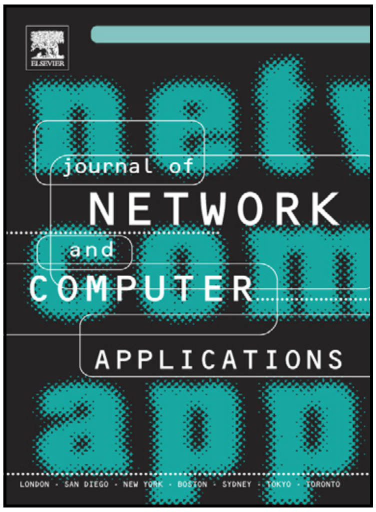

ELSEVIER
journal of
NETWORK
and
COMPUTER
APPLICATIONS
LONDON · SAN DIEGO · NEW YORK · BOSTON · SYDNEY · TOKYO · TORONTO

www.elsevier.com/locate/jnca

New identity-based three-party authenticated key agreement protocol with provable security

Hu Xiong, Zhong Chen, Fagen Li

PII: S1084-8045(12)00218-4
DOI: http://dx.doi.org/10.1016/j.jnca.2012.10.001
Reference: YJNCA939

To appear in: *Journal of Network and Computer Applications*

Received date: 5 November 2011
Revised date: 13 June 2012
Accepted date: 13 October 2012

Cite this article as: Hu Xiong, Zhong Chen and Fagen Li, New identity-based three-party authenticated key agreement protocol with provable security, *Journal of Network and Computer Applications,* http://dx.doi.org/10.1016/j.jnca.2012.10.001

New Identity-based Three-party authenticated key agreement protocol with Provable Security

Hu Xiong[a,b,c,,*], Zhong Chen[b], Fagen Li[a]

[a] *School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, P.R.China*

[b] *Key Laboratory of Network and Software Security Assurance of the Ministry of Education, Institute of Software, School of Electronics Engineering and Computer Science, Peking University, Beijing, P.R.China*

[c] *State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences, Beijing, P.R.China*

Abstract

Key agreement allows multi parties exchanging public information to create a common secret key that is known only to those entities over an insecure network. In recent years, several identity-based (ID-based) authenticated key agreement protocols have been proposed and most of them broken. In this paper, we formalize the security model of ID-based authenticated tripartite key agreement protocol and propose a provably secure ID-based authenticated key agreement protocol for three parties with formal security proof under the computational Diffie-Hellman assumption. Experimental results by using the AVISPA tool show that the proposed protocol is secure against various malicious attacks.

Keywords: Information security; Authentication; Key agreement; Three-party; Identity-based; Provable Security

1. Introduction

Key agreement is one of the fundamental cryptographic primitives which allows two or more parties to exchange information over an adversatively controlled insecure network and agree upon a common session key. After that, this session key may be used

[*] Corresponding author. Tel:+86 10 62765807; fax:+86 10 62758279
Email address: xionghu.uestc@gmail.com (Hu Xiong)

for later secure communication among these parties. As the basic building block for constructing secure, complex, higher-level protocols, key agreement protocol receives a lot of concern from academe and industry. To establish a session key between two parties, the first well known key agreement protocol has been proposed by Diffie and Hellman [1]. However, their original Diffie-Hellman protocol does not offer authentication between the two communicating entities and it is vulnerable against active man-in-the-middle attack. Over past years, dozens of approaches have been proposed to solve the problem in terms of improving security and efficiency of protocols [2, 3, 4, 5].

One research line of key agreement is to generalize the two-party key agreement into multi-party setting, amongst which the three-party case receives much interest. An elegant three-party key agreement protocol using bilinear pairings along with the application in broadcast networks have been proposed in Joux's pioneering work [6]. However, just like the basic Diffie-Hellman protocol, Joux's protocol is also insecure against the man-in-the-middle attack. To address this issue, Al-Riyami and Paterson [7] and Shim [8] presented several protocols to resist the man-in-the-middle attack appears in Joux's protocol independently. However, all of these protocols are presented in traditional public key infrastructure (PKI), in which each participant must obtain and verify other user's certificate before using its public key. It is generally considered to be costly to use and manage the certificates in traditional PKI.

