ITC 1/46
Journal of Information Technology and Control
Vol. 46 / No. 1 / 2017
pp. 138-149
DOI 10.5755/j01.itc.46.1.13845

A Multi-Party Secret Handshake Scheme Based on Chaotic Maps

Received 2016/12/20 | Accepted after revision 2017/02/03

http://dx.doi.org/10.5755/j01.itc.46.1.13845

A Multi-Party Secret Handshake Scheme Based on Chaotic Maps

Wenbo Wang, Qingfeng Cheng

School of Computer Science and Technology, Xidian University, Xi'an 710071, China; State Key Laboratory of Cryptology, P. O. Box 5159, Beijing, 100878, China; Luoyang University of Foreign Languages, Department of Language and Engineering, Luoyang, Henan 471003, China, email: qingfengc2008@sina.com

Siqi Lu

State Key Laboratory of Cryptology, P. O. Box 5159, Beijing, 100878, China; Luoyang University of Foreign Languages, Department of Language and Engineering, Luoyang, Henan 471003, China, email: qingfengc2008@sina.com

Jianfeng Ma

School of Computer Science and Technology, Xidian University, Xi'an 710071, China

Corresponding author: qingfengc2008@sina.com

The primitive secret handshake is a kind of privacy-preserving authentication protocol, in which the participants can share a common session key if and only if they come from the same group, without the leakage of the group information. Most of the current secret handshakes are realized by means of bilinear maps, whose computational cost is a lot. A new multi-party secret handshake scheme is proposed in this paper using the chaotic map, with the computational cost reducing significantly. The new protocol also supports user revocation, and has the ability of tracing users, meanwhile proved to achieve the basic security properties of secret handshakes.

KEYWORDS: secret handshakes, multi-party, chaotic maps, computational cost.

Introduction

The secret handshake protocol, which provides privacy-preserving authentication among users belonging to the same group, was first proposed by Balfanz *et al.* [1] as a two-party protocol with three-round using pairing based cryptography. Balfanz *et al.*'s protocol realizes the property of affiliation-hiding that prevents an adversary from learning anything about the group or the user identity by eavesdropping or even

executing the protocol with a legal user. However, there exists some drawbacks in this original secret handshake, for example it needs multiple credentials which is a burden in computational cost, and the protocol is linkable, which means that different sessions executed by the same user can be linked by reusing the user's certificate.

Several secret handshake protocols have been proposed [5, 12, 13, 15, 16, 24, 25, 34] after the work of Balfanz *et al.*. Castelluccia proposed a secret handshake based on CA-Oblivious encryption [3], which improves the efficiency of Balfanz *et al.*'s secret handshake scheme [1]. Xu and Yung constructed a secret handshake scheme [34] the same year, which achieves unlinkability. But their scheme only satisfies the property of k-anonymity, which means that an adversary can deduce that an executor of a session is one out of certain k users.

All the protocols introduced above only consider two participants. Ysudik and Xu first expand the number of participants to more than two, proposing the first Group Secret Handshake scheme [29] in the setting of multi-party taking part. In a group secret handshake protocol, two or more users from the same group could authenticate with each other without the leakage of group information. However, their protocol does not establish a common shared key for participants after authentication. Jarecki *et al.* introduced the notion of Affiliation-Hiding Authenticated Group Key Agreement (AH-AGKA), and proposed two concrete AH-AGKA schemes [14]. Xu *et al.* proposed the concept of Affiliation-Hiding Authenticated Asymmetric Group Key Agreement (AH-AAGKA) [35], and proposed an AH-AAGKA scheme [32] to improve the one of Jarecki *et al.* They proposed another AH-AAGKA scheme [33] which reduces the communication round to only one.

Efficient revocation is an important element in designing a secret handshake scheme. Revocation is also closely related with the property of linkability. A pseudonym instead of the real identity of a user is often used to realize the function of revocation. In a linkable protocol, the group administrator (GA) simply puts the pseudonym of a user into a certificate revocation list to revoke them, making revocation very simple. But to realize the property of unlinkability requires one-time certificates, which would cost a lot. Sorniotti and Molva proposed a secret handshake with revocation support [28], meanwhile maintaining the property of unlinkability.

Thanks to its property of affiliation hiding, the multi-party secret handshake can be used as a useful tool for secure communication among users whose identities need to be kept secret. The new types of networks, such as the wireless sensor network (WSN) [10], the vehicular ad hoc network (VANET) [11], the WSN [10], the underwater sensor network (UWSN) [27] and the wireless mesh network (WMN) [8], can use it to ensure that the users of the same property can secretly verify the identity of others, meanwhile establishing a shared key. When applied to these types of networks, one has to consider the malicious users who might damage the communicating system using network attacks such as DoS. Thus it is quite important to detect the identity of these kinds of users, meanwhile protecting other legitimate users' identities from being detected by adversaries.

Computational cost is of great importance in designing secret handshake schemes, especially when applied to concrete environment mentioned above. Most of the secret handshake schemes introduced above are realized by using the bilinear maps. It is, of course, an efficient tool to implement a secret handshake. However, the computational cost that it brings about is quite high. The notion of Chebyshev polynomial was first introduced by Mason and Handcomb in 2003 [26]. It has been used to construct authenticated key agreement [6, 21, 22, 31, 36-39] by some researchers. In addition, its computational cost is much lower than that of a bilinear map. However, its usage in the field of secret handshake is not in much concern.

In this paper, we design a new multi-party secret handshake scheme based on chaotic maps (MPSH-CM). We adopted the idea of Sorniotti and Molva [28] of using the matching reference and pseudonym to realize the function of revocation while achieving the property of unlinkability as well.

The usage of chaotic maps significantly reduces the computational cost compared to other multi-party secret handshakes using bilinear maps. Our protocol also concerns about those malicious users in the environment of concrete application, the MANET for example. When GA is aware of malicious attacks from users, it will trace the identity of malicious users and revoke them to avoid further damages.

The rest of this paper is organized as follows. Section 2 introduces Chebyshev chaotic map and the corresponding attack against it. Section 3 introduces the basic models and definitions of secret handshakes, including the security requirements. Section 4 presents our new multi-party secret handshakes based on chaotic maps. The protocol will be proved secure in Section 5. Section 6 analyzes the performance of our protocol. Section 7 concludes the paper.

