Enhanced Secure Authentication Scheme with Anonymity for Roaming in Mobility Networks

Wen-Chung Kuo
National Yunlin University of Science & Technology, Department of Computer Science and Information Engineering
No.123 University Road, Section 3, Douliou, Yunlin 64002, Taiwan, R.O.C
e-mail: simonkuo@yuntech.edu.tw

Hong-Ji Wei
University of Kang Ning, Library and Information Center
No.188, Sec. 3, Anzhong Rd., Annan Dist., Tainan City 709, Taiwan, R.O.C.

Jiin-Chiou Cheng
Southern Taiwan University of Science and Technology, Department of Computer Science and Information Engineering
No.1, Nantai St., Yongkang Dist., Tainan City 710, Taiwan, R.O.C.

crossref http://dx.doi.org/10.5755/j01.itc.43.2.4029

Abstract. In 2012, Kim and Kwak proposed an anonymous authentication scheme for mobility networks which claimed to improve upon the weakness of replay attack and man-in-the-middle attack in Mun et al.'s scheme. However, their proposed scheme is still vulnerable to replay and DoS attacks. A serious problem in their scheme is that FA cannot get the session key $K_{MF}$. In order to improve these shortcomings, we propose an enhanced secure authentication scheme with anonymity for roaming in mobility networks. The security analysis of our scheme demonstrates maintaining all of the security in Mun et al.'s scheme, but also efficiently improves upon the weaknesses in Kim-Kwak scheme.

Keywords: Authentication scheme; Anonymous authentication; Roaming authentication; Mobility networks.

1. Introduction

With popularization of smart phones and diverse applications, mobility networks are increasingly needed for mobile users. Because mobility networks transfer messages using electromagnetic waves, the message is vulnerable to be intercepted and may expose the user to privacy concerns. Many anonymous authentication schemes with roaming have been proposed for mobile networks to protect user's privacy [1, 3, 4, 6-10, 12-17]. In 2004, Zhu and Ma [17] first proposed a roaming authentication scheme with anonymity for wireless networks. However, in 2006, Lee et al. [4] pointed out that Zhu-Ma's scheme failed to anonymize the user and did not provide backward secrecy of the session key. Furthermore, Lee et al. also proposed an authentication scheme (LHL) with anonymity for wireless networks to provide anonymity and backward secrecy. Unfortunately, Wu et al. [10] proved that LHL-scheme still did not efficiently remove the security weaknesses and proposed an improved authentication scheme (WLT) with anonymity. In 2009, Lee et al. [5] and Xu and Feng [11] showed that the WLT-scheme had improved the weakness of backward secrecy, but did not efficiently protect anonymity of users.

Recently, Mun et al. [9] proposed a new framework of anonymous authentication scheme (MH-LYC) to improve the weakness of anonymity in previous schemes. Kim and Kwak [3] found that the MH-LYC scheme is still weak against replay and man-in-the-middle (MITM) attacks while also proposing an improved anonymous authentication scheme (Kim-Kwak). Unfortunately, the Kim-Kwak scheme did not improve resilience against replay and denial-of-
service(DoS) attacks even though FA cannot get the section key $K_{MF}$, i.e., MU cannot roaming in the FA’s service area, according to our security analysis. In order to provide anonymity and protection against various attacks, we propose an enhanced secure authentication scheme for roaming in mobility networks in this paper. According to our analysis, we prove that the proposed scheme not only maintains all of the security in MLYC-scheme but also improves the weakness against replay in the Kim-Kwak scheme.

The rest of paper is organized as follows: In Sections 2 and 3, we will review the Kim-Kwak scheme and prove the weakness in Kim-Kwak scheme. Then, we will propose an enhanced secure anonymous authentication scheme to overcome the weakness in the Kim-Kwak scheme in Section 4 and provide security analysis in Section 5. Concluding remarks are provided in Section 6.

