Improvement of a Mutual Authentication Protocol with Anonymity for Roaming Service in Wireless Communication

Chien-Ming Chen, Yenyu Huang, Eric Ke Wang
School of Computer Science and Technology
Harbin Institute of Technology (Shenzhen)
HIT Campus Shenzhen University Town Xili, Shenzhen, China
chienming.taiwan@gmail.com

Tsu-Yang Wu*
Fujian Provincial Key Lab of Big Data Mining and Applications
Fujian University of Technology
Fuzhou, China
wutsuyang@gmail.com

Abstract. Recently, Liu et al. proposed a mutual authentication protocol with user anonymity for wireless communication. In their paper, the authors claimed that the protocol can resist several kinds of attacks even the secret information stored in the smart card is disclosed. However, we still find two vulnerabilities in this paper. First, their protocol still fails to protect user anonymity. Second, their protocol is vulnerable to an off-line password guessing attack if an adversary can derive the secret information stored in a smart card. To solve these problems, we propose a simple but effective patch to their protocol. Furthermore, we use the formal automated application ProVerif to verify the security of our patched protocol.

Keywords: Anonymity, Roaming service, Wireless communication, Cryptanalysis

1. Introduction. Due to the rapid development of wireless networks, information exchanging between mobile devices (e.g., PDA, smart phone and laptop) have been enormously increased. Through the global roaming technology, if a mobile user roams into a foreign network, he needs to perform mutual authentication and establish a session with a foreign agent before transmitting secret information.

Recently, many authentication protocols [1, 2, 3, 4, 5, 6] for the above environment have been proposed. In 2013, Guo et al. proposed a mutual authentication key agreement protocol [4] using a smart card for wireless communications. Their protocol is based on Chebyshev chaotic maps; thus, it is more efficient than previous works. However, in 2016, Liu et al. pointed out that Guo et al.’s protocol cannot resist an impersonation attack if an adversary derives the information stored in the smart card. In order to defend such attacks, Liu et al. proposed a new protocol [5] based on quadratic residue. Liu et al. claimed that their protocol not only provide user anonymity but also resist several kinds

*Corresponding author
of attacks even if the smart card is disclosed. Unfortunately, in this paper we find that Liu et al.’s protocol still fails to protect user anonymity. Besides, if the adversary can extract the secret information stored in a victim’s smart card, he can carry out an off-line password guessing attack. In order to solve the drawbacks we found, we provide a simple but effective patch. We also use the formal automated application ProVerif to verify the security of our patched protocol.

The rest of this paper is organized as follows. Section 2 reviews Liu et al.’s protocol. In Section 3, we demonstrate that Liu et al.’s protocol is vulnerable to various attacks. In Section 4, we propose an improvement of Liu et al.’s protocol. A security analysis is provided in Section 5.

2. Review of Liu et al.’s protocol. In this section, we review Lie et al.’s protocol [5] which contains three phases, the registration phase, the mutual authentication phase and the updated password phase. Notations used in this paper are listed in Table 1.

<table>
<thead>
<tr>
<th>Notations</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HA</td>
<td>Home Agent of a mobile user</td>
</tr>
<tr>
<td>FA</td>
<td>Foreign Agent of the network</td>
</tr>
<tr>
<td>MU</td>
<td>Mobile User</td>
</tr>
<tr>
<td>PW_M</td>
<td>A password of MU</td>
</tr>
<tr>
<td>ID_X</td>
<td>Identity of an entity X</td>
</tr>
<tr>
<td>T_X</td>
<td>Time stamp by an entity X</td>
</tr>
<tr>
<td>K_FH</td>
<td>The pre-shared key between HA and FA</td>
</tr>
<tr>
<td>p_x, q_x</td>
<td>Large primes numbers</td>
</tr>
<tr>
<td>d</td>
<td>The secret key of HA</td>
</tr>
<tr>
<td>h(·)</td>
<td>A one-way hash function</td>
</tr>
<tr>
<td>·</td>
<td>String concatenation operation</td>
</tr>
<tr>
<td>⊕</td>
<td>XOR operation</td>
</tr>
</tbody>
</table>

In the Liu et al.’s protocol, modular square root (MSR) [7, 8] is utilized to ensure the security. More specifically, both Home Agent (HA) and Foreign Agent (FA) require to initialize some parameters before the authentication phase. HA selects two different large primes p_1, q_1 that satisfy p_1 = q_1 = 3 (mod 4) and computes n_1 = p_1 * q_1, it then chooses its secret key d. After that, it publishes n_1 and the one-way hash function h(·). FA also selects two distinct large primes p_2, q_2, and computes n_2 = p_2 * q_2 and then publishes n_2. Besides, FA and HA share a secret key K_FH.

