Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2015, Article ID 898716, 12 pages
http://dx.doi.org/10.1155/2015/898716

*Research Article*

# A Novel ID-Based Authentication and Key Exchange Protocol Resistant to Ephemeral-Secret-Leakage Attacks for Mobile Devices

**Yuh-Min Tseng, Sen-Shan Huang, Tung-Tso Tsai, and Li Tseng**

*Department of Mathematics, National Changhua University of Education, Jin-De Campus,
Changhua City 500, Taiwan*

Correspondence should be addressed to Yuh-Min Tseng; ymtseng@cc.ncue.edu.tw

Received 21 July 2014; Revised 27 October 2014; Accepted 31 October 2014

Academic Editor: Georgios Kambourakis

With the rapid development in wireless communications and cloud computing technologies, clients (users) often use handheld mobile devices to access remote servers via open network channels. To provide authentication and confidentiality between clients and servers, a large number of ID-based authentication and key exchange (ID-AKE) protocols have been proposed for mobile client-server environments. However, most of the existing ID-AKE protocols adopt the precomputation technique so that they become vulnerable to the ephemeral-secret-leakage (ESL) attacks, in the sense that an adversary could use the ephemeral secrets to reveal the private keys of clients from the corresponding exchange messages. In the paper, we propose a new ESL-secure ID-AKE protocol for mobile client-server environments. We formally prove that the proposed protocol satisfies the security requirements of both mutual authentication and key exchange while resisting the ESL attacks. When compared with previously proposed ID-AKE protocols, our protocol has higher security and retains computational performance, since it requires no bilinear pairing operation for mobile clients. Finally, we mention the possibility of adopting our protocol as an authentication method of the extensible authentication protocol (EAP) for wireless networks.

## 1. Introduction

A key exchange protocol allows two parties to construct a session key. A symmetric encryption scheme with the session key is used to achieve confidentiality between the two parties. However, key exchange protocols without authentication are not secure against impersonation and intruder-in-the-middle attacks. An authentication and key exchange (AKE) protocol is a key exchange protocol under which the established session key of two parties remains secret to other parties. Several famous AKE protocols [1–3] based on the traditional public-key systems [4, 5] have been proposed to provide confidentiality and mutual authentication.

In 1985, Shamir [6] proposed an identity- (ID-) based public-key system that eliminates the certificate management in conventional public-key systems. In an ID-based public-key system, a user's public key is derived from the user's identity information, such as name, social security number, and e-mail address. However, Shamir's system is not practical. In 2001, Boneh and Franklin [7] followed Shamir's idea to propose a practical ID-based encryption scheme based on the Weil pairing. Afterwards, the ID-based cryptography received significant attention. A large number of ID-based cryptographic schemes and protocols have appeared in the literatures, such as ID-based encryption schemes [8–11], ID-based signature schemes [12–14], ID-based group key exchange protocols [15–18], and ID-based AKE protocols [19–23].

*1.1. Motivation.* With rapid growth of wireless communications, clients usually employ mobile devices (e.g., smart card) to obtain services from remote servers via open network channels. In a mobile client-server environment, these mobile devices are generally resource-constrained because they possess only low-power energy and limited computing capability. In this case, cryptographic operations with expensive computations would become heavy load for mobile devices. Hence, it is a critical issue to diminish the computational load of mobile devices in AKE or ID-based AKE protocols.

To overcome the resource-constrained situation on the client side, several AKE protocols [24–26] for mobile devices have been proposed for conventional public-key systems. Also, based on Boneh and Franklin's ID-based public-key setting, a number of ID-based AKE protocols for mobile devices [27–33] have been proposed to focus on the computation issue for mobile devices. These protocols above adopted an imbalanced computation technique to reduce the client's computational cost by shifting computational burden to a powerful server.

On the other hand, the offline precomputation technique is employed to lighten the online computational load of mobile devices. In the offline precomputation phase, ephemeral secrets (or random values) are required to generate some values in advance. In the meantime, the ephemeral secrets and these precomputed values are stored in the memory of mobile devices for the usage in the online phase. As a result, a new type of attacks would occur, called ephemeral-secret-leakage (ESL) attacks [34–36], in the sense that an adversary can reveal the private keys of clients from those precomputed values or the corresponding exchange messages if the ephemeral secrets are compromised. To our knowledge, the existing ID-based AKE protocols [27–33] did not address the ESL attacks at all. In the paper, we will construct an ID-based AKE protocol which is resistant to the ESL attacks under mobile client-server environments.

*1.2. Related Work.* In 2002, Smart [19] proposed the first ID-based AKE (ID-AKE) protocol based on the Weil pairing. However, Shim [20] pointed out that Smart's protocol does not offer forward secrecy. Shim also presented a new ID-AKE protocol with the optimal number of Weil pairing operations. Afterwards, several ID-AKE protocols [21–23] were proposed to improve performance and achieve more security properties. However, the protocols mentioned above require bilinear pairing operations on both ends and are consequently not suited for low-power computing devices.

In 2005, Choi et al. [27] proposed an ID-AKE protocol for mobile client-server environments. They adopted an imbalanced computation technique which shifts the client's computational burden to a powerful server. In 2010, Wu and Tseng [28, 29] also proposed two efficient ID-AKE protocols which do not require any bilinear pairing operations on the client side under mobile client-server environments. Their protocols are proved to be secure against ID attack, impersonation attack, and passive attack and offer mutual authentication, implicit key confirmation, and partial forward secrecy. In 2012, He [30] presented a new ID-AKE protocol to improve the performance of Wu and Tseng's protocol on client side. In addition, Islam and Biswas [31] and He et al. [32], independently, proposed ID-AKE protocols based on elliptic curve cryptography (ECC) without using bilinear pairings. Chuang and Tseng [33] proposed a generalized ID-based AKA protocol for mobile multiserver environments which is suitable for both general users with a long validity period and anonymous users with a short validity period. Chuang and Tseng's protocol is secure against all known attacks and provides forward secrecy. Indeed, all the mentioned ID-AKE protocols [27–33] adopted the precomputation technique to reduce the computational load of the mobile client. In such a case, as mentioned earlier, those ID-AKE protocols would be vulnerable to ESL attacks under mobile client-server environments.

