–Lee's scheme. For password change phase, our scheme requires two hash operations more than Chuang–Lee's scheme. For key update phase, OBU_i requires two hash and one *XOR* operations lesser than that in Chuang–Lee's scheme. For the same phase, LE_i in our scheme requires one hash and two *XOR* operations less than that in Chuang–Lee's scheme. It is noticeable that, our schemes adds computational load in those phases that are executed once or occasionally such as registration and password change phase. For general/trust-extended authentication and secure communication phase that are frequently executed, our scheme is far more lightweight than Chuang–Lee's scheme. However, our scheme remedies a number of security breaches of Chuang–Lee's scheme as apparent from Table III. Thus, the comparative results advocate that the performance of our scheme is better than Chuang–Lee's scheme.

6.2. Simulation results

In this section, we present the pragmatic improvement in authentication performance of the proposed scheme as compared with Chuang–Lee's scheme by using the widely-accepted NS-2 simulation [44]. We largely focus on the setting of a highway with three tracks in different directions. Vehicles are at regular distances, and they travel with a steady speed of $40 \pm 5 m/s$ ($\approx$ 60 to 85 *miles*/*hour*). A grid topology mapping an area of $3000m \times 3000m$ comprises the simulation setting. The *LE*s and normal vehicles occur in a random distribution in the network. We assume that 6% of the vehicles are malicious in our simulation. This scenario allows us to compute the lower limit of the performance by increasing the speed and density of the vehicles to coerce *RSU*s into a high-load state. The parameters and values involved in the calculation are stated in Table IV.

We have used three modules to evaluate the performance of our proposed scheme and Chuang–Lee's scheme. The brief description of these three modules are given later.

VCreation module: In this module, a VANET is created. All the vehicular nodes are randomly positioned in the network area and are connected using wireless links. They can communicate with each other using the wireless medium and move in the network area with inconsistent speeds.

RAnalyze module: In this module, the routing of VANET is analyzed. Moreover, the traffic, communication delay, and packet loss are analyzed as throughput, delay, and energy consumption, respectively. *RSU* sends the master key to the vehicular nodes and the next *RSU* sends the certificates to the next vehicular nodes.

Exe-STEAM module: In this module, our enhanced and secure trust-extended authentication mechanism (STEAM) is executed. When a vehicular node wants to access the service, it needs to perform the login procedure. Next, the *OBU* checks the authentication state itself. If the verification holds, the vehicular node is mistrustful (*MV*) and vice-versa.

Note that NS-2 yields text-based simulation results after simulation. The average of 10 runs yields every individual result of the simulation. Then, we have interpreted these results graphically and interactively.

Table IV. Simulation parameters.

Parameters	Values
Network size	$3000m \times 3000m$
Number of normal vehicles	100
Packet size	512 bytes
Hello message interval	$100ms$
Simulation time	100 s
Transmission range(R)	$100m, 200m, 300m$
Number of LEs	5, 10, 15
Moving speed of vehicle(V)	$10m/s, 20m/s, 30m/s$
MAC protocol	IEEE 802.11 DCF

Figures 7–9 highlight the performance results of the proposed scheme in comparison with Chuang–Lee's scheme, when both are tested on different parameters. Dividing the number of authenticated vehicles by the total number of vehicles computes the percentage of authenticated vehicles and hence the *y*-axis ordinate.

Figure 7 shows the performance comparison of the proposed scheme and Chuang–Lee's scheme with respect to varied transmission range, $LE = 10$ and $V = 20m/s$. We can see that the proposed scheme with varied transmission range (i.e., $100m$, $200m$, and $300m$) is authenticating more number of vehicles than Chuang–Lee's scheme. Figure 8 shows the performance comparison of the proposed scheme and Chuang–Lee's scheme with respect to varied number of *LE*s, $R = 200m$, and $V = 20m/s$. We can see that the proposed scheme with varied number of *LE*s (i.e., 5, 10, and 15) is authenticating more number of vehicles than Chuang–Lee's scheme. Figure 9 shows the performance comparison of the proposed scheme and Chuang–Lee's scheme with respect to varied vehicle speed, $R = 200m$ and $LE = 10$. We can see that the proposed scheme with varied vehicle speed (i.e., $10m/s$, $20m/s$ and $30m/s$) is authenticating more number of vehicles than Chuang–Lee's scheme. For the reason that, in aforementioned different parameters, the communication overhead of the proposed scheme in general authentication phase is lesser as compared with Chuang–Lee's scheme. So greater number of *LE*s, larger transmission range and faster vehicle speed will contribute to a rapid increase in the percentage of authenticated vehicles in the proposed scheme. As a result, the *MV* will have greater chances of meeting a trustful vehicle. Furthermore, the proposed scheme is observed to outperform Chuang–Lee's scheme. This is because of the pivotal role played by the *TV*. It briefly acts as the *LE*s to help with the gradual authentication of the *MV*. Thus, the comparative results advocate that the performance of the proposed scheme is better than Chuang–Lee's scheme.

