An Improved Lightweight Identity Authentication Protocol for VANET

Peng Wang[1], Yining Liu[1,2], Songzhan Lv[1]

[1] School of Information and Communication, Guilin University of Electronic Technology, China
[2] Hubei Key Laboratory of Transportation Internet of Things, Wuhan University of Technology, China
glietwp@guet.edu.cn, ynliu@guet.edu.cn, yingyu8ji@gmail.com

Abstract

Recently, Li et al. proposed a lightweight identity authentication protocol (LIAP), in which a unique authentication sequence was firstly shared by a vehicle and the RSUs, then a dynamic secret session process (DSSP) was implemented to achieve the mutual authentication. However, Zhou et al. proved that LIAP was vulnerable against the location privacy tracking attack and the parallel session attack, and proposed an improved protocol. In this paper, Li's scheme and Zhou's scheme are analyzed to be impractical. Moreover, they are vulnerable against the impersonation attacks when some RSUs are compromised by the adversary. Accordingly, an improved version of LIAP is proposed, in which an asymmetrical dynamic secret session process (ADSSP) method is used to replace the DSSP, and a novel distribution model of the authentication sequences is implemented to solve the problem of feasibility. Especially, the security of the proposed protocol is enhanced greatly to resist the impersonation attack. Security and performance analysis shows that the improved version not only resists the known various attacks, but also is more efficient and practical.

Keywords: VANET, Location privacy tracking, Handover authentication, Authentication sequence, Parallel session attack

1 Introduction

Owing to the rapid development of the artificial intelligence and wireless communication technologies, vehicular ad hoc network (VANET) has become a very popular research area in recent years, which can significantly improve traffic safety and efficiency [1]. VANET is considered to play an important role in such fields as traffic management, collision warning, vehicle navigation etc. in the future. Typically, VANET consists of three parties: a large number of vehicles that use onboard units (OBUs) to provide wireless communication; a number of roadside units (RSU) that connect to the wired network and are the access points of OBUs; an authentication server (AS) that is the control center of VANET. Accordingly, there are two basic communication models in VANET, namely vehicle to infrastructure (V2I) and vehicle to vehicle (V2V). In the V2I model, a vehicle accesses RSUs to obtain the infrastructural wired network service. Since a vehicle frequently moves from one RSU's range area to another's area, the efficient handover authentication should be executed between the vehicle and the RSUs to maintain continuous communications. In practice, to design a feasible protocol, on one hand, the security requirements should be guaranteed, such as identity authentication, user anonymity and non-traceability, conditional information privacy protection, attack resistance, etc. [2-6]. On the other hand, the efficiency requirements such as getting a short handover authentication delay are also necessary [7-8].

To protect the conditional information privacy of the vehicles, Sun et al. [9] proposed a scheme using a large number of pseudonymous certificates that are pre-stored in the vehicles by a trusted authority. In Sun's scheme, whenever a vehicle performs identity authentication with other vehicle or RSU, a unique pseudonymous certificate is used and discarded after a fixed period of time. Only the trusted authority can reveal the relationship between the pseudonymous certificates and the real ID of a vehicle. Thus, the identity and the position privacy of the vehicle is protected. Meanwhile, those vehicles that don't follow the prescribed rules can be traced by the trusted authority easily. Similar methods that adopt pseudonymous certificates to preserve conditional information privacy are also used in [10-11]. However, due to the massive pseudonymous certificates that are used, the overhead of storing them is not negligible. What's more, the reapply-ing of new pseudonymous certificates is always done offline and thus is time-consuming and inconvenient in practice. Another method used to achieve conditional information privacy protection is using mix-zones. In 2007, Freudiger et al. [12] firstly proposed the notion of mix-zones, which are usually the areas that locate at the

*Corresponding Author: Yining Liu; E-mail: ynliu@guet.edu.cn
DOI: 10.3966/160792642019092005015

road intersections in a city. In mix-zones, vehicles can randomly change their pseudonymous certificates without being eavesdropped by the global passive attackers. Thus, the mix-zones can prevent the vehicles that pass through them from being traced. However, for such scheme to work efficiently in practice, the prerequisites that a vehicle must pass through some mix-zones in its route and the number of vehicles that stays in a mix-zone should not be too small must be satisfied. Accordingly, in [13-14], some mix-zones deployment schemes are proposed to solve the above mentioned problems. Group-signature based schemes [7, 15-19] are also promising in solving the problem of conditional information privacy protection. In group-signature based schemes, vehicles are formed into a group, and each vehicle signs message on behalf of the group. By this way, the identities of the concrete vehicles are concealed. However, the group-signature based schemes suffer from the heavy computational burden in the process of CRL checking and the signature verification [20]. Accordingly, Vehicular cloud computing (VCC) is proposed to solve the problem of computing and storage deficiencies with regard to a single vehicle [21]. In 2015, He et al. proposed an ID-based CPPA scheme for VANET [22], which avoided using the complex bilinear pairing computations and thus attained better performance in computation cost.

Another important issue to be concerned in V2I is the reduction of handover authentication delay for the vehicles when they move from the coverage of one RSU to another. In 2013, Li et al. [23] presented a lightweight identity authentication protocol in which a dynamic secret session process (DSSP) was proposed to perform the mutual authentication between the vehicles and the RSUs. In Li's scheme, only lightweight operations such as random sequence generation, hash, XOR, etc. are used. Thus, the protocol can attain a lower handover authentication delay compared with other schemes. Nevertheless, it was proved by Zhou et al. [24] that Li's scheme was vulnerable against parallel session attack and location privacy leakage attack. In [25-28], ID-based cryptographic methods are introduced to achieve a rapid handover authentication. In these schemes, the public keys of the vehicles, instead of being distributed by a public key infrastructure (PKI), are generated from some general information such as user name, address etc. of them. Thus, the time of key distribution is reduced and the speed of handover authentication is enhanced. However, in the ID-based approaches, every time the handover operations are executed, both vehicles and RSUs are required to compute new operational keys, which brings time and computation overhead significantly.

Accordingly, to satisfy the requirements of conditional information privacy protection as well as reduce the handover authentication delay, this paper proposes an improved lightweight identity authentication protocol based on [23-24], which has the following principal contributions:

(1) An asymmetrical dynamic secret session process (ADSSP) method is proposed to solve the problem of parallel session attack. Compared to the scheme proposed in [24] that generates two different session secret sequences in the two opposite directions, ADSSP method generates only one session secret sequence and thus is more efficient.

(2) Instead of being distributed to the RSUs by the trusted authentication server once and for all in [23-24], the authentication sequence (A_S) of a vehicle is distributed gradually by the RSUs as the vehicle passes by them. Analysis shows that this distribution mode is more feasible and more secure in practice.

(3) The security level has been improved greatly to tackle the problem of impersonation attack. Analysis shows that even if some RSU is compromised by an adversary, and the adversary obtains all the authentication sequences that stored in the RSU, the adversary can still be detected by the authentication server or the other un-compromised RSUs as soon as it initiates the impersonation attacks.

The remainder of the paper is organized as follows. In Section 2, the system model and notations are introduced. In Section 3, the related works are reviewed, and the improved version is presented in Section 4. Finally, the proposed protocol is respectively analyzed and concluded in Section 5 and Section 6.

2 System Model and Notations

2.1 System Model and Related Assumptions

In the system model, VANET consists of three major parties, an Authentication Server (AS), a number of Roadside Units (RSUs), and a large number of vehicles. The AS is assumed to be fully trusted, RSUs are assumed to be semi-trusted, and the vehicles are untrusted. To communicate with the RSUs, an on-board unit (OBU) is assumed to be mounted on each vehicle, which is generally thought to have limited capacities of computing and storing. In the following sections, the on-board unit (OBU) is used to denote a specified vehicle. In addition, both of the wired and wireless channels in VANET are not assumed to be secure. That is to say, an attacker can intercept the transmitted messages transmitted over the channels and launch various attacks.

2.2 Notations

The main notations are listed in Table 1.