In 2007, LaMacchia et al. [34] presented a strong security model for AKE protocols, which is concerned with the ESL attacks. They proposed a concrete AKE protocol resistant to ESL attacks. In their protocol, the leakage of ephemeral secrets would not damage the security of session keys and private keys of parties. In 2011, Ni et al. [35] proposed a strongly secure ID-AKE protocol which captures all basic security properties including the ESL resistance. Although Ni et al.'s protocol requires six bilinear pairing operations, one can employ the precomputation technique to reduce the computation cost if it knows the identity of the other party in communication beforehand. However, they did not address the scenarios for applications to mobile client-server environments. In 2014, Islam [36] also proposed a provably secure ID-AKE protocol resistant to ESL attack. Islam's protocol still requires two bilinear pairing operations for a client.

*1.3. Contribution and Organization.* In the paper, we propose a new ID-AKE protocol resistant to ESL attacks in mobile client-server environments, called ESL-secure ID-AKE protocol. In the proposed protocol, we also adopt the techniques of imbalanced computation and offline precomputation to reduce the computational cost required by a mobile client. Indeed, our protocol requires no bilinear pairing for mobile clients. Our protocol employs Tseng et al.'s ESL-secure signature scheme [37] to achieve the client-to-server authentication. Also, the offline precomputation is carried out prior to the execution of our protocol to achieve better performance. As compared with previously proposed protocols, our protocol is secure against the ESL attacks while retaining the computational performance. For security analysis, we first formalize the adversary's capabilities by redefining the adversarial model of ESL-secure ID-AKE protocols in mobile client-server environments. Under the computational Diffie-Hellman (CDH) assumption [7, 12], we demonstrate that our protocol is provably secure in the random oracle model [38, 39]. Finally, we discuss the relationship between our protocol and the extensible authentication protocol (EAP) for wireless networks [40–43]. It turns out that our ESL-secure ID-AKE protocol can be viewed as an authentication method of the EAP framework [40, 41].

The remainder of this paper is organized as follows. In Section 2, mathematical assumptions are presented. An adversarial model of the ESL-secure ID-AKE protocols is presented in Section 3. The proposed ESL-secure ID-AKE protocol is presented in Section 4. In Section 5, we give security analysis of the proposed protocol. Performance comparisons and discussions are given in Section 6. Finally, conclusions are drawn in Section 7.

## 2. Preliminaries

In this section, we compendiously introduce the concept of bilinear pairings, the related mathematical assumptions, and the notations used throughout this paper.

*2.1. Bilinear Pairings.* Let $G_1$ and $G_2$ be additive and multiplicative cyclic groups of large prime order $q$, respectively. A map $\hat{e} : G_1 \times G_1 \rightarrow G_2$ is called an admissible bilinear map if it satisfies the following three conditions:

(1) bilinearity: for all $P, Q \in G_1$ and $a, b \in Z_q^*$, we have $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$;

(2) nondegeneracy: for some $P, Q \in G_1$, $\hat{e}(P, Q) \neq 1$ holds;

(3) computability: given $P, Q \in G_1$, there is an efficient algorithm to compute $\hat{e}(P, Q)$.

Note that condition (1) implies that $\hat{e}(P + Q, R) = \hat{e}(P, R) \cdot \hat{e}(Q, R)$ and $\hat{e}(P, Q + R) = \hat{e}(P, Q) \cdot \hat{e}(P, R)$, for $P, Q, R \in G_1$. Full descriptions of groups, maps, and other parameters are discussed in [7, 12]. The relationship between security level and speed for pairing computations is presented in [29].

*2.2. Security Assumptions.* Let $G_1$, $G_2$, and $\hat{e}$ be defined as above. Here, we define a security assumption on which our scheme is based.

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

*Definition 1.* The CDH assumption in $G_1$ is defined as follows. Given $P, aP, bP \in G_1$ for unknown $a, b \in Z_q^*$, no probabilistic polynomial-time (PPT) adversary $\mathscr{A}$ can compute $abP$ with a nonnegligible probability. The successful probability (advantage) of the adversary $\mathscr{A}$ is presented as

$$\text{Adv}_{\mathscr{A}} = \Pr\left[\mathscr{A}(P, aP, bP) = abP \mid P \in G_1,\ a, b \in Z_q^*\right], \tag{1}$$

where the probability is measured over the random choices of $a, b \in Z_q^*$ consumed by $\mathscr{A}$.

*2.3. Notations.* For convenience, the system parameters, notations, and functions used throughout this paper are defined as follows:

$\hat{e}$: an admissible bilinear map from $G_1 \times G_1$ into $G_2$;

$P$: a generator of the group $G_1$;

$s$: the system private key $s$ randomly chosen from $Z_q^*$;

$P_{\text{pub}}$: the system public key defined by $P_{\text{pub}} = s \cdot P$;

ID: the identity of a client;

$D_{\text{ID}}$: the private key of the client ID;

$f_1(), f_2(), f_3(), f_4()$: one-way hash functions mapping from $\{0,1\}^*$ into $\{0,1\}^n$, where $n$ is a fixed length with $2^n < q$;

$H_1(), H_2()$: map-to-point hash functions mapping from $\{0,1\}^*$ into $G_1$.

## 3. Adversarial Model

Based on the security models in [34, 35], we present an adversarial model of ESL-secure ID-AKE protocols for mobile client-server environments. In 2007, LaMacchia et al. [34] presented a strong security model of AKE protocols, which addresses the ephemeral-secret-leakage (ESL) attacks. Based on LaMacchia et al.'s model, Ni et al. [35] defined the security model of strongly secure ID-AKE protocols (or named ESL-secure ID-AKE protocols) by adding the key extract query. Their model is a modification of LaMacchia et al.'s model altered from the conventional PKI-based setting to the ID-based setting.

In the following, we first describe an adversary's capabilities of ESL-secure ID-AKE protocols for mobile client-server environments. In our adversarial model, we assume that an adversary $\mathscr{A}$ is a probabilistic polynomial-time (PPT) algorithm and potentially control all communications by accessing to a set of oracles described below. In the following, we will denote the $k$th instance of the participant $U \in \{C, S\}$ by $\Pi_U^k$, where $C$ and $S$ indicate a client and the powerful server, respectively.

*Hash Queries* $(M)$. The oracle $\Pi_U^k$ keeps an initially empty list for each hash function. Upon receiving the hash query along with a message $M$, the same response is returned if the query has been asked before. Otherwise, the oracle $\Pi_U^k$ selects a random value $D$, records the pair $(M, D)$ in the list, and returns $D$ to the adversary $\mathscr{A}$.

(i) *Extract* (ID): upon receiving such a query, the oracle $\Pi_U^k$ computes the private key $D_{\text{ID}}$ associated with ID and returns it to the adversary $\mathscr{A}$. This query models *ID attacks*.

(ii) *Send* $(\Pi_U^k, M)$: upon receiving such a query, the oracle $\Pi_U^k$ executes the protocol according to $M$ and responds the corresponding results to the adversary $\mathscr{A}$. This query models *passive attacks*.

