Secure Chaotic Maps-based Group Key Agreement Scheme with Privacy Preserving

Hongfeng Zhu
(Corresponding author: Hongfeng Zhu)

Software College, Shenyang Normal University
No.253, Huang He Bei Street, Huang Gu District, Shenyang 110034, P. R. China
(Email: zhuhongfeng1978@163.com)

(Received May 27, 2014; revised and accepted Oct. 27 & Dec. 24, 2014)

Abstract

Nowadays chaos theory related to cryptography has been addressed widely, so there is an intuitive connection between group key agreement and chaotic maps. Such a connector may lead to a novel way to construct authenticated and efficient group key agreement protocols. Many chaotic maps based two-party/three-party password authenticated key agreement (2PAKA/3PAKA) schemes have been proposed. However, to the best of our knowledge, no chaotic maps based group (N-party) key agreement protocol without using a timestamp and password has been proposed yet. In this paper, we propose the first chaotic maps-based group authentication key agreement protocol. The proposed protocol is based on chaotic maps to create a kind of signcryption method to transmit authenticated information and make the calculated consumption and communicating round restrict to an acceptable bound. At the same time our proposed protocol can achieve members' revocation or join easily, which not only refrains from consuming modular exponential computing and scalar multiplication on an elliptic curve, but is also robust to resist various attacks and achieves perfect forward secrecy with privacy preserving.

Keywords: Authentication, chaotic maps, group key, random oracle model

1 Introduction

In the network information era, it is important to structure group key agreement schemes which are designed to provide a set of players, and communicating over a public network with a session key to be used to implement secure multicast sessions, e.g., video conferencing, collaborative computation, file sharing via internet, secure group chat, group purchase of encrypted content and so on.

With the rapid development of chaos theory related to cryptography [3, 4, 15, 16, 18, 34], many key agreement protocols using a chaotic map have been studied widely. These protocols using a chaotic map can mainly be divided into three directions: two-party authenticated key agreement protocols [2, 9, 10, 11, 12, 13, 14, 24, 25, 26, 27, 28, 31, 32, 33, 37, 39], three-party authenticated key agreement protocols [8, 19, 20, 29, 30, 36, 38, 40], and N-party authenticated key agreement protocols. Furthermore, we can classify the literatures [2, 8, 9, 10, 11, 12, 13, 14, 19, 20, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 36, 37, 38, 39, 40] based on their respective features in detail, such as password-based, using smart card, timestamp, anonymity and other security attributes. From the macroscopic point of view, these literatures have two main traits: On the one hand, along with some new protocols putting forward, then some flaws will be found over a period of time, such as the flaws in the literatures [11, 25, 32] are found by the literatures [2, 12, 14]. On the other hand, the evolution of the key agreement protocols using a chaotic map shows putting in new secure attributes and improving the efficiency, for example the literatures [13, 28, 33, 37]. In recent years, the three-party password-authenticated key agreement protocol using modular exponentiation or scalar multiplication on an elliptic curve has been addressed widely [30, 38]. However, these schemes need heavy computation costs and even most recent the research is still remaining on three-party authenticated key agreement protocol [36].

To the best of our knowledge, no N-party authenticated key agreement protocol based on chaotic maps has been proposed, yet. To design group authentication key agreement protocols in chaotic map setting is difficult but is very useful in many application environments. The difficult of the setting is when the number of participants increasing, and how to keep computing and communication increasing linearly or constantly. So it is quite natural to utilize N-party authenticated key agreement literature that related to cryptography. The first work in this area is by Bresson et al. [21]. As already mentioned, their proposed scheme is secure in both the random oracle model and the ideal cipher model. Next Lee presents a password-based group key protocol [5] which is not authenticated because there is no way to convince a user that the message that he receives is indeed coming from the intended participant. Recently there are three literatures about password-based group key scheme [1, 7, 22, 42] and Abdalla et al. [1] points out the literature [7] which is subjected to an off-line dictionary attack, however their efficiency is unsatisfactory.

In this paper, we put forward a new simple and efficient N-party authenticated key agreement protocol based on chaotic maps. We present our contributions below:

1) Communication round: Our proposed protocol is efficient from communication point of view as it requires only 2 rounds and uses Chebyshev chaotic maps and symmetric key encryption instead of signature for message authentication in the round 1. And in the round 2, we mainly use hash function and operations to authenticated each other and compute the group session key. These methods reduce the bandwidth of the messages sent and make the protocol faster.

2) Computation: Our protocol is based on chaotic maps without using modular exponentiation and scalar multiplication on an elliptic curve.

3) Security: The protocol can resist all common attacks, such as impersonation attacks, man-in-the-middle attacks, etc.

4) Functionality: It allows N ($N \geq 2$) users establish a secure session key over an insecure communication channel with the help of public key system with chaotic maps. The proposed protocol has provided the case of a member revocation or a new member join. Furthermore the protocol also has achieved some well-known properties, such as perfect forward secrecy, no timestamp, and execution efficiency.

The rest of the paper is organized as follows: We outline preliminaries in Section 2. Next, A Chebyshev chaotic maps-based N-party authenticated key agreement protocol is described in Section 3. Then, the security analysis and efficiency analysis are given in Section 4. This paper is finally concluded in Section 5.