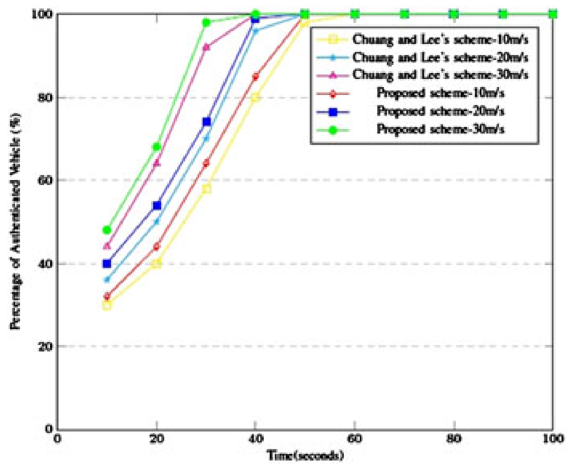

Figure 9. Performance results for varied vehicle speed: $R = 200m$ and $LE = 10$.

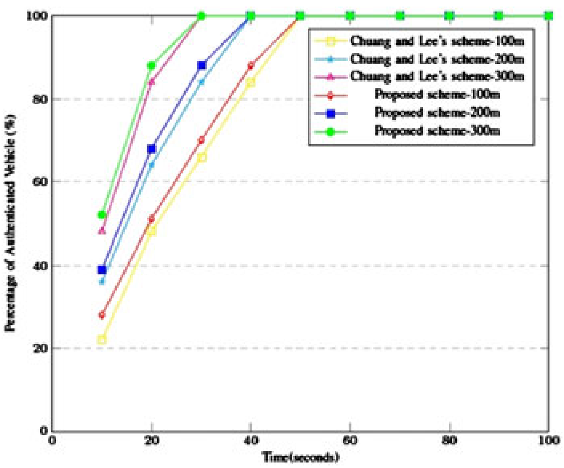

Figure 7. Performance results for varied transmission range: $LE = 10$ and $V = 20m/s$.

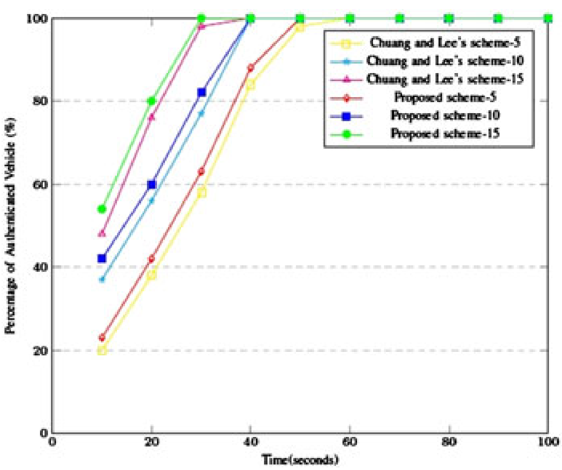

Figure 8. Performance results for varied trnumber of *LE*s: $R = 200m$ and $V = 20m/s$.

7. CONCLUSION

This paper deals with the analysis and enhancement of a recently proposed trust-extended authentication mechanism (TEAM) for VANETs by Chuang and Lee. We explain that identity guessing attack can infect the scheme in many ways causing malfunctioning of V2V communications in VANETs. As a countermeasure, we propose a new secure trust-extended authentication mechanism for VANETs. Through security analysis and performance comparison of the proposed scheme, we show that it efficiently overcomes the shortcomings of TEAM and retains positive attributes of TEAM with considerably low computational load. We have also shown the proficiency of the proposed scheme over Chuang–Lee's scheme through simulation results.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the helpful suggestions of the anonymous reviewers and the Editor. This work was supported by the National Natural Science Foundation of China under Grant nos. 61300220, 61572013, and 61572188. This work was also supported by the Information Security Education & Awareness (ISEA) Phase II Project, Department of Electronics and Information Technology (DeitY), India.