(iii) *Reveal* $(\Pi_U^k)$: upon receiving such a query, the oracle $\Pi_U^k$ outputs the corresponding session key SK if the oracle has accepted the session; otherwise, it returns a null value. This query addresses the *known-session-key security*, in the sense that a compromised session key should not endanger other session keys.

(iv) *Ephemeral-secret-leakage* $(\Pi_C^k)$: this query models *ephemeral-secret-leakage attacks*. When the adversary $\mathscr{A}$ issues this query, the oracle $\Pi_C^k$ returns the used ephemeral secret values (or random values) in the corresponding session. Note that, in our adversarial model, $\mathscr{A}$ is forbidden to issue this type of query on the server $S$.

(v) *Corrupt* $(\Pi_C^k)$: this query models *partial forward secrecy*. The adversary $\mathscr{A}$ can issue such a query on a client $C$ to obtain the private key of $C$. Therefore, a compromised private key should not endanger any previous session key between the client and the server. Here, as in [34, 35], the adversary $\mathscr{A}$ can issue

*Ephemeral-secret-leakage* query or *Corrupt* query, but not both.

(vi) *Test* ($\Pi_U^k$): when the adversary $\mathscr{A}$ sends such a query, the oracle flips an unbiased coin $b$. If $b = 1$, then the oracle $\Pi_U^k$ returns the session key SK; otherwise, it returns a random value. $\mathscr{A}$ is allowed to issue such a query only once to the oracle $\Pi_U^k$.

Here, we present the adversarial model of ESL-secure ID-AKE protocols. The reader is referred to [34–36] for detailed descriptions.

*Definition 2* (partnership). One says that $\Pi_C^k$ and $\Pi_S^t$ are partners if they can authenticate mutually and accept a common session key.

*Definition 3* (freshness). An oracle $\Pi_C^k$ with partner $\Pi_S^t$ is fresh if the following conditions hold:

(1) $\Pi_C^k$ and $\Pi_S^t$ accept a session key SK $\neq$ NULL while both of them are not requested by *Reveal* query;

(2) no *Corrupt* query can be issued before the query *Send* ($\Pi_C^k, M$) or query *Send* ($\Pi_S^t, M$) is asked.

*Definition 4* (ESL-secure ID-AKE security). An ESL-secure ID-AKE protocol for mobile multiserver environments offers existential unforgeability and possesses the secrecy of session key against adaptive chosen ID attacks if no PPT adversary $\mathscr{A}$ has a nonnegligible advantage in the following game played between $\mathscr{A}$ and a set of oracles $\Pi_U^k$, where $U \in \{C, S\}$.

(1) The adversary $\mathscr{A}$ may ask a finite number of various queries and obtain responses from the corresponding oracles.

(2) Every user is assigned a private key via the key extract phase after the system setup phase accomplishes.

(3) No *Reveal* ($\Pi_U^k$) or *Corrupt* ($\Pi_U^k$) can be issued before the *Test* ($\Pi_U^k$) is asked.

(4) The adversary $\mathscr{A}$ can issue *Ephemeral-secret-leakage* ($\Pi_C^k$) or *Corrupt* ($\Pi_C^k$), but not both.

(5) The adversary $\mathscr{A}$ may adaptively make further queries before *Test* ($\Pi_U^k$), where $\Pi_U^k$ must be fresh. Finally, $\mathscr{A}$ outputs its guess for the bit $b$ which has been previously chosen in the *Test* ($\Pi_U^k$).

*Definition 5* (advantage). Let Succ denote the event that $\mathscr{A}$ correctly guesses the bit $b$ chosen in the *Test* query. If $\mathscr{A}$ asks a *Test* ($\Pi_U^k$) and guesses the bit $b$, the successful advantage (probability) of $\mathscr{A}$ in attacking the ESL-secure ID-AKE protocol $\mathscr{P}$ is defined as $\mathrm{Adv}_{\mathscr{P}}(\mathscr{A}) = |2 \cdot \Pr[\mathrm{Succ}] - 1|$. One says that the ESL-secure ID-AKE protocol $\mathscr{P}$ is secure if $\mathrm{Adv}_{\mathscr{P}}(\mathscr{A})$ is negligible.

*Definition 6* (partial forward secrecy). An ESL-secure ID-AKE protocol $\mathscr{P}$ provides partial forward secrecy if any adversary $\mathscr{A}$ with client $C$'s private key cannot compromise previous session keys between $C$ and the server.

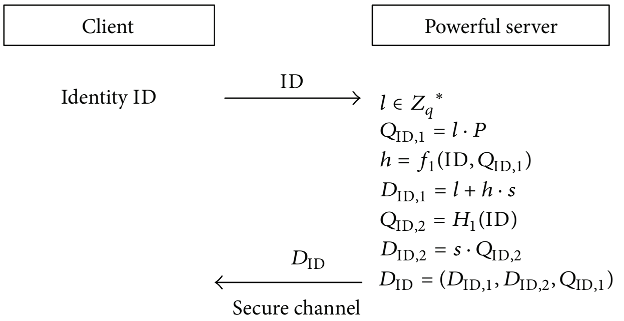

Client
Powerful server
Identity ID
ID
$l \in Z_q^*$
$Q_{\mathrm{ID},1} = l \cdot P$
$h = f_1(\mathrm{ID}, Q_{\mathrm{ID},1})$
$D_{\mathrm{ID},1} = l + h \cdot s$
$Q_{\mathrm{ID},2} = H_1(\mathrm{ID})$
$D_{\mathrm{ID},2} = s \cdot Q_{\mathrm{ID},2}$
$D_{\mathrm{ID}}$
$D_{\mathrm{ID}} = (D_{\mathrm{ID},1}, D_{\mathrm{ID},2}, Q_{\mathrm{ID},1})$
Secure channel

Figure 1: The key extract phase.

*Definition 7* (implicit key authentication). An ESL-secure ID-AKE protocol provides implicit key authentication if every client is assured that no other clients can learn its session keys with the server.

## 4. Our Protocol

In this section, we present our concrete ESL-secure ID-AKE protocol for mobile client-server environments. Our protocol consists of three phases, namely, the system setup phase, the key extract phase, and the mutual authentication and key agreement phase.

*4.1. System Setup Phase.* Our system consists of a powerful server $S$ and some mobile clients. These clients refer to users with handheld devices. A client has access to the server through open channels, such as the Internet or wireless networks. The powerful server $S$ is responsible for generating and distributing private keys to clients while providing services or applications. The server $S$ is also responsible for generating the system parameters.