Preliminaries

In this section, we introduce some basic knowledge of Chebyshev chaotic map [26]. For more information, please refer to [2, 20, 30].

Chebyshev chaotic map

Definition 1. Let $n \in Z^*, x \in [-1,1]$. A Chebyshev polynomial $T_n(x):[-1,1] \to [-1,1]$ is a polynomial in x of degree n, which is defined as:

$$T_n(x) = \cos(n \cdot \arccos x) \tag{1}$$

Its recurrence relation is defined recursively as:

$$\begin{aligned} &T_0(x) = 1, T_1(x) = 1, T_2(x) = 2x^2 - 1, \cdots \\ &T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x), n \geq 2. \end{aligned} \tag{2}$$

The Chebyshev polynomial satisfies the following properties:

1 **The semi-group property:**

$$\begin{aligned} &T_r(T_s(x)) = \cos(r\cos^{-1}(\cos(s\cos^{-1}(x)))) \\ &= \cos(rs\cos^{-1}(x)) = T_{sr}(x)) = T_s(T_r(x)) \end{aligned} \tag{3}$$

where $r, s \in Z^*$ and $x \in [-1,1]$.

2 **The chaotic property:**

When the degree n satisfies $n > 1$, the Chebyshev polynomial map $T_n(x):[-1,1] \to [-1,1]$ of degree n is a chaotic map with its invariant density being

$$f^*(x) = 1/(\pi\sqrt{1-x^2}), \tag{4}$$

for Lyaounov exponent $\lambda = \ln n$.

For the purpose of improving security, Zhang [38] extends the range of the semi-group property, proving that the semi-group property holds for Chebyshev polynomials defined on $(-\infty, +\infty)$:

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x) \bmod p, \tag{5}$$

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

$$T_r(T_s(x)) = T_{sr}(x)) = T_s(T_r(x)) \bmod p. \tag{6}$$

Definition 2. Discrete Logarithm Problem (DLP): Given two elements x and y, it is computationally intractable to find the integer s that satisfies $T_s(x) = y$.

Definition 3. Diffie-Hellman Problem (DHP): Given two elements $T_s(x)$ and $T_r(x)$, it is computationally infeasible to compute $T_{sr}(x)$.

The attack of Bergamo *et al.*

Bergamo *et al.* proposed a method of attack [2] against the cryptosystem of Kocarev and Tasev [18], which is constructed based on chaotic maps.

The attack can be performed as follows:

If an attacker can attain the elements x and $T_r(x)$, then a element r' that achieves $T_r(x) = T_{r'}(x)$ can be computed as

$$r' = \frac{\arccos(T_r(x)) + 2k\pi}{\arccos(x)}, k \in Z. \tag{7}$$

Models and definitions

In a secret handshake scheme, a set of users $u_1, \cdots, u_n$ of the same property p form a group G, and the common property p is generally regarded as the group identity of the group G. A group administrator (GA) is in charge of creating the group and issues credentials to legal users for adding members. According to the definitions introduced in [1], our multiparty secret handshake scheme consists of the following probabilistic polynomial-time algorithms:

- **Setup.** The Setup algorithm generates and outputs the public parameters *param* on the input of a security parameter l. The group public key *pk* is also included in the public parameters *param*, and the private key *sk* is kept secret by GA.

- **Add Member**. The Add Member algorithm is executed between user u and GA, and takes *param* and *sk* as input. If u is verified to own the property p, GA chooses and outputs a credential $cred_u$ for u using the group key *pk* and *sk*. A pseudonym is often used as the identity of u instead of his real identity, and is included in the credential. After receiving the credential, user u becomes a member of the group.
- **Handshake**. Suppose n users take part in the Handshake algorithm. The algorithm takes the credentials of each user as the secret input and *param* as the public input. The output of the protocol for each member is either 'reject' or 'accept'. If and only if all of the n users belong to the same group (i.e., all of the n users own the same property p), then the outputs be 'accept'. If the n outputs from the n users are all 'accept', the handshake is successfully performed and a common session key is shared among the n users.
- **Trace Member**. The Trace Member algorithm is executed by GA in order to trace the identity of member u. The algorithm takes the publicly transmitted messages and outputs the identity of u.
- **Revoke Member**. The Revoke Member algorithm is executed by GA in case that a user needs to be revoked. It takes the current revocation list (denoted as L_{rev}) and the pseudonym of u. In addition, the output is an up-to-date list L_{rev}. After the algorithm, the group will not include u as its legal member, while u will never be able to take part in any handshakes.

Security properties. A secret handshake scheme should satisfy the following security properties:

1. **Completeness**. If all of the n users belong to the same group and execute the Handshake protocol honestly, then all n users output 'accept'.
2. **Detector Resistance**. If an adversary activates a **Handshake** with a legal member, he will be able to detect the affiliation information of the member with a negligible probability. In other words, the protocol is affiliation hiding.
3. **Impersonator Resistance**. An adversary can **successfully** impersonate a legitimate member of a group with negligible probability. As explained in [15], the impersonator resistance implies the untraceability property.
4. **Unlinkability**. An adversary can check two **handshake** tuples come from the same user with negligible probability.

A new multi-party secret handshake scheme based on chaotic maps

In this section, a new multi-party secret handshake scheme based on chaotic maps (MPSH-CM) will be introduced. The construction of our protocol is as follows:

- **Setup**. Given the security parameter l, the Setup algorithm outputs the system's public parameters as $param =< G, P, q, H, W >$, where P is a random generator of a cyclic group G with order q, $|q| = l$, and $H : \{0,1\}^* \to Z_q^*$ is a cryptographic hash function. $W = wP$, the value $w \in_R Z_q^*$ is kept secret.
- **Add Member**. The GA verifies whether user $u \in \overline{U}$ possesses the property $p \in \overline{P}$, which is used as the group identity. If the verification is passed successfully, the GA chooses pseudonym $x_{u,p} \in_R Z_q^*$ for u, which is used as the unique identification of u. GA then selects $z, t \in_R Z_q^*$, which are randomly picked upon each query, and issues to u the credential $cred_{u.p} =< C_{u,1}, C_{u,2}, C_{u,3} >$ via a secure channel, where $C_{u,1} = x_{u,p} \cdot Q_p(Q_p + t)P$, $C_{u,2} = z^{-1}Q_pP$, $C_{u,3} = (zw)^{-1}Q_pP$, $Q_p = \sin(H(p))$ (the sin function is used so as to set the Q_p into the domain of the chaotic function). User u can verify the validity of the credential by randomly choosing $m \in [0,1]$ and checking the two equations:

$$T_{C_{u,1}}(m) = T_{x_{u,p}Q_p^2P + xQ_pT}(m) \tag{8}$$

and

$$T_{C_{u,2}}(m) = T_W T_{C_{u,3}}(m). \tag{9}$$

GA issues to u the matching reference $match_p = (Q_p + t)^{-1}P$ to make sure that user u can use it to verify the group identity of other users, which is also an indicator of the ability of u to verify and communicate with other users with property p.

