Provably Secure Conference Key Distribution Mechanism Preserving the Forward and Backward Secrecy

Jung-San Lee, Chin-Chen Chang, and Kuo-Jui Wei
(Corresponding author: Jung-San Lee)

Department of Information Engineering and Computer Science, Feng Chia University
No. 100, Wenhwa Rd., Seatwen, Taichung, Taiwan 40724, R.O.C.
(Email: leejs@fcu.edu.tw)
(Received Feb. 20, 2013; revised and accepted May 16, 2013)

Abstract

Due to the explosive development of emerging applications such as, Multicast service, Pay-TV systems, Teleconference, and Collaborate tasks, engineers have proposed many conference key distribution mechanisms. Hwang et al. recently proposed an efficient group key exchange method for providing secure multicast communications, which is a transformation from the two-party key exchange protocol to the group one. In this article, we point out that Hwang et al.'s protocol does not preserve the forward and backward secrecy. We therefore propose an improved version which not only inherits the advantage of previous literature, but also provides the forward and backward secrecy among group members. Besides, we give a formal analysis to the correctness of the new method based on BAN authentication logic.

Keywords: Backward secrecy, forward secrecy, group key exchange

1 Introduction

Diffie and Hellman first proposed the mechanism of two-party key exchange for providing secure communications between two involved participants in 1976 [3, 10]. In order to allow more users to share a secret key for secure communications, Mayer and Yung have proposed a compiler to convert the two-party key exchange protocol to the group one [6, 7, 8, 18]. Hence, users in the same group can quickly obtain a shared secret key to ensure following communications [1, 4, 5, 9, 12, 13, 15, 20]. Nevertheless, the group key exchange architecture proposed by Mayer and Yung is centralized. The centralized approach has two main weaknesses. First, it is un-scalable for large groups. Second, the failure of the centralized controller will lead to the failure of the whole group communication.

Later, Hwang *et al.* proposed a distributed group key exchange mechanism using the compiler suggested by Katz and Yung, which each group member has to take the responsibility for key generation and maintaining the security of the whole group [11, 16, 17, 19]. Unfortunately, we find that Hwang *et al.*'s (HLL) group key exchange protocol has two security weaknesses. First, the forward secrecy is not confirmed while a new member joins into the communication group. That is, the new member can compromise advanced group secret keys to retrieve the previous messages shared between old group members. Second, if a group member is expelled or leaved the group, the backward secrecy is not preserved. In other words, the expellee or leaving member can compromise oncoming group secret keys to learn the shared messages in the future [21, 22]. In this article, we propose an improved secure conference key distribution mechanism (SCKDM) that can get rid of the security weaknesses from which Hwang *et al.*'s group key exchange protocol suffers. In addition, we have given a formal analysis of SCKDM according to BAN authentication logic [2, 14].

The rest of this article is organized as follows. A review of Hwang *et al.*'s group key exchange mechanism is given in Section 2, followed by the security analysis of their mechanism in Section 3. We describe the improved version in Section 4. Next, security analysis of SCKDM is shown in Section 5. We finally make conclusions in Section 6.

2 A Review of HLL Group Key Exchange Mechanism

The main idea of HLL group key exchange mechanism lies in transforming the two-party key exchange protocol to the group one [11]. It is assumed that a secure Diffie-Hellman two-party key exchange protocol is available in HLL mechanism. The HLL structure is depicted in Figure 1. We then define the notations used throughout this article as follows.

- U_i: the identity of the user i, where U_1, U_2, ..., U_n are in a predefined order;
- $K(i, i+1)$: the secret key shared between U_i and U_{i+1}, where $i = 1, 2, \ldots, n-1$;
- $H(\cdot)$: a public one-way hash function;
- sk: the negotiated session key;
- $\oplus$: the exclusive-or operation.

Step 1: Each group user U_i performs the secure Diffie-Hellman two-party key exchange protocol with his/her neighbors U_{i-1} and U_{i+1}, and then negotiates

the secret keys K(i-1, i) and K(i, i+1), respectively. Note that, K(n, 1) is negotiated by U_n and U_1.

Step 2: Ui computes Zi = K(i-1, i) ⊕ K(i, i+1) and then broadcasts the computation result to all group members, where i = 2, 3, …, n-1. Note that, Z1 and Zn are computed as Z1 = K(n, 1) ⊕ K(1, 2) and Zn = K(n-1, n) ⊕ K(n, 1), respectively.