2 Preliminaries

Let n be an integer and let x be a variable with the interval $[-1, 1]$. The Chebyshev polynomial. $T_n(x) : [-1, 1] \to [-1, 1]$ is defined as $T_n(x) = \cos(n \arccos(x))$ Chebyshev polynomial map $T_n : R \to R$ of degree n is defined using the following recurrent relation [29]:

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x) \tag{1}$$

where $n \geq 2$, $T_0(x) = 1$, and $T_1(x) = x$. The first few Chebyshev polynomials are:

$$\begin{aligned} T_2(x) &= 2x^2 - 1, \\ T_3(x) &= 4x^3 - 3x, \\ T_4(x) &= 8x^4 - 8x^2 + 1, \\ &\vdots \end{aligned}$$

One of the most important properties is that Chebyshev polynomials are the so-called semi-group property which establishes that

$$T_r(T_s(x)) = T_{r \cdot s}(x). \tag{2}$$

An immediate consequence of this property is that Chebyshev polynomials commute under composition:

$$T_r(T_s(x)) = T_s(T_r(x)). \tag{3}$$

In order to enhance the security, Zhang [41] proved that semi-group property holds for Chebyshev polynomials defined on interval $(-\infty, +\infty)$. In our proposed protocol, we utilize the enhanced Chebyshev polynomials:

$$T_n(x) = (2xT_{n-1}(x) - T_{n-2}(x))(\bmod N) \tag{4}$$

where $n \geq 2$, $x \in (-\infty, +\infty)$, and N is a large prime number. Obviously,

$$T_{r \cdot s}(x) = T_r(T_s(x)) = T_s(T_r(x)). \tag{5}$$

Definition 1. *Semi-group property of Chebyshev polynomials:*

$$\begin{aligned} T_r(T_s(x)) &= \cos(r\cos^{-1}(s\cos^{-1}(x))) \\ &= \cos(rs\cos^{-1}(x)) = T_{sr}(x) \\ &= T_s(T_r(x)). \end{aligned}$$

Definition 2. *Given x and y, it is intractable to find the integer s, such that $T_s(x) = y$. It is called the Chaotic Maps-Based Discrete Logarithm problem (CMBDLP).*

Definition 3. *Given x, $T_r(x)$, and $T_s(x)$, it is intractable to find $T_{rs}(x)$. It is called the Chaotic Maps-Based Diffie-Hellman problem (CMBDHP).*

3 Group Key Agreement from Chaotic Maps

We now consider the generic construction for a two-round group key agreement from Chaotic Maps. All group participants $U_1, U_2, ..., U_n$ are organized in an ordered chain and U_{i+1} is the successor of U_i. The temporary two-party symmetric session key computed in a parallel algorithm based on Chaotic Maps-Based Diffie-Hellman problem is used as the shared secret between the participant U_i and its successor $U_{i+1}, i = 1, ..., n$. The structure of the kind of group key agreement from Chaotic Maps is illustrated in Figure 1 which includes the following two rounds.

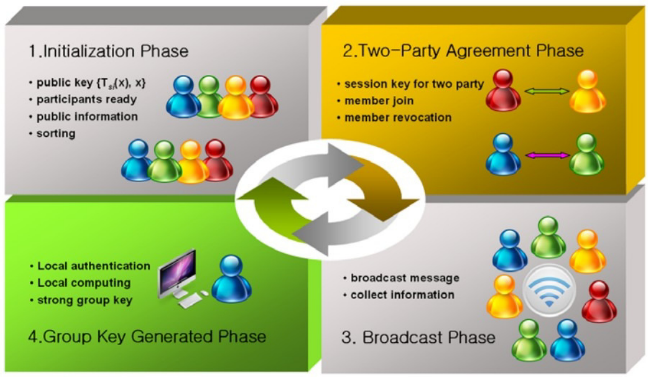

Figure 1: Structure of the PGKA phases (The phases are presented clockwise)

3.1 Setup Phase

In this phase, any user U_i has its identity ID_i, and public key $(x, T_{S_i}(x))$ and a secret key S_i based on Chebyshev chaotic maps, a chaotic maps-based one-way hash function $h(\cdot)$ [35], and a pair of secure symmetric encryption/decryption functions $E_K()/D_K()$ with key K. The concrete notation used hereafter is shown in Table 1.

3.2 Authentication and Two-party Agreement Phase

Let $U = \{U_1, U_2, ..., U_n\}$ be the set of protocol participants. All the participants $U_1, U_2, ..., U_n$ run the following process. This process is presented in Figure 2.

Remark 1. *In order to put emphasis on describing the proposed protocol, we assume that all ID information has been arranged.*

Step 1. User U_i selects a random number r_i and computes

$$\begin{aligned} K_{i,i+1} &= T_{r_i}T_{S_{i+1}}(x), \\ C_i &= E_{K_{i,i+1}}(ID_i||ID_{i+1}||T_{r_i}(x)) \\ MAC_i &= H(ID_i||ID_{i+1}||C_i||H(K_{i,i+1})||T_{r_i}(x)), \end{aligned}$$

and sends messages $\{C_i, T_{r_i}(x), MAC_i\}$ to user U_{i+1}.

