An Efficient Authentication and Key Agreement Scheme for CAV Internal Applications

Yang Li[1], Qingyang Zhang[2,3,4,5(✉)], Wenwen Cao[2,3,4,5], Jie Cui[2,3,4,5], and Hong Zhong[2,3,4,5]

[1] Anhui Province Key Laboratory of Cyberspace Security Situation Awareness and Evaluation, Hefei 230037, China
[2] School of Computer Science and Technology, Anhui University, Hefei 230039, China
qyzhang@ahu.edu.cn
[3] Anhui Engineering Laboratory of IoT Security Technologies, Anhui University, Hefei 230039, China
[4] Institute of Physical Science and Information Technology, Anhui University, Hefei 230039, China
[5] Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, Anhui University, Hefei 230039, China

Abstract. The data of applications in connected and autonomous vehicles are important, which is usually collected by service providers to improve their services, such as object detection model. But, wireless communication is susceptible to various kinds of attacks. Thus, the data of the application module needs to be securely shared to the corresponding service provider. However, current schemes are with limited performance while a service provider collects multiple application data at the same time. By adopting signcryption and chaotic map, an efficient authentication and key agreement scheme is proposed, while batch authentication is achieved for efficient message authentication of multiple applications, and the efficient revocation is realized based on Chinese remainder theorem under the assistance of trusted execution environment supported vehicle computing/communication unit. The formal security proof shows that the scheme is secure under the random oracle model, and the experiment results shows that the scheme is more efficient than related schemes and can meet the requirements of CAV.

Keywords: Connected and autonomous vehicles · Signcryption · Chaotic map · Chinese remainder theorem

1 Introduction

The number of vehicles has soared in the past few years, creating new problems for transportation, such as traffic jams and accidents. As a potential solution, connected and autonomous vehicles (CAVs) can make real-time decisions based

H. Gao et al. (Eds.): CollaborateCom 2023, LNICST 562, pp. 414–434, 2024.
https://doi.org/10.1007/978-3-031-54528-3_23

on the surrounding environment so as to control the secure driving of vehicles [1,2]. Typically, the CAV consists of five parts: positioning, perception, planning, vehicle control, and system management [3,4]. The positioning system can identify the vehicle's real-time position on the map. The perception system could identify surrounding objects, such as other surrounding vehicles, traffic signals, and surrounding obstacles. The planning system inputs the data from the perception system and positioning system. It determines the driving path and specific driving behaviors, such as lane changes. The control system indicated by the planning system could convert the control actions into vehicles, such as steering. The management system supervises the operation status of all systems and provides the human-computer interface. The CAV deployed with a large number of sensors can provide accurate perception data. These data can be processed inside the vehicle to enable the vehicle to make real-time decisions. These data are also very important to the SP. For example, a CAV can share data with traffic authorities and service providers (SP) to improve congestion across the entire traffic network [5]. In addition, an SP needs to collect a large amount of reliable data to train the artificial intelligence (AI) model to improve their services [6]. AI model requires a large amount of reliable data as input. Models trained on large amounts of data are more effective.

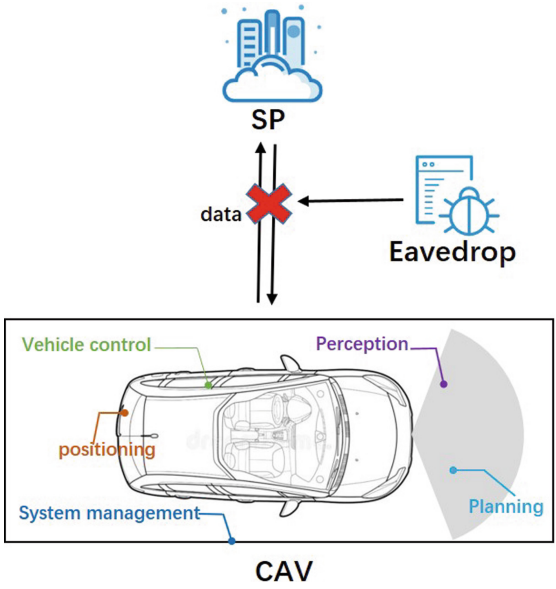

Fig. 1. Data leakage model for CAV data transmission

The data transmission between SP and CAV applications[1] is transmitted via the insecure public channel, which poses threats to privacy and security

[1] The applications here can also be in-vehicle modules connected by CAN bus.

[7–9]. The network communication may be subject to various attacks, such as impersonation attack, modification attack [10–13]. A malicious attack can cause the SP to collect incorrect data, which will lead to incorrect analytical decisions. Such attacks, as shown in Fig. 1, are fatal to the communication security of CAV. Therefore, it is necessary to take the authentication and key agreement (AKA) protocol [14,15] to protect the security of communication.

These applications share the same computing unit, i.e., vehicle computing/communication unit (VCU), and most of the computing workloads are occupied by autonomous driving applications. For driving safety, the additional algorithms, protocols, and schemes, such as the security scheme designed in this paper, should be as efficient as possible. In this case, some schemes are proposed. For example, some schemes adopt signcryption [16,17] and chaotic map [18,19] to implement authentication. They are based on a single AKA between the user and the server. However, an SP might collect a mass of data from multiple applications at the same time in one vehicle or an application for a while. Hence, these schemes need multiple AKA protocols or protocol executions are required, which is inefficient for CAVs.

In addition, anonymity makes sense for some scenarios, such as collecting data for AI model training. Adopting real identities may pose security and privacy threats for the participants [20]. Moreover, the efficient member management and key update of these applications also require to be considered.

Nowadays, many computing units have a trusted execution environment (TEE), which could isolate the targeted application from other applications. In this case, to address the aforementioned issues, we propose an efficient authentication and key agreement scheme for CAV internal applications with a TEE-supported VCU. The proposed scheme is adopted by the signcryption and chaotic map to realize security data transmission. On this basis, the scheme can generate signcryption to guarantee the security transmission of the application data. The SP can achieve efficient batch message authentication. Moreover, it could manage keys in one CAV based on the Chinese remainder theorem (CRT) with efficient revocation for the compromised application module, and the key can be updated periodically or when a compromised application module is detected.

- The scheme can realize key agreement between multiple applications and corresponding SP. On this basis, applications generate signcryption to send data to SP securely. The SP can verify and decrypt the message.
- When a large number of messages are received, batch message authentication can be realized. Benefiting from CRT, this scheme can realize efficient key updates. Moreover, for compromised applications, VCU can achieve efficient revocation.
- The related computing and communication costs indicate that the proposed scheme can achieve more efficient authentication. Under the random oracle model, the unforgeability of our scheme has been proved formally.

