Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 367264, 10 pages
http://dx.doi.org/10.1155/2014/367264

Research Article

A Provably Secure Revocable ID-Based Authenticated Group Key Exchange Protocol with Identifying Malicious Participants

Tsu-Yang Wu,[1,2] Tung-Tso Tsai,[3] and Yuh-Min Tseng[3]

[1] *Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen 518055, China*
[2] *Shenzhen Key Laboratory of Internet Information Collaboration, Shenzhen 518055, China*
[3] *Department of Mathematics, National Changhua University of Education, Jin-De Campus, Changhua City 500, Taiwan*

Correspondence should be addressed to Tsu-Yang Wu; wutsuyang@gmail.com

Received 12 March 2014; Accepted 16 May 2014; Published 1 June 2014

Academic Editor: Fei Yu

The existence of malicious participants is a major threat for authenticated group key exchange (AGKE) protocols. Typically, there are two detecting ways (passive and active) to resist malicious participants in AGKE protocols. In 2012, the revocable identity- (ID-) based public key system (R-IDPKS) was proposed to solve the revocation problem in the ID-based public key system (IDPKS). Afterwards, based on the R-IDPKS, Wu et al. proposed a revocable ID-based AGKE (RID-AGKE) protocol, which adopted a passive detecting way to resist malicious participants. However, it needs three rounds and cannot identify malicious participants. In this paper, we fuse a noninteractive confirmed computation technique to propose the first two-round RID-AGKE protocol with identifying malicious participants, which is an active detecting way. We demonstrate that our protocol is a provably secure AGKE protocol with forward secrecy and can identify malicious participants. When compared with the recently proposed ID/RID-AGKE protocols, our protocol possesses better performance and more robust security properties.

1. Introduction

In the past, group-oriented applications, such as collaboration works and teleconference, were popularly and widely used in the Internet. Authenticated group key exchange (AGKE) protocol [1] is a cryptographic primitive which provides secure group communications for users in cooperative and distributed applications. During executing the protocol, group participants not only cooperatively generate a common key which is used to encrypt the transmitted messages but also authenticate the participants' identities.

The existence of malicious participants is a major threat for AGKE protocols. The goal of malicious participants is to disturb the establishing of common keys. Hence, how to resist malicious participants in AGKE protocols becomes a critical research. Typically, there are two detecting ways to resist malicious participants. (I) Passive detection [2–4]: it involves an explicit key confirmation approach in AGKE protocols. The resulted protocols only detect the existence of malicious participants and an additional round is required. (II) Active detection [5, 6]: it adopts a noninteractive confirmed computation technique into AGKE protocols. The resulted protocols can identify the identities of malicious participants without additional round. However, the computational cost of active detection is time-consuming than the one of passive detection.

Quite recently, the revocable identity- (ID-) based public key system (R-IDPKS) was proposed to solve the revocation problem of users in the ID-based public key system (IDPKS). The concept of IDPKS was introduced by Shamir [7] in 1984 and was practiced by Bonch and Franklin [8] in 2001. Indeed, they [8] had suggested a solution that the private key generator (PKG) renews these nonrevoked users' private keys periodically to answer the revocation problem in the IDPKS. The approach can be used to revoke the compromised or misbehaving users. Nevertheless, the heavy workload arose from the PKG for renewing users' private keys periodically.

In 2008, Boldyreva et al. [9] proposed a revocable ID-based encryption (RIBE) scheme by using binary tree. Their scheme can reduce the PKG's workload mentioned in the

Boneh-Franklin solution [8]. However, this scheme is based on a weak security model called the relaxed selective-ID model [10]. In 2009, Libert and Vergnaud [11] relied on Boldyreva et al.'s RIBE to present a secure RIBE scheme under an adaptive-ID model. Recently, Seo and Emura [12] demonstrated Boldyreva et al.'s scheme [9] is vulnerable to decryption key exposure and then proposed a provably secure tree based RIBE scheme. Subsequently, Seo and Emura [13] presented a hierarchical RIBE scheme to solve the open problem mentioned in [11].

In 2011, Tseng and Tsai [14] proposed a practical RIBE scheme over a public channel. The key construction of the Tseng-Tsai scheme is different from the previous schemes [9, 11–13]. In [14], each user's private key consists of a fixed initial private key and an update key, where the update key is renewed along with the current period. For an honest (nonrevoked) user, the PKG periodically issues new update key and sends it to the user via a public channel. Upon receiving the new updating key, the user can renew her/his private key by herself/himself. To revoke a malicious user, the PKG only stops issuing the new update key in current period. Thus, the user cannot compute the newest private key. In other words, she/he cannot execute any cryptographic behaviors in later periods. Later on, several revocable ID-based cryptographic schemes based on the Tseng-Tsai R-IDPKS [14] were presented such as encryption [15], signature [16, 17], authenticated group key exchange (AGKE) [4], and signcryption [18].

In 2012, Wu et al. [4] proposed the first provably secure revocable ID-based AGKE (RID-AGKE) protocol. Their protocol adopted a passive detecting way to resist malicious participants. However, it requires three rounds and cannot identify the identities of malicious participants. In this paper, we fuse the key construction of Tseng-Tsai R-IDPKS [14] and a noninteractive confirmed computation technique [6] to present a two-round RID-AGKE protocol with identifying malicious participants. In our protocol, each group participant can confirm whether the broadcast values are correctly computed by other participants. Based on the detecting approach, our protocol can easily identify the participants who maliciously broadcast the incorrect values to disturb the common key establishing. The framework and security notions for RID-AGKE protocols are defined to formalize possible threats and attacks. We demonstrate the security of our protocol in the random oracle model [19] and under two mathematical assumptions (the computational Diffie-Hellman and the decisional bilinear Diffie-Hellman). Finally, we make the comparisons between our protocol and the recently proposed ID/RID-AGKE protocols to show the advantages of the proposed protocol.

The rest of this paper is organized as follows. We briefly review the concepts of bilinear pairings and related mathematical problems in Section 2. The security model and notions of RID-AGKE are presented in Section 3. We propose a concrete RID-AGKE protocol in Section 4. Security analysis of the proposed RID-AGKE protocol is demonstrated in Section 5. We make the performance analysis and comparisons in Section 6. Conclusions are drawn in Section 7.

2. Preliminaries

In this section, we briefly review the properties of bilinear pairings and related mathematical problems. For the details, a reader can refer to [8, 20, 21] for full descriptions.

2.1. Bilinear Pairings. Let G_1 and G_2 be two groups of a large prime order q, where G_1 is an additive cyclic group and G_2 is a multiplicative cyclic group. A bilinear pairing e is a map defined by $e : G_1 \times G_1 \rightarrow G_2$ and satisfies the following three conditions.

(1) *Bilinearity*: for all $P, Q \in G_1$ and $a, b \in Z_q$, $e(aP, bQ) = e(P, Q)^{ab}$.

(2) *Nondegeneracy*: there exist $P, Q \in G_1$ such that $e(P, Q) \neq 1$.

(3) *Computability*: for all $P, Q \in G_1$, there exists an algorithm to compute $e(P, Q)$.

2.2. Mathematical Hard Problems and Assumptions. Here, we present two mathematical hard problems and define the corresponding assumptions as follows.

(1) *Computational Diffie-Hellman (CDH) problem*: given $P, aP, bP \in G_1$ for some $a, b \in Z_q{}^*$, the CDH problem is to compute $abP \in G_1$.