Table 1. Summarizing of the main notations

symbol	Definition
OBU / *RSU* / *AAAServer*	Vehicle terminal/Roadside unit/Authentication server
UID / *metaUID* / *DID*	Account ID /Encrypted value of UID/Dynamic ID of a vehicle
K	Secret key shared by a vehicle and the authentication server
N	Nonce
RID	ID of RSU
$E(\cdot)$ / $D(\cdot)$	Encryption/Decryption function
DN / *IN*	Direct neighbor/Indirect neighbor
VSI / *OSVI* / *TBSVI*	Vehicles service information table/On-serving vehicles information table/To-be-served vehicles information table
Apk / *Ask*	Public/Secret key of Authentication server
Rpk / *Rsk*	Public/Secret key of RSU
S_r / A_S / *IS*	Random session secret sequence/Authentication sequence/Sequence generated from A_S XOR S_r
T	Timestamp
RTA / *ATA*	Request to authentication sequence/Answer to authentication sequence
$H(\cdot)$	Hash function
$Sig(\cdot)$	Signature function

3 Review and analysis of LIAP

3.1 Review of LIAP

3.1.1 Dynamic Session Secret Process (DSSP)

In LIAP, the DSSP method is proposed to realize the mutual authentication between two parties such as a vehicle and a RSU in V2I communication. Before the authentication process begins, the two parties should share a common random session secret sequence S_r in advance. S_r is composed of a set of binary numbers $X=\{x_0,x_1,\cdots,x_k,\cdots x_{i-1}\}$ where x_k denotes the $(k+1)th$ element in X, i is a redefined system parameter. The detailed process is described as follows:

Assume that two parties A and B are involved in the process, A firstly generates a Request To Authenticate (RTA) sequence RTA_A and sends it to B. The RTA sequence is with the form of a vector (r,q), where $r=\{r_i,i=0,...,m-1\}$, $1\le r_i\le|X|$, $q=\{q_i,i=1,\cdots,m-1\}$, $1\le q_i\le(|X|-r_i+1)$.

Then B generates an Answer To Authenticate (ATA) sequence ATA_B as the response to RTA_A. The ATA sequence is also a set and has the form $\{(r_0,q_0)\to a_0,(r_1,q_1)\to a_1,\cdots,(r_{m-1},q_{m-1})\to a_{m-1}\}$, where $(r_i,q_i\to a_i)$ denotes a mapping from RTA to ATA, which means a_i is obtained from truncating the elements of X from the $(r_i+1)th$ element to the $(r_i+q_i+1)th$ element. Meanwhile, B generates its own challenge sequence RTA_B, and returns the concatenation $RTA_B \,||\, ATA_B$ to A.

Thereafter, A checks to determine whether ATA_B is valid, if so, A computes ATA_A as a response to RTA_B and forwards it to B; otherwise, the process is interrupted.

At last, B checks whether the ATA_A sequence is correct. If so, the mutual authentication is successful; else, the whole authentication process fails.

3.1.2 LIAP Protocol

The LIAP protocol consists of three phases: initial phase, fast handover authentication phase and renewal phase. Before the authentication process begins, some pre-configured information should be computed and stored in the each registered vehicle and the authentication server. For example, to a specified vehicle OBU_i, firstly, a secret key $K_i = H(UID_i \,||\, PWD_i)$ is generated and stored in OBU_i and the authentication server. Where UID_i is the vehicle OBU_i's account ID distributed by the Internet Service Provider (ISP) in advance, PWD_i is a password provided by OBU_i and $H(\cdot)$ is a hash function. Secondly, an encrypted value of UID_i is computed and is denoted as $metaUID_i = E_{K_u}(UID_i)$, where $E_{K_u}(\cdot)$ is an encryption function, and K_u is the public key of the authentication server. Because the renewal phase is identical to the initial phase, we just introduce the former two phases:

Initial phase. The initial phase of LIAP is triggered whenever a vehicle OBU_i applies to join the VANET. The detailed steps are shown as follows:

Step 1. OBU_i sends to the roadside unit RSU_j a joining request as well as its $metaUID_i$.

Step 2. RSU_j generates a nonce N_R, and sends back to OBU_i the value $N_R \| RID_j$.

Step 3. OBU_i produces random number N_O, calculates the authentication sequence $A_S = H(K_i \| N_O \| N_R)$, then generates a random session secret sequence S_r and computes $IS = S_r \oplus A_S$. Afterwards, OBU_i sends $metaUID_i \| IS \| N_O \| N_R \| RTA_{OBU_i}$ to the authentication server via RSU_j.

Step 4. After receiving the information from OBU_i, the authentication server firstly calculates $UID_i \leftarrow D_{K_r}(metaUID_i)$, where K_r is the secret key of the authentication server, then employs the UID_i as an index to lookup the register table and get the corresponding secret value K_i. Subsequently, the authentication server reconstructs A_S and then computes $S_r \leftarrow IS \oplus A_S$. Finally, the authentication server generates a nonce N_A and a timestamp T_S, then sends $E_{RKu}(S_r \| T_S \| N_A)$ to RSU_j, where *RKu* is the public key of RSU_j.

Step 5. RSU_j decrypts the information from the authentication server and obtains S_r. Thus, RSU_j and OBU_i share a common random session secret sequence S_r and can commence a mutual authentication process using DSSP method, i.e. the vehicle uses ATA_{RSU} to verify the validity of RSU_j and vice versa. If the verification fails on either side, then the whole authentication process is terminated. Once the authentication process completes successfully, RSU_j sends to the authentication server the information $E_{AKu}(RID_j \| N_A)$, with which the authentication server updates its register table to track the current position of OBU_i.

Fast handover authentication phase. Before the beginning of the handover authentication phase, the authentication sequence of the vehicle OBU_i is pre-distributed by the authentication server to all the RSUs that the vehicle is expected to pass by in the future. The detailed steps are as follows:

Step 1. OBU_i transmits to RSU_{j+1} a request for handover authentication as well as its $metaUID_i$.

Step 2. RSU_{j+1} checks whether the A_S of OBU_i has existed in its memory or not, If RSU_{j+1} has not received the corresponding A_S from the authentication server, or the timestamp of the A_S has expired, then the fast handover authentication phase is terminated and the renewal phase is triggered, else, RSU_{j+1} sends its ID RID_{j+1} to OBU_i.

Step 3. Thereafter, OBU_i generates a new random sequence S_r and performs an *XOR* operation with A_S to get IS. Subsequently, the value $IS \| RTA_{OBU_i}$ is sent from OBU_i to RSU_{j+1}.

Step 4. Upon receiving the value, RSU_{j+1} calculates the session secret sequence $S_r = IS \oplus A_S$. With the shared session secret sequence, the DSSP method is used to execute the mutual authentication between OBU_i and RSU_{j+1}.

Step 5. Finally, RSU_{j+1} transmits $E_{AKu}(metaUID_i \| RID_{j+1} \| N_A)$ to the authentication server, which will update its register table later.

3.2 Security and Efficiency Analysis of LIAP

3.2.1 Problem of Location Privacy Leakage and Parallel Session Attack

Zhou et al. [24] states that LIAP is vulnerable against location privacy leakage and parallel session attack.

Location privacy leakage. In LIAP, since the $metaUID$ of a vehicle remains unchanged in the whole process as it accesses the network service, an adversary can track the vehicle by monitoring the $metaUID$. To tackle the problem of location privacy leakage, Zhou et al. [18] adopted quadratic residues operations to generate dynamic identity for the vehicles, and thus the location privacies of the vehicles were preserved.

Parallel session attack. The details of the parallel session attack are shown as follows:

Step 1. An adversary *A* impersonates a legal vehicle OBU_i and sends the intercepted $metaUID_i$ of OBU_i to two distinct RSUs simultaneously. The two RSUs are denoted as RSU_{j+1} and RSU'_{j+1} respectively.

Step 2. Thereafter, either RSU replies its ID, i.e. RID_{j+1} or RID'_{j+1}, to *A*.

Step 3. *A* chooses a random sequence IS' and a challenge RTA_A. Thereafter, the concatenation $IS' \| RTA_A$ is forwarded to RSU_{j+1}.