To simplify the complicated certificate management in PKI, Shamir [9] introduced the notion of ID-based cryptography, where the public key of each user is easily computable from this user's identity. While the private key corresponding to that identity is computed and issued secretly to the user by a trusted third party called private key generator (PKG). In this way, ID-based cryptography eliminates the need of certificates. Since Zhang *et al.*'s pioneering work [10, 11], ID-based three-party authenticated key agreement protocol has rapidly emerged and been well-studied as well recently. After that, Shim and Woo [12] showed that Zhang *et al.* 's protocol was insecure against an unknown key-share attack and gave an improved protocol. But later Shim-Woo's improved protocol was found to have security weakness itself [13]. Nalla [14] then gave a more efficient construction, which was broken by Shim [15] later. An ID-based three-party authenticated key agreement protocol with k-resilience was presented by Tso *et al.* [16]. However, Lim *et al.* [17, 18] showed that the Tso *et al.*'s construction is insecure and proposed a fix. Most recently, Hölbl *et al.* [19] proposed two most efficient ID-based three-party authenticated key agreement protocols up to now. Unfortunately, Nose [20] showed that the first protocol does not offer known session key security and the second protocol is vulnerable to the insider attack. Until now, all ID-based three-party authenticated key agreement protocols are broken. A main issue with regard to the weakness of these protocols refers to the way the security analysis is conducted: the security model is not made clear, and there is not formal analysis of the claimed security properties. Therefore, provable security, which precisely defines the way an attacker interacts with the protocol in a clear mathematical model, is theoretically and also practically meaningful to guarantee the security of authentication protocols. In fact, it is challenging to design an efficient and provably-secure ID-based three-party authenticated key agreement protocols. Here, we formalize the security model of ID-based three-party authenticated key agreement protocol and propose an efficient authentication protocol based on bilinear pairing. Our protocol's overhead is lower that that of Hölbl's protocol in both computation and communication. Furthermore, our new protocol is provably secure in the random oracle model under the Computational Diffie-Hellman assumption and has been validated by the AVISPA [21, 22] formal validation tool to show its security against various malicious attacks.

The rest of this paper is organized as follows. A brief review of some basic concepts and security notions used in our scheme is described in Section 2. In Section 3, we propose a new ID-based three-party authenticated key agreement protocol with the security proof. In Section 4, the comparison between our proposed protocol and related work is conducted. Finally, the conclusions are given in Section 5.

2. Preliminaries

In this section, we will review some fundamental backgrounds required in this paper.

2.1. Mathematical Backgrounds

Let $\mathbb{G}_1$ denote an additive group of prime order q and $\mathbb{G}_2$ be a multiplicative group of the same order. Let P be a generator of $\mathbb{G}_1$, and $\hat{e}$ be a bilinear map such that $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ with the following properties:

1. Bilinearity: For all $P, Q \in \mathbb{G}_1$, and $a, b \in \mathbb{Z}_q$, $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$.
2. Non-degeneracy: $\hat{e}(P, P) \neq 1_{\mathbb{G}_2}$
3. Computability: It is efficient to compute $\hat{e}(P, Q)$ for all $P, Q \in \mathbb{G}_1$

We note that the discrete logarithm problems in $\mathbb{G}_1$ and $\mathbb{G}_2$ are hard (in a sense made precise in [23]) and refer to [6, 23, 24, 25] for a more all-around description of how these groups, pairings and other parameters should be chosen in practice for efficiency and security. Many pairing-based cryptographic protocols are based on the hardness of the following problems [6, 25].

Definition 1. *Given $\{P, Q\} \in \mathbb{G}_1$, the Discrete Logarithm (DL) Problem consists of computing $n \in \mathbb{Z}_q$ such that $P = nQ$ whenever such n exists.*

Definition 2. *Given a tuple $\{P, aP, bP\} \in \mathbb{G}_1$, for some random values $a, b \in \mathbb{Z}_q$ the Computational Diffe-Hellman (CDH) problem consists of computing the element abP.*

Definition 3. *Given $\{P, xP, yP, zP\} \in \mathbb{G}_1$ for some random values $x, y, z \in \mathbb{Z}_q$, the Bilinear Diffie-Hellman (BDH) Problem consists of computing $\hat{e}(P, P)^{xyz} \in \mathbb{G}_2$.*

Definition 4. *Given a tuple $\{P, aP, bP\} \in G$ for some random values $a, b \in \mathbb{Z}_p$, the Divisible CDH (DCDH) problem consists of computing the element $ab^{-1}P$.*

As for the relationship between CDH problem and DCDH problem, we have the following theorem [26].

Theorem 1. *DCDH problem is equivalent to CDH problem, i.e., by solving two instances of DCDH problem, one can solve an instance of CDH problem.*

2.2. Security Definitions

2.2.1. Algorithms of an ID-based tripartite authenticated key-agreement protocol

An ID-based authenticated key-agreement protocol for three parties consists of three polynomial-time algorithms: Setup, Extract and Key Agreement. These algorithms are defined as follows.

Setup: This algorithm is run by PKG. It takes as input a security parameter l and returns a master-key and a list of system parameters $params$.

Extract: This algorithm is also run by PKG. It takes as input the parameter list $params$, master-key and an entity's identity ID_i, to produce and issue the entity's private key S_{ID_i} to ID_i secretly.

Key Agreement: This is a probabilistic polynomial-time interactive algorithm which involves three entities A, B and C. The inputs are the system parameters $params$ for A, B and C, plus $\{S_{ID_A}, ID_A\}$ for A , $\{S_{ID_B}, ID_B\}$ for B and $\{S_{ID_C}, ID_C\}$ for C. Here, S_{ID_i} is the private key of i, and ID_i is the identity of i, where $i \in \{A, B, C\}$. Eventually, if the protocol does not fail, A B and C obtain a secret session key $K_{ABC} = K_{BAC} = K_{CAB} = K$.