In the phase, the server $S$ first generates two cyclic groups $G_1$ and $G_2$ of a large prime order $q$, an admissible bilinear map $\hat{e} : G_1 \times G_1 \rightarrow G_2$, and a random generator $P$ of $G_1$, where $G_1$ and $G_2$ are additive and multiplicative groups, respectively. The server $S$ then performs the following tasks:

(1) randomly select a system private key $s \in Z_q^*$;

(2) compute the system public key $P_{\mathrm{pub}} = s \cdot P$;

(3) choose six cryptographic hash functions $H_1, H_2 : \{0,1\}^* \rightarrow G_1$ and $f_1, f_2, f_3, f_4 : \{0,1\}^* \rightarrow \{0,1\}^n$, where $n$ is a fixed length with $2^n < q$;

(4) publish public parameters and functions as

$$\mathrm{Params} = \left\langle G_1, G_2, q, P, \hat{e}, P_{\mathrm{pub}}, H_1, H_2, f_1, f_2, f_3, f_4 \right\rangle. \quad (2)$$

*4.2. Key Extract Phase.* In the key extract phase, a client submits its identity ID to the server $S$ and receives the corresponding private key $D_{\mathrm{ID}}$. The key extract phase is depicted in Figure 1. We present the detailed procedures as follows.

(1) The client submits its identity ID to the server $S$.

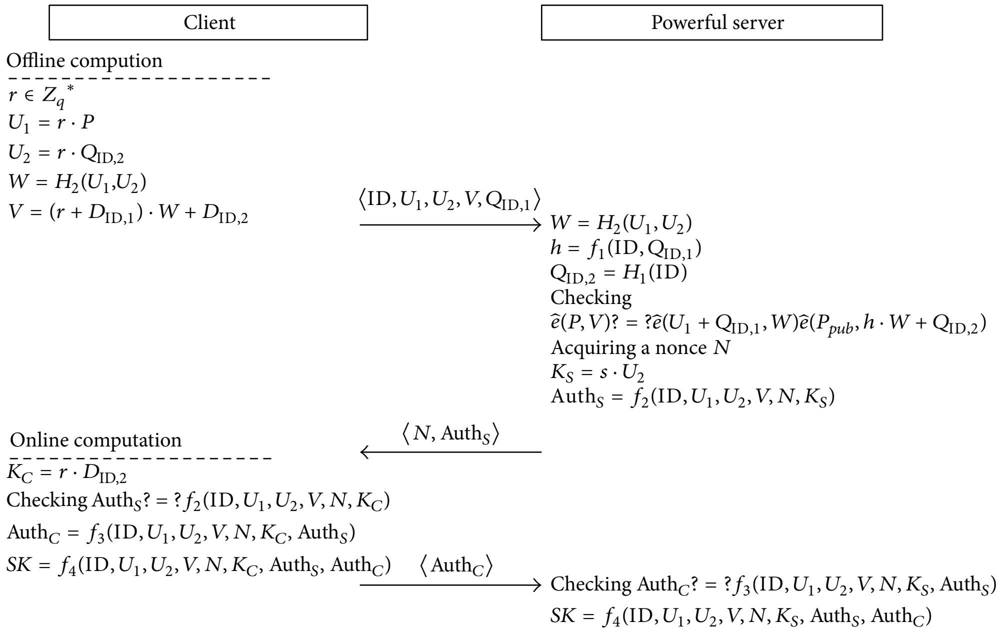

Client
Powerful server
Offline compution
$r \in Z_q{}^*$
$U_1 = r \cdot P$
$U_2 = r \cdot Q_{\mathrm{ID},2}$
$W = H_2(U_1,U_2)$
$V = (r + D_{\mathrm{ID},1}) \cdot W + D_{\mathrm{ID},2}$
$\langle \mathrm{ID}, U_1, U_2, V, Q_{\mathrm{ID},1} \rangle$
$W = H_2(U_1, U_2)$
$h = f_1(\mathrm{ID}, Q_{\mathrm{ID},1})$
$Q_{\mathrm{ID},2} = H_1(\mathrm{ID})$
Checking
$\hat{e}(P, V)? = ?\hat{e}(U_1 + Q_{\mathrm{ID},1}, W)\hat{e}(P_{pub}, h \cdot W + Q_{\mathrm{ID},2})$
Acquiring a nonce $N$
$K_S = s \cdot U_2$
$\mathrm{Auth}_S = f_2(\mathrm{ID}, U_1, U_2, V, N, K_S)$
$\langle N, \mathrm{Auth}_S \rangle$
Online computation
$K_C = r \cdot D_{\mathrm{ID},2}$
Checking $\mathrm{Auth}_S? = ?f_2(\mathrm{ID}, U_1, U_2, V, N, K_C)$
$\mathrm{Auth}_C = f_3(\mathrm{ID}, U_1, U_2, V, N, K_C, \mathrm{Auth}_S)$
$SK = f_4(\mathrm{ID}, U_1, U_2, V, N, K_C, \mathrm{Auth}_S, \mathrm{Auth}_C)$
$\langle \mathrm{Auth}_C \rangle$
Checking $\mathrm{Auth}_C? = ?f_3(\mathrm{ID}, U_1, U_2, V, N, K_S, \mathrm{Auth}_S)$
$SK = f_4(\mathrm{ID}, U_1, U_2, V, N, K_S, \mathrm{Auth}_S, \mathrm{Auth}_C)$

FIGURE 2: The mutual authentication and key exchange phase.

(2) Upon receiving a client's identity ID, the server chooses an ephemeral secret value $l \in Z_q^*$ and computes $Q_{\mathrm{ID},1} = l \cdot P$, $h = f_1(\mathrm{ID}, Q_{\mathrm{ID},1})$, $D_{\mathrm{ID},1} = l + h \cdot s$, $Q_{\mathrm{ID},2} = H_1(\mathrm{ID})$, and $D_{\mathrm{ID},2} = s \cdot Q_{\mathrm{ID},2}$.

(3) Set $D_{\mathrm{ID}} = (D_{\mathrm{ID},1}, D_{\mathrm{ID},2}, Q_{\mathrm{ID},1})$ and send it to the client via a secure channel.

*4.3. Mutual Authentication and Key Exchange Phase.* Suppose that a client with identity ID would like to communicate with the powerful server $S$ and to access services of the server. As depicted in Figure 2, the detailed interactions between the client and the server are presented as below.

(1) The client with identity ID performs offline computations in advance.