We will review the related works in Sect. 2 and provide some short preliminaries on the technologies we used, such as signcryption, chaotic maps, and CRT,

as well as the system model of the proposed scheme in Sect. 3. Then, we introduce the proposed scheme in detail in Sect. 4. After that, we prove the security of the proposed scheme. In Sect. 6, the performance of the proposed scheme is evaluated. Finally, we conclude this work.

2 Related Works

In this section, we describe the related schemes of signcryption for the Internet of Things (IoT), and the related schemes of the AKA protocols based on the chaotic map will be illustrated. The CRT scheme will be described. The related comparison is shown in Table 1.

Ting *et al.* proposed the signcryption scheme [16] to achieve secure communication for an unsecured network in wireless sensor networks. This scheme proposed heterogeneous online/offline signcryption to achieve confidentiality, integrity, and unforgeability based on computational Diffie-Hellman and elliptic curve discrete logarithm assumption. The key agreement may not be optimal, and its computational overhead can be further reduced. At the same time, for higher security, key update and compromised sensors revocation can also be further considered.

In order to access real-time information from the IoT devices, Mandal *et al.* proposed a scheme [17] to achieve authorization, authentication, and revocation for participants, which is based on a three-factor certificateless-signcryption-based to achieve secure access between the user and smart devices with the help of the gateway. The computation and communication costs of this scheme are low. This scheme takes into account the revocation of users and the addition of smart devices. But for multiple smart devices, users need to conduct multiple key agreements. Moreover, the key in this scheme may be further considered for updating.

To achieve authentication and key agreement, Roy *et al.* proposed a provably secure three-factor anonymous authentication protocol [18] with fuzzy extractor crowdsourced IoT, in which the chaotic maps have been used to ensure session-key security. But the protocol is vulnerable to offline guessing, key compromise, and user impersonation attacks.

To achieve secure access and communication mechanism for various applications, Qiu *et al.* proposed a protocol [19] based on extended chaotic maps, which could achieve secure communication between the user and server. The security of this scheme is based on the Computational Diffie-Hellman (CDH) problem. The overall calculation and communication overhead of the scheme is low. However, for the authentication between multiple IoT users and servers, the authentication overhead will increase linearly.

In addition, Cui *et al.* proposed an authentication and key agreement scheme using chaotic mapping and three-factor mutual authentication [21], and the experimental results show that the scheme is suitable for the connected vehicle environment. Similarly, this scheme does not consider multi-user scenarios. In order to achieve a faster network service for users while ensuring confidentiality

and authentication of data transmission, Xu *et al.* proposed a certificateless signcryption mechanism [22]. However, the scheme includes time-consuming bilinear pairing operations that can reduce the efficiency of authentication.

Table 1. Pros and Cons of Various Schemes

Schemes	Main tech	Advantages	Disadvantages
[16]	ECDLP	Online/offline signcryption	Fail to achieve key update and revocation
[22]	Signcryption	Resist various common attacks	Time-consuming pairing operations
[17]	Three-factor, signcryption	authorization, authentication, and revocation	Fail to achieve key update
[18]	Three-factor, chaotic maps	Low costs, achieve revocation	Cannot resist off-line guessing attack, impersonation attack
[19]	Three-factor, chaotic maps	Revocation and key update	Fail to take into account multiple users
[21]	Three-factor, chaotic maps	Revocation	Fail to take into account multiple users

3 Preliminaries and Background

We will introduce the related preliminaries, i.e., chaotic maps, and CRT, in this section. And then, the system model and threat model are described in detail, as well as the security requirements of the proposed scheme.

3.1 Preliminaries

Chaotic Maps. Assuming that $n, x \in_R Z_q^*$ are positive integer, $T_n(x)$ represents the Chebyshev polynomial [23–25] and is expressed as $T_n(x) = cos(n \cos^{-1}(x))$. The recurrence relation of Chebyshev polynomial is as follows:

$$T_n(x) = \begin{cases} 1, & if\ n = 0 \\ x, & if\ n = 1 \\ 2xT_{n-1}(x) - T_{n-2}(x), & if\ n \geq 2 \end{cases}$$

Chinese Remainder Theorem. We start with the construction of an integer a whose modulus set is k. Let a be the integer to be constructed, and X the remainder of a_i modulo k_i, i.e.,

$$X \equiv a_i \bmod k_i, where\ i = 1, ..., n$$

That is, all remainders a_i modulo k_i have the same value, named common remainder X. According to traditional CRT [26–28], X can be uniquely reconstructed as

$$X = a_1 + a_2 + ... + a_n = \sum_{i=1}^{n} a_i \beta_i \gamma_i (\mathrm{mod} \zeta)$$

where $\zeta = k_1 k_2 ... k_n, \beta_i = \frac{\zeta}{k_i}, \beta_i \gamma_i \equiv 1 \bmod k_i$.

3.2 System Model

Figure 2 demonstrates the system model of the proposed scheme. The main entity includes Vehicle Computing/Communication Unit, Service Provider, and Application.

Vehicle Computing/Communication Unit (VCU): VCU is a trusted entity deployed inside the vehicle. The credibility of the VCU can be achieved through an embedded trusted execution environment [29,30] such as Intel SGX, AMD SEV, or ARM TrustZone.

Service Provider (SP): SP can provide personalized services for vehicle internal applications, including assisting in providing vehicle peripheral information and providing entertainment services.

Application: The internal applications can obtain real-time information of the vehicle. This information will be further securely transmitted to the corresponding application provider for further processing. For example, the surrounding road condition information is reported to the traffic authority, and the traffic congestion has been improved.

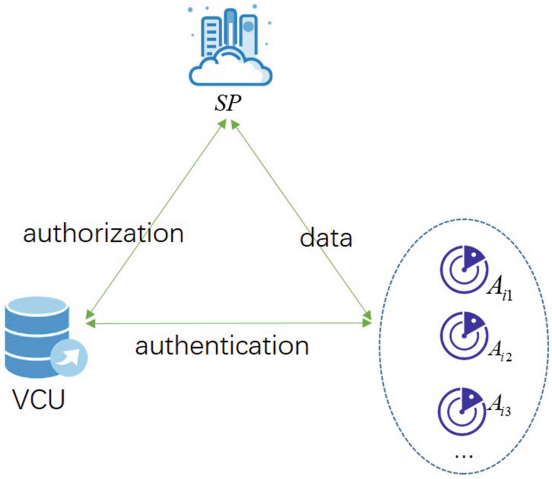

Fig. 2. System model of the proposed scheme

3.3 Threat Model

In our scenario, it is assumed that two participants, i.e., CAV and service provider, communicate through the public channel. Adversary A can modify, eavesdrop or delete the message transmitted between CAV and service provider. In addition, we assume that adversary A can adopt analysis attacks to extract all sensitive information for legitimate participants.