2.2.2. Security Model

Motivated by the model of Cao *et al.* [27] and modified Bellare-Rogaway model (mBR model) [28], we present a security model for ID-based tripartite authenticated key agreement protocols. The security of our protocol Π is defined by the following game between a challenger $\mathcal{CH}$ and an adversary $\mathcal{A}$. We use the oracle $\Pi^s_{i,j,k}$ to represent the s-th instance between participants i, j and k in a session. At the beginning of the game, $\mathcal{CH}$ runs the Setup algorithm, takes as input a security parameter l to obtain the master-key and the system parameters $params$. After that, $\mathcal{CH}$ sends $params$ to $\mathcal{A}$ and keeps the master-key secret.

$\mathcal{A}$ is modelled by a probabilistic polynomial-time turing machine. All communications go through the adversary $\mathcal{A}$. Participants only respond to the queries by $\mathcal{A}$ and do not communicate directly among themselves. $\mathcal{A}$ can relay, delete, modify, interleave or delete all the message flows in the system. Note that $\mathcal{A}$ is allowed to make a polynomial number of queries, including one Test query defined as follows.

- Corrupt(ID_i): On input an identity ID_i, $\mathcal{CH}$ outputs the private key S_{ID_i} of participant i. The adversary can issue this query at any time regardless of whether ID_i is currently executing the protocol or not. This oracle captures the idea that damage due to loss of ID_i's private key should be restricted to those sessions where ID_i will participate in the future. This oracle not only represents the notion of forward secrecy but also captures a variety of impersonation attacks. In fact, the corruption of a principle ID_i should not lead the adversary to have the ability to impersonate any principle other than ID_i, because such ability would endanger even those sessions where ID_i is not invited to participate.

- Send($\Pi^n_{j,k,i}$,M_1,M_2): $\mathcal{A}$ can send a message M_1 and M_2 of her choice to an oracle, say $\Pi^n_{j,k,i}$, in which case participant i assumes that the message has been sent by participants j and k respectively. $\mathcal{A}$ may also make a special Send query with $M_1 = \lambda_1$ and $M_2 = \lambda_2$ to the oracle $\Pi^n_{j,k,i}$, which demonstrates i to initiate a protocol run with j and k. An oracle is an initiator if the first messages it has received are λ_1 and λ_2. If an oracle does not receive messages λ_1 and λ_2 as its first message, then it is a responder oracle. This query models an active attack against the oracle $\Pi^n_{j,k,i}$.

- Reveal($\Pi^n_{j,k,i}$): $\mathcal{A}$ can ask a particular oracle to reveal the session key (if any) it currently holds to $\mathcal{A}$. The output of this query is either $\perp$ if the instance does not accept a session key or the real session key if it accepts a key. This oracle captures the idea that exposure of some session keys should not affect the security of other session keys.

- Test($\Pi^T_{I,J,K}$): At some point, $\mathcal{A}$ has to make a Test query to a *fresh* oracle $\Pi^T_{I,J,K}$ (see Definition 5). To answer the query, $\mathcal{CH}$ flips a fair coin $b \in \{0, 1\}$, and returns the session key held by $\Pi^T_{I,J,K}$ if $b = 0$, or a random sample from the distribution of the session key if $b = 1$.

Definition 5. *(Fresh oracle). Here,* $\Pi^s_{i,j,k}$ *is* fresh *if (1)* $\Pi^s_{i,j,k}$ *has accepted the request to establish a session key; (2)* $\Pi^s_{i,j,k}$ *has not been revealed; (3) there is no matching conversation*[1] *of oracle* $\Pi^s_{i,j,k}$ *has been revealed; (4) participants* $j, k \neq i$ *have not been corrupted.*

[1] Let the session ID be the concatenation of the messages in a session. Two oracles $\Pi^s_{i,j,k}$ and $\Pi^t_{j,k,i}$ are said to have a matching conversation with each other if they have the same session ID.

After a Test query, $\mathcal{A}$ can continue to query the oracles except that it cannot make a Reveal query to the test oracle $\Pi^T_{I,J,K}$ or to $\Pi^S_{J,K,I}$ who has a matching conversation with $\Pi^T_{I,J,K}$ (if it exists). Finally, $\mathcal{A}$ outputs its guess b' for b. $\mathcal{A}$'s advantage $Advantage^{\mathcal{A}}(l)$ is defined as the probability $b = b'$.

The security of ID-based authenticated key agreement protocol protocol for three parties can be defined using the concept of $\mathcal{A}$'s advantage as follows:

Definition 6. *An ID-based tripartite authenticated key agreement protocol is said to be secure if:*

1. *In the presence of a benign adversary, two oracles running the protocol both accept holding the same session key, and the session key is distributed uniformly at random on $\{0,1\}^l$; and*
2. *For any adversary $\mathcal{A}$, $Advantage^{\mathcal{A}}(l)$ is negligible.*

Finally, we introduce Computational No Reveal-mBR game (cNR-mBR game) [29] to simplify our security model[2]. The simplified game is identical to the original security game except that $\mathcal{A}$ is not allowed to ask Reveal queries and $\mathcal{A}$ no longer makes a Test query. Instead, an adversary must choose a fresh oracle $\Pi^s_{i,j,k}$ at the end of the game, and it must compute the session key instead of deciding between a session key and a random value to win the game. In such a game, the security of the protocol is defined as the probability that $\mathcal{A}$ outputs a session key K such that $K = K^s_{\Pi_{i,j,k}}$.