Step 4. RSU_{j+1} firstly computes $S_r' = IS' \oplus A_S$, then generates $ATA_{RSU_{j+1}}$ and produces its own challenge $RTA_{RSU_{j+1}}$, finally, the value $RTA_{RSU_{j+1}} \| ATA_{RSU_{j+1}}$ is sent to *A*.

Step 5. A transmits $IS^{'} \| RTA_{RSU_{j+1}}$ to $RSU^{'}_{j+1}$.

Step 6. $RSU^{'}_{j+1}$ calculates $S_r^{'} = IS^{'} \oplus A_s$ and gets $S_r^{'}$. With this $S_r^{'}$, $RSU^{'}_{j+1}$ works out $ATA_{RSU^{'}_{j+1}}$ and combines it with its challenge $RTA_{RSU_{j+1}^{'}}$, and then sends the value $ATA_{RSU_{j+1}^{'}} \| RSU_{j+1}^{'}$ to A.

Step 7. Upon receiving the value from $RSU^{'}_{j+1}$, A sends $ATA_{RSU^{'}_{j+1}}$ to RSU_{j+1} as its response to $RTA_{RSU_{j+1}}$.

Thus, the adversary is authenticated successfully by RSU_{j+1} and can access the network service freely. To tackle the problem of parallel session attacks, Zhou et al. [24] proposed a scheme in which two different session secret sequences were generated in the two opposite directions respectively. However, such solution needs to generate two distinct secret sequences on both of the vehicle's and the RSU's sides.

3.2.2 Efficiency and Security Problems about the Authentication Sequences A_S in [23-24]

In the fast handover authentication phase in [23-24], in order to speed up the handover authentication process, the authentication server will send the authentication sequence A_S of a vehicle to all the RSUs that the vehicle may possibly pass by in the future. Nevertheless, we state that such distribution method of A_S is infeasible and poses significant security risks in practice. The details are described as follows:

(1) Because of the complexity of urban roads and diversity of the vehicles' routes, it is quite difficult for an authentication server to predict the travel routes for all the vehicles. Thus, it is hard to determine which RSUs the vehicles will pass by in the future. A plausible alternative is distributing of the authentication sequences to all the RSUs deployed in the city, but such method will bring huge storage overhead for RSUs.

(2) Consider the scenario that some RSU is compromised by an adversary, then the adversary can obtain all the authentication sequences stored in the compromised RSU in [23] (or all the authentication sequences and pseudo-identities PIDs in [24]). Since the authentication sequences (or the authentication sequences and the pseudo-identities in [24]) are the only secret credentials required by other RSUs to verify the legal vehicles in the handover authentication phase, the adversary can use these authentication sequences (or authentication sequences and pseudo-identities in [24]) to impersonate all the legal vehicles and initiate a variety of attacks.

3.3 Our Contributions

In order to solve the problems of security and efficiency that exist in [23-24], an improved lightweight identity authentication protocol is proposed, which can not only achieve short handover authentication latency, but also preserve the privacy of the vehicles. Especially, the security level of the proposed protocol is improved greatly to tackle the problem of impersonation attack. The main contributions are summarized as follows:

(1) An asymmetrical dynamic secret session process (ADSSP) method is proposed to solve the problem of parallel session attack.

(2) The authentication sequence (A_S) of a vehicle is distributed gradually by the RSUs when the vehicle passes by them. Analysis shows that this distribution mode is more feasible and more secure in practice.

(3) The security level has been improved greatly to tackle the problem of impersonation attack. In our scheme, even though some RSU is compromised by an adversary and the adversary obtains the authentication sequences of the vehicles that stored in the RSU, the adversary can still be detected as soon as it launches the impersonation attacks.

4 The Improved Lightweight Identity Authentication Protocol

4.1 Asymmetrical Dynamic Secret Session Process (ADSSP)

The DSSP method introduced in [23] is symmetrical in essence. I.e., the form of the ATA sequence transmitted from one party to the other and the other way round are just the same. However, the symmetrical characteristic can be utilized by adversaries to perform parallel session attacks. To solve the problem of parallel session attacks, an Asymmetrical Dynamic Secret Session Process (ADSSP) method is proposed in this paper. The details of ADSSP method are as follows:

Suppose two parties C and R involve in the authentication process. Both parties share a random session secret sequence S_r in advance. S_r is a variable length binary number, which can also be represented as a set of elements $\mathrm{X} = \{x_1, x_2, \cdots x_i, \cdots\}$, where $x_i \in \{0,1\}$ is the i-th bit of S_r. Suppose further that C is the challenger (initiator) and R is the responder. C and R will perform the following steps to achieve mutual authentication:

(1) C generates a request to answer sequence RTA_C and forwards it to R, where RTA_C is a set of two-tuples that has the form $\{(\mathrm{r}_1, q_1), \cdots, (\mathrm{r}_m, q_m)\}$, $1 \le r_i \le |X|$, $1 \le q_i \le (|X| - r_i + 1)$, $1 \le i \le m$, m is a

predefined system parameter.

(2) After receiving RTA_C, R computes the answer to authenticate sequence ATA_R, where $ATA_R = \{(r_1,q_1)\rightarrow a_1,(r_2,q_2)\rightarrow a_2,\cdots,(r_m,q_m)\rightarrow a_m\}$. Note that $(r_i,q_i)\rightarrow a_i$ represents a mapping from RTA to ATA, which is the value obtained by truncating X from its $r_i th$ element to $(r_i+q_i)th$ element. Meanwhile, R generates its own challenge RTA_R with the same form as RTA_C. I.e. $RTA_C = \{(r_1',q_1'),(r_2',q_2'),\cdots,(r_m',q_m')\}$ Thereafter, $ATA_R \,||\, RTA_R$ is transmitted from R to C.

(3) C checks the validity of ATA_R, then calculates the intermediate value $M = \{(r_1',q_1')\rightarrow a_1',(r_2',q_2')\rightarrow a_2',\cdots,(r_m',q_m')\rightarrow a_m'\}$. Finally, the answer to authentication sequence $ATA_C = a_1' + a_2' + \cdots + a_m'$ is calculated and sent to R. Note ATA_C is the sum of all the elements in M, which is different from ATA_R in form.

(4) R checks the validity of ATA_C.

An example of ADSSP is illustrated in Figure 1. Assume the RTA sequence sent by C is $RTA_C = \{(1,2),(6,1),(9,3),(11,2)\}$, then the corresponding ATA sequence of R should be $ATA_R = 01001110$. If, in turn, the RTA sequence sent by R is $RTA_R = \{2,2),(5,1),(8,3),(12,1),$ then C firstly calculates the intermediate value $M = \{11,1,101,0\}$ and then computes the sum of the numbers in M, i.e. $ATA_C = 11+1+101+0 = 1001$ as its answer to authentication sequence.

