One-round and authenticated three-party multiple key exchange protocol from parings*

Feng LIU

School of Mathematics & Information, Ludong University, Yantai 264025, China

E-mail: liufeng23490@126.com (2010-05 Revised edition)

Abastract: One round three-party authenticated key exchange protocols are extremely important to secure communications and are now extensively adopted in network communications. These protocols allow users to communicate securely over public networks simply by using easy-to-remember long-term private keys. In 2001, Harn and Lin proposed an authentication key exchange protocol in which two parties generate four shared keys in one round, and three of these keys can provide perfect forward secrecy.This work,which aims to generalize two-party multiple key agreement sets to three-party key agreement sets,presents a three-party multiple key exchange protocol based on bilinear pairing.The proposed protocol does not require server's public key and requires only a single round. Compared with existing protocols, the proposed protocol is more efficient and provide greater security.

Keywords: Cryptography;Security;Three-party key exchange;Network security;Bilinear pairing

1 Introduction

Three-party authenticated key exchange protocols are extremely important to secure communications and are now extensively adopted in network communications. These protocols allow users to communicate securely over public networks simply by using easy-to-remember long-term private keys. Thus, secure protocols serve as basic building blocks for constructing secure, complex, higher-level protocols.For this reason, the computational efficiency, communication requirements, and round complexity of key-exchange protocols are very important and have received much attention[4].

In considering authentication between a server and each user, Lee and Hwang[8]categorizes three-party authenticated key exchange protocols into explicit server authentication and implicit server authentication. A three-party authenticated key exchange protocol with implicit server authentication can only achieve mutual authentication between two users; the server does not authenticate a user while executing the protocol. In contrast, a three-party authenticated key exchange protocol with explicit server authentication must achieve mutual authentication between a server and users. Thus, a three-party authenticated key exchange protocol with explicit server authentication typically has more steps and rounds than a three-party authenticated key exchange protocol with implicit server authentication.So,several approaches that do not use server public keys have recently been developed[8-10].

Moreover, the use of pairings has been shown promising for many three-party key exchange protocols. The pioneer work in the field was conducted by Joux[5], who showed how to implement a three-party key exchange protocol using pairings. Since in his protocol only one broadcast is required, Joux's protocol is suitable for practical implementation. However, just like

* This work is partially supported by the Ludong University Research Program under Grant NO. L20082702. I thank Peter Nose for his valuable comments.

the Diffie–Hellman protocol, Joux’s protocol does not provide authentication and thus is vulnerable to the man-in-the-middle attack. To solve the problem, Al-Riyami[12]presented several protocols some of which use pairing. Their protocols assure authenticity through use of certificates issued by a Certificate Authority (CA). The session keys are generated by both short-term keys and long-term keys. The signature of the CA assures that only the entities which are in possesion of the static keys are able to compute the session keys. Still, in a certificate system the participants must firstly verify the certificates before using the public key of a user, which requires a large amount of computing time and storage.

In 2001, Harn and Lin[1]proposed an authentication key exchange protocol which employs the digital signature technique to achieve user authentication and does not require a one-way hash function.In the protocol,two parties generate multiple shared keys after running the key agreement protocol.More precisely, if two parties compute and transmit n public keys of Diffie – Hellman protocol to each other, then n^2-1 session keys are shared between them. Later, Hwang et al. [3]proposed an efficient authentication key exchange protocol requiring less computation than Harn and Lin’ s scheme [1]. Nevertheless, the scheme [3] was broken by Lee and Wu [6]by the modification attack. Recently, Lee et al. [7]proposed two authenticated multiple key exchange protocols: one is based on ECC and the other is based on bilinear pairings. These protocols let two parties share not only one but also four session keys in authenticated manner.However, Vo et al.[13]demonstrated an impersonation attack on Lee's pairing-based authenticated key exchange protocol. They also showed that, using a long-term public key of an entity only, any attacker can impersonate the party to agree some session keys with another party. Consequently, Lee et al.’ s protocol fails to provide authenticity as they had claimed. Furthermore, they indicated that perfect forward secrecy of Lee's protocol was not guaranteed. Thus, Vo et al. proposed a simple modification to the protocol which could withstand their own attacks.

In this paper we examine the two-party authenticated key agreement protocol using pairing operations from[6]and three-party authenticated key agreement protocol using pairing operations from[2]. The main contribution includes the proposal of an one round three-party authenticated multiple key agreement protocol using pairings, which feature all security attributes[2]. Since our proposed protocol does not require any server’s public keys, it seems very simple and efficient, and can be used in many practical scenarios.Moreover, the available number of shared session keys in the protocol is more than that in [6,13].

The rest of the paper is organized as follows: Section 2 briefly explains preliminary concepts, i.e. bilinear maps and the associated computational problems. Section 3 reviews Lee's multiple key exchang protocol and Hobol's three-party protocol, and analyzes their security.Our proposed protocol is described in Section 4 with the corresponding security and efficiency discussion. In Section 5, the efficiency and security comparison of the proposed protocol and competitive protocol is conducted. Finally, a conclusion is drawn in Section 6.