Similar to [27], since $\mathcal{A}$ is formalized in a way that it can perform all kinds of known attacks in the real world, then a protocol provides desirable security attributes including *known session key security*, *forward secrecy*, *key compromise impersonation resilience* and *unknown key-share resilience* when it satisfies Definition 6.

3. Proposed ID-based authenticated key agreement protocol

In this section we propose an ID-based authenticated key agreement protocol using pairings for the three-party setting. Our protocol is based on ideas by Cao *et al.* [27]. We describe our protocols into details and give a security resp. efficiency analysis.

[2] For more details of conciseness and preciseness of modular approach we refer the reader to [29, 27].

3.1. Proposed protocol

Our proposed protocol employs ideas by Cao *et al.* [27] and Hölbl *et al.* [19]. Similarly to other ID-based authenticated key agreement protocols, the proposed one requires a private key generator (PKG) and consist of three phases: system setup, private key extraction and key agreement phase.

Setup: Given a security parameter $k \in \mathbb{Z}$, the algorithm works as follows:

1. Run the parameter generator on input k to generate a prime q, two groups $\mathbb{G}_1$, $\mathbb{G}_2$ of prime order q, a generators P of $\mathbb{G}_1$ and an admissible pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$.
2. Select a master-key $x \in_R \mathbb{Z}_q^*$, and compute $P_{pub} = xP$.
3. Choose cryptographic hash functions $H_1 : \{0,1\}^* \times \mathbb{G}_1 \to \mathbb{Z}_q^*$ and $H_2 : \{0,1\}^* \times \{0,1\}^* \times \{0,1\}^* \times \mathbb{G}_1 \times \mathbb{G}_1 \times \mathbb{G}_1 \times \mathbb{G}_1 \times \mathbb{G}_1 \times \mathbb{G}_1 \times \mathbb{G}_1 \times \mathbb{G}_1 \to \{0,1\}^k$. Finally the PKG's master-key x is kept secret and the system parameters $\{q, \mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, P_{pub}, H_1, H_2\}$ are published.

Private Key Extraction: Given a user's identity $ID_U \in \{0,1\}^*$, PKG first chooses at random $r_U \in_R \mathbb{Z}_q^*$, computes $R_U = r_U P$, $h = H_1(ID_U \| R_U)$ and $s_U = (r_U + hx)^{-1}$. It then sets this user's private key (s_U, R_U) and transmits it to user ID_i secretly.

It is easy to see that user ID_U can validate her long-term key by checking whether the equation $s_U(R_U + H_1(ID_U \| R_U)P_{pub}) = P$ holds. The long-term key is valid if the equation holds and vice versa.

Key Agreement: The proposed protocols employs ideas by Cao *et al.* [27] and Hölbl *et al.* [19]. The former protocol is extended to the three-party settings and allows the establishment of a session key between three participating entities. The message flows and computations of a protocol run are described below.

1. A, B, C: choose $a, b, c \in \mathbb{Z}_q^*$.
2. $A \to B, C : \{ID_A, R_A\}$

   $B \to A : \{ID_B, R_B, T_{BA} = b(R_A + H_1(ID_A \| R_A)P_{pub})\}$

   $C \to A : \{ID_C, R_C, T_{CA} = c(R_A + H_1(ID_A \| R_A)P_{pub})\}$

   $A \to B : T_{AB} = a(R_B + H_1(ID_B \| R_B)P_{pub})$

$A \to C : T_{AC} = a(R_C + H_1(ID_C \| R_C)P_{pub})$

$B \to C : \{ID_B, R_B\}$

$C \to B : \{ID_C, R_C, T_{CB} = c(R_B + H_1(ID_B \| R_B)P_{pub})\}$,

$B \to C : T_{BC} = b(R_C + H_1(ID_C \| R_C)P_{pub})$

3. A computes:

$K^1_{ABC} = aP + s_A T_{BA} + s_A T_{CA} = aP + bP + cP = (a+b+c)P$

$K^2_{ABC} = \hat{e}(s_A T_{BA}, s_A T_{CA})^a = \hat{e}(bP, cP)^a = \hat{e}(P,P)^{abc}$

B computes:

$K^1_{ABC} = bP + s_B T_{AB} + s_B T_{CB} = bP + aP + cP = (a+b+c)P$

$K^2_{ABC} = \hat{e}(s_B T_{AB}, s_B T_{CB})^b = \hat{e}(aP, cP)^b = \hat{e}(P,P)^{abc}$

C computes:

$K^1_{ABC} = cP + s_C T_{AC} + s_C T_{BC} = cP + aP + bP = (a+b+c)P$

$K^2_{ABC} = \hat{e}(s_C T_{AC}, s_C T_{BC})^c = \hat{e}(aP, bP)^c = \hat{e}(P,P)^{abc}$

After the protocol has finished, all three entities share the session key which is computed as $K = H_2(ID_A \| ID_B \| ID_C \| T_{AB} \| T_{AC} \| T_{BA} \| T_{BC} \| T_{CA} \| T_{CB} \| K^1_{ABC} \| K^2_{ABC})$.