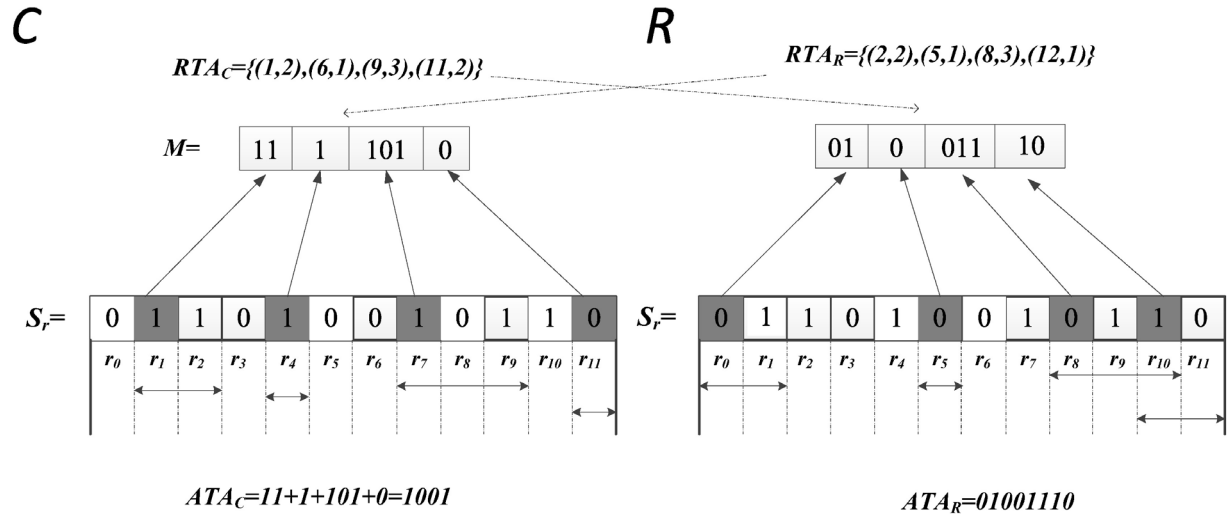


Figure 1. Illustrative example of ADSSP

4.2 Definitions of Direct Neighbor (DN) and Indirect Neighbor (IN)

In practice, as a vehicle moves from the range area of one RSU to another, a handover authentication operation should be executed between the vehicle and the related RSUs. Obviously, a vehicle can only switch between the range areas of two adjacent RSUs. Consider two roadside units RSU_j and RSU_{j+1}. RSU_j is referred as a Direct Neighbor (DN) of RSU_{j+1} and vice versa if they directly communicates with each other without other RSU relaying. Otherwise, they are referred to as Indirect Neighbor (IN) to each other.

4.3 The Detailed Scheme

In this scheme, several tables are assumed to be maintained in the authentication server and the RSUs. The tables are used to track the current states and store the related parameters of the vehicles. It is assumed that a Vehicles Service Information (VSI) table (Figure 2) is stored in the authentication server. The VSI table includes such information as user account ID, shared key, current service state, dynamic ID, present RSU, timestamp etc. of every registered vehicle in a city. It is assumed further that two other tables are saved in each RSU, one is On-Serving Vehicles Information (OSVI) table (Figure 3), in which stores such information as dynamic ID, authentication sequence, timestamp for the vehicles that are in the range area of the RSU; the other is To-Be-Served Vehicles Information (TBSVI) table, where the similar information (i.e. dynamic ID, authentication sequence, timestamp) of the vehicles that are in the range area of the RSU's direct neighbors are stored.

The proposed scheme comprises four phases, namely initial authentication phase, fast handover authentication phase, renewal phase, and withdrawal from service phase.

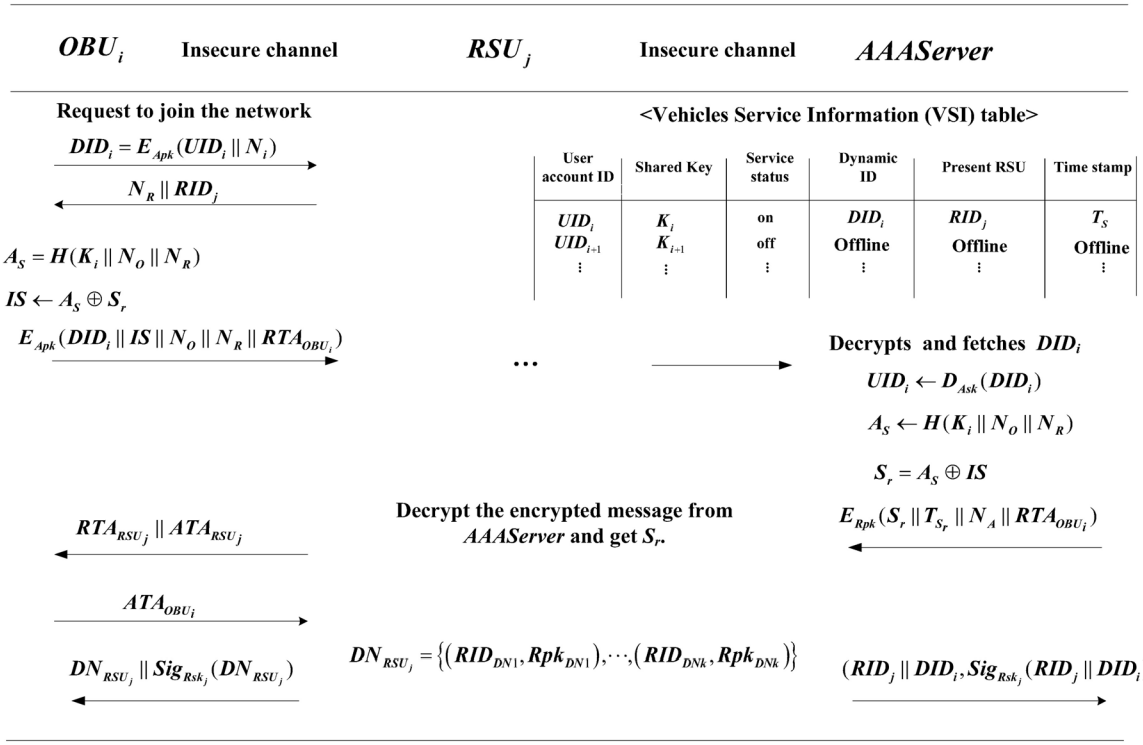


Figure 2. Illustration of the initial authentication phase

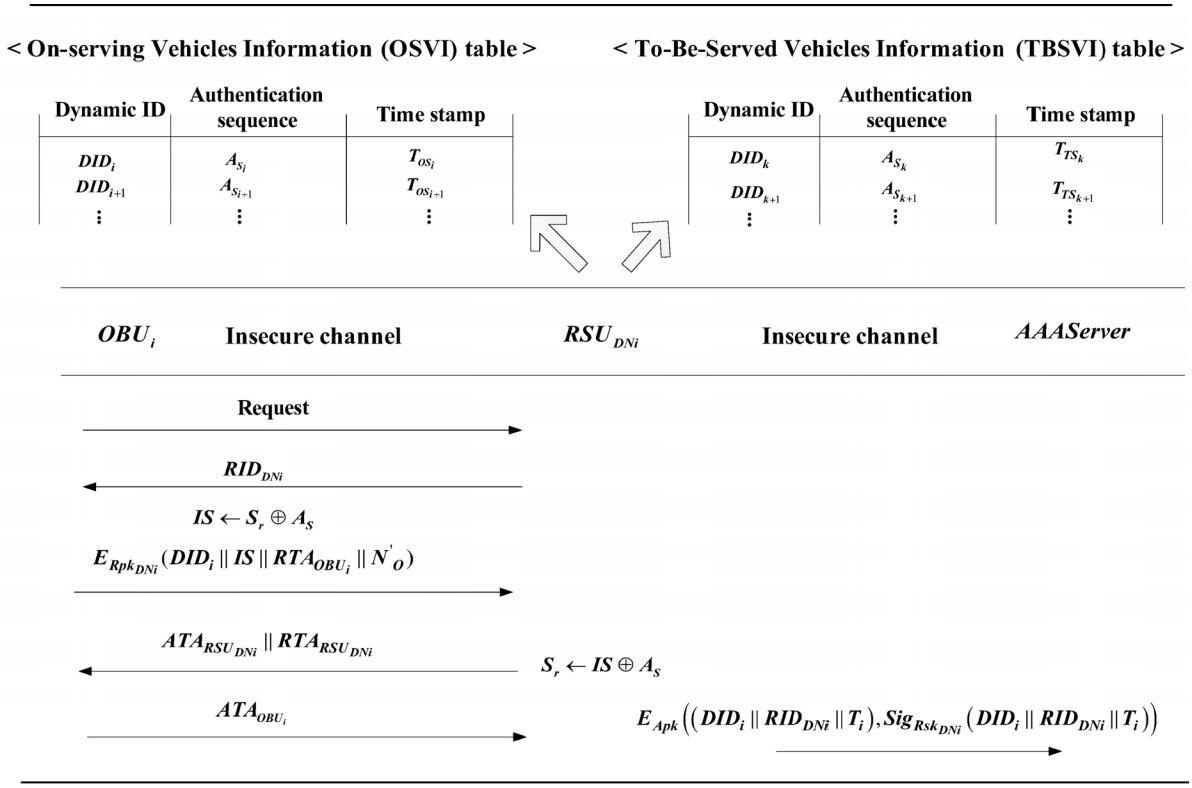


Figure 3. Fast handover authentication phase

4.3.1 Initial Authentication Phase

This phase is initiated by a vehicle whenever it requests to access the VANET. In this phase, the mutual authentication process is executed between the vehicle and the authentication server via a RSU. The detailed steps are showed in Figure 2.