2.1. Registration phase. This phase is involved if a mobile user MU desires to access the system. MU selects a password PW_M and a random number b_M and computes h(PW_M||b_M). He then sends his identity ID_M and h(PW_M||b_M) to HA through a secure channel. Once HA receives the message from MU, he computes

\[ u = (h(ID_M||d))^2 \mod n_1, \]
\[ C = h(u||ID_M), \]
\[ v = u \oplus (PW_M||b_M). \]

Then, HA issues a smart card stores n_1, h(·), v, C and sends this smart card to MU through a secure channel. Finally, MU stores the random number b_M into the smart card (Fig. 1).
2.2. Mutual authentication phase. In this phase, as shown in Fig. 2, MU and FA perform the mutual authentication and establish a session key under the assistance of HA.

**Step 1.** MU inserts his smart card into the smart card reader and inputs his ID\(_M\) and PW\(_M\). The device then calculates \(u^* = v \oplus h(PW_M || b_M)\), \(C^* = h(u^* || ID_M)\). Next, the device will check \(C^* = C\). If the value of \(C^*\) is not equal to that of \(C\), the device terminates this login request for a period of time. Otherwise, the device randomly selects \(n_a, n_b\) and computes CID = ID\(_M\) \(\oplus\) n\(_a\), \(A = (ID_M || n_a)^2 mod\ n_1\), \(B = (ID_F || n_b)^2 mod\ n_2\) and \(R_1 = h(u^* || n_a || B || T_M)\). Then, it sends the message \(m_1 = \{CID, A, B, R_1, ID_H, T_M\}\) to FA.

**Step 2.** Once FA receives the message \(m_1\), FA checks whether the timestamp \(T_M\) is valid. If it is valid, FA selects a random number \(n_F\) and continues to compute \(S1 = h(K_{FH} || A || CID || B || R_1 || n_F || ID_F || T_M || T_F)\). Then FA sends the following messages \(m_2 = \{A, CID, B, R_1, n_F, ID_F, S1, T_M, T_F\}\) to HA.

**Step 3.** Upon receiving the message \(m_2\) from FA, HA first checks whether the timestamp \(T_F\) equals the current time or not. If so, HA then computes \(S1^* = h(K_{FH} || A || CID || B || R_1 || n_F || ID_F || T_M || T_F)\) and checks \(S1^* = S1\). If the two values are equal, HA obtains \(n_a^*\) from MSR of A with the knowledge of ID\(_H\) and computes ID\(_M^*\) = CID \(\oplus\) n\(_a^*\), \(u^* = (h(ID_M || d))^2 mod\ n_1\), \(R_1^* = h(u^* || n_a^* || B || T_M)\). Next, HA checks \(R_1^* = R_1\). If the equation holds, HA computes \(R_2 = h(u^* || n_a^* || ID_F || n_F || T_M)\) and \(S_2 = h(K_{FH} || n_F || B || R_2 || T_M || T_F)\), then sends the message \(m_3 = \{R_2, S_2\}\) to FA.