2. Review of Kim and Kwak Scheme

In this section, we will briefly review the anonymous authentication scheme proposed by Kim and Kwak[3]. There are three phases in this scheme: registration, authentication and key establishment and update session key. The notations of the Kim-Kwak scheme are shown in Table 1 and the procedure of the Kim-Kwak scheme follows.

2.1. Registration phase

In this phase, the new MU computes $h(ID_{MU} \parallel p_a)$ and $h(p_a \parallel N)$ and then sends $ID_{MU}$, $h(ID_{MU} \parallel p_a)$ and $h(p_a \parallel N)$ to HA for registration. After registering with HA, MU will get a smart card with $ID_{MU}$, $ID_{HA}$, $B$, $K$, $V$, $y$ and $h(C)$ from HA and then MU stores $N$ into it. Fig. 1 shows the procedure of the registration phase in detail.
Table 1. Notations

<table>
<thead>
<tr>
<th>Items</th>
<th>Explain</th>
</tr>
</thead>
<tbody>
<tr>
<td>𝑁𝑈</td>
<td>Mobile User</td>
</tr>
<tr>
<td>𝐹𝐴</td>
<td>Foreign Agent</td>
</tr>
<tr>
<td>𝐻𝐴</td>
<td>Home Agent</td>
</tr>
<tr>
<td>𝐼𝐷𝑋</td>
<td>Identity of an entity 𝑋</td>
</tr>
<tr>
<td>ℎ(·)</td>
<td>One-way hash function</td>
</tr>
<tr>
<td>𝑁/𝑁’</td>
<td>Random nonce of current session/ Random nonce of next session</td>
</tr>
<tr>
<td>⊕</td>
<td>Exclusive OR operation</td>
</tr>
<tr>
<td>[ ]</td>
<td>Concatenation operation</td>
</tr>
<tr>
<td>𝑓𝐾(·)</td>
<td>MAC generation function by using key 𝐾</td>
</tr>
<tr>
<td>𝐾𝑌</td>
<td>Session key between entity 𝑋 and 𝑌</td>
</tr>
<tr>
<td>PRNG(·)</td>
<td>Pseudo random number generator</td>
</tr>
<tr>
<td>𝐸𝑥/𝐷玥</td>
<td>Symmetric encryption/decryption with key 𝐾</td>
</tr>
<tr>
<td>𝑝𝑎</td>
<td>Password of mobile user</td>
</tr>
<tr>
<td>𝑥</td>
<td>Secret key of home agent</td>
</tr>
<tr>
<td>𝑦</td>
<td>Random nonce generates for each mobile user</td>
</tr>
<tr>
<td>𝑃</td>
<td>A point on the elliptic curve 𝐸𝑝(𝑎, 𝑏)</td>
</tr>
</tbody>
</table>

2.2. Authentication and key establishment phase

MU can be authenticated by HA via FA after registering with HA. In this phase, MU computes 𝑐1 = 𝐾 ⊕ ℎ(𝑝𝑎 𝑝 𝑁) , 𝑐2 = ℎ(𝑥 ⊕ ℎ(𝐼𝐷𝑀𝑈 𝑝 𝑝)) , 𝑐3 = ℎ(𝑥 ⊕ 𝐼𝐷𝑀𝑈) ⊕ 𝑉 , 𝑐4 = 𝐾 ⊕ ℎ(𝑝𝑎 𝑝 𝑁) and 𝑐5 = ℎ(𝑝𝑎 𝑝 𝑁) ⊕ ℎ(𝑝𝑎 𝑝 𝑁) and then sends 𝐼𝐷𝐻𝐴, 𝑐2, 𝑐3, 𝑐4 and 𝑐5 to FA. Next, FA transfers 𝐼𝐷𝐻𝐴, 𝑐2, 𝑐3, 𝑐4 and 𝑐5 to HA for authenticating MU. HA will check these messages to authenticate MU after receiving them from FA. After HA authenticates MU, he will send 𝐼𝐷𝐻𝐴, 𝐼𝐷𝐹𝐴, 𝑐6; 𝑐6 and 𝑎𝑝 to FA. Finally, FA can establish the session key between itself and MU by 𝑎𝑝 when receiving the above messages from HA. Fig. 2 shows the procedure of the authentication and key establishment phase in detail.