As shown in Figure 2, assume a vehicle OBU_i is currently in the range area of RSU_j and wants to access the network. The initial authentication phase

will be implemented, and the details are shown as follows:

(1) OBU_i sends a request and its dynamic identity $DID_i = E_{Apk}(UID_i \| N_i)$ to RSU_j, where N_i is a nonce.

(2) RSU_j generates a nonce N_R, and sends $N_R \| RID_j$ to OBU_i.

(3) After receiving the response from RSU_j, OBU_i generates a nonce N_O and computes the authentication sequence $A_S = H(K_i \| N_O \| N_R)$, then produces a session secret sequence S_r and XOR it with A_S to get $IS = S_r \oplus A_S$. Finally, a request to authentication sequence RTA_{OBUi} is generated, and the encrypted value $E_{Apk}(DID_i \| IS \| N_O \| N_R \| RTA_{OBU_i})$ is computed and then sent to the authentication server $AAAServer$ via RSU_j.

(4) The authentication server decrypts the value received from OBU_i and gets DID_i, which is then decrypted to get user account identity UID_i. With UID_i, the authentication server queries its VSI table to find the corresponding K_i of the vehicle OBU_i. Afterwards, the authentication sequence $A_S = H(K_i \| N_O \| N_R)$ is computed and used to work out the random secret session sequence $S_r = A_S \oplus IS$. Finally, the authentication server generates a nonce N_A, and sets a timestamp T_{S_r} for S_r, then forwards $E_{Rpk}(S_r \| T_{s_r} \| N_A \| \mathrm{RTA}_{OBUi})$ to RSU_j.

(5) RSU_j computes ATA_{RSU_j} as the response to RTA_{OBU_i}. Then RSU_j generates its challenge RTA_{RSU_j}, and forwards $RTA_{RSU_j} \| ATA_{RSU_j}$ to OBU_i.

(6) OBU_i checks the validity of ATA_{RSU_j}, then computes the intermidate value M. Finally, the answer sequence ATA_{OBU_i} is computed and sent to RSU_j. ATA_{OBU_i} is the sum of the elements in set M.

(7) RSU_j checks the validity of ATA_{OBU_i}, if yes, the vehicle is authenticated successfully. Thereafter, RSU_j stores the corresponding DID_i and A_S of OBU_i in its OSVI table. Meanwhile, a timestamp T_{OS} is set for the new table entry. The timestamp T_{OS} is used to control the amount of time that a vehicle can stay in a RSU's range area. Once the timestamp expires, RSU_j will log out the vehicle compulsively.

(8) Finally, RSU_j transmits $DN_{RSU_j} = \{(RID_{DN1}, Rpk_{DN1}), \cdots, (RID_{DNi}, Rpk_{DNi})\}$ and its signature to OBU_i, where DN_{RSU_j} includes the IDs and public keys of all the RSU_j's direct neighbors. Furthermore, the information $\{RID_j \| DID_i, Sig_{Rsk_j}(RID_j \| DID_i)\}$ is sent to the authentication server. The authentication server verifies the signature $Sig_{Rsk_j}(RID_j \| DID_i)$, then stores $RID_j \| DID_j$1 into its VSI table, and sets the service status of OBU_i to be "on".

4.3.2 Fast Handover Authentication Phase

To speed up the process of handover authentication. As soon as the initial authentication finishes, the dynamic ID DID_i as well as the authentication sequence A_s of the vehicle OBU_i are encrypted by RSU_j and then transmitted to all its DNs. Thereafter, each DN decrypts the received information and stores it into its TBSVI table. Meanwhile, a timestamp T_{TS} is set for the newly added table entry. The timestamp is used to control the storage overhead of RSU_j. Once the timestamp T_{TS} expires, the corresponding entry stored in the TBSVI table will be deleted.

The fast handover authentication phase is initiated by the vehicle when it switches from the range area of one RSU to another. Assume OBU_i switches from the range area of RSU_j to its direct neighbor RSU_{DNi}. The detailed steps are described as follows, which are also illustrated in Figure 3.

(1) OBU_i sends a request for handover authentication to RSU_{DNi}.

(2) RSU_{DNi} responds with its ID RID_{DNi} to OBU_i.

(3) Thereafter, OBU_i generates a new random session secret sequence S_r and computes $IS = S_r \oplus A_S$. Subsequently, $E_{Rpk_{DNi}}(DID_i \| IS \| RTA_{OBU_i} \| N'_O)$ is computed and sent to RSU_{DNi}, where Rpk_{DNi} is the public key of RSU_{DNi} and N'_O is a nonce.

(4) Upon receiving the value from OBU_i, RSU_{DNi} firstly decrypts it and obtains DID_i, then queries the TBSVI table to check whether the DID_i exists in the table and the timestamp T_{TS} is valid. If so, RSU_{DNi} computes the session secret sequence $S_r = IS \oplus A_S$; else, the handover authentication phase is terminated and turned to renewal phase.

(5) Now, with the shared S_r, the mutual authentication is executed between OBU_i and RSU_{DNi} by means of ADSSP method. i.e., RSU_{DNi} computes $ATA_{RSU_{DNi}}$ as the response to RTA_{OBU_i} and generates its own challenge $RTA_{RSU_{DNi}}$, then forwards $ATA_{RSU_{DNi}} \| RTA_{RSU_{DNi}}$ to RSU_{DNi}.

(6) OBU_i checks to determine whether $ATA_{RSU_{DNi}}$ is

valid or not, if so, OBU_i computes the intermediate value M, and then calculates the answer sequence ATA_{OBU_i}. Finally, ATA_{OBU_i} is forwarded it to RSU_{DN_i}. Otherwise, the handover authentication phase is terminated.

(7) RSU_{DNi} checks to determine the correctness of the answer sequence ATA_{OBU_i}. If so, RSU_{DNi} sends the value $E_{Apk}\left(\left(DID_i \| RID_{DNi} \| Ti\right), Sig_{Rsk_{DNi}}\left(DID_i \| RID_{DNi} \| Ti\right)\right)$ to the authentication server, where Ti is the current time information, which increases the randomness of the encrypted value; otherwise, the process is terminated.

Up to now, the handover authentication completes, and OBU_i can accept the network service from RSU_{DNi}. But the following operations should be performed further to update the related tables in the authentication server and the related RSUs, which will record the current state of the vehicle OBU_i.

(1) The authentication server updates the corresponding table entry about OBU_i in its VSI table. For example, the state of " Current RSU" is set to be RID_{DNi} and timestamp T_S is reset.

(2) RID_{DNi} deletes the table entry about OBU_i in its TBSVI table. Instead, adds a new table entry about OBU_i into its OSVI table, which means OBU_i is in its range area.

(3) RSU_{DNi} encrypts the value $DID_i \| A_S$ with the public keys of all of its DNs and then sends the encrypted values to the DNs respectively. Upon receiving the message, each of the DNs queries its OSVI table to check whether the entry has existed. If so, then deletes it and establishes a new entry into the TBSVI table. Else, establishes a new entry in the TBSVI table.

4.3.3 Renewal Phase

In this phase, RSU just re-executes the initial phase to complete the verification between $AAAServer$ and the vehicle ·

4.3.4 Withdrawal from Service Phase

When a vehicle wants to exit from the current network service, it sends a request for logout as well as its dynamic ID to the authentication server via the RSU whose range area the vehicle is in. Afterwards, the authentication server searches its VSI table to find the entry about the vehicle, and changes the vehicle's service status to be "off". In practice, in case that a vehicle forgets to or maliciously refuses to perform the logout operation even though it has left the range area, the authentication server can check the corresponding timestamp T_S of the vehicle to see whether it has expired. If so, then the authentication server can log out the vehicle and set the service status of the vehicle as "off" compulsively. A logged out vehicle should turn to the initial authentication phase to reapply for new network service.