  (a) Random select an ephemeral secret $r \in Z_q^*$.
  (b) Compute $U_1 = r \cdot P$ and $U_2 = r \cdot Q_{\mathrm{ID},2}$.
  (c) Compute $W = H_2(U_1, U_2)$ and $V = (r + D_{\mathrm{ID},1}) \cdot W + D_{\mathrm{ID},2}$.
  (d) Send $\langle \mathrm{ID}, U_1, U_2, V, Q_{\mathrm{ID},1} \rangle$ to the server.

(2) Upon receiving $\langle \mathrm{ID}, U_1, U_2, V, Q_{\mathrm{ID},1} \rangle$, the server $S$ performs the following tasks.

  (a) Compute $W = H_2(U_1, U_2)$, $h = f_1(\mathrm{ID}, Q_{\mathrm{ID},1})$, and $Q_{\mathrm{ID},2} = H_1(\mathrm{ID})$.
  (b) Check whether the equality $\hat{e}(P, V) = \hat{e}(U_1 + Q_{\mathrm{ID},1}, W) \cdot \hat{e}(P_{\mathrm{pub}}, h \cdot W + Q_{\mathrm{ID},2})$ holds or not. If so, the server $S$ accepts the request. Otherwise, the server terminates the process.
  (c) Acquire a nonce $N$.
  (d) Compute $K_S = s \cdot U_2$ and $\mathrm{Auth}_S = f_2(\mathrm{ID}, U_1, U_2, V, N, K_S)$.
  (e) Finally, send $\langle N, \mathrm{Auth}_S \rangle$ to the client.

(3) Upon receiving $\langle N, \mathrm{Auth}_S \rangle$, the client authenticates the server $S$ by performing the following tasks.

  (a) Compute $K_C = r \cdot D_{\mathrm{ID},2}$.
  (b) Check whether the equality $\mathrm{Auth}_S = f_2(\mathrm{ID}, U_1, U_2, V, N, K_C)$ holds or not. If so, the client accepts the server $S$.
  (c) Compute $\mathrm{Auth}_C = f_3(\mathrm{ID}, U_1, U_2, V, N, K_C, \mathrm{Auth}_S)$.
  (d) Compute a session key $\mathrm{SK} = f_4(\mathrm{ID}, U_1, U_2, V, N, K_C, \mathrm{Auth}_S, \mathrm{Auth}_C)$.
  (e) The client sends $\langle \mathrm{Auth}_C \rangle$ to the server $S$.

(4) Upon receiving $\langle \mathrm{Auth}_C \rangle$, the server $S$ authenticates and establishes a session key by performing the following tasks.

  (a) Check whether the equality $\mathrm{Auth}_C = f_3(\mathrm{ID}, U_1, U_2, V, N, K_S, \mathrm{Auth}_S)$ holds or not. If so, the server accepts the client.
  (b) Compute a session key $\mathrm{SK} = f_4(\mathrm{ID}, U_1, U_2, V, N, K_S, \mathrm{Auth}_S, \mathrm{Auth}_C)$.

In the following, we present the correctness of the equality in Step (2)(b):

$$\begin{aligned}
\hat{e}(P, V) &= \hat{e}\left(P, \left(r + D_{\mathrm{ID},1}\right) \cdot W + D_{\mathrm{ID},2}\right) \\
&= \hat{e}\left(P, \left(r + l + h \cdot s\right) \cdot W + s \cdot Q_{\mathrm{ID},2}\right) \\
&= \hat{e}\left(P, \left(r + l\right) \cdot W + h \cdot s \cdot W + s \cdot Q_{\mathrm{ID},2}\right) \\
&= \hat{e}\left(P, \left(r + l\right) \cdot W + s \cdot \left(h \cdot W + Q_{\mathrm{ID},2}\right)\right)
\end{aligned}$$

$$= \hat{e}(P,(r+l)\cdot W)\cdot \hat{e}(P, s\cdot (h\cdot W + Q_{\mathrm{ID},2}))$$
$$= \hat{e}((r+l)\cdot P, W)\cdot \hat{e}(s\cdot P, h\cdot W + Q_{\mathrm{ID},2})$$
$$= \hat{e}(U_1 + Q_{\mathrm{ID},1}, W)\cdot \hat{e}\left(P_{\mathrm{pub}}, h\cdot W + Q_{\mathrm{ID},2}\right). \tag{3}$$

On the other hand, we have $K_S = K_C$ since $K_S = s\cdot U_2 = s\cdot r\cdot Q_{\mathrm{ID},2} = r\cdot D_{\mathrm{ID},2} = K_C$. And, in this case, we say that $\mathrm{Auth}_S$ and $\mathrm{Auth}_C$ are valid, and the client and the server have established a common session key SK.

## 5. Security Analysis

In this section, we present the security analysis of our proposed protocol in the random oracle model [38, 39]. In the following, five theorems are given to prove that the proposed protocol achieves the security requirements of ESL-secure ID-AKE protocols for mobile client-server environments. These security requirements include client-to-server authentication, server-to-client authentication, key agreement, implicit key confirmation, and partial forward secrecy. In Theorems 8 and 10, we show that the proposed protocol provides the client-to-server and server-to-client authentications under ID, impersonation, and ephemeral-secret-leakage attacks, respectively. Hence, the proposed protocol offers mutual authentication. In Theorem 9, we will show that the proposed protocol provides secure key agreement under known-session-key attacks. Furthermore, the implicit key confirmation and partial forward secrecy are achieved by Theorems 11 and 12, respectively.

*5.1. Client-to-Server Authentication.* First, we prove that an adversary cannot impersonate a legitimate client to communicate with the server under the CDH assumption. We establish this by the methods similar to those in [24, 28], in which a *Simulation* $\mathscr{B}$ is employed to simulate all the queries and oracles which occurred in our proposed protocol. In the following, we use the notations $\Pi_S^i$ and $\Pi_C^j$ to indicate the $i$th instance of the server $S$ and the $j$th instance of a client $C$, respectively.