Step 2. After receiving the messages $\{C_i, T_{r_i}(x), MAC_i\}$, user U_{i+1} firstly computes $T_{S_{i+1}}T_{r_i}(x) = K_{i+1,i}$ to extract C_i to get ID information. Then user U_{i+1} verifies MAC_0 through computing

$$H(ID_i||ID_{i+1}||C_i||H(K_{i+1,i})||T_{r_i}(x)).$$

If $H(ID_i||ID_{i+1}||C_i||H(K_{i+1,i})||T_{r_i}(x)) = MAC_i$ holds, then U_{i+1} selects a random number r_{i+1} and compute

$$\begin{aligned} K_{i+1,i} &= T_{r_{i+1}}T_{S_i}(x) \\ SK &= T_{r_{i+1}}T_{r_i}(x), \\ C_{i+1} &= E_{K_{i+1,i}}(ID_i||ID_{i+1}||T_{r_{i+1}}(x)), \\ MAC_{i+1} &= H(ID_i||ID_{i+1}||C_{i+1}||T_{r_{i+1}}(x) \\ &\quad ||H(K_{i+1,i})||SK). \end{aligned}$$

Finally user U_{i+1} sends messages $\{C_{i+1}, T_{r_{i+1}}(x), MAC_{i+1}\}$ to user U_i.

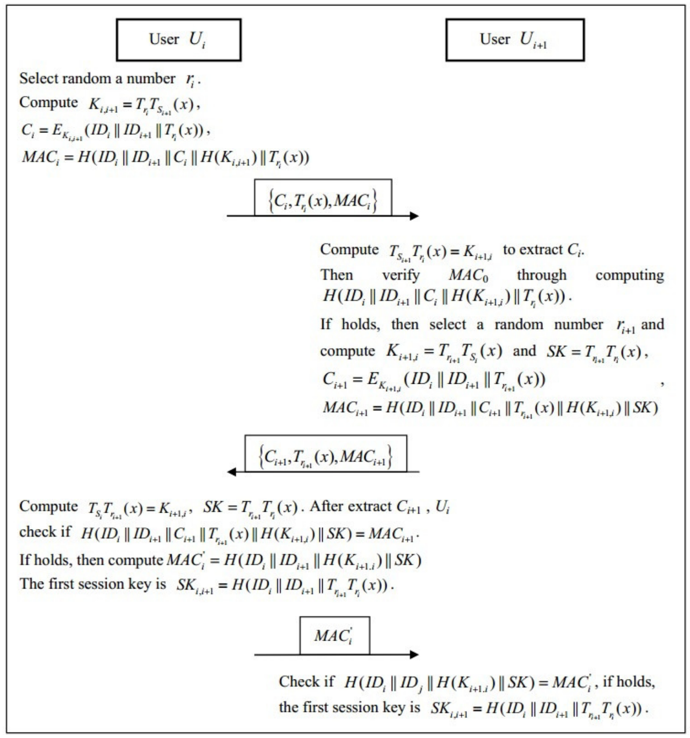

Figure 2: Two-party agreement phase

Step 3. After receiving the messages $\{C_{i+1}, T_{r_{i+1}}(x), MAC_{i+1}\}$, user U_i uses S_i and r_i to compute $T_{S_i}T_{r_{i+1}}(x) = K_{i+1,i}$ and $SK = T_{r_{i+1}}T_{r_i}(x)$. User U_i uses $K_{i+1,i}$ to extract C_{i+1} and computes $H(ID_i||ID_{i+1}||C_{i+1}||T_{r_{i+1}}(x)||H(K_{i+1,i})||SK)$ and then checks if it equals MAC_{i+1}.

If not, user U_i terminates it. Otherwise, user U_i computes $MAC_i' = H(ID_i||ID_{i+1}||H(K_{i+1.i})||SK)$ and $SK_{i,i+1} = H(ID_i||ID_{i+1}||T_{r_{i+1}}T_{r_i}(x))$. User U_i sends MAC_i' to user U_{i+1}, and at the same time takes $SK_{i,i+1} = H(ID_i||ID_{i+1}||T_{r_{i+1}}T_{r_i}(x))$ as the session key.

Step 4. Upon receiving MAC_i', user U_{i+1} computes $H(ID_i||ID_{i+1}||H(K_{i+1.i})||SK)$ and checks if it equals MAC_i'. If not, user U_{i+1} terminates it. Otherwise, user U_{i+1} uses $SK_{i,i+1} = H(ID_i||ID_{i+1}||T_{r_{i+1}}T_{r_i}(x))$ as the session key.

Table 1: Notations

Symbols	Definition
U_i, ID_i	The Participant i and its identity information;
U	Set of protocol participants;
$(x, T_{S_i}(x))$	Public key based on Chebyshev chaotic maps;
S_i	Secret key based on Chebyshev chaotic maps;
$E_K(\cdot)/D_K(\cdot)$	A pair of secure symmetric encryption/decryption functions with the key K;
r_i	Random nonce chosen by each U_i;
$\oplus$	A bitwise Xor operator;
$\|\|$	Two adjacent messages are concatenated;
H	A chaotic maps-based one-way hash function.

The phase can be simultaneous and parallel. Finally, each participant has two two-party agreement keys ($SK_{i,i+1}$ and $SK_{i-1,i}$) with its sucessor and predecessor (U_1 computes $SK_{1,2}$ and $SK_{n,1}$).