3.2. Security proof

To employ the modular approach [29], we need first transform our protocol Π into a corresponding protocol π which is identical to Π except that Π produces a hashed session key while π utilizes the input string of the hash function as the session key. To this end, we turn our new protocol Π into a related protocol π, which is similar to the former except that π uses the string $(ID_A \| ID_B \| ID_C \| T_{AB} \| T_{AC} \| T_{BA} \| T_{BC} \| T_{CA} \| T_{CB} \| K^1_{ABC} \| K^2_{ABC})$ as the session key while Π uses $H_2(ID_A \| ID_B \| ID_C \| T_{AB} \| T_{AC} \| T_{BA} \| T_{BC} \| T_{CA} \| T_{CB} \| K^1_{ABC}$ $\| K^2_{ABC})$. Then we prove the cNR-mBR security of π.

Lemma 1. *Suppose that if for protocol π there is an adversary $\mathcal{A}$ who can win the cNR-mBR game with advantage at least ε, then the CDH problem can be solved with non-negligible advantage by an algorithm $\mathcal{CH}$.*

Proof. Suppose $\mathcal{CH}$ is given an instance $(aP, bP) \in G$ of the CDH problem, and is tasked to compute $cP \in G$ with $c = ab \bmod p$. To do this, $\mathcal{CH}$ simulates a challenger with $\mathcal{A}_1$. $\mathcal{CH}$ stipulates the hash function H_1 and maintains an H_1-list which is initialized empty. The number of participants in the game is denoted by $n_p(k)$ and the number of sessions each participant may be involved in is denoted by $n_s(k)$. The private key for the i-th participant ID_i is (s_i, R_i) and ID_i is the corresponding identifier. $\mathcal{CH}$ generates ID_i's private key as follows:

$\mathcal{CH}$ first chooses $I \in \{1, \cdots, n_p(k)\}$ randomly, then chooses $R_I \in_R G$ and sets $\{\perp, R_I\}$ as ID_I's private key. The system public key can be denoted as $P_{pub} = H_1(ID_I, R_I)^{-1}$ $(bP - R_I)$ which implicitly means that $s_I^{-1}P = bP$. For ID_i with $i \in \{1, \cdots, n_p(k)\}$ and $i \neq I$, $\mathcal{CH}$ sets the private key by first choosing at random $(s_i, h_i) \in_R \mathbb{Z}_p^*$. Then $\mathcal{CH}$ computes $R_i = s_i^{-1}P - h_i P_{pub}$ and sets (s_i, R_i) as ID_i's private key. After that, $\mathcal{CH}$ adds $\{ID_i, R_i, s_i, h_i\}$ to the H_1-list for $i \in \{1, \cdots, n_p(k)\}$.

Then $\mathcal{CH}$ picks at random $J, K \in \{1, \cdots, n_p(k)\} \neq I$, $v \in \{1, \cdots, n_s(k)\}$, and $\mathcal{CH}$ starts $\mathcal{A}$ by answering $\mathcal{A}$'s queries as follows:

$H_1(ID_i, R_i)$ query: If the tuple $\{ID_i, R_i, s_i, h_i\}$ is already in the H_1-list, $\mathcal{CH}$ responds with h_i, otherwise, $\mathcal{CH}$ chooses $h_i \in_R \mathbb{Z}_p^*$, adds $\{ID_i, R_i, s_i, h_i\}$ to the H_1-list and returns h_i to $\mathcal{A}$.

$\mathsf{Corrupt}(ID_i)$: Whenever $\mathcal{CH}$ receives this query, if $ID_i = ID_I$, $\mathcal{CH}$ aborts; else, $\mathcal{CH}$ searches for a tuple $\{ID_i, R_i, s_i, h_i\}$ in H_1-list which is indexed by ID_i and returns (R_i, s_i) as the answer.

$\mathsf{Send}(\Pi^s_{i,j,k}, M_1, M_2)$: If $\Pi^s_{i,j,k} \neq \Pi^v_{J,K,I}$, then $\mathcal{CH}$ acts according to the protocol specification. Otherwise, $\mathcal{CH}$ responds with the tuples (ID_j, upk_j, R_j, aP).

The probability that $\mathcal{A}$ chooses $\Pi^v_{J,K,I}$ as the Test oracle and that $ID_i = ID_I$, $ID_j = ID_J$ and $ID_k = ID_K$ is $\frac{1}{n_s(k)n_p(k)(n_p(k)-1)}$. In this case, $\mathcal{A}$ would not have corrupted ID_I, and so $\mathcal{CH}$ would not have aborted. If $\mathcal{A}$ can win in such game, then at the end of this game, $\mathcal{A}$ will output its guess of the session key of the form $\{0,1\}^* \times \{0,1\}^* \times \{0,1\}^* \times \mathbb{G}_1^6 \times A \times B$, and $\mathcal{CH}$ can output $A - s_J M_1 - s_K M_2$ where M_1 and M_2 are the input messages of $\mathsf{Send}(\Pi^s_{I,J,K}, M_1, M_2)$ query. Thus $\mathcal{CH}$ can solve the DCDH problem with non-negligible probability $c\frac{1}{n_s(k)n_p(k)(n_p(k)-1)}$ within $t(k)$ where c is a constant. Then according to Theorem 1, the CDH problem can be solved with advantage at least $(c\frac{1}{n_s(k)n_p(k)(n_p(k)-1)})^2$.