(2) *Decisional bilinear Diffie-Hellman (DBDH) problem*: given $P, aP, bP, cP, dP \in G_1$ for some $a, b, c, d \in Z_q{}^*$, the DBDH problem is to distinguish $(P, aP, bP, cP, dP, e(P, P)^{abc}$ from $(P, aP, bP, cP, dP, e(P, P)^d)$.

Definition 1 (CDH assumption). Given $P, aP, bP \in G_1$ for some $a, b \in_R Z_q{}^*$, there does not exist a probabilistic polynomial-time algorithm A with a nonnegligible probability to compute $abP \in G_1$. The advantage of A within running time t is defined as $Adv_{CDH}(t) = \Pr[A(P, aP, bP) = abP \mid P, aP, bP \in G_1]$.

Definition 2 (DBDH assumption). Given $P, aP, bP, cP, dP \in G_1$ for some $a, b, c, d \in_R Z_q{}^*$, there does not exist a probabilistic polynomial-time algorithm A with nonnegligible probability to distinguish $(P, aP, bP, cP, dP, e(P, P)^{abc})$ from $(P, aP, bP, cP, dP, e(P, P)^d)$. The advantage of A within running time t is defined as $Adv_{DBDH}(t) = \Pr[A(e(P, P)^{abc}, e(P, P)^d) = 1 \mid P, aP, bP, cP, dP \in G_1]$.

3. Model and Notions

In this section, we define the model and notions for RID-AGKE protocol. Note that some of the following definitions and notations are referred to in [4, 6, 22–24].

Initialization. The initialization of RID-AGKE protocol has three algorithms.

(1) *Setup Algorithm.* This algorithm is a probabilistic algorithm which takes as input a security parameter k and a total

number z of periods. It returns a system private key s and public parameters *param*. Note that the whole life time of the system is divide into distinct periods $1, 2, \ldots, z$. Here, *param* is made public.

(2) *Initial Key Extract Algorithm.* This algorithm is a deterministic algorithm which takes as input the system private key s and a participant's identity ID. It returns the participant's initial private key DID.

(3) *Key Update Algorithm.* This algorithm is a deterministic algorithm which takes as input the system private key s, a participant's identity ID, and a period index j, where $1 \leq j \leq z$. It returns the participant's update key TID_j.

Here, note that the participant's private key for period j is defined by $DID_j = DID + TID_j$.

Related Notions. For simplicity, there is a fixed set $G = \{U_1, U_2, \ldots, U_n\}$ with polynomial size of potential participants. Assume that each participant U_i has a unique identity $ID_i \in \{0,1\}^*$. Any subset of G may run a RID-AGKE protocol many times (possibly concurrently) in some period index j to establish a group session key, where $1 \leq j \leq z$ and z is a total number of periods. Note that the set of participants' identities, $\mathbf{ID} = \{ID_1, ID_2, \ldots, ID_n\}$ is known by all participants (including adversary).

An instance t of participant U in period j is denoted by $\Pi_U^{j(t)}$, where t is a positive integer. Each instance $\Pi_U^{j(t)}$ has associated with seven variables as follows.

(i) $state_U^{j(t)}$: it presents the current state of instance $\Pi_U^{j(t)}$.

(ii) $acc_U^{j(t)}$ and $term_U^{j(t)}$: they take Boolean values to demonstrate whether $\Pi_U^{j(t)}$ *has accepted* or terminated. Informally, we say that an instance *has accepted* meaning that it does not detect any incorrect behavior. An instance is called terminating if it has sent and received messages. Note that a terminated instance may also possibly accept.

(iii) $used_U^{j(t)}$: it indicates whether $\Pi_U^{j(t)}$ is used in a RID-AGKE protocol.

(iv) $pid_U^{j(t)}$: the partner ID of instance $\Pi_U^{j(t)}$ is a set which contains the identities of participants in the group with whom $\Pi_U^{j(t)}$ wants to establish a group session key (including U itself).

(v) $sid_U^{j(t)}$: the session ID of instance $\Pi_U^{j(t)}$ is a concatenation of all messages sent and received by the instance in a given execution of RID-AGKE protocol.

(vi) $sk_U^{j(t)}$: a group session key which is accepted by instance $\Pi_U^{j(t)}$.

In the following definitions, we will only focus on the three variables $pid_U^{j(t)}$, $sid_U^{j(t)}$, and $sk_U^{j(t)}$. The remaining variables will be left implicit. We say that two instances $\Pi_U^{j(t)}$ and $\Pi_{U'}^{j(v)}$ are *partnered* if (1) they have accepted the same group session key; (2) $sid_U^{j(t)} = sid_{U'}^{j(v)}$; and (3) $pid_U^{j(t)} = pid_{U'}^{j(v)}$.

Adversarial Model. An adversary A can be viewed as a probabilistic polynomial-time algorithm. Here, we assume that A can potentially control all communications in a RID-AGKE protocol. The interaction between A and instances of participants in the protocol is modeled by the following oracles.

(i) *Execute* (V, j): when A makes *Execute query* on (V, j), it executes the RID-AGKE protocol between the unused instances of participants in V for period index j and then returns a transcript of the execution, where V is a subset of G. Here, *Execute query* is used to model passive attacks.

(ii) *Inextract* (ID_U): when A makes *Initial key extract query* on identity ID_U, it generates an initial private key DID_U corresponding to ID_U and returns it to A, where $ID_U \notin \mathbf{ID}$.

(iii) *Kupdate* (ID_U, j): when A makes *Key update query* on (ID_U, j), it generates an update key $TID_{U,j}$ corresponding to (ID_U, j) and returns it to A, where $ID_U \notin \mathbf{ID}$ and j is a period index.

(iv) *Send* (U, j, t, M): when A makes *Send query* on (U, j, t, M), it sends message M to instance $\Pi_U^{j(t)}$ and then returns the reply generated by this instance according to procedures of RID-AGKE protocol.

(v) *Reveal* (U, j, t): when A makes *Reveal query* on (U, j, t), it returns a group session key $sk_U^{j(t)}$ for a terminated instance $\Pi_U^{j(t)}$. Here, *Reveal query* is used to model known session key attacks.

(vi) *Corrupt* (ID_U, j): when A makes *Corrupt query* on (ID_U, j), it returns a private key $DID_{U,j}$ of ID_U in period j. Note that *Corrupt query* models the corruption of this participant at a time in which it is not currently executing the protocol. We say that a participant U is honest if and only if no *Corrupt query* has been made by A.

(vii) *Test* $(U, j, t,)$: at any time, the adversary A makes *Test query* only once to this oracle during A's execution. In this moment, a random coin $b \in \{0,1\}$ is selected. If $b = 1$, a group session key $sk_U^{j(t)}$ is retuned. Otherwise, a random value is retuned. Here, *Test query* is used to model the semantic security of group session key.

According to the above adversarial model, we define two types of adversaries. A *passive adversary* is allowed to make *Execute*, *Reveal*, *Corrupt*, and *Test* queries. An *active adversary* is allowed to make the above all queries. In order to get more precise analysis, we still use *Execute query* though it can be substituted by making *Send query* repeatedly.

Remark 3. According to the adversarial model above, the adversary A can compute the participant U's private key $DID_{U,j} = DID_U + TID_{U,j}$ for period index j while A makes both *Initial key extract query* on ID_U and *Key update query* on (ID_U, j) simultaneously. Hence, we disallow A to make both queries in the same time.

Correctness. A RID-AGKE protocol is called *correct* if the following three conditions hold.

(1) All participants are honest and all messages are delivered honestly.

