A Conditional Privacy Preserving Authentication and Multi Party Group Key Establishment Scheme for Real-Time Application in VANETs

SWAPNIL PALIWAL[1] and ANVITA CHANDRAKAR[1]

[1]Vellore Institute of Technology, Vellore, Tamil Nadu 632014 India

Author and Email: Swapnil Paliwal (e-mail: swapnil.paliwal18@gmail.com), Anvita Chandrakar (e-mail: anvita.ch01@gmail.com).

ABSTRACT Vehicular Ad-hoc Networks (VANETs) are a cardinal part of intelligent transportation system (ITS) which render various services in terms of traffic and transport management. The VANET is used to manage growing traffic and manage data about traffic conditions, weather, road conditions, speed of the vehicle, etc. Even though, VANETs are self-sufficient and effective networks but they still suffer from various security and privacy issues. VANETs need to ensure that an adversary should not be able to breach user associated data and delete or modify the exchanged messages for its gains, as these messages comprise of sensitive data. In this paper, we have proposed an authentication and key-agreement protocol based on cryptographic hash functions which makes it lightweight in nature and also suitable for VANET environment. Moreover, to enhance the security and reliability of the entire system, the proposed key-agreement protocol makes use of random session modulus to compute a dynamic session key i.e. for every session, vehicles generate their session specific secret modulus which are then converged to form a common group session key. The formal verification of the proposed work is done using Real – or – Random oracle model, AVISPA and BAN Logic while informal security analysis shows that the proposed protocol can withstand various attacks. The simulation results and analysis prove that the proposed work is efficient and has a real-time application in VANET environment.

Keywords: Password Authentication, VANET, Group-Key generation, Dynamic modulus based Key Exchange Protocol, Lightweight Authentication, Cryptanalysis.

I. INTRODUCTION

Gradually, the concept of smart cities is experiencing substantial research involvement to evolve urban living environment. It has been estimated that around 15 billion investment can be expected for smart city infrastructure and management in few years [31]. One of the top ten smart city technologies which is continuously trending since previous year is vehicle to vehicle (V2V) communications where the vehicles share/receive real-time data [55]. It has been analyzed for over 350 cities that they suffer from 9-66% level of road congestion [54]. The exponential increment in the number of vehicles on the road has led to increment in difficulties for managing and controlling them [33]. Therefore, need for intelligent transportation systems (ITS) emerged which would provide efficient, effective, safe and better transportation and traffic management services. VANET (Vehicular Ad-Hoc Network) falls in the category of such systems. VANET improves driving conditions, traffic monitoring and efficiency and road safety by providing weather conditions, traffic statuses, road warnings, emergency alerts to its vehicles [4, 5, 8, 9, 10, 33, 42]. VANET is a wireless ad-hoc network of vehicles where vehicles perform V2V (vehicle to vehicle) and V2I (vehicle to infrastructure) communication. Before proceeding to the security requirements and challenges of VANET, we provide a brief overview of VANET. VANET comprises of mainly three entities - OBU (On-Board Unit), RSU (Road Side Unit) and TA (Trusted Authority). Each vehicle is equipped with OBU which is a wireless communication device. The OBU is used to communicate with RSU or other OBU for transmission or reception of traffic messages between each other. OBU is a tamper proof device, which is responsible for performing cryptographic operations along with that, each OBU consists of identity of the vehicle, authentication credentials, keys unique to the vehicle, etc. Each vehicle's

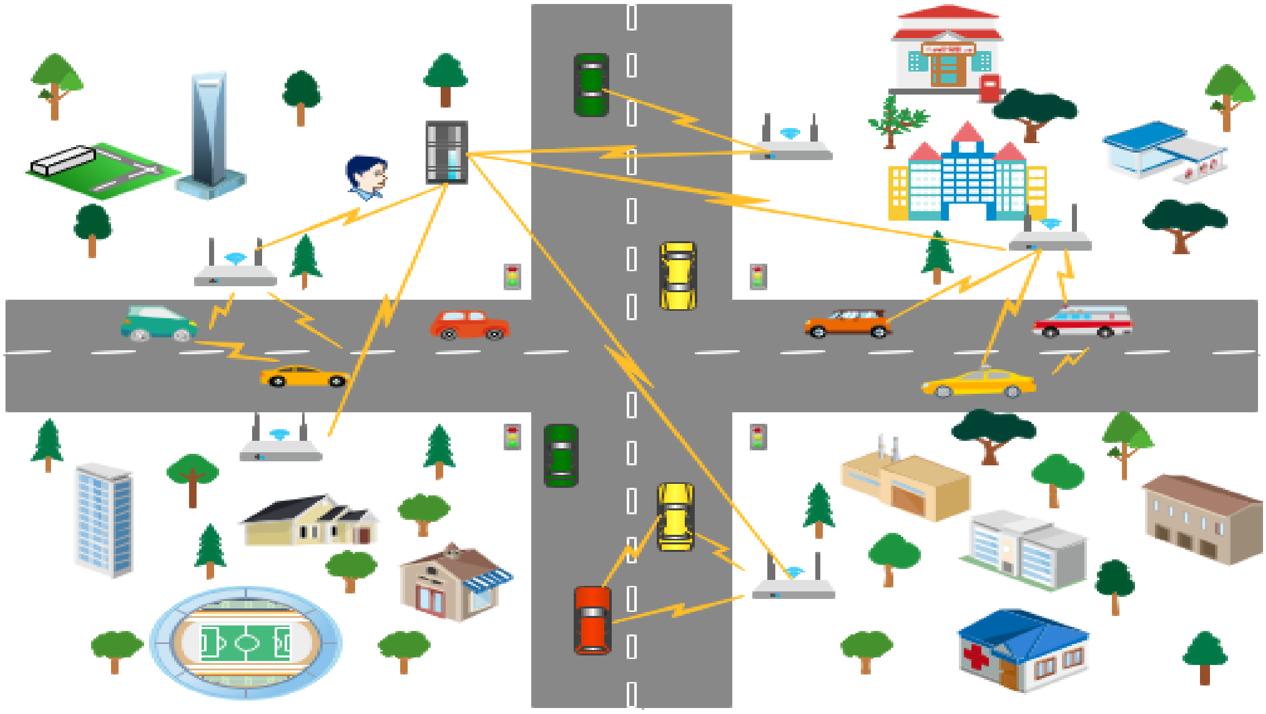


Figure 1. Brief Overview of VANET

OBU should be registered with TA for the OBU to begin any kind of communication with RSU. The OBU presents a provision to change certain information (such as passwords, certificates, etc.) respective to that OBU. All the entities communicate using IEEE 802.11p wireless communication protocol [26]. RSU is a wireless interface located on road side which acts as an intermediary for the communication between OBU and TA. The vehicles are mobile in a particular region communicate their intended message for TA with that region's RSU (The region enclosed by the radio coverage of an RSU becomes that certain RSU's region) [27, 33, 50]. This unit is not completely trusted so any message passed to the RSU should not be interpretable by it. The communications (both V2V and V2I) occurring in VANET follow a short range wireless communication protocol called DSRC (Dedicated Short Range communication, according to DSRC, it requires a vehicle to send one message every 100–300 ms.) protocol [5, 8, 11, 65, 30, 33, 36] which helps the vehicles and RSU to transmit/receive messages with many other vehicles. Moreover, the role of RSU is to handle Infrastructure to Infrastructure (I2I) and Vehicle to Infrastructure (V2I) communications [27]. TA is an internet connection provider which also stores & updates data associated with different vehicles and respective RSUs. Furthermore, it also attains data from the entities within the system. The above data comprises information such as traffic reports, weather condition reports, etc. The TA authenticates each vehicle (where the authentication credentials are generated by the respective OBU) when the vehicle initiates the conversation and can also trace that vehicle so as to detect any malicious behavior of any vehicle in the region. Thus, if the legal entities desire to communicate with TA they must send authentication credentials adhered to which the TA processes and decides whether to accept the given communication request or reject it. In the VANET environment we assume that the TA is highly secure as the corruption of the said entity will result in disruption of the whole system. Moreover, the TA must have high computation power, large storage space and computation ability as to accommodate growing user base. The TA generates and distributes group key to a particular OBU in that RSU region so that the group can communicate with each other and exchange information delineated earlier. This trusted unit holds high computation power and high storage capability so as to store the immense data communicated by numerous vehicles. The proposed group key agreement phase has been motivated by [50], where the trusted authority makes use of every entities modulus and converges on the common group session key, but to make the system dynamic we introduce session specific modulus i.e. every entity generates a session specific modulus, using which the trusted authority generates the common session key and transmits to the vehicles in the region, the above approach makes the system more dynamic and secure even under circumstances when the adversary gains knowledge of an entity's modulus for the given session (the same does not hold for [50]). Moreover, we believe introducing a session specific modulus will further enhance the overall system, as it will provide reliability and security because the modulus value is dynamic for every session and predicting one modulus is equivalent to finding collisions in hash function. We believe that the developed system must be in such a fashion that it can withstand several attacks, thus as per [8, 9, 19, 33, 37, 38, 43, 50, 60, 63, 68, 70] both formal and informal security analysis is performed in security analysis section.

Even though, VANET seems a self-sufficient and effective network but it still suffers from various security, privacy and performance issues. It needs to be ensured that privacy of the vehicle such as identity, location, direction of the vehicle, etc. aren't revealed to ~~a~~ malicious entities. Moreover, the system should be responsively alert, speedy, deliver less computation cost and benign efficacy. Existence of these issues have encouraged various research investments and efforts. In various works, the authentication scheme or key-exchange protocols are based upon either ECC (Elliptic curve cryptosystem) [7] or bilinear pairing. But such protocols encounter high computation overhead due to complex message processing algorithms involved, therefore for a real-time system as VANET, lightweight authentication and key-agreement protocols need to be designed to ensure faster message generation and message authentication [5, 8, 10, 13, 15, 33, 34, 36, 50, 60, 68]. For deploying a lightweight authentication and key agreement scheme cryptographic hash functions based are used because they are computationally faster, exhibit less communication overhead and communication cost [6, 12, 13, 24, 33, 60, 61]. The proposed scheme is lightweight and secure as the hash function used is collision resistant, withstands various attacks because it provides insufficient information to the adversary. The insufficient information collected by the adversary is caused due to the usage of dynamic modulus [16] as the basis of the proposed key-exchange protocol. The main contribution of the paper can be viewed as follows.

(1) A novel lightweight conditional privacy preserving authentication and group key agreement protocol scheme, where the entities make use of secret and dynamic modulus as to enhance security. The proposed work is such that exposure of all current session based group key agreement information will not be sufficient to derive future information of modulus or the session key.

(2) A novel scheme where the authentication process is expedited, here the suppress relay even from the road side unit is detected. To best of our knowledge, it is the only work focusing on dynamic modulus and sequence number based VANET authentication scheme detecting suppress replay attack from the road side unit.

(3) The proposed work makes use of only hash and xor based functions and a sequence number as to expedite the authentication process.

(4) Secure and dynamic group key establishment protocol, where the entities make use of dynamic session modulus, thus every entity generates its unique session specific modulus using which they converges on the common session key.

(5) Cryptanalyzing PW-CPPA-GKA scheme and overcoming the vulnerabilities and weaknesses described in the review (of PW-CPPA-GKA) and the related works section (of different schemes).

In-depth analysis of the proposed scheme is done with regards of both formal (BAN logic, AVISPA and Real-or-Random oracle model) and informal analysis (against various wicked attacks). The presented security analysis proves that the proposed scheme is capable of real-time application in VANET environment. The components of the rest of the paper are as follows. Section 2 presents related works. Section 3 presents review of PW-CPPA-GKA [33], section 4 presents analysis of PW-CPPA-GKA[33], section 5 presents the proposed work , section 6 presents password change phase, section 7 presents security analysis (BAN logic, Simulation in Real-or-Random oracle model, AVISPA and Informal Security Analysis), section 8 presents performance evaluation and section 9 presents conclusion.

II. Related Works

VANET can be described as a mobile network (thus it is a subset of MANET) which uses the mobile vehicles in the network as nodes and provide communication amongst them and the fixed infrastructure nearby [44]. The authentication schemes witnessed for VANET are based on either certificate based conditional privacy preserving authentication (CPPA) or identity based public key cryptography. The area of growing focus is also directing towards hash based conditional privacy preserving authentication schemes. [5] presents a secure and light-weight authentication scheme based on MSR technique for VANETs. The proposed authentication scheme can provide mutual authentication between vehicle and RSU, which can further improve the communication security in VANETs. The said work is however vulnerable as it fails to withstand against password guessing attack, KSSTI attack (as its knowledge can help forge valid credentials) and suppress replay attack. [66] tries to provide an approach for enhancing security in Vehicular Network using ECC and Diffie–Hellman Approach. The proposed work is however vulnerable to suppress replay attack, KSSTI attack (as its knowledge can help derive pseudonym and key related parameters), computational DOS attack and absence of user authentication phase. Proposed works in [5, 45, 46, 66] are possessing a very high storage requirements, thus the TA and $RSUs$ must store all the public and private key pairs of the legitimate user and manage them; thus a large storage space is required to accommodate these systems [50]. It can be evident from both the above approaches that the proposed approaches require high storage space, computation cost and frequent updates as to ensure privacy, these issues result in degrading the overall quality of the system as the VANET requires speedy delivery and processing of data as to accommodate growing traffic demands. Moreover, there is further issue of management as many users will request for revocation of credentials or changing the credentials and managing lists of such requests is a must as to avoid illegal usage. Thus as a result it becomes difficult to adapt such approaches in VANET environment. To reduce the dependency on certificate based CPPA schemes, identity based public key cryptography (PKC) is used where there is no need for storing the certificate data on (OBU) and $RSUs$, the concept of identity based public key cryptography was proposed by Shamir [33] where the public key can be calculated by user's identity (IP address, email address, vehicle number plate, etc.) and the resulting private key can be generated by a private key generator (PKG) which is a trusted party. The identity based approach reduces the burden of storing complex credentials but they put a lot of computational overhead. The proposed work in [63] presents a secure privacy-preserving authentication scheme with cuckoo filter, supporting application for both of V2V communications and V2I communications in VANET. The Cuckoo Filter and the binary search techniques are included to improve the authentication efficiency of the batch message verification phase. The proposed work is of lightweight in nature but it has several security vulnerabilities for instance, there is an absence of a password change phase, thus making it vulnerable to offline guessing attacks. The work in [63] also fails to provide un-linkability. The above is because the authentication credentials in the initial handshake are static, thus the adversary can link the user entering a particular vehicle across different RSU for the first time. [63] requires high storage space to champion a real-time application in VANET. Furthermore, the proposed work is vulnerable to clock synchronization issue (as a direct consequence it is vulnerable to suppress replay) [69] and KSSTI attack (session specific temporary information can help an adversary derive pseudonyms and secret parameters associated with communication). [64] presents a conditional privacy preserving authentication protocol which can be used for V2V and V2I communications or a combination of them. [64] presents a combination of RSU based and TPD (Tamper proof device) based schemes in which the main keys of the network and the vital information of the network are stored in the TPD of RSUs. The proposed work is however, vulnerable to computational DOS attack (due to heavy computational overhead needed to generate and verify a message), suppress replay attack (as it clock synchronization issue exists) and is vulnerable to key compromise attack (as knowledge of this can help an adversary impersonate a legitimate entity). [67] presents a privacy-preserving authentication scheme which achieves full aggregation in VANET, and the scheme satisfies the security requirements in VANET. The proposed work has been divided into two steps (for sign phase) and utilizes a pre-calculation method to reduce the computation cost in sign phase. RSU can aggregate multiple signatures into a single one, and the length of aggregated signature is a constant size which greatly reduces the transmission overhead between RSU and application server and the efficiency of verification for application is improved. The proposed work is however, vulnerable to computational DOS attack (due to heavy computation requirements), suppress replay attack (due to clock synchronization issue) and KSSTI attack (the knowledge of session specific information will help an adversary attain the pseudonym and generate false signatures). Moreover, in [64, 66, 67] the OBU is not enabled with authentication credentials, thus any adversary can gain access to the network without having to provide authentication credentials. The proposed work in [64, 66, 67] will be problematic in areas which requires fast processing (the need of fast processing is a must in the VANET environment as there will be different requests from RSU, TA and vehicle V, which will need to be processed and any overhead in processing will result in overall delay of processing the request, hence the said delay will degrade the overall quality of the system), like VANET. It can be evident from above that both the approaches of certification and identity based cryptography consequence a high computation overhead. To overcome the above issues a lightweight authentication scheme utilizing faster computation techniques such as hashing is used. [33] presents a hash based CPPA and group key agreement scheme which provides efficient authentication message generation and verification. The proposed work has motivated us to design such lightweight systems (utilizing authentication and key agreement phases) which can be implemented in areas which witness sporadic associations in the regions, such as VANET. The proposed work is however vulnerable to various attacks such as known session specific temporary information attack, suppress replay attack, fails to provide un-linkability, mutual authentication of the session key (from second session key exchange message delivered in the system), known session key attack or session key compromise attack. Moreover the proposed work in [33] cannot be secure against entity corruption as well. [50] is another hash based CPPA and group-key agreement scheme which makes use of Chinese Remainder Theorem (CRT) to randomly distribute the group session key amongst the vehicles, the proposed scheme is lightweight in nature and makes use static modulus of every entity for the mode of secure communication. The above approach is secure but it fails to overcome Traceability by adversary (when the identity is known by the adversary), suppress replay attack, provide un-linkability (as the pseudonym is constant, and thus the adversary can link the two message from the same vehicle, similar to [58]), modification by legal entity and lacks a password change phase. [60] presents a decentralized authentication and key agreement protocol for VANETs which supports three different categories of mutual authentication. The proposed scheme is efficient as it uses only one-way hash functions and bitwise XOR operations, thus making it suitable for VANET environment, but the proposed work is vulnerable to clock synchronization issue and as a result it is vulnerable to suppress replay attack. The proposed work in [61] has used simple XOR operations and hash functions too, to design a lightweight authentication and key-agreement protocol, but the proposed work is vulnerable to KSSTI attack, clock synchronization issue, as a consequence, it is vulnerable to suppress replay attack and is also vulnerable to De-synchronization of the session key as the adversary can cause Key-offset Attack. [62] presents an authentication scheme and key agreement scheme fabricated using elliptic curve which ensures user anonymity, the proposed work is however vulnerable to clock synchronization issue, suppress replay attack, password guessing attack, and computational DOS attack (it requires heavy computation). [14] presents a scheme where every entity generates a unique modulus for every communication. It also depicts as of how the same approach can be practically implemented in distributing a group session key. The proposed work in [14] is vulnerable to replay attack, anonymity, impersonation, De-synchronization and Computational DOS attack. [65] presents a scheme which enhances the security of value added services, ensures the minimal delay for request verification, response and service deliveries. The work has several limitations for instance, the identity of the user is used as it is, thus making it vulnerable to impersonation attack. Moreover, the said work does not utilize timestamps, thus it is vulnerable to replay attacks. The proposed work makes extensive use of exponentiations, thus computational DOS attack can be used by an adversary to disrupt the system. Furthermore, the proposed work does not support password change phase, thus successful offline password guessing attack can be used to break the semantic security of the work. The scheme also fails to provide un-linkability, traceability as identity of the user can be fetched by the entity who is observing the public channel.