3.4 Security Objectives

Message Authentication and Integrity. For a received message, the message receiver must first determine that the message is sent by a legitimate sender, and judges whether the message has been tampered with or forged by an adversary.

Identity Privacy Preserving. The SP and A can not adopt real identity to communicate directly, and they could use a pseudo-identity to achieve identity privacy. Apart from the VCU, any third party can not track the true identity of the message sender.

Traceability. After receiving the request from the SP, the VCU have the ability to recover the real identity to achieve trace from the message of its App_i pseudo-identity.

Resistance to Ordinary Attacks. The typical attacks, such as modification, replay and impersonation, should be withstood by the proposed scheme, thus the communication is secure.

4 Proposed Scheme

This part presents our proposed in detail. The proposed scheme mainly includes six parts, namely the system initialization phase, pseudonym generation phase of App_i and SP_j, the session key generation phase, the message signcryption phase, the message decryption and verification phase, the traceability and revocation phase. The notations used in this scheme are shown in Table 2. Figure 3 shows the whole phase of the proposed scheme.

4.1 System Initialization

VCU generates system parameters, including generating the paired private and public keys.

1) VCU sets the elliptic curve E. The generator P of E is randomly chosen.
2) VCU chooses the hash functions $H : 0, 1^* \rightarrow Z_q^*$ and $w, r_E \in_R Z_q^*$. Then it computes $P_{pub} = wP$, $R_E = r_E P$, and then $s_E = r_E + wH(R_E)$.
3) VCU publishes the system parameters P, H, P_{pub}, R_E.

Table 2. Notations

Notations	Definitions
VCU	Vehicle Computing/Communication Unit
SP_j	Service Provider j
App_i	The application i of the corresponding SP_j
ID_i, AID_i	The real identity and pseudonym of App_i
ID_j, AID_j	The real identity and pseudonym of SP_j
$H(.)$	The secure cryptographic hash operation
a_i	The secret number of App_i
x, r_E	The secret number of VCU
w, P_{pub}	The secret key and the public key of VCU
s_E	The secret number for legitimate group App_i
SK	The session key for legitimate group member App_i and SP_j
$T_n(x)$	Chaotic maps polynomial
$\oplus$	The exclusive-OR operation
$\|\|$	Concatenation operation

4.2 Pseudonym Generation

In this phase, VCU achieves authorization to the SP_j, and it achieves authentication for legal applications, then it sends the secret value to the corresponding applications.

1) App_i sends ID_i to VCU through a secure channel, and VCU chooses the positive prime number $a_i \in_R Z_q^*$, and sends a_i to App_i.
2) App_i computes $AID_{i1} = a_iP, AID_{i2} = ID_i \oplus H(a_iP_{pub})$, and saves the $AID_i = (AID_{i1}, AID_{i2})$.
3) SP_j sends ID_j to VCU through a secure channel, and VCU chooses a number $x \in_R Z_q^*$, and computes $T_{S_E}(x)$. Then VCU securely sends $x, T_{S_E}(x)$ to SP_j.
4) SP_j chooses random number $u \in_R Z_q^*$, computes $AID_{j1} = uP, AID_{j2} = ID_j \oplus H(uP_{pub})$, and saves the $AID_j = (AID_{j1}, AID_{j2})$. At the same time, the VCU calculates the hash values of multiple SPs $H(ID_j)$, adds them to the hash list of SP H_{list}, and broadcasts the list.

4.3 Key Agreement

When an SP_j wants to access the data for some corresponding applications App_i, it needs to initiate a request.

1) SP_j computes $C_1 = T_u(x)$,$C_2 = T_u(T_{S_E}(x))$,$C_3 = H(ID_j) \oplus H(C_2)$, and $C_4 = H(AID_j||C_1||C_2)$. Then it computes the session key $SK = H(AID_j||C_2)$. Finally, SP_j sends AID_j, C_1, C_3, C_4 to App_i for verifying.

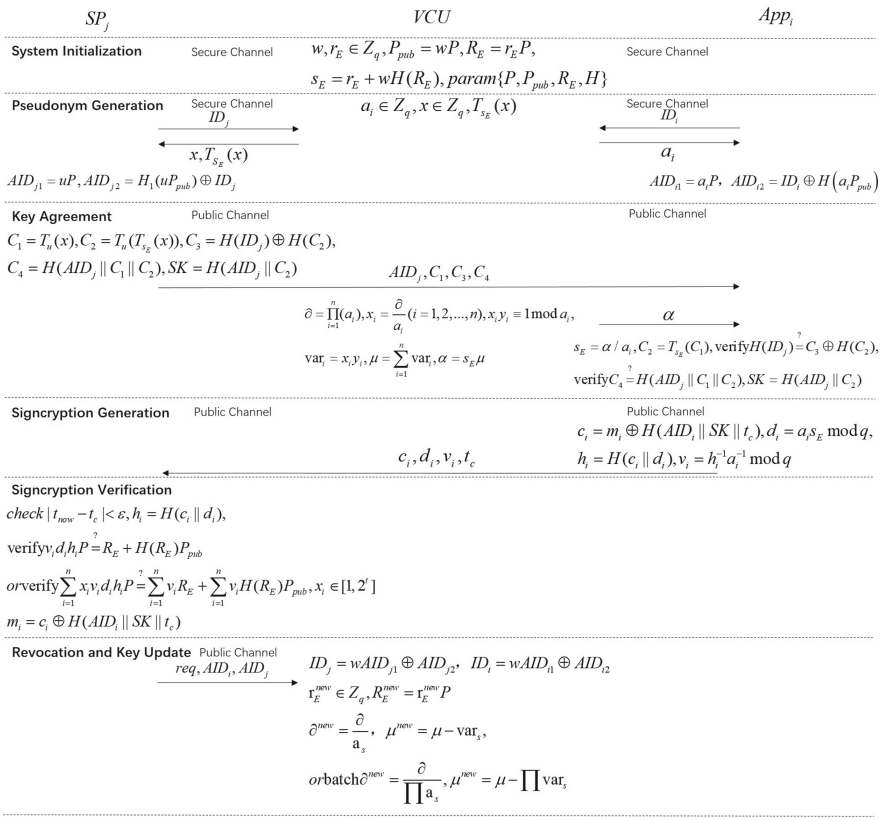

Fig. 3. Interaction Phase of the Proposed Scheme

2) VCU will compute $\partial = \prod_{i=1}^{n}(a_i), x_i = \frac{\partial}{a_i}(i = 1, 2, ..., n), x_i y_i \equiv 1 \bmod a_i, \mathrm{var}_i = x_i y_i, \mu = \sum_{i=1}^{n} \mathrm{var}_i$. Then VCU computes $\alpha = s_E \mu$. VCU will send α to App_i on the public channel.
3) After receiving AID_j, C_1, C_3, C_4 and α, App_i first computes $s_E = \alpha / a_i$, and $C_2 = T_{s_E}(C_1)$. It computes $H(ID_j) = C_3 \oplus H(C_2)$, and then checks whether $H(ID_j)$ exist in the list H_{list}. If existing, then it could verify $C_4 = H(AID_j || C_1 || C_2)$ equals or not. If the equal holds, then it computes $SK = H(AID_j || C_2)$.

