IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Ingenierie des Systemes d'Information
Vol. 24, No. 4, August, 2019, pp. 397-402
Journal homepage: http://iieta.org/journals/isi

A Novel Privacy Protection Protocol for Vehicular Ad Hoc Networks Based on Elliptic Curve Bilinear Mapping

Haijuan Zang[1*], Yan Huang[2], Hongbo Cao[1], Chenchen Li[1]

[1] College of Computer Engineering, Jiangsu University of Technology, Changzhou 213001, China
[2] Jintan Sub-Bureau of Public Security Bureau of Chang Zhou city, Changzhou 213200, China

Corresponding Author Email: zhjuan@jsut.edu.cn

https://doi.org/10.18280/isi.240406

Received: 12 April 2019
Accepted: 2 July 2019

Keywords:
vehicular ad hoc networks (VANETs), conditional privacy protection (CPP), group signature, elliptic curve bilinear mapping

ABSTRACT

The vehicular ad hoc networks (VANETs) face serious privacy threats, due to the numerous vehicles, variable node speeds and network openness. To tackle the threats, this paper proposes a conditional privacy protection (CPP) mechanism based on group signature anonymous authentication and the cryptographic algorithms of bilinear pairings on elliptic curve. Unlike most existing group signature mechanisms, this mechanism can achieve anonymous and non-connectable conditions at the same time, and allow the trust authority (TA) to track the identity of the sender of any controversial message. Finally, simulation results show that the CPP outperformed the group signature-based (GSB) protocol and the human anonymous keys-based (HAB) protocol in verification speed, tacking efficiency and scalability.

1. INTRODUCTION

Vehicle-to-vehicle and vehicle-to-roadside communications architectures coexist in vehicular ad hoc networks (VANETs) to provide active safety for vehicles. The communications are supported by sensors, wireless channels and the vehicle-mounted computing platform. However, the wireless signals in the VANETs are prone to be eavesdropped and maliciously modified. For example, the attackers may alter the key information of vehicles, such as location and license plate number [1, 2]. Therefore, the VANETs security and privacy have become a research hotspot in recent years.

In the VANETs, both the identity and location of a vehicle should be kept private, i.e. non-connectable, undeniable and untraceable. The identity and location privacies are usually realized through anonymity or pseudonym. Many valuable solutions have been presented to achieve the anonymity of vehicle identity, namely, public key encryption, symmetric key, group signature (e.g. ring signature, elliptic encryption, hyperbolic mapping) and identity-based encryption. Hubaux et al. [3] were the first to use a set of public key pairs and certificates to realize VANETs pseudonym communication, and introduce electronic license plate as a special identifier of vehicles to satisfy special demands. Raya and Hubaux [4] proposed the human anonymous keys-based (HAB) protocol to satisfy non-connectivity based on anonymous certificate. Relying on the traditional public key infrastructure (PKI), the HAB protocol hide the real identity of the vehicle with numerous anonymous certificates issued by the certification center, and prevents illegal tracking through periodic replacement of certificates.

Group signature is the most promising technology for conditional privacy protection. Drawing on group signature and pseudonym communication mechanism, Erritali et al. [5] designed a geographical location routing plan to protect routing security and location privacy. This practical plan can effectively protect the information of node geographical location and user privacy, and withstand various active and passive attacks. To achieve fast authentication, Song et al. [6] put forward the vehicle authentication plan of ZL 06 group signature, which enjoys strong security, high efficiency and short signature length. Zhang et al. [7] added the signature public key into the signature parameters, such that the signature identity can be decrypted and verified without generating any redundant data.

The elliptic curve cryptography (ECC) stands out for its ability to provide a high level of security with small keys. Lin et al. [8] proposed a group signature-based (GSB) protocol that integrates bilinear mapping into group signature. The GSB protocol does not need to store many kana keys or certificate in the vehicle unit, but only one private key and group public key. In addition, the certificate revocation list is very short and easy to update [9]. Liu et al. [10] coupled bilinear mapping with elliptic curve, creating a secure and effective group signature plan with road side unit (RSU) as group administrator. However, there are many problems with the security and privacy plans based on group signature: the group members are difficult to be deleted safely and effectively, and the signature algorithm is highly complex, to name but a few [11-13].