Theorem 2. *[29] Suppose that a key agreement protocol Π produces a hashed session key on completion of the protocol (via hash function H) and that Π has strong partnering*[3]*. If the cNR-mBR security of the corresponding π is probabilistic polynomial time reducible to the hardness of the computational problem of some relation f, and the session string decisional problem for Π is polynomial time reducible to the decisional problem of f, then the mBR security of Π is probabilistic polynomial time reducible to the hardness of the Gap problem of f, assuming that H is a random oracle.*

Theorem 3. *Our protocol Π is secure in the random oracle model assuming the hardness of Gap Diffie-Hellman problem.*

Proof. The theorem follows directly from Theorem 2 and Lemma 1. Thus, the protocol provides known session key security, key compromise impersonation resilience and unknown key share resilience, which is satisfied even in face of Kaliski's UKS attack, as shown in [31].

Theorem 4. *Our protocol has the perfect forward secrecy property if the BDH problem is hard.*

Proof. Suppose that A, B and C established a session key K using our protocol, and later, their private key (s_A, R_A), (s_B, R_B) and (s_C, R_C) were compromised. Let a, b and c be the secret random numbers chosen by A, B and C, respectively, during the establishment of their session key. It is easy to see that, to compute the established session key K, the adversary who owns (s_A, R_A), (s_B, R_B) and (s_C, R_C), can compute $s_B T_{AB} = s_C T_{AC} = aP$, $s_A T_{BA} = s_C T_{BC} = bP$ and $s_A T_{CA} = s_B T_{CB} = cP$ for unknown a, b and c. However, to compute the value of $\hat{e}(P, P)^{abc}$ without the knowledge of either a, b and c, the adversary must have the ability to solve the BDH problem. Under the BDH assumption, this probability is negligible. Hence, our protocol has the perfect forward secrecy property.

3.3. Formal analysis using AVISPA

Besides the above analysis, we have also provided a formal analysis of our proposed protocol by Automated Validation of Internet Security Protocols and Applications (AVISPAs) [21, 22] to validate various security properties that our protocol has been designed.

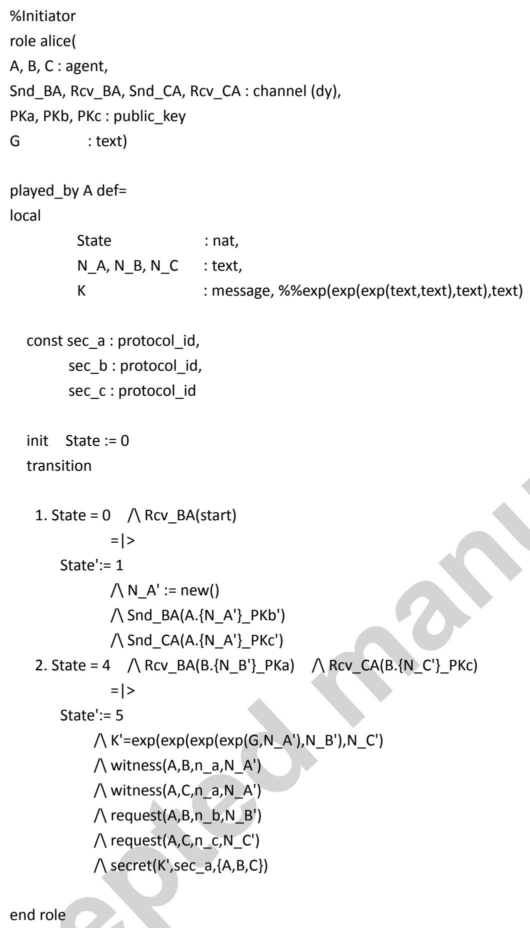
[3] If there exists an adversary $\mathcal{A}$, which when attacking Π in an mBR game with non-negligible probability in the security parameter l, can make any two oracles $\Pi^u_{a,b,c}$ and $\Pi^v_{b,c,a}$ accept holding the same session key when they are not partners, then we say that Π has weak partnering. If Π does not have weak partnering, then we say that Π has strong partnering.

The AVISPA, which is a push-button tool for the analysis of Internet security-sensitive protocols, provides High-Level Protocol Specification Language (HLPSL) for the description of security protocols and specifying their security properties. The current version of AVISPA is featured with four back-ends that implement a variety of automatic analysis techniques: On-the-fly Model-Checker (OFMC), Constraint- Logic-based Attack Searcher (CL-AtSe), SAT-based Model-Checker (SATMC), and Tree Automata based on Automatic Approximations for the Analysis of Security Protocols (TA4SP). The Dolev-Yao attacker [32], which can overhear, intercept messages, inject new messages or modify messages in transit, is also implemented in AVISPA. Thus, it is appropriate for the analysis of large-scale security protocols.