III. Review of PW-CPPA-GKA

The proposed scheme in [33] presents a lightweight conditional privacy preserving authentication and key agreement protocol suitable for VANET environments. The proposed work has four phases (of interest) (1) Offline Registration of Road Side Unit (RSU) and Vehicle, (2) Authentication Message Generation, (3) Authentication Message Verification and (4) Group-Key generation phase (where vehicles are joining and leaving the region). Now, we provide brief overview of all the phases.

A. RSU Registration Phase

In this phase the Road side units register with TA as follows:

(1) The RSU_i ($\forall i \in N$) securely informs to TA the network to which the said unit is connected.

(2) The TA then selects a unique identity ID_{r_i} and long term key SK_{r_i} which is then communicated to the said RSU_i unit along with prime field $\mathbb{Z}_q^*$, a large prime modulus q over the field $\mathbb{Z}_q^*$ and a secure one way hash function (belonging to a collision resilient family $\mathcal{H}$) H where $\{0,1\}^* \to \{0,1\}^l$. Thus, the TA shares the information $\{\mathbb{Z}_q^*, q, H, ID_{r_i}, SK_{r_i}\}$ with the RSU_i and the RSU_i saves the said information securely.

B. Vehicle Registration Phase

In this phase the vehicle V_j registers with the TA as follows:

(1) The V_j ($\forall j \in N$) selects its own unique identity ID_{v_j} and password PW_{v_j} .

(2) The V_j then chooses a random number b_{v_j} and computes $B_{v_j} = H(PW_{v_j} \| b_{v_j})$. The V_j then sends these credentials $\{ID_{v_j}, PW_{v_j}\}$ to the TA over a secure channel.

(3) The TA upon receiving the credentials, selects a random number e_{v_j} (the probability that two vehicles have the same random number is miniscule) and computes $A_{v_j} = H(x \| e_{v_j})$, $C_{v_j} = A_{v_j} \oplus B_{v_j}$ and $D_{v_j} = H(Id_{v_j} \| B_{v_j} \| A_{v_j})$. Once the computation is complete it embeds the information $\{C_{v_j}, D_{v_j}, e_{v_j}, H, q, \mathbb{Z}_q^*\}$ on the OBU_j .

(4) The TA upon completion of the above step, dispatches the OBU_j to the owner of the said vehicle (via a secure channel) V_j followed by which the owner embeds the random number b_{v_j} on the OBU_j . Thus the following are the credentials on the OBU_j $\{C_{v_j}, D_{v_j}, b_{v_j}, e_{v_j}, H, q, Z_q^*\}$.

C. Authentication Message Generation Phase

In this phase the owner enters the valid credentials which the owner had provided in the registration phase adhered to which the OBU_j at the V_j generates authentication message and an anonymous ID AID_{v_j} . The following are the steps involved in this phase:

(1) The owner enters the ID_{v_j} and PW_{v_j} into the On-Board Unit.

(2) The OBU_j of the said vehicle fetches the random number b_{v_j} from the stored credentials OBU_j and computes $B_{v_j}^* = H(PW_{v_j} \| b_{v_j})$, $A_{v_j}^* = C_{v_j} \oplus B_{v_j}^*$ and $D_{v_j}^* = H(ID_{v_j} \| B_{v_j}^* \| A_{v_j}^*)$.

(3) If $D_{v_j}^* \neq D_{v_j}$ then the OBU_j will reject the login request and ask to re-enter the valid authentication credentials. If it holds i.e. $D_{v_j}^* = D_{v_j}$ is valid then the OBU_j will fetch the current fresh timestamp T_{v_j} and compute $AID_{v_j} = ID_{v_j} \oplus H(A_{v_j} \| T_{v_j})$. Moreover, the OBU_j ge-

nerates a fresh nonce x_{v_j} and computes $y_{v_j} = x_{v_j} \oplus H($ $A_{v_j} \| AID_{v_j})$ and also the message authentication code $atv_j = H(A_{v_j} \| ID_{v_j} \| T_{v_j})$. The OBU_j generates the message $M_{v_j} = \left\langle AID_{v_j}, e_{v_j}, y_{v_j}, T_{v_j}, atv_j \right\rangle$ and sends it to RSU_i over the network (public channel).

(4) The RSU_i upon receiving the message M_{v_j} verifies the freshness of the message i.e. if $(T_{v_j}{}^* - T_{v_j}) \le \Delta T_{v_j}$ holds or not. Here $T_{v_j}{}^*$ is the time at which the RSU_i attained the message M_{v_j}. If $(T_{v_j}{}^* - T_{v_j}) \le \Delta T_{v_j}$ does not hold then the request is rejected else the RSU_i computes $\left\langle ID_{r_i}, M_{v_j}, T_{r_i}, atr_i{}^j \right\rangle$ where T_{r_i} is the fresh timestamp generated by the RSU_i and $atr_i{}^j = H(M_{v_j}$ $\| SK_{r_i} \| T_{r_i})$. Once the above credentials are computed they are sent over the public channel to the TA.

D. Authentication Message Verification Phase

In this phase the TA verifies the attained authentication message from the said RSU_i and if authentication is successful the message is accepted else the message is rejected.

(1) The message $\left\langle ID_{r_i}, M_{v_j}, T_{r_i}, atr_i{}^j \right\rangle$ sent by the RSU_i and received by the TA is initially verified for the freshness of the message for which the TA generates the current timestamp $T_{r_i}{}^*$ and checks if $(T_{r_i}{}^* - T_{r_i})$ $\le \Delta T$ holds or not. If it doesn't then the request is rejected else the message is accepted as fresh. Here, the TA now computes $atr_i{}^{j*} = H(M_{v_j} \| SK_{r_i} \| T_{r_i})$ and A_{v_j} $= H(x \| e_{v_j})$.

(2) If $atr_i{}^{j*} = atr_i{}^j$ holds then TA has authenticated the RSU_j unit and now proceeds by authenticating the message by the vehicle. Now, the TA computes $ID_{v_j}{}^* = AID_{v_j} \oplus H(A_{v_j} \| T_{v_j})$ and along with it computes $atv_j{}^* = H(A_{v_j} \| ID_{v_j}{}^* \| T_{v_j})$. If $atv_j{}^* = atv_j$ holds then the V_j is successfully authenticated by the TA.

Once the authentication is successful then the group-key generation phase occurs.

E. Group-Key Generation Phase

The vehicles in the said RSU_i region must each communicate with a unique session key within the region where the said key will be constantly updated as to ensure new vehicles joining the region are not able to attain the previous group key and the vehicles leaving the region are not able to derive current group session key. We initially depict the first session key exchanged with the initial vehicle V_1 joining the region. The following are the steps:

(1) Following from the authentication phase, once $atv_j{}^* = atv_j$ holds then the TA computes $xv_j = yv_j \oplus H(A_{v_j} \| AID_{v_j})$, adhered to this the TA selects two random numbers $\{zt_1, G_0\} \in \mathbb{Z}_q{}^*$ and computes the current group session key for the said vehicle as $G_1 = x_{v_j} \cdot z_{t_1} \cdot G_0 \bmod(q)$.

(i) The TA selects the current timestamp Tt_1 and then computes $G_1{}' = (xv_j{}^{-1} \cdot G_1 \bmod(q)) \oplus xv_j$ and $Att_1 = H(G_1 \| G_1{}' \| Tt_1)$. Now, the TA unicasts the said message $\left\langle G_1{}', ATT_1, Tt_1 \right\rangle$ to the vehicle.

(ii) Once the OBU_j attains the message, it calculates the group session key by computing $G_1 = (G_1{}' \oplus x_{v_j}) \cdot x_{v_j} \bmod(q)$. Then the OBU_j checks if $H(G_1 \| G_1{}' \| Tt_1)^* \underset{=}{?} Att_1$ holds or not, if it doesn't then the request is rejected and the V_j intimates the TA. If the above condition is satisfied then the V_j accepts the group key.

(2) If another vehicle enters then after the vehicle authentication, the group key is updated as follows: $G_2 = (x_{v_j} \cdot zt_j) \cdot (x_{v_j}{}'.zt_j{}') \cdot G_0 \bmod(q)$ where $(x_{v_j}{}', zt_j{}')$ are the credentials associated with the second vehicle. Thus, the above session key is communicated with both the vehicles; for first vehicle the group key message is communicated as $\left\langle G_2{}'', Att_2{}'', Tt_2 \right\rangle$ where $G_2{}'' = (G_1{}^{-1} \cdot G_2 \bmod(q)) \oplus G_1$ and $Att_2{}'' = H(G_2 \| G_2{}''$ $\| Tt_2)$ and for second vehicle the group key message is communicated as $\left\langle G_2{}', Att_2, Tt_2 \right\rangle$ where $G_2{}'$ $= (xv_j{}^{-1} \cdot G_2 \bmod(q)) \oplus xv_j$ and $Att_2 = H(G_2 \| G_2{}'$ $\| Tt_2)$. Here, xv_j is taken from the second vehicle's initial authentication message. Similarly, the session key is updated and then communicated to all of the vehicles which join the region subsequently.

(3) Here, if third vehicle enters the region then the new key is shared with vehicles $\{v_1, v_2\}$ using the approach depicted in step 2 (thus both receive the same authentication message) and for vehicle v_3, the new key is communicated using the approach depicted in step 1.

(4) If a vehicle exits the region of the RSU_i (it generates the authentication credentials requesting for exiting the region as depicted earlier i.e. the vehicle generates $M_{v_j} = \left\langle AID_{v_j}{}^{**}, e_{v_j}, y_{v_j}{}^{**}, T_{v_j}{}^{**}, atv_j{}^{**} \right\rangle$ and further sends

it to RSU_i which then sends $\left\langle ID_{r_i}, M_{v_j}{}^{**}, T_{r_i}{}^{**}, atr_i{}^{j**} \right\rangle$) then the current session key of n vehicles is updated i.e. G_t is updated with $G^{''}_{t-1}$, where $G_t = \prod_{j=1}^{j=t}(xv_j \cdot zt_j) \cdot G_0 \bmod(q)$, $G^{''}_{t-1} = f_t \cdot G_t \bmod(q)$ (here f_t is a random number $f_t \in \mathbb{Z}_q{}^*$). The said session key is communicated (unicasted) with the remaining vehicles in the region by sending $\{\overline{G^{''}_{t-1}}, \overline{Att_k}, \overline{Tt_t}\}$ to each v_k (old vehicles in the region) where $\overline{G^{''}_{t-1}} = G^{'}_{t-1} \oplus H(\overline{Tt_t} \parallel A_{v_k})$.

IV. Analysis of PW-CPPA-GKA

In this section we highlight several vulnerabilities and weaknesses in the PW-CPPA-GKA [33] scheme. The analysis is done keeping the abilities of an adversary in check. The adversarial abilities and different analysis approaches are taken from [13, 23, 56, 58].

A. Vulnerable to known session-specific temporary information attack

In an authentication scheme with a key agreement protocol, if the exchanged session key is secure even if the session specific temporary information such as nonce (random number), timestamp, etc. are revealed or compromised then the scheme is secure against known session specific temporary information attack [59, 68]. In PW-CPPA-GKA [33] the session key is dependent on x_{v_j} which is a random number selected by the V_j for the group key generation process, thus it is a session specific temporary information. Here if an adversary is given access to x_{v_j} (similar to [59]) then the following are the steps an adversary can take to perform this attack:

Step 1. Once the vehicle V_j gets authenticated it will receive $\left\langle G_a{}', ATT_a, Tt_a \right\rangle$ over a public channel. The adversary can capture the message $\left\langle G_a{}', ATT_a, Tt_a \right\rangle$.

Step 2. The adversary at this instance can compute $G_a = (G_a{}' \oplus x_{v_j}) \cdot x_{v_j} \bmod(q)$ and $H(G_a \parallel G_a{}' \parallel Tt_a)^* \overset{?}{=} Att_a$ to procure and validate the attained key.

Thus, the adversary by the knowledge of temporary session information can attain the session key. Thus, PW-CPPA-GKA [33] fails to withstand known session specific temporary information attack.

B. Suppress replay attack

The entities make use of timestamps as to prevent the replay attacks but due to clock synchronization issue the adversary might intercept a message of the OBU_j from a valid session and again replay it within the time bound (say ΔT) where TA accepts it. The problem will exist due to difference in clock drift rates thus as a result there will exist clock synchronization problem. The following are the steps involved to successfully perform this attack:

Step 1. An active adversary can capture a valid authentication request $M_{v_j} = \left\langle AID_{v_j}, e_{v_j}, y_{v_j}, T_{v_j}, atv_j \right\rangle$ sent by the OBU_j and replay it to TA within the time bound ΔT.

Step 2. The TA now accepts the replayed authentication message M_{v_j} as valid.

Thus, the adversary has been successfully authenticated by the TA. Moreover, if the adversary desires to attain the group session key after exiting the region, the adversary can do so with the help of suppress replay. To proceed with suppress replay, a malicious vehicle V_j (a corrupted entity) can perform following steps:

Step 1. The malicious vehicle V_j can suppress replay the exit request message $M_{v_j} = \left\langle AID_{v_j}{}^{**}, e_{v_j}, y_{v_j}{}^{**}, T_{v_j}{}^{**}, atv_j{}^{**} \right\rangle$ to the TA.

Step 2. the TA will regard this message as valid and as a group joining result (authentication message generation scheme for joining and exiting is same). The above message will be accepted ~~is~~ because the message has been replayed within the time bound ΔT, thus the TA will accept the said request as valid.

Step 3. Once, the credentials of the message are validated then the TA will generate an updated group session key G_{t+1} of the region RSU_i as $G_{i+1} = x_{v_{j+1}} \cdot z_{t_{i+1}} \cdot G_i \bmod(q)$, further the TA will compute $G_{i+1}{}' = (xv_{j+1}{}^{-1} \cdot G_{i+1} \bmod(q)) \oplus xv_{j+1}$ and $Att_{i+1} = H(G_{i+1} \parallel G_{i+1}{}' \parallel Tt_{i+1})$. Once the above credentials are generated then the TA unicasts the said message $\left\langle G_{i+1}{}', ATT_{i+1}, Tt_{i+1} \right\rangle$ to the malicious vehicle.

Thus the session key is now casted to the exited vehicle V_j (in this case, the perfect forward secrecy is hindered, as the exited vehicle under these circumstances is able to attain the group key of the region). Thus, PW-CPPA-GKA [33] is vulnerable to clock synchronization problem and as a result, it is vulnerable to suppress replay attack.

C. Fails to provide un-linkability

In the proposed work of PW-CPPA-GKA [33] the authentication message generated by the OBU_j for entering and exiting from in the RSU_i region is $M_{v_j} = \left\langle AID_{v_j}, e_{v_j}, y_{v_j}, T_{v_j}, atv_j \right\rangle, M_{v_j} = \left\langle AID_{v_j}{}^{**}, e_{v_j}, y_{v_j}{}^{**}, T_{v_j}{}^{**}, atv_j{}^{**} \right\rangle$. The said generated message is then forwarded to the RSU_i, thus the roadside unit generates its authentication message as $\left\langle ID_{r_i}, M_{v_j}, T_{r_i}, atr_i{}^j \right\rangle$. In the proposed work although there is a presence of random pseudonym $AID_{v_j} = ID_{v_j} \oplus H(A_{v_j} \parallel T_{v_j})$, $y_{v_j} = x_{v_j} \oplus H(A_{v_j} \parallel AID_{v_j})$

and $atv_j = H(A_{v_j} \| ID_{v_j} \| T_{v_j})$ due to presence of fresh timestamp T_{v_j} and random number x_{v_j}, but the same e_{v_j} will be used for every authentication message i.e. $M_{v_j}{}^{1} = \left\langle AID_{v_j}{}^{1}, e_{v_j}, y_{v_j}{}^{1}, T_{v_j}{}^{1}, atv_j{}^{1} \right\rangle, ..., M_{v_j}{}^{n} = \left\langle AID_{v_j}{}^{n}, e_{v_j}, y_{v_j}{}^{n}, T_{v_j}{}^{n}, atv_j{}^{n} \right\rangle$, $M_{v_j}{}^{1**} = \left\langle AID_{v_j}{}^{1**}, e_{v_j}, y_{v_j}{}^{1**}, T_{v_j}{}^{1**}, atv_j{}^{1**} \right\rangle, ..., M_{v_j}{}^{n**} = \left\langle AID_{v_j}{}^{n**}, e_{v_j}, y_{v_j}{}^{n**}, T_{v_j}{}^{n**}, atv_j{}^{n**} \right\rangle$. Thus, the adversary can link the same authentication message to a particular anonymous vehicle with the help of the constant e_{v_j}, similar to [58]. Hence, the proposed work fails to provide unlinkability.

D. Fails to provide anonymity if user identity is known

It is possible that an adversary may attain knowledge of identity of the vehicle via some wicked means (such as Dumpster Diving, social engineering, pretext, vishing, phishing or other means) [51]. In the proposed PW-CPPA-GKA [33], the knowledge of identity linked with the random number e_{v_j} within the RSU_i region can disrupt the system as the adversary at any given point further, will be able to trace the vehicle (by observing the public channel and checking for the number e_{v_j} passed over the said RSU network) in a given RSU region. Thus, the anonymous identity AID_{v_j} can no longer provide anonymity.

E. Fails to withstand corrupted entities in the system

It is possible that a V_j is malicious. The proposed work PW-CPPA-GKA [33] is highly vulnerable when there is a presence of a corrupted entity in the RSU_i region. In the said work, the malicious vehicle will receive the new group session key (say G_l) when the said vehicle enters the region. When another vehicle enters the RSU_i region or the malicious vehicle exits the region, it sends the authentication message adhered to which the group session key is updated and communicated to all the vehicles in the region. The following are the steps a corrupted entity can use to disrupt the system:

Step 1. The malicious entity say V_j will block all the group key update messages (for old vehicles in the RSU_i region) $\left\langle G^{''}, Att^{''}, Tt \right\rangle$ or $\left\langle \overline{G^{''}}, \overline{Att^{''}}, \overline{Tt} \right\rangle$ from reaching its destination.