In general, the above studies face two common problems. First, the group administrator needs to manage numerous private keys, for each group member is allocated with a private key and member certificate. This calls for an effective algorithm to decrypt the signature. Second, the privacy protection of group members is conditional. If a member has caused a fault or harmed the VANETs security, its privacy should be exposed temporarily for investigation by the authorities. Our research aims to solve these two problems effectively. This paper studies the security and privacy protection mechanism of the VANETs, and then proposes a conditional privacy protection (CPP) mechanism based on

group signature anonymous authentication. Under this mechanism, the anonymous and non-connectable conditions can be achieved at the same time; the trust authority can track the identity of the sender of any controversial message. Neither function is possible in most existing group signature mechanisms.

2. CONDITIONAL PRIVACY PROTECTION BASED ON ELLIPTIC CURVE BILINEAR MAPPING

2.1 Elliptic curve bilinear mapping algorithm

2.1.1 Elliptic curve cryptography

The ECC is a new generation public key cryptosystem with high security, small key and good flexibility [14, 15]. The elliptic curve password achieves the same level of security with a shorter key than the Rivest-Shamir-Adleman (RSA) password. The previous research has shown that the safety of a 1,024bit RSA password can be realized by a 160bit elliptic curve password.

In the ECC, the bilinear relationship between groups can be defined as follows:

It is assumed that the elliptic curve E_k defined on finite field K satisfies the non-homogeneous Weierstrass equation [16]:

$\mathrm{E(K)}: y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6.$

where, a_1, a_2, a_3, a_4 and $a_6 \in K$. Let $\Delta \neq 0$ be the discriminant of E_1. Suppose E_1 and E_2 are two elliptic curves defined on finite field K. If there exist u, r s and t that can transform equation E_1 into equation E_2 by:

$(\mathrm{x}, \mathrm{y}) \rightarrow (u^2x + r, u^2y + u^2xsx + t).$

Then E_1 and E_2 are isomorphic. The isomorphism can be described as:

$E_1/K \cong E_2/K.$

Let L be the extension field of K and O be the point at infinity. Then, the L set of rational points on E can be expressed as:

$\mathrm{E(L)} = \{(\mathrm{x}, \mathrm{y}) \in \mathrm{L} \times \mathrm{L}: y^2 + a_1xy + a_3y - x^3 - a_2x^2 - a_4x - a_6 = 0\} \cup \{o\}.$

If the characteristic p of finite field K is greater than 3, then the equation can be transformed into $y^2 = x^3 + ax + b$ through coordinates conversion, where a and b $\in$ K and $4a^3 + 27b^2 \neq 0$.

2.1.2 Bilinear mapping

Let G and G' be two additive cyclic groups, and G_T be a multiplicative cyclic group. Suppose these three groups have the same order q, that is, $|\mathrm{G}| = |\mathrm{G}'| = |G_T| = \mathrm{q}$. It is assumed that P is the producer of G, P′ is the producer of G′, and Ψ is the isomorphism function from G′ to G, i.e. $\Psi(\mathrm{P}') = \mathrm{P}$.

Then, an effective bilinear map can be established as: $e: \mathrm{G} \times \mathrm{G}' \rightarrow G_T$, where e has the following features:

(1) Bi-linearity: for all the $P_1 \in G$, $Q_1 \in G'$and $\mathrm{a}, \mathrm{b} \in Z_q^*$, there exists $e(aP_1, bQ_1) = e(P_1, Q_1)^{ab}$.

(2) Computability: for any $P_1 \in G$ and $Q_1 \in G'$, there is an effective algorithm to compute $e(P_1, Q_1)$.

Next, the bilinear map e can be constructed through improved Weil pairing or Tate pairing on the elliptic curve. For example, the Tate pair on Miyaji-Nakabayashi-Takano (MNT) curve provides effective performance, where G ≠ G′. The one-way isomorphism Ψ can be implemented with the trail map, when q is a prime number of 160bit. Then, G can be expressed as a prime number of 16bit [17, 18]. According to the construction of e, the discrete logarithm problem of G can reach the security level of 80bit [19].