2 Preliminaries

In this section, we briefly describe preliminaries which are needed later in the paper. We give the basic definition and properties of bilinear pairings, the computational problems which are fundamental when discussing authenticated key agreement protocols, security attributes desired for sound authenticated key agreement protocols and efficiency properties.

2.1 Bilinear Pairings

Let G_1 be an additive group generated by P, whose order is a prime q, and G_2 be a multiplicative group of the same order q; a bilinear pairing is a map

$$e: G_1 \times G_1 \rightarrow G_2$$

with the following properties :

- Bilinear: for all $P, Q \in G_1$ and $e(c_1 P, c_2 Q) = e(P, Q)^{c_1 c_2}$.
- Non-degenerate: there exists $P \in G_1$ such that $e(P, P) \neq 1$.
- Computable: given $P, Q \in G_1$, there is an efficient algorithm to compute $e(P, Q)$.

2.2 Computational problems

- Computational Diffie-Hellman (CDH) problem: given a triple

$$(P, c_1 P, c_2 P) \in G_1$$

for $c_1, c_2 \in Z_q^*$, find the element $c_1 c_2 P$.

- Decision Diffie-Hellman (DDH) problem: given a quadruple

$$e(P, c_1 P, c_2 P, c_3 P) \in G_1$$

for $c_1, c_2, c_3 \in Z_q^*$, decide whether $c_3 = c_1 c_2 \bmod q$ or not.

- Gap Diffie-Hellman (GDH) problem: a class of problems where the CDH problem is hard but the DDH problem is easy.

Groups where the CDH problem is hard but the DDH problem is easy are called GDH groups.

3. Lee's and Hobol's authenticated key exchange protocols based on bilinear pairings

This section briefly reviews the two-party multiple protocol developed by Lee[6], and the three-party protocol developed by Holbl[2], and explicates the weaknesses of them. Let $\mathcal{A}$, $\mathcal{B}$ and $\mathcal{C}$ be three communication parties.

3.1 Lee's two-party multiple protocol from pairings

We firstly review Lee's multiple key exchange protocol based on bilinear pairings.

Initiate

Let $X_{\mathcal{U}} \in Z_q^*$ and $Y_{\mathcal{U}} (= X_{\mathcal{U}} P)$ be $\mathcal{U}$'s long-term private key and long-term public key, $Cert(Y_{\mathcal{U}})$ be the certificate of $\mathcal{U}$ s long-term public key signed by a trusted party（TP）

Ex-massage

$\mathcal{A}$:chooses $a_1, a_2 \in \mathbb{Z}_q^*$,and computes $T_{\mathcal{A}1} = a_1 P$, $T_{\mathcal{A}2} = a_2 P$, $S_{\mathcal{A}} = (a_1 K_{\mathcal{A}1} + a_2 K_{\mathcal{A}2}) T_{\mathcal{A}1} + X_{\mathcal{A}} T_{\mathcal{A}2}$;

$\mathcal{A} \rightarrow \mathcal{B}$: $\{T_{\mathcal{A}1}, T_{\mathcal{A}2}, S_{\mathcal{A}}, Cert(Y_{\mathcal{A}})\}$,where $K_{\mathcal{A}i}$ is the x-coordinate value of $T_{\mathcal{A}i}$.

$\mathcal{B}$: chooses $b_1, b_2 \in \mathbb{Z}_q^*$,and computes $T_{\mathcal{B}1} = b_1 P$, $T_{\mathcal{B}2} = b_2 P$, $S_{\mathcal{B}} = (b_1 K_{\mathcal{B}1} + b_2 K_{\mathcal{B}2}) T_{\mathcal{B}1} + X_{\mathcal{B}} T_{\mathcal{B}2}$;

$\mathcal{B} \to \mathcal{A} : \{T_{\mathcal{B}1}, T_{\mathcal{B}2}, S_{\mathcal{B}}, Cert(Y_{\mathcal{B}})\}$,where $K_{\mathcal{B}i}$ is the x-coordinate value of $T_{\mathcal{B}i}$.

Co-keys

$$\mathcal{A}: \ e(S_{\mathcal{B}}, P) \overset{?}{=} e(K_{\mathcal{B}1}T_{\mathcal{B}1} + K_{\mathcal{B}2}T_{\mathcal{B}2}, T_{\mathcal{B}1})e(T_{\mathcal{B}2}, Y_{\mathcal{B}})$$

$$K_{11} = e(a_1 T_{\mathcal{B}1}, Y_{\mathcal{A}} + Y_{\mathcal{B}});$$

$$K_{12} = e(a_1 T_{\mathcal{B}2}, Y_{\mathcal{A}} + Y_{\mathcal{B}});$$

$$K_{21} = e(a_2 T_{\mathcal{B}1}, Y_{\mathcal{A}} + Y_{\mathcal{B}});$$

$$K_{22} = e(a_2 T_{\mathcal{B}2}, Y_{\mathcal{A}} + Y_{\mathcal{B}}).$$

$$\mathcal{B}: e(S_{\mathcal{A}}, P) \overset{?}{=} e(K_{\mathcal{A}1}T_{\mathcal{A}1} + K_{\mathcal{A}2}T_{\mathcal{A}2}, T_{\mathcal{A}1})e(T_{\mathcal{A}2}, Y_{\mathcal{A}})$$