Step 2. Now the malicious entity V_j will generate a new session key using the previous session key G_l. Once, the malicious session key $Key_{Malicious}$ is generated the malicious vehicle will generate authentication message $\left\langle G_{Malicious}{}^{''}, Att_{Malicious}{}^{''}, Tt_{Malicious} \right\rangle$ where $G_{Malicious}{}^{''} = (G_l^{-1} \cdot G_{Malicious} \bmod(q)) \oplus G_l$, $Att_{Malicious}{}^{''} = H(\ G_{Malicious} \| G_{Malicious}{}^{''} \| Tt_{Malicious})$ and $Tt_{Malicious}$ (is timestamp) and multicast the message to old vehicles (by capturing the valid messages of $\left\langle G^{''}, Att^{''}, Tt \right\rangle$ and sending the message $\left\langle G_{Malicious}{}^{''}, Att_{Malicious}{}^{''}, Tt_{Malicious} \right\rangle$ to the same destination as $\left\langle G^{''}, Att^{''}, Tt \right\rangle$.) $V_{old} = \{V_1, V_2, ...V_\lambda\}$.

Step 3. All the vehicles of the set $V_{old} = \{V_1, V_2, ...V_\lambda\}$ will accept this message as valid. The above is because the $Att_{Malicious}{}^{''*} \overset{?}{=} Att_{Malicious}{}^{''}$ will hold.

Thus as a result, the V_j even after exiting the said region will be able to attain the confidential information from the vehicles of the region till the set $V_{old} = \phi$. Moreover, the said entity can still be in the said region and block communication amongst different entity by generating wrong session keys for all the vehicles and multicasting them ($\left\langle G^{''}, Att^{''}, Tt \right\rangle$,$\left\langle G_q{}^{''}, Att_q{}^{''}, Tt_q \right\rangle$,...,$\left\langle G_z{}^{''}, Att_z{}^{''}, Tt_z \right\rangle$, here the session key on each one of them is different, thus making it vulnerable to a key-offset attack [59]). As a result, all the entities (possessing the same session key G_l) will possess different session keys and thus will not be able communicate.

F. Fails to withstand session key compromise attack

The proposed work PW-CPPA-GKA [33] is not secure against the said attack [53]. The knowledge of current session key can help an adversary derive the future group key. The following are the steps an adversary can take to procure future session keys.

Step 1. Assume that an active adversary has the knowledge of current session key $G_{t-1}{}^{'}$ and there is an entry of new vehicle in the region of RSU_i. Thus the adversary can attain the group key by capturing the message $\left\langle G_m{}^{''}, Att_m{}^{''}, Tt_{m2} \right\rangle$ (intended for old vehicles in the region, which has been multicast).

Step 2. Now, the said adversary can proceed by computing $(G_m{}^{''} \oplus G_{t-1}{}^{'}) \cdot G_{t-1}{}^{'} \bmod(q) = G_m$ along with $H(G_m \| G_m{}^{''} \| Tt_{m2})^{*} \overset{?}{=} Att_m{}^{''}$ for attaining and verifying the session key credentials.

Thus, the proposed work in [33] is vulnerable to known session key attack or session key compromise attack.

G. Fails to provide mutual authentication of the session key

In the proposed work PW-CPPA-GKA [33], the session key, when the vehicle V_{k1} joins the region is communicated as $\left\langle G_{k1}{}', ATT_{k1}, Tt_{k1} \right\rangle$. Here, the session key G_{k1} is computed as $[G_{k1}{}' \oplus xv_{k1}] \cdot xv_{k1} \bmod(q) = G_{k1}$. Moreover, when the vehicle V_{k1} exits the region, the updated session key is communicated to each V_k (present in the region, apart from V_{k1}) being $\{\overline{G^{''}}_{t-1}, \overline{Att_k}, \overline{Tt_t}\}$ where $\overline{G^{''}}_{t-1} = G^{'}_{t-1} \oplus H(\overline{Tt_t} \| A_{v_k})$. The mutual authentication exists above but it fails when the group key is communicated to the old vehicle $\left\langle G_t^{''}, Att_t^{''}, Tt_t \right\rangle$ in the region because as depicted earlier it is possible that an adversary under certain circumstances can trick an honest entity to attain and accept the incorrect session key as valid, and the TA will not be aware of such mischief as vehicle never intimates the TA with respect to the session key it had attained. Thus as a result the proposed PW-CPPA-GKA fails to provide mutual authentication of the updated session key.

It is thus evident that PW-CPPA-GKA [33] fails real-time application in VANET environment.

V. Proposed Work

The proposed work comprises of seven phases (i) System Initialization Phase, (ii) RSU registration phase, (iii) Vehicle Registration Phase, (iv) Authentication Message Generation Phase, (v) Authentication Message Verification Phase, (vi) Group Key Generation Phase and (vii) Authentication Message Generation and Verification Phase for Vehicle Exiting Phase.

A. System Initialization

Trusted Authority specifies the requirement for communication and we assume the communication to be secure under such initialization and specifications. The following are the specifications:

(1) Hashing Algorithm $h()$ which is collision resistant function (belonging to a collision resilient family $\mathcal{H}$). The said function takes in a random string input and returns a fixed size output i.e. $\{0,1\}^* \rightarrow \{0,1\}^l$ thus it is referred to as a compression function.

(2) Threshold Modulo Min Range (Min_{Range}) is set for the modulus values. This is necessary as all of the entities will be generating their own modulus value to converge on the group session key, thus they shall generate a modulus such that it is greater than this threshold as to ensure proper exchange environment. The session key will be generated in this Min_{Range} parameter.

(3) Modulus condition Evaluation ($Mod_{Condition}$) rules are enabled. This is essential because if all modulus values (generated by on-board unit of the vehicle) satisfy the given condition ($Mod_{Condition}$); the following will help TA to produce new session key ephemerals conveniently.

(4) Sequence number parameter is enabled (on TA, OBU and RSU). It is used to overcome suppress replay attack and expedite authentication process.

Here, all the above information is shared by TA with all RSUs and OBUs at the time of registration.

B. Road Side Unit Registration

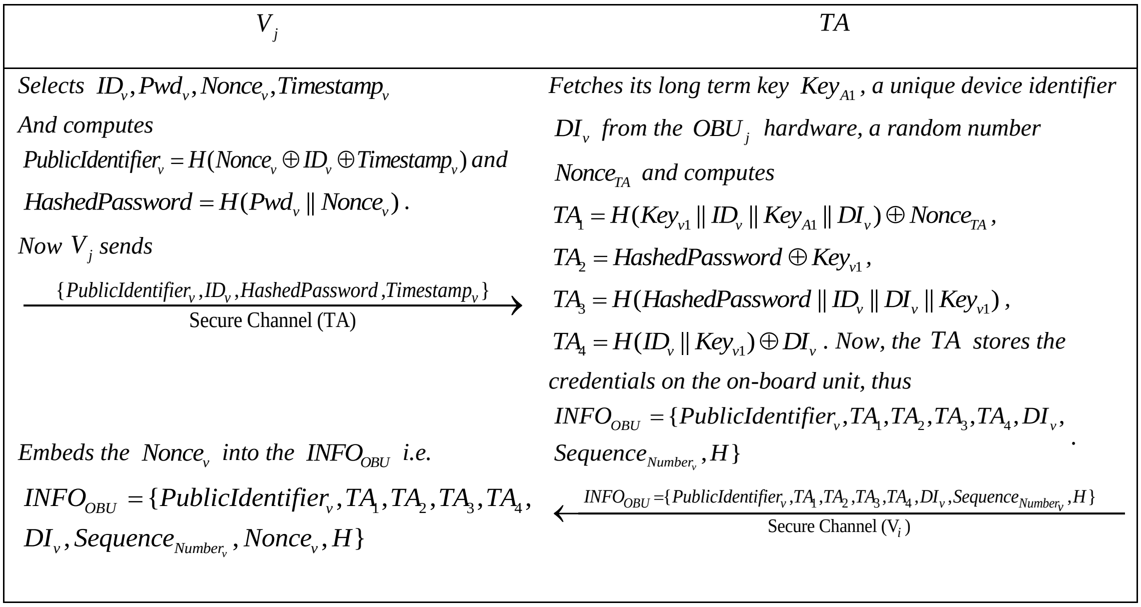
In this phase the Road side units (RSU_i , where $i \in \{1, 2, ...n\}$) are registered securely with the trusted authority TA . Following are the steps:

(1) The RSU_i connected to a network sends its network information to which the said unit is connected to and the unique device identifier (DI_R) such as (ASIN, UUID, CPU serial number, some hardware chip number, etc.) [35] to TA .

(2) The TA then selects a unique identity ID_R , a unique static random pseudonym $PublicIdentifier_R$ (generation scheme of pseudonym for RSU and V are different), a long term secret key Key_{R1} and an empty random sequence number $Sequence_{Number_R}$ which are securely sent as credentials $\{PublicIdentifier_R,\ ID_R, Key_{R1}, Sequence_{Number_R}, H, Min_{Range}, Mod_{Condition}\}$ to the RSU_i . Once the credentials are sent, both the TA and RSU_i store the credentials $\{PublicIdentifier_R, ID_R, Key_{R1}, DI_R, Sequence_{Number_R}\}$.

C. Vehicle Registration Phase

In this phase a vehicle (V_j , where $j \in \{1, 2, ...n\}$) is securely registered with TA such that it can initiate communication with TA in the future (for various purposes). The following are the steps involved:



V_j

Selects $ID_v, Pwd_v, Nonce_v, Timestamp_v$

And computes

$PublicIdentifier_v = H(Nonce_v \oplus ID_v \oplus Timestamp_v)$ *and*

$HashedPassword = H(Pwd_v \| Nonce_v)$.

Now V_j *sends*

$\xrightarrow[\text{Secure Channel (TA)}]{\{PublicIdentifier_v, ID_v, HashedPassword, Timestamp_v\}}$

Embeds the $Nonce_v$ *into the* $INFO_{OBU}$ *i.e.*

$INFO_{OBU} = \{PublicIdentifier_v, TA_1, TA_2, TA_3, TA_4, DI_v, Sequence_{Number_v}, Nonce_v, H\}$

TA

Fetches its long term key Key_{A1} *, a unique device identifier* DI_v *from the* OBU_j *hardware, a random number* $Nonce_{TA}$ *and computes*

$TA_1 = H(Key_{v1} \| ID_v \| Key_{A1} \| DI_v) \oplus Nonce_{TA}$,

$TA_2 = HashedPassword \oplus Key_{v1}$,

$TA_3 = H(HashedPassword \| ID_v \| DI_v \| Key_{v1})$,

$TA_4 = H(ID_v \| Key_{v1}) \oplus DI_v$. *Now, the* TA *stores the credentials on the on-board unit, thus*

$INFO_{OBU} = \{PublicIdentifier_v, TA_1, TA_2, TA_3, TA_4, DI_v, Sequence_{Number_v}, H\}$.

$\xleftarrow[\text{Secure Channel } (V_i)]{INFO_{OBU} = \{PublicIdentifier_v, TA_1, TA_2, TA_3, TA_4, DI_v, Sequence_{Number_v}, H\}}$

Figure 2. Overview of Vehicle Registration Phase

(1) The V_j selects an unique identity ID_V , a password Pwd_v , a random number $Nonce_v$ and the current timestamp $Timestamp_v$.

(2) The V_j computes pseudonym where $PublicIdentifier_v = H(Nonce_v \oplus ID_v \oplus Timestamp_v)$ and hashed password where $HashedPassword = H(Pwd_v \| Nonce_v)$.

(3) The V_j sends the credentials $\{PublicIdentifier_v, ID_v, HashedPassword, Timestamp_v\}$ to the TA securely.

(4) The TA generates a random number $Nonce_{TA}$, a random sequence number $Sequence_{Number_v}$, fetches a unique device identifier from the hardware of the OBU_j and using its long term key Key_{A1} computes $TA_1 = H(Key_{v1} \| ID_v \| Key_{A1} \| DI_v) \oplus Nonce_{TA}$, $TA_2 = HashedPassword \oplus Key_{v1}$, $TA_3 = H(HashedPassword \| ID_v \| DI_v \| Key_{v1})$ and $TA_4 = H(ID_v \| Key_{v1}) \oplus DI_v$. Now, the TA assembles and embeds the computed information as $INFO_{OBU} = \{PublicIdentifier_v, TA_1, TA_2, TA_3, TA_4, DI_v, Sequence_{Number_v}, H\}$ adhered to which the said OBU_j is securely sent to owner of V_j , whereas the TA stores the above credentials securely. Once the owner of the V_j attains the OBU_j , the said owner attaches the device OBU_j on the vehicle V_j and embeds the secret nonce $Nonce_v$ onto the OBU_j . Thus, the complete information present on the OBU_j : $\{PublicIdentifier_v, TA_1, TA_2, TA_3, TA_4, DI_v, Sequence_{Number_v}, Nonce_v, H\}$.

D. Authentication Message Generation for Group Entry Phase

In this phase, the OBU_j of the said vehicle V_j generates authentication message so that it can attain the group session key and communicate with the vehicles in the region. The message generated comprises of authentication code. The following are the steps:

(1) The owner of the vehicle V_j enters the selected identity ID_v and the unique password Pwd_v into the on-board unit OBU_j .

(2) The OBU_j fetches the random number $Nonce_v$ and computes $HashedPassword^* = H(Pwd_v \| Nonce_v)$, $TA_2 \oplus HashedPassword = Key_{v1}{}^*$ and $TA_4 \oplus H(ID_v \| Key_{v1}) = DI_v{}^*$. Once the above credentials are generated then the OBU_j further computes $H(HashedPassword \| ID_v \| DI_v{}^* \| Key_{v1}{}^*)^*$ and checks if $H(HashedPassword^* \| ID_v \| DI_v{}^* \| Key_{v1}{}^*)^* \overset{?}{=} TA_3$ holds or not. If it doesn't then the request is rejected.

(3) If the authentication is successful, the OBU_j now fetches the current timestamp TS_v , nonce $Nonce_{v1}$ and a random sequence number $Sequence_{Number_v}$ (from $INFO_{OBU}$) . Once the above credentials are generated and fetched, the OBU_j computes $H_{\alpha 1} = H(TS_v \| ID_v \| Key_{v1} \| TA_1 \| DI_v) \oplus Nonce_{v1}$, $H_{\alpha 2} = H(Nonce_{v1} \| TS_v$

$\| Key_{v1} \| TA_4 \| Sequence_{Number_v})$ and $H_{\alpha 3} = H(Nonce_{v1} \| Key_{v1} \| ID_v \| TS_v \| TA_1 \| DI_v \| TA_4) \oplus Sequence_{Number_v}$. Furthermore, the session specific moduli generated by the OBU_j is as follows: $h_1 : H(Nonce_{v1} \| TS_v \| DI_v \| Key_{v1})$, $h_2 : H(PublicIdentifier_v \| Key_{v1} \| TS_v \| Nonce_{v1} \| Sequence_{Number_v})$ and $h_3 : H(Key_{v1} \| Nonce_{v1} \| DI_v \| Sequence_{Number_v} \| TS_v \| PublicIdentifier_v)$.

(4) Once the session modulus is generated, the OBU_j checks whether the moduli values are within the threshold value Min_{Range} and verify the $Mod_{Condition}$ condition. If it doesn't hold then the OBU_j generates the authentication message and session modulus in step 3 again. If the conditions (Min_{Range} and $Mod_{Condition}$) hold then the OBU_j sends the message $Mssg_1 = \{PublicIdentifier_v, TS_v, H_{\alpha 1}, H_{\alpha 2}, H_{\alpha 3}\}$ to RSU_i over a public channel.

(5) The RSU_i after receiving the message $Mssg_1 = \{PublicIdentifier_v, TS_v, H_{\alpha 1}, H_{\alpha 2}, H_{\alpha 3}\}$ checks for the freshness of the message i.e. it checks if $(T_R - TS_v) \le \Delta T$ holds (where T_R is the time at which the RSU_i received the message). If the above condition does not hold then the RSU_i rejects the request, else the RSU_i fetches the current timestamp TS_R and then computes the following $H_{\alpha 4} = H(Nonce_{R0} \| Key_{R1} \| ID_R \| TS_R \| Mssg_1) \oplus Sequence_{Number_R}$, $H_{\alpha 5} = H(DI_R \| Mssg_1 \| TS_R \| Key_{R1}) \oplus Nonce_{R0}$, $H_{\alpha 6} = H(Nonce_{R0} \| Key_{R1} \| ID_R \| TS_R \| DI_R \| Sequence_{Number_R})$. Here the $Sequence_{Number_R}$ is initialized with a random number and the RSU_i changes the counter (generates another random number) for every iteration and the change is updated (the stored credentials are updated with this sequence number). Finally, the RSU_i, over a public channel, sends the message $Mssg_2 = \{PublicIdentifier_R, TS_R, H_{\alpha 4}, H_{\alpha 5}, H_{\alpha 6}, Mssg_1\}$ to the TA.

E. Authentication Message Verification Phase

In this phase, the TA proceeds by verifying the authentication message and as a result it decides to either accept or reject the message.

(1) The TA after receiving the message $Mssg_2 = \{PublicIdentifier_R, TS_R, H_{\alpha 4}, H_{\alpha 5}, H_{\alpha 6}, Mssg_1\}$ from the RSU_i, validates for its freshness by checking if $(T_{TA} - TS_R) \le \Delta T$ holds or not. Here T_{TA} is the time at which TA received the message. If the above condition doesn't hold, then the TA rejects the request, else the TA fetches the $PublicIdentifier_R$ and then fetches all the credentials associated with the said identifier ($\{ID_R, DI_R, Key_{R1}\}$). Following this, the TA fetches $H_{\alpha 5}, Mssg_1$ and computes $H_{\alpha 5} \oplus H(DI_R \| Mssg_1 \| TS_R \| Key_{R1}) = Nonce_{R0}{}^*$, $H_{\alpha 4} \oplus H(Nonce_{R0}{}^* \| Key_{R1} \| ID_R \| TS_R \| Mssg_1) = Sequence_{Number_R}{}^*$ adhered to which the TA checks if $Sequence_{Number_R}$ is existing in storage, if it does then the message is rejected.

(2) If the above condition is false, the TA further computes and then checks if $H(Nonce_{R0}{}^* \| Key_{R1} \| ID_R \| TS_R \| DI_R \| Sequence_{Number_R}{}^*)^* \overset{?}{=} H_{\alpha 6}$ holds or not, if it doesn't then the request is rejected. If condition does satisfy then the TA updates $Sequence_{Number_R}$ information.

(3) If the above condition holds then the TA proceeds by validating the message $Mssg_1 = \{PublicIdentifier_v, TS_v, H_{\alpha 1}, H_{\alpha 2}, H_{\alpha 3}\}$. Once the message is fetched, the TA fetches TS_v and checks if $(T_{TA} - TS_v) \le \Delta T$ holds or not. If it doesn't then the request is rejected else, the TA fetches the $PublicIdentifier_v$ and then further fetches all the credentials associated with the said identifier ($\{ID_v, DI_v, Key_{v1}, TA_1, TA_2, TA_3\}$). Subsequent to which the TA computes $H_{\alpha 1} \oplus H(TS_v \| ID_v \| Key_{v1} \| TA_1 \| DI_v) = Nonce_{v1}{}^*$ and $H_{\alpha 3} \oplus H(Nonce_{v1}{}^* \| Key_{v1} \| ID_v \| TS_v \| TA_1 \| DI_v \| TA_4) = Sequence_{Number_v}{}^*$ and checks if $Sequence_{Number_v}$ is existing in storage, if it doesn't then the message is rejected.