– **Handshakes**. Suppose there are n users that take part in the secret handshake, which are noted as $\{u_i\}, i = 1,...,n$. Here we treat u_0 as u_n, which means the subscript i is set to $i \bmod n$. Moreover, we use p_i to represent the property that u_i owns. The simplified description of the phase is shown in Fig.1.

Step 1. u_i randomly chooses $\tau_i, s_i \in \mathbf{Z}_q^*$ and computes the chaotic functions

$$M_{i,1} = T_{s_i C_{i,1}}(Q_{p_i}), \tag{10}$$

$$M_{i,2} = T_{s^{-1}C_{i,2}}(Q_{p_i}), \tag{11}$$

$$M_{i,3} = T_{s^{-1}C_{i,3}}(Q_{p_i}), \tag{12}$$

$$M_{i,4} = T_{Q_p P s_i}(Q_p). \tag{13}$$

Step 2. u_i publishes the message $M_i = \{M_{i,1}, M_{i,2}, M_{i,3}, M_{i,4}, \tau_i\}$.

Step 3. After receiving messages $\{M_j\}_{j\neq i}$ from other users, u_i first verifies whether u_{i-1} is a revoked user or not by checking the following equation:

$$T_{match_{p_i}}(M_{i-1,1}) = T_{rev}(M_{i-1,4}). \tag{14}$$

If there exits $rev \in L_{rev}$ such that it satisfies equation (14), u_i exposes the revoked user u_{i-1} and discards the current instance by output the 'reject' message; or the protocol proceeds.

Step 4. If there is no 'reject' message, u_i will compute X_i as follows:

$$X_i = T_{s_i^{-1}C_{i,2}} T_W(M_{i+1,3}) - T_{s_i^{-1}C_{i,2}} T_P(M_{i-1,2}). \tag{15}$$

u_i then publishes the message $\{X_i, \tau_i\}$.

Step 5. After receiving $\{X_j\}_{j\neq i}$ and $\{\tau_j\}_{j\neq i}$ from other users, u_i checks whether $\{\tau_j\}_{j\neq i}$ matches those in $\{M_j\}_{j\neq i}$ and checks if the equation $\sum_{i=1} X_i = 0$ stands. If not, a 'reject' message will be published, and the protocol is ceased.

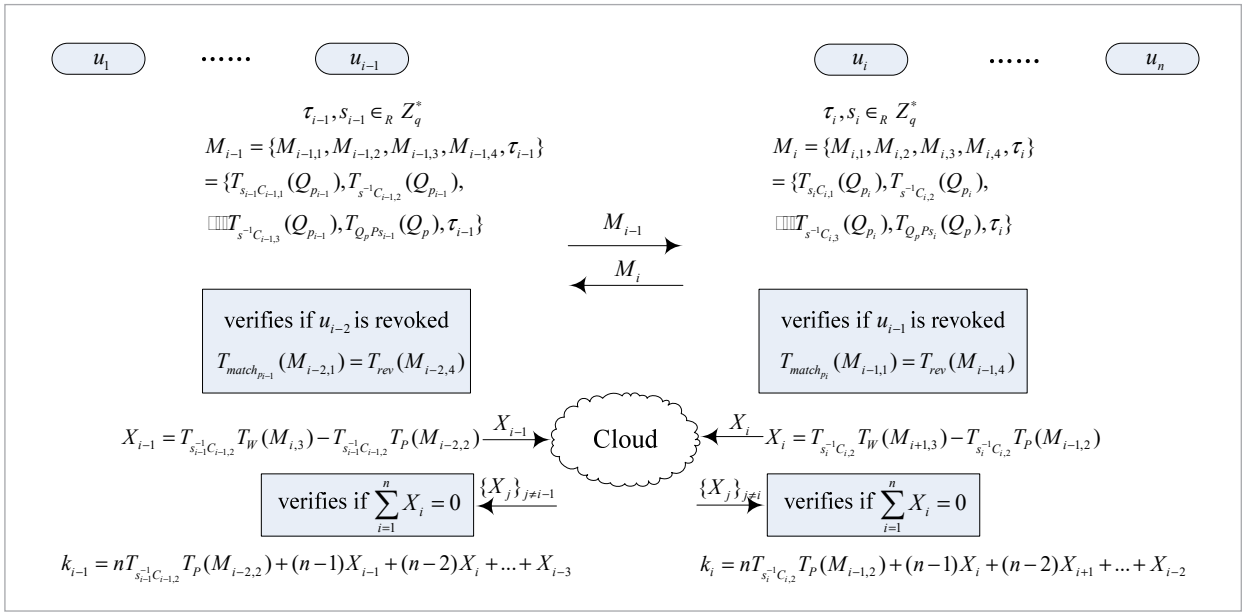
Step 6. u_i computes

$$k_i = nT_{s_i^{-1}C_i}, T_P(M_{i-1,2}) + (n-1)X_i + (n-2)X_{i+1} + ... + X_{i-2} \tag{16}$$

as the shared key and outputs an 'accept' message.

Figure 1

The handshake phase of MPSH-CM



- **Trace Member**. If GA finds that there exist malicious users causing damages to the system, it uses the message $M_{i,1}$, and finds the exact user with pseudonym x_i. This can be done because GA knows all pseudonyms of users and can use this knowledge to exhaustive verify whether

$$M_{i,1} = T_{x^* Q_p (Q_p + t)P}(M_{i,4}), \tag{17}$$

until a x^* which satisfies equation (17) is found, which means the identity of the user is found.

