Journal of Information Hiding and Multimedia Signal Processing ©2016 ISSN 2073-4212
Ubiquitous International Volume **7**, Number **6**, November **2016**

Security Improvement on A Three Party Password Based Authenticated Key Exchange Scheme Using Chaotic Maps

Yu Chen[1], Chien-Ming Chen[2], Jeng-Shyang Pan[1,2], Tsu-Yang Wu[2*] and Shuai Liu[2]

[1]Fujian Provincial Key Laboratory of Big Data Mining and Applications
Fujian University of Technology, Fuzhou 350118, China

[2]Innovative Information Industry Research Center, School of Computer Science and Technology
Shenzhen Graduate School, Harbin Institute of Technology
Shenzhen 518055, China

cheny@fjut.edu.cn, chienming.taiwan@gmail.com, jengshyangpan@gmail.com, liush000@hotmail.com
*Corresponding author's email: wutsuyang@gmail.com

Received May, 2016; revised September, 2016

Abstract. *Recently, password based authenticated key exchange (or called PAKE for short) with chaotic maps has been received much attention for researchers. In 2013, Xie et al. proposed a three party PAKE scheme (based on chaotic maps). However, their scheme had been proved insecure by Lee et al. in 2015. In this paper, we first show that Xie et al.'s scheme is also vulnerable to a replay attack. To overcome this attack, we further propose an improvement based on their scheme. Finally, we demonstrate the security of our improvement with the BAN logic.*

Keywords: Three party PAKE, Chaotic maps, Cryptanalysis, BAN logic

1. **Introduction.** In order to protect the transmitted messages over a public channel, it is required to encrypt under a secure key. Authenticated key exchange (AKE) is a cryptographic primitive. It allows participants not only establishing a secure session key but also authenticated each other. Thus, various AKE schemes and protocols have been proposed such as password based AKE (or called PAKE for short).

Recently, PAKE schemes and protocols based on chaotic maps have been received much attention for researchers because they only use the participants' passwords and the properties of chaos to achieve the session key establishment and the entities authentication. PAKE (based on chaotic maps) was firstly designed for two party situation. We call that 2PAKE [1, 2, 3] which is suitable for the client-server environments rather than the client-client environments. For this reason, three party password based authenticated key exchange (or called 3PAKE for short) has been described. Various 3PAKE schemes and protocols have been proposed in [4, 5, 6, 7, 8, 9, 10, 11, 12, 13].

In 2013, Xie et al. [9] proposed a 3PAKE scheme based on chaotic maps. However, Lee et al. [12] found some security flaws in their scheme. In this paper, we first demonstrate that Xie et al.'s scheme is also vulnerable to a replay attack. Then, we propose an improvement based on their scheme. Finally, we adopt the BAN logic [14] to demonstrate the security of our improvement.

2. **Chaotic maps.** In this section, we introduce the basic concepts of the chaotic maps. We select the extended Chebyshev polynomial [15] which developed from the Chebyshev polynomial [16]

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

as an instantiation of the chaotic map. By the recursive approach, we can obtain some examples of the Chebyshev's polynomial: $T_2(x) = 2x \cdot T_1(x) - T_0(x) = 2x^2 - 1$, $T_3(x) = 2x \cdot T_2(x) - T_1(x) = 4x^3 - 3x$, $T_4(x) = 2x \cdot T_3(x) - T_2(x) = 8x^4 - 8x^2 + 1$.

It is easy to see that $T_n(x)$ is a polynomial of degree n. If the variable $x \in [-1, 1]$, then it implies $T_n(x) \in [-1, 1]$. Hence, we can define a special case of the Chebyshev polynomial $T_n(x) : [-1, 1] \to [-1, 1]$ by $T_n(x) = \cos(n \cdot \arccos(x))$. For $n \geq 2$ the Chebyshev polynomial $T_n(x)$ satisfies the following two properties:

(1) The semi group property.

$$\begin{aligned} T_a(T_b(x)) &= \cos(a \cdot \arccos(\cos(b \cdot \arccos(x)))) \\ &= \cos(ab \cdot \arccos(x)) \\ &= T_{ab}(x) \\ &= T_b(T_a(x)) \end{aligned}$$

for any positive integers a, b and $x \in [-1, 1]$.

(2) The chaotic property. $T_n(x)$ is a prototype of a chaotic map. It has a unique absolutely continuous invariant measure $\mu(x) = \frac{1}{\pi\sqrt{1-x^2}}$ with positive Lyapunov exponent $\lambda = \ln n$.

An enhanced Chebeshev's polynomial is defined on $(-\infty, \infty)$, $T_n(x) \equiv 2x \cdot T_{n-1}(x) - T_{n-2}(x) \bmod p$ for $n \geq 2$ and p is a large prime while the semi group property, $T_a(T_b(x)) \equiv T_{ab}(x) \equiv T_b(T_a(x) \bmod p$ for any a, $b \geq 2$ still holds.

3. **Cryptanalysis of Xie et al.'s scheme.**

3.1. **A briefly review.** Here, we briefly review Xie et al.'s scheme [9]. The steps of their scheme are depicted in Fig. 1.