(2) $acc_U^{j(t)} = acc_{U'}^{j(v)} =$ "True" and $sk_U^{j(t)} = sk_{U'}^{j(v)}$.

(3) $sid_U^{j(t)} = sid_{U'}^{j(v)}$and $pid_U^{j(t)} = pid_{U'}^{j(v)}$ for all participants $U, U' \in V \subseteq G$ with instances $\Pi_U^{j(t)}$ and $\Pi_{U'}^{j(v)}$.

Freshness. We say that an instance $\Pi_U^{j(t)}$is called *fresh* (or called holding a *fresh* group session key $sk_U^{j(t)}$) if the following three conditions hold.

(1) $\Pi_U^{j(t)}$ has accepted a group session key $sk_U^{j(t)}$.

(2) Neither $\Pi_U^{j(t)}$ nor its partners have been made *Reveal query*.

(3) No *Corrupt query* has been made on $ID_V \in pid_U^{j(t)}$ before *Send query* to $\Pi_U^{j(t)}$ or *Send query* to $\Pi_{U'}^{j(v)}$, where $ID_{U'} \in pid_U^{j(t)}$.

Here, we assume all instances are *fresh*. Note that the notion of *freshness* is defined appropriately for the purpose of forward secrecy.

Secure RID-AGKE. A secure RID-AGKE protocol contains the following four parts.

(1) *Freshness.*

(2) *Security of RID-AGKE Protocol*. The security of RID-AGKE protocol is defined in the following game played between an active adversary A and a set of instances:

(a) *initialization*: the system private key, public parameters, and participants' private keys are generated in this phase;

(b) *query*: A may make different types of queries to oracles and gets back the answers corresponding to the RID-AGKE protocol;

(c) *guess*: finally, the adversary A outputs its guess for the coin b in *Test query* and terminates.

In this game, the goal of A is to distinguish a group session key from a random value. Let *Succ* be the event that A correctly guesses the coin b in *Test query*. The advantage of A in attacking a RID-AGKE protocol Ψ is defined by $Adv_{A,\Psi}(k) = |2 \cdot \Pr[Succ] - 1|$. We say that the protocol Ψ is secure, if the advantage $Adv_{A,\Psi}(k)$ is negligible.

(3) *Forward Secrecy*. We say that a RID-AGKE protocol Ψ provides *forward secrecy*. It means that though an adversary A obtains participants' private keys in Ψ, the previous establishing group session keys is preserved. The advantage of A in attacking the protocol Ψ within running time t is defined by $Adv_{\Psi}^{RIDAGKE\text{-}fs}(t, q_{ex}, q_s)$, where q_{ex} and q_s are the maximum numbers of making *Execute* and *Send queries*, respectively.

(4) *Authentication*. We say that a RID-AGKE protocol Ψ provides *implicit key authentication* if all participants in Ψ are guaranteed that nobody other than their partners can learn the session key. In other words, any adversary should not learn the key. Note that this security property does not guarantee that the partners have computed the key.

Malicious Participant. A participant U_m is called *malicious* in a RID-AGKE protocol Ψ if he is a legal participant but is fully controlled by adversary. The goal of malicious participant is to disturb the group key establishing in Ψ.

4. Concrete Protocol

In this section, we propose a concrete RID-AGKE protocol with identifying malicious participants. Our protocol fuses the Tseng-Tsai R-IDPKS [14] and a noninteractive confirmed computation technique [6]. In the initialization phase, given a security parameter k and a total number z of periods, a private key generator (PKG) executes *Setup algorithm* to generate the system private key s and the public parameters $param = \{G_1, G_2, e, q, P, P_{pub}, H_1, H_2, H_3, H_4\}$ defined in Notations section at the end of the paper.

When a participant U with identity $ID_U \in \{0,1\}^*$ wants to obtain her/his initial private key DID_U, the PKG runs *Initial key extract algorithm* to compute $DID_U = s \cdot H_1(ID_U) = s \cdot QID_U$ and returns it to U via a secure channel. For a nonrevoked participant U with identity ID_U in time period j, the PKG runs *Key update algorithm* to compute her/his update key $TID_{U,j} = s \cdot H_2(ID_U, j) = s \cdot RID_{U,j}$ and returns it to U via a public channel, where $1 \le j \le z$. Hence, any nonrevoked participant U can update her/his private key $DID_{U,j} = DID_U + TID_{U,j}$ by itself in period j.

Let $G = \{U_1, U_2, \dots, U_n\}$ be a set of participants who want to establish a group session key SK_j in period j. We assume that each U_i has a unique identity $ID_i \in \{0,1\}^*$ as public key and U_i's private key is $DID_{i,j} = DID_i + TID_{i,j}$ for period j. Note that the indices are subject to modulo n; that is, U_{n+1} and U_0 denote U_1 and U_n, respectively. Finally, $m \in \{0,1\}^*$ is a preknown common message by all participants. The details of proposed RID-AGKE protocol are described as follows.

Round 1. Each participant U_i randomly selects a secret value $a_i \in Z_q{}^*$ and computes $P_i = a_i \cdot P$, $h_i = H_3(ID_i, PID_j, j, m, P_i)$, and $V_i = DID_{i,j} + a_i \cdot h_i \cdot P_{pub}$, where PID_j denotes the concatenation of all participants' identities in period j; that is, $PID_j = ID_1 \parallel ID_2 \parallel \cdots \parallel ID_n$. Finally, each U_i broadcasts (ID_i, j, P_i, V_i) to other participants.

Round 2. Upon receiving $(ID_{i-1}, j, P_{i-1}, V_{i-1})$ and $(ID_{i+1}, j, P_{i+1}, V_{i+1})$, each U_i first verifies them by checking

$$e\left(P, \sum_{k\in\{-1,1\}} V_{i+k}\right) \overset{?}{=} e\left(P_{pub}, \sum_{k\in\{-1,1\}} H_1\left(ID_{i+k}\right) + H_2\left(ID_{i+k}, j\right) + h_{i+k} \cdot P_{i+k}\right), \tag{1}$$

where $h_{i+k} = H_3(ID_{i+k}, PID_j, j, m, P_{i+k})$. If the verification is true, each U_i uses her/his secret value a_i to compute $D_i = e(a_i \cdot (P_{i+1} - P_{i-1}), P_{pub})$. Then, U_i randomly selects a value $r_i \in Z_q{}^*$ and computes a tuple $(ID_i, j, D_i, \alpha_i, \beta_i, \gamma_i)$, where $\alpha_i = r_i \cdot P$, $\beta_i = r_i \cdot (P_{i+1} - P_{i-1})$, $\gamma_i = r_i \cdot P_i + w_i \cdot a_i \cdot P_{pub}$, $w_i = H_4(ID_i \parallel PID_j \parallel j \parallel D_i \parallel S_j, P_{i+1} - P_{i-1}, \alpha_i, \beta_i)$, and $S_j = P_1 \parallel P_2 \parallel \cdots \parallel P_n$. Finally, U_i sends this tuple to all other participants.

Group Key Computation. Upon receiving all $(ID_k, j, D_k, \alpha_k, \beta_k, \gamma_k)$ for $k = 1, 2, \ldots, n$ except i, each U_i verifies them by checking

$$\begin{gathered} e\left(P, \sum_{k=1, k \neq i}^{n} \gamma_k\right) = \prod_{k=1, k \neq i}^{n} e\left(P_k, \alpha_k + w_k \cdot P_{pub}\right), \\ e\left(P_{k+1} - P_{k-1}, \gamma_k\right) \overset{?}{=} e\left(\beta_k, P_k\right) \cdot D_k^{w_k}. \end{gathered} \tag{2}$$

If the two verifications hold, U_i can confirm that each D_k is computed by U_k using her/his secret a_k honestly for $k = 1, 2, \ldots, n$ except i. Finally, in period j, each participant U_i can compute the group session key $SK_j = e(a_i \cdot P_{i-1}, P_{pub})^n \cdot D_i^{n-1} \cdot D_{i+1}^{n-2} \cdots D_{i-2}$.