4.4 Signcryption Generation

The App_i could encrypt the message m_i by computing $c_i = m_i \oplus H(AID_i || SK || t_c)$. And it computes the signcryption as $d_i = a_i s_E \bmod q, h_i =$

$H(c_i||d_i), v_i = h_i^{-1}a_i^{-1} \bmod q$ where t_c represents the timestamp. Then App_i send c_i, d_i, v_i, t_c to SP_j through VCU.

4.5 Signcryption Verification

After receiving the encrypted message, the SP_j needs to verify and decrypt the message.

1) It first checks the timestamp by $|t_{now} - t_c| \leq \varepsilon$, where t_{now} represents the current time.
2) The SP_j verifies the signcryption by computing $h_i = H(c_i||d_i)$. For one message, it could verify whether this equation $v_i d_i h_i P = R_E + H(R_E)P_{pub}$ holds. For n messages, it could achieve batch verification by check whether this equation $\sum_{i=1}^{n} x_i v_i d_i h_i P = \sum_{i=1}^{n} v_i R_E + \sum_{i=1}^{n} v_i H(R_E)P_{pub}, x_i \in [1, 2^t]$ holds.
3) If the signcryption is verified, it then gets the message by calculating $m_i = c_i \oplus H(AID_i||SK||t_c)$.

4.6 Revocation and Key Update

When the batch authentication is failed, SP_j needs to execute a binary search to find the AID_i that generated the signcryption. It can send a request to the VCU to trace the true identity, which generate the signcryption.

After receiving the request and AID_i, AID_j, the VCU verifies the authenticity of the request and follows these steps:

1) VCU could reveal the real identity of SP_j by calculating ID_j while $ID_j = H(wAID_{j1}) \oplus AID_{j2}$, and checks if ID_j is a valid participant. If valid, continues, or else reject the request.
2) VCU receives the request, it could reveal the real identity of App_i by calculating $ID_i = H(wAID_{i1}) \oplus AID_{i2}$.
3) When the VCU detects a malicious application, it needs to update the key. In addition, the key can be updated regularly even when the system is running normally. VCU updates the new secret value $\mathrm{r}_E^{new} \in {}_RZ_q^*$, and compute $R_E^{new} = \mathrm{r}_E^{new} P$.
4) VCU checks the ID_i in the database, and it realizes the revocation of malicious applications through the following operations:
For a malicious application, it could compute $\partial^{new} = \frac{\partial}{\mathrm{a}_s}\mu^{new} = \mu - \mathrm{var}_s$ to replace the ∂, μ.

For some malicious applications, it could achieve batch revocation by computing $\partial^{new} = \frac{\partial}{\prod \mathrm{a}_s}, \mu^{new} = \mu - \prod \mathrm{var}_s$.

5 Security Analysis

We prove that the proposed scheme is secure under the random oracle model through a formal security proof, in this section. Then, we analyze in detail how the proposed scheme could achieve the aforementioned security requirements in Sect. 3.

5.1 Formal Security Proof

Theorem 1. *Assuming A2S be an event that A may violate the secure communication process between Application and Service Provider. D_{id} and D are the identity dictionary of size $|D_{id}|$ and $|D|$ respectively, and both follow the regular of uniform distribution. Assume that Adv_A^{CMDLP} is the advantage of A in solving chaotic map-based discrete logarithm problem (CMDLP) in polynomial time.*

Proof. Let A represent the adversary who opposes the secure communication procedure. Within the time complexity limit t, the query is executed only at most q_e times, the query is sent q_s times, and the hash query is executed q_h times. Hence,

$$\begin{aligned} Adv_{A2S}^{AKA}(A) \leq & \frac{2(q_s+q_e)}{|D_{id}|} + \frac{q_h^2+q_s}{2^l} + \frac{(q_s+q_e)^2}{p} \\ & + 2q_s \max\{\frac{1}{|D|}, \varepsilon\} + 2q_h((q_s+q_e)^2+1) \\ & * Adv_A^{CMCDH}(A)(t+(q_s+q_e)t_m) \end{aligned} \tag{1}$$

Experiment Exp_0: In this experiment, the real simulated attack is performed in a random oracle model. A has access to all oracles. So we have

$$Adv_{A2S}^{AKA}(A) = 2Pr[E_0] - 1 \tag{2}$$

Experiment Exp_1: This experiment simulates random predictions H through the management of the hash list. Since all predictions are simulated as real attacks, the experiment cannot be distinguished from the actual execution of the protocol. Thus, we have

$$F_1 = |Pr[E_1] - Pr[E_0]| = 0 \tag{3}$$

Experiment Exp_2: This experiment also demonstrates all oracle's predictions Send, Execute, Reveal, Corrupt, and Test. Once A obtains the true identity of A or SP from the identity space, we stop simulating these guessed identity attacks. If this is not the case, $Pr[E_1]$ and $Pr[E_2]$ are indistinguishable:

$$F_2 = |Pr[E_2] - Pr[E_1]| \leq \frac{q_s+q_e}{|D_{id}|} \tag{4}$$

Experiment Exp_3: In this experiment, all oracles are also simulated. There are two conflicting styles in Exp_3. If both collisions occur, the adversary A will launch a replay attack to win the game. According to the birthday paradox, we can draw the possibility of collision. The probability of hash collision is $\frac{q_h^2}{2^{l+1}}$, and probability of random number collision is at $\frac{(q_s+q_e)^2}{2p}$. Hence, the distinguished probability for $Pr[E_2]$ and $Pr[E_3]$ can be represented as:

$$F_3 = |Pr[E_3] - Pr[E_2]| \leq \frac{q_h^2}{2^{l+1}} + \frac{(q_s+q_e)^2}{2p} \tag{5}$$

Experiment Exp_4: Here, all the predictions model in Exp_3 are also used in this experiment. When Corrupt(S) is queried, adversary A can extract the information $C1$, $C3$, $C4$ and s_E stored in the legitimate application. To get the session key, A needs to know the secret value x, r_E, and u. It is difficult to recover x and s_E from the messages $C1$, $C3$ and $C4$. The adversary cannot obtain the correct session key because there is no useful secret value in the entire communication message. So we have

$$F_4 = |Pr[E_4] - Pr[E_3]| \leq q_s \max\{\frac{1}{|D|}, \varepsilon\} \tag{6}$$