2.3. Update session key phase

If MU continually stays at the same FA, it can update the session key with FA. The update session key phase in Kim-Kwak scheme [3] is the same with MHLYC-scheme [9].

3. Weakness of Kim-Kwak Scheme

In 2012, Kim and Kwak [3] proposed an improved anonymous authentication scheme to overcome that MHLYC was susceptible to the replay attack and MITM attacks. According to our security analysis, however, the Kim-Kwak scheme did not improve resilience against replay or DoS attacks. Another issue of this scheme is that FA cannot get the session key 𝐾 𝑀. Following is a brief analysis of the security of the Kim-Kwak scheme.

3.1. Replay attack: in the authentication and key establishment phase

In this phase, the attacker is able to intercept messages 𝐼𝐷𝐻𝐴, 𝑐2, 𝑐3, 𝑐4 and 𝑐5 between MU and FA and tries to replay it to HA to impersonate MU. Because HA does not store ℎ(𝑝𝑎 𝑝 𝑁) into its database after authenticating MU, the attacker still can authenticate with HA by using intercepted messages and impersonate MU to communicate with others successfully.

3.2. DoS attack: in update session key phase

In this phase, the attacker can calculate 𝑏′𝑝, for 𝑖 = 1, ..., 𝑛, and constantly send it to FA for update session key. Because FA does not check the validity of 𝑏′𝑝, it will make a response with 𝑎𝑝 and 𝑆′𝑀𝐹𝑖 for each request. Therefore, the attacker can mount DoS attack with a flood of packets for request to block services of FA.

3.3. FA cannot get the session key 𝐾𝑀

FA obtains 𝑎𝑝 and 𝑏′𝑝 from HA and MU, respectively. FA still can not compute 𝐾𝑀 = ℎ(𝑎𝑏𝑝) because 𝑎𝑝 is calculated by HA and it is difficult to derive a from 𝑎𝑝. Therefore, FA cannot calculate 𝐾𝑀 and establish the session key with MU in the authentication and key establishment phase.

4. Proposed Enhanced Secure Anonymous Authentication Scheme

MU wants to register with HA before using FA’s roaming service. The registration phase procedure is as follows:

(R.1) 𝑀𝑈 → 𝐻𝐴: 𝐼𝐷𝑀𝑈, ℎ(𝐼𝐷𝑀𝑈 𝑝 𝑝), ℎ(𝑝𝑎 𝑝 𝑁) MU generates a random number 𝑁 by PRNG(·) and computes ℎ(𝐼𝐷𝑀𝑈 𝑝 𝑝) and ℎ(𝑝𝑎 𝑝 𝑁) with his own password 𝑝𝑎.

(R.2) 𝐻𝐴 → 𝑀𝑈: 𝐼𝐷𝑀𝑈, 𝐼𝐷𝐻𝐴, 𝐵, 𝐾, 𝑉, ℎ(·) HA stores ℎ(𝐼𝐷𝑀𝑈 𝑝 𝑝) and ℎ(𝑝𝑎 𝑝 𝑁). Then, HA stores 𝐼𝐷𝑀𝑈, 𝐼𝐷𝐻𝐴, 𝐵, 𝐾, 𝑉, ℎ(·) in the smart card and delivers it to MU through a secure channel.

4.2. Authentication and establishment of session key

MU can conduct anonymous authentication while roaming via FA after registering with HA. The procedure of authentication and establishment of session key phase is shown as follows:
Step 6. Compute $c_5' = h(p_a \parallel N_{i+1}) \parallel h(p_a \parallel N_i)$.