Identifying Malicious Participant. When a malicious participant U_m tries to send a wrong tuple $(ID_m, D_m, j, \alpha_m, \beta_m, \gamma_m)$ to disrupt the establishment of group session key, he will be identified as a malicious participant by using the following two verifying equations: $e(P, \gamma_k) \overset{?}{=} e(P_k, \alpha_k + w_k \cdot P_{pub})$ and $e(P_{k+1} - P_{k-1}, \gamma_k) \overset{?}{=} e(\beta_k, P_k) \cdot D_k^{w_k}$. Later on, U_m will be deleted from the participant set G and other honest participants may rerun the protocol.

5. Security Analysis

In this section, we prove the security of the proposed RID-AGKE protocol in the random oracle model [19] and under the CDH and DBDH assumptions.

ID and Forgery Attacks

Theorem 4. *The proposed RID-AGKE protocol is secure against ID and forgery attacks.*

Proof. Note that we adopt a revocable ID-based signature (RIDS) scheme [16] in Round 1 and a pairing-based signature scheme [6] in Round 2, respectively. The two signature schemes had been proven secure against ID and forgery attacks for single signature and multiple signatures with batch verification. Therefore, the proposed RID-AGKE protocol Ψ is secure against ID and forgery attacks.

Secure RID-AGKE Providing Forward Secrecy. Now, we demonstrate that the proposed RID-AGKE protocol Ψ is a secure RID-AGKE providing forward secrecy. Note that we use a similar technique in [3, 4, 6] to prove Theorem 5. □

Theorem 5. *Assume that four hash functions H_1, H_2, H_3, and H_4 are random oracles. Then, the proposed RID-AGKE protocol Ψ is a secure RID-AGKE providing forward secrecy under the decisional bilinear Diffie-Hellman (DBDH) and the computational Diffie-Hellman (CDH) assumptions. Concretely,*

$$\begin{aligned} & Adv_{\Psi}^{RIDAGKE\text{-}fs}\left(t, q_{ex}, q_s\right) \\ & \quad \leq 2nq_{ex} \cdot Adv_{DBDH}(t) + Adv_{\Psi}^{forge}(t), \end{aligned} \tag{3}$$

where q_{ex} and q_s are total numbers of making Execute and Send queries, respectively. Note that $Adv_{\Psi}^{forge}(t)$ denotes the advantage of any forgers successfully attacking the protocol Ψ.

Proof. Assume that A is an active adversary in attacking the proposed RID-AGKE protocol Ψ with a nonnegligible advantage. Now, we consider the two possible cases. The first case is that A with the advantage can impersonate a participant (i.e., forging authentication transcripts). Another case is that A with the advantage can break the protocol Ψ without modifying any transcripts.

Case 1. We assume that the adversary A with an adaptive impersonation ability can break the RID-AGKE protocol Ψ. Using A, we would like to construct a forger F which can return valid signature tuples (ID, j, aP, V) and $(ID, j, D, \alpha, \beta, \gamma)$ with respect to the proposed protocol Ψ as follows. The forger F first generates all needed system parameters and keys. Then, F simulates the oracle queries made by A. This simulation is called perfect indistinguishable from A's oracle queries except that A makes *Corrupt query* on (ID, j), where j is a period index. If it occurs, F fails and stops. Otherwise, when A generates two signature tuples (ID, j, aP, V) and $(ID, j, D, \alpha, \beta, \gamma)$, F returns the tuples (ID, j, aP, V) and $(ID, j, D, \alpha, \beta, \gamma)$. Let *Forge* be the event that the adversary A successfully generates two valid signature tuples. Then, the probability that F successfully returns two valid signature tuples is bounded by $\Pr_A[Forge] \leq Adv_{F,\Psi}^{forge}(t) \leq Adv_{\Psi}^{forge}(t)$.

Case 2. We assume that the adversary A can break the proposed RID-AGKE protocol without modifying any transcripts. We first focus on the case that A makes *Execute query* once on $(ID_1, ID_2, \ldots, ID_n, j)$ and then extends this to the case that A makes multiple *Execute queries*, where the number of participants n and period j are selected by A. The real execution of Ψ is given by

$$param = \left\{ \begin{array}{c} \left(G_1, G_2, e\right) \longleftarrow \text{PKG}; P \longleftarrow G_1; s \longleftarrow Z_q{}^*; P_{pub} = s \cdot P; \\ QID_1, \ldots, QID_n, RID_{1,j}, \ldots, RID_{n,j} \longleftarrow G_1; \\ DID_{1,j} = \left(QID_1 + RID_{1,j}\right) \cdot s, \ldots, DID_{n,j} = \left(QID_n + RID_{n,j}\right) \cdot s : \\ \left(G_1, G_2, e, P, P_{pub}, PID\right) \end{array} \right\},$$

$$Real = \left\{ \begin{array}{c} a_1, \dots, a_n, h_1, \dots, h_n, r_1, \dots, r_n, w_1, \dots, w_n \longleftarrow Z_q{}^*; \\ P_1 = a_1 P, \dots, P_n = a_n P; V_1 = DID_{1,j} + a_1 h_1 P_{pub}, \dots, V_n = DID_{n,j} + a_n h_n P_{pub}; \\ D_1 = e(P_2 - P_n, P_{pub})^{a_1}, \dots, D_n = e(P_1 - P_{n-1}, P_{pub})^{a_n}; \\ \alpha_1 = r_1 P, \dots, \alpha_n = r_n P; \beta_1 = r_1 \left(P_2 - P_n\right), \dots, \beta_n = r_n \left(P_1 - P_{n-1}\right); \\ \gamma_1 = r_1 P_1 + w_1 a_1 P_{pub}, \dots, \gamma_n = r_n P_n + w_n a_n P_{pub}; \\ T = \left(P_1, \dots, P_n, V_1, \dots, V_n, D_1, \dots, D_n, \alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n, \gamma_1, \dots, \gamma_n\right); \\ SK_j = e(a_1 P_n, P_{pub})^n \cdot D_1^{n-1} \cdot D_2^{n-2} \cdots D_{n-1} : \left(j, T, SK_j\right) \end{array} \right\}, \tag{4}$$

where T denotes the transcript and SK_j is the group session key for period j.