5 Performance Analysis

5.1 Security Analysis

It is assumed that an adversary can eavesdrop all the information transmitted on the unsecure channels, and can initiate a variety of attacks such as key compromising, replaying attack, location detection, impersonating valid users and so on.

5.1.1 Mutual Authentication

The improved protocol adopts ADSSP method to carry out the authentication between the vehicles and the RSUs. Since ADSSP method employs a bidirectional challenge-response mode, it can realize mutual authentication for the two engaged parties.

5.1.2 Compromising of the Session Secret Sequence S_r

In ADSSP, a vehicle and a RSU mutually exchange the $RTA \| ATA$ sequences twice. Assume an adversary attempts to rebuild the S_r by intercepting both the $RTA \| ATA$ sequences transmitted between the vehicle and the RSU, and in the worst case that the challenge $RTA = \left\{\left(r_i, q_i\right) | i = 1 \cdots L, L \le C_1/2, 1 \le r_i \le C_1, q_i = 1\right\}$. If the element r_i do not repeat in the two RTA sequences, then the success rate for the adversary to rebuild the valid S_r is only $1/L^2$. Furthermore, in practice, the length of RTA is always variable, so it is more difficult for an adversary to reconstruct the correct S_r.

5.1.3 Compromising of the Secret Key K

Consider the worst scenario, an adversary is assumed to obtain the authentication sequences A_S of the vehicle OBU_i and is lucky enough to get the correct nonce N_R, he must execute hash operation about $2^{C_K + C_O - 1}$ times to figure out the secret key K_i from the corresponding A_S, where C_K and C_O are the length of the secret key K_i and the nonce N_O respectively. So, in practice, as long as the length of K_i and N_O is long enough, the secret key K_i is robust against the brute force attacks.

5.1.4 Location Privacy Problem

In the initial authentication phase in ILIAP, the identity information sent by the vehicle OBU_i is not its account ID UID_i, but a dynamic ID $DID_i = E_{Apk}(UID_i \| N_i)$, which conceals the account ID in the encrypted value. Furthermore, with the random number N_i, every time the vehicle requests to join the VANET in the initial authentication phase, the dynamic identity DID_i that is forwarded to the RSU is distinct. Besides, in the fast handover authentication phase, the dynamic ID DID_i is concealed in the encrypted value $E_{Rpk_{DNi}}(DID_i \| IS \| RTA_{OBU_i} \| N'_O)$, which makes an adversary impossible to extract the valid DID_i without the secret key of RSU_{DNi}. Thus, the location privacy of the vehicle is protected.

5.1.5 Resistance of Parallel Session Attack

The ADSSP method adopted in the proposed protocol inherently has the ability to resist the parallel session attack. Consider the scenario shown in Figure 4. It is assumed that RSU_{k+1} and RSU'_{k+1} have received the authentication sequence A_S of the valid user OBU_i in the initial authentication phase, and an adversary A attempt to impersonate the OBU_i to access the network service in the handover authentication phase. The detailed process is as follows:

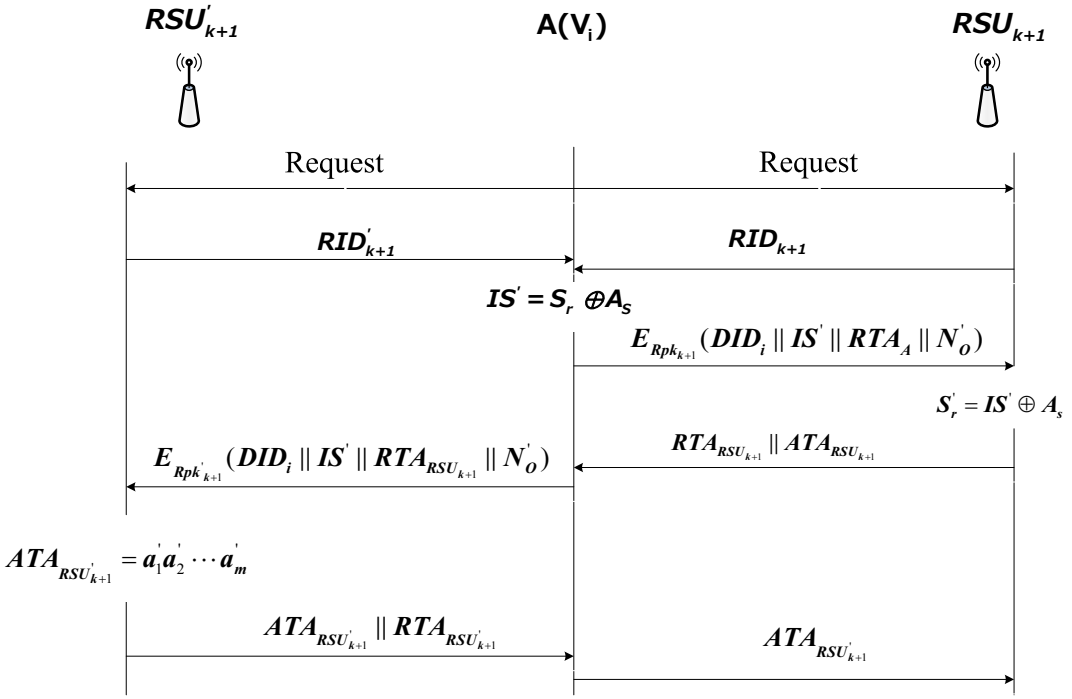


Figure 4. Analysis of parallel session attack resistance for ADSSP

(1) A sends handover authentication requests to RSU_{k+1} and RSU'_{k+1} simultaneously.

(2) RSU_{k+1} and RSU'_{k+1} response their IDs to A respectively.

(3) A generates a session secret sequence S_r and chooses an arbitrary number as the authentication sequence A'_S, then computes $IS' = S_r \oplus A'_S$ and generates a request to authentication sequence RTA_A. Thereafter, the encrypted value $E_{Rpk_{k+1}}(DID_i \| IS' \| RTA_A \| N'_O)$ is forwarded to RSU_{k+1}. Note that A has no knowledge of the correct authentication sequence A_S, but it can obtain the valid dynamic ID DID_i through eavesdropping in the initial authentication phase.

(4) RSU_{k+1} calculates $S'_r = IS' \oplus A_S$, figures out its ATA sequence and generates its request to authentication sequence $RTA_{RSU_{k+1}} = \{(r'_1, q'_1), \cdots, (r'_m, q'_m)\}$ then transmits $RTA_{RSU_{k+1}} \| ATA_{RSU_{k+1}}$ to the adversary A.

(5) A transmits to RSU'_{k+1} the value $E_{Rpk'_{k+1}}(DID_i \| IS' \| RTA_{RSU_{k+1}} \| N'_O)$.

(6) RSU'_{k+1} responses to A the value $RTA_{RSU'_{k+1}} \| ATA_{RSU'_{k+1}}$, note the answer to authentication sequence is $ATA_{RSU'_{k+1}} = a'_1 a'_2 \cdots a'_m$, where $a'_i \leftarrow (r'_i, q'_i)$, $1 \le i \le m$.

(7) A obtains the sequence $ATA_{RSU'_{k+1}}$ sent by RSU'_{k+1} and then forwards it to RSU_{k+1}.

(8) Since the expected answer sequence from A

should be $ATA_A = a_1^{'} + a_2^{'} + \cdots + a_m^{'}$, which is not equal to the value $ATA_{RSU_{k+1}^{'}} = a_1^{'} a_2^{'} \cdots a_m^{'}$, A fails to be verified by the roadside unit RSU_{k+1}.

5.1.6 Resistance of Impersonation Attack

We consider the worst-case scenario in which a roadside unit RSU_A is compromised by an adversary A, and thus A obtains all the information such as A_S, dynamic ID etc. of the legal vehicles stored in RSU_A. With these information, A can initiate impersonation attack. However, it can be proved that even in such a scenario, the impersonation attack can be detected as soon as the adversary performs the handover authentication operation from the range area of one RSU to another. The detailed analysis is showed as follows.