$$K_{11} = e(T_{\mathcal{A}1} b_1, Y_{\mathcal{A}} + Y_{\mathcal{B}});$$

$$K_{12} = e(T_{\mathcal{A}1} b_2, Y_{\mathcal{A}} + Y_{\mathcal{B}});$$

$$K_{21} = e(T_{\mathcal{A}2} b_1, Y_{\mathcal{A}} + Y_{\mathcal{B}});$$

$$K_{22} = e(T_{\mathcal{A}2} b_2, Y_{\mathcal{A}} + Y_{\mathcal{B}}).$$

However, Vo et al. demonstrated an impersonation attack on the protocol. And,they showed that, using a long-term public key of an entity only, any attacker could impersonate the entity to agree some session keys with another entity. Consequently, Lee et al.'s protocol fails to provide authenticity as they have claimed. Furthermore,Vo indicated that perfect forward secrecy of their protocol was not guaranteed.When attackers know long-term private keys of $\mathcal{A}$ and $\mathcal{B}$, $x_{\mathcal{A}}$ and $x_{\mathcal{B}}$, respectively, the attackers easily compute the previous session keys as follows:

$$K_{11} = e(a_1 T_{\mathcal{B}1}, Y_{\mathcal{A}} + Y_{\mathcal{B}}) = e(T_{\mathcal{B}1}, a_1(X_{\mathcal{A}} + X_{\mathcal{B}})P) = e(T_{\mathcal{B}1}, (X_{\mathcal{A}} + X_{\mathcal{B}})T_{\mathcal{A}1})$$

Thus, They proposed a simple modification to the protocol which can withstand our attack. Unfortunately,in Vo's enhanced protocol each participant must firstly verify the certificates before using the public key of a user, which required a large amount of computing time and storage.

3.2 Holbl's authenticated three-party protocol from pairings

In this section, we review Hobol's three-party key exchange protocol based on bilinear pairings.

Initiate

For a user with identity ID_i , the public key is derived as $Q_i = H(ID_i)$ and the private key as $S_i = sQ_i$ .Both parameters are computed by the $\mathcal{PKG}$ and, afterwards, S_i is issued to the party via **a secure channel**.

Ex-massage

$\mathcal{A}$: chooses $a, r_{\mathcal{A}} \in \mathbb{Z}_q^*$,and computes $P_{\mathcal{A}} = aP, U_{\mathcal{A}} = r_{\mathcal{A}}Q_{\mathcal{A}}, V_{\mathcal{A}} = (r_{\mathcal{A}} + H(P_{\mathcal{A}}, U_{\mathcal{A}}))S_{\mathcal{A}}$;

$$\mathcal{A} \to \mathcal{B}, \mathcal{C} : \{P_{\mathcal{A}}, U_{\mathcal{A}}, V_{\mathcal{A}}\}.$$

$\mathcal{B}$: chooses $b, r_{\mathcal{B}} \in \mathbb{Z}_q^*$,and computes $P_{\mathcal{B}} = aP, U_{\mathcal{B}} = r_{\mathcal{B}} Q_{\mathcal{B}}, V_{\mathcal{B}} = (r_{\mathcal{B}} + H(P_{\mathcal{B}}, U_{\mathcal{B}}))S_{\mathcal{B}}$;

$$\mathcal{B} \to \mathcal{A}, \mathcal{C} : \{P_{\mathcal{B}}, U_{\mathcal{B}}, V_{\mathcal{B}}\}.$$

$\mathcal{C}$: chooses $c, r_{\mathcal{C}} \in \mathbb{Z}_q^*$,and computes $P_{\mathcal{C}} = cP, U_{\mathcal{C}} = r_{\mathcal{C}} Q_{\mathcal{C}}, V_{\mathcal{C}} = (r_{\mathcal{C}} + H(P_{\mathcal{C}}, U_{\mathcal{C}}))S_{\mathcal{C}}$;

$$\mathcal{C} \to \mathcal{A}, \mathcal{B} : \{P_{\mathcal{C}}, U_{\mathcal{C}}, V_{\mathcal{C}}\}.$$