3.3 Broadcast and Group Key Agreement Generated Phase

This process is presented in Figure 3.

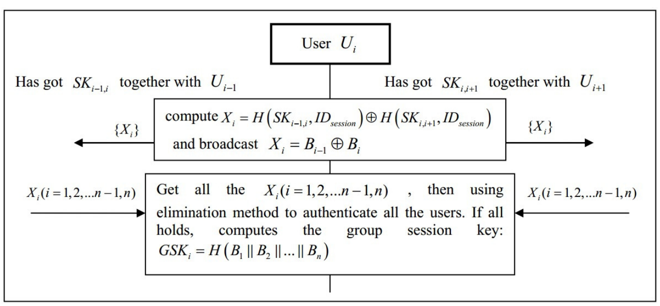

Figure 3: Group Key Agreement Generated Phase

The participants $U_i, i = 2, ..., n$, compute and broadcast X_i, where $X_i = B_{i-1} \oplus B_i = H(SK_{i-1,i}, ID_{session}) \oplus H(SK_{i,i+1}, ID_{session})$. Note that the first participant U_1 computes and broadcasts $X_1 = H(SK_{n,1}, ID_{session}) \oplus H(SK_{1,2}, ID_{session})$. Here $ID_{session}$ is the public ephemeral information that consists of participants' identities and a nonce, aiming to make the protocol secure against known-key attacks. To sum it up, we can see Table 2.

Finally, with secret $SK_{i-1,i}$ and $SK_{i,i+1}$ the participant $U_i\,(i = 1, ..., n)$ computes B_i and further get all $B_j\,(j = 1, ..., n)$ using continuous XOR method. Then, the participant $U_i, (i = 1, ..., n)$ compares B_{i-1} and$H(SK_{i-1,i}, ID_{session})$ locally. Furthermore, each participant $U_i (i = 1, ..., n)$ verifies if $X_1 \oplus X_2 \oplus X_3 \oplus ... \oplus X_{n-1} \oplus X_n = 0$ holds and all participants will continue to compute the group key. If not, output an error symbol $\perp$ and abort. After all participants accomplish the verifying, they compute the group session key $GSK_i = H(B_1||B_2||...||B_n)$. Obviously, $GSK_1 = GSK_2 = ... = GSK_n$. This will be the common strong group session key agreed by all participants.

3.4 A Member Revocation or a New Member Join Phase

A Member Revocation: Assume that a participant leaves the group. Then group members change the group size into $(n-1)$. The U_{x-1} participants U_{x-1} and U_{x+1} respectively remove the shared values $SK_{x-1,x}$ and $SK_{x,x+1}$ with U_x. The participant U_{x+1} becomes the new successor of participant U_{x-1}. Aiming to update group key, the participant U_{x-1} needs to send new message C_{x-1} to its new successor U_{x+1} and U_{x+1} needs to send new message $C^{'}_{x+1}$ to its new predecessor U_{x-1}. Then,the participant U_{x+1} verifies the validity of the message $\{C_{x-1}, T_{r_{x-1}}(x), MAC_{x-1}\}$ and computes the secret $SK_{x-1,x+1}$ which is a new shared secret between U_{x-1} and U_{x+1}. Each party U_j that follows U_x changes their index to $(j-1)$. Then, recomputed Section 3.3, all the $(n-1)$ participants implement the above protocol to get a new group session key.

A New Member Join: Assume that a new entity joins the group of which size is n. Then, the new participant U_{n+1}, becomes the successor of participant U_n and the participant U_1 becomes the successor of participant U_{n+1}.

The participant U_n sends message $\{C_n,\ T_{r_n}(x), MAC_n\}$ according to ID_n and ID_{n+1} to its new successor U_{n+1} while U_{n+1} sends message $\{C_{n+1},\ T_{r_{n+1}}(x), MAC_{n+1}\}$ to U_{n+1} based on ID_nandID_{n+1}.

From the message C_n and $C^{'}_{n+1}$, the new participant U_{n+1} verifies the validity of the message and computes the secret $SK_{n,n+1}$ which is the new shared secret between U_n and its new successor U_{n+1}. At the same time, the first participant U_1 updates its secret with $SK_{n+1,1}$ in Figure 4. Then, recomputed Section 3.3, the participants in the group implement the above protocol to get a new group session key.

Table 2: The value of B_i

Notations	B_1	B_2	$\cdots$	B_i	$\cdots$
Value	$H(SK_{1,2}, ID_{Session})$	$H(SK_{2,3}, ID_{Session})$	$\cdots$	$H(SK_{i,i+1}, ID_{Session})$	$\cdots$

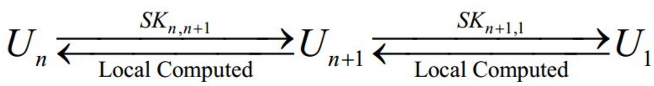

Figure 4: A new member join case

Remark 2. *The proposed protocol, when member revoke and join, the computation and communication complexity is increasing with N linearly.*

4 Security Consideration and Efficiency Analysis

Assume there are three secure components, including the two problems CMBDLP and CMBDHP cannot be solved in polynomial-time, a secure chaotic maps-based one-way hash function, and a secure symmetric encryption. Assume that the adversary has full control over the insecure channel including eavesdropping, recording, intercepting, modifying the transmitted messages. However, the adversary could neither get the temporary values r_i chosen in the local machine nor guess ID_i correctly at the same time.