- **Revoke Member**. GA keeps a public revocation list L_{rev}, where GA has the right of read and write, while others can only read the list. The list consists of the items $rev = xP$. If a user u is being revoked for some reason (e.g., u carries out a DoS attack to the system), then GA traces the identity of u, i.e. the pseudonym $x_{u,p}$ and the item $rev = x_{u,p}P$ will be added to the list.

Security

In this section, it is proved that our protocol obeys the four security properties: Completeness, Detector Resistance, Impersonator Resistance and Unlinkability as introduced in Section 3.

Completeness

If all of the n users own the same property p, and execute the Handshake protocol honestly, it can be easily verified that $\sum_{i=1}^{n} X_i = 0$:

$$\begin{aligned}\sum_{i=1}^{n} X_i &= X_1 + X_2 + \ldots + X_n \\ &= T_{s_1^{-1} z_1^{-1} s_2^{-1} z_2^{-1}} T_{Q_p^2 P^3}(Q_p) - T_{s_1^{-1} z_1^{-1} s_n^{-1} z_n^{-1}} T_{Q_p^2 P^3}(Q_p) \\ &\quad + T_{s_2^{-1} z_2^{-1} s_3^{-1} z_3^{-1}} T_{Q_p^2 P^3}(Q_p) - T_{s_2^{-1} z_2^{-1} s_1^{-1} z_1^{-1}} T_{Q_p^2 P^3}(Q_p) + \ldots \\ &\quad + T_{s_n^{-1} z_n^{-1} s_1^{-1} z_1^{-1}} T_{Q_p^2 P^3}(Q_p) - T_{s_n^{-1} z_n^{-1} s_{n-1}^{-1} z_{n-1}^{-1}} T_{Q_p^2 P^3}(Q_p) \\ &= 0.\end{aligned} \tag{18}$$

Let

$$\begin{aligned}B_{i-1} &= T_{s_i^{-1} C_{i,2}} T_P(M_{i-1,2}) = T_{s_i^{-1} z_i^{-1} s_{i-1}^{-1} z_{i-1}^{-1}} T_{Q_p^2 P^3}(Q_p) \\ B_i &= T_{s_i^{-1} C_{i,2}} T_P(M_{i-1,2}) + X_i = T_{s_i^{-1} z_i^{-1} s_{i+1}^{-1} z_{i+1}^{-1}} T_{Q_p^2 P^3}(Q_p) \\ B_{i+1} &= T_{s_i^{-1} C_{i,2}} T_P(M_{i-1,2}) + X_i + X_{i+1} = T_{s_{i+1}^{-1} z_{i+1}^{-1} s_{i+2}^{-1} z_{i+2}^{-1}} T_{Q_p^2 P^3}(Q_p) \\ &\ldots \\ B_{i-2} &= T_{s_i^{-1} C_{i,2}} T_P(M_{i-1,2}) + X_i + \ldots + X_{i-2} \\ &= T_{s_{i-2}^{-1} z_{i-2}^{-1} s_{i-1}^{-1} z_{i-1}^{-1}} T_{Q_p^2 P^3}(Q_p).\end{aligned} \tag{19}$$

Then

$$\begin{aligned}&B_{i-1} + B_i + \ldots + B_{i-2} \\ &= n T_{s_i^{-1} C_{i,2}} T_P(M_{i-1,2}) + (n-1) X_i + (n-2) X_{i+1} + \ldots + X_{i-2} \\ &= k_i.\end{aligned} \tag{20}$$

As we defined before, the subscript i is set to $i \bmod n$, which means that

$$\begin{aligned}k_1 &= B_n + B_1 + \ldots B_{n-1} = \sum_{i=1}^{n} B_i, \\ k_2 &= B_1 + B_2 + \ldots B_n = \sum_{i=1}^{n} B_i, \\ &\ldots \\ k_n &= B_{n-1} + B_n + \ldots B_{n-2} = \sum_{i=1}^{n} B_i.\end{aligned} \tag{21}$$

That is to say that $k_1 = k_2 = \ldots = k_n$.

Detector resistance

For an adversary A, whose goal is to detect the affiliation information of a user without legal credential, we define a game denoted as **GameDetect. GameDetect** is executed between adversary A and a challenger B. It is developed as follows:

Init: A first sets $Choice = \{u_0, p_0, u_1, p_1\}$. Then B simulates the process of **Setup** and **Add Member**, and sets the public parameters and the revocation list of all groups public.

Queries: A queries B for handshake tuples $< cred_{u_i,p_i}, match_{p_i}, x_{u_i,p_i} >$ for a random number of given pairs $(u_i, p_i) \in \overline{U} \times \overline{P}, i \neq 0,1$. A is then free to take part in the MPSH-CM protocol with legal users.

Challenge: The challenger B picks a random bit $\phi \in \{0,1\}$. Note that A does not have the legal credential for either u_0 or u_1. Then A and B engage in the handshake protocol. B releases the message M_ϕ. A attempts to distinguish which property B owns.

Output: The adversary A outputs $\phi' = 0$ or 1 as its guess.

Analysis of *A*'s response:

If A can verify that the message M_ϕ contains the property p_0, then outputs $\phi' = 0$; if not, $\phi' = 1$ is output. All the possible messages that B might release are:

$$\begin{aligned} M_0 &= \{M_{0,1}, M_{0,2}, M_{0,3}, M_{0,4}, \tau_0\} \\ &= \{T_{s_0 x_{u_0,p_0} Q_{p_0} (Q_{p_0}+t)P}(Q_{p_0}), T_{s_0^{-1} z_0^{-1} Q_{p_0} P}(Q_{p_0}), \\ &\quad T_{s_0^{-1}(z_0 w)^{-1} Q_{p_0} P}(Q_{p_0}), \tau_0\}, \\ M_1 &= \{M_{1,1}, M_{1,2}, M_{1,3}, M_{1,4}, \tau_1\} \\ &= \{T_{s_1 x_{u_1,p_1} Q_{p_1} (Q_{p_1}+t)P}(Q_{p_1}), T_{s_1^{-1} z_1^{-1} Q_{p_1} P}(Q_{p_1}), \\ &\quad T_{s_1^{-1}(z_1 w)^{-1} Q_{p_1} P}(Q_{p_1}), \tau_1\}. \end{aligned} \quad (22)$$