Co-key

$$\mathcal{A} : \ e(V_{\mathcal{B}} + V_{\mathcal{C}}, P) \overset{?}{=} e((r_{\mathcal{B}} + H(P_{\mathcal{B}}, U_{\mathcal{B}}))Q_{\mathcal{B}} + (r_{\mathcal{C}} + H(P_{\mathcal{C}}, U_{\mathcal{C}}))Q_{\mathcal{C}}, P_{\mathcal{PKG}})$$

$$K_{\mathcal{A}} = e(P_{\mathcal{B}}, P_{\mathcal{C}})^a = e(P, P)^{a+b+c};$$

$$\mathcal{B} : \ e(V_{\mathcal{A}} + V_{\mathcal{C}}, P) \overset{?}{=} e((r_{\mathcal{A}} + H(P_{\mathcal{A}}, U_{\mathcal{A}}))Q_{\mathcal{A}} + (r_{\mathcal{C}} + H(P_{\mathcal{C}}, U_{\mathcal{C}}))Q_{\mathcal{C}}, P_{\mathcal{PKG}})$$

$$K_{\mathcal{B}} = e(P_{\mathcal{A}}, P_{\mathcal{C}})^b = e(P, P)^{a+b+c};$$

$$\mathcal{C} : \ e(V_{\mathcal{A}} + V_{\mathcal{B}}, P) \overset{?}{=} e((r_{\mathcal{A}} + H(P_{\mathcal{A}}, U_{\mathcal{A}}))Q_{\mathcal{A}} + (r_{\mathcal{B}} + H(P_{\mathcal{B}}, U_{\mathcal{B}}))Q_{\mathcal{B}}, P_{\mathcal{PKG}})$$

$$K_{\mathcal{C}} = e(P_{\mathcal{A}}, P_{\mathcal{B}})^c = e(P, P)^{a+b+c}.$$

Observe that the proposed protocol requires the availability of **a secure channel** from $\mathcal{PKG}$ to each of the participants individually. However, communication over **the secure channel** is clearly not publicly veriable,when a dispute emerged. Moreover, communicating parties can share only one session key after running the key agreement protocol.

4. Proposed three-party multiple key agreement protocol

We note that a distinctive feature of Lee protocol is that no secure channels between $\mathcal{PKG}$ and the participants are assumed. All communication is done over (authenticated) public channels using public key signature. And, the initialization of Lee is done without any interaction between the $\mathcal{PKG}$ and the participants. In fact, participants may enter or leave theprotocol *dynamically*; the only requirement is that a participant holds a registered public key.

Compared to Holbl protocol, we thy to add the requirement for the multiple protocol that if parties compute and transmit n public keys of Diffie–Hellman protocol to each other, then $n^2 - 1$ session keys are shared between them.

Based on our observation we have just made about why the protocols are infeasible, we propose that our enhanced protocol should be modified in a hybrid way. The setup phase is kept unchanged from Lee protocol.

We now describe the revised protocol,as follows:

Initiate

For a user with long-term private key $X_i \in Z_q^*$,the long-term public key is derived as $Y_i(= X_i P)$ and the certificate of Y_i is $Cert(Y_i)$ which is signed by a trusted party($\mathcal{TP}$).

$\underline{\mathcal{Ex\text{-}massage}}$

$\mathcal{A}$:chooses $a_1, a_2 \in \mathbb{Z}_q^*$,and computes

$$T_{A1} = a_1 P, T_{A2} = a_2 P, S_{A1} = a_1 X_A + a_2, S_{A2} = a_2 X_A + a_1;$$

$$\mathcal{A} \to \mathcal{B}, \mathcal{C} : \{T_{A1}, T_{A2}, S_{A1}, S_{A2}, Cert(Y_A)\} ;$$

$\mathcal{B}$: chooses $b_1, b_2 \in \mathbb{Z}_q^*$,and computes

$$T_{B1} = b_1 P, T_{B2} = b_2 P, S_{B1} = b_1 X_B + b_2, S_{B2} = b_2 X_B + b_1;$$

$$\mathcal{B} \to \mathcal{A}, \mathcal{C} : \{T_{B1}, T_{B2}, S_{B1}, S_{B2}, Cert(Y_B)\} ;$$

$\mathcal{C}$: chooses $c_1, c_2 \in \mathbb{Z}_q^*$,and computes

$$T_{C1} = c_1 P, T_{C2} = c_2 P, S_{C1} = c_1 X_C + c_2, S_{C2} = c_2 X_C + c_1;$$

$$\mathcal{C} \to \mathcal{A}, \mathcal{B} : \{T_{C1}, T_{C2}, S_{C1}, S_{C2}, Cert(Y_C)\} ;$$

$\underline{\mathcal{Co\text{-}keys}}$

$\mathcal{A}$:Upon receiving $\{T_{B1}, T_{B2}, S_{B1}, S_{B2}, Cert(Y_B)\}$ and $\{T_{C1}, T_{C2}, S_{C1}, S_{C2}, Cert(Y_C)\}$, $\mathcal{A}$ checks the equations:

$$e((S_{B1} + S_{B2})P - (T_{B1} + T_{B2}), P) \overset{?}{=} e(T_{B1} + T_{B2}, Y_B) ,$$

$$e((S_{C1} + S_{C2})P - (T_{C1} + T_{C2}), P) \overset{?}{=} e(T_{C1} + T_{C2}, Y_C) ;$$