In this section, we classify the functions of group authentication key agreement scheme based on chaotic maps into two types, auxiliary function and essential function. We also prove that our proposed scheme achieves the security and efficiency goals.

4.1 Auxiliary Function

Privacy Preserving

In our protocol, the users' sensitive information such as identities is private to both the participants and the adversaries. During the whole scheme, the privacy is protected by the one-way hash function and symmetric encryption with chaotic maps-based for transferring over insecure channel and cannot be retrieved from the transmission messages. The user's identity is always combined with a nonce as $E_{K_{i,i+1}}(ID_i||ID_{i+1}||T_{r_i}(x))$ transmitting to the next participant.

Natural Resistance

Our protocol is based on public key system with chaotic maps without smart card or password, so its naturally resists many attacks, such as SEG attack [23], Password guessing attack, Stolen-verifier attack and so on.

No Clock Synchronization

The proposed protocol solves the clock synchronization problem with no timestamp mechanism. Instead, we introduce fresh random number r_i and r_{i+1} to provide the challenge response security mechanism so that replay attack cannot threaten the proposed scheme while no clock synchronization is needed.

4.2 Essential Function

Mutual Authentication, Group Authentication and Key Agreement

The proposed scheme allows the participant U_{i+1} to authenticate the participant U_i by checking whether $H(ID_i||ID_{i+1}||C_i||H(K_{i+1,i})||T_{r_i}(x)) \stackrel{?}{=} MAC_i$. Furthermore only owning the secret key S_{i+1} can extract C_i to get the secret message to verify the receiving message. About group authentication phase, each participant $U_i (i = 1, ..., n)$ verifies if $X_1 \oplus X_2 \oplus X_3 \oplus ... \oplus X_{n-1} \oplus X_n = 0$ holds and all participants will continue to compute the group key. If not, output an error symbol $\perp$ and abort.

Resist Well-known Attacks

1) Impersonation Attack/Man-in-the-Middle Attack

   An adversary cannot impersonate the user U_i to cheat the participant, because it is not able to get the secret key of the user U_i and afterwards cannot extract C'_{i+1} to compute two-party session key. From the above analysis, we can know that an adversary is unable to achieve success by impersonating and replaying. On the other hand, because $\{C_i, T_{r_i}(x), MAC_i\}$, $\{C_{i+1}, T_{r_{i+1}}(x), MAC_{i+1}\}$ and $X_i, 1 \leq i \leq n$ contain the users' identities, a man-in-the-middle attack cannot succeed.

2) Replay Attack

   An adversary cannot start a replay attack against our scheme because of the freshness of r_i in each session. If $T_{r_i}(x)$ has appeared before or the status shows in process, the participant U_{i+1} rejects the session request. If the adversary wants to launch the replay attack successfully, it must compute and modify $T_{r_i}(x)$ and C_i correctly which is impossible.

3) Known-key Security

   Since two-party session key $SK_{i,i+1} = H(ID_i||ID_{i+1}||T_{r_i}T_{r_{i+1}}(x))$ is depended on the random nonces r_i and r_{i+1}, and the generation

Table 3: Descriptions the model of Canetti and Krawczyk

Symbols	Definition
parties $P_1, \cdots, P_n$	Modelled by probabilistic Turing machines.
Adversary *wedge*	A probabilistic Turing machine which controls all communication, with the exception that the adversary cannot inject or modify messages (except for messages from corrupted parties or sessions), and any message may be delivered at most once.
Send query	The adversary can control over Parties' outgoing messages via the **Send** query. Parties can be activated by the adversary launching **Send** queries.
Two sessions matching	If the outgoing messages of one are the incoming messages of the other.

of nonces is independent in all sessions, an adversary cannot compute the previous and the future session keys when he knows one session key. About the group session key $GSK_i = H(B_1||B_2||...||B_n)$ which based on all random nonces $r_i, 1 \leq i \leq n$, an adversary cannot compute the previous and the future group session keys when he knows one group session key.

4) Perfect Forward Secrecy
In the proposed scheme, the session key $SK_{i,i+1} = H(ID_i||ID_{i+1}||T_{r_i}T_{r_{i+1}}(x))$ is related with r_i and r_{i+1}, which were chosen by user U_i and user U_{i+1}, respectively. Because of the intractability of the CMBDLP and CMBDHP problem, an adversary cannot compute the previously established session keys. About the group session key $GSK_i = H(B_1||B_2||...||B_n)$ which based on all random nonces $r_i, 1 \leq i \leq n$, an adversary cannot compute the previously established group session keys yet.