According to the intractability of the DLP problem, it is hard to compute a with the knowledge of $T_a(x)$ and x. That is to say, with Q_{p_i} (which can be computed by A by the hash function H and property p), A can compute none of the three values: $s_i^{-1} z_i^{-1} Q_{p_i} P$, $s_i x_{u_i,p_i} Q_{p_i} (Q_{p_i} + t)P$ and $s_i^{-1}(z_i w)^{-1} Q_{p_i} P$, not to mention that the property p is hidden by the random elements s_i and z_i in the message. Or to say that A can verify whether the message M_ϕ contains p_0 or p_1 with the probability being

$$\begin{aligned} &\Pr[A \text{ verifies whether } \phi = 0 \text{ or } 1] \\ &\le \Pr[A \text{ solves DLP}] = \varepsilon, \end{aligned} \quad (23)$$

where ε is negligible. So A guesses ϕ' with the probability that

$$\Pr[\phi' = 0] = \Pr[\phi' = 1] = \frac{1}{2} + \varepsilon. \quad (24)$$

Thus, the probability that A wins the game **GameDetect** is:

$$\Pr[A \text{ wins GameDetect}] = \Pr[\phi' = \phi] = \frac{1}{2} + \varepsilon. \quad (25)$$

So A can win the game **GameDetect** with negligible advantage, and it comes to the conclusion that our protocol obeys the security property of detector resistance.

Impersonator resistance

For an adversary A, whose goal is to impersonate a legitimate user without owning a legal credential, we define a game denoted as **GameImp**. The game is executed between adversary A and a challenger B. It is developed as follows:

Init: A first sets $Choice = \{u_*, p_*\}$. Then B simulates the process of **Setup** and **Add Member**, and sets the public parameters and the revocation list public.

Queries: A queries B for handshake tuples $< cred_{u_i,p_i}, match_{p_i}, x_{u_i,p_i} >$ for a random number of given pairs $(u_i, p_i) \in \bar{U} \times \bar{P}$ except the *Choice* that A chooses, that is to say $(u_i, p_i) \neq (u_*, p_*)$. A is then free to take part in the MPSH-CM protocol with legal users. When the adversary A decides that the query phase is over, the challenger B revokes all the u_i that A queries.

Challenge: B acts as user u_* with property p_* and engages in the protocol with A. Note that A does not have the legal credential for u_*. A attempts to generate the correct key and let B believe that she is a legitimate user with property p_*.

Output: If the adversary A succeeds in computing a corresponding correct session key and executing the **Handshake** phase with B successfully, the output is "1". Otherwise, the game outputs "0". What's needed to be added is that A should pass the revocation check in order to successfully win the game, which means that A cannot use the identities that she queried before (they are all revoked soon after the **Query** phase is over).

Analysis of *A*'s response:

Here there are only two users taking part, so n equals 2. As A does not know the legal credential of any identity, she fakes one by choosing some numbers randomly and sends the following messages:

$$\begin{aligned} M_0 &= \{M_{0,1}, M_{0,1}, M_{0,3}, M_{0,4}, \tau_0\} \\ &= \{T_a(Q_{p_*}), T_b(Q_{p_*}), T_c(Q_{p_*}), T_d(Q_{p_*}), \tau_0\}, \end{aligned} \quad (26)$$

where $a, b, c, d, \tau_0 \in_R Z_q^*$. The message that B outputs is

$$M_1 = \{M_{1,1}, M_{1,2}, M_{1,3}, M_{1,4}, \tau_1\} = \{T_{s_1 x_{u_*,p_*} Q_{p_*}(Q_{p_*}+t)P}(Q_{p_*}), T_{s_1^{-1}z_1^{-1}Q_{p_*}P}(Q_{p_*}), T_{s_1^{-1}(z_1 w)^{-1}Q_{p_*}P}(Q_{p_*}), \tau_1\}. \quad (27)$$

Then the session key of B is computed as

$$k_1 = 2T_{s_1^{-1}C_{1,2}} T_P(M_{0,2}) + X_1 = T_{s_1^{-1}z_1^{-1}Q_{p_*}P^2}(M_{0,2}) + T_{s_1^{-1}z_1^{-1}Q_{p_*}WP}(M_{0,2}). \quad (28)$$

while the session key of A is computed as

$$k_0 = 2T_b(M_{1,2}) + X_0 = 2T_b T_P(Q_{p_*}). \quad (29)$$

If the handshake can be successfully executed, A must have computed the correct session key, which means that $k_1 = k_0$, that is to say A must choose the right b and c that satisfy the following equation:

$$k_1 = T_{s_1^{-1}z_1^{-1}Q_{p_*}P^2} T_b(Q_{p_*}) + T_{s_1^{-1}z_1^{-1}Q_{p_*}WP} T_c(Q_{p_*}) = k_0 = 2T_{s_1^{-1}z_1^{-1}Q_{p_*}P^2} T_b(Q_{p_*}), \quad (30)$$

which is equal to

$$T_c T_{s_1^{-1}z_1^{-1}Q_{p_*}WP}(Q_{p_*}) = T_b T_{s_1^{-1}z_1^{-1}Q_{p_*}P^2}(Q_{p_*}). \quad (31)$$

As A does not know the value of $s_1^{-1}z_1^{-1}$, we denote the above equation as

$$T_c(\alpha) = T_b(\beta), \quad (32)$$

where $\alpha = T_{s_1^{-1}z_1^{-1}Q_{p_*}WP}(Q_{p_*})$ and $\beta = T_{s_1^{-1}z_1^{-1}Q_{p_*}P^2}(Q_{p_*})$ are known for A.

Suppose that b is set to be one definite value. Then $T_b T_\beta(Q_{p_*})$ can be easily computed. We set the value of $T_b T_\beta(Q_{p_*})$ to be Δ. Then it remains to compute c that satisfies $T_c(\alpha) = \Delta$, which is obviously equal to solve the problem of DLP. That is to say that

$$\Pr[A \text{ computes the correct } b \text{ and } c] = \Pr[A \text{ solves DLP}] = \varepsilon, \quad (33)$$

with ε being negligible, which means that

$$\mathrm{Adv}_A^{\mathrm{GameImp}} = \Pr[\mathrm{GameImp} = 1] = \varepsilon. \quad (34)$$

So A can win the game **GameImp** with negligible advantage, concluding that our protocol obeys the security property of Impersonator Resistance.