2.2 The design of CPP

Figure 1 shows a typical three-layer VANET consisting of onboard units (OBUs), roadside units (RSUs) and the trusted authority (TA) [20]. The CPP protocol includes five steps: system initialization, establishment and update of the RSU neighbor list, generation of OBU key, signature and verification of messages, as well as tracking and certificate revocation of the sender of controversial message.

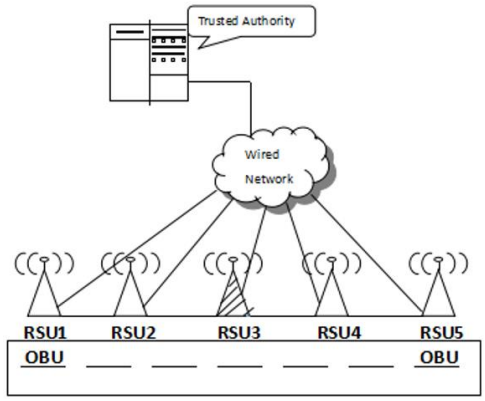

Figure 1. Structure of a three-layer VANET

2.2.1 System initialization

Before using the system initialization algorithm, the system parameters should be generated in the following steps:

(1) Under the given safety parameter K, run the bilinear parameter generator Gen with TA to produce a set of bilinear parameters (q, G, G′, G_T, e, P, P').

(2) Select random numbers u and v $\in Z_q^*$ with TA as the self-determined master key, and compute $\mathrm{U} = \mathrm{uP} \in \mathrm{G}$, $\mathrm{U}' = \mathrm{uP}' \in \mathrm{G}'$ and $\mathrm{V} = \mathrm{vP} \in \mathrm{G}$.

(3) Select the hash functions in two cryptographies f and g with TA, where$f, g: \{0,1\}^* \rightarrow Z_q^*$; Select a secure symmetric encryption algorithm $Enc_k()$, where k is the key of the algorithm.

(4) Output the system parameters as follows:

(q, G, G′, G_T, e, P, P', U, V, U', f, g, $Enc_k()$).

Then, the system parameters and master keys (u, v) are inputted to the system initialization algorithm, such that the TA can extract the private key sk_i by inputting an ID ID_i.

2.2.2 Establishment and update of the RSU neighbor list

In the VANET, each RSU is fixed and wired to the adjacent RSUs. For each RSU, the adjacent RSUs in various direction can be collected into its neighbor list. This table contains the location information of all adjacent RSUs (e.g. the neighbor list $\{L_1, L_3\}$of RSU2.

The RSUs must be verified periodically by the TA to prevent possible attacks. Any RSU failing to pass the verification will be considered as undermined, and be reported to its adjacent RSUs. Then, the location information of this RSU will be removed from the neighbor list of each adjacent RSU. After the removal, the updated neighbor list will be broadcasted to passing vehicles. Upon receiving the broadcast, the OBU will replace the original neighbor list in the storage unit with the updated version.

2.2.3 Generation of OBU key

Traditionally, each OBU must reserve a large space to store the member revocation list. In the CPP protocol, however, there is no need to make such a reservation. When the OBU reaches the coverage of the next trusted RSU (L_j) in the neighbor list, it sends a request for anonymous key certificate to that RSU. To complete the communication of the request in the coverage, the vehicle speed and density should both be restricted. After receiving the request, the RSU will firstly check if the OBU is in the latest member revocation list of the TA. If yes, the RSU will reject the request.

The OBU has a temporary private key$x$ and an anonymous public key $(Y, cert_i)$, which is consistent with x. In our protocol, the request response includes two communications between the OBU and the RSU. The two communications were analyzed below to judge if they may face security risks.