**Theorem 8.** *Assume that a probabilistic polynomial-time (PPT) adversary $\mathscr{A}$ can violate the client-to-server authentication with a nonnegligible advantage by making at most $q_S$, $q_C$, $q_{H1}$, $q_{H2}$, and $q_k$ queries, for $k = 1,\dots,4$, respectively, to the $\Pi_S^i$ oracle of the server S and the $\Pi_C^j$ oracle of the client C, $H_1$, $H_2$, and $f_k$, for $k = 1,\dots,4$. Then there is a challenger $\mathscr{B}$ that can solve the CDH problem with a nonnegligible probability.*

*Proof.* We assume that there is a probabilistic polynomial-time algorithm $\mathscr{A}$ with an advantage $\varepsilon_0$ within time $t_0$ to perform adaptive chosen message attacks, ID attacks, and ephemeral-secret-leakage attacks to our proposed protocol. By Lemma 1 in [12], $\mathscr{A}$ can break the protocol with an advantage $\varepsilon \le \varepsilon_0(1-1/q)/q_{H1}$ within running time $t \le t_0$ under adaptive chosen message, ephemeral-secret-leakage, and fixed-ID attacks. Without loss of generality, we set $\mathrm{ID}_T$ as the fixed target identity. If the oracle $\Pi_S^i$ of the server $S$ accepts with no partner, it means that $\mathscr{A}$ has successfully impersonated the client $C$ to the server $S$ and violated the client-to-server authentication.

Next, we would like to construct an algorithm $\mathscr{B}$ to solve the CDH problem by appealing to $\mathscr{A}$. Namely, upon receiving a random instance $(P, aP, bP)$ in $G_1$ with unknown $a, b \in Z_q^*$, the algorithm $\mathscr{B}$ is able to derive $abP$ by interacting with $\mathscr{A}$. Here, we will adopt the methods similar to those in [24, 28] and Tseng et al.'s ESL-secure signature scheme [37] to achieve the client-to-server authentication. To simulate the actual situations, we employ the algorithm $\mathscr{B}$ (called Simulation $\mathscr{B}$) to make the initialization and respond to $\mathscr{A}$ according to our protocol.

(i) Initialization: at first, Simulation $\mathscr{B}$ generates the system parameters $\langle G_1, G_2, q, P, \hat{e}\rangle$ and sets the system public key $P_{\mathrm{pub}} = s\cdot P$, where $s$ is the system private key. Simulation $\mathscr{B}$ then sends the public parameters to $\mathscr{A}$. Simulation $\mathscr{B}$ maintains the lists $L_{H1}$, $L_{H2}$, and $L_k$, $k = 1,\dots,4$, to respond consistently without collision to the hash queries $H_1$, $H_2$, and $f_k$, $k = 1,\dots,4$. These lists are initially empty.

(ii) $f_1(\mathrm{ID}, Q_{\mathrm{ID},1})$: upon receiving such a query, the same response is given if the query has been asked before. Otherwise, $\mathscr{B}$ randomly selects a value $r_1 \in \{0,1\}^n$, records the tuple $(\mathrm{ID}, Q_{\mathrm{ID},1}, r_1)$ in the list $L_1$, and returns $r_1$ to $\mathscr{A}$.

(iii) $f_2(\mathrm{ID}, U_1, U_2, V, N, K)$: upon receiving such a query, the same response is given if the query has been asked before. Otherwise, $\mathscr{B}$ randomly selects a value $r_2 \in \{0,1\}^n$, records the tuple $(\mathrm{ID}, U_1, U_2, V, N, K, r_2)$ in the list $L_2$, and returns $r_2$ to $\mathscr{A}$.

(iv) $f_3(\mathrm{ID}, U_1, U_2, V, N, K, \mathrm{Auth}_S)$: upon receiving such a query, the same response is given if the query has been asked before. Otherwise, $\mathscr{B}$ randomly selects a value $r_3 \in \{0,1\}^n$, records the tuple $(\mathrm{ID}, U_1, U_2, V, N, K, \mathrm{Auth}_S, r_3)$ in the list $L_3$, and returns $r_3$ to $\mathscr{A}$.

(v) $f_4(\mathrm{ID}, U_1, U_2, V, N, K, \mathrm{Auth}_S, \mathrm{Auth}_C)$: upon receiving such a query, the same response is given if the query has been asked before. Otherwise, $\mathscr{B}$ randomly selects a value $r_4 \in \{0,1\}^n$, records the tuple $(\mathrm{ID}, U_1, U_2, V, N, K, \mathrm{Auth}_S, \mathrm{Auth}_C, r_4)$ in $L_4$, and returns $r_4$ to $\mathscr{A}$.

(vi) $H_1(\mathrm{ID})$: upon receiving such a query, the same response is given if the query has been asked before. Otherwise, $\mathscr{B}$ selects a random value $w \in Z_q^*$ and sets $Q_{\mathrm{ID},2} = w\cdot P - bP$ if $\mathrm{ID} = \mathrm{ID}_T$; $Q_{\mathrm{ID},2} = w\cdot P$, otherwise. Simulation $\mathscr{B}$ records the tuple $(\mathrm{ID}, w, Q_{\mathrm{ID},2})$ in $L_{H1}$ and returns $Q_{\mathrm{ID},2}$ to $\mathscr{A}$.

(vii) $H_2(U_1, U_2)$: upon receiving such a query, the same response is given if the query has been asked before. Otherwise, $\mathscr{B}$ selects a random value $u \in Z_q^*$, records the tuple $(U_1, U_2, W = u\cdot bP)$ in $L_{H2}$, and returns $W$ to $\mathscr{A}$.

(viii) *Extract* (ID); upon receiving such a query and $\mathrm{ID} \neq \mathrm{ID}_T$, $\mathscr{B}$ accesses to the corresponding tuples $(\mathrm{ID}, Q_{\mathrm{ID},1}, r_1)$ and $(\mathrm{ID}, w, Q_{\mathrm{ID},2})$ in the lists $L_{f_1}$ and $L_{H_1}$, respectively. And then, Simulation $\mathscr{B}$ chooses a random value $v$ and returns the private key $D_{\mathrm{ID}} = (D_{\mathrm{ID},1} = v, D_{\mathrm{ID},2} = w \cdot P_{\mathrm{pub}}, Q_{\mathrm{ID},1} = v \cdot P - r_1 \cdot P_{\mathrm{pub}})$ to $\mathscr{A}$. If $\mathrm{ID} = \mathrm{ID}_T$, $\mathscr{B}$ aborts.

(ix) *Ephemeral-secret-leakage* $(\Pi_C^j)$: when $\mathscr{A}$ issues this query, $\mathscr{B}$ returns the ephemeral secret value $r$ adopted in the corresponding session. This query models ephemeral-secret-leakage attacks. Note that $\mathscr{A}$ needs not to issue this query to the server $S$ since, in our protocol, no ephemeral secret value is used on the server side.

(x) *Send queries:* there are four cases.

(1) When $\mathscr{A}$ issues a *Send* $(\Pi_C^j$, "start"), $\mathscr{B}$ uses the signing query in Tseng et al.'s ESL-secure signature scheme [37] to generate a valid signature $\langle \mathrm{ID}, U_1, U_2, V, Q_{\mathrm{ID},1}\rangle$ for $\mathscr{A}$.