2.3. Updated password phase. When MU desires to change his password, he inserts the smart card into the device and input his ID\(_M\) and PW\(_M\). The device computes

\[
\begin{align*}
u &= v \oplus h(PW_M || b_M), \\
C^* &= h(u || ID_M).
\end{align*}
\]

Then, the device checks \(C^* = C\). If the value of \(C^*\) is not equal to that of \(C\), the device terminates the password change phase for a period of time. Otherwise, MU inputs a new password PW\(_M^*\) and the device computes \(v^'\) to replace \(v\) on the memory of the smart card.

\[
v^' = v \oplus h(PW_M || b_M) \oplus h(PW_M^* || b_M)
\]

**Figure 1.** Registration phase of Liu etal.'s protocol.
3. Cryptanalysis of Liu et al.’s protocol. For the past decade, various authenticated key exchange protocols have been proposed but many of them have been proven insecure [9, 10, 11, 12, 13, 14]. In Liu et al.’s protocol, they claimed that their protocol can provide user anonymity and resist offline password guessing attack even when the smart card is disclosed. However, we find that this protocol still fails to protect user anonymity. More specifically, with the message eavesdropped from the authentication phase, an adversary can obtain the identity of the MU. Besides, if the secret information in the corresponding card is exposed, the adversary can easily carry out an offline password guessing attack.

3.1. Fails to protect user anonymity. The anonymity protects the privacy of participating entities, which ensures the untraceability of them during communication. The most common way is to conceal user’s real identity. In Liu et al.’s paper, XOR operation and random number are used to protect the user anonymity. Unfortunately, we find that if an adversary eavesdrops the login message $m_1$, he can guess the MU’s identity in polynomial time by the following steps.

**Step 1.** The adversary $E$ eavesdrops the public channel and obtains the message $m_1$, he then extracts $R_1, A, CID$ from $m_1$.

**Step 2.** $E$ guesses the value $ID_M'$ and computes $n'_a = CID \oplus ID_M'$, $A' = (ID_M' || n_a')^2 \mod n_1$. Then it verifies $A' = A$.

**Step 3.** If the verification succeeds, the adversary considers $ID_M'$ as the MU’s identity. Otherwise, he repeats **Step 1**.

Since the identity guessing attack mentioned above does not need to interact with $FA$ and $HA$, it is easy to launch and succeed in polynomial time. Once the adversary guesses the MU’s identity, he can trace the message from it. Thus, Liu et al.’s cannot provide user anonymity.

3.2. Suffer from the offline password guessing attack. From the aforementioned analysis, the adversary can obtain the MU’s identity by performing offline identity guessing attack. Then, we demonstrate Liu et al.’s protocol suffers from password offline guessing attack using the compromised identity $ID_M$ and the information stored in the smart card. The adversary can guess the password as follows:
Step 1. The adversary \( E \) eavesdrops the public channel and obtains the message \( m_1 \), he then extracts \( R_1, CID, T_M \) from \( m_1 \).

Step 2. The adversary guesses the password \( PW'_M \) and computes \( u' = v \oplus h(PW'_M \bmod b_M) \), \( n_a = CID \oplus ID_M, R'_1 = h(u'|n_a||B||T_M) \), where \( b_M \) and \( v \) are stored in the card and \( ID_M \) is gained from offline identity guessing attack. Then verifies \( R'_1 = R_1 \).

Step 3. If the two values equal, the adversary believes \( PW'_M \) is \( MU \)’s password. Otherwise, he repeats Step 1.

Since the \( MU \)’s password can be guessed by the value of \( R_1 \), Liu et al.’s cannot withstand offline password guessing attack.

4. Possible improvement. The reason for such attacks is because the adversary can successfully obtain the identity of the \( MU \). To overcome these weaknesses, a simple but effective solution is to use one-way hash function to protect the random number \( n_a \). In the step 1 of the authentication process, we can modify the calculation of \( CID \), using \( CID = ID_M \oplus h(n_a) \) instead.

Theorem 4.1. Our patched protocol has the property of user anonymity.