In the first communication, the OBU sends R_1 and C to the RSU. R_1 contains no private information of the OBU. The private information of OBU in C is symmetrically encrypted with R_2. The calculation of R_2 requires that the random parameter r_1 is generated by the OBU and has never been sent, or that the private key of RSU satisfies $B_j = \frac{1}{h(L_j)+u}P$, where h is a random oracle. Hence, it is very difficult for the attacker to calculate R_2 without knowing r_1 and h. In other words, it is unlikely to obtain the private information of the vehicle by decoding C.

2.2.4 Signature and verification of messages

After requesting the temporary key pair (x, Y) with certificate, the OBU will sign and send the message within the valid period.

Step 1. Message formatting

Table 1. Format of security message field

Group ID	Load	Signature	Anonymous key	Temporary message
2bits	100bits	40bits	26bits	121bits

As shown in Table 1, the security message field covers five parts. The first part is the group ID, which identifies the group of the vehicle and represents the identity of the TA. The second part is the message load (100bits), including the current location, time, driving direction, speed and deceleration/acceleration of the vehicle or current traffic conditions. The third part is the signature σ_M of the message load. The fourth part is the temporary key pair (x, Y) of the OBU. The last part is the certificate of the temporary key.

Step 2. Signing and sending the message

(1) Select a random number r ∈ Z_q^* to calculate R = rP ∈ G and $s_r = r + x \cdot h(M, R) mod\ q$, and then determine the signature $\sigma_M = (R, s_r)$.

(2) Create a message Msg : $[ID_{TA}||M||\sigma_M||Y||Cert_i]$ according to the format of the security message and send it out.

2.2.5 Tracking and certificate revocation of the sender of controversial message

Let $[ID_{TA}||M||\sigma_M||Y||Cert_i]$ be the controversial message Msg. Then, the following algorithm is adopted to track the OBU that sends this message.

Step 1. The TA quickly locates the RSU that issues the certificate $Cert_i$ in Msg with the master key.

(1) The TA obtains (T_U, T_v) from the certificate $Cert_i$;

(2) The TA calculates uA_j with the master key (u, v):

uA_j: $uT_V - vT_U = uA_j + u\alpha V - v\alpha U = uA_j + \alpha uvP - \alpha uvP = uA_j$.

(3) Taking uA_j as a condition in the tracking list, the TA searches for the (ID_j, uA_j) input during registration. In this way, the identity ID_j of the RSU can be quickly determined through processing $Cert_i$.

(4) The TA sends a search request to the RSU.

Step 2. Upon receiving the TA's request, the RSU searches the local certificate list, retrieves the pseudonym RID_j of the OBU of Msg sender, and returns the RID_j to the TA.

(1) The RSU obtains the anonymous public key Y from the message Msg.

(2) Taking Y as a condition, the RSU searches the local certificate list and recovers the (RID_i, Y, R_2, σ_1) stored during the OBU key request.

(3) The RSU returns the pseudonym RID_j of the OBU and the signature (R_2, σ_1) of Y to the TA.

Step 3. The TA restores the real identity of OBU from the pseudonym RID_j.

(1) The TA decrypts RI$D_i = Enc_v(ID_i)$ with the master key v and recovers the real identity ID_j from RID_j.

(2) The TA validates the signature (R_2, σ_1) of Y, which provides evidence of non-repudiation of the OBU request key.

(3) The TA broadcasts the pseudonym RID_j to all RSUs, and each RSU adds RID_j into the local revocation list. Then, the OBU can no longer apply the temporary key from the RSU. In this way, the problem of certificate revocation in the VANETs can be solved effectively.

2.3 Security proof

2.3.1 Signature security

In the random oracle model, the signature $\sigma_M = (R, s_r)$ can resist the existential forgery of the adaptive chosen-ciphertext attack. The signature safety was analyzed as follows: Suppose an enemy A, taking M and Y as inputs, outputs an existential forgery at a nonnegligible probability in the polynomial time. Let $h()$ be a random oracle. Then, the enemy A can produce two forgeries, namely, $\sigma_M = (R, s_r)$ and $\sigma'_M = (R, s'_r)$, of the same message M, using the same parameters and different hash functions, according to the forking lemma.