In *Real*, each $D_i = e(P_{i+1} - P_{i-1}, P_{pub})^{a_i} = e(a_i P_{i+1}, P_{pub})/e(a_i P_{i-1}, P_{pub}) = e(a_i a_{i+1} P, P_{pub})/e(a_{i-1} a_i P, P_{pub})$. by the bilinear pairing operations. We can use a random value $d_{1,2} \in Z_q{}^*$ to substitute $a_1 \cdot a_2$. Thus, a new distribution $Fake_1$ is obtained as follows:

$$Fake_1 = \left\{ \begin{array}{c} d_{1,2}, a_1, \dots, a_n, h_1, \dots, h_n, r_1, \dots, r_n, w_1, \dots, w_n \longleftarrow Z_q{}^*; \\ P_1 = a_1 P, \dots, P_n = a_n P; \\ V_1 = DID_{1,j} + a_1 h_1 P_{pub}, \dots, V_n = DID_{n,j} + a_n h_n P_{pub}; \\ D_1 = \dfrac{e\left(d_{1,2} P, P_{pub}\right)}{e\left(a_n a_1 P, P_{pub}\right)}, D_2 = \dfrac{e\left(a_2 a_3 P, P_{pub}\right)}{e\left(d_{1,2} P, P_{pub}\right)}, \dots, D_n = \dfrac{e\left(a_n a_1 P, P_{pub}\right)}{e\left(a_{n-1} a_n P, P_{pub}\right)}; \\ \alpha_1 = r_1 P, \dots, \alpha_n = r_n P; \beta_1 = r_1 \left(P_2 - P_n\right), \dots, \beta_n = r_n \left(P_1 - P_{n-1}\right); \\ \gamma_1 = r_1 P_1 + w_1 a_1 P_{pub}, \dots, \gamma_n = r_n P_n + w_n a_n P_{pub}; \\ T = \left(P_1, \dots, P_n, V_1, \dots, V_n, D_1, \dots, D_n, \alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n, \gamma_1, \dots, \gamma_n\right); \\ SK_j = e(a_1 P_n, P_{pub})^n \cdot D_1^{n-1} \cdot D_2^{n-2} \cdots D_{n-1} : \left(j, T, SK_j\right) \end{array} \right\}. \tag{5}$$

Note that A can obtain all private keys $DID_{i,j}$ and hash values h_i by making *Corrupt* and *Hash queries*. It means that A can compute all $a_i \cdot P_{pub} = h_i^{-1} \cdot (V_i - DID_{i,j})$ for $i = 1, 2, \dots, n$. Since the discrete logarithm assumption in G_1 is intractable, A cannot obtain some information about a_i from $a_i \cdot P_{pub}$ for $i = 1, 2, \dots, n$.

In the following claim, we want to show that to distinguish two distributions *Real* from $Fake_1$ can be reduced to solve the decisional bilinear Diffie-Hellman (DBDH) problem. Let $\varepsilon(t) = Adv_{DBDH}(t)$.

Claim. For any algorithm A with running time t, we have

$$\begin{aligned} &\left| \Pr\left[\left(j, T, SK_j\right) \longleftarrow Real \,\middle|\, A\left(j, T, SK_j\right) = 1\right] \right. \\ &\qquad \left. - \Pr\left[\left(j, T, SK_j\right) \longleftarrow Fake_1 \,\middle|\, A\left(j, T, SK_j\right) = 1\right] \right| \\ &\quad \leq \varepsilon\left(t\right). \end{aligned} \tag{6}$$

Proof. As mentioned above, each $D_i = e(a_i a_{i+1} P, P_{pub}) /e(a_{i-1} a_i P, P_{pub}) = e(P, P_{pub})^{a_i a_{i+1}} e(P, P_{pub})^{a_{i-1} a_i}$. Here, we use $\Gamma_{i,i+1}$ to substitute $e(P, P_{pub})^{a_i a_{i+1}}$ and then each D_i can be written into $\Gamma_{i,i+1}/\Gamma_{i-1,i}$ for $i = 1, 2, \dots, n$. Hence, the group session key SK_j also can be written into $(\Gamma_{n,1})^n \cdot D_1^{n-1} \cdot D_2^{n-2} \cdots D_{n-1}$, where $(\Gamma_{n,1})^n = e(P, P_{pub})^{n a_n a_1} = e(a_1 P_n, P_{pub})^n$.

To solve the DBDH problem, we use a technique to dispose the related parameter. Considering the following algorithm D which inputs $P_a = aP$,$P_b = bP$, and $P_c = cP \in G_1$ for some $a, b, c \in_R Z_q{}^*$. D first generates (j, T, SK_j) according to the distribution $Dist^1$. Then, D runs $A(j, T, SK_j)$ and outputs whatever A outputs. The distribution $Dist^1$ is defined as follows:

$$Dist^1 = \left\{ \begin{array}{c} a_1, \dots, a_n, h_1, \dots, h_n, u_1, \dots, u_{n-2}, r_1, \dots, r_n, w_1, \dots, w_n \longleftarrow Z_q{}^*; \\ P_1 = a_1 P, \dots, P_n = a_n P; V_1 = DID_{1,j} + a_1 h_1 P_{pub}, \dots, V_n = DID_{n,j} + a_n h_n P_{pub}; \\ \Gamma_{1,2} = g_{sab} \in G_2, \Gamma_{2,3} = e\left(P_b, P_{pub}\right)^{u_1}, \Gamma_{i,i+1} = e\left(P, P_{pub}\right)^{u_{i-2} u_{i-1}} \text{ for } i = 3 \text{ to } n - 1 \\ \Gamma_{n,1} = e(P_a, P_{pub})^{u_{n-2}}; D_1 = \dfrac{\Gamma_{1,2}}{\Gamma_{n,1}}, \dots, D_n = \dfrac{\Gamma_{n,1}}{\Gamma_{n-1,n}}; \\ \alpha_1 = r_1 P, \dots, \alpha_n = r_n P; \beta_1 = r_1 \left(P_2 - P_n\right), \dots, \beta_n = r_n \left(P_1 - P_{n-1}\right); \\ \gamma_1 = r_1 P_1 + w_1 a_1 P_{pub}, \dots, \gamma_n = r_n P_n + w_n a_n P_{pub}; \\ T = \left(P_1, \dots, P_n, V_1, \dots, V_n, D_1, \dots, D_n, \alpha_1, \dots, \alpha_n, \beta_1, \dots, \beta_n, \gamma_1, \dots, \gamma_n\right); \\ SK_j = (\Gamma_{n,1})^n \cdot D_1^{n-1} \cdot D_2^{n-2} \cdots D_{n-1} : \left(j, T, SK_j\right) \end{array} \right\}. \tag{7}$$

Note that this distribution depends on P_a, P_b, and P_c.

By the above distribution $Dist^1$, let $\Gamma_{1,2} = e(P, P_{pub})^{ab} = e(P,P)^{sab}$. Then, we can obtain another distribution called $Dist^1_{DBDH}$. Obviously, $Dist^1_{DBDH}$ is identical to *Real* because

$$\begin{aligned}SK_j &= (\Gamma_{n,1})(\Gamma_{1,2})(\Gamma_{2,3})\cdots(\Gamma_{n-2,n-1})(\Gamma_{n-1,n})\\ &= e(P_a, P_{pub})^{u_{n-2}} \cdot e(P,P)^{sab}\\ &\quad\cdot e\left(P_b, P_{pub}\right)^{u_1} \cdots e\left(P, P_{pub}\right)^{u_{n-4}u_{n-3}}\\ &\quad\cdot e(P, P_{pub})^{u_{n-3}u_{n-2}}\\ &= e(P,P)^{su_{n-2}a+sab+sbu_1+\cdots+su_{n-4}u_{n-3}+su_{n-3}u_{n-2}}.\end{aligned} \tag{8}$$

Similarly, let $\Gamma_{1,2} = e(P_c, P_{pub}) = e(P,P)^{sc}$ for some $c \neq ab \in Z_q{}^*$. Then, we can obtain another distribution called $Dist^1_{Random}$. Obviously, $Dist^1_{Random}$ is identical to $Fake_1$ because

$$\begin{aligned}SK_j &= (\Gamma_{n,1})(\Gamma_{1,2})(\Gamma_{2,3})\cdots(\Gamma_{n-2,n-1})(\Gamma_{n-1,n})\\ &= e(P_a, P_{pub})^{u_{n-2}} \cdot e(P,P)^{sc}\\ &\quad\cdot e\left(P_b, P_{pub}\right)^{u_1} \cdots e\left(P, P_{pub}\right)^{u_{n-4}u_{n-3}}\\ &\quad\cdot e(P, P_{pub})^{u_{n-3}u_{n-2}}\\ &= e(P,P)^{su_{n-2}a+sc+sbu_1+\cdots+su_{n-4}u_{n-3}+su_{n-3}u_{n-2}}.\end{aligned} \tag{9}$$

Therefore, we have

$$\begin{aligned}&\left|\Pr\left[\left(j,T,SK_j\right) \longleftarrow Real \;\middle|\; A\left(j,T,SK_j\right)=1\right]\right.\\ &\quad\left.-\Pr\left[\left(j,T,SK_j\right) \longleftarrow Fake_1 \;\middle|\; A\left(j,T,SK_j\right)=1\right]\right|\\ &\leq \varepsilon(t).\end{aligned} \tag{10}$$

This completes the proof of claim.