If these verification hold, $\mathcal{A}$ computes eight shared session keys as follows:

$$K_{111} = e(a_1 X_A T_{B1}, S_{C1}P - T_{C2}) \cdot e(a_1 X_A T_{C1}, S_{B1}P - T_{B2}) \cdot e(a_1(S_{B1}P - T_{B2}), S_{C1}P - T_{C2})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_1 b_1 c_1} ;$$

$$K_{112} = e(a_1 X_A T_{B1}, S_{C2}P - T_{C1}) \cdot e(a_1 X_A T_{C2}, S_{B1}P - T_{B2}) \cdot e(a_1(S_{B1}P - T_{B2}), S_{C2}P - T_{C1})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_1 b_1 c_2} ;$$

$$K_{121} = e(a_1 X_A T_{B2}, S_{C1}P - T_{C2}) \cdot e(a_1 X_A T_{C1}, S_{B2}P - T_{B1}) \cdot e(a_1(S_{B2}P - T_{B1}), S_{C1}P - T_{C2})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_1 b_2 c_1} ;$$

$$K_{122} = e(a_1 X_A T_{B2}, S_{C2}P - T_{C1}) \cdot e(a_1 X_A T_{C2}, S_{B2}P - T_{B1}) \cdot e(a_1(S_{B2}P - T_{B1}), S_{C2}P - T_{C1})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_1 b_2 c_2} ;$$

$$K_{211} = e(a_2 X_A T_{B1}, S_{C1}P - T_{C2}) \cdot e(a_2 X_A T_{C1}, S_{B1}P - T_{B2}) \cdot e(a_2(S_{B1}P - T_{B2}), S_{C1}P - T_{C2})$$

$$= e((X_{\mathcal{A}}X_{\mathcal{B}} + X_{\mathcal{A}}X_{\mathcal{C}} + X_{\mathcal{B}}X_{\mathcal{C}})P, P)^{a_2b_1c_1};$$

$$K_{212} = e(a_2X_{\mathcal{A}}T_{\mathcal{B}1}, S_{\mathcal{C}2}P - T_{\mathcal{C}1}) \cdot e(a_2X_{\mathcal{A}}T_{\mathcal{C}2}, S_{\mathcal{B}1}P - T_{\mathcal{B}2}) \cdot e(a_2(S_{\mathcal{B}1}P - T_{\mathcal{B}2}), S_{\mathcal{C}2}P - T_{\mathcal{C}1})$$

$$= e((X_{\mathcal{A}}X_{\mathcal{B}} + X_{\mathcal{A}}X_{\mathcal{C}} + X_{\mathcal{B}}X_{\mathcal{C}})P, P)^{a_2b_1c_2};$$

$$K_{221} = e(a_2X_{\mathcal{A}}T_{\mathcal{B}2}, S_{\mathcal{C}1}P - T_{\mathcal{C}2}) \cdot e(a_2X_{\mathcal{A}}T_{\mathcal{C}1}, S_{\mathcal{B}2}P - T_{\mathcal{B}1}) \cdot e(a_2(S_{\mathcal{B}2}P - T_{\mathcal{B}1}), S_{\mathcal{C}1}P - T_{\mathcal{C}2})$$

$$= e((X_{\mathcal{A}}X_{\mathcal{B}} + X_{\mathcal{A}}X_{\mathcal{C}} + X_{\mathcal{B}}X_{\mathcal{C}})P, P)^{a_2b_2c_1};$$

$$K_{222} = e(a_2X_{\mathcal{A}}T_{\mathcal{B}2}, S_{\mathcal{C}2}P - T_{\mathcal{C}1}) \cdot e(a_2X_{\mathcal{A}}T_{\mathcal{C}2}, S_{\mathcal{B}2}P - T_{\mathcal{B}1}) \cdot e(a_2(S_{\mathcal{B}2}P - T_{\mathcal{B}1}), S_{\mathcal{C}2}P - T_{\mathcal{C}1})$$

$$= e((X_{\mathcal{A}}X_{\mathcal{B}} + X_{\mathcal{A}}X_{\mathcal{C}} + X_{\mathcal{B}}X_{\mathcal{C}})P, P)^{a_2b_2c_2}\ .$$

$\mathcal{B}$:Upon receiving $\{T_{\mathcal{A}1}, T_{\mathcal{A}2}, S_{\mathcal{A}1}, S_{\mathcal{A}2}, Cert(Y_{\mathcal{A}})\}$ and $\{T_{\mathcal{C}1}, T_{\mathcal{C}2}, S_{\mathcal{C}1}, S_{\mathcal{C}2}, Cert(Y_{\mathcal{C}})\}$, $\mathcal{B}$ checks the equations:

$$e((S_{\mathcal{A}1} + S_{\mathcal{A}2})P - (T_{\mathcal{A}1} + T_{\mathcal{A}2}), P) \overset{?}{=} e(T_{\mathcal{A}1} + T_{\mathcal{A}2}, Y_{\mathcal{A}})\ ,$$

$$e((S_{\mathcal{C}1} + S_{\mathcal{C}2})P - (T_{\mathcal{C}1} + T_{\mathcal{C}2}), P) \overset{?}{=} e(T_{\mathcal{C}1} + T_{\mathcal{C}2}, Y_{\mathcal{C}})\ ;$$