Note that R = rP, $s_r = r + x \cdot h(M, R) mod\ q$ and $s'_r = r + x \cdot h'(M, R) mod\ q$. Then, it can be computed that $s_r - s'_r = x\left(h(M, R) - h'^{(M,R)}\right) mod\ q$ $x = (s_r - s'_r)(h(M, R) - h'(M, R))^{-1} mod\ q$. The computed result contradicts the discrete logarithm assumption, indicating that the signature σ_M is unforgeable. In other words, the signature σ_M can resist attacks of false and fake messages.

2.3.2 CPP security

The security of the CPP was analyzed according to the requirements on group signature and our design objectives.

(1) Unforgeability: Only group members can sign a message on behalf of the group. To produce a legal group signature in line with the CPP, the registered legal group members must compute the private key S_i based on the system parameters and the TA master key. In other words, non-group members cannot sign any message on behalf of the group.

(2) Anti-framing: Neither the group administrator nor any group member can sign any message in the name of any other member. The TA, as the group administrator, is a trusted agent

that will not pretend to be others in message signing. In addition, the private key S_i used for signature is a combination of the TA's master key and the identity RID_i of OBU.

(3) Coalition resistance: The group members cannot conspire to produce an untraceable signature message. In the CPP, the TA can trace the real identity of the OBU, using the certificate in the protocol-compliant security message. This certificate is allocated by the trusted RSU, and not processable by the OBU.

(4) Traceability: The TA can find the signer of controversial message.

(5) Revocability of illegal members: The TA can revoke the membership of the sender of controversial message. The pseudonym of the sender will be sent to the RSU that assists the TA in the tracking process, and also added to the local revocation list. In this way, the malicious OBU will be excluded from the VANET.

3. COMPARISON WITH RELEVANT PROTOCOLS

To verify its performance, our CPP protocol was simulated on Matlab. The performance of this protocol was analyzed in four aspects: OBU storage overhead, request for temporary anonymity key, time cost of message verification and tracking complexity.

3.1 OBU storage overhead

This subsection compares the OBU storage overhead of the CPP, the GSB and the HAB [4, 8]. Both the GSB and the HAB have the function of conditional privacy protection.

In the CPP, each OBU needs to store three items: the private key S_i assigned by TA at registration, the temporary anonymous key pair with certificates assigned by the RSU, and the newly updated RSU trusted neighbor list. It is assumed that each item occupies a storage unit. Without needing to store the member revocation list, the OBU has only three storage units for storage overhead, i.e. S_{My}=3.

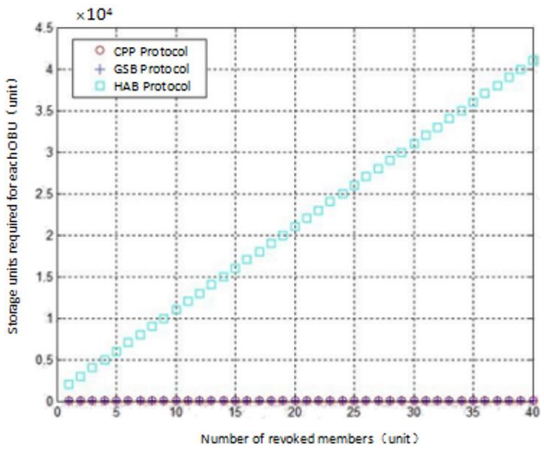

Figure 2. Comparison of OBU storage overhead of the three protocols

In the HAB, each OBU stores not only its own N_{okey} anonymous key pairs, but also the anonymous public key and the certificate of public key for all members in the revocation list. If n OBUs are revoked, the total storage overhead of the OBU in the HAB can be expressed as $S_{HAB}=(n+1)\times N_{okey}$. The N_{okey} must be a large number, because OBU needs to change the anonymous key frequently. If N_{okey} is 10^4, the S_{HAB} will be equal to $(n+1)\times 10^4$.

In the GSB, each OBU needs to store a private key assigned by the TA, plus n revoked public keys in the revocation list. Therefore, the total storage overhead of OBU is $S_{GSB}=n+1$.