Correctness. In Liu et al.’s protocol, the adversary can guess the identity of \( MU \), and then successfully compute the parameter \( A' \) and verify the equality \( A' = A \) holds or not. In the patched protocol, with the message \( m_1 = \{CID, A, B, R_1, ID_H, T_M\} \) acquired in the login phase, it is very hard for an adversary to derive \( MU \)’s identity \( ID_M \) from \( CID \) and \( A \), where \( CID = ID_M \oplus h(n_a) \), and \( A = (ID_H||n_a)^2 \mod n_1 \). To carry out an off-line identity guessing attack, let the adversary \( E \) guesses the \( MU \)’s identity \( ID' \). If the adversary wants to verify the identity, he has to obtain the value of \( n_a \) and \( A' \), where \( n'_a = CID \oplus ID'_M, A' = (ID'_M||n'_a)^2 \mod n_1 \). However, the value he can only get is \( h(n_a) \). As \( h(\cdot) \) is one-way hash function, extract \( n_a \) from \( h(n_a) \) is almost impossible. He cannot gain the value of \( A' \) due to the difficulty of extracting \( MSR \) of a quadratic residue modulo \( n \). Hence, our patched protocol can provide user anonymity.

Theorem 4.2. Our patched protocol can withstand an off-line password guessing attack.

Correctness. In Liu et al.’s protocol, the adversary can obtain the identity of \( MU \) and steal the corresponding smart card. With the data \( \{n_1, h(\cdot), v, C, b_M\} \) from the smart card, the adversary can guess the possible password and compute the parameter \( R'_1 \) and use the equality \( R'_1 = R_1 \) to verify the result. In our patched protocol, let the adversary guess the possible password \( PW'_M \). To verify the value, he needs to compute \( u', n_a \) and \( R'_1 \), where \( u' = v \oplus h(PW'_M \bmod b_M), h(n_a) = CID \oplus ID_M, R'_1 = h(u||n_a \bmod B||T_M) \). It is clear that with the protection of \( h(\cdot) \), \( n_a \) cannot be obtained. Hence, the value of \( R'_1 \) cannot be computed. Thus, the adversary has no ways to judge whether the \( PW'_M \) is correct or not. In a word, our patched protocol can resist an off-line password guessing attack.

5. Security Analysis. In this paper, we use the formal automated application ProVerif to verify the security of our patched protocol. Proverif is an automatic cryptographic protocol tool. As we can obtained the result as under that correctness and key’s secrecy of our patched protocol is evaluated.

The whole simulation process are:

- First, a public channel \( ch \) is defined for the communications, a secure channel \( sch \) is used for registering. \( SKmu \) are the session keys generated by the mobile user and \( SKfa \) are the session keys generated by the foreign agent of the network (Fig. 3). Then comes the functions and rules (Fig. 4).
Second, the purpose of the verification is to verify the following three queries. The former two are about whether an adversary can have the shared session keys or not, and the rest ones are about the correctness of the authentication process. (Figure 5)

- The process of Mobile User. (Figure 6)
- The process of Home Agent of a Mobile User. (Figure 7)
- The process of Foreign Agent of the Network. (Figure 8)
- The process of main execution. (Figure 9)
Improvement of a Mutual Authentication Protocol

**Figure 6. The Process of Mobile User**

```plaintext
(--- MU’s process ---*)
let
ProcessMU(n2:bitstring, IDm:bitstring, IDf:bitstring, IDh:bitstring) =
  new PWm:bitstring; (* the user’s password *)
  new bm:bitstring;
  let HPWm = h(con(PWm, bm)) in
  out(sch, (IDm, HPWm));
(* --- registration:1 ........*)
  in(sch, (x1:bitstring, xu:bitstring, xv:bitstring, xc:bitstring));
(* --- registration:2 ........*)
  ![event UserStarted(IDm);
  let u = xor(xw, HPWm) in
  let C = h(con(u1, IDm)) in
  ![if C = xc then
  new na:bitstring;
  new nb:bitstring;
  new Tmb:bitstring;
  let CID = xor(IDm, h(na)) in
  let A = MSR(IDh, na, x1) in
  let B = MSR(IDf, nb, n2) in
  let R1 = h(con(con(u1, na), B, Tm)) in
  out(ch, (CID, A, B, R1, IDh, Tm)); (* -- authentication:1 ---*)
  in(ch, (x1:bitstring, xx1:bitstring, xx2:bitstring, xxQ:bitstring));
  let R2 = h(con(con(u1, na), xx1:bitstring, xxQ)) in
  ![if R2 = xx2 then
  let SK = h(con(nb, xx1)) in
  out(ch, secn(SKMw, SK));
  ![event UserAuthed(IDf);
  O (* --- authentication:3 ---*)).
```