Unlinkability

For an adversary A, whose goal is to verify whether two handshake instances are executed by the same user, we define a game named **Link**. The game is executed between A and a challenger B. A is able to engage in protocol executions, not to say eavesdropping the protocol instance. Note that the pseudonym x_{u_i,p_i} is the only element that associates with the identity, and it only appears in $M_{i,1}$ and can only be verified by equation (14) described above in our protocol. So we mainly concern the messages $M_{i,1}$ and $M_{i,4}$. The game **Link** develops as follows:

Init: B simulates the process of **Setup** and **Add Member**, and sets the public parameters and the revocation list of all groups public.

Queries: A queries B for handshake tuples $< cred_{u_i,p_i}, match_{p_i}, x_{u_i,p_i} >$ for a random number of given pairs $(u_i, p_i) \in \overline{U} \times \overline{P}$. A is then free to take part in the MPSH-CM protocol with legal users.

Challenge: The adversary A chooses the property p_* as the challenging property. The challenger B is given an instance $< T_a(Q_{p_*}), T_b(Q_{p_*}), T_\sigma(Q_{p_*}) >$ of the DHP problem in chaotic maps. B randomly picks $s \in Z_q^*$ and uses the property p_* to compute two handshake tuples as follows:

$$\{M_{0,1}, M_{0,4}\} = \{T_{sQ_{p_*}(Q_{p_*}+t)P} T_b(Q_{p_*}), T_{sQ_{p_*}P}(Q_{p_*})\}$$
$$\{M_{1,1}, M_{1,4}\} = \{T_{Q_{p_*}(Q_{p_*}+t)P} T_\sigma(Q_{p_1}), T_{Q_{p_*}P} T_a(Q_{p_*})\} \quad (35)$$

Output: If the adversary A concerns that the two messages are from the same user, she outputs "1"; otherwise, the output is "0".

Analysis of *A*'s response:

As explained before, only in the phase of verifying a user is revoked or not does the identity be verified. Indeed, if A wins the game, she can tell that the two messages contain the same identity (the pseudonym) x_*. Assume that A can win the game, then the same item $rev = x_{u,p}P$ can be used to revoke both credentials of

the two messages. According to equation (14), there exits the following system

$$\begin{cases} T_{(Q_{p_*}+t)^{-1}P} T_{sQ_{p_*}(Q_{p_*}+t)P} T_b(Q_{p_*}) = T_{x_*P} T_{sQ_{p_*}P}(Q_{p_*}) \\ T_{(Q_{p_*}+t)^{-1}P} T_{Q_{p_*}(Q_{p_*}+t)P} T_\sigma(Q_{p_*}) = T_{x_*P} T_{Q_{p_*}P}\, \mathrm{T}_a(Q_{p_*}) \end{cases} \quad (36)$$

According to the method of attack proposed by Bergamo *et al.*, we can solve the first equation as

$$x_* = \frac{\arccos(T_{sQ_{p_*}P^2} T_b(Q_{p_*})) + 2k\pi}{\arccos(Q_{p_*}) \cdot sQ_{p_*}P^2}, \quad (37)$$

we set $k = 0$ here for simplicity. Plug the value

$$x_* = \frac{\arccos(T_{sQ_{p_*}P^2} T_b(Q_{p_*}))}{\arccos(Q_{p_*}) \cdot sQ_{p_*}P^2} \quad (38)$$

into the second equation, one can get

$$T_{Q_{p_*}P^2} T_{ab}(Q_{p_*}) = T_{Q_{p_*}P^2} T_\sigma(Q_{p_*}), \quad (39)$$

which is the positive answer to the DHP problem.

So A can win the game with the advantage of

$$\mathrm{Adv}_A^{\mathrm{Link}} = \Pr[A \text{ solves DHP}] = \varepsilon \quad (40)$$

with ε being negligible. So it comes to the conclusion that our protocol obeys the property of unlinkability.

Performance

In this section, we analyze the performance of our protocol MPSH-CM. We define the following symbols for the convenience of computational cost evaluation as shown in Table 1.

According to [4, 7, 17], the quantitative relation of these symbols can be estimated approximately as: $T_m \approx T_h$, $T_c \approx 175T_h$, $T_e \approx 240T_h$, $T_{\hat{e}} \approx 1440T_h$. We can easily see that the cost of computing a Chebyshev polynomial is much lower than that of computing a bilinear map, and it is even lower than that of the exponentiation operation over a field. However, the bilinear map is literally in most common use in the field of designing secret handshakes currently. So our protocol is supposed to be more efficient than most of the multi-party secret handshake protocols. The quantitative relation of these symbols can also be used to convert one symbol to the other one. We compare our scheme with other existing group secret handshake schemes, as stated in Table 2. The computational cost of each user u_i is estimated by the symbols given in Table 1; the symbols are all converted to symbol T_h for comparison. As shown in Table 2, the computational cost per user of our scheme is approximately $(1404+n)T_h$, while it is much higher of other schemes listed in the table than that of ours. So we can come to the conclusion that our protocol can be executed with relatively low computational cost while maintaining the security properties that a secret handshake protocol requires.

Table 1
Symbols for computational cost evaluation

Symbols	Meanings
T_h	necessary time for performing a Hash function
T_e	necessary time for performing the exponentiation over a field
T_c	necessary time for performing a Chebyshev polynomial
$T_{\hat{e}}$	necessary time for performing the bilinear map
T_m	necessary time for performing the multiplication over a field

Table 2
Computational cost comparison

Schemes	Computation cost(per user)
Ours	$(n+4)T_m + 8T_c + T_h \approx (1404+n)T_h$
Schemes 1 [14]	$(n+6)T_e + (n+3)T_m + 3T_h \approx (1446+241n)T_h$
Schemes 2 [14]	$(n+4)T_e + (n+2)T_m + 5T_h \approx (967+241n)T_h$
Schemes 3 [40]	$4T_{\hat{e}} + 2T_e + (3n+8)T_m + (n+2)T_h \approx (6250+4n)T_h$
Schemes 4 [33]	$4T_{\hat{e}} + O(n)T_e \approx (5560+240\cdot O(n))T_h$
Schemes 5 [32]	$T_{\hat{e}} + O(n)T_e + 2\cdot O(n)T_m \approx (1440+242\cdot O(n))T_h$

Conclusions

We presented a multi-party secret handshake scheme MPSH-CM. We proved that the MPSH-CM scheme realizes the security property of detector resistance, impersonator resistance and unlinkability, based on the property of chaotic map in addition with the intractability of the DLP and Diffie-Hellman Problem (DHP). What is more, the MPSH-CM scheme provides the ability of tracing the identity of users for GA in concern of malicious users, thus avoiding the attacks from the inside users (e.g., the DoS attack). The usage of the chaotic maps instead of the most common used bilinear maps, causes a significant reduction in the computation cost. Comparing with other existing group secret handshake schemes, our scheme performs better in computational cost, which indicates that our scheme is more suitable for actual applications.