Figure 2 shows how the OBU storage overhead of each protocol varies with the number n of revoked members. It can be seen that the OBU storage overhead of the CPP fluctuated less violently than that of the HAB or the GSB, with the growing number of revoked members. The results indicate that our protocol is superior to the HAB and the GSB in OBU storage overhead, an evidence of its good scalability of member revocation list.

3.2 Request for temporary anonymous key

Each key request must be responded within the effective coverage of the selected RSU, and the temporary key be generated under the exact time limit. This calls for constraints on the vehicle speed and density. Thus, the time cost of the response to the key request can reflect the efficiency of the protocol. In the CPP, this time cost mainly arises from two operations, namely, point multiplication and mapping. Table 2 lists the time cost of the CPP measured by the MNT curve, with the embedding level k of 6 and q of 160bit. The two operations were executed on a Pentium IV CPU (3.0GHz). According to the results in the table, T_k=13T_{pmul}+tT_{pair}=34.8ms.

Table 2. The time cost of the response to the key request

Symbol	Description	Time cost
T_{pmul}	Time for each point multiplication	0.6ms
T_{pair}	The time for each mapping	4.5ms

The following assumptions were made to simulate the actual scenario:

(1) According to the highway speed limit and vehicle performance, the average speed v was set to 10~40m/s. Besides, the effective coverage R_{range} of each RSU is assumed to be 300m.

(2) Considering the actual situation of China's highways, the simulated highway is assumed to be a two-way four-lane highway. In the effective coverage of the RSU, the vehicle density d is assumed to vary from 100 to 300. Since the OBU needs to request for a key in the coverage of a new RSU, the total number of key requests S_{req} of all OBUs in the effective coverage of the RSU S_{req} must be equal to d.

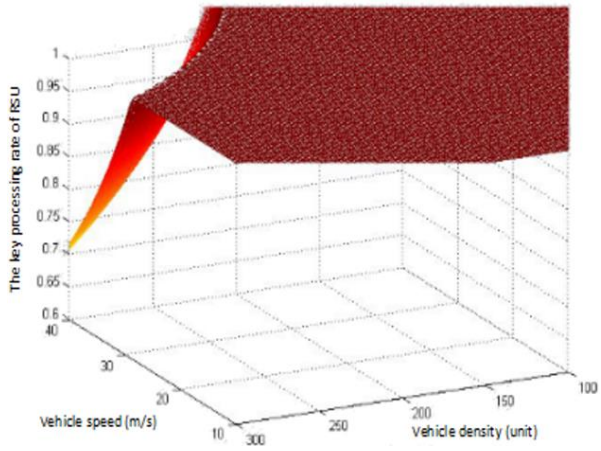

Figure 3. Key processing rate of the RSU

Figure 3 shows the key processing rate S_{ratio} of the RSU with the changes of v and d, where $100 \le d \le 300$ and $10 \le v \le 40$. It is easy to infer that the RSU effectively handled the key requests in most cases, for the $S_{ratio}=1$. With the growth in v and d, the S_{ratio} gradually declined, and the decrease was over 70% at the most. Hence, the CPP is both feasible and effective.

3.3 Time cost of message verification

In the CPP, it takes 11 point multiplications and 3 mapping operations to verify a message. Hence, the time cost of message verification is:

$T_{My}=11T_{pmul}+3T_{pair}=11\times0.6+3\times4.5=20.1$ms.

In the GSB, the time cost T_{GSB} of message verification depends on the number of revoked OBUs in the member revocation list. According to the literature, $T_{GSB}=6T_{pmul}+(3+2n)T_{pair}=6\times0.6+(3+2n)\times4.5=17.1+9n$(ms). Thus, the time overhead ratio of the CPP to the GSB can be expressed as $T=\frac{T_{GSB}}{T_{My}}$.