(4) If the above condition is false then TA verifies if $H(Nonce_{v1}{}^* \| TS_v{}^* \| Key_{v1} \| TA_4 \| Sequence_{Number_v}{}^*)^* \overset{?}{=} H_{\alpha 2}$ holds. If the condition does satisfy, then the TA updates the $Sequence_{Number_v}$ information (with $NEWSequence_{Number_v}$) and communicates it later (when session key is exchanged). Thus, if the same message is replayed then the request will be rejected as the same sequence number doesn't exist in the storage.

F. Group-Key Generation Phase

In this phase, the authenticated vehicles belonging to the RSU_i regions attain the common group shared key. In this phase, once the vehicle is authenticated, the TA proceeds by generating a session key and distribute it to all the entities in the region. This section depicts the scenario when there is a presence of one vehicle in the region and when there are more than one vehicle in the RSU_i region.

(1) If vehicle V_1 enters the RSU_i region and gets authenticated by the TA then the TA further computes session moduli, $h_1 : H(Nonce_{v1} \| TS_v \| DI_v \| Key_{v1})$,

$h_2 : H(PublicIdentifier_v \,||\, Key_{v1} \,||\, TS_v \,||\, Nonce_{v1} \,||\, Sequence_{Number_v})$, $h_3 : H(Key_{v1} \,||\, Nonce_{v1} \,||\, DI_v \,||\, Sequence_{Number_v} \,||\, TS_v \,||\, PublicIdentifier_v)$ contributed by the vehicle V_1.

(2) Now the TA generates current timestamp TS_{TA} as well as nonce $Nonce_{TA}$ and further computes $NewPublicIdentifier_v = H(Nonce_{v1} \,||\, PublicIdentifier_v \,||\, TS_{TA} \,||\, Key_{v1} \,||\, DI_v \,||\, Nonce_{TA1} \,||\, TS_v \,||\, Sequence_{Number_v})$ adhered to which the TA selects two random numbers $\{a,b\}$ which are not constrained by the Min_{Range} parameter (thus ephemeral value can be greater than the min range defined value and as a result it can be greater than $\{h_1,h_2,h_3\}$ respectively). Once the random numbers are selected, next the TA computes a non-zero $Key_{Session} = [\{a \bmod(h_1)\}\cdot\cdot\{b \bmod(h_2)\}]\bmod(h_3)$.

(3) Since the session key is computed, the TA computes $H_{\alpha 7} = a \,||\, PublicIdentifier_v \,||\, b \,||\, TS_{TA} \,||\, h(b \,||\, PublicIdentifier_v \,||\, Nonce_{v1} \,||\, a \,||\, Key_{v1} \,||\, ID_v \,||\, Nonce_{TA1}) \oplus NEWSequence_{Number_v}$, $H_{\alpha 8} = h(Nonce_{v1} \,||\, TS_v \,||\, Key_{v1} \,||\, PublicIdentifier_v \,||\, TS_{TA}) \oplus Nonce_{TA1} \oplus Nonce_{v1}$ and $H_{\alpha 9} = H(a \,||\, Nonce_{v1} \,||\, PublicIdentifier_v \,||\, Key_{v1} \,||\, New_{Sequence_{Number_v}} \,||\, TS_v \,||\, Nonce_{TA1})$. Following the said computation it unicasts $Mssg_3 = \{PublicIdentifier_v, H_{\alpha 7}, H_{\alpha 8}, H_{\alpha 9}\}$ to the vehicle V_1. Now, the TA updates the new pseudonym and sequence number while maintaining the old credentials (to prevent De-synchronization).

(4) When the OBU_j at vehicle V_j receives the message $Mssg_3$ it fetches TS_{TA} from $H_{\alpha 7}$ and checks if $(T_O - TS_{TA}) \le \Delta T$ holds (where T_O is the fresh timestamp generated by OBU_j when it received the message $Mssg_3$). Accordingly, OBU_j fetches $Nonce_{TA1}$, $NEWSequence_{Number_v}$ from $H_{\alpha 8}, H_{\alpha 9}$ by computing $H_{\alpha 8} \oplus h(Nonce_{v1} \,||\, TS_v \,||\, Key_{v1} \,||\, PublicIdentifier_v \,||\, TS_{TA}) \oplus Nonce_{v1} = Nonce_{TA1}^{*}$, $NEWSequence_{Number_v}^{*} = h(b \,||\, PublicIdentifier_v \,||\, Nonce_{v1} \,||\, a \,||\, Key_{v1} \,||\, ID_v \,||\, Nonce_{TA1}) \oplus NEWSequence_{Number_v} \oplus h(b \,||\, PublicIdentifier_v \,||\, Nonce_{v1} \,||\, a \,||\, Key_{v1} \,||\, ID_v \,||\, Nonce_{TA1}^{*})^{*}$. Now, the unit OBU_j fetches $Nonce_{TA1}$, $NEWSequence_{Number_v}$ and verifies it by checking if computed authentication code value $H(a \,||\, Nonce_{v1} \,||\, PublicIdentifier_v \,||\, Key_{v1} \,||\, NewSequence_{Number_v}^{*} \,||\, TS_v \,||\, Nonce_{TA1}^{*})^{*} \overset{?}{=} H_{\alpha 9}$ holds. If the above condition holds then the OBU_j computes the new pseudonym $NewPublicIdentifier_v = H(Nonce_{v1} \,||\, PublicIdentifier_v \,||\, TS_{TA} \,||\, Key_{v1} \,||\, DI_v \,||\, Nonce_{TA1} \,||\, TS_v \,||\, Sequence_{Number_v})$ together with the non-zero session key $Key_{Session} = [\{a \bmod(h_1)\}\cdot\{b \bmod(h_2)\}]\bmod(h_3)$ and up-dates the $PublicIdentifier_v, NEWSequence_{Number_v}$ content on $INFO_{OBU}$.

(5) If another vehicle say V_2 enters the said RSU_i region and gets authenticated by the TA then the TA computes $h_1^2 : H(Nonce_{v1}^2 \,||\, TS_v^2 \,||\, DI_v^2 \,||\, Key_{v1}^2)$, $h_2^2 : H(PI_v^2 \,||\, Key_{v1}^2 \,||\, TS_v^2 \,||\, Nonce_{v1}^2 \,||\, Sequence_{Number_v}{}^2)$, $h_3^2 : H(Key_{v1}^2 \,||\, Nonce_{v1}^2 \,||\, DI_v^2 \,||\, Sequence_{Number_v}{}^2 \,||\, TS_v^2 \,||\, PI_v^2)$ and attains $\{h_1^2, h_2^2, h_3^2\}$ from V_2, thus the total modulus values are $\{h_1, h_2, h_3, h_1^2, h_2^2, h_3^2\}$. Now, the TA by utilizing the previous session key $Key_{Session}$ generates a random number f (which is again not constrained by Min_{Range} parameter) and computes the new session key using $Key_{New} = [(Key_{Session})\bmod(h_3^c)\cdot\cdot(f)\bmod(h_2^b)]\bmod(h_1^a)$ (where h_1^a, h_2^b, h_3^c are the greatest of the two sets of values $\{h_1, h_2, h_3\}$, $\{h_1^2, h_2^2, h_3^2\}$ attained from $\{V_1, V_2\}$).

(6) Once the new session key is generated, the TA initially, by utilizing the parameters of vehicle V_1 generates another random number a_1 (reduces it with $\bmod(h_1)$) and further computes $[Key_{New}\cdot a_1^{-1}\bmod(h_3)]\bmod(h_3) = b_1$ and checks if reduced b_1 i.e. $b_1 \bmod(h_2)$ still satisfies the condition $Key_{New} = [a_1 \bmod(h_1)\cdot b_1 \bmod(h_2)]\bmod(h_3)$. If it doesn't then the TA regenerates a_1. The TA can also re-compute a_1, b_1 as $a_1 = [a_1 + n\cdot h_1]$, $b_1 = [b_1 + m\cdot h_2]$ ($\forall(n,m)\in N$). Once the new a_1, b_1 is generated, $Mssg_3 = \{PublicIdentifier_v, H_{\alpha 7}, H_{\alpha 8}, H_{\alpha 9}\}$ is computed by TA and sent to vehicle V_1. Similarly for vehicle V_2, the TA generates ephemerals (a_2, b_2) using the approach depicted above and sends $Mssg_3^2 = \{PublicIdentifier_v^2, H_{\alpha 7}^2, H_{\alpha 8}^2, H_{\alpha 9}^2\}$, which is the fresh authentication message to vehicle V_2. Where once the OBU of the vehicle V_2 attains the message it validates it, attains the group session key, updates the sequence number and its pseudonym. Whereas the vehicle V_1 validates the message and attains the group session key.

The same process continues for n vehicles entering a given RSU_i region.

G. Authentication Message Generation and verification for Group Leaving Phase

In this phase, the vehicle V_j desires to exit the RSU_i region. The following are the steps involved in this phase:

Table I. Notations and descriptions

Notations	Description
TA	Trusted Authority.
RSU	Road Side Unit.
OBU	On Board Unit.
$PublicIdentifier_v, PublicIdentifier_{RSU}$	Pseudonym of vehicle or RSU identities.
$NEWPublicIdentifier_v$	New pseudonym of the vehicle
ID_v, ID_R	Identity of the user or Road Side Unit.
Key_{A1}	Long term symmetric key of Trusted Authority.
Key_{v1}, Key_{R1}	Long term symmetric key of OBU and RSU respectively.
$H()$	Secure One-way Cryptographic Hash function.
Min_{Range}	Minimum range value set for all modulus.
$Mod_{Condition}$	Condition a modulus value must satisfy to be accepted.
ΔT	Permissible time delay.
$\|\|$	Concatenation operation.
DI_v, DI_R	It is a unique identifier of the device, either OBU or RSU.
$\oplus$	Bitwise XOR operation.
Pwd_v	Password set on the OBU.
TA_1, TA_4	Secure credentials used for generating authentication credentials.
$Sequence_{Number_v}, Sequence_{Number_R}$	A random sequence number of vehicle and road side unit.
$Key_{Session}$	Current group session key
Key_{New}	New group session key
TS_v	Timestamp selected by the OBU.
TS_R	Timestamp selected by the RSU.
$Nonce_v, Nonce_{v1}, Nonce_{R1}, Nonce_{TA1}$	Random number generated by OBU, RSU and TA respectively

(1) The OBU_j of the vehicle V_j generates a new timestamp TS_{v1} and computes $H_{\alpha 9} = H(a \,\|\, h_1 \,\|\, b \,\|\, Key_{v1} \,\|\, Nonce_{v1} \,\|\, TS_{v1} \,\|\, PublicIdentifier_v \,\|\, Nonce_{TA1})$, $H_{\alpha 10} = H(Nonce_{TA1} \,\|\, Key_{v1} \,\|\, TS_{v1} \,\|\, NEWSequence_{Number_v} \,\|\, a)$, $H_{\alpha 11} = H(NewPublicIdentifier_v \,\|\, TS_{TA} \,\|\, TS_{v1} \,\|\, b \,\|\, Nonce_{v1})$ (here $NEWSequence_{Number_v}, Nonce_{v1}$ are from the messages $Mssg_3, Mssg_1$) adhered to which it sends the message $Mssg_4 = \{NewPublicIdentifier_v, TS_{v1}, H_{\alpha 9}, H_{\alpha 10}, H_{\alpha 11}\}$ to RSU_i.

(2) The RSU_i as before, generates the authentication message $Mssg_5$ similar to $Mssg_2$ and sends the authentication message to TA.

(3) TA authenticates the request from RSU_i and updates the credentials. For authenticating the message $Mssg_4$, the TA initially validates its freshness. If the message is fresh then the TA fetches the public identifier information and its associated credentials adhered to which TA checks if any of $H_{\alpha 9}{}^{*} \overset{?}{=} H_{\alpha 9}$, $H_{\alpha 10}{}^{*} \overset{?}{=} H_{\alpha 10}$, $H_{\alpha 11}{}^{*} \overset{?}{=} H_{\alpha 11}$ holds. If it holds then the authentication is successful, else the request is rejected. Now, the TA marks the use of $NEWSequence_{Number_v}$ for exit purpose i.e. if TA witnesses the same sequence number $NEWSequence_{Number_v}$ from the same vehicle for an exit purpose, the request will be rejected. Thus, the $NEWSequence_{Number_v}$ can only be used for joining purpose in the next iteration.

(4) In the above scenario once the vehicle is successfully authenticated, the TA will update the existing hash tuple i.e. it will remove all the hashes contributed by the vehicle V_j and attain new $\{h_1{}^{a}, h_2{}^{b}, h_3{}^{c}\}$, $\{h_1{}^{\beta}, h_2{}^{\beta}, h_3{}^{\beta}, \ldots, h_1{}^{\alpha}, h_2{}^{\alpha}, h_3{}^{\alpha}\}$ adhered to which the TA generates a random number f^{α} and generates new session key as $Key_{New}{}^{\alpha+1} = [(Key_{Session}{}^{\alpha}) \bmod (h_3{}^{c}) \cdot \cdot (f^{\alpha}) \bmod (h_2{}^{b})] \bmod (h_1{}^{a})$.

Once the new session key is generated the TA generates (a_α, b_α) pair for all the existing vehicles in the region and communicates the same using message $Mssg_6$ which is similar to that of $Mssg_3$.

VI. Password Change Phase

In this phase, the vehicle V_j desires to change the password set on the OBU_j with new password. The following are the steps involved in this phase:

(1) The owner of the vehicle V_j enters the selected identity ID_v and the unique password Pwd_v into the on-board unit OBU_j. The OBU_j fetches the

V_j	RSU_i	TA
After successful authentication the V_j *fetches its public identifier* $PublicIdentifier_v$, *a random number* $Nonce_{v1}$, *the device identifier* DI_v *and the sequence number* $Sequence_{Number_v}$ *adhered to which* V_j *computes* $H_{\alpha1}=H(TS_v \parallel ID_v \parallel Key_{v1} \parallel TA_1 \parallel DI_v)\oplus Nonce_{v1}$, $H_{\alpha2}=H(Nonce_{v1} \parallel TS_v \parallel Key_{v1} \parallel TA_4 \parallel Sequence_{Number_v})$, $H_{\alpha3}=H(Nonce_{v1} \parallel Key_{v1} \parallel ID_v \parallel TS_v \parallel TA_1 \parallel DI_v \parallel TA_4)\oplus Sequence_{Number_v}$ *and generates the secret session specific modulus* $h_1 : H(Nonce_{v1} \parallel TS_v \parallel DI_v \parallel Key_{v1})$, $h_2 : H(PublicIdentifier_v \parallel Key_{v1} \parallel TS_v \parallel Nonce_{v1} \parallel Sequence_{Number_v})$, $h_3 : H(Key_{v1} \parallel Nonce_{v1} \parallel DI_v \parallel Sequence_{Number_v} \parallel TS_v \parallel PublicIdentifier_v)$. *In case when the vehicle is exiting the region it sends* $Mssg_4$.	*The* RSU_i *checks if* $(T_R - TS_v)\le \Delta T$ *holds, if it does then the* RSU_i *fetches its credentials, generates random number* $Nonce_R$, *a sequence* $Sequence_{Number_R}$ *and a current timestamp* TS_R *and computes* $H_{\alpha4}=H(Nonce_{R0} \parallel Key_{R1} \parallel ID_R \parallel TS_R \parallel Mssg_1)\oplus Sequence_{Number_R}$, $H_{\alpha5}=H(DI_R \parallel Mssg_1 \parallel TS_R \parallel Key_{R1})\oplus Nonce_{R0}$, $H_{\alpha6}=H(Nonce_{R0} \parallel Key_{R1} \parallel ID_R \parallel TS_R \parallel DI_R \parallel Sequence_{Number_R})$.	*The* TA *checks if* $(T_{TA} - TS_R)\le \Delta T$ *holds, if it does then the* TA *fetches the* $Nonce_{R0}, Sequence_{Number_R}$ *by computing* $H_{\alpha5}\oplus H(DI_R \parallel Mssg_1 \parallel TS_R \parallel Key_{R1}) = Nonce_{R0}^{*}$, $H_{\alpha4}\oplus H(Nonce_{R0}^{*} \parallel Key_{R1} \parallel ID_R \parallel TS_R \parallel Mssg_1) = Sequence_{Number_R}^{*}$ *and checks if the sequence number exists in the storage, if the above doesn't hold then the* TA *checks if* $H_{\alpha6}^{*}\overset{?}{=}H_{\alpha6}$ *holds. Now* TA *checks if* $(T_{TA} - TS_v)\le \Delta T$ *holds, if the above condition holds then the* TA *fetches the* $Nonce_{v1}, Sequence_{Number_v}$ *by computing* $H_{\alpha1}\oplus H(TS_v \parallel ID_v \parallel Key_{v1} \parallel TA_1 \parallel DI_v) = Nonce_{v1}^{*}$, $H_{\alpha3}\oplus H(Nonce_{v1}^{*} \parallel Key_{v1} \parallel ID_v \parallel TS_v \parallel TA_1 \parallel DI_v \parallel TA_4) = Sequence_{Number_v}^{*}$ *And checking if the said* $Sequence_{Number_v}$ *exists in the storage, then the* TA *checks if* $H_{\alpha2}^{*}\overset{?}{=}H_{\alpha2}$ *holds, if above holds then the* TA *computes* h_1, h_2, h_3 *adhered to which the* TA *updates the existing hash value set and attains* $\{h_1^{a}, h_2^{b}, h_3^{c}\}$ *(in case of vehicle exiting the hash values set is reduced). Once the above credentials are attained then the* TA *computes the new session key* Key_{New} *and computes* (a_α, b_α) *pairs for all the vehicles in the region. The* TA *now generates* $Mssg_3$ *or* $Mssg_6$ *for all the vehicles.*
$\xrightarrow[\text{Public Channel (RSU}_i\text{)}]{Mssg_1/Mssg_4}$	$\xrightarrow[\text{Public Channel (TA)}]{Mssg_2/Mssg_5}$	$\xleftarrow[\text{Public Channel (V}_1,V_2,V_3,...,V_n\text{)}]{Mssg_3/Mssg_6,...,Mssg_3^{n}/Mssg_6^{n}}$
After attaining the message $Mssg_3 / Mssg_6$ OBU_j *checks if* $(T_O - TS_{TA})\le \Delta T$ *holds, if it does then the* OBU_j *computes* $H_{\alpha8}\oplus h(Nonce_{v1} \parallel TS_v \parallel Key_{v1} \parallel PublicIdentifier_v \parallel TS_{TA})\oplus Nonce_{v1} = Nonce_{TA1}^{*}$, $NEWSequence_{Number_v}^{*} = h(\ b \parallel PublicIdentifier_v \parallel Nonce_{v1} \parallel a \parallel Key_{v1} \parallel ID_v \parallel Nonce_{TA1})\oplus NEWSequence_{Number_v} \oplus h(b \parallel PublicIdentifier_v \parallel Nonce_{v1} \parallel a \parallel Key_{v1} \parallel ID_v \parallel Nonce_{TA1}^{*})^{*}$ *And checks if* $H_{\alpha9}^{*}\overset{?}{=}H_{\alpha9}$ *holds, if it does then the* OBU_j *computes the session key* $Key_{Session} = [a \bmod(h_1)\cdot b \bmod(h_2)] \bmod(h_3)$ *and updates the sequence number and pseudonym, if it is a new entity in the region.*		

Figure 3. **Brief Overview of authentication and key-agreement protocol**

random number $Nonce_v$ and in addition to it the OBU_j computes $HashedPassword^* = H(\ Pwd_v \,\|\, Nonce_v)$, $TA_2 \oplus HashedPassword = Key_{v1}{}^*$, $TA_4 \oplus H(ID_v \,\|\, Key_{v1}) = DI_v{}^*$ adhered to which the OBU_j further computes $H(HashedPassword \,\|\, ID_v \,\|\, DI_v{}^* \,\|\, Key_{v1}{}^*)^*$ and checks if $H(HashedPassword^* \,\|\, ID_v \,\|\, DI_v{}^* \,\|\, Key_{v1}{}^*)^* \overset{?}{=} TA_3$ holds or not.