Using the same process in $Fake_1$, we can define other distributions $Fake_i$ for $i = 2, 3, \ldots, n$. By a similar approach in claim, we can obtain the following n-1 equations in (11) for any adversary A with running time t

$$\begin{aligned}&\left|\Pr\left[\left(j,T,SK_j\right) \longleftarrow Fake_1 \;\middle|\; A\left(j,T,SK_j\right)=1\right]\right.\\ &\quad\left.-\Pr\left[\left(j,T,SK_j\right) \longleftarrow Fake_2 \;\middle|\; A\left(j,T,SK_j\right)=1\right]\right|\\ &\leq \varepsilon(t),\\ &\qquad\vdots\\ &\left|\Pr\left[\left(j,T,SK_j\right) \longleftarrow Fake_{n-1} \;\middle|\; A\left(j,T,SK_j\right)=1\right]\right.\\ &\quad\left.-\Pr\left[\left(j,T,SK_j\right) \longleftarrow Fake_n \;\middle|\; A\left(j,T,SK_j\right)=1\right]\right|\\ &\leq \varepsilon(t).\end{aligned} \tag{11}$$

This implies

$$\begin{aligned}&\left|\Pr\left[\left(j,T,SK_j\right) \longleftarrow Real \;\middle|\; A\left(j,T,SK_j\right)=1\right]\right.\\ &\quad\left.-\Pr\left[\left(j,T,SK_j\right) \longleftarrow Fake_n \;\middle|\; A\left(j,T,SK_j\right)=1\right]\right|\\ &\leq n\cdot\varepsilon(t).\end{aligned} \tag{12}$$

In $Fake_n$, the values $d_{1,2}, d_{2,3}, \ldots, d_{n-1,n}, d_{n,1}$ are constrained by T according to the following n equations:

$$\begin{aligned}\log_g D_1 &= s\cdot\left(d_{1,2} - d_{n,1}\right),\\ \log_g D_2 &= s\cdot\left(d_{2,3} - d_{1,2}\right), \ldots,\\ \log_g D_n &= s\cdot\left(d_{n,1} - d_{n-1,n}\right),\end{aligned} \tag{13}$$

where $g = e(P,P)$. Since SK_j can be expressed as $e(P,P)^{sd_{1,2}+sd_{2,3}+\cdots+sd_{n,1}}$, we can obtain $\log_g SK_j = sd_{1,2} + sd_{2,3} + \cdots + sd_{n,1}$. Because $sd_{1,2} + sd_{2,3} + \cdots + sd_{n,1}$ is linear and independent from the set $\{\log_g D_i = s\cdot(d_{i,i+1} - d_{i-1,i}) \mid i = 1, 2, \ldots, n\}$, it implies that SK_j is independent for the transcript T. In other words, for any adversary A

$$\begin{aligned}&\Pr\left[\left(j,T,SK_{j,0}\right) \longleftarrow Fake_n, SK_{j,1} \longleftarrow G_2 \mid A\left(j,T,SK_{j,b}\right)\right.\\ &\qquad\left.=1, b \longleftarrow \{0,1\}\right] = \frac{1}{2}.\end{aligned} \tag{14}$$

Therefore, the advantage of A on the event $\neg Forge$ is bounded by $2n \cdot Adv_{DBDH}(t)$. Combining the two cases, the advantage of A is bounded by

$$Adv_{\Psi}^{RIDAGKE\text{-}fs}(t, 1, q_s) \leq 2n\cdot Adv_{DBDH}(t) + Adv_{\Psi}^{forge}(t). \tag{15}$$

Finally, a standard hybrid argument immediately demonstrates that

$$\begin{aligned}Adv_{\Psi}^{RIDAGKE\text{-}fs}(t, q_{ex}, q_s) \leq{}& 2nq_{ex}\cdot Adv_{DBDH}(t)\\ &+ Adv_{\Psi}^{forge}(t) \text{ for } q_{ex} > 1.\end{aligned} \tag{16}$$

□

Under the decisional bilinear Diffie-Hellman (DBDH) assumption, the advantage $Adv_{DBDH}(t)$ is negligible. By Theorem 4, the advantage $Adv_{\Psi}^{forge}(t)$ is also negligible. Hence, we can obtain that the advantage $Adv_{\Psi}^{RIDAGKE\text{-}fs}(t, q_{ex}, q_s)$ is negligible according to the result in Theorem 5. It implies that the proposed RID-AGKE protocol Ψ is a secure RID-AGKE providing forward secrecy.

Identifying Malicious Participant

Theorem 6. *The proposed RID-AGKE protocol can identify malicious participants.*

Table 1: Comparisons between our protocol and the previously proposed AGKE protocols.

	Tseng's AGKE [25]	Choi et al.'s ID-AGKE [26]	Wu et al.'s ID-AGKE [6]	Wu et al.'s RID-AGKE [4]	Our protocol
Public key setting	ElGmal	IDPKS	IDPKS	R-IDPKS	R-IDPKS
Certificate management	Required	Not required	Not required	Not required	Not required
Rounds	2	2	2	3	2
Computational cost for each participant	$(8n-2)\,T_{exp}$ + $(n+1)\,T_{inv}$	$6TG_e$ + $(n+11)\,TG_{mul}$ + $(n+3)\,TG_H$	$(3n+3)\,TG_e$ + $(n+10)\,TG_{mul}$ + $3TG_H + (n-1)\,T_{exp}$	$8TG_e$ + $(2n+8)\,TG_{mul}$ + $4nTG_H$	$(3n+2)\,TG_e$ + $(n+9)\,TG_{mul}$ + $4TG_H + (n-1)\,T_{exp}$
Security	Provably secure	Existing attack [27]	Provably secure	Provably secure	Provably secure
Revocation functionality	Using CRL [28]	No	No	Yes	Yes
Resistant to malicious participants	Yes (confirmed computation)	No	Yes (confirmed computation)	Yes (explicit key confirmation)	Yes (confirmed computation)
Identifying malicious participants	Yes	No	Yes	No	Yes

Proof. Note that in Round 2 a noninteractive confirmed computation technique is involved in adopted pairing-based signature scheme. The security of confirmed computation had been proven in [6]. Concretely, each participant U_i can confirm the broadcasted value D_k is computed by U_k using her/his secret a_k after passing two verifying equations for $k = 1, 2, \ldots, n$ except i. Hence, if there is a participant U_m who broadcasts a wrong D_m to disturb the group session key establishing, he will be identified as a malicious participant. In other words, the proposed RID-AGKE protocol can identify malicious participants by using the confirmed computation technique. □

6. Performance Analysis and Comparisons

For convenience to evaluate the computational cost, we focus on the time-consuming pairing-based operations as follows:

(i) TG_e: the time of executing a bilinear map operation $e : G_1 \times G_1 \rightarrow G_2$;

(ii) TG_{mul}: the time of executing a point scalar multiplication operation in G_1;

(iii) TG_H: the time of executing a map-to-point hash function $H_1, H_2 : \{0,1\}^* \rightarrow G_1$;

(iv) T_{exp}: the time of executing a modular exponentiation operation over a finite field F_p, where p is a large prime number;

(v) T_{inv}: the time of executing a modular multiplicative inverse operation over a finite field F_p, where p is a large prime number.