Experiment Exp_5: In this experiment, we considered the probability A forged authentication value c_i, d_i, v_i, but do not use the random oracle to make corresponding queries. Oracle can stop the game with the correct value, and Exp_5 becomes indistinguishable from Exp_4 to A. So we will have

$$F_5 = |Pr[E_5] - Pr[E_4]| \leq \frac{q_s}{2^l} \tag{7}$$

Experiment Exp_6: In experiment Exp_6, after the previous Test query, we assume that adversary A is Corrupt(A). Similar to the aforementioned experiments, in the hash oracle, the probability of u and s_E in the same session is $\frac{1}{(qs+qe)^2}$ if the session key SK can be obtained. We define adversary A's advantages as $Adv_A^{CMCDH}(A)(t + (q_e + q_s)t_m)$, and t is the longest time. In addition, t_m is point multiplication time in Elliptic Curve Cryptography. Thus, the game could be won with minimum q_h hash queries. Hence, we have

$$\begin{aligned} F_6 &= |Pr[E_6] - Pr[E_5]| \\ &\leq q_h(q_s + q_e)^2 Adv_A^{CMCDH}(A)(t + (q_e + q_s)t_m) \end{aligned} \tag{8}$$

In addition, if the Test query randomly returns real bit guesses, A will successfully against the oracle. So, we will get

$$Pr[E_7] = Pr[E_6] = 1/2 \tag{9}$$

Therefore, we will get the equal from $F_1, F_2, F_3, F_4, F_5, F_6$

$$
\begin{aligned}
|Pr[E_0] - 1/2| &= |Pr[E_6] - Pr[E_5]| \\
&\leq |Pr[E_1] - Pr[E_0]| + |Pr[E_2] - Pr[E_1]| \\
&+ |Pr[E_3] - Pr[E_2]| + |Pr[E_4] - Pr[E_3]| \\
&+ |Pr[E_5] - Pr[E_4]| + |Pr[E_6] - Pr[E_5]| \\
&= F_1 + F_2 + F_3 + F_4 + F_5 + F_6 \\
&\leq \frac{(q_s + q_e)}{|D_{id}|} + \frac{q_h^2}{2^{l+`}} + \frac{q_s}{2^l} + \frac{(q_s + q_e)^2}{2p} \\
&+ q_s \max\{\frac{1}{|D|}, \varepsilon\} + q_h((q_s + q_e)^2 + 1) \\
&* Adv_A^{CMCDH}(A)(t + (q_s + q_e)t_m)
\end{aligned} \tag{10}
$$

Finally, according to Game $E_0 - E_6$, we could get

$$
\begin{aligned}
Adv_{A2S}^{AKA}(A) &\leq \frac{2(q_s + q_e)}{|D_{id}|} + \frac{q_h^2 + q_s}{2^l} + \frac{(q_s + q_e)^2}{p} \\
&+ 2q_s \max\{\frac{1}{|D|}, \varepsilon\} + 2q_h((q_s + q_e)^2 + 1) \\
&* Adv_A^{CMCDH}(A)(t + (q_s + q_e)t_m)
\end{aligned} \tag{11}
$$

5.2 Security Analysis

Message Authentication. This scheme can realize single message authentication by judging whether this equation $v_i d_i h_i P = R_E + H(R_E)P_{pub}$ holds. It could achieve batch message authentication for massive messages by judging whether this equation $\sum_{i=1}^{n} x_i v_i d_i h_i P = \sum_{i=1}^{n} v_i R_E + \sum_{i=1}^{n} v_i H(R_E)P_{pub}, x_i \in [1, 2^t]$ hold.

Identity Privacy Preserving. SP only needs to collect reliable data of real applications, and there is no need to obtain its real identity. This scheme can realize anonymous communication between SP and the application. For application App_i, only itself and the VCU can obtain his real identity. The a_i and w are kept securely, and the attacker cannot overcome the $ECDLP$ problem within a polynomial time, so the pseudonym is security.

Traceability. When necessary, the ECU can recover the true identity. Since w is only unique to ECU, it can obtain the real identity of the application by calculating $ID_i = H(wAID_{i1}) \oplus AID_{i2}$.

Resistance to Ordinary Attacks. Here, the security of the proposed scheme will be evaluated based on the evaluation criteria and threat model.

Impersonation Attack. In order to pretend to be a server, an adversary A must calculate the effective AID_j, C_1, C_3, C_4. Because of $C_1 = T_u(x)$, $C_2 = T_u(T_{s_E}(x))$, $C_3 = H(ID_j) \oplus H(C_2)$, $C_4 = H(AID_j||C_1||C_2)$, and then $SK = H(AID_j||C_2)$ is computed by $AID_j||C_2$, it is protected by the one-way hash function. A must obtain these secret parameters or guess the correct value in polynomial time. In order to obtain these secret parameters, A needs to have ID_j, u, x. However, it is computationally difficult for an adversary A to guess these values in polynomial time. At the same time, the scheme can achieve anonymity and cannot restore the real identity of the SP_j. An adversary A cannot calculate a valid message. Therefore, the program can resist impersonation attacks.

Modificaiton Attack. Suppose that adversary A initiates a message modification attack, and A successfully calculates the session key on the premise that $T_{s_E u}$ is calculated. However, our formal security proof shows that if the forged message is successfully verified, the difficult problem of $CMCDH$ can be solved in polynomial time. However, it is generally accepted that it is difficult to calculate $CMCDH$ in polynomial time. Therefore, the scheme can resist modification attacks.

Replay Attack. Assume that the adversary A obtains the request message AID_j,C_1,C_2,C_3 and c_i,d_i,v_i,t_c on the public channel. If A re-sends to the application. Due to the existence of the time stamp, the time difference between the time when the message is received and the time when the message is generated is first verified each time. Therefore, the replayed message cannot pass the verification of the message receiver. That is, this scheme could resist replay attacks.

Session Temporary Information Attack. In the proposed scheme, if temporary information is leaked, A can calculate $SK = H(AID_j||C_2)$. A needs to calculate $C_2 = T_{us_E}(x)$. Since A has no way to point out the correct C_2, which is calculated from the chaotic map. Therefore, the scheme can resist the session's temporary information attack.

Man-in-the-Middle Attack. In this scheme, we assume that the adversary A gets the message AID_j, C_1, C_2, C_3 in the public channel. In order to successfully launch a man-in-the-middle attack, A must forge a new message $AID_j^*, C_1^*, C_2^*, C_3^*$ or replay the previous message. As we discussed before, impersonation attacks and replay attacks can be resisted in the proposed scheme. Therefore, the forged message of A cannot be verified by the verifier. Therefore, the scheme can resist man-in-the-middle attacks.