Step 3: While receiving all Zj's, each Ui can obtain other secret keys K(j, j+1)'s by means of the following inference, where j = 1, 2, …, n.

$$K(j, j+1)$$
$$K(j+1, j+2) = Z_{j+1} \oplus K(j, j+1)$$
$$K(j+2, j+3) = Z_{j+2} \oplus K(j+1, j+2)$$
$$\vdots$$
$$K(j-1, j) = Z_{j-1} \oplus K(j-1, j).$$

Step 4: After collecting all secret keys, Ui can compute the group session key as follows,

$$sk = H(K(1, 2), K(2, 3), \ldots, K(n-1, n)).$$

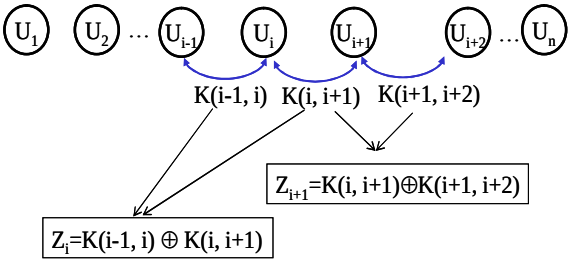

Figure 1: The structure of HLL mechanism

3 Security Analyses of HLL Group Key Exchange Mechanism

We subsequently show that HLL group key exchange mechanism can not achieve the forward secrecy and the backward secrecy with corresponding examples in Subsections 3.1 and 3.2, respectively.

3.1 The Forward Secrecy

Assume that the total number of current group members is n, n > 4. And let U_{n+1} be a new member. After U_{n+1} joins into the group, as shown in Figure 2, the secret keys K(n, n+1) and K(n+1, 1) are generated for U_{n+1}. While receiving all Z_i's, i = 1, 2, …, n+1, U_{n+1} can obtain all other participants' secret keys to compute the current group session key. Furthermore, U_{n+1} can easily recover the past group session keys shared between U_1 to U_j, where j = 4, 5, ..., n, by means of the following computation,

sk' = H(K(1, 2), K(2, 3), …, K(j-1, j)).

Hence, U_{n+1} can learn the previous messages shared between old group members. That is, the forward secrecy is not preserved in Hwang *et al.*'s group key exchange mechanism.

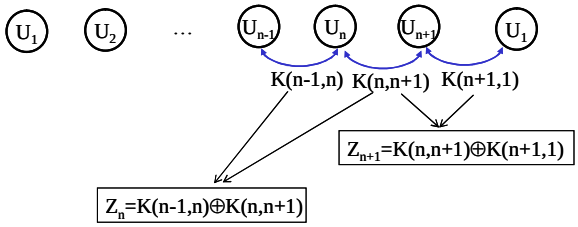

Figure 2: A new member Un+1 joins into the group

Example 1. *Let U_6 be a new member. After U_6 collects Z_1, Z_2, Z_3, Z_4, Z'_5, and Z_6, as shown in Figure 3, U_6 can obtain old session key sk by computing*

$K(5,6) \oplus Z'_5 = K(4,5)$,
$K(4,5) \oplus Z_4 = K(3,4)$,
$K(3,4) \oplus Z_3 = K(2,3)$,
$K(2,3) \oplus Z_2 = K(1,2)$, *and*
$sk = H(K(1,2), K(2,3), K(3,4), K(4,5))$.

That is, U_6 can learn the previous messages shared among U_1, U_2, ..., and U_5.

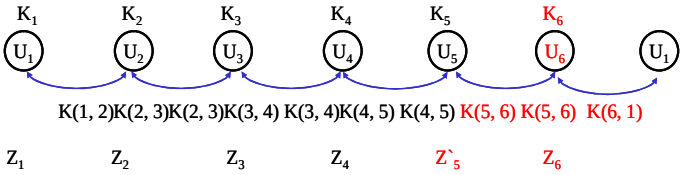

Figure 3: An instance of member join operation

3.2 The Backward Secrecy

Assume that the total number of current group members is n, n > 4. In case that a member U_d wants to leave the communication group, as illustrated in Figure 4, U_{d-1} and U_{d+1} have to negotiate a new secret key K(d-1, d+1). Now, the new group session key is computed as follows.

sk" = H(K(1, 2), K(2, 3), …,K(d-2, d-1), K(d-1, d+1), K(d+1, d+2) …, K(n-1, 1)).