(2) When $\mathscr{A}$ issues a *Send* $(\Pi_S^i, \langle \mathrm{ID}, U_1, U_2, V, Q_{\mathrm{ID},1}\rangle)$ and $\mathrm{ID} \neq \mathrm{ID}_T$, $\mathscr{B}$ computes $W = H_2(U_1, U_2)$, $h = f_1(\mathrm{ID}, Q_{\mathrm{ID},1})$, and $Q_{\mathrm{ID},2} = H_1(\mathrm{ID})$ and checks whether the equality $\widehat{e}(P, V) = \widehat{e}(U_1 + Q_{\mathrm{ID},1}, W) \cdot \widehat{e}(P_{\mathrm{pub}}, h \cdot W + Q_{\mathrm{ID},2})$ holds. If the equality holds, $\mathscr{B}$ accepts the request, acquires a nonce $N$, and computes $K_S = s \cdot U_2 = s \cdot r \cdot w \cdot P = r \cdot w \cdot P_{\mathrm{pub}}$ and $\mathrm{Auth}_S = f_2(\mathrm{ID}, U_1, U_2, V, N, K_S)$. Finally, $\mathscr{B}$ returns $\langle N, \mathrm{Auth}_S\rangle$ to $\mathscr{A}$. Otherwise, $\mathscr{B}$ declines the request. On the other hand, if $\mathrm{ID} = \mathrm{ID}_T$, $\mathscr{B}$ acquires a nonce $N$, selects a random value $\mathrm{Auth}_S \in \{0,1\}^n$, and returns $\langle N, \mathrm{Auth}_S\rangle$ to $\mathscr{A}$. In this case, $\mathscr{A}$ is unable to verify the validity of $\langle N, \mathrm{Auth}_S\rangle$ due to the lack of $D_{\mathrm{ID},2}$ and $K_C = r \cdot D_{\mathrm{ID},2}$.

(3) Upon receiving the *Send* $(\Pi_C^j, \langle N, \mathrm{Auth}_S\rangle)$ with $C$'s identity ID distinct from $\mathrm{ID}_T$, $\mathscr{B}$ computes $K_C = r \cdot w \cdot P_{\mathrm{pub}} = r \cdot D_{\mathrm{ID},2}$ and checks whether the equality $\mathrm{Auth}_S = f_2(\mathrm{ID}, U_1, U_2, V, N, K_C)$ holds. If the equality holds, the oracle $\Pi_C^j$ accepts the session. Then $\mathscr{B}$ computes $\mathrm{Auth}_C = f_3(\mathrm{ID}, U_1, U_2, V, N, K_C, \mathrm{Auth}_S)$ and returns $\langle \mathrm{Auth}_C\rangle$ to $\mathscr{A}$. Otherwise, $\mathscr{B}$ declines the request. On the other hand, if $\mathrm{ID} = \mathrm{ID}_T$, $\mathscr{B}$ selects a random value $\mathrm{Auth}_C \in \{0,1\}^n$ and returns $\langle \mathrm{Auth}_C\rangle$ to $\mathscr{A}$. In this case, $\mathscr{A}$ is unable to verify the validity of $\langle \mathrm{Auth}_C\rangle$ due to the lack of $D_{\mathrm{ID},2}$ and $K_C = r \cdot D_{\mathrm{ID},2}$.

(4) When $\mathscr{A}$ makes a *Send* $(\Pi_S^i, \langle \mathrm{Auth}_C\rangle)$, $\mathscr{B}$ checks whether $\mathrm{Auth}_C = f_3(\mathrm{ID}, U_1, U_2, V, N, K_S, \mathrm{Auth}_S)$ holds. If so, the oracle $\Pi_S^i$ accepts the session and terminates. Otherwise, the oracle $\Pi_S^i$ also terminates while not accepting.

By the responses to those queries above, $\mathscr{B}$ is perfectly indistinguishable from the proposed protocol. If $\mathscr{A}$ could violate the client-to-server authentication with a nonnegligible advantage, it would be required to send two valid messages $\langle \mathrm{ID}, U_1, U_2, V, Q_{\mathrm{ID},1}\rangle$ and $\langle \mathrm{Auth}_C\rangle$ to the oracle $\Pi_S^i$. In such a case, since $\langle \mathrm{ID}, U_1, U_2, V, Q_{\mathrm{ID},1}\rangle$ is valid, it must satisfy the equality $\widehat{e}(P, V) = \widehat{e}(U_1 + Q_{\mathrm{ID},1}, W) \cdot \widehat{e}(P_{\mathrm{pub}}, h \cdot W + Q_{\mathrm{ID},2})$, where $W = H_2(U_1, U_2)$, $h = f_1(\mathrm{ID}, Q_{\mathrm{ID},1})$, and $Q_{\mathrm{ID},2} = H_1(\mathrm{ID})$. We also know that $\langle \mathrm{ID}, U_1, U_2, V, Q_{\mathrm{ID},1}\rangle$ can be viewed as a signature on the message $U_2$ as in Tseng et al.'s ESL-secure signature scheme [37]. Hence, if $\mathscr{A}$ can violate the client-to-server authentication, $\mathscr{B}$ can solve the CDH problem with a nonnegligible advantage by adopting the same approach in [37]. On the other hand, to generate a valid message $\langle \mathrm{Auth}_C\rangle$, $\mathscr{A}$ must obtain $D_{\mathrm{ID},2}$ since $\langle \mathrm{Auth}_C\rangle$ is derived from $K_C$ which is exactly $r \cdot D_{\mathrm{ID},2}$ (here, we assume that $\mathscr{A}$ can obtain the ephemeral secret value $r$ via ESL attacks). This enables $\mathscr{B}$ to resolve the CDH problem $(P, aP, bP)$ in $G_1$ with unknown $a, b \in Z_q^*$, namely, to evaluate $abP$ by computing $w \cdot aP - D_{\mathrm{ID},2}$ since $w$ is a known value in the list $L_{H1}$ and $D_{\mathrm{ID},2} = a \cdot Q_{\mathrm{ID},2}$ due to the tuple $\langle P, P_{\mathrm{pub}} = aP, Q_{\mathrm{ID},2} = w \cdot P - bP\rangle$. Therefore, the proposed protocol is secure against adaptive chosen message attacks, ID attacks, and ephemeral-secret-leakage attacks and provides the client-to-server authentication. □

*5.2. Key Agreement.* In the following, we prove that the proposed protocol provides key agreement under the CDH assumption. Simulation $\mathscr{B}$ is used to simulate the actual situation in our protocol.