Traceability. This scheme can achieve traceability for malicious applications A_i and SP_j. We assume that there is an application whose message authentication

fails. At this time, the message verifier feeds the message back to the VCU. VCU can trace malicious applications by calculating $ID_i = H(wAID_{i1}) \oplus AID_{i2}$ from AID_i.

Conditional Anonymous Provision. The proposed scheme can realize the anonymity of SP_j and applications A_i during message transmission. At the same time, the pseudonym can be updated regularly, such as a period of time or the discovery of malicious applications. As described earlier, while ensuring anonymity, traceability can be achieved when needed.

Session Key Security. Adversary A could get the all necessary parameters in the public channel, but the session keys must be secured. We assume that A can obtain AID_j, C_1, C_2, C_3, A want to compute the correct SK. However, in order to calculate the session key SK, A needs to calculate T_{us_E}. But it is considered difficult for A to extract random numbers u and s_E from T_{us_E}. Therefore, even if these secret parameters are leaked, A cannot calculate SK. That is, the scheme can achieve session key security.

Efficient Session Key Update. In order to ensure the security of the session key and the secure transmission of the message, this scheme can realize periodic key update. When the VCU detects a malicious application, it immediately executes the key update operation as $r_E^{new} \in {}_RZ_q^*, R_E^{new} = \mathrm{r}_E^{new} P$. At last, the r_E^{new}, R_E^{new} could replace the r_E, R_E.

Mutual Authentication. Mutual authentication can be achieved in our proposed scheme between the application App_i and the service provider SP_j. The application App_i could use the list H_{list} to check whether the service provider SP_j is legal, and it could calculate $C_4 \stackrel{?}{=} H(AID_j||C_1||C_2)$. SP_j could calculate $v_i d_i h_i P \stackrel{?}{=} R_E + H(R_E)P_{pub}$ to verify the message sent by the application App_i.

Efficient Revocation. The proposed scheme can realize the efficient revocation of malicious applications. If the SP_j's message authentication fails, it sends a message to the VCU. If the VCU confirms the message, it can achieve revocation as $\partial^{new} = \frac{\partial}{\mathrm{a}_s}, \mu^{new} = \mu - \mathrm{var}_s$ for the application. Moreover, this scheme can realize batch revocation as $\partial^{new} = \frac{\partial}{\prod \mathrm{a}_s}, \mu^{new} = \mu - \prod \mathrm{var}_s$ of the group of malicious applications.

6 Performance Evaluation

In this section, we analyze the security and computational cost of some schemes [16–19], and demonstrate the result in the form of tables.

The communication costs of different schemes are compared based on the same parameters. The related security parameters we used are as follows. We chose a 160-bit identity for all participants. For the prime number p, the length is 256 bits. And the random number is 128-bit in length, as well as the elliptic curve point is 160-bit in length. The hash function we used is SHA-160. The packet size of the symmetric encryption algorithm is 128-bit. Finally, the timestamp is a length of 16 bits.

CC represents communication cost, CMF represents the round number of communication message flows. We should note that some low-cost operations, such as XOR operations and concatenation operations, are ignored. And we use the following notations in this paper. The notation T_c is the calculation time overhead for expansion chaotic maps, and the notation T_m is the time overhead for point multiplication calculation on an elliptic curve. In addition, the notation T_s is the calculation time overhead for the symmetric cryptographic operation, such as AES. Moreover, the notation T_h is the calculation time overhead for the one-way hash operation.

As shown in Table 3, Ting *et al.*'s scheme [16] and Mandal *et al.*'s scheme [17] could achieve signcryption, but their scheme uses real identities, so transmission on a public channel cannot guarantee higher security. Their scheme cannot achieve efficient key update. Ting *et al.*'s scheme [16] does not provide resisting replay attack and efficient revocation. Ray *et al.*'s scheme [18] and Qiu *et al.*'s scheme [19] could achieve security authentication based on extended chaotic maps. Similarly, these schemes are constructed using real identities ID. Using real identities on public channels may reveal personal privacy information. Qiu *et al.*'s scheme [19] does not support efficient session key update.

Table 3. Security and Functionality Comparison

	[16]	[17]	[18]	[19]	Our scheme
Impersonation attack	√	√	√	√	√
Modificaiton attack	√	√	√	√	√
Replay attacks	×	√	√	√	√
Session temporary information attack	√	√	√	√	√
Man-in-the-middle attack	√	√	√	√	√
Traceability	×	×	×	×	√
Conditional anonymous provision	×	×	×	×	√
Session key security	√	√	√	√	√
Efficient session key update	×	×	√	×	√
Mutual authentication	√	√	√	√	√
Efficient revocation	×	√	√	√	√

Table 4 shows the cost overhead for the scheme in the authentication phases. Ting *et al.*'s scheme [16] will cost $2T_m+2T_h+3T_m+3T_h = 5T_m+5T_h$ to generate

and verify signcryption. Mandal *et al.*'s scheme [17] will cost $8T_m+12T_h+3T_m+8T_h = 11T_m + 20T_h$ to generate and verify signcryption, which requires slightly more computational cost. But this scheme can realize the authentication of users and smart devices with the assistance of the gateway. Roy *et al.*'s scheme [18] will cost $9T_h+2T_c+6T_h+T_c = 15T_h+3T_c$ to achieve user authentication. Qiu *et al.*'s scheme [19] will cost $10T_h+3T_c+8T_h+3T_c = 18T_h+6T_c$ to achieve secure authentication. During the authentication phases, the total computational cost of the proposed scheme is $6T_h + 2T_c + 2T_m + 5T_h + T_c = 2T_m + 11T_h + 3T_c$. We can see that more computational cost is required by our proposed scheme slightly than Qiu *et al.*'s scheme [19], but, compared with other schemes, the scheme in this paper can achieve more functions and better security.

Table 4. Cost comparison

Protocols	User cost	Server cost	Total cost	CC	CMF
[16]	$2T_m + 2T_h$	$3T_m + 3T_h$	$5T_m + 5T_h$	672 bits	1
[17]	$8T_m + 12T_h$	$3T_m + 8T_h$	$11T_m + 20T_h$	1728 bits	2
[18]	$9T_h + 2T_c$	$6T_h + T_c$	$15T_h + 3T_c$	960 bits	2
[19]	$10T_h + 3T_c$	$8T_h + 3T_c$	$18T_h + 6T_c$	1376 bits	2
Ours	$6T_h + 2T_c + 2T_m$	$5T_h + T_c$	$2T_m + 11T_h + 3T_c$	1056 bits	2

Ting *et al.*'s scheme [16] will cost $160 + 128 * 2 + 128 * 2 = 672bits$ to send $c, R_s, K1, d, v$ to receiver. the communication cost of Mandal *et al.*'s scheme [17] is $(160*3+128*2+160+16+128*2)+(128*2+160+16+128) = 1728bits$. The total communication cost of Roy *et al.*'s scheme [18] is $(160+128+160*2+16)+(160 * 2 + 16) = 960bits$. The communication cost of Qiu *et al.*'s scheme [19] is $(160+128+160*3)+(128+160*3) = 1376bits$. The total communication cost of the proposed scheme is $(160+128+160+160)+(160+128*2+16) = 1056bits$. The communication overhead of the proposed scheme is slightly higher than that of Roy *et al.*'s scheme [18], but it is significantly lower than Mandal *et al.*'s scheme [17] and Qiu *et al.*'s scheme [19].