If these verification hold, $\mathcal{B}$ computes eight shared session keys as follows:

$$K_{111} = e(b_1X_{\mathcal{B}}T_{\mathcal{A}1}, S_{\mathcal{C}1}P - T_{\mathcal{C}2}) \cdot e(b_1X_{\mathcal{B}}T_{\mathcal{C}1}, S_{\mathcal{A}1}P - T_{\mathcal{A}2}) \cdot e(b_1(S_{\mathcal{A}1}P - T_{\mathcal{A}2}), S_{\mathcal{C}1}P - T_{\mathcal{C}2})$$

$$= e((X_{\mathcal{A}}X_{\mathcal{B}} + X_{\mathcal{A}}X_{\mathcal{C}} + X_{\mathcal{B}}X_{\mathcal{C}})P, P)^{a_1b_1c_1};$$

$$K_{112} = e(b_1X_{\mathcal{B}}T_{\mathcal{A}1}, S_{\mathcal{C}2}P - T_{\mathcal{C}1}) \cdot e(b_1X_{\mathcal{B}}T_{\mathcal{C}2}, S_{\mathcal{A}1}P - T_{\mathcal{A}2}) \cdot e(b_1(S_{\mathcal{A}1}P - T_{\mathcal{A}2}), S_{\mathcal{C}2}P - T_{\mathcal{C}1})$$

$$= e((X_{\mathcal{A}}X_{\mathcal{B}} + X_{\mathcal{A}}X_{\mathcal{C}} + X_{\mathcal{B}}X_{\mathcal{C}})P, P)^{a_1b_1c_2};$$

$$K_{121} = e(b_2X_{\mathcal{B}}T_{\mathcal{A}1}, S_{\mathcal{C}1}P - T_{\mathcal{C}2}) \cdot e(b_2X_{\mathcal{B}}T_{\mathcal{C}1}, S_{\mathcal{A}1}P - T_{\mathcal{A}2}) \cdot e(b_2(S_{\mathcal{A}1}P - T_{\mathcal{A}2}), S_{\mathcal{C}1}P - T_{\mathcal{C}2})$$

$$= e((X_{\mathcal{A}}X_{\mathcal{B}} + X_{\mathcal{A}}X_{\mathcal{C}} + X_{\mathcal{B}}X_{\mathcal{C}})P, P)^{a_1b_2c_1};$$

$$K_{122} = e(b_2X_{\mathcal{B}}T_{\mathcal{A}1}, S_{\mathcal{C}2}P - T_{\mathcal{C}1}) \cdot e(b_2X_{\mathcal{B}}T_{\mathcal{C}2}, S_{\mathcal{A}1}P - T_{\mathcal{A}2}) \cdot e(b_2(S_{\mathcal{A}1}P - T_{\mathcal{A}2}), S_{\mathcal{C}2}P - T_{\mathcal{C}1})$$

$$= e((X_{\mathcal{A}}X_{\mathcal{B}} + X_{\mathcal{A}}X_{\mathcal{C}} + X_{\mathcal{B}}X_{\mathcal{C}})P, P)^{a_1b_2c_2};$$

$$K_{211} = e(b_1X_{\mathcal{B}}T_{\mathcal{A}2}, S_{\mathcal{C}1}P - T_{\mathcal{C}2}) \cdot e(b_1X_{\mathcal{B}}T_{\mathcal{C}1}, S_{\mathcal{A}2}P - T_{\mathcal{A}1}) \cdot e(b_1(S_{\mathcal{A}2}P - T_{\mathcal{A}1}), S_{\mathcal{C}1}P - T_{\mathcal{C}2})$$

$$= e((X_{\mathcal{A}}X_{\mathcal{B}} + X_{\mathcal{A}}X_{\mathcal{C}} + X_{\mathcal{B}}X_{\mathcal{C}})P, P)^{a_2b_1c_1};$$

$$K_{212} = e(b_1X_{\mathcal{B}}T_{\mathcal{A}2}, S_{\mathcal{C}2}P - T_{\mathcal{C}1}) \cdot e(b_1X_{\mathcal{B}}T_{\mathcal{C}2}, S_{\mathcal{A}2}P - T_{\mathcal{A}1}) \cdot e(b_1(S_{\mathcal{A}2}P - T_{\mathcal{A}1}), S_{\mathcal{C}2}P - T_{\mathcal{C}1})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_2 b_1 c_2} \ ;$$

$$K_{221} = e(b_2 X_B T_{A2}, S_{C1}P - T_{C2}) \cdot e(b_2 X_B T_{C1}, S_{A2}P - T_{A1}) \cdot e(b_2(S_{A2}P - T_{A1}), S_{C1}P - T_{C2})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_2 b_2 c_1} \ ;$$

$$K_{222} = e(b_2 X_B T_{A2}, S_{C2}P - T_{C1}) \cdot e(b_2 X_B T_{C2}, S_{A2}P - T_{A1}) \cdot e(b_2(S_{A2}P - T_{A1}), S_{C2}P - T_{C1})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_2 b_2 c_2} \ .$$