Since U_d keeps other original group members' secret keys, U_d can easily obtain the new generated secret keys K(d-1, d+1) by means of the following inference,

$$K(j, j+1)$$
$$K(j+1, j+2) = Z_{j+1} \oplus K(j, j+1)$$
$$K(j+2, j+3) = Z_{j+2} \oplus K(j+1, j+2)$$
$$\vdots$$
$$K(j-1, j) = Z_{j-1} \oplus K(j-1, j).$$

where j = 1, 2, …, n+1. That is, U_d can successfully compromise the current group session key sk" to learn the current messages shared among all the group members. Consequently, the backward secrecy is not confirmed in Hwang *et al.*'s mechanism.

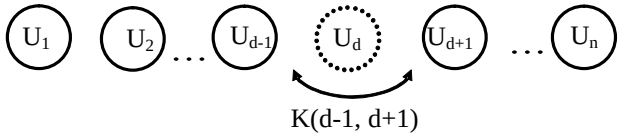

Figure 4: A member Ud leaves the group

Example 2. *Let U_d be an expellee of the group, as illustrated in Figure 5. After U_d leaves the group, U_{d-1} and U_{d+1} have to negotiate a new shared key K(d-1, d+1). Furthermore, the current group key is changed to sk'' = H(K(1,2), K(2,4), K(4,5)). If U_3 ever stored K(1,2) in the database and intercepted the broadcasted messages $Z`_2$ and Z'_4, U_3 then can acquire sk'' by computing*

$K(1,2) \oplus Z'_2 = K(2,4)$,

$sk'' = H(K(1,2), K(2,4), K(4,5))$.

Consequently, U_3 still can learn the messages shared among the group members.

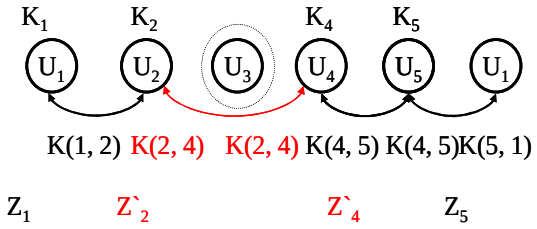

Figure 5: An instance of member leave operation

4 The Improved Secure Conference Key Distribution Mechanism (SCKDM)

To repair Hwang *et al.*'s group key exchange mechanism, we propose an improved version which can preserve the forward secrecy and the backward secrecy. The notations used in our proposed version are the same as those of Hwang *et al.*'s scheme. Besides the group key exchange operation, our scheme has other two main operations: the member join operation and the member leave operation. The details of SCKDM are described as follows.

4.1 The Group Key Exchange Operation

This operation makes all group members be able to negotiate a common session key to ensure the following communications.

Step 1: Each group member U_i performs the secure Diffie-Hellman two-party key exchange protocol with his/her neighbors U_{i-1} and U_{i+1}, and then negotiates the secret keys K(i-1, i) and K(i, i+1), respectively. Note that, K(n, 1) is negotiated by U_n and U_1.

Step 2: Each Ui computes Zi = K(i-1, i) ⊕ K(i, i+1) and then broadcasts the computation result to all group members, where i = 2, 3, …, n-1. Note that, Z1 and Zn are computed as Z1 = K(n, 1) ⊕ K(1, 2) and Zn = K(n-1, n) ⊕ K(n, 1), respectively.

Step 3: After receiving all Zj's, each Ui can get other secret keys K(j, j+1)'s by means of the following inference, where j = 1, 2, …, n.

$K(j, j+1)$

$K(j+1, j+2) = Z_{j+1} \oplus K(j, j+1)$

$K(j+2, j+3) = Z_{j+2} \oplus K(j+1, j+2)$

$\vdots$

$K(j-1, j) = Z_{j-1} \oplus K(j-1, j)$.

Step 4: Upon collecting all secret keys, U_i can compute the group session key as follows,

$sk = H(K(1, 2), K(2, 3), \ldots, K(n-1, n), K(n, 1))$.

4.2 The Member Join Operation

This operation makes the group key exchange mechanism be able to confirm the forward secrecy. While a new user U_{n+1} joins into the communication group, as shown in Figure 2, the secret keys K(n, n+1) and K(n+1, 1) are generated for U_{n+1}. Besides, Z_n is updated as $Z_n = K(n-1, n) \oplus K(n,n+1)$, and Z_{n+1} is constructed as

$$Z_{n+1} = K(n, n+1) \oplus K(n+1,1).$$

Finally, the current group session key is computed as

$$sk = H(K(1, 2), K(2, 3), \ldots, K(n, n+1), K(n+1, 1)).$$

Note that each group member including the new one can obtain other participants' secret keys by performing Step 3 of the group key exchange operation.

4.3 The Member Leave Operation

This operation makes the group key exchange mechanism be able to preserve the backward secrecy. As shown in Figure 4, While a member U_d leaves the communication group, U_{d-1} and U_{d+1} have to negotiated a new secret key $K_{new}(d-1,d+1)$ by performing a secure Diffie-Hellman two-party key exchange protocol defined in Section 2. At the same time, other group members have to perform Step 1 of the group key exchange operation. Then, the new group session key is computed as

$$sk = H(K_{new}(1,2), K_{new}(2,3), \ldots, K_{new}(d-2,d-1), K_{new}(d-1, d+1), K_{new}(d+1,d+2) \ldots, K_{new}(n-1,n), K_{new}(n,1)),$$

where $K_{new}(j,j+1)$'s and $K_{new}(n,1)$are the new generated secret keys for j = 1, 2, …, n-1 and $j \neq d-1, d$.

5 Security Analyses

In the following, we demonstrate SCKDM by BAN logic [2, 14] and show that it can preserve the forward secrecy and the backward secrecy with corresponding instances in Subsections 5.1, 5.2, and 5.3, respectively.

5.1 Authentication Proof by BAN Logic

BAN authentication logic is an important and formal tool for analyzing authentication protocols [2, 14]. Since the construction of the conference key follows a chain rule, we only need to prove that two entities can share their secret keys through a middleman. As illustrated in Figure 1, we have to show that U_{i-1} and U_{i+1} can exchange their secret keys via U_i. Notations used to prove SCKDM follow those of BAN logic [2]. Both of U_{i-1} and U_{i+1} possess two secret keys shared with their neighbors, thus SCKDM must achieve the following goals.

Goal-1

$U_{i-1}| \equiv U_{i-1} \xleftrightarrow{K(i,i+1)} U_{i+1}$

$U_{i+1}| \equiv U_{i-1} \xleftrightarrow{K(i,i+1)} U_{i+1}$

$U_{i-1}| \equiv U_{i+1}| \equiv U_{i-1} \xleftrightarrow{K(i,i+1)} U_{i+1}$

Goal-2

$U_{i-1}| \equiv U_{i-1} \xleftrightarrow{K(i-1,i)} U_{i+1}$

$U_{i+1}| \equiv U_{i-1} \xleftrightarrow{K(i-1,i)} U_{i+1}$

$U_{i+1}| \equiv U_{i-1}| \equiv U_{i-1} \xleftrightarrow{K(i-1,i)} U_{i+1}$

Goal-3

$U_{i-1}| \equiv U_{i-1} \xleftrightarrow{K(i-2,i-1)} U_{i+1}$

$U_{i+1}| \equiv U_{i-1} \xleftrightarrow{K(i-2,i-1)} U_{i+1}$

$U_{i+1}| \equiv U_{i-1}| \equiv U_{i-1} \xleftrightarrow{K(i-2,i-1)} U_{i+1}$

Goal-4

$U_{i-1}| \equiv U_{i-1} \xleftrightarrow{K(i+1,i+2)} U_{i+1}$

$U_{i+1}| \equiv U_{i-1} \xleftrightarrow{K(i+1,i+2)} U_{i+1}$

$U_{i-1}| \equiv U_{i+1}| \equiv U_{i-1} \xleftrightarrow{K(i+1,i+2)} U_{i+1}$

According to BAN logic, the informal form of the communication must be transferred to an idealized form first. The generic type of SCKDM can be illustrated as the following four messages:

M1. $U_i \rightarrow U_{i-1} : Z_i, N_i$

M2. $U_i \rightarrow U_{i+1} : Z_i, N_i$

M3. $U_{i-1} \rightarrow U_{i+1} : Z_{i-1}, N_{i-1}$

M4. $U_{i+1} \rightarrow U_{i-1} : Z_{i+1}, N_{i+1}$.

Note that Nx denotes the statement relevant to participant x, including personal information and the freshness of the message. We further transfer those messages into the idealized form as

I1. $U_i \rightarrow U_{i-1} : \{N_i, U_{i+1} |\equiv U_{i-1} \xleftrightarrow{K(i,i+1)} U_{i+1}\}_{K(i-1,i)}$

I2. $U_i \rightarrow U_{i+1} : \{N_i, U_{i-1} |\equiv U_{i-1} \xleftrightarrow{K(i-1,i)} U_{i+1}\}_{K(i,i+1)}$

I3. $U_{i-1} \rightarrow U_{i+1} : \{N_{i-1}, U_{i-1} \xleftrightarrow{K(i-2,i-1)} U_{i+1}\}_{K(i-1,i)}$

I4. $U_{i+1} \rightarrow U_{i-1} : \{N_{i+1}, U_{i-1} \xleftrightarrow{K(i+1,i+2)} U_{i+1}\}_{K(i,i+1)}$.

To complete the analysis of SCKDM, we give the following basic assumptions:

A1. $U_{i+1} |\equiv U_i \xleftrightarrow{K(i,i+1)} U_{i+1}$

A2. $U_i |\equiv U_i \xleftrightarrow{K(i,i+1)} U_{i+1}$

A3. $U_{i+1} |\equiv U_{i-1} \xleftrightarrow{K(i,i+1)} U_{i+1}$

A4. $U_{i-1} |\equiv U_{i-1} \xleftrightarrow{K(i-1,i)} U_i$

A5. $U_i |\equiv U_{i-1} \xleftrightarrow{K(i-1,i)} U_i$

A6. $U_{i-1} |\equiv U_{i+1} \Rightarrow U_{i-1} \xleftrightarrow{K(i,i+1)} U_{i+1}$

A7. $U_{i-1} |\equiv \#(N_i)$

A8. $U_{i+1} |\equiv \#(N_i)$

A9. $U_{i-1} |\equiv U_{i-1} \xleftrightarrow{K(i-1,i)} U_{i+1}$

A10. $U_{i+1} |\equiv U_{i-1} \Rightarrow U_{i-1} \xleftrightarrow{K(i-1,i)} U_{i+1}$

A11. $U_{i+1} |\equiv \#(N_{i-1})$

A12. $U_{i+1} |\equiv U_{i-1} \Rightarrow U_{i-1} \xleftrightarrow{K(i-2,i-1)} U_{i+1}$

A13. $U_{i-1} |\equiv U_{i-1} \xleftrightarrow{K(i-2,i-1)} U_{i+1}$

A14. $U_{i-1} |\equiv \#(N_{i+1})$

A15. $U_{i-1} |\equiv U_{i+1} \Rightarrow U_{i-1} \xleftrightarrow{K(i+1,i+2)} U_{i+1}$

A16. $U_{i+1} |\equiv U_{i-1} \xleftrightarrow{K(i+1,i+2)} U_{i+1}$

We can now proceed with the analysis of SCKDM by three logical postulates: *message-meaning* rule, *nonce-verification* rule, and *jurisdiction* rule [2].

Proof of Goal-1:

According to A1 and A2, the first message I1 gives,

R1. $U_{i-1} \triangleleft \{N_i, U_{i+1} |\equiv U_{i-1} \xleftrightarrow{K(i,i+1)} U_{i+1}\}_{K(i-1,1)}$.

Thus, by *message-meaning* rule, we have

R2. $U_{i-1}| \sim U_{i+1}| \equiv U_{i-1} \xleftrightarrow{K(i,i+1)} U_{i+1}$.

By A7 and R2, using *nonce-verification* rule, we can derive,

R3. $U_{i-1}| \equiv U_{i+1}| \equiv U_{i-1} \xleftrightarrow{K(i,i+1)} U_{i+1}$.

By A6 and R3, we can further derive,

R4. $U_{i-1}| \equiv U_{i-1} \xleftrightarrow{K(i,i+1)} U_{i+1}$,

by *jurisdiction* rule. According to A3, R3, and R4, we have the beliefs of Goal-1. □

Proof of Goal-2:

The second message I2 is similar to I1, giving:

R5. $U_{i+1} \triangleleft \{N_i, U_{i-1} |\equiv U_{i-1} \xleftrightarrow{K(i-1,i)} U_{i+1}\}_{K(i,i+1)}$.

By *message-meaning* rule, we can further obtain,

R6. $U_{i+1}| \sim U_{i-1}| \equiv U_{i-1} \xleftrightarrow{K(i-1,i)} U_{i+1}$.