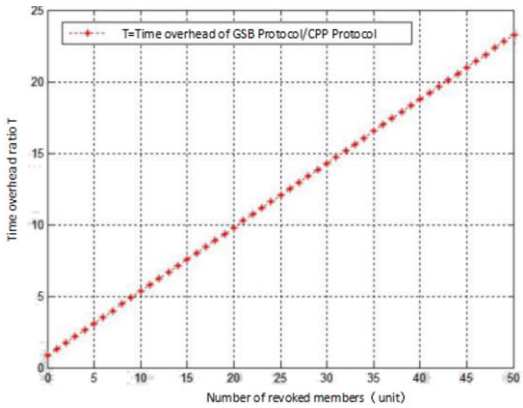

Figure 4. Time overhead ratio

Figure 4 shows how the time overhead ratio varies with the number n of revoked members. It can be seen that, with the increase of n, the T value increased significantly. Hence, the CPP outperformed the GSB in the cost of message verification. The more the number of revoked OBUs in the member revocation list, the greater the advantage of the CPP.

3.4 Tacking complexity

The tracking complexity of the CPP hinges on the computing complexity of the TA. The linear and binary search algorithms were adopted in the CPP, the HAB and the GSB to compare the computing complexities of the three protocols. The symbols of computing complexity are described in Table 3. The tracking complexities are listed in Table 4. It can be seen that the CPP achieved better results than the HAB and the GSB in linear search, and comparable effects with the latter in binary search.

Table 3. Symbol description

Symbol	Description	Magnitudes
N_{rsu}	The number of RSUs in the system	10^4
N_{rkey}	The number of anonymous keys processed by an RSU in a cycle	10^3
N_{obu}	The number of OBUs in the system	10^7
N_{okey}	The number of anonymous keys of an OBU	10^4

Table 4. The tracking complexities of the three protocols

Protocol	Linear search	Binary search
CPP	$O(N_{rsu}+N_{rkey})$	$O(\log(N_{rsu}\cdot N_{rkey}))$
HAB	$O(N_{obu}\cdot N_{okey})$	$O(\log(N_{obu}\cdot N_{okey}))$
GSB	$O(N_{obu})$	$O(\log(N_{obu}))$

4. CONCLUSIONS

In recent years, the VANETs have undergone rapid structural changes to enhance the real-time communication between nodes. This calls for a quick and effective method for security authentication of vehicles in the network. This paper develops an anonymous authentication mechanism for data security and privacy protection of VANETs. The mechanism realizes anonymous authentication based on the features of elliptic curve cryptography, such that no identity information will be leaked in wireless channels. Simulation results show that the CPP achieved faster message verification, consumed fewer space, and realized lower computing complexity than common protocols, while ensuring the privacy and traceability of messages. The future research will try to realize secure communication in congested environment, using mobile RSUs.

ACKNOWLEDGEMENT

This paper is supported by the National Natural Science Foundation of China (No. 61672270); Supported by the National Natural Science Foundation of China (No. 61702236); Supported by Changzhou Sci & Tech Program, (No. CJ20179027).

REFERENCES

[1] Isaac, J.T., Zeadally, S., Camara, J.S. (2010). Security attacks and solutions for vehicular ad hoc networks. IET Communications, 4(7): 894-903. http://dx.doi.org/10.1049/iet-com.2009.0191

[2] Mejri, M.N., Ben-Othman, J., Hamdi, M. (2014). Survey on VANET security challenges and possible cryptographic solutions. Vehicular Communications, 1(2): 53-66. http://dx.doi.org/10.1016/j.vehcom.2014.05.001

[3] Hubaux, J.P., Capkun, S., Luo, J. (2004). The security and privacy of smart vehicles. IEEE Security and Privacy Magazine, 2(3): 49-55. http://dx.doi.org/10.1109/MSP.2004.26

[4] Raya, M., Hubaux, J.P. (2007). Securing vehicular ad hoc networks. Journal of Computer Security, 15(1): 39-68. http://dx.doi.org/10.3233/JCS-2007-15103

[5] Erritali, M., Ouahidi, B., Bourget, D. (2013). Secured geographic routing protocol for vehicular ad hoc networks VANETs. Networked Systems, 7853: 311-315. http://dx.doi.org/10.1007/978-3-642-40148-0_30