C :Upon receiving $\{T_{A1}, T_{A2}, S_{A1}, S_{A2}, Cert(Y_A)\}$ and $\{T_{B1}, T_{B2}, S_{B1}, S_{B2}, Cert(Y_B)\}$, C checks the equations:

$$e((S_{A1} + S_{A2})P - (T_{A1} + T_{A2}), P) \overset{?}{=} e(T_{A1} + T_{A2}, Y_A) \ ,$$

$$e((S_{B1} + S_{B2})P - (T_{B1} + T_{B2}), P) \overset{?}{=} e(T_{B1} + T_{B2}, Y_B) \ ;$$

If these verification hold, C computes eight shared session keys as follows:

$$K_{111} = e(c_1 X_C T_{B1}, S_{A1}P - T_{A2}) \cdot e(c_1 X_C T_{A1}, S_{B1}P - T_{B2}) \cdot e(c_1(S_{B1}P - T_{B2}), S_{A1}P - T_{A2})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_1 b_1 c_1} \ ;$$

$$K_{112} = e(c_2 X_C T_{B1}, S_{A1}P - T_{A2}) \cdot e(c_2 X_C T_{A1}, S_{B1}P - T_{B2}) \cdot e(c_2(S_{B1}P - T_{B2}), S_{A1}P - T_{A2})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_1 b_1 c_2} \ ;$$

$$K_{121} = e(c_1 X_C T_{B2}, S_{A1}P - T_{A2}) \cdot e(c_1 X_C T_{A1}, S_{B2}P - T_{B1}) \cdot e(c_1(S_{B2}P - T_{B1}), S_{A1}P - T_{A2})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_1 b_2 c_1} \ ;$$

$$K_{122} = e(c_2 X_C T_{B2}, S_{A1}P - T_{A2}) \cdot e(c_2 X_C T_{A1}, S_{B2}P - T_{B1}) \cdot e(c_2(S_{B2}P - T_{B1}), S_{A1}P - T_{A2})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_1 b_2 c_2} \ ;$$

$$K_{211} = e(c_1 X_C T_{B1}, S_{A2}P - T_{A1}) \cdot e(c_1 X_C T_{A2}, S_{B1}P - T_{B2}) \cdot e(c_1(S_{B1}P - T_{B2}), S_{A2}P - T_{A1})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_2 b_1 c_1} \ ;$$

$$K_{212} = e(c_2 X_C T_{B1}, S_{A2}P - T_{A1}) \cdot e(c_2 X_C T_{A2}, S_{B1}P - T_{B2}) \cdot e(c_2(S_{B1}P - T_{B2}), S_{A2}P - T_{A1})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_2 b_1 c_2} \ ;$$

$$K_{221} = e(c_1 X_C T_{B2}, S_{A2}P - T_{A1}) \cdot e(c_1 X_C T_{A2}, S_{B2}P - T_{B1}) \cdot e(c_1(S_{B2}P - T_{B1}), S_{A2}P - T_{A1})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_2 b_2 c_1};$$

$$K_{222} = e(c_2 X_C T_{B2}, S_{A2}P - T_{A1}) \cdot e(c_2 X_C T_{A2}, S_{B2}P - T_{B1}) \cdot e(c_2(S_{B2}P - T_{B1}), S_{A2}P - T_{A1})$$

$$= e((X_A X_B + X_A X_C + X_B X_C)P, P)^{a_2 b_2 c_2} \quad .$$

5. Analysis

Correctness. The correctness of shared keys is easily to notice by comparing key computation in *Co-keys* verification phase in Section 4. The following is the correctness of tetrad $\{T_{A1}, T_{A2}, S_{A1}, S_{A2}\}$ (similar for tetrads $\{T_{B1}, T_{B2}, S_{B1}, S_{B2}\}$ and $\{T_{C1}, T_{C2}, S_{C1}, S_{C2}\}$) verification:

$$e((S_{A1} + S_{A2})P - (T_{A1} + T_{A2}), P) = e((a_1 + a_2)X_A P, P) = e(T_{A1} + T_{A2}, Y_A)$$

Trivial attack. An attacker may directly try to compute the session key from the transmitted transcripts $\{T_{i1}, T_{i2}, S_{i1}, S_{i2}, Cert(Y_i)\}$. However, due to the difficulties of the discrete logarithm problem and CCDH problem, the trivial attack is useless to our proposed three-party protocol.

Impersonation attack. Impersonation attack is infeasible since if an attacker wants to produce a forged message of $\mathcal{A}$,the attacker has to compute S_{A1} and S_{A2} in order to pass $\mathcal{B}$ or $\mathcal{C}$'s verification. He/she has to solve the discrete logarithm problem and Schnorr signature, but the signature scheme has been proven to be secure under the random oracle modle[11]. Furthermore, given T_{A1} and T_{A2} of the attacker's choice, he/she still needs to compute $a_1 X_A P = a_1 Y_A$, $a_2 X_A P = a_2 Y_A$. However, computing $a_1 X_A P$, $a_2 X_A P$ from Y_A is to solve the computational Diffie–Hellman problem in group $\mathbb{G}_1$, which is believed to be computationally infeasible.