Step 7. Check whether $c_5'$ equals to $c_5$. If it exists HA, stores $p_a \parallel N_{i+1}$ in its database for next session.

Then, HA computes $c_6 = h(K \parallel h(p_a \parallel N_{i+1}) \parallel h(p_a \parallel N_i))$, $c_7 = h(ID_{FA} \parallel h(p_a \parallel N_{i+1}) \parallel h(p_a \parallel N_i))$, and $c_8 = E_V(aP \parallel c_7)$. Thus, HA sends $ID_{FA}, ID_{HA}, c_6, c_7, c_8$ to FA. Otherwise, if $c_5'$ does not equal $c_5$, HA rejects this communication request between $MU$ and $HA$.

(A.4) $FA \rightarrow MU: ID_{HA}, ID_{FA}, c_6, c_7, c_8, aP$

FA checks $ID_{HA}$ and $aP$. If they exist in the database, FA authenticates $HA$ and sends $ID_{HA}, ID_{FA}, c_6$ and $c_7$ to $MU$.

(A.5) $MU \rightarrow FA: bP, S_{MF}, UID$

$MU$ checks that the information $ID_{HA}$ from $FA$ is equal to the original $ID_{HA}$ which has been sent to $FA$ previously. If it exists, then $MU$ computes $c_6' = h(K \parallel h(p_a \parallel N_{i+1}) \parallel h(p_a \parallel N_i))$, $c_7' = h(ID_{FA} \parallel h(p_a \parallel N_{i+1}) \parallel h(p_a \parallel N_i))$, $D_V(c_8)$ and compares $c_6'$ and $c_7'$ with received $c_6$ and $c_7$ for authenticating $HA$ and $FA$, respectively. If they are equal, $MU$ selects a random number $b$ and computes $bP$, $UID = h(ID_{MU} \parallel h(y))$, $K_{MF} = h(abP)$ and $S_{MF} = f_{K_{MF}}(ID_{FA} \parallel bP \parallel UID)$. Then $MU$ sends $bP, S_{MF}$ and $UID$ to $FA$.

(A.6) After receiving the message from $MU$, $FA$ computes $KMF = h(abP)$ and $S_{MF} = f_{K_{MF}}(ID_{FA} \parallel bP \parallel UID)$ and compares $S_{MF}$ with received SMF for authenticating $MU$. If they are equal, $FA$ authenticates $MU$ and stores $bP, UID$ and $KMF$ into its database.

4.3. Update session key phase

If $MU$ stays in $FA$’s region for some time, $MU$ must update the session key with $FA$. The procedure to update the session key is shown as follows:

(U.1) $MU \rightarrow FA: E_{K_{MF_{i-1}}}(b_iP), UID$

$MU$ selects a new random number $b_i$ and computes $b_iP$. Then, $MU$ encrypts $b_iP$ with $K_{MF_{i-1}} = h(a_{i-1}b_{i-1}P)$ and sends $E_{K_{MF_{i-1}}}(b_iP)$ and $UID$ to $FA$.

(U.2) $FA \rightarrow MU: E_{K_{MF_{i-1}}}(a_iP), S_{MF_i}$

FA extracts $K_{MF_{i-1}}$ from the database by received $UID$ and computes $D_{K_{MF_{i-1}}}(E_{K_{MF_{i-1}}}(b_iP))$ to obtain $b_iP$. Then, $FA$ selects a new random number $a_i$ and computes $a_iP$ and $S_{MF_i} = f_{K_{MF_i}}(a_iP)$. Then, $FA$ stores $K_{MF_i}$ into its database.

(U.3) After receiving $E_{K_{MF_{i-1}}}(a_iP), S_{MF_i}$ from $FA$, $MU$ computes $K_{MF_{i-1}} = h(a_{i-1}b_{i-1}P)$ and $D_{K_{MF_{i-1}}}(E_{K_{MF_{i-1}}}(a_iP))$ with $K_{MF_{i-1}}$ to obtain $a_iP$. Then, $MU$ computes $b_iP$, $K_{MF_i} = h(a_iP)$, $S_{MF_i} = f_{K_{MF_i}}(a_iP)$ and checks whether $S_{MF_i}$ equals received $S_{MF_i}$.