Note that in the withdrawal from service phase, before exiting from the VANET, a legal vehicle must apply to the authentication server for logging off. Thus, an adversary can only use the A_S and dynamic ID information of an online vehicle to impersonate it (otherwise, the authentication server can detect the impostor immediately. The reason is that after the completion of the handover authentication phase, the information $DID_i \| RID_{DNi} \| T_i$ will be sent to the authentication server. Thus, the authenticacion server can detect immediately that an offline vehicle is being served).

In the following subsection, three cases are discussed according to the location of the adversary A relative to the impersonated legal vehicle OBU_i. Without loss of generality, we assume that the current position of OBU_i is in the range area of RSU_L and the adversary A switches to the range area of $RSU_{L^{'}}$.

Case 1. A switches to the range area of $RSU_{L^{'}}$, and $RSU_{L^{'}}$ equals RSU_L (i.e. both A and OBU_i are in the range area of RSU_L). Since it's impossible for a vehicle to switch to the range area that it is currently in, RSU_L can detect the existence of the imposter A immediately.

Case 2. A switches to the range area of $RSU_{L^{'}}$, and $RSU_{L^{'}}$ is an indirect neighbor of RSU_L. In this case, note that before the completion of the fast handover authentication phase, $RSU_{L^{'}}$ will send to the authentication server the dynamic ID of OBU_i and its own ID $RID_{L^{'}}$. The authentication server can detect the existence of the imposter A immediately for it is impossible for a vehicle to "jump" from the range area of one RSU to that of its indirect neighbor.

Case 3. A switches to the range area of $RSU_{L^{'}}$, and $RSU_{L^{'}}$ is a direct neighbor of RSU_L. Note that at the end of the handover authentication phase, $RSU_{L^{'}}$ will send to its direct neighbors the dynamic ID and A_S of the vehicle, which informs RSU_L that OBU_i is currently in its own range area. This will make RSU_L detect the existence of the imposter A immediately, for it is impossible for a vehicle exists in the range areas of itself and its direct neighbor simultaneously.

5.1.7 Qualitative Comparison of Security Performance

A Qualitative comparison of the security performance for the proposed protocol and the existing protocols [23-24] is illustrated in Table 2. As shown in Table 2, only the proposed protocol can resist various attacks, and the literatures [23-24] are vulnerable against impersonation attacks in case of some RSU is compromised.

Table 2. Comparison of security performance

performance	[23]	[24]	Our protocol
Mutual authentication	Yes	Yes	Yes
Location privacy	unsecure	secure	secure
Replay attack	secure	secure	secure
Impersonation attack	unsecure	unsecure	secure
Parallel session attack	unsecure	secure	secure

5.2 Handover Authentication Efficiency Analysis and Evaluation

The computation overheads in handover authentication are evaluated in this subsection for the protocols [24, 29-30] and the proposed protocol. In order to acquire the computation overhead consumed by a single operation in these literatures, the pairing-based cryptography (PBC) library [31] and multi-precision integer and rational arithmetic C/C++ library (MIRACL) [32] are used and installed on a computer with a 3.2G HZ CPU and 8G of memory. In addition, each of the operation is run 50 times and the average value is considered. The operations and their computation overheads are listed in Table 3.

Table 3. Computation overhead of single operation

Operations	Details	Time (milliseconds)
PM	Point multiplication	2.258
BP	Bilinear pairing	6.443
H	Hash (SHA-256)	0.021
EXP	Exponentiation in Bilinear Group	3.212
ENC	AES-128 encryption	0.902
DEC	AES-128 decryption	7.357
MM	Modular multiplication	1.657
MR	Modular square root	2.942

Table 4 presents the comparison of the computation overheads for the existing protocols and the proposed protocol. The overheads are the sum of the time consumed on both the vehicle and the RSU's side. As shown in Table 4, in Chaudhry et al.'s protocol [30], the operations on vehicle's side is $PM \times 1 + H \times 2$ and $PM \times 1 + H \times 4$ on RSU's side. So the total computation cost on both sides is $27.222ms$. In He et al. 's protocol [29], the computation cost on the vehicle's side and the RSU's side is $PM \times 4 + H \times 5 + EXP \times 2$ and $BP \times 3 + H \times 5 + EXP \times 2$ respectively. Thus the whole computation overhead is 168.777ms. In Zhou et al.'s protocol [18], the computation cost is $MM \times 1$ on the vehicle's side and $MR \times 1 + ENC \times 1$ on the RSU's side. The computation overhead on both sides is 5.501 ms. In the proposed protocol, the computation cost is $ENC \times 1$ on vehicle's side and $DEC \times 1 + ENC \times 2$ on the RSU's side, while the computation overhead on both sides is 10.063 ms. It can be seen that Zhou et al. [24] and the proposed protocol are much faster than the protocols adopted in [29-30]. The reason is that the formers choose lightweight operations instead of time-consuming pairing related operations to carry out the mutual authentication. In addition, although the protocol in Zhou et al. [24] are faster than the proposed protocol, the proposed protocol is stronger in the security performance and can resist impersonation attacks in case of some RSU is compromised.

Table 4. Comparison of computation overheads

Protocols	Vehicle's side	RSU's side	Total (ms)
Chaudhry et al. [30]	$PM \times 1 + H \times 2$	$PM \times 1 + H \times 4$	27.222
He et al. [29]	$PM \times 4 + H \times 5 + EXP \times 2$	$BP \times 3 + H \times 5 + EXP \times 2$	168.777
Zhou et al. [24]	$MM \times 1$	$MR \times 1 + ENC \times 1$ MR1+ENC*1	5.501
Proposed protocol	$ENC \times 1$	$DEC \times 1 + ENC \times 2$	10.063

6 Conclusion

In this paper, we first discuss the protocols proposed in Li et al. [23] and Zhou et al. [24], then propose an asymmetrical dynamic secret session process (ADSSP) method to replace the DSSP in [23]. What's more, a novel distribution model of the authentication sequences is put forward to solve the problem of feasibility in reality. Security analysis shows that the proposed protocol is robust to various attacks. In addition, performance analysis shows that the proposed protocol is more efficient in the handover authentication process than the two latest protocols [29-30].

Acknowledgements

This work was partly supported by National Natural Science Foundation of China under grant No. 61662016, Key projects of Guangxi Natural Science Foundation under grant No. 2018JJD170004.

References

[1] S. J. Horng, S. F. Tzeng, T. Li, X. Wang, P. Huang, M. K. Khan, Enhancing Security and Privacy for Identity-Based Batch Verification Scheme in VANETs, *IEEE Transactions on Vehicular Technology*, Vol. 66, No. 4, pp. 3235-3248, April, 2017.

[2] P. Vijayakumar, M. Azees, L. J. Deborah, CPAV: Computationally Efficient Privacy Preserving Anonymous Authentication Scheme for Vehicular Ad Hoc Networks, *2015 IEEE 2nd International Conference on Cyber Security and Cloud Computing*, New York, USA, 2015, pp. 62-67.

[3] J. Shao, R. Lu, X. Lin, C. Zuo, New Threshold Anonymous Authentication for VANETs, *2015 IEEE/CIC International Conference on Communications in China (ICCC)*, Shenzhen, China, 2015, pp. 1-6.

[4] X. Cheng, L. Yang, X. Shen, D2D for Intelligent Transportation Systems: A Feasibility Study, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 16, No. 4, pp. 1784-1793, August, 2015.

[5] F. Wei, P. Vijayakumar, Q. Jiang, R. Zhang, A Mobile Intelligent Terminal Based Anonymous Authenticated Key Exchange Protocol for Roaming Service in Global Mobility Networks, *IEEE Transactions on Sustainable Computing*. DOI: 10.1109/TSUSC.2018.2817657.

[6] X. Liu, Y. Li, J. Qu, Q. Jiang, MAKA: Provably Secure Multi-factor Authenticated Key Agreement Protocol, *Journal of Internet Technology*, Vol. 19, No. 3, pp. 669-677, May, 2018.