REFERENCES

1. Hubaux JP, Capkun S, Luo J. The security and privacy of smart vehicles. *IEEE Security & Privacy* 2004; **3**: 49–55.
2. IEEE Std 1609.2-2006. IEEE trial-use standard for wireless access in vehicular environments-security services for applications and management messages. *IEEE Vehicular Technology Society Standard* 2006; **1609**: 1–63.
3. Li F, Wang Y. Routing in vehicular ad hoc networks: A survey. *IEEE Vehicular Technology Magazine* 2007; **2**(2): 12–22.
4. Shi Z, Beard C, Mitchell K. Analytical models for understanding space, backoff, and flow correlation in csma wireless networks. *Wireless networks* 2013; **19** (3): 393–409.
5. Liu X, Fang Z, Shi L. Securing vehicular ad hoc networks. *2nd IEEE International Conference on Pervasive Computing and Applications (ICPCA 2007)*, Birmingham, 2007; 424–429. DOI: 10.1109/ICPCA.2007.4365481.
6. Lin X, Sun X, Ho PH, Shen X. Gsis: a secure and privacy-preserving protocol for vehicular communications. *IEEE Transactions on Vehicular Technology* 2007; **56**(6): 3442–3456.
7. Li CT, Hwang MS, Chu YP. A secure and efficient communication scheme with authenticated key establishment and privacy preserving for vehicular ad hoc networks. *Computer Communications* 2008; **31**(12): 2803–2814.
8. Lin X, Lu R, Zhang C, Zhu H, Ho PH, Shen X. Security in vehicular ad hoc networks. *IEEE Communications Magazine* 2008; **46**(4): 88–95.
9. Huang JL, Yeh LY, Chien HY. Abaka: An anonymous batch authenticated and key agreement scheme for value-added services in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology* 2011; **60** (1): 248–262.
10. Jiang Q, Khan MK, Lu X, Ma J, He D. A privacy preserving three-factor authentication protocol for e-health clouds. *Journal of Supercomputing* 2016. DOI: 10.1007/s11227-015-1610-x.
11. Jiang Q, Wei F, Fu S, Ma J, Li G, Alelaiwi A. Robust extended chaotic maps-based three-factor authentication scheme preserving biometric template privacy. *Nonlinear Dynamics* 2016; **83** (4): 2085–2101.
12. Jiang Q, Ma J, Lu X, Tian Y. An efficient two-factor user authentication scheme with unlinkability for wireless sensor networks. *Nonlinear Dynamics* 2015; **8**(6): 1070–1081.
13. Jiang Q, Ma J, Li G, Li X. Improvement of robust smart-card-based password authentication scheme. *International Journal of Communication Systems* 2015; **28**(2): 383–393.
14. Zhang X, Liu C, Nepal S, Pandey S, Chen J. A privacy leakage upper bound constraint-based approach for cost-effective privacy preserving of intermediate data sets in cloud. *IEEE Transactions on Parallel and Distributed Systems* 2013; **24**(6): 1192–1202.
15. Zhang X, Yang LT, Liu C, Chen J. A scalable two-phase top-down specialization approach for data anonymization using mapreduce on cloud. *IEEE Transactions on Parallel and Distributed Systems* 2014; **25**(2): 363–373.
16. Zhang X, Liu C, Nepal S, Chen J. An efficient quasi-identifier index based approach for privacy preservation over incremental data sets on cloud. *Journal of Computer and System Sciences* 2013; **79**(5): 542–555.
17. Zeadally S, Hunt R, Chen YS, Irwin A, Hassan A. Vehicular ad hoc networks (vanets): Status, results, and challenges. *Telecommunication Systems* 2012; **50**(4): 217–241.
18. Raya M, Papadimitratos P, Hubaux JP. Securing vehicular communications. *IEEE Wireless Communications Magazine* 2006; **13**(5): 8–15.
19. Lin X, Sun X, Wang X, Zhang C, Ho PH, Shen XS. Tsvc: timed efficient and secure vehicular communications with privacy preserving. *IEEE Transactions on Wireless Communications* 2008; **7** (12): 4987–4998.
20. Zhang C, Lin X, Lu R, Ho PH, Shen X. An efficient message authentication scheme for vehicular communications. *IEEE Transactions on Vehicular Technology* 2008; **57**(6): 3357–3368.
21. Huang D, Misra S, Verma M, Xue G. Pacp: An efficient pseudonymous authentication-based conditional privacy protocol for vanets. *IEEE Transactions on Intelligent Transportation Systems* 2011; **12**(3): 736–746.
22. Shim KA. Cpas: An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks. *IEEE Transactions on Vehicular Technology* 2012; **61**(4): 1874–1883.
23. Choi J, Jung S. Unified security architecture and protocols using third party identity in v2v and v2i