Acknowledgments

The authors would like to thank the editor and anonymous referees for their helpful comments on this paper. This work is funded by open project funding of the State Key Laboratory of Cryptology (MMKFKT 201514).

References

1. D. Balfanz, G. Durfee, N. Shankar, D. Smetters, J. Staddon, H. Wong. Secret handshakes from pairing-based key agreements. IEEE Symposium on Security and Privacy, Berkeley, CA, USA, IEEE Computer Society, 2003, 180-196. https://doi.org/10.1109/secpri.2003.1199336
2. P. Bergamo, P. D'Arco, A. De Santis, L. Kocarev. Security of public-key cryptosystems based on Chebyshev polynomials. IEEE Transactions on Circuits and Systems I: Regular Papers, 2005, 52(7), 1382-1393. https://doi.org/10.1109/TCSI.2005.851701
3. C. Castelluccia, S. Jarecki, G. Tsudik. Secret handshakes from CA-oblivious encryption. Cryptology-ASIACRYPT 2004. LNCS, Springer, Berlin-Heidelberg, 2004, 3329, 293-307.
4. C. Fan, W. Sun, V. Huang. Provably secure randomized blind signature scheme based on bilinear pairing. Computers & Mathematics with Applications, 2010, 60(2), 285-293. https://doi.org/10.1016/j.camwa.2010.01.021
5. J. Gu, Z. Xue. An improved efficient secret handshakes scheme with unlinkability. IEEE Communications Letters, 2011, 15(2), 259-261. https://doi.org/10.1109/LCOMM.2011.122810.102229
6. C. Guo, C. C. Chang. Chaotic maps-based password authenticated key agreement using smart cards. Communications in Nonlinear Science Numerical Simulation, 2013, 18(6), 1433-1440. https://doi.org/10.1016/j.cnsns.2012.09.032

7. D. Guo, Q. Wen, W. Li, H. Zhang, Z. Jin. Analysis and improvement of ‚chaotic map based mobile dynamic ID authenticated key agreement scheme'. Wireless Personal Communications, 2015, 83(1), 35-48. https://doi.org/10.1007/s11277-015-2378-2
8. P. Guo, J. Wang, X. H. Geng, C. S. Kim, J.-U. Kim. A variable threshold-value authentication architecture for wireless mesh networks. Journal of Internet Technology, 2014, 15(6), 929-936.
9. D. B. He, N. Kumar, J. H. Chen, C. -C. Lee, N. Chilamkurti, S.-S. Yeo. Robust anonymous authentication protocol for health-care applications using wireless medical sensor networks. Multimedia Systems, 2015, 21(1), 49-60. https://doi.org/10.1007/s00530-013-0346-9
10. D. B. He, N. Kumar, N. Chilamkurti. A secure temporal-credential-based mutual authentication and key agreement scheme with pseudo identity for wireless sensor networks. Information Sciences, 2015, 321, 263-277. https://doi.org/10.1016/j.ins.2015.02.010
11. D. B. He, S. Zeadally, B. W. Xu, X. Y. Huang. An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks. IEEE Transactions on Information Forensics and Security, 2015, 10(12), 2681-2691. https://doi.org/10.1109/TIFS.2015.2473820
12. H. Huang, Z. Cao. A novel and efficient unlinkable secret handshakes scheme. IEEE Communications Letters, 2009, 13(5), 363-365. https://doi.org/10.1109/LCOMM.2009.081880
13. S. Jarecki, J. Kim, G. Tsudik. Beyond secret handshakes: affiliation-hiding authenticated key exchange. CT-RSA 2008. Moscone Center, San Francisco, CA, USA, LNCS 4964, Springer, Berlin-Heidelberg, 2008, 352-369.
14. S. Jarecki, J. Kim, G. Tsudik. Group secret handshakes or affiliation-hiding authenticated group key agreement. CT-RSA 2007, San Francisco, CA, USA, LNCS 4377, Springer, Berlin-Heidelberg, 2007, 287-308.
15. S. Jarecki, X. Liu. Private mutual authentication and conditional oblivious transfer. Advances in Cryptology-CRYPTO 2009, Santa Barbara, CA, USA, LNCS 5677, Springer, Berlin-Heidelberg, 2009, 90-107.
16. S. Jarecki, X. Liu. Unlinkable secret handshakes and key-private group key management schemes. ACNS 2007, Zhuhai, China, LNCS, Springer, Berlin-Heidelberg, 2007, 4521, 270-287. https://doi.org/10.1007/978-3-540-72738-5_18
17. N. Koblitz, A. Menezes, S. Vanstone. The state of elliptic curve cryptography. Designs, Codes and Cryptography, 2000, 19(2-3), 173-193. https://doi.org/10.1023/A:1008354106356
18. L. Kocarev, Z. Tasev. Public-key encryption based on Chebyshev maps. 2003 IEEE Symposium on Circuits and Systems (ISCAS ,03), IEEE Computer Society, 2003, 3, 28-31. https://doi.org/10.1109/iscas.2003.1204947
19. T. Kohda, A. Tsuneda, A. J. Lawrance. Correlational properties of Chebyshev chaotic sequences. Journal of time series analysis, 2000, 21(2), 181-191. https://doi.org/10.1111/1467-9892.00180
20. T. Kohda, A. Tsuneda. Pseudonoise sequences by chaotic nonlinear maps and their correlation properties. IEICE Transactions on Communications, 1993, 76(8), 855-862.
21. C. C. Lee, C. T. Li, S. T. Chiu, Y. M. Lai. A new three-party-authenticated key agreement scheme based on chaotic maps without password table. Nonlinear Dynamics, 2015, 79(4), 2485-2495. https://doi.org/10.1007/s11071-014-1827-x
22. C. C. Lee, C. W. Hsu. A secure biometric-based remote user authentication with key agreement scheme using extended chaotic maps. Nonlinear Dynamics, 2013, 71(1-2), 201-211. https://doi.org/10.1007/s11071-012-0652-3
23. C. C. Lee, D. C. Lou, C. T. Li, C. W. Hsu. An extended chaotic maps-based protocol with key agreement for multi-server environments. Nonlinear Dynamics, 2014, 76(1), 853-866. https://doi.org/10.1007/s11071-013-1174-3
24. M. Manulis, B. Pinkas, B. Poettering. Privacy-preserving group discovery with linear complexity. Applied Cryptography and Network Security, Beijing, China, LNCS, Springer, Berlin-Heidelberg, 2010, 6123, 420-437. https://doi.org/10.1007/978-3-642-13708-2_25
25. M. Manulis, B. Poettering, G. Tsudik. Affiliation-hiding key exchange with untrusted group authorities. Applied Cryptography and Network Security, Beijing, China, LNCS, Springer, Berlin-Heidelberg, 2010, 6123, 402-419. https://doi.org/10.1007/978-3-642-13708-2_24
26. J. C. Mason, D. C. Handscomb. Chebyshev Polynomials. In: Chapman & Hall/CRC Press, London, 2003.
27. J. Shen, H. W. Tan, J. Wang, J. W. Wang, S. Y. Lee. A novel routing protocol providing good transmission reliability in underwater sensor networks. Journal of Internet Technology, 2015, 16(1), 171-178.
28. A. Sorniotti, R. Molva. Secret handshakes with revocation support. Information, Security and Cryptology-ICISC 2009, Seoul, Korea, LNCS, Springer,