(2) Once the condition holds, the OBU_j requests the user to enter new password credentials. The owner of the vehicle V_j enters the new password $Pwd_v{}'$ and the OBU_j updates the credentials by setting $HashedPassword = H(Pwd_v{}' \,\|\, Nonce_v)$, $TA_2{}' = HashedPassword' \oplus Key_{v1}$, $TA_3{}' = H(Hashed\ Password' \,\|\, ID_v \,\|\, DI_v \,\|\, Key_{v1})$, $TA_4 = H(\ ID_v \,\|\, Key_{v1}) \oplus DI_v$ and embeds the new information $INFO_{OBU} = \{PublicIdentifier_v, TA_1,\ TA_2{}'\ , TA_3{}', TA_4\ , DI_v, H\}$ on the on-board unit OBU_j .

VII. Security Analysis

In this section, we provide analysis so that it is evident to the readers as of how the proposed scheme is secure against various attacks.

A. Formal Verification: Authentication Proof of our scheme using BAN Logic

BAN logic is used for examining the security of authentication protocol. Our aim is to ensure that successful authentication is taking place (we have defined various goals and want to ensure that they are achieved), we use the logic defined in [20, 21, 22, 15, 40] to provide security proof of our protocol. For attaining the results we have utilized approach similar to that of [22, 15]. Following are the set of rules.

1. **Message Meaning Rule:** $R_1 : \dfrac{P \mid\equiv P \xleftrightarrow{K} Q, P \triangleleft \{X\}_K}{P \mid\equiv Q \mid\sim X}$

2. **Nonce Verification Rule:** $R_2 : \dfrac{P \mid\equiv \#(X), P \mid\equiv Q \mid\sim X}{P \mid\equiv Q \mid\equiv X}$

3. **Jurisdiction Rule:** $R_3 : \dfrac{P \mid\equiv Q \Rightarrow X, P \mid\equiv Q \mid\equiv X}{P \mid\equiv X}$

4. **Freshness Rule:** $R_4 : \dfrac{P \mid\equiv \#(X)}{P \mid\equiv \#(X, Y)}$

5. **Belief Rule:** $R_5 : \dfrac{P \mid\equiv (X, Y)}{P \mid\equiv (X)}$

6. **Seeing Rule:** $R_6 : \dfrac{P \triangleleft (X, Y)}{P \triangleleft (X)}$

7. **ER1 or Hash Verification Rule (Fetched from [15]):** $R_7 : \dfrac{P \mid\equiv Q \xleftrightarrow{K} P, P \triangleleft f(X, Y)}{P \mid\equiv Q \mid\sim X}$

8. **Session Key Rule:** $R_8 : \dfrac{P \mid\equiv \#(X), P \mid\equiv Q \mid\equiv X}{P \mid\equiv P \xleftrightarrow{K} Q}$

9. **Multipart Message:** $R_9 : \dfrac{P \mid= X, P \mid= Y}{P \mid= (X, Y)}$

10. **Hash Inference Rule:** $R_{10} : \dfrac{P \mid\equiv Q \mid\sim H(X), P \triangleleft X}{P \mid\equiv Q \mid\sim X}$

Now, we highlight the belief within the protocol adhered to which we proceed further. Initial assumptions (beliefs) are,

(1) $V \mid\equiv V \xleftrightarrow{PublicIdentifier_v, Key_{v1}, TA_1, TA_4, DI_v, ID_v} TA$ or $TA \mid\equiv V \xleftrightarrow{PublicIdentifier_v, Key_{v1}, TA_1, TA_4, DI_v, ID_v} TA$,

(2) $RSU \mid\equiv RSU \xleftrightarrow{PublicIdentifier_R, Key_{R1}, DI_R, ID_R} TA$ or $TA \mid\equiv RSU \xleftrightarrow{PublicIdentifier_R, Key_{R1}, DI_R, ID_R} TA$,

(3) $TA \mid\equiv RSU \Rightarrow (H_{\alpha 4}, H_{\alpha 5}, H_{\alpha 6}, Sequence_{Number_R})$,

(4) $TA \mid\equiv V \Rightarrow (H_{\alpha 1}, H_{\alpha 2}, H_{\alpha 3})$,

(5) $V \mid\equiv TA \Rightarrow (H_{\alpha 7}, H_{\alpha 8}, H_{\alpha 9})$,

(6) $TA \mid\equiv \#(Timestamp_v, Timestamp_R, Timestamp_{TA})$,

(7) $V \mid\equiv \#(Timestamp_v, Timestamp_{TA})$,

(8) $RSU \mid\equiv \#(Timestamp_v, Timestamp_R)$.

The goals we want to achieve are:

(1) $TA \mid\equiv \#(H_{\alpha 4}, H_{\alpha 5}, H_{\alpha 6})$,

(2) $TA \mid\equiv RSU \mid\equiv (H_{\alpha 4}, H_{\alpha 5}, H_{\alpha 6})$,

(3) $TA \mid\equiv \#(Sequence_{Number_R})$,

(4) $TA \mid\equiv TA \xleftrightarrow{Sequence_{Number_R}} RSU$,

(5) $TA \mid\equiv \#(H_{\alpha 1}, H_{\alpha 2}, H_{\alpha 3})$,

(6) $TA \mid\equiv V \equiv (H_{\alpha 1}, H_{\alpha 2}, H_{\alpha 3})$,

(7) $TA \mid\equiv TA \xleftrightarrow{h_1, h_2, h_3} V$,

(8) $TA \mid\equiv TA \xleftrightarrow{Key_{session}} V$,

(9) $V \mid\equiv \#(H_{\alpha 7}, H_{\alpha 8}, H_{\alpha 9})$,

(10) $V \mid\equiv TA \equiv (H_{\alpha 7}, H_{\alpha 8}, H_{\alpha 9})$,

(11) $V \mid\equiv Sequence_{Number_V}$,

(12) $V \mid\equiv TA \xleftrightarrow{Sequence_{Number_V}} V$,

(13) $V \mid\equiv TA \xleftrightarrow{Key_{Session}} V$,

(14) $V \mid\equiv TA \mid\equiv TA \xleftrightarrow{Key_{Session}} V$,

(15) $TA \mid\equiv V \mid\equiv TA \xleftrightarrow{Key_{Session}} V$.

We now represent the messages in the idealized form, thus we get $RSU \triangleleft Mssg_1 : PublicIdentifier_V, Timestamp_v, H_{\alpha 1}, H_{\alpha 2}, H_{\alpha 3}$, $TA \triangleleft Mssg_2 : PublicIdentifier_R, Timestamp_R, H_{\alpha 4}, H_{\alpha 5}, H_{\alpha 6}, Mssg_1$, $V \triangleleft PublicIdentifier_v, H_{\alpha 7}, H_{\alpha 8}, H_{\alpha 9}$.

We now provide the proof of our scheme using the BAN logic.

A. Using the message $Mssg_2$, rule R_6 we get $TA \triangleleft (H_{\alpha4}, H_{\alpha5}, H_{\alpha6})$.

B. Using previous from step A, belief 2 and rule R_7 we get $TA \mid\equiv RSU \mid\sim (H_{\alpha4}, H_{\alpha5}, H_{\alpha6})$.

C. With the result of step A, belief 6 and rule R_4 we get $TA \mid\equiv \#(H_{\alpha4}, H_{\alpha5}, H_{\alpha6})$. **GOAL 1**

D. With the result of step B, C and rule R_2 we get $TA \mid\equiv RSU \mid\equiv (H_{\alpha4}, H_{\alpha5}, H_{\alpha6})$. **GOAL 2**

E. With the result of steps D, belief 3 and rule R_3 we get $TA \mid\equiv (H_{\alpha4}, H_{\alpha5}, H_{\alpha6})$.

F. Using message $Mssg_2$, with the result of step E and rule R_5 we derive an important result i.e. we attain $TA \mid\equiv Sequence_{Number_R} \left(\dfrac{TA \mid\equiv (Sequence_{Number_R}, (H_{\alpha4}))}{TA \mid\equiv Sequence_{Number_R}} \right)$.

G. Using the result of step C and rule R_4 we get $TA \mid\equiv \#(Sequence_{Number_R})$. **GOAL 3**

H. With the result of step C, D and rule R_8 we get $TA \mid\equiv TA \xleftrightarrow{Sequence_{Number_R}} RSU$. **GOAL 4**

I. Using the message $Mssg_1$, rule R_6 we get $TA \triangleleft (H_{\alpha1}, H_{\alpha2}, H_{\alpha3})$.

J. With the result of step I, belief 2 and rule R_7 we get $TA \mid\equiv RSU \mid\sim (H_{\alpha1}, H_{\alpha2}, H_{\alpha3})$.

K. With the result of step I, belief 6 and rule R_4 we get $TA \mid\equiv \#(H_{\alpha1}, H_{\alpha2}, H_{\alpha3})$. **GOAL 5**

L. With the result of step J, K and rule R_2 we get $TA \mid\equiv V \mid\equiv (H_{\alpha1}, H_{\alpha2}, H_{\alpha3})$. **GOAL 6**

M. With the result of steps L, belief 4 and rule R_3 we get $TA \mid\equiv (H_{\alpha1}, H_{\alpha2}, H_{\alpha3})$.

N. With the result of step K and rule R_4 we get $TA \mid\equiv \#(Nonce_{v1}, Sequence_{Number_V})$.

O. With the result of step K, L and rule R_8 we get $TA \mid\equiv TA \xleftrightarrow{h_1, h_2, h_3} V$. **GOAL 7**

P. With the result of step K, L and rule R_8 we get $TA \mid\equiv TA \xleftrightarrow{Key_{Session}} V$. **GOAL 8**

Q. Using the message $Mssg_3$, rule R_6 we get $V \triangleleft (H_{\alpha7}, H_{\alpha8}, H_{\alpha9})$.

R. With the result of step Q, belief 1 and rule R_7 we get $V \mid\equiv TA \mid\sim (H_{\alpha7}, H_{\alpha8}, H_{\alpha9})$.

S. With the result of step R, belief 7 and rule R_4 we get $V \mid\equiv \#(H_{\alpha7}, H_{\alpha8}, H_{\alpha9})$. **GOAL 9**

T. With the result of step Q, R and rule R_2 we get $V \mid\equiv TA \mid\equiv (H_{\alpha7}, H_{\alpha8}, H_{\alpha9})$. **GOAL 10**

U. With the result of steps T, belief 5 and rule R_3 we get $V \mid\equiv (H_{\alpha7}, H_{\alpha8}, H_{\alpha9})$.

V. With the result of step T and rule R_5 we get $V \mid\equiv Sequence_{Number_V}$. **GOAL 11**

W. Using the result of the steps S, T and rule R_8 we get $V \mid\equiv TA \xleftrightarrow{Sequence_{Number_V}} V$. **GOAL 12**

X. Using the result of the steps S, T and rule R_8 we get $V \mid\equiv TA \xleftrightarrow{Key_{Session}} V$. **GOAL 13**

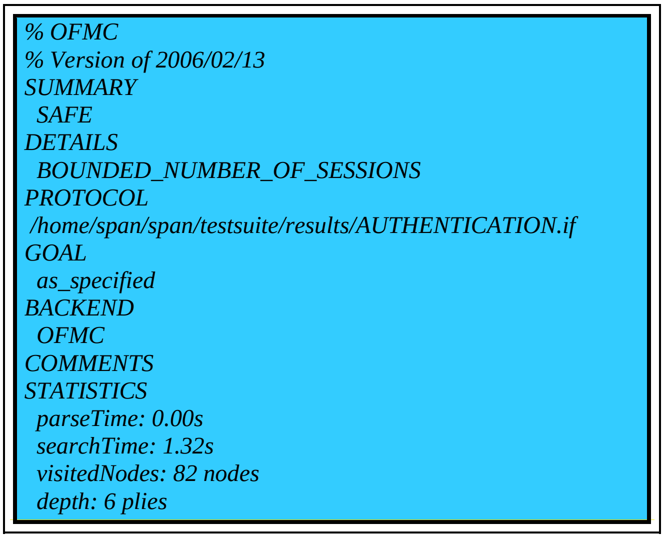
Y. Using the result X, belief 5 and rule R_2 (similar to [22]) we get $V \mid\equiv TA \mid\equiv TA \xleftrightarrow{Key_{Session}} V$. **GOAL 14**

Z. Using the result P, belief 4 and rule R_2 (similar to [22]) we get $TA \mid\equiv V \mid\equiv TA \xleftrightarrow{Key_{Session}} V$. **GOAL 15**

B. Formal Verification: Authentication proof using AVISPA

AVISPA is a simulation tool for security verification which justifies the security of the authentication schemes. It is also a software which is of type role related, where each participants acts as a role [3, 22]. The AVISPA software is first written in HLPSL which is a High Level Protocol Specification Language which is then translated into IF which is an Intermediate Format lower level language and it can directly be read by AVISPA's back end. The table II depicts the result in OFMC backend.

Table II. Result in OFMC Backend



```
% OFMC
% Version of 2006/02/13
SUMMARY
  SAFE
DETAILS
  BOUNDED_NUMBER_OF_SESSIONS
PROTOCOL
  /home/span/span/testsuite/results/AUTHENTICATION.if
GOAL
  as_specified
BACKEND
  OFMC
COMMENTS
STATISTICS
  parseTime: 0.00s
  searchTime: 1.32s
  visitedNodes: 82 nodes
  depth: 6 plies
```

C. Formal Analysis: Simulation in Real-or-Random oracle model

In our paper we make use of Real-Or-Random oracle model to prove the semantic security of the scheme [13, 50, 57]. A design of a security model is very beneficial as it helps us in delivering robust security goals. The Real-Or-Random oracle model has following components which are of interest:

(1) **Participants:** The participants are the Vehicles (V) and the Trusted Authority (TA) where the communication is taking place between one of the subset of V ($V_i \subseteq V$) and a TA, each of the vehicle from the subset must communicate keeping the abilities of adversary in check. Consider $\prod_{V_i}^{u}, \prod_{TA}^{v}$ be the two instances u, v of the subset vehicle u and a Trusted Authority v. The above are defined as an oracle.

(2) **Accepted State:** A given instance say $\prod^u$ will be in an accepted state if it has received the last expected message from the protocol. The session IDs are defined by concatenating every exchanged message (sent and received) by the given instance $\prod^u$ in the right order.

(3) **Partnering:** Two instances $\prod^u, \prod^v$ are said to be partners of each other if they satisfy all of the below conditions in parallel. (1) The Two instances $\prod^u, \prod^v$ are in the Accepted State, (2) The Two instances $\prod^u, \prod^v$ are in possession of the same session ID and are mutually authenticated and (3) The Two instances $\prod^u, \prod^v$ are mutual partners of each other.

(4) **Adversary:** The adversary has various abilities which it might use to disrupt the scheme and perform wicked attacks. The adversary can play both active and passive attacks; the passive attack is where the adversary is observing the communication flow across two instances $\prod^u, \prod^v$ while active attack is where the adversary takes an active part in the communication. The adversary can perform various attacks such as modification, forging, masquerade, etc. The following are the queries which define various adversarial abilities. (1) $Exec(\prod^u, \prod^v)$: It is a passive attack where the adversary is able to capture various messages which are passed through the channel and the goal of the adversary here is to attain useful information to attain useful information from the communication, thus it can be viewed as a basic eavesdropping. (2) $SND(\prod^u, Msg_\alpha)$: It is an active attack where an adversary tries to impersonate an honest vehicle $V_i \subseteq V$ by sending fraudulent message Msg_α and it receives a reply/response from TA, in this attack an adversary usually takes a message from the previously exchanged sessions and tries to tamper with the message or replay (or suppress replay) the same message again, in hope it gets accepted. (3) $Corrupt_{OBU}(\prod^u)$: It is an attack where the adversary is able to attain all the credentials from the On-Board Unit (OBU), the said will be used by an adversary to gain secret information about either the password or the keys so that the adversary can impersonate the vehicle $V_i \subseteq V$. (4) $Test(\prod^u)$: The said query is used to model the sematic security of the session key, the above query executed for a session then the said query returns the session key to the adversary or a random string length equivalent to the size of the actual key, the output is based on the initial assumption of bit 'b' made by an adversary. For instance, if the bit 'b' is guessed correctly then the said query returns the session key. If the previous assumption is wrong then a random string of same length is returned; thus we can say that the sematic security of the session key depends on the adversary ability to guess the bit 'b'.

Theorem1. *In the proposed scheme, the advantage of an adversary $Adv_p^{KAVANET}$ for breaking the semantic security of session key can be concluded as $Adv_p^{KAVANET} \le \frac{q_h^2}{|H|} + \frac{2q_{send}}{|D|}$, where $q_h, q_{send}, |H|, |D|$ are the number of hash queries, number of send queries, space of secure one-way hash function and the size of password dictionary respectively.*

Proof. For proving the semantic security of the session key, we distribute the proof across different games and through succession of each game the ability of the adversary is increasing. The advantage $Adv_p^{KAVANET}$ that the adversary A, throughout these games can attain, is taken by finding the differences between them. We define a game as $G_i^{SKVANET}$ which means that the ith game is being played by the adversary A to break the semantic security of the proposed scheme and attain the group session key.