5) Key Compromise Impersonation Attacks (KCI Attacks)
Informally, an adversary is said to impersonate a party B to another party A if B is honest and the protocol instance at A accepts the session with B as one of the session peers but there exists no such partnered instance at B [17]. In a successful KCI attack, an adversary with the knowledge of the long-term private key of a party A can impersonate B to A. We assume that an adversary can know U_1 and $U_3^{'}$s secret keys S_1 and S_3, then he can impersonate U_2 to cheat U_1 and U_3, and $U_4...U_n$, and to get the group session key $GSK_i = H(B_1||B_2||...||B_n)$. But above-mentioned process will not achieve and the attack course terminates at the beginning. Because an adversary cannot own the U_2's secret key S_2, and he cannot pass validation of U_3: An adversary do not possess $U_2^{'}$s secret key S_2, so he cannot compute $T_{S_i}T_{r_{i+1}}(x) = K_{i+1,i}$, and then he cannot compute the MAC_i' $= H(ID_i||ID_{i+1}||H(K_{i+1.i})||SK)$, finally U_3 will check if $H(ID_i||ID_j||H(K_{i+1.i})||SK) = MAC_i^{'}$. If not, user U_3 terminates it. The key compromise impersonation attacks will fail.

4.3 The Provable Security of Our Scheme

We recall the definition of session-key security in the authenticated-links adversarial model of Canetti and Krawczyk [6]. The basic descriptions are shown in Table 3.

We allow the adversary access to the queries **SessionStateReveal, SessionKeyReveal**, and **Corrupt**.

1) SessionStateReveal(s): This query allows the adversary to obtain the contents of the session state, including any secret information. s means no further output.

2) SessionKeyReveal(s): This query enables the adversary to obtain the session key for the specified session s, so long as s holds a session key.

3) Corrupt(Pi): This query allows the adversary to take over the party P_i, including long-lived keys and any session-specific information in $P_i^{'}$s memory. A corrupted party produces no further output.

4) Test(s): This query allows the adversary to be issued at any stage to a completed, fresh, unexpired session s. A bit b is then picked randomly. If $b = 0$, the test oracle reveals the session key, and if $b = 1$, it generates a random value in the key space. The adversary can then continue to issue queries as desired, with the exception that it cannot expose Λ the test session. At any point, the adversary can try to guess b. Let $GoodGuess^{\Lambda}(k)$ be the event that the adversary Λ correctly guesses b, and we define the advantage of adversary Λ as $Advantage^{\Lambda}(k) = \max\{0, |\Pr[GoodGuess^{\Lambda}(k)] - \frac{1}{2}|\}$, where k is a security parameter.

A session s is locally exposed with P_i: If the adversary has issued **SessionStateReveal(s)**, **SessionKeyReveal(s)**, **Corrupt(Pi)** before s is expired.

Table 4: Security of our proposed protocol

Privacy preserving	Natural resistance	No. clock synchronization	Mutual and group authentication	Impersonation
Provided	Provided	Provided	Provided	Provided
Man in the Middle Attack	Replay Attack	Known Key Security	Perfect Forward Secrecy	Key Compromise Impersonation
Provided	Provided	Provided	Provided	Provided

Table 5: Efficiency of our proposed protocol for one participant

Hash	XOR	Symmetric En/decryption	Modular Multiplication	Modular Exponent	Elliptic Curve Multiplication	Elliptic Curve Addition	Chebyshev Polynomial	Round Number
9	n	4	0	0	0	0	6	2

Definition 4. *A key exchange protocol Π_1 in security parameter k is said to be session-key secure in the adversarial model of Canetti and Krawczyk if for any polynomial-time adversary Λ, is satisfied. To show that the second part of the definition is satisfied, assume that there is a polynomial-time adversary Λ with a non-negligible advantage ε in standard model. We claim that Algorithm 1 forms a polynomial- time distinguisher for CMBDHP having non-negligible advantage.*

1) *If two uncorrupted parties have completed matching sessions, these sessions produce the same key as output;*

2) *$Advantage^{\Lambda}(k)$ is negligible.*

Algorithm 1 CMBDHP distinguisher

Input : $H, E_K()/D_K(), (x, T_{k_A}(x)), (x, T_{k_B}(x))$

1: $r \xleftarrow{R} \{1,...,k\}$, where k is an upper bound on the number of sessions activated by Λ in any interaction.

2: Invoke Λ and simulate the protocol to Λ, except for the $r-th$ activated protocol session.

3: For the $r-th$ session, let Alice send $\{i, T_{R_A}(x), ID_A, ID_B, C_1\}$ to Bob, and let Bob send $\{i, T_{R_B}(x), ID_A, ID_B, C_2\}$ to Alice, where i is the session identifier. Both Alice and Bob can compute the session key $SK = H(T_{R_A}T_{R_B}(x))$ locally after authenticating each other by one-round messages and public information.

4: **if** the $r-th$ session is chosen by Λ as the test session **then**

5: Provide Λ as the answer to the test query.

6: $d \leftarrow \Lambda$'s output.

7:**else**

8: $d \xleftarrow{R} \{0,1\}$.

9:**end if**

Output: d

Theorem 1. *Under the CMBDHP assumption, using the Algorithm 1 to compute session key is session-key secure in the adversarial model of Canetti and Krawczyk [6].*

Proof. The proof is based on the proof given by [6, 26]. There are two uncorrupted parties in matching sessions output the same session key, and thus the first part of **Probability analysis.** It is clear that Algorithm 1 runs in polynomial time and has non-negligible advantage. There are two cases where the r-th session is chosen by Λ as the test session:

1) If the r-th session is not the test session, then Algorithm 1 outputs a random bit, and thus its advantage in solving the CMBDHP is 0.