networks. *Wireless Communications and Mobile Computing* 2012; **12**(15): 1326–1337.
24. Almulla M, Zhang Q, Boukerche A, Ren Y. An efficient k-means authentication scheme for digital certificates revocation validation in vehicular ad hoc networks. *Wireless Communications and Mobile Computing* 2014; **14**(16): 1546–1563.
25. Lee CC, Lai YM. Toward a secure batch verification with group testing for vanet. *Wireless Networks* 2013; **19**(6): 1441–1449.
26. Lu R, Lin X, Liang X, Shen X. A dynamic privacy-preserving key management scheme for location-based services in vanets. *IEEE Transactions on Intelligent Transportation Systems* 2012; **13**(1): 127–139.
27. Xiong H, Beznosov K, Qin Z, Ripeanu M. Efficient and spontaneous privacy-preserving protocol for secure vehicular communication. *2010 IEEE International Conference on Communications (ICC)*, Cape Town, 2010; 1–6.
28. Xiong H, Chen Z, Li F. Efficient and multi-level privacy-preserving communication protocol for vanet. *Computers & Electrical Engineering* 2012; **38** (3): 573–581.
29. Lu R, Lin X, Zhu H, Ho PH, Shen X. Ecpp: Efficient conditional privacy preservation protocol for secure vehicular communications. *The 27th IEEE Conference on Computer Communications (INFOCOM 2008)*, Phoenix, AZ, 2008.
30. Freudiger J, Raya M, Félegyházi M, Papadimitratos P, Hubaux J.P. Mix-zones for location privacy in vehicular networks. *Proceedings of the First International Workshop on Wireless Networking for Intelligent Transportation Systems (Win-ITS)*, Vancouver, 2007.
31. Studer A, Shi E, Bai F, Perrig A. Tacking together efficient authentication, revocation, and privacy in vanets. *6th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON'09)*, Rome, 2009; 1–9.
32. Hsiao HC, Studer A, Chen C, Perrig A, Bai F, Bellur B, Iyer A. Flooding-resilient broadcast authentication for vanets. *Proceedings of the 17th Annual International Conference on Mobile Computing and Networking*, Las Vegas, Nevada, 2011; 193–204.
33. Yeh LY, Chen YC, Huang JL. Paacp: A portable privacy-preserving authentication and access control protocol in vehicular ad hoc networks. *Computer Communications* 2011; **34**(3): 447–456.
34. Horng SJ, Tzeng SF, Wang X, Qiao S, Gong X, Khan MK. Cryptanalysis on a portable privacy-preserving authentication and access control protocol in vanets. *Wireless personal communications* 2014; **79**(2): 1445–1454.
35. Horng SJ, Tzeng SF, Pan Y, Fan P, Wang X, Li T, Khan MK. b-specs+: Batch verification for secure pseudonymous authentication in vanet. *IEEE Transactions on Information Forensics and Security* 2013; **8** (11): 1860–1875.
36. Wang M, Liu D, Zhu L, Xu Y, Wang F. Lespp: Lightweight and efficient strong privacy preserving authentication scheme for secure vanet communication. *Computing* 2014; **98**(7): 1–24.
37. Wu WC, Chen YM. the authentication scheme and access control protocol for vanets. *Entropy* 2014; **16** (11): 6152–6165.
38. Chuang MC, Lee JF. Team: Trust-extended authentication mechanism for vehicular ad hoc networks. *IEEE Systems Journal* 2014; **8**(3): 749–758.
39. Dolev D, Yao AC. On the security of public key protocols. *IEEE Transactions on Information Theory* 1983; **29**(2): 198–208.
40. Das ML. Two-factor user authentication in wireless sensor networks. *IEEE Transactions on Wireless Communications* 2009; **8**(3): 1086–1090.
41. Papadimitratos P, Buttyan L, Holczer TS, Schoch E, Freudiger J, Raya M, Ma Z, Kargl F, Kung A, Hubaux JP. Secure vehicular communication systems: Design and architecture. *IEEE Communications Magazine* 2008; **46**(11): 100–109.
42. Guette G, Bryce C. Using tpms to secure vehicular ad-hoc networks (vanets). In *Information Security Theory and Practices. Smart Devices, Convergence and Next Generation Networks*. Springer, 2008; 106–116.
43. Wagan AA, Mughal BM, Hasbullah H. Vanet security framework for trusted grouping using tpm hardware. *Second International Conference on Communication Software and Networks (ICCSN'10)* 2010: 309–312.
44. The network simulator 2 (ns2) [online]. (Available from: http://www.isi.edu/nsnam/ns/) [Accessed in September 2015].