Here, we first analyze the computational cost of our protocol. In Round 1, $2TG_{mul}$ is required to compute (P_i, V_i). In Round 2, each participant requires $3TG_e + 7TG_{mul} + 4TG_H$ to verify $(ID_{i+k}, j, P_{i+k}, V_{i+k})$ for $k \in \{-1, 1\}$ and to generate $(D_i, \alpha_i, \beta_i, \gamma_i)$. In the group key computation phase, $(3n-1)TG_e + nTG_{mul} + (n-1)T_{exp}$ is required to verify all $(ID_i, j, D_i, \alpha_i, \beta_i, \gamma_i)$ and to compute a group key SK_j. Note that to evaluate $SK_j = e(a_i \cdot P_{i-1}, P_{pub})^n \cdot D_i^{n-1} \cdot D_{i+1}^{n-2} \cdot \cdot \cdot D_{i-2}$ is required $TG_e + TG_{mul}$ since $SK_j = A_{i-1} \cdot A_i \cdot A_{i+1} \cdots A_{i-2}$, where $A_{i-1} = e(a_i \cdot P_{i-1}, P_{pub})$, $A_i = A_{i-1} \cdot D_i$, $A_{i+1} = A_i \cdot D_{i+1}, \ldots$, and $A_{i-2} = A_{i-3} \cdot D_{i-2}$. As a result, each participant requires $(3n+2)TG_e + (n+9)TG_{mul} + 4TG_H + (n-1)T_{exp}$ in our protocol.

In Table 1, we compare our RID-AGKE protocol with four previously proposed AGKE protocols which include Tseng's AGKE protocol [25], Choi et al.'s ID-AGKE protocol [26], Wu et al.'s ID-AGKE protocol [6], and Wu et al.'s RID-AGKE protocol [4] in terms of the public key setting, number of rounds, computational cost, and security properties. One recent non-ID-based and non-RID-based AGKE protocol with identifying malicious participants was proposed by Tseng [25]. Since Tseng's protocol is based on the ElGmal system [29], each participant must verify the other participants' certificates for participant authentication. It will increase the required computational costs for verifying certificates, besides $(8n-2)T_{exp} + (n+1)T_{inv}$. On the contrary, Choi et al.'s ID-AGKE [26], Wu et al.'s ID-AGKE [6], Wu et al.'s RID-AGKE [4], and our protocol rely on the IDPKS system [8] or the R-IDPKS system [14]. Thus, they need not manage and verify the participants' certificates. However, Choi et al.'s ID-AGKE [26] suffered from an insider colluding attack demonstrated by Wu and Tseng [27].

For Wu et al.'s ID-AGKE [6], Wu et al.'s RID-AGKE [4], and our protocol, they are provably secure and are able to resist malicious participants. It is easy to see that our protocol is more efficient than Wu et al.'s ID-AGKE [6] even though both protocols can identify malicious participants via confirmed computation approach. More importantly, Wu et al.'s ID-AGKE protocol [6] does not provide a solution to revoke the compromised or misbehaving user in the group. It is very serious because these revoked participants should not be allowed to establish a common key with other legal (nonrevoked) participants. In another aspect, Wu et al.'s RID-AGKE [4] is a three-round protocol and adopts explicit key confirmation approach to resist malicious participants.

Though their protocol can detect the existence of malicious participants, it cannot still identify malicious participant. Our proposed RID-AGKE is a two-round protocol and provides an active detection mechanism to identify malicious participants. According to Table 1, the advantage of our protocol is demonstrated.

7. Conclusions

In this paper, we have fused the Tseng-Tsai R-IDPKS system and a noninteractive confirmed computation technique to propose the first RID-AGKE protocol with identifying malicious participants. The framework and security notions for RID-AGKE protocols have been defined to formalize the possible threats and attacks. When compared with the recently proposed ID/RID-AGKE protocols resistant to malicious participants, our protocol has better performance and provides an active detection way to identify malicious participants. In the random oracle model and under two mathematical assumptions (CDH and DBDH), we have proven that the proposed protocol is a secure RID-AGKE protocol with forward secrecy and identifying malicious participants.

Notations

e:	A bilinear map, $e : G_1 \times G_1 \rightarrow G_2$, defined in Section 2.1
s:	The system private key, $s \in {Z_q}^*$
P:	A generator of group G_1
P_{pub}:	The system public key, $P_{pub} = s \cdot P$
ID_i:	The identity of participant U_i
DID_i:	The participant U_i's initial private key
$TID_{i,j}$:	The participant U_i's update key for period j
$DID_{i,j}$:	The participant U_i's private key for period j, $DID_{i,j} = DID_i + TID_{i,j}$
H_1:	A map-to-point hash function, $H_1 : \{0,1\}^* \rightarrow G_1$
H_2:	A map-to-point hash function, $H_2 : \{0,1\}^* \rightarrow G_1$
H_3:	A hash function, $H_3 : \{0,1\}^* \times G_1 \rightarrow Z_q$
H_4:	A hash function, $H_4 : \{0,1\}^* \times G_1^3 \rightarrow Z_q$.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors thank the referees for their valuable comments and constructive suggestions. This research was partially supported by Shenzhen Peacock Project of China (no. KQC201109020055A) and Shenzhen Strategic Emerging Industries Program of China (no. ZDSY20120613125016389).

References

[1] M. Burmester and Y. Desmedt, "A secure and efficient conference key distribution system," in *Advances in Cryptology—EUROCRYPT '94*, vol. 950 of *Lecture Notes in Computer Science*, pp. 275–286, 1995.

[2] J. Katz and J. S. Shin, "Modeling insider attacks on group key-exchange protocols," in *Proceedings of the12th ACM Conference on Computer and Communications Security (CCS '05)*, pp. 180–189, November 2005.

[3] T.-Y. Wu, Y.-M. Tseng, and C.-W. Yu, "A secure ID-Based authenticated group key exchange protocol resistant to insider attacks," *Journal of Information Science and Engineering*, vol. 27, no. 3, pp. 915–932, 2011.

[4] T.-Y. Wu, Y.-M. Tseng, and T.-T. Tsai, "A revocable ID-based authenticated group key exchange protocol with resistant to malicious participants," *Computer Networks*, vol. 56, no. 12, pp. 2994–3006, 2012.

[5] Y.-M. Tseng, "A robust multi-party key agreement protocol resistant to malicious participants," *The Computer Journal*, vol. 48, no. 4, pp. 480–487, 2005.

[6] T.-Y. Wu and Y.-M. Tseng, "Towards ID-based authenticated group key exchange protocol with identifying malicious participants," *Informatica*, vol. 23, no. 2, pp. 315–334, 2012.