Berlin-Heidelberg, 2010, 5984, 274-299. https://doi.org/10.1007/978-3-642-14423-3_19

29. G. Tsudik, S. Xu. A flexible framework for secret handshakes. Privacy Enhancing Technologies, Cambridge, UK, LNCS, Springer, Berlin-Heidelberg, 2006, 4258, 295-315. https://doi.org/10.1007/11957454_17
30. D. Xiao, X. Liao, S. Deng. A novel key agreement protocol based on chaotic maps. Information Sciences, 2007, 177(4), 1136-1142. https://doi.org/10.1016/j.ins.2006.07.026
31. Q. Xie, J. Zhao, X. Yu. Chaotic maps-based three-party password-authenticated key agreement scheme. Nonlinear Dynamics, 2013, 74(4), 1021-1027. https://doi.org/10.1007/s11071-013-1020-7
32. C. Xu, H. Guo, Z. Li, Y. Mu. New construction of affiliation-hiding authenticated group key agreement. Security and Communication Networks, 2013, 6(6), 723-734. https://doi.org/10.1002/sec.606
33. C. Xu, L. Zhu, Z. Li, F. Wang. One-round affiliation-hiding authenticated asymmetric group key agreement with semi-trusted group authority. The Computer Journal, 2015, 58(10), 2509-2519. https://doi.org/10.1093/comjnl/bxu099
34. S. Xu, M. Yung. K-anonymous secret handshakes with reusable credentials. 11th ACM Conference on Computer and Communications Security (CCS 2004), Washington DC, USA, 2004, 158-167. https://doi.org/10.1145/1030083.1030105
35. C. Xu, Z. Li, Y. Mu, H. Guo, T. Guo. Affiliation-hiding authenticated asymmetric group key agreement. The Computer Journal, 2012, 55(10), 1180-1191. https://doi.org/10.1093/comjnl/bxs022
36. K. Xue, P. Hong. Security improvement on an anonymous key agreement protocol based on chaotic maps. Communications in Nonlinear Science Numerical Simulation, 2012, 17(7), 2969-2977. https://doi.org/10.1016/j.cnsns.2011.11.025
37. E. Yoon. Efficiency and security problems of anonymous key agreement protocol based on chaotic maps. Communications in Nonlinear Science Numerical Simulation, 2012, 17(7), 2735-2740. https://doi.org/10.1016/j.cnsns.2011.11.010
38. L. Zhang. Cryptanalysis of the public key encryption based on multiple chaotic systems. Chaos, Solitons & Fractals, 2008, 37(3), 669-674. https://doi.org/10.1016/j.chaos.2006.09.047
39. F. Zhao, P. Gong, S. Li, M. Li, P. Li. Cryptanalysis and improvement of a three-party key agreement protocol using enhanced Chebyshev polynomials. Nonlinear Dynamics, 2013, 74(1-2), 419-427. https://doi.org/10.1007/s11071-013-0979-4
40. L. Zhou, W. Susilo, Y. Mu. New construction of group secret handshakes based on pairings. Information and Communications Security, Zhengzhou, China LNCS, Springer, Berlin-Heidelberg, 2007, 4861, 16-30. https://doi.org/10.1007/978-3-540-77048-0_2

Summary / Santrauka

The primitive secret handshake is a kind of privacy-preserving authentication protocol, in which the participants can share a common session key if and only if they come from the same group, without the leakage of the group information. Most of the current secret handshakes are realized by means of bilinear maps, whose computational cost is a lot. A new multi-party secret handshake scheme is proposed in this paper using the chaotic map, with the computational cost reducing significantly. The new protocol also supports user revocation, and has the ability of tracing users, meanwhile proved to achieve the basic security properties of secret handshakes.

Slaptas pasisveikinimas yra toks privatumą saugantis autentiškumo nustatymo protokolas, kuriame dalyviai dalinasi bendru sesijos raktu, jeigu (ir tik jeigu) jie priklauso tai pačiai grupei. Tokiu būdu nenuteka jokia tai grupei priklausanti informacija. Dauguma dabartinių slaptųjų pasisveikinimų yra išreikšti dvitiesių žemėlapių pagalba, kurių skaičiuojamoji kaina yra didelė. Nauja daugiašalio slaptojo pasisveikinimo schema, kuri pristatoma šiame straipsnyje, naudoja chaotiškuosius žemėlapius (angl. Chaotic maps), ir tai yra būdas gerokai sumažinti išlaidas. Naujas protokolas taip pat leidžia panaikinti vartotojų prieigą bei turi galimybę atsekti vartotojus, tuo pačiu metu užtikrindamas pagrindines slaptųjų pasisveikinimų apsaugos ypatybes.