Security of an Anonymous Key Agreement Protocol Based on Chaotic Maps

Ya-Fen Chang, Yuo-Ju Yen, and Hui-Feng Huang
Department of Computer Science and Information Engineering, National Taichung University of Science and Technology, Taichung, Taiwan
cyf@nutc.edu.tw, arngstar@gmail.com, phoenix@nutc.edu.tw

Abstract

Xue and Hong proposed an anonymous authentication scheme with key agreement based on chaotic maps. After analyzing Xue and Hong’s scheme thoroughly, we find that their scheme cannot ensure user anonymity as claimed. In this paper, we will show this found security flaw which Xue and Hong’s scheme suffers from.

Keywords: Key agreement, chaotic map, anonymity

1. Introduction

Key agreement provides a mechanism to have communication parties negotiate a shared session key for secure communications. The first key agreement protocol is Diffe-Hellman key exchanging protocol [3]. In the 1990s, the chaos theory becomes a popular research topic [2][7], and plenty of chaos-based systems have been proposed
Recently, Xue and Hong [15] analyzed Niu et al.’s scheme [10] and proposed an improvement to provide user anonymity and improve performance bottleneck. They claimed that their anonymous authentication scheme with key agreement based on Chebyshev polynomial chaotic maps possessed three advantages. (1) User anonymity is provided such that no one knows who is connecting with the server. (2) No third party needs to be involved such that the performance is highly improved. (3) If a client is compromised, the system security will not be threatened. However, after analyzing Xue and Hong’s scheme, we find that their scheme cannot ensure user anonymity as claimed.

The rest of this paper is organized as follows. In Section 2, we introduce Chebyshev polynomial chaotic map and its properties. In Section 3, we briefly review Xue and Hong’s anonymous authentication scheme with key agreement based on chaotic maps. In Section 4, the found flaw is explicitly shown. At last, we conclude this paper in Section 5.

2. Definition and Properties of Chebyshev Polynomial Chaotic Map

In this section, we introduce Chebyshev polynomial chaotic map and its two properties [10][12][13].

2.1 Chebyshev Polynomial Chaotic Map

The Chebyshev polynomial of degree $n$ is defined as follows:

$$T_n(x) = \cos(n \cdot \arccos x), \text{ where } -1 \leq x \leq 1$$

For $n \geq 2$, $T_0(x) = 1$, and $T_1(x) = x$, the recurrent formulas are as follows:

$$T_2(x) = 2x^2 - 1$$
$$T_3(x) = 4x^3 - 3x$$
$$T_4(x) = 8x^4 - 8x^2 + 1$$
$$\vdots$$
$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x)$$

2.2 Semi-group Property

Semi-group property is one of the most important properties of Chebyshev polynomial chaotic maps, and this property makes Chebyshev polynomial chaotic maps used in key agreement protocols or public key encryption schemes. The details are as following:
\[ T_r(T_s(x)) = \cos (r \cdot \arccos (\cos (s \cdot \arccos x))) \]
\[ = \cos (rs \cdot \arccos x) \]
\[ = T_s (T_r (x)) \]
\[ = T_{sr} (x) \]

2.3 Chaotic Property

Chaotic property of Chebyshev polynomial chaotic maps is as follows: For \( n > 1 \) and Lyaounov exponent \( \lambda = \ln n > 0 \), the Chebyshev polynomial map \( T_n : [-1, 1] \rightarrow [-1, 1] \) is a chaotic map of invariant density \( f^s(x) = 1 / (\pi \sqrt{1-x^2}) \).

3. Review of Xue and Hong’s Anonymous Authentication Scheme with Key Agreement Based on Chaotic Maps

In this section, Xue and Hong’s anonymous authentication scheme with key agreement based on chaotic maps is reviewed. In their scheme, there exist two participants, a server \( S \) and a client \( C_i \). Xue and Hong’s scheme consists of two phases: registration phase and anonymous authentication and key agreement phase. The notations used in this paper are listed in Table 1. The details are as follows.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>( S )</td>
<td>Server</td>
</tr>
<tr>
<td>( C_i )</td>
<td>Client</td>
</tr>
<tr>
<td>( ID_i )</td>
<td>The identity of ( C_i )</td>
</tr>
<tr>
<td>( S_S )</td>
<td>( S )'s master key</td>
</tr>
<tr>
<td>( x )</td>
<td>The Chebyshev polynomial’s seed</td>
</tr>
<tr>
<td>( r, s )</td>
<td>The Chebyshev polynomial’s degree</td>
</tr>
<tr>
<td>( SK_i )</td>
<td>Session key between ( S ) and ( C_i )</td>
</tr>
<tr>
<td>( R, a, b, c )</td>
<td>Random numbers</td>
</tr>
<tr>
<td>( H() )</td>
<td>One-way hash function</td>
</tr>
<tr>
<td>( \oplus )</td>
<td>Exclusive- or operator</td>
</tr>
<tr>
<td>( E_K() )</td>
<td>Symmetric encryption algorithm, where ( K ) is the secret key</td>
</tr>
</tbody>
</table>

3.1 Registration Phase

When a new user \( C_i \) with identity \( ID_i \) wants to access resources provided by \( S \) or
establish a session key $SK_i$ with $S$, $C_i$ must register at $S$ at first. The details are as follows:

Step 1: $C_i$ randomly chooses his/her password $PW_i$ and sends $\{ID_i, H(PW_i)\}$ to $S$ via a secure channel.

Step 2: After getting $C_i$'s request, $S$ chooses a random number $R_i$ and computes $M_i$ and $reg_i$, where $M_i = H(ID_i, R_i) \mod 2^{\text{length}(ID)}$ and $reg_i = H(ID_i, H(PW_i)) \oplus H(S_S \oplus M_i)$. Next, $S$ sends $R_i$ and $reg_i$ to $C_i$ via a secure channel.

Step 3: After getting $S$'s reply, $C_i$ stores $R_i$ and $reg_i$ and keeps $reg_i$ secretly.

### 3.2 Anonymous Authentication and Key Agreement Phase

When a registered client $C_i$ wants to access resources of $S$, he/she needs to be authenticated by $S$ and a session key $SK_i$ will be established. The details are as follows:

Step 1: $C_i$ selects $a$, $b$, $x$, and $r$ randomly, where $a$ and $b$ are nonce, $x \in (-\infty, +\infty)$ is the Chebyshev polynomial’s seed, and $r$ is the Chebyshev polynomial’s degree. Then, $C_i$ computes $Q_i$ and $W_i$, where $Q_i = reg_i \oplus H(b)$ and $W_i = H(ID_i, H(PW_i)) \oplus H(b)$. Next, $C_i$ computes $Y_i$, $M_i$, and $E_w'(ID_i, a, Y_i)$, where $Y_i = H(T_i(x))$ and $M_i = H(ID_i, R_i) \mod 2^{\text{length}(ID)}$. $C_i$ sends $\{sn, x, R_i, M_i, Q_i, E_w'(ID_i, a, Y_i)\}$ to $S$, where $sn$ is the session number.

Step 2: After getting $\{sn, x, R_i, M_i, Q_i, E_w'(ID_i, a, Y_i)\}$ from $C_i$, $S$ computes $H(S_S \oplus M_i)$ and $W_i$, where $W_i = Q_i \oplus H(S_S \oplus M_i) = reg_i \oplus H(b) \oplus H(S_S \oplus M_i) = H(ID_i, H(PW_i)) \oplus H(S_S \oplus M_i) \oplus H(b) \oplus H(S_S \oplus M_i) = H(ID_i, H(PW_i)) \oplus H(b)$. $S$ uses $W_i$ to decrypt $E_w'(ID_i, a, Y_i)$ to get $(ID_i, a, Y_i)$. Then $S$ computes $M^*_i = H(ID_i, R_i) \mod 2^{\text{length}(ID)}$ and checks whether $M^*_i = M_i$ or not. If it holds, this request is indeed sent from $C_i$; otherwise, $S$ rejects $C_i$’s login request and terminates this phase immediately. $S$ selects two numbers $s$ and $c$ randomly, where $c$ is a nonce and $s$ is the Chebyshev polynomial’s degree.

Next, $S$ computes $E_w(ID_i, c, T_i(x))$ and sends $\{sn, ID_S, E_w(ID_i, c, T_i(x))\}$ to $C_i$.