Assume there are two participants U_A and U_B desire to establish a session key SK through a trusted server S. Note that the server's private key is k and the corresponding public key is $(x, T_k(x))$ based on chaotic maps. U_A and U_B share their passwords pw_A and pw_B with S. The detailed steps are describe as follows.

Step 1. U_A selects a random a and computes $K_{AS} = T_a(T_k(x))$, $H_A = h(T_a(x)||ID_A||ID_B||pw_A)$, and $C_1 = E_{K_{AS}}(ID_A||ID_B||H_A)$, where $h()$ denotes a one-way hash function based on chaotic maps, $E_K()$ denotes a secure symmetric encryption function with key K, and the identities of U_A and U_B are denoted by ID_A and ID_B. Then, U_A sends $m_1 = \{T_a(x), ID_A, C_1\}$ to U_B.

Step 2. Upon receiving m_1, U_B selects a random b and computes $K_{BS} = T_b(T_k(x))$, $H_B = h(T_b(x)||ID_B||ID_A||pw_B)$, and $C_2 = E_{K_{BS}}(ID_B||ID_A||H_B)$. Then, U_B sends $\{m_1, m_2\}$ to S, where $m_2 = \{T_b(x), ID_B, C_2\}$.

Step 3. Upon receiving $\{m_1, m_2\}$, S first computes $K_{SA} = T_k(T_a(x))$, $D_{K_{SA}}(C_1) = \{ID_A, ID_B, H_A\}$, $K_{SB} = T_k(T_b(x))$, and $D_{K_{SB}}(C_2) = \{ID_B, ID_A, H_B\}$, where $D_K()$ denotes a secure symmetric decryption function with key K. Then, S verifies whether $H_A = h(T_a(x)||ID_A||ID_B||pw_A)$ and $H_B = h(T_b(x)||ID_B||ID_A||pw_B)$ hold. If both hold, S computes $H_{SB} = h(T_a(x)||pw_B)$, $C_3 = E_{K_{SB}}(ID_B||ID_A||T_a(x)||H_{SB})$, $H_{SA} = h(T_b(x)||pw_A)$, and $C_4 = E_{K_{SA}}(ID_A||ID_B||T_b(x)||H_{SA})$. Then, S sends $m_3 = \{C_3, C_4\}$ to U_B.