2) If the r-th session is the test session, then Λ will succeed with advantage ε, since the simulated protocol provided to Λ is indistinguishable from the real protocol. The latter case occurs with probability $1/k$, so the overall advantage of the CMBDHP distinguisher is ε/k, which is non-negligible.

□

4.4 Practical in Pervasive and Ubiquitous Computing Environment

Compared to RSA and ECC, Chebyshev polynomial computation problem offers smaller key sizes, faster computation, as well as memory, energy and bandwidth savings. In our proposed protocol, no time-consuming modular exponentiation and scalar multiplication on elliptic curves are needed. However, Xiao et al. [34] and Wang [29] proposed several methods to solve the Chebyshev polynomial computation problem. In addition for getting the group key agreement, our proposed protocol uses hash function and $\oplus$ operations, and both of them are all high efficient algorithm.

To the best of our knowledge, no N-party authenticated key agreement protocol based on chaotic maps has been proposed, so there are no literatures to contrast and we sum up our proposed protocol as show in Table 4 (Security) and Table 5 (Efficiency). Furthermore the case of members revocation or new members join also have provided in the paper.

5 Conclusions

We put forward the first N-party authenticated key agreement protocol based on chaotic maps, symmetric key encryption, hash function and $\oplus$ operations which are all better algorithm than RSA and ECC and so on. From the Table 5, we can see easily that ours protocol computing and communication increasing constantly along with the number of participants N, and only XOR operation increasing linearly with the number of participants N. Security of our proposed protocol is also satisfactory from the Table 4. Next we will extend the proposed protocol to high level security attributes such as fairness or entanglement and so on.

References

[1] M. Abdalla, E. Bresson E, Chevassut O, "Password-based group key exchange in a constant number of rounds," in *Proceedings of Public Key Cryptography (PKC'06)*, pp. 427–442, 2006.

[2] G. Alvarez, "Security problems with a chaos-based deniable authentication scheme," *Chaos, Solitons and Fractals*, vol. 26, no. 1, pp.7–11, 2005.

[3] M. S. Baptista, "Cryptography with chaos," *Physics Letters A*, vol. 240, no. 1, pp. 50–54, 1998.

[4] S. Behnia, A. Akhshani, S. Ahadpour, H. Mahmodi, A. Akhavan, "A fast chaotic encryption scheme based on piecewise nonlinear chaotic maps," *Physics Letters*, vol. 366, pp. 391–396, 2007.

[5] E. Bresson, O. Chevassut and D. Pointcheval, "Group Diffie-Hellman key exchange secure against dictionary attack," in *Advances in Cryptography (Asiacrypt'02)*, LNCS 2501, pp. 497–514, Springer, 2002.

[6] R. Canetti, and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *Advances in Cryptography (EUROCRYPT'01)*, LNCS 2045, pp. 453–474, Springer, 2001.

[7] R. Dutta, R. Barua, "Password-based encrypted group key agreement," *International Journal of Network Security*, vol. 3, no. 1, PP. 23–34, July 2006.

[8] T. H. Feng, C. H. Ling, and M. S. Hwang, "Cryptanalysis of Tan's Improvement on a Password Authentication Scheme for Multi-server Environments," *International Journal of Network Security*, vol. 16, no. 4, pp. 318–321, 2014.

[9] P. Gong, P. Li, W. Shi, "A secure chaotic maps-based key agreement protocol without using smart cards," *Nonlinear Dynamics*, vol. 70, pp. 2401–2406, 2012.

[10] C. Guo, C. C. Chang, "Chaotic maps-based password-authenticated key agreement using smart cards," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 6, pp. 1433–1440, 2012.

[11] X. Guo, J. Zhang, "Secure group key agreement protocol based on chaotic Hash," *Information Sciences*, vol. 180, no. 20, pp. 4069–4074, 2010.

[12] S. Han, "Security of a key agreement protocol based on chaotic maps," *Chaos Solitons Fractals*, pp. 764–768, 2008.

[13] S. Han, E. Chang, "Chaotic map based key agreement without clock synchronization," *Chaos Solitons Fractals*, vol. 39, pp. 1283–1289, 2009.

[14] D. He, "Cryptanalysis of a key agreement protocol based on chaotic hash," *International Journal of Electronic Security and Digital Forensics*, vol. 5, no. 3, pp.172–177, 2013.

[15] I. Hussain, T. Shah, M. Gondal, "A novel approach for designing substitution-boxes based on nonlinear chaotic algorithm," *Nonlinear Dynamics*, vol. 70, no. 3, pp. 1791–1794, 2012.

[16] I. Hussain, T. Shah, M. Gondal, and H. Mahmood, "An efficient approach for the construction of LFT S-boxes using chaotic logistic map," *Nonlinear Dynamics*, vol. 71, no. 1-2, pp. 133–140, 2013.

[17] J. Katz, J. S. Shin, "Modelling insider attacks on group key-exchange protocols," in *Proceedings of the 12th ACM Conference on Computer and Communications Security*, pp. 180–189, 2005.

[18] M. Khan, T. Shah, H. Mahmood, M. Gondal, "An efficient method for the construction of block cipher with multi-chaotic systems," *Nonlinear Dynamics*, vol. 71, pp. 489–492, 2013.