[7] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Identity-Based Cryptosystems and Signature Schemes*, vol. 196 of *Lecture Notes in Computer Science*, pp. 47–53, 1985.

[8] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," *SIAM Journal on Computing*, vol. 32, no. 3, pp. 586–615, 2003, Preliminary version: in Advances in Cryptology—CRYPTO '01, vol. 2139 of Lecture Notes in Computer Science, pp. 213–229, 2001.

[9] A. Boldyreva, V. Goyal, and V. Kumart, "Identity-based encryption with efficient revocation," in *Proceedings of the 15th ACM conference on Computer and Communications Security (CCS '08)*, pp. 417–426, October 2008.

[10] R. Canetti, S. Halevi, and J. Katz, "A forward-secure public-key encryption scheme," *Journal of Cryptology*, vol. 20, no. 3, pp. 265–294, 2007, Preliminary version: in Advances in Cryptology—EUROCRYPT 2003, vol. 2656 of Lecture Notes in Computer Science , pp. 255–271, 2003.

[11] B. Libert and D. Vergnaud, "Adaptive-ID secure revocable identity-based encryption," in *Topics in Cryptology—CT-RSA 2009*, vol. 5473 of *Lecture Notes in Computer Science*, pp. 1–15, 2009.

[12] J. H. Seo and K. Emura, "Revocable identity-based encryption revisited: security model and construction," in *Public-Key Cryptography—PKC 2013*, vol. 7778 of *Lecture Notes in Computer Science*, pp. 216–234, 2013.

[13] J. H. Seo and K. Emura, "Efficient delegation of key generation and revocation functionalities in identity-based encryption," in *Topics in Cryptology—CT-RSA 2013*, vol. 7779 of *Lecture Notes in Computer Science*, pp. 343–358, 2013.

[14] Y.-M. Tseng and T.-T. Tsai, "Efficient revocable ID-based encryption with a public channel," *The Computer Journal*, vol. 55, no. 4, pp. 475–486, 2012.

[15] T.-T. Tsai, Y.-M. Tseng, and T.-Y. Wu, "A fully secure revocable ID-based encryption in the standard model," *Informatica*, vol. 23, no. 3, pp. 487–505, 2012.

[16] T.-Y. Wu, T.-T. Tsai, and Y.-M. Tseng, "Revocable ID-based signature scheme with batch verifications," in *Proceedings of the*

8th International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP '12), pp. 49–54, July 2012.

[17] T. T. Tsai, Y. M. Tseng, and T. Y. Wu, "Provably secure revocable ID-based signature in the standard model," *Security and Communication Networks*, vol. 6, no. 10, pp. 1250–1260, 2013.

[18] T.-Y. Wu, T.-T. Tsai, and Y.-M. Tseng, "A revocable ID-based signcryption scheme," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 3, no. 3, pp. 240–251, 2012.

[19] M. Bellare and P. Rogaway, "Random oracles are practical: a paradigm for designing efficient protocols," in *Proceedings of the 1st ACM Conference on Computer and Communications Security (CCS '93)*, pp. 62–73, November 1993.

[20] L. Chen, Z. Cheng, and N. P. Smart, "Identity-based key agreement protocols from pairings," *International Journal of Information Security*, vol. 6, no. 4, pp. 213–241, 2007.

[21] T.-Y. Wu and Y.-M. Tseng, "An ID-based mutual authentication and key exchange protocol for low-power mobile devices," *The Computer Journal*, vol. 53, no. 7, pp. 1062–1070, 2010.

[22] K. Y. Choi, J. Y. Hwang, and D. H. Lee, "Efficient ID-based group key agreement with bilinear maps," in *Public Key Cryptography—PKC 2004*, vol. 2947 of *Lecture Notes in Computer Science*, pp. 130–144, 2004.

[23] J. Katz and J. S. Shin, "Modeling insider attacks on group key-exchange protocols," in *Proceedings of the12th ACM Conference on Computer and Communications Security (CCS '05)*, pp. 180–189, November 2005.

[24] R. Steinwandt and A. S. Corona, "Attribute-based group key establishment," *Advances in Mathematics of Communications*, vol. 4, no. 3, pp. 381–398, 2010.

[25] Y.-M. Tseng, "A communication-efficient and fault-tolerant conference-key agreement protocol with forward secrecy," *Journal of Systems and Software*, vol. 80, no. 7, pp. 1091–1101, 2007.

[26] K. Y. Choi, J. Y. Hwang, and D. H. Lee, "ID-based authenticated group key agreement secure against insider attacks," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E91-A, no. 7, pp. 1828–1830, 2008.

[27] T. Y. Wu and Y. M. Tseng, "Comments on an ID-based authenticated group key agreement protocol with withstanding insider attacks," *IEICE Transactions on Fundamentals*, vol. E92-A, no. 10, pp. 2638–2640, 2009.

[28] R. Housley, W. Polk, W. Ford, and D. Solo, "Internet X.509 public key infrastructure certificate and certificate revocation list (CRL) profile," RFC 3280, IETF, Anaheim, Calif, USA, 2002.

[29] T. ElGamal, "A public-key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469–472, 1985.

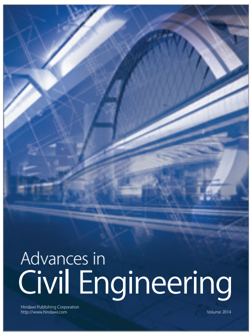
Journal of
Engineering

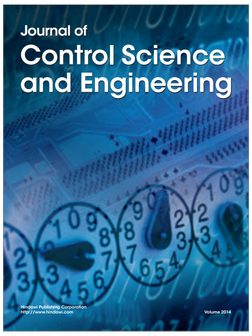
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

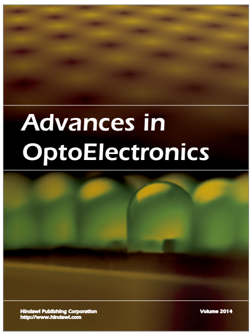
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

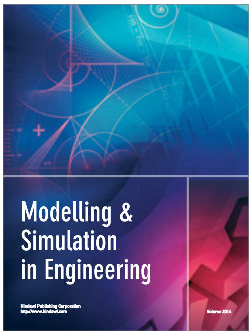
Journal of
Robotics

Journal of
Electrical and Computer Engineering

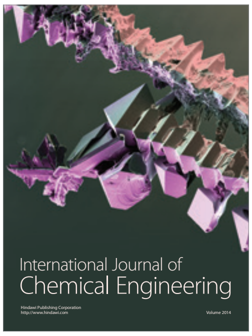

Advances in
OptoElectronics

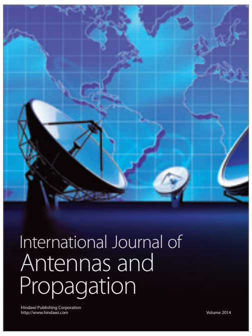
VLSI Design

International Journal of
Navigation and Observation

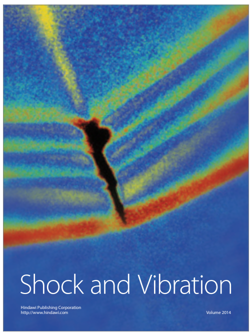

Modelling & Simulation in Engineering

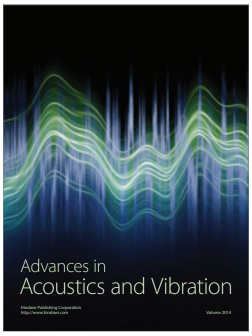
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration