Cryptanalysis of an Identity-Based Multiple Key Agreement Scheme

Qingfeng Cheng
Luoyang University of Forgein Languages
Luoyang, Henan 471003-P.R.China
[e-mail: qingfengc2008@sina.com]

Abstract

Multiple key agreement (MKA) protocols allow two parties to generate two or more session keys in one session, which will be used for future secure communications in public network. In recent years, many MKA protocols have been proposed. However, most of them do not consider ephemeral key compromise resilience, and some of them still exists security flaws. In this paper, we analyze the scheme proposed by Dehkordi and Alimoradi in 2011, which is announced with stronger security. We will present ephemeral key compromise attack and impersonation attack against Dehkordi and Alimoradi's protocol. For overcoming these security flaws, we also propose an improvement of Dehkordi and Alimoradi's protocol.

Keywords: multiple key agreement, bilinear pairing, mutual authentication, ephemeral key compromise attack, impersonation attack

1. Introduction

Authenticated key agreement (AKA) protocols play an important role in secure communications. An AKA protocol allows two or more parties to generate one or more shared session keys for the futrue communications. Since Diffie and Hellman proposed the most famous Diffie-Hellman key agreement protocol [1] in 1976, many AKA protocols as the extensions of Diffie-Hellman key agreement protocol have been proposed.

As we know, a secure AKA protocol should be able to withstand both passive attacks and active attacks. The following security attributes of AKA protocols had been identified as fundamental requirments. More details can refer to [2, 3].

- **Known-Key Security**. Each run of a key agreement protocol should generate an unique and independent session key. No adversary has non-negligible adavantage to compute future session keys, even though it has learned the past session keys.
- **Perfect Forward Secrecy**. Even if two parties' private key are compromised, no adversary can have non-negligible advantage to recover the past session keys.
- **Key Compromise Impersonation Resilience**. No adversary has non-negligible adavantage to impersonate a party to cheat the other party, even if it has achieved the party's private key.
- **Unknown Key Share Resilience**. No adversary has non-negligible advantage on

coercing others into sharing a session key with other parties when it is actually sharing this session key with a different party.

- **No Key Control**. No adversary can have non-negligible advantage on forcing the session key to be a pre-selected value.

In addition, ephemeral key compromise resilience first proposed by Krawczyk [4] has also been considered in the design of many AKA protocols. Krawczyk pointed out that many applications may boost protocol performance by pre-computing ephemeral pair (x, g^x) for later use in the protocol, and these stored pairs are more vulnerable to leakage than static private key. In 2007, LaMacchia, Lauter and Mityagin [5] added ephemeral key reveal query in the extended Canetti-Krawczyk (eCK) model, which was based on the Canetti-Krawczyk model [6]. Their motivations to include ephemeral key compromise resilience in eCK model comes from active adversarial attacks or compromise of the random number generator. Recently, Moriyama and Okamoto [7] showed that an AKA protocol proven secure in the eCK model could be vulnerable to ephemeral key compromise attack by some realistic side-channel attacks, such as up-to-date power-analysis.

- **Ephemeral Key Compromise Resilience**. The adversary can obtain the ephemeral private keys of parties, which are chosen directly from a group. But the session keys under attack still remains secure. It means that the adversary cannot compute the accepted session keys.

Multiple key agreement (MKA) protocols as a research direction in AKA protocols aims to generate two or more shared session keys in one session. The pioneer work in the field was proposed by Harn and Lin [8] in 1998, called HL protocol. But Yen and Joye [9] pointed out that HL protocol existed security flaws, and proposed a new MKA protocol. In 1999, Wu et al. [10] showed that Yen and Joye's protocol also was insecure. In 2001, Harn and Lin [11] proposed an improvement of HL protocol. Unforunately, the improvement of HL protocol cannot resist unknown key share attack [12] and impersonantion attack [13]. In 2008, Lee et al. [14] presented a novel two-party MKA protocol based on bilinear pairings. Vo et al. [15] in 2010 showed that Lee et al.'s protocol cannot provide mutual authentiaction and resist impersonation attack. Furthermore, Vo et al. proposed a simple modification to Lee et al.'s protocol, called VLYK protocol. However, Cheng and Ma [16] showed that Vo et al.'s MKA protocol cannot resist reflection attack. It means that the VLYK protocol fails to provide mutual authentication. Moreover, there are also others MKA schemes such as [17][18][19][20] [21][22][23].

More recently, Dehkordi and Alimoradi [24] proposed an identity-based MKA protocol based on bilinear pairings, which used the challenge-response method to verify the iedentities of two parties in the protocol. They announced that their protocol satisfied many security properties such as the property of key compromise impersonation security, the property of unknown key security and strong security property. In this paper, we will show that their protocol cannot resist impersonation attack and ephemeral key compromise attack. This means that the adversary with ephemeral keys can compute session keys easily. In addition, the adversary can impersonate a party to cheat the other party. Furthermore, we also propose an improved protocol, which can withstand ephemeral key compromise attack and impersonation attack.

The organizations of the rest paper as follows. In Section 2, we introduce the properties of bilinear pairing and the assumptions. In Section 3, we briefly review the Dehkordi and Alimoradi's MKA protocol. In Section 4, we present impersonation attack and ephemeral key compromise attack on the Dehkordi and Alimoradi's protocol. Section 5 proposes an

improvement of Dehkordi and Alimoradi's protocol, which can resist our attacks. Finally, the paper is concluded in Section 6.

2. Preliminaries

The Dehkordi and Alimoradi's protocol is based on the bilinear pairings over elliptic curves. In this section, we briefly introduce the definition of bilinear pairing and present the hardness assumption on which the security of the Dehkordi and Alimoradi's MKA protocol relies.

2.1 Bilinear Pairing

We describe the basic definition and properties of the pairing. More details can refer to [25] and [26]. Let q be a large prime, G_1 be an additive group of prime order q, and G_2 be a multiplicative group of prime order q. $\hat{e}: G_1 \times G_1 \to G_2$ is a bilinear pairing that has the following properties:

- **Bilinearity**. For any $Q,W \in G_1$ and $a,b \in Z_q^*$, we have $\hat{e}(aQ,bW) = \hat{e}(Q,W)^{ab}$.
- **Non-degeneracy**. There exists $Q,W \in G_1$ such that $\hat{e}(Q,W) \neq 1$.
- **Computability**. For any $Q,W \in G_1$, there exists an efficient algorithm to compute $\hat{e}(Q,W)$.

2.2 Diffie-Hellman Problems

In this subsection, we describe some hard problems. For more details refer to [26].

- **Discrete Logarithm (DL) Problem**. Given two elements $Q_1, Q_2 \in G_1$. Find the integer n whenever such an integer exists, such that $Q_1 = nQ_2$.
- **Computational Diffie-Hellman (CDH) Problem**. Let P be a generator of the group G_1. Given (P, aP, bP) with $a,b \in Z_q^*$, computes $abP \in G_1$.
- **Bilinear Diffie-Hellman (BDH) Problem**. Let P be a generator of the group G_1. Given (P, aP, bP, cP) with $a,b,c \in Z_q^*$, computes $\hat{e}(P,P)^{abc} \in G_2$.

We say that G_1 and G_2 satisfy the DL, CDH and BDH assumptions if no feasible adversary can solve the DL, CDH and BDH problems with non-negligible probability.

3. Review of Dehkordi et al's Protocol

In this section, we briefly review Dehkordi and Alimoradi's protocol, which is composed of the following three stages.

3.1 System Initialization Stage

Let G_1 be an additive group of prime order q and G_2 be a multiplicative group of the same prime order q. P is a generator of group G_1. $\hat{e}: G_1 \times G_1 \to G_2$ described in Section 2 is a bilinear pairing. The key generation center (KGC) chooses a value s in Z_q^* randomly as the master private key and computes $P_{pub} = sP$ as its master public key. $H: \{0,1\}^* \to G_1$ and

$H_1 : \{0,1\}^* \to Z_q^*$ are two hash functions selected by KGC. The system's public parameters are $\{p, q, G_1, G_2, P, P_{pub}, H, H_1, \hat{e}\}$.

3.2 Key Extract Stage

This phase is run by the KGC for all the parties under it. Each party with the identity $ID \in \{0,1\}^*$ is issued a public key $Q_{ID} = H(ID)$ and a private key $S_{ID} = sQ_{ID}$, which is generated and sent via a secure channel by the KGC.

3.3 Multiple Key Agreement Stage

In this subsection, we briefly review the multiple key agreement stage and suppose that *A* and *B* denote two parties involved in the protocol. They can generate four shared keys respectively in one session.

Step1: Party *B* selects a random value r_B in Z_q^*, and then computes and sends $C = r_B Q_{IDB}$ to party A.

Step2: After receiving the message $\{C, B\}$, party A first selects a random value r_A in Z_q^*, and then computes $T = r_A Q_{IDA}$ and $\bar{Y} = (r_A + H_1(C \| IDB \| IDA))S_{IDA}$. Finally A sends $\{T, \bar{Y}, A\}$ to party B.

Step3: After receiving the message $\{T, \bar{Y}, A\}$, party B first computes $f = H_1(T \| IDA \| IDB)$ and $f' = H_1(C \| IDB \| IDA)$, then B computes $Y = (c + f)S_{IDB}$ and checks the verification equation $\hat{e}(P, \bar{Y}) = \hat{e}(P_{pub}, T + f'Q_{IDA})$. If the verification equation is valid , then party *B* sends the message $\{Y, B\}$ to party A and computes the common session keys as follows:

$$
\begin{aligned}
K_{B1} &= \hat{e}(T, S_{IDB})^{r_B} \\
&= \hat{e}(r_A Q_{IDA}, sQ_{IDB})^{r_B} \\
&= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_A r_B}
\end{aligned}
$$

$$
\begin{aligned}
K_{B2} &= \hat{e}(Q_{IDA}, S_{IDB}) K_{B1} \\
&= \hat{e}(Q_{IDA}, sQ_{IDB}) K_{B1} \\
&= \hat{e}(Q_{IDA}, Q_{IDB})^{s} K_{B1}
\end{aligned}
$$

$$
\begin{aligned}
K_{B3} &= \hat{e}(Q_{IDA}, S_{IDB})^{r_B} K_{B1} \\
&= \hat{e}(Q_{IDA}, sQ_{IDB})^{r_B} K_{B1} \\
&= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_B} K_{B1}
\end{aligned}
$$

$$
\begin{aligned}
K_{B4} &= \hat{e}(T, S_{IDB}) K_{B1} \\
&= \hat{e}(r_A Q_{IDA}, sQ_{IDB}) K_{B1} \\
&= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_A} K_{B1}
\end{aligned}
$$

Step4: Party A computes $f = H_1(T \| IDA \| IDB)$ and checks the verification equation

$$\hat{e}(P,Y) = \hat{e}(P_{pub}, C + fQ_{IDB}).$$

If the verification is valid , then party A computes the common session keys as follows:

$$\begin{aligned} K_{A1} &= \hat{e}(S_{IDA}, C)^{r_A} \\ &= \hat{e}(sQ_{IDA}, r_B Q_{IDB})^{r_A} \\ &= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_A r_B} \end{aligned}$$

$$\begin{aligned} K_{A2} &= \hat{e}(S_{IDA}, Q_{IDB}) K_{A1} \\ &= \hat{e}(sQ_{IDA}, Q_{IDB}) K_{A1} \\ &= \hat{e}(Q_{IDA}, Q_{IDB})^{s} K_{A1} \end{aligned}$$

$$\begin{aligned} K_{A3} &= \hat{e}(S_{IDA}, C) K_{A1} \\ &= \hat{e}(sQ_{IDA}, r_B Q_{IDB}) K_{A1} \\ &= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_B} K_{A1} \end{aligned}$$

$$\begin{aligned} K_{A4} &= \hat{e}(S_{IDA}, Q_{IDB})^{r_A} K_{A1} \\ &= \hat{e}(sQ_{IDA}, Q_{IDB})^{r_A} K_{A1} \\ &= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_A} K_{A1} \end{aligned}$$

4. Cryptanalysis of Dehkordi et al's Protocol

In this section, we show that Dehkordi and Alimoradi's protocol is insecure against ephemeral key compromise attack and impersonation attack.

4.1 Impersonation Attack

In this subsection, we present impersonation attack on the Dehkordi and Alimoradi's protocol. If an impersonation attack on the protocol can be mounted successfully, the adversary E can cheat party B to believe that he has shared four secret session keys with party A, who does not involve in this session. It means that the Dehkordi and Alimoradi's protocol cannot provide mutual authenticity.

The adversary E can mount impersonation attack on the Dehkordi and Alimoradi's protocol as follows:

Step1: Party B selects a random value r_B in Z_q^*, and then computes and sends $C = r_B Q_{IDB}$ to party A.

Step2: The adversary E intercepts the message $\{C, B\}$ and selects a random value r_E in Z_q^*. Then the adversary E computes $f' = H_1(C \| IDB \| IDA)$, $T_E = -f'Q_{IDA} + r_E P$ and $\overline{Y}_E = r_E P_{pub}$. Finally, the adversary E impersonate party A sends the message $\{T_E, \overline{Y}_E, A\}$ to party B.

Step3: After receiving the message $\{T_E, \overline{Y}_E, A\}$, party B computes $f = H_1(T_E \| IDA \| IDB)$ and $f' = H_1(C \| IDB \| IDA)$, then B computes $Y = (c + f)S_{IDB}$ and checks the verification equation $\hat{e}(P, \overline{Y}_E) = \hat{e}(P_{pub}, T_E + f'Q_{IDA})$.

Since

$$\hat{e}(P, \overline{Y}_E) = \hat{e}(P, r_E P_{pub}) = \hat{e}(P, P)^{sr_E},$$

$$\begin{aligned}\hat{e}(P_{pub}, T_E + f'Q_{IDA}) &= \hat{e}(sP, T_E + f'Q_{IDA}) \\ &= \hat{e}(sP, (-f'Q_{IDA} + r_E P) + f'Q_{IDA}) \\ &= \hat{e}(sP, r_E P) \\ &= \hat{e}(P, P)^{sr_E},\end{aligned}$$

so the verification equation will be valid. Party B will send the message $\{Y, B\}$ to party A and compute the common four session keys as follows:

$$K_{B1} = \hat{e}(T_E, S_{IDB})^{r_B}$$

$$\begin{aligned}K_{B2} &= \hat{e}(Q_{IDA}, S_{IDB})K_{B1} \\ &= \hat{e}(Q_{IDA}, sQ_{IDB})K_{B1} \\ &= \hat{e}(Q_{IDA}, Q_{IDB})^s K_{B1}\end{aligned}$$

$$\begin{aligned}K_{B3} &= \hat{e}(Q_{IDA}, S_{IDB})^{r_B} K_{B1} \\ &= \hat{e}(Q_{IDA}, sQ_{IDB})^{r_B} K_{B1} \\ &= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_B} K_{B1}\end{aligned}$$

$$K_{B4} = \hat{e}(T_E, S_{IDB})K_{B1}$$

Step4: The adversary E intercepts the message $\{Y, B\}$ and ends the session. He does not know how to compute the values of four session keys. But he has cheated party B successfully in this session.

4.2 Ephemeral Key Compromise Attack

In the Dehkordi and Alimoradi's protocol, they claimed that their protocol is secure even if the adversary learns two ephemeral random values of party A and party B. In this subsection, we will show that their protocol cannot resist ephemeral key compromise attack. It means that the adversary E with two ephemeral key can compute four shared session keys successfully.

The adversary E can first use the public message $\{T, IDA, IDB\}$ to compute the value

$$f = H_1(T \| IDA \| IDB).$$

Then the adversary E can compute $(r_B + f)^{-1}$ in Z_q^* if he learns the ephemeral key r_B, which is selected by party B.

Now, the adversary E can recover the private key S_{IDB} from the public message Y as

follows:

$$(r_B + f)^{-1}Y = (r_B + f)^{-1}(r_B + f)S_{IDB}$$
$$= S_{IDB}$$

Finally, the adversary E can use ephemeral key r_B and private key S_{IDB} to compute the four shared session keys successfully. It also can be mounted this attack to party A in the similar way.

5. Improvement of Dehkordi et al's Protocol

In this section, we propose an improvement of Dehkordi and Alimoradi's protocol and analyze the security of improved protocol.

5.1 Improved Protocol

In this subsection, we propose an improvement of Dehkordi and Alimoradi's protocol. For overcoming impersonation attack, we add a new hash function $H_2 : \{0,1\}^* \rightarrow Z_q^*$ selected by KGC. The proposed protocol performs as follows.

Step1: Party B selects a random value r_B in Z_q^*, and then computes and sends $\{C = r_B Q_{IDB}, B\}$ to party A.

Step2: After receiving the message $\{C, B\}$, party A first selects a random value r_A in Z_q^*, and then computes $T = r_A Q_{IDA}$, $f_A = \hat{e}(r_A S_{IDA} + H_2(T, IDA, IDB)S_{IDA}, C + H_2(C, IDB, IDA)Q_{IDB})$. Finally A computes the value $\bar{X} = H_1(f_A, IDA, IDB)$ and sends $\{T, \bar{X}, A\}$ to party B.

Step3: After receiving $\{T, \bar{X}, A\}$, party B first computes $f_B = \hat{e}(T + H_2(T, IDA, IDB)Q_{IDA}, r_B S_{IDB} + H_2(C, IDB, IDA)S_{IDB})$ and $\bar{Y} = H_1(f_B, IDB, IDA)$, then B computes $X_B = H_1(f_B, IDA, IDB)$ and checks the verification equation $\bar{X} = X_B$. If the verification equation is valid , then party B sends the message $\{\bar{Y}, B\}$ to party A and computes the common session keys as follows.

$$
\begin{aligned}
K_{B1} &= \hat{e}(T, S_{IDB})^{r_B} \\
&= \hat{e}(r_A Q_{IDA}, sQ_{IDB})^{r_B} \\
&= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_A r_B}
\end{aligned}
$$

$$
\begin{aligned}
K_{B2} &= \hat{e}(Q_{IDA}, S_{IDB}) K_{B1} \\
&= \hat{e}(Q_{IDA}, sQ_{IDB}) K_{B1} \\
&= \hat{e}(Q_{IDA}, Q_{IDB})^{s} K_{B1}
\end{aligned}
$$

$$
\begin{aligned}
K_{B3} &= \hat{e}(Q_{IDA}, S_{IDB})^{r_B} K_{B1} \\
&= \hat{e}(Q_{IDA}, sQ_{IDB})^{r_B} K_{B1} \\
&= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_B} K_{B1}
\end{aligned}
$$

$$
\begin{aligned}
K_{B4} &= \hat{e}(T, S_{IDB}) K_{B1} \\
&= \hat{e}(r_A Q_{IDA}, sQ_{IDB}) K_{B1} \\
&= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_A} K_{B1}
\end{aligned}
$$

Step4: Party A computes $Y_A = H_1(f_A, IDB, IDA)$ and checks the verification equation

$$\overline{Y} = Y_A .$$

If the verification is valid , then party A computes the common session keys as follows:

$$
\begin{aligned}
K_{A1} &= \hat{e}(S_{IDA}, C)^{r_A} \\
&= \hat{e}(sQ_{IDA}, r_B Q_{IDB})^{r_A} \\
&= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_A r_B}
\end{aligned}
$$

$$
\begin{aligned}
K_{A2} &= \hat{e}(S_{IDA}, Q_{IDB}) K_{A1} \\
&= \hat{e}(sQ_{IDA}, Q_{IDB}) K_{A1} \\
&= \hat{e}(Q_{IDA}, Q_{IDB})^{s} K_{A1}
\end{aligned}
$$

$$
\begin{aligned}
K_{A3} &= \hat{e}(S_{IDA}, C) K_{A1} \\
&= \hat{e}(sQ_{IDA}, r_B Q_{IDB}) K_{A1} \\
&= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_B} K_{A1}
\end{aligned}
$$

$$
\begin{aligned}
K_{A4} &= \hat{e}(S_{IDA}, Q_{IDB})^{r_A} K_{A1} \\
&= \hat{e}(sQ_{IDA}, Q_{IDB})^{r_A} K_{A1} \\
&= \hat{e}(Q_{IDA}, Q_{IDB})^{sr_A} K_{A1}
\end{aligned}
$$

5.2 Security Analysis

In this subsection, we will show that the improved protocol is secure against impersonation attack and ephemeral key compromise attack.

Impersonation Attack Resilience

In the improved protocol, no adversary E can cheat party B (or party A) to believe that he has shared four secret session keys with party A (or party B).

Since BDH assumption is hard and $H_1(\cdot), H_2(\cdot)$ are two random oracles, no adversary E can use the public message $\{T, \bar{X}, A\}$ (or $\{C, B\}$) to compute the value

$$f_A = \hat{e}(r_A S_{IDA} + H_2(T, IDA, IDB) S_{IDA}, C + H_2(C, IDB, IDA) Q_{IDB}) \text{ (or}$$
$$f_B = \hat{e}(T + H_2(T, IDA, IDB) Q_{IDA}, r_B S_{IDB} + H_2(C, IDB, IDA) S_{IDB})\text{).}$$

Further, no adversary E can compute $Y_A = H_1(f_A, IDB, IDA)$ (or $X_B = H_1(f_B, IDA, IDB)$). It means that no adversary E can impersonate a party to cheat the other party sucessfully.

Ephemeral Key Compromise Attack Resilience

In the improved protocol, even if the adversary learns two ephemeral random values of party A and party B, he also cannot compute four shared session keys successfully.

Since BDH assumption is hard and $H_1(\cdot), H_2(\cdot)$ are two random oracles, no adversary E can use the public message $\{T, \bar{X}, A\}$ to compute the value

$$f_A = \hat{e}(r_A S_{IDA} + H_2(T, IDA, IDB) S_{IDA}, C + H_2(C, IDB, IDA) Q_{IDB}).$$

It means the adversary E cannot recover ephemeral key r_A (or S_{IDA}), even if he learns the private key S_{IDA} (or r_A). Similarly, the adversary E also cannot recover ephemeral key r_B (or S_{IDB}), even if he learns the private key S_{IDB} (or r_B). So no adversary can compute the session key K_{B1} and the ssion key K_{A1}. In fact, our improved protocol has the strong security property. This means that in case of disclosing of the pairs (S_{IDA}, S_{IDB}) or (r_A, S_{IDB}) or (r_B, S_{IDA}) or (r_A, r_B), the adversary cannot compute the shared session keys.

6. Conclusion

It is well known that the design of secure authenticated key exchange protocol is very hard. In this paper, we have pointed out that Dehkordi and Alimoradi's MKA protocol is insecure against the impersonation attack and the ephemeral key compromise attack. To overcome these security vulnerability, we propose an improved protocol, which can successfully avoid the weakness existed in the original Dehkordi and Alimoradi's protocol.

References

[1] W. Diffie, M. Hellman, "New directions in cryptography," *IEEE Transaction on Information Theory*, vol. 22, no. 6, pp.644-654, November, 1976.

[2] S. Blake-Wilson, D. Johnson, A. Menezes, "Key exchange protocols and their security analysis," in *Proc. of 6th IMA International Conference on Cryptography and Coding*, pp.30-45, December 17-19, 1997.

[3] L. Law, A. Menezes, M. Qu, J. Solinas, S. Vanstone, "An efficient protocol for authenticated key agreement," *Designs, Codes and Cryptography*, vol. 28, no. 2, pp.119-134, March, 2003.

[4] H. Krawczyk, "HMQV: a high-performance secure Diffie-Hellman protocol," in *Proc. of CRYPTO 2005*, pp.546-566, August 14-18, 2005.
[5] B. LaMacchia, K. Lauter, A. Mityagin, "Stronger security of authenticated key exchange," in *Proc. of ProvSec 2007*, pp.1-16, November 1-2, 2007.
[6] R. Canetti, H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *Proc. of EUROCRYPT 2001*, pp.453-474, May 6-10, 2001.
[7] D. Moriyama, T. Okamoto, "An eCK-secure authenticated key exchange protocol without random oracles," *KSII Transactions on Internet and Information Systems*, vol. 5, no. 3, pp.607-625, March, 2011.
[8] L. Harn, H. Lin, "An authenticated key agreement protocol without using one-way functions," in *Proc. of 8th National conference on Inormation Security*, pp.155-160, May, 1998.
[9] S. Yen, M. Joye, "Improved authenticated multiple-key agreement protocol," *Electronics Letters*, vol. 34, no. 18, pp.1738-1739, November, 1998.
[10] T. Wu, W. He, C. Hsu, "Security of authenticated multiple-key agreement protocols," *Electronic Letters*, vol. 35, no. 5, pp.391-392, March, 1999.
[11] L. Harn, H. Lin, "Authenticated key agreement without using one-way hash functions," *Electronics Letters*, vol. 37, no. 10, pp.629-630, May, 2001.
[12] K. Shim, "Unknown key-share attack on authenticated multiple-key agreement protocol," *Electronics Letters*, vol. 39, no. 1, pp.38-39, January, 2003.
[13] H. Zhou, L. Fan, J. Li, "Remarks on unknown key-share attack on authenticated multiple-key agreement protocol," *Electronics Letters*, vol. 39, no. 17, pp.1248-1249, September, 2003.
[14] N. Lee, C. Wu, C. Wang, "Authenticated multiple key exchange protocols based on elliptic curves and bilinear pairings," *Computers and Electrical Engineering*, vol. 34, no. 1, pp.12-20, January, 2008.
[15] D. Vo, H. Lee, C. Yeun, K. Kim, "Enhancements of authenticated multiple key exchange protocol based on bilinear pairings," *Computers and Electrical Engineering*, vol. 36, no. 1, pp.155-159, January, 2010.
[16] Q. Cheng, C. Ma, "Analysis and improvement of an authenticated multiple key exchange protocol," *Computers and Electrical Engineering*, vol. 37, no. 2, pp.187-190, March, 2011.
[17] M. Hwang, C. Lin, C. Lee, "Improved Yen-Joye's authenticated multiple-key agreement protocol," *Electronics Letters*, vol. 38, no. 23, pp.1429-1431, December, 2002.
[18] H. Yeh, H. Sun, T. Hwang, "Improved authenticated multiple-key agreement protocol," *Computers and Mathematics with Applications*, vol. 46, no. 2-3, pp.207-211, July, 2003.
[19] H. Chien, J. Jan, "Improved authenticated multiple-key agreement protocol without using conventional one-way function," *Applied Mathematics and Computation*, vol. 147, no. 2, pp.491-497, January, 2004.
[20] K. Shim, S. Woo, "Weakness in ID-based one round authenticated tripartite multiple-key agreement protocol with pairings," *Applied Mathematics and Computation*, vol. 166, no. 3, pp.523-530, July, 2005.
[21] M. Dehkordi, R. Alimoradi, "Authenticated key agreement protocol," *China Communicatons*, vol. 7, no. 5, pp.1-8, November, 2010.
[22] Z. Tan, "A provably secure identity-based authentication multiple key agreement protocol," *China Communicatons*, vol. 8, no. 2, pp.26-33, March, 2011.
[23] Z. Tan, "Efficient identity-based authenticated multiple key exchange protocol," *Computers and Electrical Engineering*, vol. 37, no. 2, pp.191-198, March, 2011.
[24] M. Dehkordi, R. Alimoradi, "Identity-based multiple key agreement scheme," *KSII Transactions on Internet and Information Systems*, vol. 5, no. 12, pp.2392-2402, December, 2011.
[25] A. Joux, "A one round protocol for tripartite Diffie-Hellman," in *Proc. of 4th international symposium on algorithmic number theory*, pp.385-394, July 2-7, 2000.
[26] D. Boneh, M. Franklin, "Identity-based encryption from the Weil pairing," *SIAM Journal on Computing*, vol. 32, no. 3, pp.586-615, May, 2001.