[19] H. Lai, J. Xiao, L. Li, Y. Yang, "Applying semi-group property of enhanced Chebyshev polynomials to anonymous authentication protocol," *Mathematical Problems in Engineering*, vol. 2012, Article ID 454823, 2012.

[20] C. Lee, C. Li, C. Hsu, "A three-party password-based authenticated key exchange protocol with user anonymity using extended chaotic maps," *Nonlinear Dynamics*, vol. 73, pp. 125–132, 2013.

[21] S. M. Lee, J. Y. Hwang and D. H. Lee, "Efficient password-based group key exchange," in *Proceedings of TrustBus*, pp. 191–199, 2004.

[22] H. Li, C. K. Wu, J. Sun, "A general compiler for password-authenticated group key exchange protocol," *Information Processing Letters*, vol. 110, pp. 160–167, 2010.

[23] T. H. Liu, Q. Wang, H. F. Zhu, "A multi-function password mutual authentication key agreement scheme with privacy preserving," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 5, no. 2, pp. 165–178, 2014.

[24] Y. Niu, X. Wang, "An anonymous key agreement protocol based on chaotic maps," *Communication Nonlinear*, vol. 16, no. 4, pp.1986–1992, 2011.

[25] F. Özkaynak, S. Yavuz, "Designing chaotic S-boxes based on time-delay chaotic system," *Nonlinear Dynamics*, vol. 74, no. 3, pp. 551–557, 2013

[26] A. Stolbunov, "Reductionist security arguments for public-key cryptographic schemes based on group action," in *The Norwegian Information Security Conference (NISK'09)*, pp. 97–109, 2009.

[27] Z. Tan, "A chaotic maps-based authenticated key agreement protocol with strong anonymity," *Nonlinear Dynamics*, vol. 72, pp. 311–320, 2013.

[28] H. Tseng, R. Jan, W. Yang, "A chaotic maps-based key agreement protocol that preserves user anonymity," *International Conference on Communications*, pp. 1–6, 2009.

[29] X. Wang, J. Zhao, "An improved key agreement protocol based on chaos," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, pp. 4052–4057, 2010.

[30] S. Wu, K. Chen, Q. Pu, Y. Zhu, "Cryptanalysis and enhancements of efficient three-party password-based key exchange scheme," *International Journal of Communication Systems*, vol. 26, no. 5, pp. 674–686, 2013.

[31] T. Xiang, K. Wong, X. Liao, "On the security of a novel key agreement protocol based on chaotic maps," *Chaos Solitons Fractals*, vol. 40, pp. 672–675, 2009.

[32] D. Xiao, X. Liao, S. Deng, "A novel key agreement protocol based on chaotic maps," *Information Science*, vol. 177, pp. 1136–1142, 2007.

[33] D. Xiao, X. Liao, S. Deng, "Using time-stamp to improve the security of a chaotic maps-based key agreement protocol," *Information Sciences*, vol. 178, no. 6, pp. 1598–1602, 2008.

[34] D. Xiao, X. Liao, K. Wong, "An efficient entire chaos-based scheme for deniable authentication," *Chaos Solitons and Fractals*, vol. 23, no. 4, pp. 1327–1331, 2005.

[35] D. Xiao, F. Shih, X. Liao, "A chaos-based hash function with both modification detection and localization capabilities," *Communications in Nonlinear Science and Numerical Simulation*, vol. 15, pp. 2254–2261, 2010.

[36] Q. Xie, J. M. Zhao, X. Y. Yu, "Chaotic maps-based three party password-authenticated key agreement scheme," *Nonlinear Dynamics*, vol. 74, pp. 1021–1027, 2013.

[37] K. Xue, P. Hong, "Security improvement on an anonymous key agreement protocol based on chaotic maps," *Communication Nonlinear*, vol. 17, pp. 2969–2977, 2012.

[38] J. Yang, T. Cao, "Provably secure three-party password authenticated key exchange protocol in the standard model.," *Journal of Systems and Software*, vol. 85, pp. 340–350, 2012.

[39] E. Yoon, "Efficiency and security problems of anonymous key agreement protocol based on chaotic maps," *Communication Nonlinear*, vol. 17, pp. 2735–2740, 2012.

[40] E. Yoon, I. Jeon, "An efficient and secure Diffie-Hellman key agreement protocol based on Chebyshev chaotic map," *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, pp. 2383–2389, 2011.

[41] L. Zhang, "Cryptanalysis of the public key encryption based on multiple chaotic systems," *Chaos Solitons Fractals*, vol. 37, no. 3, pp. 669–674, 2008.

[42] M. H. Zheng, H. H. Zhou, J. Li, G. H. Cui, " Efficient and provably secure password-based group key agreement protocol," *Computer Standards and Interfaces*, vol. 31, no. 5, pp. 948–953, 2009.

Hongfeng Zhu obtained his Ph.D. degree in Information Science and Engineering from Northeastern University. Hongfeng Zhu is a full associate professor of the Kexin software college at Shenyang Normal University. He is also a master's supervisor. He has research interests in wireless networks, mobile computing, cloud computing, social networks, network security and quantum cryptography. Dr. Zhu had published more than 50 international journal papers (SCI or EI journals) on the above research fields.