The overhead of signature generation and verification for different schemes is shown in Fig. 4. The signature generation cost of the proposed scheme is higher than that of Ray *et al.*'s scheme [18] and Qiu et al.'s scheme [19]. However, the signature verification cost of the proposed scheme is lower than that of the compared schemes. In addition, the communication overhead is shown in Fig. 5.

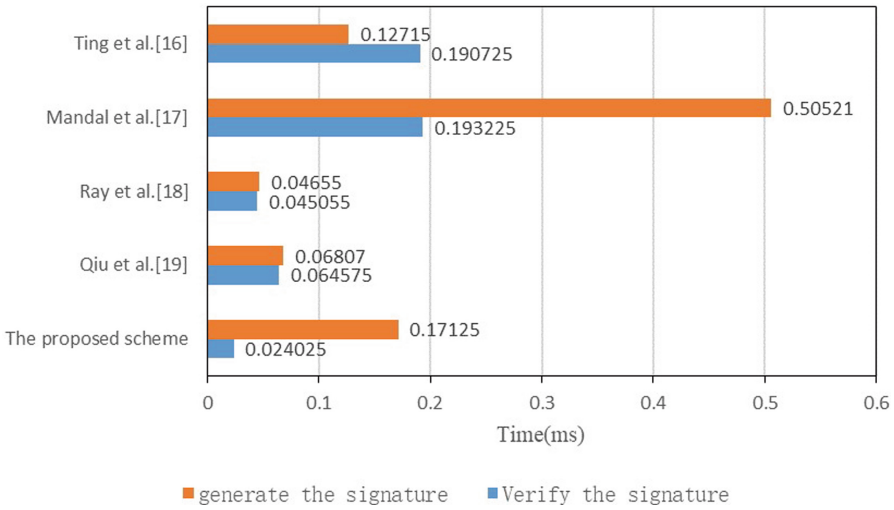

Fig. 4. The computational overhead of generating and verifying signatures

This scheme can realize batch message authentication and its computational overhead is shown in Fig. 6. The results show that the scheme can be used in the case of a large number of messages. The scheme has good performance for the application of large amounts of data security communication in CAV.

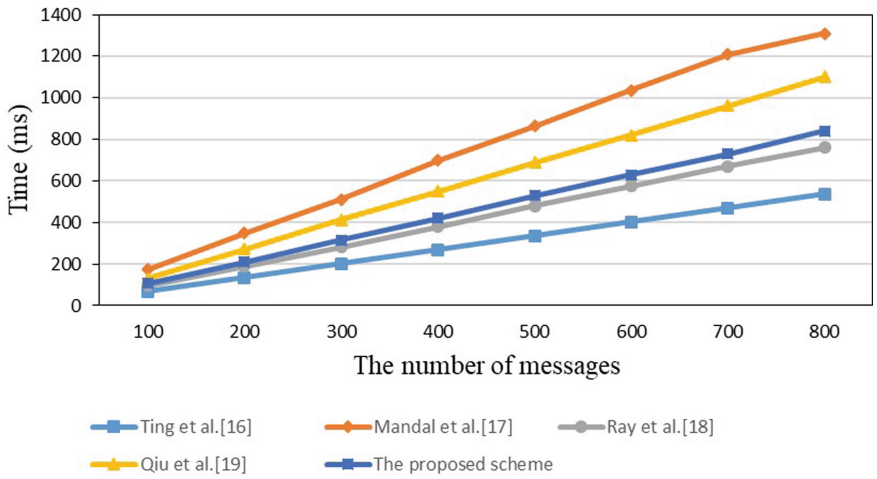

Fig. 5. The communication overhead of multiple messages

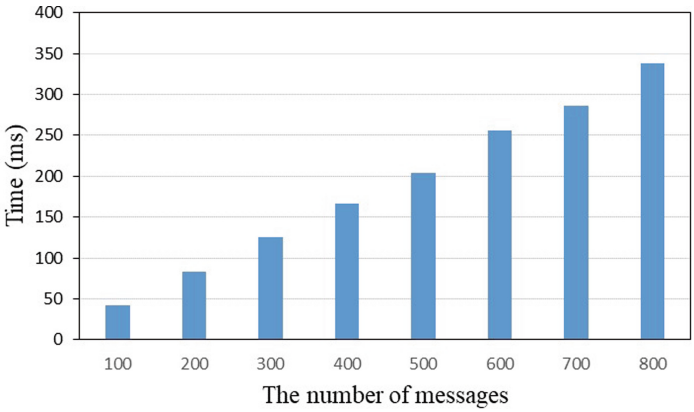

Fig. 6. The computational overhead of massive message authentication

7 Conclusion

To solve the secure data transmission between the CAV internal application module and the corresponding SP, this paper proposes a signcryption scheme based on the chaotic map, which realizes the secure transmission of data between the application module and SP with the assistance of VCU. SP can realize batch message authentication, which could achieve efficient message authentication. This scheme is based on conditional anonymity to achieve mutual authentication between applications and SP. When the VCU knows that there is a compromised application, It could revoke the application module and update the key. The formal security proof shows that the scheme is secure under the random oracle model. Security analysis shows that the scheme can meet the requirements of CAV. The comparison of related schemes shows that the scheme is more efficient and may be applied to the secure communication of CAV.

Acknowledgments. The work was supported in part by Open Fund of Anhui Province Key Laboratory of Cyberspace Security Situation Awareness and Evaluation, in part by the National Natural Science Foundation of China under Grant 62272002, Grant 62202005, and Grant 62202008, in part by the Excellent Youth Foundation of Anhui Scientific Committee under Grant 2108085J31, in part by the Natural Science Foundation of Anhui Province, China under Grant 2208085QF198.