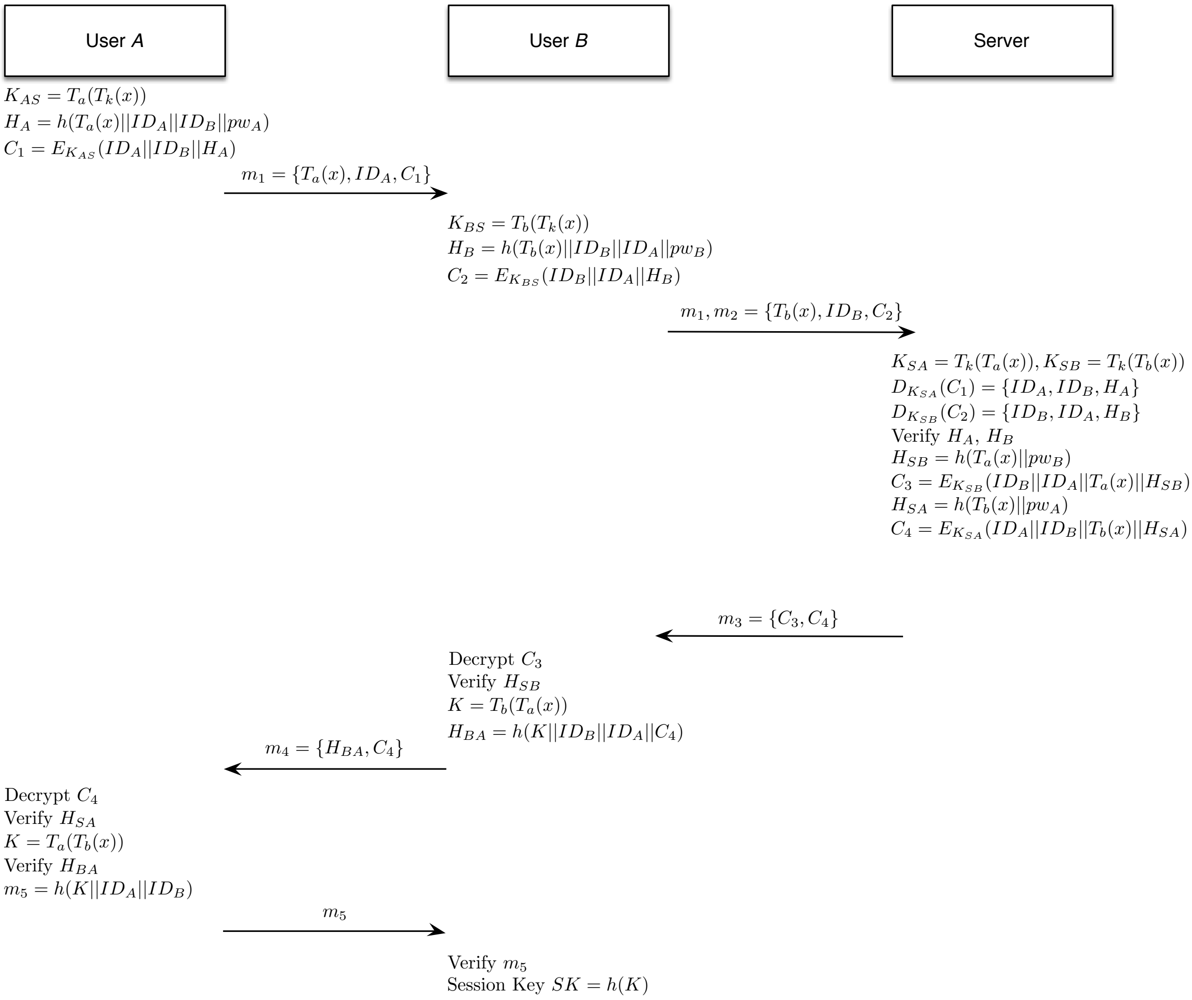
User A
User B
Server
$K_{AS} = T_a(T_k(x))$
$H_A = h(T_a(x)||ID_A||ID_B||pw_A)$
$C_1 = E_{K_{AS}}(ID_A||ID_B||H_A)$
$m_1 = \{T_a(x), ID_A, C_1\}$
$K_{BS} = T_b(T_k(x))$
$H_B = h(T_b(x)||ID_B||ID_A||pw_B)$
$C_2 = E_{K_{BS}}(ID_B||ID_A||H_B)$
$m_1, m_2 = \{T_b(x), ID_B, C_2\}$
$K_{SA} = T_k(T_a(x)), K_{SB} = T_k(T_b(x))$
$D_{K_{SA}}(C_1) = \{ID_A, ID_B, H_A\}$
$D_{K_{SB}}(C_2) = \{ID_B, ID_A, H_B\}$
Verify H_A, H_B
$H_{SB} = h(T_a(x)||pw_B)$
$C_3 = E_{K_{SB}}(ID_B||ID_A||T_a(x)||H_{SB})$
$H_{SA} = h(T_b(x)||pw_A)$
$C_4 = E_{K_{SA}}(ID_A||ID_B||T_b(x)||H_{SA})$
$m_3 = \{C_3, C_4\}$
Decrypt C_3
Verify H_{SB}
$K = T_b(T_a(x))$
$H_{BA} = h(K||ID_B||ID_A||C_4)$
$m_4 = \{H_{BA}, C_4\}$
Decrypt C_4
Verify H_{SA}
$K = T_a(T_b(x))$
Verify H_{BA}
$m_5 = h(K||ID_A||ID_B)$
m_5
Verify m_5
Session Key $SK = h(K)$

FIGURE 1. Xie et al.'s scheme

Step 4. Upon receiving m_3, U_B computes $D_{K_{BS}}(C_3) = \{ID_B, ID_A, T_a(x), H_{SB}\}$ and verifies whether $H_{SB} = h(T_a(x)||pw_B)$ holds. If it holds, U_B computes $K = T_b(T_a(x))$ and $H_{BA} = h(K||ID_B||ID_A||C_4)$. Then, U_B sends $m_4 = \{H_{BA}, C_4\}$ to U_A.

Step 5. Upon receiving m_4, U_A computes $D_{K_{AS}}(C_4) = \{ID_A, ID_B, T_b(x), H_{SA}\}$ and verifies whether $H_{SA} = h(T_b(x)||pw_A)$ holds. If it holds, U_A computes $K = T_a(T_b(x))$ and verifies whether $H_{BA} = h(K||ID_B||ID_A||C_4)$ holds. If it holds, U_A computes $m_5 = h(K||ID_A||ID_B)$ and sends it to U_B.

Step 6. Upon receiving m_5, U_B verifies whether $m_5 = h(K||ID_A||ID_B)$. If it holds, U_A and U_B share a session key $SK = h(K)$.

3.2. **Replay attack.** Here, we demonstrate a replay attack in Xie et al.'s scheme [9]. Assume that there is a passive adversary $\mathcal{A}$ eavesdrops $m_1 = \{T_a(x), ID_A, C_1\}$ from U_A and $m_2 = \{T_b(x), ID_B, C_2\}$ from U_B. Later, $\mathcal{A}$ sends $\{m_1, m_2\}$ to S. Since these two messages m_1 and m_2 are generated by honest participants, the identities of U_A and U_B are authenticated by S although U_A and U_B are not intended to initiate a protocol. It shows that $\mathcal{A}$ can successfully convince S to authenticate two phantom participants in initiating the protocol. Thus, we can conclude that under this replay attack, S still cannot confirm whether U_A and U_B are really initiating a 3AKE scheme or not. Sequently, the