Game0: $G_0^{SKVANET}$ is a real attack in the random oracle model, where an adversary A guesses the bit 'b' before beginning of the games. According to definition we have

$$Adv_p^{KAVANET} = |2\Pr[succ_0] - 1| \qquad \textbf{(1)}$$

Where $succ$ defines an event where the adversary A is successful in breaking the semantic security of the proposed key agreement protocol.

Game1: $G_1^{SKVANET}$ is a passive attack where the adversary is eavesdropping on the public channel where the communication is taking place between the vehicle and the trusted authority. Thus in this game an adversary launches $Exec()$ query and based on the attained information from the said communication, the adversary executes $Test()$ query and then decides whether the output is the actual session key or a random number. It can be easily concluded that even when above two queries are provoked, the adversary will not be successful in breaking the semantic security of the proposed scheme as $Key_{New} = [(Key_{Session}) \bmod(h_3^c) \cdot (f) \bmod(h_2^b)] \bmod(h_1^a)$ and the information about the Key_{New} is not present on any of the exchanged message

$\{Mssg_1, Mssg_3, Mssg_4, Mssg_6\}$ whereas the information about the Modulus crumbs ($\{Nonce_{v1}, Nonce_{TA}, DI_v, Key_{v1}, Sequence_{Number_v}\}$) present on the message $\{Mssg_1, Mssg_4\}$ will be enough for an adversary to generate the values of moduli being $h_1 : H(Nonce_{v1} \| TS_v \| DI_v \| Key_{v1})$, $h_2 : H(PublicIdentifier_v \| Key_{v1} \| TS_v \| Nonce_{v1} \| Sequence_{Number_v})$, $h_3 : H(Key_{v1} \| Nonce_{v1} \| DI_v \| Sequence_{Number_v} \| TS_v \| PublicIdentifier_v)$. But the adversary will not be able to do so because the adversary needs to find multiple collisions to converge and attain the session modulus values. In the above game, if the key space is large then no information can be archived by the adversary using the above game, thus we conclude that

$$|\Pr[Succ_0] - \Pr[Succ_1]| = 0 \qquad \textbf{(2)}$$

Game2: $G_2^{SKVANET}$ is an active attack, where an adversary A is taking an active part in the communication, it is where an adversary has an access to $SND()$ query, thus an adversary is sending a query and is attaining a response in return from the TA. In this game the adversary procures an access to both $SND()$ query and an oracle H_{Oracle}, thus an adversary A queries hash oracle to find a successful collision to produce fraudulent messages which will be accepted and in turn result in successful authentication of the adversary A. In the proposed scheme, the message $Mssg_1$ comprising $\{H_{\alpha1}, H_{\alpha2}, H_{\alpha3}\}$ (where $H_{\alpha1} = H(TS_v \| ID_v \| Key_{v1} \| TA_1 \| DI_v) \oplus Nonce_{v1}$, $H_{\alpha2} = H(Nonce_{v1} \| TS_v \| Key_{v1} \| TA_4 \| Sequence_{Number_v})$, $H_{\alpha3} = H(Nonce_{v1} \| Key_{v1} \| ID_v \| TS_v \| TA_1 \| DI_v \| TA_4) \oplus Sequence_{Number_v}$) is bounded by a public identifier (which is updated after every session) and a timestamp, same holds for $Mssg_3$ which comprises of $\{H_{\alpha7}, H_{\alpha8}, H_{\alpha9}\}$ (where the values of $H_{\alpha7}$, $H_{\alpha8}$, $H_{\alpha9}$ are $H_{\alpha7} = a \| PublicIdentifier_v \| b \| TS_{TA} \| h(b \| PublicIdentifier_v \| Nonce_{v1} \| a \| Key_{v1} \| ID_v \| Nonce_{TA1}) \oplus NEWSequence_{Number_v}$, $H_{\alpha8} = h(Nonce_{v1} \| TS_v \| Key_{v1} \| PublicIdentifier_v \| TS_{TA}) \oplus Nonce_{TA1} \oplus Nonce_{v1}$, $H_{\alpha9} = H(a \| Nonce_{v1} \| PublicIdentifier_v \| Key_{v1} \| New_{Sequence_{Number_v}} \| TS_v \| Nonce_{TA1})$). Thus we can conclude that there will be no successful collisions in the $SND()$ query due to presence of fresh timestamp and a unique public identifier for every session. We utilize a conclusion associated with hash function from birthday paradox in order to prove the advantage of an adversary to break the semantic security of the proposed key agreement protocol. According to birthday paradox, the success of finding a collision is $\frac{q_h^2}{2|H|}$. Thus, from the above we have

$$|\Pr[Succ_1] - \Pr[Succ_2]| \le \frac{q_h^2}{2|H|} \qquad \textbf{(3)}$$

Game3: $G_3^{SKVANET}$ is an attempt made by an adversary to derive secret values present on the on-board unit (OBU), so that the adversary can be successful in performing various wicked attacks. In this game, an adversary is given an access to the $Corrupt_{OBU}()$ query, thus the adversary is now in possession of $\{PublicIdentifier_v, TA_1, TA_2, TA_3, TA_4, DI_v, Sequence_{Number_v}, H\}$ but in order to successfully attain the exchanged session key credentials, an adversary must rely on a successful dictionary attack to attain useful information and use the OBU. Thus, we can conclude

$$|\Pr[Succ_2] - \Pr[Succ_3]| \le \frac{q_{send}}{|D|} \qquad \textbf{(4)}$$

Game4: $G_4^{SKVANET}$ is a continuation of $G_3^{SKVANET}$ where final attempt is made, here all of the above queries can be utilized by an adversary A to break the semantic security of the proposed group key agreement protocol to attain the session key. If even now the adversary is unable to break the semantic security and attain the session key then the adversaries attempt to disrupt the scheme lies in a guess. Thus we have

$$\Pr[Succ_3] = \frac{1}{2} \qquad \textbf{(5)}$$

The result of step **(5)** can be utilized to conclude to the desired result delineated earlier. Now, utilizing the traingular inequality we have

$$|\Pr[succ_1] - \Pr[succ_3]| \le \frac{q_h^2}{2|H|} + \frac{q_{send}}{|D|} \qquad \textbf{(6)}$$

Since, the value of $\Pr[Succ_3] = \frac{1}{2}$ (attained from **(5)**) we have $|\Pr[succ_1] - \frac{1}{2}| \le \frac{q_h^2}{2|H|} + \frac{q_{send}}{|D|}$, using this result and the result of **(1)**, **(2)** we attain the following result $2[\Pr[succ_0] - \frac{1}{2}] \le 2[\frac{q_h^2}{2|H|} + \frac{q_{send}}{|D|}]$and we also achieve **(7)**

$$Adv_p^{KAVANET} \le \frac{q_h^2}{|H|} + \frac{2q_{send}}{|D|} \qquad \textbf{(7)}$$

Hence, we conclude that if the given hash space and the relative size of dictionary is large then it is infeasible for an adversary to break the semantic security and attain the session key.

Theorem2. *The proposed scheme conditionally prevents the identity and secret information of the vehicle when the messages are communicated over the network. The advantage the adversary will have in extracting all the useful information in impersonating can be given by* $Adv_A^{attain} \le \frac{3q_h^{\ 2}}{2|H|}$.

Proof. We play another game G_i^{Brk} where the adversary is given access to $Exec()$, $SND()$ queries and a hash oracle H_{Oracle}, thus now the adversary attains $\{Mssg_1, Mssg_3\}$ adhered to which it finds collision in $H_{\alpha 2}$ (the advantage for finding a successful collision using birthday paradox is $Adv_A^{temp_1} \le \frac{q_h^{\ 2}}{2|H|}$) and then xors the value of $Sequence_{Number_v}$ with $H_{\alpha 3}$. Then it finds a collision in $H_{\alpha 3}$ (the advantage in this case is $Adv_A^{temp_2} \le \frac{q_h^{\ 2}}{2|H|}$) adhered to this the adversary must xor the value of attained $Nonce_{v1}$ with H_{α_1} and find collision in it (the advantage here is $Adv_A^{temp_3} \le \frac{q_h^{\ 2}}{2|H|}$), thus as a result the adversary has now successfully attained the subsequent values $\{Nonce_{v1}, ID_v, DI_v, Key_{v1}, Sequence_{Number_v}, TA_1, TA_4\}$. Once the adversary has attained the said values, $\{Nonce_{v1}, ID_v, DI_v, Key_{v1}, Sequence_{Number_v}, TA_1, TA_4\}$ the adversary computes $h_1 : H(Nonce_{v1} \| TS_v \| DI_v \| Key_{v1})$, $h_2 : H(PublicIdentifier_v \| Key_{v1} \| TS_v \| Nonce_{v1} \| Sequence_{Number_v})$, $h_3 : H(Key_{v1} \| Nonce_{v1} \| DI_v \| Sequence_{Number_v} \| TS_v \| PublicIdentifier_v)$ and then captures the message $Mssg_3$ for attaining the values of (a,b) pair (the adversary after attaining the value of these pairs, computes and procures the common session key). Moreover, in case of message $Mssg_3$, the adversary also attains the value of new sequence number $NEWSequence_{Number_v}$. Now, using the birthday paradox we have a total advantage of $3 \cdot [\frac{q_h^{\ 2}}{2|H|}]$, we can thus conclude that by using the queries $Exec()$, $SND()$ (while performing the passive and active attacks) and a hash oracle H_{Oracle} the advantage the adversary A has is $Adv_A^{temp_1} + Adv_A^{temp_2} + Adv_A^{temp_3} = Adv_A^{attain} \le \frac{3q_h^{\ 2}}{2|H|}$.

D. Informal Security Analysis

In this section security analysis of the proposed scheme is presented, the security analysis depicts as of how the proposed scheme is secure against various attacks. This section depicts as of how the proposed work overcomes various vulnerabilities depicted in analysis of [33] and also presents as of how the proposed work is secure against the attacks mentioned in the introduction section.

Proposition 1. *The proposed scheme preserves the identity and the device identifier information.*

Proof. In the proposed scheme the vehicle makes use of an unique random pseudonym $PublicIdentifier_v$ (which changes after every session) to communicate with the TA via RSU_i. The pseudonym initially comprises of $H(Nonce_v \oplus ID_v \oplus Timestamp_v)$ and since the adversary does not possess the knowledge of $\{Nonce_v\ \ , Timestamp_v\}$, thus the adversary cannot guess the value of identity of the vehicle, whereas in the later stages the value $NewPublicIdentifier_v = H(Nonce_{v1} \| PublicIdentifier_v \| TS_{TA} \| Key_{v1} \| DI_v \| Nonce_{TA1} \| TS_v \| Sequence_{Number_v})$ and since there is no presence of identity here, the adversary cannot attain any useful information about identity using the updated pseudonym information. To attain the device identifier and identity information the adversary has to find collisions in the message $Mssg_1, Mssg_3$ comprising $\{H_{\alpha 1}, H_{\alpha 3}, H_{\alpha 7}\}$ but the adversary cannot attain the above credentials due to difficulty in finding collisions in the hash function.

Proposition 2. *The proposed scheme provides traceability*

Proof. In the proposed scheme based on the pseudonym $PublicIdentifier_v$ the trusted authority TA can fetch the true identity credentials which are securely stored. Moreover, since the pseudonym $PublicIdentifier_v$ and authentication message $Mssg_1$ is dynamic and constantly changing it is computationally infeasible for any corrupted entity to derive true user identity credentials. Thus, as a direct consequence only the TA can trace valid entities in the region.

Proposition 3. *The proposed scheme provides un-linkability.*

Proof. In the proposed scheme, the authentication message is sent when vehicle is entering or exiting the region. The OBU_j of the vehicle proceeds by generating authentication message $Mssg_1$ for which OBU_j computes the following $H_{\alpha 1} = H(TS_v \| ID_v \| Key_{v1} \| TA_1 \| DI_v) \oplus Nonce_{v1}$, $H_{\alpha 2} = H(Nonce_{v1} \| TS_v \| Key_{v1} \| TA_4 \| Sequence_{Number_v})$, $H_{\alpha 3} = H(Nonce_{v1} \| Key_{v1} \| ID_v \| TS_v \| TA_1 \| DI_v \| TA_4) \oplus Sequence_{Number_v}$ and sends the message $Mssg_1 = \{PublicIdentifier_v, TS_v, H_{\alpha 1}, H_{\alpha 2}, H_{\alpha 3}\}$ to RSU_i. The RSU_i after validating the message $Mssg_1$, computes $H_{\alpha 4} = H(Nonce_{R0} \| Key_{R1} \| ID_R \| TS_R \| Mssg_1) \oplus Sequence_{Number_R}$, $H_{\alpha 5} = H(DI_R \| Mssg_1 \| TS_R \| Key_{R1}) \oplus Nonce_{R0}$, $H_{\alpha 6} = H(Nonce_{R0} \| Key_{R1} \| ID_R \| TS_R \| DI_R \| Sequence_{Number_R})$ and sends the message

$Mssg_2 = \{PublicIdentifier_R, TS_R,\ \ H_{\alpha 4}, H_{\alpha 5}, H_{\alpha 6}, Mssg_1\}$ to the TA over a public channel. Due to presence of fresh timestamp, random authentication credentials and pseudonym $PublicIdentifier_v$ on every message it is not feasible for an adversary to link two authentication message from the same vehicle.

Proposition 4. *The proposed scheme withstands known key attack.*
Proof. If an adversary knows the current session key ${Key_{Session}}^{t}$ then it is not feasible for an adversary to predict the future non-zero session key ${Key_{Session}}^{t+1}$ as the new session key is generated using the old session key ${Key_{Session}}^{t}$, a new random number $f^{\,t+1}$ and $\{{h_1}^{a}, {h_2}^{b}, {h_3}^{c}\}$ (which could be updated, as there will be elimination of modulus values contributed by the said entity) as $Key_{New} = [({Key_{Session}}^{t}) \bmod({h_3}^{c}) \cdot (f^{\,t+1}) \bmod({h_2}^{b})] \bmod({h_1}^{a})$. Here, the TA generates (a_α, b_α) pairs and distribute it to all the vehicles in the region via $Mssg_3$ or $Mssg_6$ Thus, even if an adversary is able to capture any of the (a_m, b_m) from the distributed $Mssg_3$ or $Mssg_6$, the adversary will not be able to attain the future session key as the adversary does not have any corresponding $\{{h_1}^{\beta}, {h_2}^{\beta}, {h_3}^{\beta}\}$ value to converge on the session key. Thus, the knowledge of current session key will provide sufficient information for an adversary to converge on the future session keys.

Proposition 5. *The proposed scheme withstands offline password guessing attack.*
Proof. It will be infeasible for an adversary to predict the password in the proposed work as both the authentication messages $Mssg_1 = \{PublicIdentifier_v, TS_v, H_{\alpha 1}, H_{\alpha 2}, H_{\alpha 3}\}$ and $Mssg_4 = \{NewPublicIdentifier_v, TS_{v1}, H_{\alpha 9}, H_{\alpha 10}, H_{\alpha 11}\}$ do not possess any password related information on them. Thus as a result, the adversary will not be able to predict authentication related information.

Proposition 6. *The proposed scheme withstands De-synchronization attack*
Proof. In this particular attack, either the pseudonym $PublicIdentifier$, session key $Key_{session}$ or the sequence number $Sequence_{Number_x}$ are different in the memory of TA, V_j . In the proposed scheme, the TA sends an authentication message which mainly comprises of $Mssg_3 = \{PublicIdentifier_v, H_{\alpha 7}, H_{\alpha 8}, H_{\alpha 9}\}$ ($H_{\alpha 7} = a \,\|\, PublicIdentifier_v \,\|\, b \,\|\, TS_{TA} \,\|\, h(b \,\|\, PublicIdentifier_v \,\|\, Nonce_{v1} \,\|\, a \,\|\, Key_{v1} \,\|\, ID_v \,\|\, Nonce_{TA1}) \oplus NEWSequence_{Number_v}$, $H_{\alpha 8} = h(Nonce_{v1} \,\|\, TS_v \,\|\, Key_{v1} \,\|\, PublicIdentifier_v \,\|\, TS_{TA}\ ) \oplus Nonce_{TA1} \oplus Nonce_{v1}$, $H_{\alpha 9} = H(a \,\|\, Nonce_{v1} \,\|\, PublicIdentifier_v \,\|\, Key_{v1} \,\|\, New_{Sequence_{Number_v}} \,\|\, TS_v \,\|\, Nonce_{TA1})$). And if the vehicle exits the region it sends the authentication message $Mssg_4 = \{NewPublicIdentifier_v, TS_{v1}, H_{\alpha 9}, H_{\alpha 10}, H_{\alpha 11}\}$ to the TA where the TA can verify credentials by validating the message i.e. if ${H_{\alpha 9}}^{*} \overset{?}{=} H_{\alpha 9}, {H_{\alpha 10}}^{*} \overset{?}{=} H_{\alpha 10}, {H_{\alpha 11}}^{*} \overset{?}{=} H_{\alpha 11}$ holds, if it does then the TA can be certain that the same group key, new sequence number and random number was exchanged. Thus, as a result the proposed work withstands De-synchronization. Moreover, if the above validating message fails then TA maintains the log of previous sequence number and perform resynchronization by tracking the vehicle in the next RSU_i region.

Proposition 7. *The proposed scheme presents mutual authentication.*
Proof. The user sends the authentication message $Mssg_1$ along with its sequence number $Sequence_{Number_v}$, thus as a result, when the TA receives the message it authenticates the user. Moreover, when the message $Mssg_3$ (where $H_{\alpha 7} = a \,\|\, PublicIdentifier_v \,\|\, b \,\|\, TS_{TA} \,\|\, h(b \,\|\, PublicIdentifier_v \,\|\, Nonce_{v1} \,\|\, a \,\|\, Key_{v1} \,\|\, ID_v \,\|\, Nonce_{TA1}) \oplus NEWSequence_{Number_v}$, $H_{\alpha 8} = h(Nonce_{v1} \,\|\, TS_v \,\|\, Key_{v1} \,\|\, PublicIdentifier_v \,\|\, TS_{TA}) \oplus Nonce_{TA1} \oplus Nonce_{v1}, H_{\alpha 9} = H(a \,\|\, Nonce_{v1} \,\|\, PublicIdentifier_v \,\|\, Key_{v1} \,\|\, New_{Sequence_{Number_v}} \,\|\, TS_v \,\|\, Nonce_{TA1})$) is sent by the TA to the OBU_j of the vehicle V_j then the OBU_j validates the attained message and if validation holds, the OBU_j is certain that the communicating entity is legitimate. Thus, as a result the V_j has been validated by TA and TA has been validated by V_j . Same holds when the vehicle exits the given region RSU_i (thus under both the condition there is a presence of mutual authentication).