Our proposed protocol has been translated into HLSPL. In order to describe our protocol clearly, each role encrypt their contribution factor using other users' public key PKu instead of the R_U, ID_U and P_{pub}, which are used to blind their contribution factor. In this paper, we only present one of the roles initiator shown in Fig. 1 as an example. Once the HLPSL specification has been debugged, it was checked automatically for attack detection using the AVISPA verification tools. No revealed attacks were found, and the security goals are achieved.

4. Comparison with competitive protocols

In this section we compare the efficiency and security properties of the proposed protocols with Hölbl *et al.*'s protocol [19] (the known most efficient ID-based authenticated three-party key agreement protocol) in terms of efficiency and security. In the comparison of computation efficiency, Pa stands for pairing operation, S for scalar multiplication in $\mathbb{G}_1$, and E for exponentiation operation. Bandwidth is measured by the maximal length of a single message in transmission (auxiliary message such as entity identifier is not counted), and P stands for a point in $\mathbb{G}_1$ and p for an element in $\mathbb{Z}_q^*$. We compare different protocols' security strength in terms of Known-key secrecy (K-KS), Perfect forward secrecy (PFS), Key-compromise impersonation (K-CI), Unknown key-share (UK-S), No key control (KC), and Provable Security (PS). The comparison results are shown in Table



```
%Initiator
role alice(
A, B, C : agent,
Snd_BA, Rcv_BA, Snd_CA, Rcv_CA : channel (dy),
PKa, PKb, PKc : public_key
G                : text)

played_by A def=
local
            State                   : nat,
            N_A, N_B, N_C      : text,
            K                          : message, %%exp(exp(exp(text,text),text),text)

   const sec_a : protocol_id,
           sec_b : protocol_id,
           sec_c : protocol_id

   init    State := 0
   transition

     1. State = 0    /\ Rcv_BA(start)
                  =|>
         State':= 1
                  /\ N_A' := new()
                  /\ Snd_BA(A.{N_A'}_PKb')
                  /\ Snd_CA(A.{N_A'}_PKc')
     2. State = 4    /\ Rcv_BA(B.{N_B'}_PKa)    /\ Rcv_CA(B.{N_C'}_PKc)
                  =|>
         State':= 5
               /\ K'=exp(exp(exp(exp(G,N_A'),N_B'),N_C')
               /\ witness(A,B,n_a,N_A')
               /\ witness(A,C,n_a,N_A')
               /\ request(A,B,n_b,N_B')
               /\ request(A,C,n_c,N_C')
               /\ secret(K',sec_a,{A,B,C})

end role
```

Figure 1: Role of Initiator

1 where ✓ means "satisfy" and × "not satisfy".

Table 1: Computational effort per user

Protocol	computation	bandwidth	K-KS	PFS	K-CI	UK-S	KC	PS
Hölbl-I [19][a]	$2Pa + 2S + E$	P	×	×	✓	✓	✓	×
Hölbl-II [19][b]	$3Pa + 5S + E$	$3P$	✓	✓	✓	×	✓	×
Our protocol	$Pa + 5S + E$	$2P$	✓	✓	✓	✓	✓	✓

a. K-KS of Hölbl-I [19] is broken due to Nose's attack [20]

b. UK-S of Hölbl-II [19] is broken due to Nose's attack [20]

Given the relative computation cost of pairing and exponential over $\mathbb{F}_{p^\alpha}$ are approximately 20 times and 3 times higher than that of the scalar multiplication [27], we can estimate that the computation cost of our protocol is 98.6% of Hölbl *et al.*'s first protocol which provides weaker security and 61.2% of their second protocol. The bandwidth load in our protocols is 66.7% of that in Hölbl *et al.*'s second protocol while is two times more than that of their first protocol. Considering the improvements in both security and computation, this cost seems reasonable. What's more, when R_U is stored, our protocol's communication load is as low as that of Hölbl *et al.*'s first protocol.

5. Conclusion

The notion and security models of ID-based authenticated three-party key agreement protocol are formalized. The models capture the essence of the possible adversaries in the notion of authenticated key agreement. A concrete construction of ID-based authenticated three-party key agreement protocol from the bilinear maps is presented. Our protocol is computationally efficient that it only needs one pairing computation for each participant. Furthermore, the security of our protocol is proved in the random oracle model under the computational Diffie-Hellman assumption.

6. Acknowledgement

This work is supported by the National Natural Science Foundation of China under Grant No. 61003230 and Open fund of State Key Laboratory of Information Security.

[1] Diffie W, Hellman M. New Directions in Cryptography. IEEE Transactions on Information Theory 1976; 22(6): 644-54.

[2] Dutta R, Barua R. Overview of Key Agreement Protocols. Cryptology ePrint Archive: Report 2005/289; 2005.

[3] Menezes A, Oorschot PC, Vanstone SA. Handbook of applied cryptography. USA: CRC Press; 1997.

[4] Sood SK, Sarje AK, Singh K. A secure dynamic identity based authentication protocol for multi-server architecture. Journal of Network and Computer Applications 2011; 34(2): 609-18.

[5] Li CT, Hwang MS. An efficient biometrics-based remote user authentication scheme using smart cards. Journal of Network and Computer Applications 2010; 33(1): 1-5.