[7] J. Shao, X. Lin, R. Lu, C. Zuo, A Threshold Anonymous Authentication Protocol for VANETs, *IEEE Transactions on Vehicular Technology*, Vol. 65, No. 3, pp. 1711-1720, March, 2016.

[8] A. Ghosh, V. V. Paranthaman, G. Mapp, O. Gemikonakli, Exploring Efficient Seamless Handover in VANET Systems Using Network Dwell Time, *Eurasip Journal on Wireless Communications and Networking*, Vol. 2014, No. 1, pp. 1, December, 2014.

[9] Y. Sun, R. Lu, X. Lin, X. Shen, J. Su, An Efficient Pseudonymous Authentication Scheme with Strong Privacy Preservation for Vehicular Communications, *IEEE Transactions on Vehicular Technology*, Vol. 59, No. 7, pp. 3589-3603, September, 2010.

[10] R. Lu, X. Lin, H. Zhu, P. H. Ho, X. Shen, ECPP: Efficient Conditional Privacy Preservation Protocol for Secure Vehicular Communications, *IEEE INFOCOM 2008 - The 27th Conference on Computer Communications*, Phoenix, AZ, 2008, pp. 1229-1237.

[11] C. D. Jung, C. Sur, Y. Park, K. H. Rhee, A Robust Conditional Privacy-Preserving Authentication Protocol in VANET, *Security and Privacy in Mobile Information and Communication Systems, First International ICST Conference*, Turin, Italy, 2009, pp. 35-45.

[12] J. Freudiger, M. Raya, M. Félegyházi, P. Papadimitratos, J. P. Hubaux, Mix-Zones for Location Privacy in Vehicular Networks, *ACM Workshop on Wireless Networking for Intelligent Transportation Systems*, Vancouver, BC, Canada, August, 2007, pp. 1-7.

[13] Y. Sun, B. Zhang, B. Zhao, X. Su, J. Su, Mix-zones Optimal Deployment for Protecting Location Privacy in VANET, *Peer-to-Peer Networking and Applications*, Vol. 8, No. 6, pp. 1108-1121, November, 2015.

[14] B. Amro, Protecting Privacy in VANETs Using Mix Zones With Virtual Pseudonym Change, *International Journal of Network Security and Its Applications*, Vol.10, No.1, pp.11-21, January, 2018.

[15] J. Y. Hwang, S. Lee, B. -H. Chung, H. S.Chok, D. H. Nyang, Group Signatures with Controllable Linkability for Dynamic Membership, *Information Sciences*, Vol. 222, No. 10, pp. 761-778, February, 2013.

[16] J. Y. Hwang, L. Chen, H. S. Cho, D. Nyang, Short Dynamic Group Signature Scheme Supporting Controllable Linkability, *IEEE Transactions on Information Forensics and Security*, Vol. 10, No. 6, pp. 1109-1124, June, 2015.

[17] J. Guo, J. P. Baugh, S. Wang, A Group Signature Based Secure and Privacy-Preserving Vehicular Communication Framework, *2007 Mobile Networking for Vehicular Environments*, Anchorage, AK, 2007, pp. 103-108.

[18] G. Calandriello, P. Papadimitratos, J.-P. Hubaux, A. Lioy, Efficient and Robust Pseudonymous Authentication in VANET, *International Workshop on Vehicular Ad Hoc Networks, Vanet 2007*, Montréal, Québec, Canada, 2007, pp. 19-28.

[19] Y. Liu, C. Cheng, J. Cao, T. Jiang, An Improved Authenticated Group Key Transfer Protocol Based on Secret Sharing, *IEEE Transactions on Computers*, Vol. 62, No. 11, pp. 2335-2336, 2013.

[20] Y. Liu, W. Guo, Q. Zhong, G. Yao, Lightweight Vehicular Authentication Protocol (LVAP) Using Group Communication, *International Journal of Communication Systems*, Vol. 30, No. 16, e3317, 2017.

[21] Q. Jiang, J. Ni, J. Ma, L. Yang, X. Shen, Integrated Authentication and Key Agreement Framework for Vehicular Cloud Computing, *IEEE Network*, Vol. 32, No. 3, pp. 28-35, May/June, 2018.

[22] D. He, S. Zeadally, B. Xu, X. Huang, An Efficient Identity-Based Conditional Privacy-Preserving Authentication Scheme for Vehicular Ad Hoc Networks, *IEEE Transactions on Information Forensics and Security*, Vol. 10, No. 12, pp. 2681-2691, December, 2015.

[23] J. -S. Li, K. -H. Liu, A Lightweight Identity Authentication Protocol for Vehicular Networks, *Telecommunication Systems*, Vol. 53, No. 4, pp. 425-438, August, 2013.

[24] Z. Zhou, H. Zhang, Z. Sun, An Improved Privacy-Aware Handoff Authentication Protocol for VANETs, *Wireless Personal Communications*, Vol. 97, No. 3, pp. 3601-3618, December, 2017.

[25] M. Azees, P. Vijayakumar, L. J. Deboarh, EAAP: Efficient Anonymous Authentication with Conditional Privacy-Preserving Scheme for Vehicular Ad Hoc Networks, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 18, No. 9, pp. 2467-2476, September, 2017.

[26] C. Zhang, R. Lu, P. Ho, A. Chen, A Location Privacy Preserving Authentication Scheme in Vehicular Networks, *2008 IEEE Wireless Communications and Networking Conference*, Las Vegas, USA, 2008, pp. 2543-2548.

[27] K. Masmoudi, H. Afifi, Building Identity-based Security Associations for Provider-provisioned Virtual Private Networks, *Telecommunication Systems*, Vol. 39, No. 3-4, pp. 215-222, December, 2008.

[28] Y. Kim, W. Ren, J. -Y. Jo, Y. Jiang, J. Zheng, SFRIC: A Secure Fast Roaming Scheme in Wireless LAN Using ID-Based Cryptography, *2007 IEEE International Conference on Communications*, Glasgow, Scotland, 2007, pp. 1570-1575.

[29] D. He, D. Wang, Q. Xie, K. Chen, Anonymous Handover Authentication Protocol for Mobile Wireless Networks with Conditional Privacy Preservation, *Science China. Information Sciences*, Vol. 60, No. 5, pp. 1, May, 2017.

[30] S. A. Chaudhry, M. S. Farash, H. Naqvi, S. H. Islam, T. Shon, A Robust and Efficient Privacy Aware Handover Authentication Scheme for Wireless Networks, *Wireless Personal Communications An International Journal*, Vol. 93, No. 2, pp. 311-335, December, 2015.

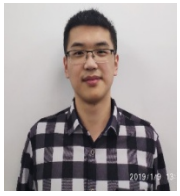
[31] Pairing-based Cryptography Library, http://crypto.stanford.edu/pbc/.

[32] MIRACL Library, http://www.shmus.ie/index.php.

Biographies

Peng Wang received the B.E. degree in industrial automation from China University of Mining and Technology, XuZhou, China, in 1999. He received the M.E. degree in detection technology and automation device from Guilin University of Electronic Technology, Guilin, China, in 2007. He is currently working toward the Ph.D. degree in information a-nd Communication Engineering in Guilin University of Electronic Technology, China. His research interests includes information security protocol and applied cryptography.

Yining Liu is currently a professor in School of Computer and Information Security, Guilin University of Electronic Technology, China. He received the B.S. degree in Applied Mathematics from Information Engineering University, Zhengzhou, China, in 1995, the M.E. in Computer Software and Theory from Huazhong University of Science and Technology, Wuhan, China, in 2003, and the Ph.D. degree in Mathematics from Hubei University, Wuhan, China, in 2007. His research interests include applied cryptography, and data privacy protocol.



Songzhan Lv is now pursuing his M.E. degree in Computer Science in Guilin University of Electronic Technology, Guilin, China. He has received her B.E. degree in electronic and information engineering from University of Zaozhuang, Zaozhuang, China, in 2017. His research focuses on the VANET authentication.