If it exists, $MU$ not only authenticates $FA$ but also uses the new session key $K_{MF_i}$ to communicate.

5. Security and Performance Analysis

In this section, we analyze our proposed scheme in terms of security and performance and demonstrate the comparisons of security and performance with previous proposed schemes [3, 4, 9, 10, 17] in Table 2 and Table 3, respectively.

5.1. Anonymity

Assume the attacker intercepts messages $ID_{HA}, c_2, c_3, c_4$ and $c_7$ from $MU$ to $FA$. The attacker cannot ascertain the real identity of the mobile user because the attacker does not know $N_i, N_{i+1}, x$ and $p_a$.

5.2. Secrecy of session key

In the authentication and key establishment and update session key phases, $MU$ and $FA$ utilize
different \( a_iP \) and \( b_iP \) to establish the session key \( K_{MF_i} = h(\text{a}(ib_iP)) \). Since \( a_i \) and \( b_i \) are different for each session and they are not determined by context, the attacker cannot calculate \( K_{MF_i} \) by \( K_{MF_{i-1}} \) or \( K_{MF_{i+1}} \).

### 5.3. Man-in-the-middle attack

In the authentication and establishment of session key phase, the attacker cannot establish a fake Man-in-the-middle session key between \( MU \) and \( FA \) because of mutual authentication between \( MU \) and \( FA \) by \( c_B \) and \( S_{MF} \), respectively.

### 5.4. Replay attack for authentication and establishment of session key phase

In the authentication and establishment of session key phase, \( MU \) selects a new random number \( N_{i+1} \) and computes \( c_4 = K \oplus h(p_a \parallel N_{i+1}) \) and \( c_5 = h(h(p_a \parallel N_i) \parallel h(p_a \parallel N_i)) \). When receiving messages from \( MU \), \( HA \) computes \( c_4' \) and compares it with received \( c_4 \) for authenticating \( MU \). If it exists, \( HA \) stores \( h(p_a \parallel N_{i+1}) \) into its database for the next authentication phase. Otherwise, \( HA \) denies this connection.

### Table 2. Comparisons of security functionality

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Anonymity</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Secrecy of session key</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Prevent impersonation attack</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Prevent replay attack</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Prevent MITM attack</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Prevent DoS attack</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Mutual authentication (MU-FA)</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Mutual authentication (MU-HA)</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>( FA ) can establish key</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Table 3. Comparisons of computational overhead in authentication phase

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( MU )</td>
<td>8H+5XOR</td>
<td>3H+3XOR</td>
<td>5H+2XOR</td>
<td>3H+2XOR</td>
<td>2H+3XOR</td>
<td>10H + 5XOR</td>
</tr>
<tr>
<td></td>
<td>+2S+2P</td>
<td>+ 2S</td>
<td>+ 1S + 2P</td>
<td>+ 2S</td>
<td>+ 2S</td>
<td>2S + 2P</td>
</tr>
<tr>
<td>( FA )</td>
<td>2H+ 1S</td>
<td>4H+1XOR</td>
<td>4H+2XOR</td>
<td>2H+ 1S</td>
<td>2H +1XOR</td>
<td>1H+ 1S</td>
</tr>
<tr>
<td></td>
<td>+ 1P</td>
<td>+1S + 2A</td>
<td>+ 1S + 2P</td>
<td>+ 2A</td>
<td>+ 1S + 2A</td>
<td>2P</td>
</tr>
<tr>
<td>( HA )</td>
<td>4H + 4XOR</td>
<td>5H + 3XOR</td>
<td>3H + 3XOR</td>
<td>5H + 3XOR</td>
<td>5H + 3XOR</td>
<td>3H + 4XOR</td>
</tr>
<tr>
<td></td>
<td>+1S + 1P</td>
<td>+1S + 3A</td>
<td>+ 1S + 3A</td>
<td>+ 1S + 3A</td>
<td>+ 1S + 3A</td>
<td>1S</td>
</tr>
<tr>
<td>Total</td>
<td>14H + 9XOR</td>
<td>12H + 7XOR</td>
<td>12H + 7XOR</td>
<td>10H + 5XOR</td>
<td>9H + 7XOR</td>
<td>14H + 9XOR</td>
</tr>
<tr>
<td></td>
<td>+4S + 4P</td>
<td>+4S + 5A</td>
<td>+2S + 4P</td>
<td>+4S + 5A</td>
<td>+4S + 5A</td>
<td>4S + 4P</td>
</tr>
<tr>
<td>Computation time(sec)</td>
<td>2.0027</td>
<td>2.5026</td>
<td>2.0016</td>
<td>2.5025</td>
<td>2.50245</td>
<td>2.0027</td>
</tr>
</tbody>
</table>