Based on *nonce-verification* rule, we can use A8 and R6 to derive,

R7. $U_{i+1}| \equiv U_{i-1}| \equiv U_{i-1} \xleftrightarrow{K(i-1,i)} U_{i+1}$.

By A10 and R7, we can apply *jurisdiction* rule to derive,

R8. $U_{i+1}| \equiv U_{i-1} \xleftrightarrow{K(i-1,i)} U_{i+1}$.

Thus, we have the beliefs of Goal-2, A9, R7, and R8. □

Proof of Goal-3:

In the third message I3, U_{i-1} uses the shared key K(i-1,i) to convince U_{i+1} that the message is really from U_{i-1}, using *message-meaning* rule, giving:,

R9. $U_{i+1} \triangleleft \{N_{i-1}, U_{i-1} \xleftrightarrow{K(i-2,i-1)} U_{i+1}\}_{K(i-1,i)}$,

R10. $U_{i+1} |\equiv U_{i-1}| \sim U_{i-1} \xleftrightarrow{K(i-2,i-1)} U_{i+1}$.

By A11, we have,

R11. $U_{i+1} |\equiv \#(U_{i-1} \xleftrightarrow{K(i-2,i-1)} U_{i+1})$.

We then apply *nonce-verification* rule to derive,

R12. $U_{i+1} |\equiv U_{i-1}| \equiv U_{i-1} \xleftrightarrow{K(i-2,i-1)} U_{i+1}$.

By A12 and R12, adopting *jurisdiction* rule, we can obtain

R13. $U_{i+1} |\equiv U_{i-1} \xleftrightarrow{K(i-2,i-1)} U_{i+1}$.

Consequently, we have the beliefs of Goal-3, A13, R12,

and R13. □

Proof of Goal-4:

With the shared key K(i,i+1), U_{i-1} can obtain the following from message I4,

R14. $U_{i-1} \triangleleft \{N_{i+1}, U_{i-1} \xleftrightarrow{K(i+1,i+2)} U_{i+1}\}_{K(i,i+1)}$.

Using *message-meaning* rule, we can derive,

R15. $U_{i-1} |\equiv U_{i+1}| \sim U_{i-1} \xleftrightarrow{K(i+1,i+2)} U_{i+1}$,

by R14. Besides, according to A14, we have,

R16. $U_{i-1} |\equiv \#(U_{i-1} \xleftrightarrow{K(i+1,i+2)} U_{i+1})$.

R15 and R16 lead to:

R17. $U_{i-1} |\equiv U_{i+1}| \equiv U_{i-1} \xleftrightarrow{K(i+1,i+2)} U_{i+1}$.

By A15 and R17, we then can apply *jurisdiction* rule to derive,

R18. $U_{i-1} \equiv U_{i-1} \xleftrightarrow{K(i+1,i+2)} U_{i+1}$.

Eventually, we have the beliefs of Goal-4, A16, R17, and R18. □

5.2 To Preserve the Forward Secrecy

At the beginning, we assume that the total number of current group members is n, n > 4. As shown in Figure 2, while a new user U_{n+1} joins into the communication group, the secret keys K(n, n+1) and K(n+1, 1) are generated for U_{n+1}. After receiving all Z_i's, i = 1, 2, …, n+1, U_{n+1} can obtain all other participants' secret keys to compute the current group session key. Nevertheless, U_{n+1} can not recover the past group session keys shared among U_1 to U_j, where j = 4, 5, …, n. It is due to that U_{n+1} can not obtain K(j, 1)'s to compute those past session keys by means of the following computation.

sk = H(K(1, 2), K(2, 3), …, K(j-1, j), K(j, 1)).

As a result, the forward secrecy is preserved in our proposed group key exchange mechanism.

Example 3. *Let U_6 be a new member. After U_6 collects Z_1, Z_2, Z_3, Z_4, Z'_5, and Z_6, as shown in Figure 3, U_6 can obtain new session key sk by computing*

$K(5,6) \oplus Z'5 = K(4,5)$,
$K(4,5) \oplus Z4 = K(3,4)$,
$K(3,4) \oplus Z3 = K(2,3)$,
$K(2,3) \oplus Z2 = K(1,2)$, and
$sk' = H(K(1,2), K(2,3), \ldots, K(5,6), K(6,1))$.