Step 3: After getting $\{sn, ID_S, E_w(ID_i, c, T_i(x))\}$, $C_i$ uses $W_i$ to decrypt $E_w(ID_i, c, T_i(x))$ to obtain $(ID_S, c, T_i(x))$. Then $C_i$ computes $E_w(ID_i, a, T_i(x)), SK_i$ and $Ver_i$, where $SK_i = T_i(T_i(x)) = T_{sr}(x) = T_{rs}(x)$ and $Ver_i = H(ID_S, a, c, SK_i)$. At last, $C_i$ sends $\{sn, Ver_i, E_w(ID_i, a, T_i(x))\}$ to $S$.

Step 4: After getting $\{sn, Ver_i, E_w(ID_i, a, T_i(x))\}$, $S$ uses $W_i$ to decrypt $E_w(ID_i, a, T_i(x))$ to retrieve $T_i(x)$. $S$ computes $Y^*_i = H(T_i(x))$ and checks whether $Y^*_i = Y_i$ or not. If it holds, $T_i(x)$ is valid; otherwise, $S$ terminates this phase and sends a reject message to $C_i$. $S$ computes $Ver^*_i = H(ID_S, a, c, SK_i)$ and checks whether $Ver^*_i = Ver_i$. If it holds, $C_i$ is authenticated; otherwise, $S$ terminates this phase immediately. $S$ computes $SK_i$ and $Ver_S$, where $SK_i = T_s(T_i(x)) = T_{rs}(x) = T_{sr}(x)$ and $Ver_S = H(ID_i, a, c, SK_i)$.
Finally, the server $S$ sends $\{sn, Veri_S\}$ to $C_i$.

**Step 5:** After getting $\{sn, Veri_S\}$, $C_i$ computes $Veri_S^* = H(ID_i, a, c, SK_i)$ and checks whether $Veri_S^* = Veri_S$. If it holds, $S$ is authenticated.

Finally, $S$ and $C_i$ share a session key $SK_i$ which can be used for secure communication.

### 4. Security Analysis of Xue and Hong’s Scheme

The previous section reviews Xue and Hong’s anonymous authentication scheme with key agreement based on chaotic maps in detail. They claimed that their scheme provided user anonymity. After analyzing their scheme, we find that a client can be easily traced such that user anonymity is not ensured in their scheme. Why user anonymity is not provided in their scheme is shown as follows.

After registration phase, $C_i$ gets the corresponding $R_i$ and $reg_i$ from $S$. Then in Step 1 of anonymous authentication and key agreement phase, $C_i$ computes $Q_i, W_i, Y_i, M_i$ and $E_w(ID_i, a, Y_i)$ and sends $\{x, R_i, M_i, Q_i, E_w(ID_i, a, Y_i)\}$ as the login request to $S$. $R_i$ and $ID_i$ are fixed, and $M_i = H(ID_i, R_i) \mod 2^{\text{length}(ID_i)}$. That is, $M_i$ is fixed as well, and anyone can trace $C_i$ by $M_i$ as $C_i$’s alias.

### 5. Conclusions

Xue and Hong proposed an authentication scheme with key agreement based on chaotic maps in 2012 and claimed that their scheme could provide user anonymity. However, after analyzing Xue and Hong’s scheme thoroughly, we find that their scheme cannot ensure user anonymity. It is because $M_i$ transmitted in Step 1 of anonymous authentication and key agreement phase is fixed and can be regarded as $C_i$’s alias. Actually, to overcome this found security flaw, transmitted parameters in different sessions should differ from each other.

### References


Biography

Ya-Fen Chang is a Professor of Department of Computer Science and Information Engineering at National Taichung University of Science and Technology in Taiwan. She received her BS degree in computer science and information engineering from National Chiao
Tung University and Ph.D. degree in computer science and information engineering from National Chung Cheng University, Taiwan. Her current research interests include electronic commerce, information security, cryptography, mobile communications, image processing, and data hiding.

Yuo-Ju Yen received the BS degree and the MS degree in computer science and information engineering from National Taichung University of Science and Technology, Taichung, Taiwan in 2012 and 2014, respectively. Her current research interests include information security and cryptography.

Hui-Feng Huang received her M. S. and Ph.D. degrees in Mathematics from National Taiwan University and Computer Science and Information Engineering from National Chung Cheng University, respectively. Currently, she is a professor at the Department of Computer Science and Information Engineering in National Taichung University of Science and Technology. Her research interests focus on the areas of cryptography and information security, network security, algorithm, and electronic commerce etc.