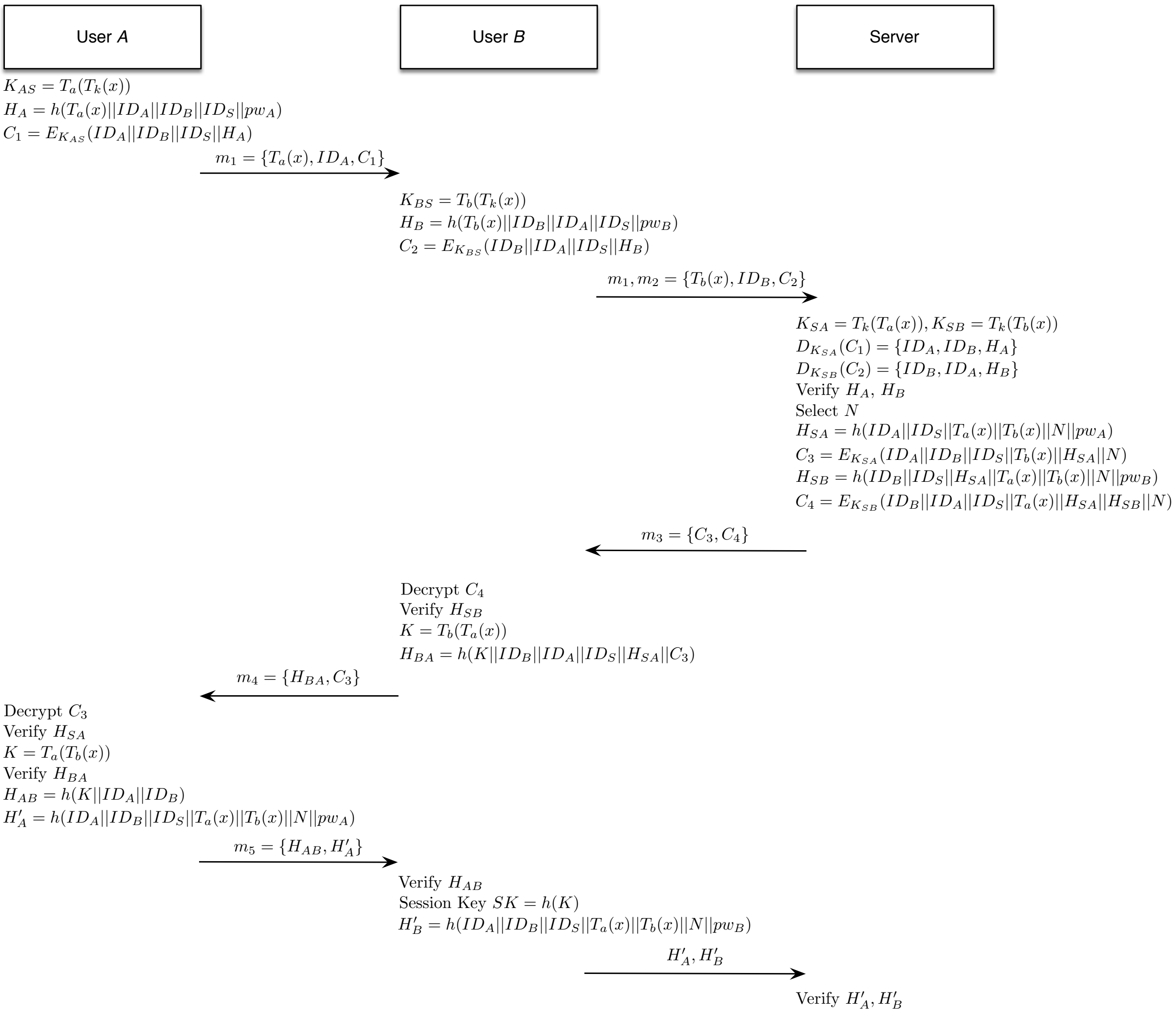
User A
User B
Server
$K_{AS} = T_a(T_k(x))$
$H_A = h(T_a(x)||ID_A||ID_B||ID_S||pw_A)$
$C_1 = E_{K_{AS}}(ID_A||ID_B||ID_S||H_A)$
$m_1 = \{T_a(x), ID_A, C_1\}$
$K_{BS} = T_b(T_k(x))$
$H_B = h(T_b(x)||ID_B||ID_A||ID_S||pw_B)$
$C_2 = E_{K_{BS}}(ID_B||ID_A||ID_S||H_B)$
$m_1, m_2 = \{T_b(x), ID_B, C_2\}$
$K_{SA} = T_k(T_a(x)), K_{SB} = T_k(T_b(x))$
$D_{K_{SA}}(C_1) = \{ID_A, ID_B, H_A\}$
$D_{K_{SB}}(C_2) = \{ID_B, ID_A, H_B\}$
Verify H_A, H_B
Select N
$H_{SA} = h(ID_A||ID_S||T_a(x)||T_b(x)||N||pw_A)$
$C_3 = E_{K_{SA}}(ID_A||ID_B||ID_S||T_b(x)||H_{SA}||N)$
$H_{SB} = h(ID_B||ID_S||H_{SA}||T_a(x)||T_b(x)||N||pw_B)$
$C_4 = E_{K_{SB}}(ID_B||ID_A||ID_S||T_a(x)||H_{SA}||H_{SB}||N)$
$m_3 = \{C_3, C_4\}$
Decrypt C_4
Verify H_{SB}
$K = T_b(T_a(x))$
$H_{BA} = h(K||ID_B||ID_A||ID_S||H_{SA}||C_3)$
$m_4 = \{H_{BA}, C_3\}$
Decrypt C_3
Verify H_{SA}
$K = T_a(T_b(x))$
Verify H_{BA}
$H_{AB} = h(K||ID_A||ID_B)$
$H'_A = h(ID_A||ID_B||ID_S||T_a(x)||T_b(x)||N||pw_A)$
$m_5 = \{H_{AB}, H'_A\}$
Verify H_{AB}
Session Key $SK = h(K)$
$H'_B = h(ID_A||ID_B||ID_S||T_a(x)||T_b(x)||N||pw_B)$
H'_A, H'_B
Verify H'_A, H'_B