Proposition 8. *The proposed scheme resists replay and suppress replay attacks.*
Proof. The replay attack can easily be detected by checking if the attained message lies within the threshold ΔT . The above holds for all the messages i.e. $\{Mssg_1, Mssg_2, Mssg_3, Mssg_4, Mssg_5, Mssg_6\}$. In the proposed scheme, the suppress replay attack can be detected as the OBU_j (of the V_j) makes use of a unique session sequence number $Sequence_{Number_v}$ which is updated and communicated by TA (to OBU_j) for every iteration, if the sequence numbers attained from the authentication message $Mssg_1$ (after checking for its freshness) is not existing in the storage unit of TA then the message is fresh but replayed as a result the request gets rejected. In case of message $Mssg_4$, the TA keeps track of the new sequence number

$NEWSequence_{Number_v}$ and if the group exit request is again replayed, the TA can detect it (the $NEWSequence_{Number_v}$ has to be legally used for next group joining request only) as generation of both group joining and exiting schemes are different. Furthermore, in the case of message from RSU_i (for messages $Mssg_2, Mssg_4$), if the sequence number is existing on the storage unit of TA then the message is discarded.

Proposition 9. *The proposed scheme provides perfect forward secrecy.*
Proof. In the proposed scheme, when the vehicle V_j is exiting the region, the TA computes a new session using the old session key ${Key_{Session}}^t$, a new random number f^{t+1} and $\{h_1{}^a, h_2{}^b, h_3{}^c\}$ (which could be updated) being $Key_{New} = [({Key_{Session}}^t) \bmod (h_3{}^c) \cdot (f^{t+1}) \bmod (h_2{}^b)] \bmod (h_1{}^a)$. Once the new session key is generated the TA sends the authentication message $Mssg_6$ (similar to $Mssg_3$) to all the vehicles. The OBU_j exiting vehicle V_j although knows the key ${Key_{Session}}^t$ but the said OBU_j has no knowledge of either f^{t+1} or possess any valid modulus, as a result the OBU_j even if given access to any of (a_β, b_β) pair, will not be able to attain the new session key Key_{New}. Thus, the proposed scheme provides perfect forward secrecy.

Proposition 10. *The proposed scheme provides perfect backward secrecy.*
Proof. In the proposed work, when the vehicle V_j enters the said RSU_i region (comprising of t vehicles), the OBU_j of the vehicle V_j sends the authentication message $Mssg_1$ adhered to which if the authentication is successful then the TA updates the session key being $Key_{New} = [({Key_{Session}}^t) \bmod (h_3{}^c) \cdot (f^{t+1}) \bmod (h_2{}^b)] \bmod (h_1{}^a)$. Thus, once the session key is updated then the TA sends the authentication message $Mssg_3$ to the vehicle V_j, thus by the knowledge of the new session key Key_{New}, the adversary will not be able to attain the previous session key ${Key_{Session}}^t$. Thus, as a result the proposed scheme provides perfect backward secrecy.

Proposition 11. *The proposed scheme withstands impersonation attack.*
Proof. In the proposed work, if an adversary desires to get authenticated by the TA, then the said adversary will have to generate a successful authentication message i.e. $Mssg_1 = \{PublicIdentifier_v, TS_v, H_{\alpha 1}, H_{\alpha 2}, H_{\alpha 3}\}$. Since the adversary does not have any knowledge of $\{TA_1, TA_4, Seqeunce_{Number_v}, Key_{v1}, DI_v, ID_v\}$, thus the adversary cannot generate successful authentication message which will be accepted by TA as legitimate. Moreover, the suppress replay attack will also be detected and the authentication request will be rejected. Hence, The proposed scheme withstands impersonation attack.

Proposition 12. *The proposed scheme withstands modification attack.*
Proof. In the proposed work, if there is modification in the any of the communicated message i.e. $\{Mssg_1, Mssg_2, Mssg_3, Mssg_4, Mssg_5, Mssg_6\}$ then it can be detected as the corresponding message authentication code will not match. Thus, as a result the proposed scheme withstands modification attack.

Proposition 13. *The proposed scheme withstands known session specific temporary information attack.*
Proof. In the proposed work, the following $h_1 : H(Nonce_{v1} \| TS_v \| DI_v \| Key_{v1})$, $h_2 : H(PublicIdentifier_v \| Key_{v1} \| TS_v \| Nonce_{v1} \| Sequence_{Number_v})$, $h_3 : H(Key_{v1} \| Nonce_{v1} \| DI_v \| Sequence_{Number_v} \| TS_v \| PublicIdentifier_v)$. Thus, the knowledge of session specific temporary information such as $\{PublicIdentifier_v, Nonce_{v1}, Sequence_{Number_v}, TS_v\}$ will not be sufficient for an adversary to predict the session key or the new pseudonym. Thus, the adversary will not be able to compute the session key or the new pseudonym as the adversary cannot derive the value of $\{h_1, h_2, h_3\}$ or $H(Nonce_{v1} \| PublicIdentifier_v \| TS_{TA} \| Key_{v1} \| DI_v \| Nonce_{TA1} \| TS_v \| Sequence_{Number_v})$ due to insufficient information. Thus, the proposed work is secure against known session specific temporary information attack.

VIII. Performance Analysis

TABLE III. EXECUTION TIME OF DIFFERENT CRYPTOGRAPHIC OPERATIONS

Cryptographic Operation ↓	Execution Time Operation	Running Time (milliseconds)
T_{sm-ecc}	**Elliptical Curve multiplication**	0.442 ms
T_{pa-ecc}	**Elliptical Curve Addition**	0.0018 ms
T_{sm-bp}	**Scalar Multiplication Bilinear Pairing**	1.709 ms
T_{bp}	**Bilinear Pairing**	4.211 ms
T_{mtp}	**Map-to-Point hash operation**	4.406 ms
T_{pa-bp}	**Point Addition Bilinear Pairing**	0.0071 ms
T_h	**Hash Operation**	0.0001 ms
T_{Exp}	**Modular Exponentiation**	0.522 ms
$T_{E/D}$	**Symmetric Key Encryption/Decryption**	0.1303 ms
T_{Inv}	**Modular Inverse**	0.174 ms

TABLE IV. EXECUTION TIME OF AUTHENTICATION MESSAGE GENERATION AND VERIFICATION PHASE OF DIFFERENT SCHEMES

Schemes ↓	Total Computation Cost	Total Execution Time
[5]	$4T_{E/D}+12T_h$	0.522 ms
[33]	$10T_h$	0.001 ms
[50]	$2T_{E/D}+6T_h$	0.261 ms
[14]	$3T_{Exp}+2T_{Inv}$	1.914 ms
[60]	$24T_h$	0.0024 ms
[61]	$2T_{E/D}+8T_h$	0.261 ms
[62]	$17T_{sm-ecc}+6T_{pa-ecc}+12T_h$	7.526 ms
[63]	$4T_{sm-ecc}+3T_h+1T_{pa-ecc}$	1.770 ms
[64]	$3T_{bp}+2T_{mtp}+T_{sm-ecc}$	21.887 ms
[65]	$12T_{Exp}+8T_h+2T_{inv}$	6.6128 ms
[66]	$4T_{sm-ecc}+T_{bp}+6T_h+4T_{exp}$	8.068 ms
[67]	$3T_{bp}+4T_{sm-ecc}+1T_h$	14.321 ms
Our	$18T_h$	0.0018 ms

The performance of the proposed scheme is evaluated with other relevant work in the context of security features, costs and execution time. The said analysis proves that the proposed work has real-time application in VANET environment. The previous section focused on proving security of the proposed work focusing on both formal and informal security verification and analysis which proved the security of the proposed work against various attacks. This section highlights overall overheads in terms of computation and communication cost, thus providing supporting data as to prove the effectiveness of the scheme.

A. Execution Time of Different Schemes

In this section, the execution cost and time of the proposed work is compared with that of other schemes'. The execution time of different cryptographic operations can be viewed in table III and the overall execution time of different schemes can be viewed in table IV. The execution of different cryptographic operation is taken from [33, 22]. Thus the execution environment can be viewed in [33, 22]. The Table III depicts the overall execution time of different schemes.

In the proposed work the total computation cost is $18T_h$ because when the vehicle V_j owner enters the authentication credentials $\{ID_v, Pwd_v\}$ the OBU_j computes $3T_h$ to extract long term key Key_{v_1} and validate authentication credentials. If the authentication credentials are valid then the OBU_j generates authentication credentials and session modulus which takes computation time of $6T_h$ adhered to which the RSU_i sends the authentication message which results in total computation time of $3T_h$. Once the TA attains the authentication credentials the TA computes $6T_h$ to validate both the requests from OBU_j and RSU_i. Thus, from the above description it is evident that the proposed scheme takes the total computation cost as $18T_h$. The execution cost of different schemes can be viewed in [5, 33, 50, 14, 60, 61, 62, 63, 64, 65, 66, 67] respectively. The total execution time of the different schemes (for authentication message generation and verification phases) can be viewed in table IV and figure 4.

B. Total Communication Overhead

We assume the size of message digest to be 64 bytes, size of timestamp to be 4 bytes, device identifier to be 8 bytes, Public Identifiers to be 16bytes, while all other values are taken to be 64 bytes. For [5] Yang et al. the authentication message is $\{\langle\alpha\rangle_\beta, (ID_{RSU_j})_\alpha, \{Pid_i, r_i, s_i, t_s, msg_i\}_\alpha\}$ where total cost results in 64X1 + 64X1 + 64X4 (message is 128 bytes, r and s occupy 64 bytes each whereas the public identifier is 16 bytes) + 4 = 404 bytes. For PW-CPPA-GKA [33] the authentication message is $\{ID_{r_i}, M_{v_j}, Tr_i, atr_i^j\}$ where $M_{v_j} = \left\langle AID_{v_j}, e_{v_j}, y_{v_j}, T_{v_j}, atv_j \right\rangle$, $atv_j = H(A_{v_j} \| ID_{v_j} \| T_{v_j})$ and $atr_i^j = H(Mv_j \| SKr_i \| Tr_i)$

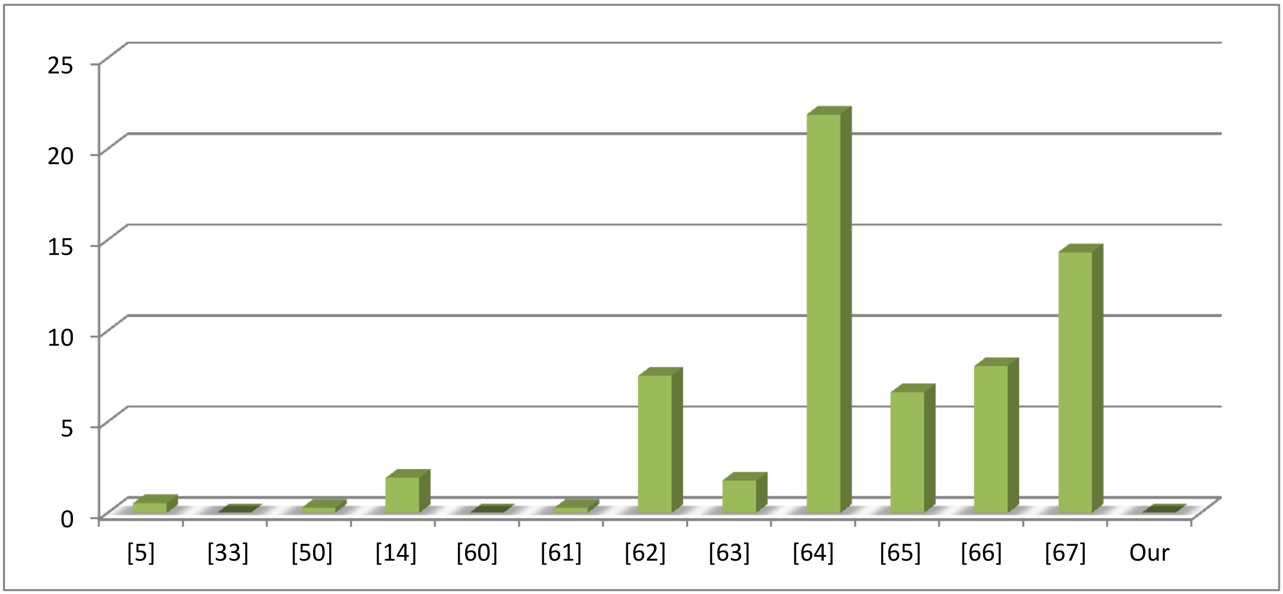


Figure 4. Total Execution Time of Different Schemes

thus the authentication message size is 64X5 + 4 + 4 = 328 bytes. For HCPA-GKA [50] the authentication message is $\{ID_{r_i}, M_{v_j}, T_{r_i}, atr_i\}$ (in $M_{v_j} = \{AID_{v_j}, u_{v_i}, T_{v_i}, atv_j\}$ AID_{v_j} $, u_{v_i}, atv_i$ are 64 bytes each and T_{v_i} is 4 bytes) thus the overall length of authentication message is 64X3 + 4 + 64X2 + 4= 328 bytes. For Paliwal et al. [14] three message exchanges are made to ensure identity exchange of the user ($\{(g^x)^z = (id)^z, (g^{xz})^y, (g^x)^y\}$) thus as a result 64X3 = 192 bytes are communicated for exchanging identity information. For Wazid et al. [60] the messages $\{M_1, M_2, T_1, M_7, T_3\}, \{M_4, M_5, T_2\}$ are exchanged where messages M comprise hash based information thus, the total message size is 64X5 + 4X3 = 332 bytes.

For Zhang et al. [61] the authentication message is $\{T_1, Y_1, Y_2, E_s(n_j) \quad , T_2, Y_3, Y_4, r_j, AID_j\}$ thus, the total communication cost is 4 + 64X2 + 64 + 4 + 64X2 + 64X 2 = 456 bytes. The Zhou et al. [62] scheme the authentication message is $\{ID_{HA}, AID_j, E_j, T_{vj}, h_j, E_{FA}, T_{FA}, \epsilon\}$ 16X2 + 64 + 4 + 64X2 + 4 + 64 = 296 bytes. For Assar et al. [63] the authentication message is $\{PID_i, T_i, m_i, R_i, W_i, s_{i,1}, s_{i,2}\}$ thus, the total communication cost is 64X1 + 4 + 64X5 = 388 bytes. For Pournaghi et al. [64] the authentication message is $\{PID_i^1, PID_i^2\}$, $\{PID_i, \delta_i, M_i, ID_{RSU_j}\}$ thus, the total communication cost is 64X6 = 384 bytes. For Muthumeenakshi et al. [65] the authentication message is $\{id_A, id_B, X_1, X_2\}, \{id_S, Z_1\}, \{id_s, Z_2\}, \{id_A, id_B, Y_1, Y_2\}, \{id_A, id_B$

TABLE V. TOTAL COMMUNICATION COST OF DIFFERENT SCHEMES

Schemes ↓	**Total Communication Cost**
[5]	404 bytes
[33]	328 bytes
[50]	328 bytes
[14]	192 bytes
[60]	332 bytes
[61]	456 bytes
[62]	296 bytes
[63]	388 bytes
[64]	384 bytes
[65]	1408 bytes
[66]	192 bytes
[67]	260 bytes
Our	424 bytes

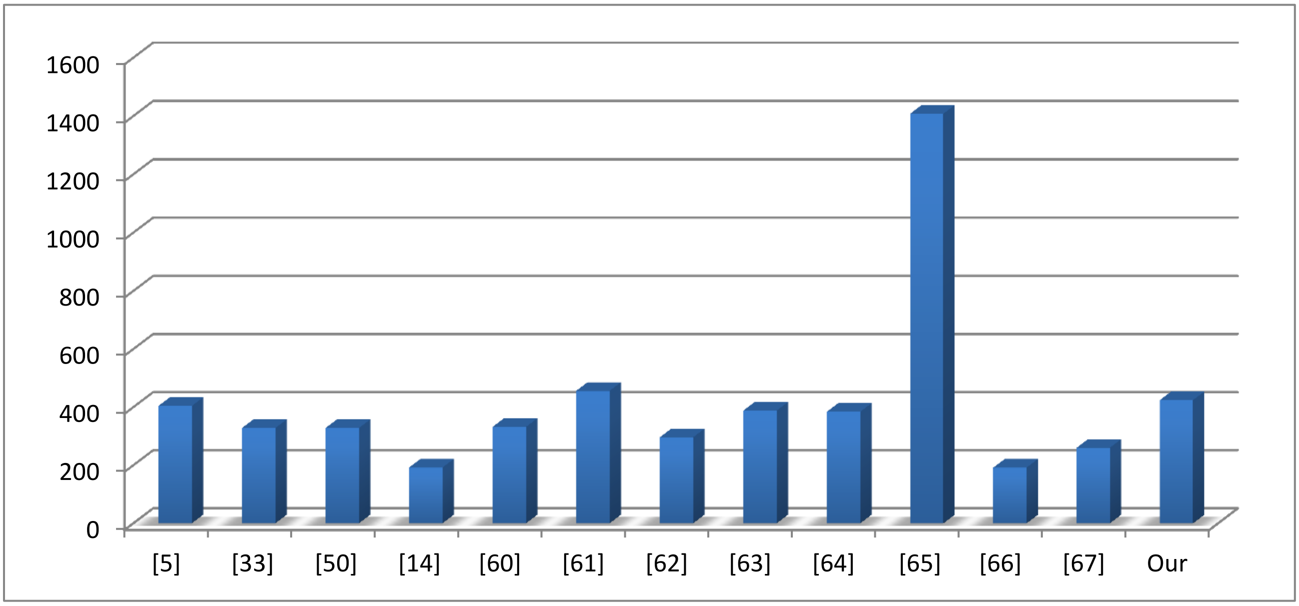


Figure 5. Total Communication Cost

$,Y_1,Y_2\}$ $,\{\alpha_A,\beta_A\},\{\alpha_B,\beta_B\},\{\gamma_A,\beta_B\},\{\gamma_B,\beta_A\}$ thus, the total cost of communication is 64X4 + 64X2 + 64X2 + 64X4 + 64X4 +64X2 +64 X2 +64X2 = 1408 bytes. For Balaji et al.[66] the authentication message is $\{del_a, input_{user}, key\}$ which results in 64X3 = 192 bytes. For Zhong et al. [67] the authentication message is $\{PID_i, m_i, vpk_i, t_i, \sigma_i\}$ where t_i is the timestamp and thus the overcall communication overheads becomes 64X3 + 4 + 64X1 = 260 bytes. In the proposed scheme the authentication message is $Mssg_2 = \{PublicIdentifier_R, TS_R, H_{\alpha 4}, H_{\alpha 5}, H_{\alpha 6}, Mssg_1\}$ Where $H_{\alpha 4} = H(Nonce_{R0} \| Key_{R1} \| ID_R \| TS_R \| Mssg_1)$ $\oplus Sequence_{Number_R}$, $H_{\alpha 5} = H(DI_R \| Mssg_1 \| TS_R$ $\| Key_{R1}) \oplus Nonce_{R0}$, $H_{\alpha 6} = H(Nonce_{R0} \| Key_{R1} \| ID_R \| TS_R$ $\| DI_R \| Sequence_{Number_R})$ and $Mssg_1 = \{PublicIdentifier_v$ $,TS_v, H_{\alpha 1}, H_{\alpha 2}, H_{\alpha 3}\}$ which further comprises of $H_{\alpha 1} = H(TS_v \| ID_v \| Key_{v1} \| TA_1 \| DI_v) \oplus Nonce_{v1}$, $H_{\alpha 2}$ $= H(Nonce_{v1} \| TS_v \| Key_{v1} \| TA_4 \| Sequence_{Number_v})$ and $H_{\alpha 3} = H(Nonce_{v1} \| Key_{v1} \| ID_v \| TS_v \| TA_1 \| DI_v \| TA_4)$ $\oplus Sequence_{Number_v}$. Thus, the total communication cost is 16 + 4 + 64X3 + (16 + 4 + 64X3) = 424 bytes. Table V and Figure 5 depicts the total communication cost. Although, the communication cost is slightly higher than the cited prior arts but it can be justified as the proposed work possess more features (Table VI depicts various security features) than others and is faster than many other schemes. Thus, the proposed work is much more suitable for VANET environment than the existing lightweight authentication scheme.