[6] Song, J.H., Wong, V.W., Leung, V.C. (2010). Wireless location privacy protection in vehicular ad-hoc networks. Mobile Networks and Applications, 15(1): 160-171. http://dx.doi.org/10.1007/s11036-009-0167-4

[7] Zhang, J.Z., Zou, J.C., Wang, Y.M. (2010). An improved group signature scheme. Trust, Privacy, and Security in Digital Business, 185-194. http://dx.doi.org/10.1007/11537878_19

[8] Lin, X.D., Sun, X.T., Ho, P.H., Shen, X.M. (2007). GSIS: A secure and privacy-preserving protocol for vehicular communications. IEEE Trans. Vehicular Technology, 56(6): 3442-3456. http://dx.doi.org/10.1109/TVT.2007.906878

[9] Lu, Z.J., Liu, W.C., Wang, Q., Qu, G., Liu, Z.L. (2018). A privacy-preserving trust model based on blockchain for VANETs. IEEE Access, 6: 45655-45664. http://dx.doi.org/10.1109/ACCESS.2018.2864189

[10] Liu, L.X., Liu, M.Y., Shen, X.L. (2012). Research on conditional privacy-preserving protocol in VANET. Application Research of Computers, 29(2): 683-686. http://dx.doi.org/10.3969/j.issn.1001-3695.2012.02.075

[11] Bayat, M., Barmshoory, M., Rahimi, M., Aref, M.R. (2015). A secure authentication scheme for VANETs with batch verification. Wireless Networks, 21(5): 1733-1743. http://dx.doi.org/10.1007/s11276-014-0881-0

[12] Chhatwal, S.S., Sharma, M. (2015). Detection of impersonation attack in VANETs using BUCK Filter and VANET Content Fragile Watermarking (VCFW). 2015 International Conference on Computer Communication and Informatics (ICCCI), pp. 1-5. http://dx.doi.org/10.1109/ICCCI.2015.7218093

[13] Nam, J., Choo, K.K., Han, S., Kim, M., Paik, J., Won, D. (2015). Efficient and anonymous two-factor user authentication in wireless sensor networks: achieving user anonymity with lightweight sensor computation. Plos One, 10(4): e0116709. http://dx.doi.org/10.1371/journal.pone.0116709

[14] Miller, V.S. (1986). Use of elliptic curves in cryptography. Advances in Cryptology - CRYPTO ’85 Proceedings, pp. 417-426. https://doi.org/10.1007/3-540-39799-X_31

[15] Koblitz, N. (1987). Elliptic curve cryptosystems. Mathematics of Computation, 48(177): 203-209. http://dx.doi.org/10.1090/S0025-5718-1987-0866109-5

[16] Paar, C., Pelzl, J. (2012). Understanding Cryptography: A Textbook for Students and Practitioners. Springer Publishing Company, Incorporated. http://dx.doi.org/10.1007/978-3-642-04101-3

[17] Blake, I.F., Murty, V.K., Xu, G. (2006). Refinements of Miller's algorithm for computing the Weil/Tate pairing. Journal of Algorithms, 58(2): 134-149. http://dx.doi.org/10.1016/j.jalgor.2005.01.009

[18] Scott, M., Barreto, P.S.L.M. (2006). Generating more MNT elliptic curves. Designs Codes & Cryptography, 38(2): 209-217. http://dx.doi.org/10.1007/s10623-005-0538-1

[19] Bellare, M., Kohno, T., Shoup, V. (2006). Stateful public-key cryptosystems: How to encrypt with one 160-bit exponentiation. Proceedings of the 13th ACM conference on Computer and communications security - CCS '06, pp. 380-389. http://dx.doi.org/10.1145/1180405.1180452

[20] Yang, T., Wang, Y.K., Ge, Y.F., Lin, Y. (2015). An efficient and accountable privacy-preserving protocol for VANET. Computer Engineering, 41(11): 186-189. http://dx.doi.org/10.3969/j.issn.1000-3428.2015.11.032