FIGURE 2. Our improvement

impact of this vulnerability will lead to an incorrect login status to the server and may releasing unauthorized access of content, for example.

4. **Improved scheme.** In this section, we propose an improvement for Xie et al.'s scheme which can resist the mentioned replay attack. Note that the notations and parameters are same as ones defined in Xie et al.'s scheme. The steps of our improvement are depicted in Fig. 2. The detailed steps are described as follows.

Step 1. U_A selects a random a and computes $K_{AS} = T_a(T_k(x))$, $H_A = h(T_a(x)||ID_A|| ID_B||ID_S||pw_A)$, and $C_1 = E_{K_{AS}}(ID_A||ID_B||ID_S||H_A)$, where ID_S denotes the identity of server S. Then, U_A sends $m_1 = \{T_a(x), ID_A, C_1\}$ to U_B.

Step 2. Upon receiving m_1, U_B selects a random b and computes $K_{BS} = T_b(T_k(x))$, $H_B = h(T_b(x)||ID_B||ID_A||ID_S||pw_B)$, and $C_2 = E_{K_{BS}}(ID_B||ID_A||ID_S||H_B)$. Then, U_B sends $\{m_1, m_2\}$ to S, where $m_2 = \{T_b(x), ID_B, C_2\}$.

Step 3. Upon receiving $\{m_1, m_2\}$, S first computes $K_{SA} = T_k(T_a(x))$, $D_{K_{SA}}(C_1) = \{ID_A, ID_B, H_A\}$, $K_{SB} = T_k(T_b(x))$, and $D_{K_{SB}}(C_2) = \{ID_B, ID_A, H_B\}$. Then, S verifies whether $H_A = h(T_a(x)||ID_A||ID_B||ID_S||pw_A)$ and $H_B = h(T_b(x)||ID_B||ID_A||ID_S||pw_B)$ hold. If both hold, S select a nonce N and computes $H_{SA} = h(ID_A||ID_S||T_a(x)||T_b(x)||N|| pw_A)$, $C_3 = E_{K_{SA}}(ID_A||ID_B||ID_S||T_b(x)||H_{SA}||N)$, $H_{SB} = h(ID_B||ID_S||H_{SA}||T_a(x)||$

$T_b(x)||N||pw_B)$, and $C_4 = E_{K_{SB}}(ID_B||ID_A||ID_S||T_a(x)||H_{SA}||H_{SB}||N)$. Then, S sends $m_3 = \{C_3, C_4\}$ to U_B.

Step 4. Upon receiving m_3, U_B computes $D_{K_{BS}}(C_4) = \{ID_B, ID_A, ID_S, T_a(x), H_{SA}, H_{SB}, N\}$ and verifies whether $H_{SB} = h(ID_B||ID_S||H_{SA}||T_a(x)||T_b(x)||N||pw_B)$ holds. If it holds, U_B computes $K = T_b(T_a(x))$ and $H_{BA} = h(K||ID_B||ID_A||ID_S||H_{SA}||C_3)$. Then, U_B sends $m_4 = \{H_{BA}, C_3\}$ to U_A.

Step 5. Upon receiving m_4, U_A computes $D_{K_{AS}}(C_3) = \{ID_A, ID_B, ID_S, T_b(x), H_{SA}, N\}$ and verifies whether $H_{SA} = h(ID_A||ID_S||T_a(x)||T_b(x)||N||pw_A)$ holds. If it holds, U_A computes $K = T_a(T_b(x))$ and verifies whether $H_{BA} = h(K||ID_B||ID_A||ID_S||H_{SA}||C_3)$ holds. If it holds, U_A computes $H_{AB} = h(K||ID_A||ID_B)$ and $H'_A = h(ID_A||ID_B||ID_S||T_a(x)|| T_b(x)||N|| pw_A)$. Then, U_A sends $m_5 = \{H_{AB}, H'_A\}$ to U_B.

Step 6. Upon receiving m_5, U_B verifies whether $H_{AB} = h(K||ID_A||ID_B)$. If it holds, U_A and U_B share a session key $SK = h(K)$. Then, U_B computes $H'_B = h(ID_A||ID_B||ID_S|| T_a(x)||T_b(x)||N||pw_B)$ and sends $m_6 = \{H'_A, H'_B\}$ to S.

Step 7. Upon receiving m_6, S verifies whether $H'_A = h(ID_A||ID_B||ID_S||T_a(x)||T_b(x)|| N||pw_A)$ and $H'_B = h(ID_A||ID_B||ID_S||T_a(x)||T_b(x)||N||pw_B)$ hold. If both holds, S can assure that U_A and U_B have successfully established a common session key.

5. Security analysis.

The BAN logic [14] is widely used to analyze the security of authenticated key agreement protocols. Here, we demonstrate the security of our improvement using the BAN logic. Firstly, we define some notations and rules about BAN logic as follows:

5.1. Notations.

1. $P \mid\equiv X$: P believes X or called P would be entitled to believe X. In particular, P may act as though X is true.
2. $P \lhd X$: P sees X. Someone has sent a message containing X to P and P can read and repeat X.
3. $P \mid\sim X$: P once said X. P sent a message including X at some time. Note that it does not know whether the message was sent long ago or during the current run of the protocol, but it knows that $P \mid\equiv X$ when the message was sent.
4. $P \mid\Rightarrow X$: P has jurisdiction over X. P controls X which is subject to jurisdiction of P and P is trusted for X.
5. $\sharp(X)$: X is fresh. X has not been sent in a message at any time before the execution of current round of the protocol.
6. $P \overset{K}{\longleftrightarrow} Q$: P and Q may use the shared key K to communicate securely. We say that K is good, if K will never be discovered by any principal except P or Q, or a principal trusted by either P or Q.
7. $P \overset{X}{\rightleftharpoons} Q$: The formula X is a secret known only to P and Q, and possibly to principals trusted by P and Q.
8. $\{X\}_K$: The formula X is encrypted under a key K.
9. $\langle X\rangle_Y$: The formula X is combined with a secret Y.

5.2. Rules.

1. Message meaning rule for shared keys: $\dfrac{P \mid\equiv Q \overset{K}{\longleftrightarrow} P, P \lhd \{X\}_K}{P \mid\equiv Q \mid\sim X}$. It means that if P believes that K is a shared key with Q and P sees X encrypted under K, then P believes that Q once said X.

2. Message meaning rule for shared secrets: $\frac{P \mid\equiv Q \overset{Y}{\rightleftharpoons} P, P \triangleleft \langle X \rangle_Y}{P \mid\equiv Q \mid\sim X}$. It means that if P believes that Y is a secret known only to P and Q and P sees X under Y, then P believes that Q once said X.
3. Nonce verification rule: $\frac{P \mid\equiv \sharp(X), P \mid\equiv Q \mid\sim X}{P \mid\equiv Q \mid\equiv X}$. It means that if P believes that X is fresh and Q once said X, then P believes Q believes X.
4. Jurisdiction rule: $\frac{P \mid\equiv Q \mid\Rightarrow X, P \mid\equiv Q \mid\equiv X}{P \mid\equiv X}$. It means that if P believes that Q has jurisdiction over X and believes Q believes X, then P believes X.
5. Belief rule I: $\frac{P \mid\equiv X, P \mid\equiv Y}{P \mid\equiv (X, Y)}$. It means that if P believes X and P believes Y then P believes (X, Y).
6. Belief rule II: $\frac{P \mid\equiv Q \mid\equiv (X, Y)}{P \mid\equiv Q \mid\equiv X}$. It means that if P believes Q believes (X, Y) then P believes Q believes X.

5.3. **Goals.** We want to show that our improvement should achieve the following goals:
$G_1 : A \mid\equiv (A \overset{SK}{\longleftrightarrow} B)$.
$G_2 : B \mid\equiv (A \overset{SK}{\longleftrightarrow} B)$.
$G_3 : S \mid\equiv (A \overset{SK}{\longleftrightarrow} B)$.
$G_4 : A \mid\equiv B \mid\equiv (A \overset{SK}{\longleftrightarrow} B)$.
$G_5 : B \mid\equiv A \mid\equiv (A \overset{SK}{\longleftrightarrow} B)$.
$G_6 : S \mid\equiv A \mid\equiv (A \overset{SK}{\longleftrightarrow} B)$.
$G_7 : S \mid\equiv B \mid\equiv (A \overset{SK}{\longleftrightarrow} B)$.

5.4. **Idealize the communication messages.** Here, we idealize the communication messages of our improvement listed as below:
$M_1 : A \rightarrow B : (T_a(x), ID_A)$.
$M_2 : A \rightarrow S : (T_a(x), ID_A, C_1)$.
$M_3 : B \rightarrow S : (T_b(x), ID_B, C_2)$.
$M_4 : S \rightarrow A : C_3$.
$M_5 : S \rightarrow B : C_4$.
$M_6 : B \rightarrow A : H_{BA}$.
$M_7 : A \rightarrow B : H_{AB}$.
$M_8 : A \rightarrow S : H'_A$.
$M_9 : B \rightarrow S : H'_B$.

5.5. **Initial state assumptions.** We define some initial state assumptions of our improvement as follows:
$A_1 : A \mid\equiv \sharp(a)$.
$A_2 : B \mid\equiv \sharp(b)$.
$A_3 : A \mid\equiv \sharp(T_a(x))$.
$A_4 : B \mid\equiv \sharp(T_b(x))$.
$A_5 : A \mid\equiv A \overset{pw_A}{\rightleftharpoons} S$.
$A_6 : S \mid\equiv A \overset{pw_A}{\rightleftharpoons} S$.
$A_7 : B \mid\equiv B \overset{pw_B}{\rightleftharpoons} S$.
$A_8 : S \mid\equiv B \overset{pw_B}{\rightleftharpoons} S$.

$A_9 : A \models B \mapsto T_b(x)$.

$A_{10} : B \models A \mapsto T_a(x)$.

A_1 and A_2 mean that A and B generate fresh random values a and b, respectively. Hence, we assume that they are freshness. This implies that A_3 and A_4 are reasonable according to A_1 and A_2. A_5 and A_6 are valid because the password pw_A is chosen by server S and shares with the user A. Similarly, A_7 and A_8 are valid. By A_2 and the computation of $T_b(x)$, we have A_9 is valid. By the similar approach, A_{10} is valid.

5.6. **Detailed description.** Based on the rules of the BAN logic, we prove that our improvement can achieve the defined goals using the initial state assumptions.

For the goal 1. By M_6, we have $S_1 : A \models B \models H_{BA}$. Since H_{BA} contains H_{SA} and H_{SA} contains $T_b(x)$, we can obtain $S_2 : A \models B \models T_b(x)$ by the belief rule II. According to A_9 and S_2, we can obtain $A \models T_b(x)$ by the jurisdiction rule. Since $SK = h(K) = h(T_a(T_b(x)))$, it implies $A \models (A \stackrel{SK}{\longleftrightarrow} B)$.

For the goal 2. By M_1, we have $S_3 : B \models A \models T_a(x)$. According to A_{10} and S_3, we can obtain $B \models T_a(x)$ by the jurisdiction rule. Since $SK = h(K) = h(T_b(T_a(x)))$, it implies $B \models (A \stackrel{SK}{\longleftrightarrow} B)$.

For the goal 3. By M_8, we have $S_4 : S \lhd \langle H'_A \rangle_{pw_A}$. According to A_6 and S_4, we can obtain $S_5 : S \models A \mid\sim H'_A$ by the message meaning rule for shared secrets. Since H'_A is fresh, we have $S_6 : S \models \sharp(H'_A)$. According to S_6 and S_5, we can obtain $S_7 : S \models A \models H'_A$ by the nonce verification rule. Since H'_A contains $T_b(x)$, it implies $S_8 : S \models A \models T_b(x)$ by the belief rule II. By the similar approach, we can obtain $S_9 : S \models B \models T_a(x)$. Finally, according to S_8 and S_9 we can obtain $S \models (A \models T_b(x), B \models T_a(x))$ by the belief rule I. Since $SK = h(K) = h(T_a(T_b(x))) = h(T_b(T_a(x)))$, we have $S \models (A \stackrel{SK}{\longleftrightarrow} B)$.

For the goal 4. Since $SK = h(K)$, by the goal 1 we can obtain $S_{10} : A \models (B \stackrel{K}{\longleftrightarrow} A)$. By M_6, we have $S_{11} : A \lhd \{H_{BA}\}_K$. According to S_{10} and S_{11}, we can obtain $S_{12} : A \models B \mid\sim H_{BA}$ by the message meaning rule for shared keys. Since H_{BA} is fresh, we have $S_{13} : A \models \sharp(H_{BA})$. By S_{13} and S_{12}, we can obtain $A \models B \models H_{BA}$ by the nonce verification rule. Because H_{BA} contains H_{SA} and H_{SA} contains $T_a(x)$, it implies $A \models B \models T_a(x)$ by the belief rule II. Since $SK = h(K) = h(T_b(T_a(x)))$, we have $A \models B \models (A \stackrel{SK}{\longleftrightarrow} B)$.

For the goal 5. Since $SK = h(K)$, by the goal 2 we can obtain $S_{14} : B \models (A \stackrel{K}{\longleftrightarrow} B)$. By M_7, we have $S_{15} : B \lhd \{H_{AB}\}_K$. According to S_{14} and S_{15}, we can obtain $S_{16} : B \models A \mid\sim H_{AB}$ by the message meaning rule for shared keys. Since H_{AB} is fresh, we have $S_{17} : B \models \sharp(H_{AB})$. By S_{17} and S_{16}, we can obtain $B \models A \models H_{AB}$ by the nonce verification rule. Because H_{AB} contains K and K contains $T_b(x)$, we can obtain $B \models A \models T_b(x)$, by the belief rule II. Since $SK = h(K) = h(T_a(T_b(x)))$, we have $B \models A \models (A \stackrel{SK}{\longleftrightarrow} B)$.

For the goal 6. By S_8, we have $S \models A \models T_b(x)$. Since $SK = h(K) = h(T_a(T_b(x)))$, it implies $S \models A \models (A \stackrel{SK}{\longleftrightarrow} B)$.

For the goal 7. By S_9, we have $S \models B \models T_a(x)$. Since $SK = h(K) = h(T_b(T_a(x)))$, it implies $S \models B \models (A \stackrel{SK}{\longleftrightarrow} B)$.

6. **Conclusions.** Until now, there are many 3PAKE schemes and protocols have been proposed. However most of them seems correct, they have been shown to be insecure. In this paper, we have identified a replay attack on Xie et al.'s 3PAKE scheme. Meanwhile, we have proposed an improvement based on their scheme. The security analysis of our improvement is proved by the BAN logic. We hope our design can provide a new solution for 3PAKE to resist the kind of attacks in the future.

Acknowledgement. We thank anonymous referees for the comments and suggestions. This work of Tsu-Yang Wu was supported by Natural Scientific Research Innovation Foundation in Harbin Institute of Technology (No. HIT.NSRIF. 2015089) and the work of Chien-Ming Chen was supported in part by the Project NSFC (National Natural Science Foundation of China) under Grant number 61402135 and in part by Shenzhen Technical Project under Grant number JCYJ20150513151706574.

REFERENCES

[1] Y. Niu and X. Wang, An anonymous key agreement protocol based on chaotic maps, *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, no. 4, pp. 1986–1992, 2011.

[2] T.-F. Lee, An efficient chaotic maps-based authentication and key agreement scheme using smart-cards for telecare medicine information systems, *Journal of Medical Systems*, vol. 37, no. 6, 2013.

[3] C. Guo and 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, 2013.

[4] K. Xue and P. Hong, Security improvement on an anonymous key agreement protocol based on chaotic maps, *Communications in Nonlinear Science and Numerical Simulation*, vol. 17, no. 7, pp. 2969–2977, 2012.

[5] M. S. Farash and M. A. Attari, An enhanced authenticated key agreement for session initiation protocol, *Information Technology and Control*, vol. 42, no. 4, pp. 333–342, 2013.

[6] C.-C. Lee and C.-W. Hsu, A secure biometric-based remote user authentication with key agreement scheme using extended chaotic maps, *Nonlinear Dynamics*, vol. 71, no. 1-2, pp. 201–211, 2013.

[7] C.-C. Lee, C.-T. Li, and C.-W. Hsu, A three-party password-based authenticated key exchange protocol with user anonymity using extended chaotic maps, *Nonlinear Dynamics*, vol. 73, no. 1-2, pp. 125–132, 2013.

[8] F. Zhao, P. Gong, S. Li, M. Li, and P. Li, Cryptanalysis and improvement of a three-party key agreement protocol using enhanced chebyshev polynomials, *Nonlinear Dynamics*, vol. 74, no. 1-2, pp. 419–427, 2013.

[9] Q. Xie, J. Zhao, and X. Yu, Chaotic maps-based three-party password-authenticated key agreement scheme, *Nonlinear Dynamics*, vol. 74, no. 4, pp. 1021–1027, 2013.

[10] M. S. Farash and M. A. Attari, An efficient and provably secure three-party password-based authenticated key exchange protocol based on chebyshev chaotic maps, *Nonlinear Dynamics*, vol. 77, no. 1-2, pp. 399–411, 2014.

[11] X. Hu and Z. Zhang, Cryptanalysis and enhancement of a chaotic maps-based three-party password authenticated key exchange protocol, *Nonlinear Dynamics*, vol. 78, no. 2, pp. 1293–1300, 2014.

[12] C.-C. Lee, C.-T. Li, S.-T. Chiu, and Y.-M. Lai, A new three-party-authenticated key agreement scheme based on chaotic maps without password table, *Nonlinear Dynamics*, vol. 79, no. 4, pp. 2485–2495, 2015.

[13] X. Li, J. Niu, S. Kumari, M. K. Khan, J. Liao, and W. Liang, Design and analysis of a chaotic maps-based three-party authenticated key agreement protocol, *Nonlinear Dynamics*, vol. 80, no. 3, pp. 1209–1220, 2015.

[14] M. Burrows, M. Abadi, and R. M. Needham, A logic of authentication, in *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 426, pp. 233–271, The Royal Society, 1989.

[15] L. Kocarev, J. Makraduli, and P. Amato, Public-key encryption based on chebyshev polynomials, *Circuits, Systems and Signal Processing*, vol. 24, no. 5, pp. 497–517, 2005.

[16] X. Liao, F. Chen, and K.-W. Wong, On the security of public-key algorithms based on Chebyshev polynomials over the finite field Z_N, *Computers, IEEE Transactions on*, vol. 59, pp. 1392–1401, Oct 2010.