C. Security Features

In this section we take up features provided by the proposed scheme and make tangible comparison with the other relevant scheme. Table VI. Compares the various security features. "YES" implies that the proposed scheme can withstand the said attack or it possesses the said feature. While "NO" states that the said work cannot withstand the given attack or it doesn't possess the given feature.

IX. Conclusion

In this paper we presented a lightweight authentication and key agreement protocol suitable for VANET environment. The proposed work makes use of dynamic session modulus contributed by each entity to converge on the given session key, the said modulus set is dynamic itself as the vehicles are exiting and entering the region. Moreover, we present a secure way of updating the pseudonyms of the vehicle while ensuring its privacy. The authentication message and pseudonym update phases are in such a fashion that knowledge of identity or the authentication message of the vehicle will not be sufficient for an adversary to trace the vehicle. Security of the proposed work is witnessed with the help of AVISPA, BAN logic and simulation in Real-or-Random oracle model. Moreover, the security is of the scheme is further enhanced by making use of different sequence numbers during vehicle entry and exit phases. The security analysis and simulation results convey that the proposed scheme is secure and is highly capable of application in VANET environment, the above is not the case with many referenced articles. Moreover, the proposed work proved that it is secure against various attacks and also overcomes the vulnerabilities presented in PW-CPPA-GKA scheme.

TABLE VI. COMPARISON OF VARIOUS SECURITY FEATURES

Features ↓	[5]	[33]	[50]	[14]	[60]	[61]	[62]	[63]	[64]	[65]	[66]	[67]	Our
Password Change Phase	NO	YES	NO	NO	YES	YES	NO	NO	NO	NO	NO	NO	YES
KSSTI Attack	NO	NO	YES	NO	YES	NO	YES	NO	YES	NO	YES	NO	YES
Impersonation Attack	YES	YES	YES	NO	YES	YES	YES	YES	YES	NO	YES	YES	YES
Clock Synchronization Issue	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	YES
Un-Linkability	YES	NO	NO	NO	YES	YES	YES	NO	YES	NO	NO	YES	YES
Known Key Attack	NO	NO	NO	NO	YES	YES	YES	YES	NO	YES	YES	YES	YES
Suppress Replay Attack	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	NO	YES
Mutual Authentication	YES	NO	YES	NO	YES	YES	YES	YES	YES	YES	YES	YES	YES
De-Synchronization Attack	YES	NO	YES	NO	YES	NO	YES	YES	YES	YES	YES	YES	YES
Modification Attack	YES	YES	YES	NO	YES	YES	YES	YES	YES	YES	YES	YES	YES
Traceability	YES	NO	NO	NO	YES	YES	YES	YES	YES	NO	NO	YES	YES
Computational DOS Attack	YES	YES	YES	YES	YES	YES	NO	YES	NO	NO	NO	NO	YES
Dynamic Session Modulus	NO	NO	NO	YES	NO	NO	NO	NO	NO	NO	NO	NO	YES
Suitable for VANETS	YES	NO	NO	NO	YES	YES	YES	YES	NO	NO	NO	NO	YES

YES : Secure, *NO* : Insecure

X. References

[1] Adrian, D., Bhargavan, K., Durumeric, Z., Gaudry, P., Green, M., Halderman, J.A., Heninger, N., Springall, D., Thomé, E., Valenta, L. and VanderSloot, B., 2015, October. Imperfect forward secrecy: How Diffie-Hellman fails in practice. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security* (pp. 5-17). ACM.

[2] Valenta, L., Cohney, S., Liao, A., Fried, J., Bodduluri, S. and Heninger, N., 2016, February. Factoring as a service. In International Conference on Financial Cryptography and Data Security (pp. 321-338). Springer, Berlin, Heidelberg.

[3] Viganò, L., 2006. Automated security protocol analysis with the AVISPA tool. Electronic Notes in Theoretical Computer Science, 155, pp.61-86.

[4] Shao, J., Lin, X., Lu, R. and Zuo, C., 2016. A threshold anonymous authentication protocol for VANETs. IEEE Transactions on vehicular technology, 65(3), pp.1711-1720.

[5] Yang, X., Yi, X., Khalil, I., Zeng, Y., Huang, X., Nepal, S., Yang, X. and Cui, H., 2019. A lightweight authentication scheme for vehicular ad hoc networks based on MSR. Vehicular communications, 15, pp.16-27.

[6] Sharma, G. and Kalra, S., 2018. A lightweight multi-factor secure smart card based remote user authentication scheme for cloud-IoT applications. *Journal of information security and applications*, *42*, pp.95-106.

[7] Lo, N.W. and Tsai, J.L., 2016. An efficient conditional privacy-preserving authentication scheme for vehicular sensor networks without pairings. *IEEE Transactions on Intelligent Transportation Systems*, *17*(5), pp.1319-1328.

[8] He, D., Zeadally, S., Xu, B. and Huang, X., 2015. An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks. IEEE Transactions on Information Forensics and Security, 10(12), pp.2681-2691.

[9] Bayat, M., Barmshoory, M., Rahimi, M. and Aref, M.R., 2015. A secure authentication scheme for VANETs with batch verification. *Wireless networks*, *21*(5), pp.1733-1743.

[10] Jianhong, Z., Min, X. and Liying, L., 2014. On the security of a secure batch verification with group testing for VANET. *International Journal of Network Security*, *16*(5), pp.351-358.

[11] Shim, K.A., 2012. An Efficient Conditional Privacy-Preserving Authentication Scheme for Vehicular Sensor Networks. *IEEE Transactions on Vehicular Technology*, *61*(4), pp.1874-1883.

[12] Liu, Y., Wang, L. and Chen, H.H., 2015. Message authentication using proxy vehicles in vehicular ad hoc networks. *IEEE Transactions on Vehicular Technology*, *64*(8), pp.3697-3710.

[13] Shen, J., Zhou, T., Wei, F., Sun, X. and Xiang, Y., 2018. Privacy-preserving and lightweight key agreement protocol for V2G in the social internet of things. *IEEE Internet of Things Journal*, *5*(4), pp.2526-2536.

[14] Paliwal, S. and Kumar, C.A., 2017, December. A Novel Multi-party Key Exchange Protocol. In *International Conference on Intelligent Systems Design and Applications* (pp. 597-607). Springer, Cham.

[15] Gope, P. and Hwang, T., 2015. A realistic lightweight authentication protocol preserving strong anonymity for securing RFID system. *computers & security*, *55*, pp.271-280.

[16] SYMMETRIC KEY ENCRYPTION USING PRIVATE MODULO OPERATIONS. Indian Patent Application Number 201641034920.
http://ipindiaservices.gov.in/PublicSearch/PublicationSearch/ApplicationStatus. Last Accessed (Jun 2018).

[17] Copyright Granted Software license. Swapnil Paliwal. http://copyright.gov.in/Documents/ERegister/E-Register_June_2018.pdf,
https://drive.google.com/drive/folders/1v72uUz68Q_C-L-YJ2mdDt1CMkhjb6EKW. Last Accessed (Mar 2019).

[18] Lv, C., Ma, M., Li, H., Ma, J. and Zhang, Y., 2013. An novel three-party authenticated key exchange protocol using one-time key. *Journal of Network and Computer Applications*, *36*(1), pp.498-503.

[19] S. Paliwal, 2017. New secure and reliable polygraphic cryptosystem. ACCENTS Transactions on Information Security, Vol 2(8) .DOI: http://dx.doi.org/10.19101/TIS.2017.28002

[20] Burrows, M., Abadi, M. and Needham, R.M., 1989. A logic of authentication. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, *426*(1871), pp.233-271.

[21] Wessels, J. and BV, C.F., 2001. Application of BAN-logic. *CMG FINANCE BV*, *19*, pp.1-23.

[22] Ali, R., Pal, A.K., Kumari, S., Karuppiah, M. and Conti, M., 2018. A secure user authentication and key-agreement scheme using wireless sensor networks for agriculture monitoring. *Future Generation Computer Systems*, *84*, pp.200-215.

[23] Canetti, R. and Krawczyk, H., 2001, May. Analysis of key-exchange protocols and their use for building secure channels. In *International Conference on the Theory and Applications of Cryptographic Techniques* (pp. 453-474). Springer, Berlin, Heidelberg.

[24] William Stallings. Cryptography and Network Security: principles and practice 5th Edition. Prentice Hall Press Upper Saddle River, NJ, USA ©2010. ISBN:0136097049 9780136097044.

[25] Ranjan, P. and Ahirwar, K.K., 2011, January. Comparative study of vanet and manet routing protocols. In *Proc. of the International Conference on Advanced Computing and Communication Technologies (ACCT 2011)* (pp. 517-523).

[26] Chang, C.Y., Yen, H.C. and Deng, D.J., 2016. V2V QoS guaranteed channel access in IEEE 802.11 p VANETs. *IEEE Transactions on Dependable and Secure Computing*, *13*(1), pp.5-17.

[27] Ad Hoc Mobile Wireless Networks: Protocols And Systems. Publisher Pearson Education India, 2007 ISBN 8131715108, 9788131715109.

[28] DePerry, D., Ritter, T. and Rahimi, A., 2013. Traffic Interception & Remote Mobile Phone Cloning with a Compromised CDMA Femtocell. *DEF CON*, *21*(201), p.3.

[29] Introduction to Engineering Mathematics Volume-III. H K Dass, Rajnish Verma & Dr. Rama Verma. S. Chand Publishing. ISBN: 9789352834037.

[30] Sánchez-García, J., García-Campos, J.M., Reina, D.G., Toral, S.L. and Barrero, F., 2016. On-siteDriverID: A secure authentication scheme based on Spanish eID cards for vehicular ad hoc networks. *future generation computer systems*, *64*, pp.50-60.

[31] $15-b investments in smart cities. https://www.thehindubusinessline.com/economy/15b-investments-in-smart-cities/article9867761.ece. Last Accessed (April 2019).

[32] UPI 2.0. https://www.bhimupi.org.in/upi2. Last Accessed (April 2019).

[33] Islam, S.H., Obaidat, M.S., Vijayakumar, P., Abdulhay, E., Li, F. and Reddy, M.K.C., 2018. A robust and efficient password-based conditional privacy preserving authentication and group-key agreement protocol for VANETs. *Future Generation Computer Systems*, *84*, pp.216-227.

[34] Lu, R., Lin, X., Zhu, H., Ho, P.H. and Shen, X., 2008, April. ECPP: Efficient conditional privacy preservation protocol for secure vehicular communications. In *IEEE INFOCOM 2008-The 27th Conference on Computer Communications* (pp. 1229-1237). IEEE.

[35] Kevin Laracey .Mobile phone payment processing methods and systems. https://patents.google.com/patent/US8380177B2/en. Last Accessed (Feb 2019).

[36] Xie, Y., Wu, L., Shen, J. and Alelaiwi, A., 2017. EIAS-CP: new efficient identity-based authentication scheme with conditional privacy-preserving for VANETs. *Telecommunication Systems*, *65*(2), pp.229-240.

[37] Hasrouny, H., Samhat, A.E., Bassil, C. and Laouiti, A., 2017. VANet security challenges and solutions: A survey. *Vehicular Communications*, *7*, pp.7-20.

[38] Raw, R.S., Kumar, M. and Singh, N., 2013. Security challenges, issues and their solutions for VANET. *International journal of network security & its applications*, *5*(5), p.95.

[39] Adrian, D., Bhargavan, K., Durumeric, Z., Gaudry, P., Green, M., Halderman, J.A., Heninger, N., Springall, D., Thomé, E., Valenta, L. and VanderSloot, B., 2015, October. Imperfect forward secrecy: How Diffie-Hellman fails in practice. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security* (pp. 5-17). ACM.

[40] Teepe, W., 2009. On BAN logic and hash functions or: how an unjustified inference rule causes problems. *Autonomous Agents and Multi-Agent Systems*, *19*(1), pp.76-88.

[41] Turkanović, M., Brumen, B. and Hölbl, M., 2014. A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the Internet of Things notion. *Ad Hoc Networks*, *20*, pp.96-112.

[42] Sucasas, V., Mantas, G., Saghezchi, F.B., Radwan, A. and Rodriguez, J., 2016. An autonomous privacy-preserving authentication scheme for intelligent transportation systems. *Computers & Security*, *60*, pp.193-205.

[43] Odelu, V., Saha, S., Prasath, R., Sadineni, L., Conti, M. and Jo, M., 2019. Efficient Privacy Preserving Device Authentication in WBANs for Industrial e-Health Applications. *Computers & Security*.

[44] Ali, I., Hassan, A. and Li, F., 2019. Authentication and privacy schemes for vehicular ad hoc networks (VANETs): A survey. Vehicular Communications.

[45] Raya, M. and Hubaux, J.P., 2007. Securing vehicular ad hoc networks. Journal of computer security, 15(1), pp.39-68.

[46] Lu, R., Lin, X., Zhu, H., Ho, P.H. and Shen, X., 2008, April. ECPP: Efficient conditional privacy preservation protocol for secure vehicular communications. In IEEE INFOCOM 2008-The 27th Conference on Computer Communications (pp. 1229-1237). IEEE.

[47] J.K. Liu, T.H. Yuen, M.H. Au, W. Susilo, Improvements on an authentication scheme for vehicular sensor networks, Expert Syst. Appl. 41 (5) (2014) 2559–2564.

[48] C.-C. Lee and Y. Lai, Toward a secure batch verification with group testing for VANET, Wireless Networks, vol. 19, no. 6, pp. 1441–1449, 2013.

[49] Zhong, H., Huang, B., Cui, J., Xu, Y. and Liu, L., 2018. Conditional privacy-preserving authentication using registration list in vehicular ad hoc networks. IEEE Access, 6, pp.2241-2250.

[50] Cui, J., Tao, X., Zhang, J., Xu, Y. and Zhong, H., 2018. HCPA-GKA: A hash function-based conditional privacy-preserving authentication and group-key agreement scheme for VANETs. Vehicular communications, 14, pp.15-25.

[51] Identity crimes. Last accessed (April 2019).

[52] IP Address Spoofing. Last accessed (April 2019).

[53] Session key establishment protocols. Last accessed (April 2019).

[54] TOMTOM TRAFFIC INDEX MEASURING CONGEST-ION WORLDWIDE. Last accessed (April 2019).

[55] Top 10 Smart City Trends for 2018. Last accessed (April 2019).

[56] Xiong Li, Jiangwei Niu, Saru Kumari, Fan Wu, Arun Kumar Sangaiah and Kim-Kwang Raymond Choo, A Three-factor Anonymous Authentication Scheme for Wireless Sensor Networks in Internet of Things Environments, Journal of Network and Computer Applications,
http://dx.doi.org/10.1016/j.jnca.2017.07.001.

[57] J. Shen, D. Liu, Q. Liu, X. Sun, and Y. Zhang, "Secure authentication in cloud big data with hierarchical attribute authorization structure," IEEE Transactions on Big Data, 2017, doi: 10.1109/TBDATA.2017.2705048.

[58] Xu, D., Chen, J. & Liu, Q. J Ambient Intell Human Comput (2019) 10: 611. https://doi.org/10.1007/s12652-018-0710-x.

[59] Islam, S.H. and Biswas, G.P., 2012. An improved pairing-free identity-based authenticated key agreement protocol based on ECC. Procedia Engineering, 30, pp.499-507.

[60] Wazid, M., Das, A.K., Kumar, N., Odelu, V., Reddy, A.G., Park, K. and Park, Y., 2017. Design of lightweight authentication and key agreement protocol for vehicular ad hoc networks. IEEE Access, 5, pp.14966-14980.

[61] Zhang, Yuxia, and Fengtong Wen. "A Lightweight Secure and Efficient Authentication and Key Agreement Protocol for VANET." IOP Conference Series: Earth and Environmental Science. Vol. 234. No. 1. IOP Publishing, 2019.

[62] Zhou, Y., Long, X., Chen, L. and Yang, Z., 2019. Conditional privacy-preserving authentication and key agreement scheme for roaming services in VANETs. Journal of Information Security and Applications, 47, pp.295-301.

[63] Asaar, M.R., Salmasizadeh, M., Susilo, W. and Majidi, A., 2018. A secure and efficient authentication technique for vehicular ad-hoc networks. IEEE Transactions on Vehicular Technology, 67(6), pp.5409-5423.

[64] Pournaghi, S.M., Zahednejad, B., Bayat, M. and Farjami, Y., 2018. NECPPA: A novel and efficient conditional privacy-preserving authentication scheme for VANET. Computer Networks, 134, pp.78-92.

[65] Muthumeenakshi, R., Reshmi, T.R. and Murugan, K., 2017. Extended 3PAKE authentication scheme for value-added services in VANETs. Computers & Electrical Engineering, 59, pp.27-38.

[66] Balaji, N.A., Sukumar, R. and Parvathy, M., 2019. Enhanced dual authentication and key management scheme for data authentication in vehicular ad hoc network. Computers & Electrical Engineering, 76, pp.94-110.

[67] Zhong, H., Han, S., Cui, J., Zhang, J. and Xu, Y., 2019. Privacy-preserving authentication scheme with full aggregation in VANET. Information Sciences, 476, pp.211-221.

[68] Lee, C.C., Lai, Y.M. and Cheng, P.J., 2016. An efficient multiple session key establishment scheme for VANET group integration. IEEE Intelligent Systems, 31(6), pp.35-43.

[69] Gong, L., 1992. A security risk of depending on synchronized clocks. Operating Systems Review, 26(1), pp.49-53.

[70] Coulouris, G.F., Dollimore, J. and Kindberg, T., 2005. Distributed systems: concepts and design. pearson education.