**Figure 7. The Process of Home Agent of a Mobile User**

```plaintext
(--- HA’s process ---*)
let UserRegl(dh(x1, y1, T1)) =
  new ch:bitstring;
  let u = MSR(dh(x1), y1) in
  let C = h(con(u1, dh(x1))) in
  ![if C = h(con(u1, dh(x1))) then
  ![let SK = h(con(Kf1, n2)) in
  ![let O = h(con(SK, n2)) in
  ![out(ch, (IDh, y2, R2)); (* --- authentication:4 ---*)
  ![O.
```

**Figure 8. The Process of Foreign Agent of the Network**

```plaintext
(--- FA’s process ---*)
let ProcessFA(Kfh:bitstring, IDf:bitstring, n2:bitstring) =
  in(ch, (yID:bitstring, yA:bitstring, yB:bitstring, yR1:bitstring, yIDh:bitstring, yIDf:bitstring, yIDf:bitstring));
  new nf:bitstring;
  new Tf:bitstring;
  ![let S1 = h(con(con(con(con(con(S1, yTM, T1)))); (* --- authentication:2 ---*)
  ![in(ch, (yR2:bitstring, yS2:bitstring));
  let S2 = h(con(con(con(Kfh, n2), yB, yR2, yIDh, yTM, T1));
  ![if S2 = yS2 then
  ![let nb = get(NyB(IDn2)) in
  ![let SK = h(con(nb, nf)) in
  ![out(ch, secn(SKf2, SK));
  ![let O = h(con(SK, nf)) in
  ![out(ch, (IDf, yR2, O)); (* --- authentication:4 ---*)
  ![O.
```
6. Conclusions. In this paper, we analyze the remote user authentication protocol with smart cards for wireless communication proposed by Liu et al. Although their protocol uses MSR to ensure the security, it is still subjected to two security issues, the failure of user anonymity and password protection. We later provide a simple but effective patch to enhance the protocol so that it can resist the two security issues.

Acknowledgement. The work of Chien-Ming Chen was supported in part by Shenzhen Technical Project under Grant number JCYJ20170307151750788 and in part by Shenzhen Technical Project under Grant number KJSCX20170327161755. The work of Eric Ke Wang was supported in part by National Natural Science Foundation of China (No.61572157), grant No.2016A030313660 from Guangdong Province Natural Science Foundation, JCYJ2016 0608161351559 from Shenzhen Municipal Science and Technology Innovation Project. The work of Tsu-Yang Wu was supported in part by Natural Science Foundation of Fujian Province (2018J01636).

REFERENCES


**Chien-Ming Chen** received his PHD from the National Tsing Hua University, Taiwan. He is currently an associate professor Harbin Institute of Technology (Shenzhen), China. He has published over 40 journal papers and 50 conference papers. His current research interests include network security, mobile internet, wireless sensor network and cryptography.

**Yenyu Huang** is currently pursuing the M.S. degree in Harbin Institute of Technology (Shenzhen), China. Her current research interests includes security protocol and network security.

**Eric Ke Wang** is an associate professor of Harbin Institute of Technology (Shenzhen), China. Currently, he work as a senior researcher at Key Laboratory of Shenzhen Internet Information Collaborative Technology and Application of HIT. He received a PhD from department of computer science, the University of Hong Kong in 2009. His main research interests include network security, deep learning. He has obtained two granted projects from National Science Funding (NSFC) of China. Besides, he has developed two software platforms for opportunistic social networks and obtained two authorized related patents.
Tsu-Yang Wu received the PhD degree in Department of Mathematics, National Changhua University of Education, Taiwan in 2010. Currently, he is an associate professor in College of Information Science and Engineering at Fujian University of Technology, China. In the past, he is an assistant professor in Innovative Information Industry Research Center at Shenzhen Graduate School, Harbin Institute of Technology. He is a member of Chinese Cryptology and Information Security Association (CCISA) and China Computer Federation (CCF). He serves as associate editor of two international journals: Data Science and Pattern Recognition, Journal of Network Intelligence. His research interests include cryptography and Network security.