Without the knowledge of K(5,1), U_6 is unable to compute the past group key as

$sk = H(K(1,2), K(2,3), \ldots, K(4,5), K(5,1))$. That is, U_6 cannot read the messages shared among U_1, U_2, …, and U_5.

5.3 To Preserve the Backward Secrecy

In this subsection, we show that our scheme can confirm the backward secrecy. At first, we assume that the total number of current group members is n, n > 4. In case that a member U_d leaves the communication group, as illustrated in Figure 4, U_{d-1} and U_{d+1} have to negotiated a new secret key K_{new}(d-1, d+1) by performing a secure Diffie-Hellman two-party key exchange protocol defined in Section 2. Simultaneously, other group members have to perform Step 1 of the group key exchange operation to generate new secret keys K_{new}(j, j+1)'s, where j = 1, 2, …, n and j ≠ d-1, d. Then, the new group session key is computed as

$$sk'' = H(K_{new}(1, 2), K_{new}(2, 3), \ldots, K_{new}(d-2, d-1), K_{new}(d-1, d+1), K_{new}(d+1, d+2) \ldots, K_{new}(n-1, n), K_{new}(n, 1)).$$

Without the knowledge of new generated secret keys, U_d can not compromise the current group session key sk". Consequently, our proposed scheme is capable of preserving the backward secrecy.

Example 4. *Let U_d be an expellee of the group, as illustrated in Figure 5. After U_d withdraws from the group, U_{d-1} and U_{d+1} have to negotiate a new shared key K(d-1, d+1). Furthermore, U_1, U_2, U_4, and U_5 must perform the secure Diffie-Hellman protocol to obtain new secret keys K'(1,2), K'(4,5), and K'(5,1). Hence, the current group key is changed to sk'' = H(K(1,2), K(2,4), K(4,5), K(5,1)). Since U_3 does not share secret keys with other members anymore, U_3 can not compute the current group key sk'' to learn any information shared among remainders of the group.*

6 Conclusions

In this article, we have discussed the forward secrecy and the backward secrecy for Hwang *et al.*'s group key exchange mechanism. As shown in Subsections 3.1 and 3.2, the forward secrecy and the backward secrecy are not preserved in HLL mechanism. Hence, we propose an improved version which can get rid of the security weaknesses from which Hwang *et al.*'s scheme suffered. Furthermore, the correctness of SCKDM is formally analyzed by BAN authentication logic.

References

[1] G. Ateniese, M. Steiner, and G. Tsudik, "New multi-party authentication services and key agreement protocols," *IEEE Journal on Selected Areas in Communications*, vol. 18, no. 4, pp. 628-639, 2000.

[2] M. Burrows, M. Abadi, and R. Needham, "A logic of authentication," *ACM Transactions on Computer Systems*, vol. 8, no. 1, pp. 18-36, 1990.

[3] R. Bird, I. Gopal, A. Herzberg, P. Janson, S. Kutten, R. Molva, and M. Yung, "Systematic design of two-party authentication protocols," *IEEE Journal on Selected Areas in Communications*, vol. 11, no. 5, pp. 679-693, 1993.

[4] M. Bellare and P. Rogaway, "Entity authentication and key distribution," in *Cryptology-Crypto '93*, pp. 232-249, Springer-Verlag, 1994.

[5] C. Boyd and J. M. G. Nieto, "Round-optimal contributory conference key agreement," in *Public-Key Cryptography*, pp. 161-174, Springer-Verlag, 2003.

[6] E. Bresson, O. Chevassut, and D. Pointcheval, "Provably authenticated group diffie-hellman key exchange - The dynamic case," in *Cryptology-Asiacrypt '01*, pp. 290-309, Springer-Verlag, 2001.

[7] E. Bresson, O. Chevassut, and D. Pointcheval, "Dynamic group diffie-hellman key exchange under standard assumptions," in *Cryptology-Eurocrypt '0*2, pp. 321-336, Springer-Verlag, 2002.

[8] E. Bresson, O. Chevassut, D. Pointcheval, and J. J. Quisquater, "Provably authenticated group Diffie-Hellman key exchange," in *Proceedings of the 8th Annu. ACM Conference on Computer and Communications Security*, pp. 255-264, Philadelphia, 2001.

[9] T. Y. Chang and M. S. Hwang, "User-anonymous and short-term conference key distribution system via link-layer routing in mobile communications," *International Journal of Mobile Communications*, vol. 9, no. 2, pp. 144-158, 2011.

[10] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory*, vol. 22, pp. 644-654, 1976.