Known key security. Because random numbers are used in each step differently, the shared keys also differ for each step. Even the shared keys in a protocol session are exposed, attackers fail to relate these keys with the keys in other session since they are independent.

Key-compromise impersonation. If $\mathcal{A}$'s long-term private key is exposed, it does not enable an attacker to impersonate $\mathcal{B}$ or $\mathcal{C}$ to $\mathcal{A}$ .This can be eliminated since $\mathcal{A}$ uses $\mathcal{B}$ or $\mathcal{C}$'s public key in her shared secret keys computation. Even the attacker could masquerade the message sent to $\mathcal{A}$ in *Co-keys*, but, ultimately, the attacker is unable to compute the shared keys without knowing $\mathcal{B}$ or $\mathcal{C}$ 'slong-term private keys.

Perfect forward secrecy. In our protocol, when long-term private keys of each party, X_A , X_B and X_C are revealed, deriving session keys is still infeasible. Intuitively, we could see that, an attacker is given Y_A , Y_B and $T_{B1}(= b_1 P)$ for instance, the attacker has to find out $b_1 Y_B$ in order to compute the shared key K_{111} .However,this is a computation Bilinear Diffie–Hellman problem which is computationally infeasible.

Performance. The performance comparison between Holblal.'s protocol and ours is presented in Table 1. In this table, Sm and Pa represent for scalar multiplication and point addition on an elliptic curve, respectively; e is pairing computation and Mul is the modular multiplication. As shown in this table, our revised protocol has the same computation compared with Holblal.'s protocol at all steps including the key computation. At this step, we require three more elliptic curve point multiplication operations in each key computation,and require one less parings operation. However, the elliptic curve point multiplication operation is negligible comparing with pairing computation. Therefore, we could consider the performance of the revised protocol is efficient than the original one .

Table 1 Performance evaluation

(iff an average of one session key)

Step	*Holbl*[2]	*Our protocol*
Computation of short-term public keys	$3Sm$	$0.25Sm+0.25Mul$
Verification	$2e+2Pa+2Sm$	$0.5e+0.5Pa+0.25Sm$
Key computation	$e+Sm$	$0.25e+0.25Pa+0.38Sm$
Available shared session keys	1	8
Secure channal	*Yes*	*No*

6. Conclusion

In this paper, we showed that Lee et al.'s authenticated multiple key exchange protocol based on bilinear pairings and Holbl's authenticated three-party protocol fail to provide authenticity or need a secure channel,respectively. We also provided a revised version of these protocol which prevent the weakneasses, but yet which does not add significantly to the communications or computational overhead for the protocol. Note that, bilinear pairings can provide beneficial properties, one has to carefully utilize them when designing cryptographic protocols.

References

[1] Harn L, Lin H-Y. Authenticated key agreement without using one-way hash function. Electron Lett ,2001;37(10):629-630.

[2] Holbl M, Welzer T, Brumen B.Two proposed identity-based three-party authenticated key agreement protocols from pairings.computers&security 29 (2010)244–252

[3] Hwang R J, Shiau S H, Lai C H. An enhanced authentication key exchange protocol. Advanced information networking and applications, 2003. In: Proceedings of the 17th international conference on AINA 2003; p. 202 – 205.

[4] Jeong I.R, Katz J. and Lee D.H. One-Round Protocols for Two-Party Authenticated Key Exchange. M. Jakobsson, M. Yung, J. Zhou (Eds.): ACNS 2004, LNCS 3089, pp. 220 – 232, 2004

[5] Joux A. A one round protocol for tripartite Diffie–Hellman. In:Proceedings of the 4th international symposium on algorithmic number theory. LNCS 1838. USA: Springer-Verlag;2000. p. 385–94.

[6] Lee N-Y, Wu C-N. Improved authentication key exchange protocol without using one-way

hash function. ACM Operat Syst Rev, 2004,38(2):85-92.

[7] Lee N-Y, Wu C-N, Wang C-C. Authenticated multiple key exchange protocols based on elliptic curves and bilinear pairings. Comput Electr Eng, 2008,34(1):12 – 20.

[8] Lee T-F, Hwang T. Simple password-based three-party authenticated key exchange without server public keys. Information Sciences 180 (2010) 1702 – 1714

[9] Lin C-L, Sun H-M, Steiner M, Hwang T. Three-party encrypted key exchange without server public-keys. IEEE Communications Letters 5 (12) (2001)497 – 499.

[10] Lu R, Cao Z. Simple three-party key exchange protocol. Computers and Security, 26 (1) (2007) 94 – 97

[11]Pointcheval D and Stern J. Security proofs for signatures. Eurocrypt'96,387-398, 1996

[12] S S Al-Riyami and K G Paterson. Tripartite authenticated key agreement protocols from pairings. Cryptology eprint Archive 2002, Report 2002/035.

[13] Vo D-L, Lee H, Yeun C-Y, Kim K. Enhancements of authenticated multiple key exchange protocol based on bilinear pairings.Computers and Electrical Engineering,36 (2010) 155-159