References

1. Kim, K., Kim, J.S., Jeong, S., Park, J.H., Kim, H.K.: Cybersecurity for autonomous vehicles: review of attacks and defense. Comput. Secur. **103**, 102150 (2021)
2. Zhang, Q., et al.: Openvdap: an open vehicular data analytics platform for cavs. In: 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS), pp. 1310–1320 (2018)

3. Sroka, P., Kliks, A.: Towards edge intelligence in the automotive scenario: a discourse on architecture for database-supported autonomous platooning. J. Commun. Netw. **24**(2), 192–208 (2022)
4. Liu, S., Tang, J., Zhang, Z., Gaudiot, J.: Computer architectures for autonomous driving. Computer **50**(8), 18–25 (2017)
5. He, J., et al.: Cooperative connected autonomous vehicles (CAV): research, applications and challenges. In: 2019 IEEE 27th International Conference on Network Protocols (ICNP), pp. 1–6. IEEE (2019)
6. Ma, Y., Wang, Z., Yang, H., Yang, L.: Artificial intelligence applications in the development of autonomous vehicles: a survey. IEEE/CAA J. Automatica Sinica **7**(2), 315–329 (2020)
7. Wei, W., Yang, R., Gu, H., Zhao, W., Chen, C., Wan, S.: Multi-objective optimization for resource allocation in vehicular cloud computing networks. IEEE Trans. Intell. Transp. Syst. **23**(12), 25536–25545 (2021)
8. Cui, J., Wei, L., Zhang, J., Xu, Y., Zhong, H.: An efficient message-authentication scheme based on edge computing for vehicular ad hoc networks. IEEE Trans. Intell. Transp. Syst. **20**(5), 1621–1632 (2018)
9. Zhang, Q., Wu, J., Zhong, H., He, D., Cui, J.: Efficient anonymous authentication based on physically unclonable function in industrial internet of things. IEEE Trans. Inf. Forensics Secur. **18**, 233–247 (2023)
10. Li, T., Shang, M., Wang, S., Filippelli, M., Stern, R.: Detecting stealthy cyberattacks on automated vehicles via generative adversarial networks. In: 2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC), pp. 3632–3637. IEEE (2022)
11. Zhang, Q., Zhong, H., Cui, J., Ren, L., Shi, W.: Ac4av: a flexible and dynamic access control framework for connected and autonomous vehicles. IEEE Internet Things J. **8**(3), 1946–1958 (2021)
12. Mousavinejad, E., Yang, F., Han, Q.L., Ge, X., Vlacic, L.: Distributed cyber attacks detection and recovery mechanism for vehicle platooning. IEEE Trans. Intell. Transport. Syst. **21**, 3821–3834 (2019)
13. Chattopadhyay, A., Lam, K.Y., Tavva, Y.: Autonomous vehicle: security by design. IEEE Trans. Intell. Transport. Syst. **22**, 7015–7029 (2020)
14. Wazid, M., Das, A.K., Kumar, N., Vasilakos, A.V., Rodrigues, J.J.: Design and analysis of secure lightweight remote user authentication and key agreement scheme in internet of drones deployment. IEEE Internet Things J. **6**(2), 3572–3584 (2018)
15. Bagga, P., Das, A.K., Wazid, M., Rodrigues, J.J., Choo, K.K.R., Park, Y.: On the design of mutual authentication and key agreement protocol in internet of vehicles-enabled intelligent transportation system. IEEE Trans. Veh. Technol. **70**(2), 1736–1751 (2021)
16. Ting, P., Tsai, J., Wu, T.: Signcryption method suitable for low-power IoT devices in a wireless sensor network. IEEE Syst. J. **12**(3), 2385–2394 (2018)
17. Mandal, S., Bera, B., Sutrala, A.K., Das, A.K., Choo, K.R., Park, Y.: Certificateless-signcryption-based three-factor user access control scheme for IoT environment. IEEE Internet Things J. **7**(4), 3184–3197 (2020)
18. Roy, S., Chatterjee, S., Das, A.K., Chattopadhyay, S., Kumari, S., Jo, M.: Chaotic map-based anonymous user authentication scheme with user biometrics and fuzzy extractor for crowdsourcing internet of things. IEEE Internet Things J. **5**(4), 2884–2895 (2018)
19. Qiu, S., Wang, D., Xu, G., Kumari, S.: Practical and provably secure three-factor authentication protocol based on extended chaotic-maps for mobile lightweight devices. IEEE Trans. Depend. Secure Comput. **19**(2), 1338–1351 (2022)

20. Tangade, S., Manvi, S.S., Lorenz, P.: Decentralized and scalable privacy-preserving authentication scheme in vanets. IEEE Trans. Veh. Technol. **67**(9), 8647–8655 (2018)
21. Cui, J., Yu, J., Zhong, H., Wei, L., Liu, L.: Chaotic map-based authentication scheme using physical unclonable function for internet of autonomous vehicle. IEEE Trans. Intell. Transp. Syst. **24**(3), 3167–3181 (2023)
22. Xu, G., Dong, J., Ma, C., Liu, J., Cliff, U.G.O.: A certificateless signcryption mechanism based on blockchain for edge computing. IEEE Internet Things J. (2022)
23. Bergamo, P., D'Arco, P., De Santis, A., Kocarev, L.: Security of public-key cryptosystems based on Chebyshev polynomials. IEEE Trans. Circuits Syst. I Regul. Pap. **52**(7), 1382–1393 (2005)
24. Zhang, L.: Cryptanalysis of the public key encryption based on multiple chaotic systems. Chaos Solitons Fractals **37**(3), 669–674 (2008)
25. Abbasinezhad-Mood, D., Ostad-Sharif, A., Mazinani, S.M., Nikooghadam, M.: Provably-secure escrow-less Chebyshev chaotic map-based key agreement protocol for vehicle to grid connections with privacy protection. IEEE Trans. Ind. Inf. **16**, 7287–7294 (2020)
26. Lu, R., Heung, K., Lashkari, A.H., Ghorbani, A.A.: A lightweight privacy-preserving data aggregation scheme for fog computing-enhanced IoT. IEEE Access **5**, 3302–3312 (2017)
27. Zhang, J., Cui, J., Zhong, H., Chen, Z., Liu, L.: PA-CRT: Chinese remainder theorem based conditional privacy-preserving authentication scheme in vehicular ad-hoc networks. IEEE Trans. Depend. Secure Comput. **18**, 722–735 (2019)
28. Xiong, H., Chen, J., Mei, Q., Zhao, Y.: Conditional privacy-preserving authentication protocol with dynamic membership updating for vanets. IEEE Trans. Depend. Secure Comput. **19**, 2089–2104 (2020)
29. Maene, P., Götzfried, J., De Clercq, R., Müller, T., Freiling, F., Verbauwhede, I.: Hardware-based trusted computing architectures for isolation and attestation. IEEE Trans. Comput. **67**(3), 361–374 (2017)
30. Zhang, Q., Zhong, H., Shi, W., Liu, L.: A trusted and collaborative framework for deep learning in IoT. Comput. Netw. **193**, 108055 (2021)