[11] J. Y. Hwang, S. M. Lee and D. H. Lee, "Scalable key exchange transformation: from two–party to group," *Electronic Letters*, vol. 40, no. 12, pp. 728-729, 2004.

[12] M. S. Hwang, "Dynamic participation in a secure conference scheme for mobile communications", *IEEE Transactions on Vehicular Technology*, vol. 48, no. 5, pp. 1469-1474, Sep. 1999.

[13] M. S. Hwang and W. P. Yang, "Conference key distribution protocols for digital mobile communication systems," *IEEE Journal on Selected Areas in Communications*, vo1. 13, no. 2, pp. 416-420, Feb. 1995.

[14] R. J. Hwang and F. F. Su, "A new efficient authentication protocol for mobile networks," *Computer Standards & Interfaces*, vol. 28, pp. 241-252, 2005.

[15] I. Ingemarsson, D. T. Tang, and C. K. Wong, "A conference key distribution system," *IEEE Transactions on Information Theory*, vol. 28, no. 5, pp. 714-720, 1982.

[16] J. Katz and M. Yung, "Scalable protocols for authenticated group key exchange," in *Cryptology-Crypto '03*, LNCS 2729, pp. 110-125, Springer-Verlag, 2003.

[17] S. M. Lee, H. J. Kim, D. H. Lee, J. I. Lim and C. S. Park, "Scalable group key management with minimally trusted third party," in *Proceedings of the 4th International Workshop on Information Security Applications*, pp. 575-583, Korea, 2003.

[18] A. Mayer and M. Yung, "Secure protocol transformation via expansion from two-party to groups," in *Proceedings of the 6th ACM Conference on Computer and Communications Security*, pp. 83-92, Singapore, 1999.

[19] M. Steiner, G. Tsudik, and M. Waidner, "Key agreement in dynamic peer groups," *IEEE Transactions on Parallel and Distributed Systems*, vol. 11, no. 8, pp. 769-780, 2000.

[20] C. C. Yang, T. Y. Chang, and M. S. Hwang, "A new anonymous conference key distribution system based on the elliptic curve discrete logarithm problem," *Computer Standards and Interfaces*, vol. 25, no. 2, pp. 141-145, May 2003.

[21] W. Yuan, L. Hu, and J. Chu, "Cryptanalysis of Lee et al.'s authenticated group key agreement," *Procedia Engineering*, vol. 15, pp. 1421-1425, 2011.

[22] J. Zhao, D. Gu, and W. Cheng, "On weaknesses of the HDCP authentication and key exchange protocol and its repair," *Mathematical and Computer Modelling*, vol. 55, no. 1-2, pp. 19-25, Jan. 2012

Jung-San Lee received the BS degree in computer science and information engineering in 2002 and his Ph.D in computer science and information engineering in 2008, both from National Chung Cheng University, Chiayi, Taiwan. Since 2012, he has worked as an associate professor in the Department of Information Engineering and Computer Science at Feng Chia University, Taichung, Taiwan. His current research interests include information security, image processing, and watermarking.

Chin-Chen Chang received his BS degree in applied mathematics in 1977 and his MS degree in computer and decision sciences in 1979, both from the National Tsing Hua University, Hsinchu, Taiwan. He received his Ph.D in computer engineering in 1982 from the National Chiao Tung University, Hsinchu, Taiwan. During the academic years of 1980-1983, he was on the faculty of the Department of Computer Engineering at the National Chiao Tung University. From 1983-1989, he was on the faculty of the Institute of Applied Mathematics, National Chung Hsing University, Taichung, Taiwan. From 1989 to 2004, he has worked as a professor in the Institute of Computer Science and Information Engineering at National Chung Cheng University, Chiayi, Taiwan. Since 2005, he has worked as a professor in the Department of Information Engineering and Computer Science at Feng Chia University, Taichung, Taiwan. Dr. Chang is a Fellow of IEEE, a Fellow of IEE and a member of the Chinese Language Computer Society, the Chinese Institute of Engineers of the Republic of China, and the Phi Tau Phi Society of the Republic of China. His research interests include computer cryptography, data engineering, and image compression.

Kuo-Jui Wei received the Bachelor degree in information engineering and computer science in 2009. He is currently pursuing her Ph.D. degree in Information Engineering and Computer Science in Feng Chia University, Taichung, Taiwan. His current research interests include information security and mobile communications.