Attackers can attempt to perform replay attacks by the following two steps:

**Step 1.** Attackers intercept \( I_{HA}, c_2, c_3, c_4 \) and \( c_5 = h(h(p_a \parallel N_{i+1}) \parallel h(p_a \parallel N_i)) \) from \( MU \) to \( FA \).

**Step 2.** Attackers replay the intercepted message to \( FA \).

However, the attacker still can not authenticate with \( HA \) by replaying the previous \( I_{HA}, c_2, c_3, c_4 \) and \( c_5 \) from \( MU \) to \( FA \) because \( c'_5 = h(h(p_a \parallel N_{i+2}) \parallel h(p_a \parallel N_{i+1})) \) is not equal to \( c_5 \).

### 5.5. Replay attack for update session key phase

In the update session key phase, \( MU \) encrypts messages with the last session key \( K_{MF_{i-1}} \) while updating the session key with \( FA \). Because the session keys between the present phase and the last phase are different and have no correlation, the attacker can not update the session key by replaying messages transmitted from \( MU \) to \( FA \) in the update session key phase.
5.6. Denial of service attack

In the update session key phase, MU encrypts bP with $K_{MF_{t-1}}$ and sends it with UID to FA. Because FA will extract a and $K_{MF_{t-1}}$ from its database by UID for verifying $bP$ and $K_{MF_{t-1}}$ is different for each session, the attacker cannot mount DoS attack to block services of FA.

5.7. Performance comparison versus other schemes

In order to compare performance between schemes, we calculate the total operations in authentication phase for each scheme. From [2], authors provide an equivalence rate for comparison where RSA = 2, DES = 2,000, and SHA1 = 20,000 operations per second. So for computing execution time, we equate Asymmetric, Symmetric and Hash operations with 0.5, 0.0005, and 0.000005 seconds, respectively. Note that XOR operations are discounted and considered free in these comparisons.

Table 3 shows that the computation of our proposed scheme with previous schemes [3, 4, 9, 10, 17] and proves that the computational cost of our proposed scheme is similar to Kim-Kwak scheme. Our security and performance analysis demonstrates that our proposed scheme not only retains the same computational overhead as Kim-Kwak scheme but also overcomes all weaknesses mentioned in Section 3. Furthermore, Our proposed scheme also provides several security functionalities such as anonymity, secrecy of session key and mutual authentication, and preventing impersonation, replay, MITM and DoS attacks.

6. Conclusions

In this paper, security analysis of the Kim-Kwak scheme determined susceptibility to replay and DoS attacks. In addition, MU cannot roam in the FA’s service area because FA cannot create a session key $K_{MF}$ with MU. In response, we propose an enhanced anonymous authentication scheme with roaming for mobility networks to overcome these weaknesses. We prove that our proposed scheme not only prevents replay and DoS attacks but also allows FA to establish a session key $K_{MF}$ with MU.

References


Received April 2013.