[6] Joux A. A one round protocol for tripartite Diffie-Hellman. In: 4th International symposium on algorithmic number theory. Lecture notes in computer science, vol. 1838; 2000. p. 385-94.

[7] Al-Riyami S, Paterson K. Tripartite Authenticated Key Agreement Protocols from Pairings. In: 9th IMA International Conference of Cryptography and Coding'03, Lecture notes in computer science, vol. 2898; 2003. p. 332-359.

[8] Shim K. Efficient one-round tripartite authenticated key agreement protocol from Weil pairing. Electronics Letters 2003; 39(2): 208-09.

[9] Shamir A. Identity-Based Cryptosystems and Signature Schemes. In: Advances in cryptology–CRYPTO 1984, Lecture notes in computer science, vol. 196; 1984. p.47-53.

[10] Zhang F, Liu S, Kim K. ID-Based One Round Authenticated Tripartite Key Agreement Protocol with Pairings. Cryptology ePrint Archive: Report 2002/122; 2002.

[11] Liu S, Zhang F, Chen F. ID-based tripartite key agreement protocol with pairing. In: IEEE International Symposium on Information Theory–ISIT'03, Chicago, USA, 2003. p. 136-43.

[12] Shim K, Woo S. Weakness in ID-based one round authenticated tripartite multiple-key agreement protocol with pairings. Applied Mathematics and Computation 2005; 166(3): 523-530.

[13] Chou JS, Lin CH, Chiu CH. Weakness of Shim's new ID-based tripartite multiple-key agreement protocol. WSEAS Transactions on Information Science and Applications 2006; 3(7): 1407-10.

[14] Nalla D. ID-based tripartite key agreement with signatures. Cryptology ePrint Archive: Report 2003/144; 2003.

[15] Shim K. Cryptanalysis of ID-based tripartite authenticated key agreement protocols. Cryptology ePrint Archive: Report 2003/115; 2003.

[16] Tso R, Okamoto T, Takagi T, Okamoto E. An ID-based Non-Interactive Tripartite Key Agreement Protocol with K-Resilience. In: Proceedings of the 2005 Symposium on Cryptography and Information Security–SCIS 2005, Maiko Kobe, Japan, 2005. p. 1-6.

[17] Lim MH, Lee S, Moon S. Cryptanalysis of Tso et al.'s ID-Based Tripartite Authenticated Key Agreement Protocol. In: 3rd International Conference on Information Systems Security–ICISS '07, Lecture notes in computer science, vol. 4812; 2007. p. 64-76.

[18] Lim MH, Lee S. An Improved One-Round ID-Based Tripartite Authenticated Key Areement Protocol. Cryptology ePrint Archive: Report 2007/189; 2007.

[19] Hölbl M, Welzer T, Brumen B. Two proposed identity-based three-party authenticated key agreement protocols from pairings. Computers & Security 2010; 29(2): 244-52.

[20] Nose P. Security weaknesses of authenticated key agreement protocols, Information Processing Letters 2011; 111(14): 687-96.

[21] Clarke EM, Grumberg O, Peled DA. Model checking. Cambridge: MIT Press; 1999.

[22] AVISPA v1.1 User Manual, 2006. Available: http://www.avispa-project.org/.

[23] Boneh D, Franklin M. Identity-based encryption from the Weil pairing. In: Advances in cryptology–CRYPTO'02, Lecture notes in computer science, vol. 2139; 2001, p. 213-29.

[24] Barreto PSLM, Kim KY, Lynn B. Efficient algorithms for pairing-based cryptosystems. In: Advances in cryptology–CRYPTO 2002, Lecture notes in computer science, vol. 2442; 2002, p. 354-68.

[25] Boneh D, Lynn B, Shacham H. Short signatures from the Weil pairing. Journal of Cryptology 2004; 17(4): 297-319.

[26] Bao F, Deng R, Zhu H. Variations of Diffie-Hellman Problem. In: 5th International Conference of Information and Communications Security–ICICS 2003, Lecture notes in computer science, vol. 2836; 2003, p. 301-12.

[27] Cao X, Kou W, Du X. A pairing-free identity-based authenticated key agreement protocol with minimal message exchanges. Information Sciences 2010; 180(15): 2895-903.

[28] Bellare M, Rogaway P. Random oracles are practical: a paradigm for designing efficient protocols. In: Proceedings of the 1st ACM Conference Computer and Communications Security–CCS'93, Fairfax, Virginia, USA, 1993. p. 62-73.

[29] Kudla C, Paterson KG. Modular security proofs for key agreement protocols. In: Advances in cryptology–ASIACRYPT 2005, Lecture notes in computer science, vol. 3788; 2005, p. 549-65.

[30] Nalla D, Reddy KC. ID-based tripartite authenticated key agreement protocols frompairings. Cryptology ePrint Archive: Report 2003/004; 2003.

[31] Kaliski Jr BS. An unknown key-share attack on the MQV key agreement protocol. ACM Transactions on Information and System Security 2001; 4(3): 275-88.

[32] Dolev D, Yao A. On the security of public key protocols. IEEE Transactions on Information Theory 1983; 29(2): 198-208.