**Theorem 9.** *Assume that a PPT adversary $\mathscr{A}$ can guess the value b correctly involved in the Test query with a nonnegligible advantage by making at most $q_S$, $q_C$, $q_{H1}$, $q_{H2}$, and $q_k$ queries, for $k = 1, \ldots, 4$, respectively, to the $\Pi_S^i$ oracle of the server S and the $\Pi_C^j$ oracle of the client C, $H_1$, $H_2$, and $f_k$ for $k = 1, \ldots, 4$. Then there is a challenger $\mathscr{B}$ that can solve the CDH problem with a nonnegligible advantage.*

*Proof.* Firstly, we know that the adversary $\mathscr{A}$ can guess the unbiased coin $b$ correctly with the probability 1/2 in the *Test* query. Let the symbol Osk denote the event that $\mathscr{A}$ obtains the correct session key. Assume that $\mathscr{A}$ can guess the value $b$ correctly with a nonnegligible advantage $\varepsilon$. Hence, $\mathscr{A}$ obtains the correct session key with the advantage $\Pr[\mathrm{Osk}] \geq \varepsilon/2$.

Without loss of generality, we denote by the symbols Test $(\Pi_C^j)$ and Test $(\Pi_S^i)$, respectively, the successful events of obtaining the correct session key in the *Test* query to the client and the server. Note that $\mathscr{A}$ can issue the *Test* query to the client or the server. Because the client actively connects to the server, we have the inequality

$$\begin{aligned} &\Pr\left[\mathrm{Osk} \wedge \mathrm{Test}\left(\Pi_S^i\right) \wedge \mathrm{Event}^{C2S}\right] \\ &\quad + \Pr\left[\mathrm{Osk} \wedge \mathrm{Test}\left(\Pi_S^i\right) \wedge \neg\mathrm{Event}^{C2S}\right] \\ &\quad + \Pr\left[\mathrm{Osk} \wedge \mathrm{Test}\left(\Pi_C^j\right)\right] \geq \frac{\varepsilon}{2}, \end{aligned} \tag{4}$$

where the symbol $\text{Event}^{C2S}$ is the event violating the client-to-server authentication. Also, let $\text{Pr}_{C2S}$ denote the probability of the event $\text{Event}^{C2S}$. Then, we have

$$\begin{aligned}&\text{Pr}\left[\text{Osk} \wedge \text{Test}\left(\Pi_S^i\right) \wedge \neg\text{Event}^{C2S}\right] \\ &\quad + \text{Pr}\left[\text{Osk} \wedge \text{Test}\left(\Pi_C^j\right)\right] \geq \frac{\varepsilon}{2} - \text{Pr}_{C2S},\end{aligned} \tag{5}$$

for some instances $i$ and $j$ of the server $S$ and the client $C$, respectively.

In the following simulation, we employ the algorithm $\mathscr{B}$ (called Simulation $\mathscr{B}$) to make the initialization and to respond to $\mathscr{A}$ according to our proposed protocol. Without loss of generality, we set $\text{ID}_T$ as the fixed target identity.

(i) Initialization: at first, $\mathscr{B}$ generates the system parameters $\langle G_1, G_2, q, P, \hat{e}\rangle$ and sets the system public key $P_{\text{pub}} = s \cdot P$, where $s$ is the system private key. Then $\mathscr{B}$ sends the public parameters to $\mathscr{A}$ and maintains the lists $L_{H1}, L_{H2}, L_k$, $k = 1, \ldots, 4$, to respond consistently without collision to the hash queries $H_1$, $H_2$, and $f_k$, $k = 1, \ldots, 4$, respectively. These lists are initially empty.

(ii) *Extract* (ID): it is the same as in the proof of Theorem 8.

(iii) $H_1, H_2, f_k$ *queries* (for $k = 1, \ldots, 4$): they are the same as in the proof of Theorem 8.

(iv) *Reveal* ($\Pi_U^k$): on receiving such a query, Simulation $\mathscr{B}$ returns the associated session key SK if the corresponding oracle accepts the session; otherwise, it returns a null value. This query addresses the known-key security in the sense that a compromised session key should not endanger other session keys.

(v) *Ephemeral-secret-leakage* ($\Pi_C^j$): this query models ephemeral-secret-leakage attacks. When $\mathscr{A}$ issues this query, $\mathscr{B}$ returns the ephemeral secret value $r$ adopted in the corresponding session.

(vi) *Corrupt* ($\Pi_C^j$): this query models partial forward secrecy. The adversary $\mathscr{A}$ can issue this query to a client $C$ to obtain its private key. Therefore, a compromised private key should not endanger previous session keys between the client and the server. Note that, as in [34, 35], $\mathscr{A}$ can issue either *Ephemeral-secret-leakage* query or *Corrupt* query, but not both.

(vii) *Test* ($\Pi_U^k$): when the adversary $\mathscr{A}$ sends such a query, the oracle $\Pi_U^k$ flips an unbiased coin $b$. If $b = 1$, then the oracle returns the session key SK; otherwise, it returns a random value. $\mathscr{A}$ is allowed to issue this query only once to the oracle $\Pi_U^k$.

In the simulation above, Simulation $\mathscr{B}$ is perfectly indistinguishable from the proposed protocol unless the event $\text{Event}^{C2S}$ occurs. And, we can see that the event $\exists j$, Osk $\wedge$ Test ($\Pi_C^j$) is equal to the event $\exists i$, Osk $\wedge$ Test ($\Pi_S^i$) $\wedge\, \neg\text{Event}^{C2S}$ so that we have $\text{Pr}[\text{Osk} \wedge \text{Test}(\Pi_C^j)] \geq \varepsilon/2 - \text{Pr}_{C2S}$. Hence, by the simulation of the oracle $\Pi_C^j$ of the client $C$, we have

$$\begin{aligned}&\text{Pr}\Bigg[\text{SK} = f_4\,(\text{ID}, U_1, U_2, V, N, K_C, \\ &\qquad \text{Auth}_S, \text{Auth}_C) \mid \begin{array}{c} N \in Z_q^* \\ U_1, U_2, K_C \leftarrow G_1 \end{array}\Bigg] \\ &\quad \geq \frac{\varepsilon}{2} - \text{Pr}_{C2S}.\end{aligned} \tag{6}$$

Note that, if $\varepsilon$ is nonnegligible, the probability $\varepsilon/2 - \text{Pr}_{C2S}$ is also nonnegligible since the probability $\text{Pr}_{C2S}$ is negligible by Theorem 8. Also, $\mathscr{A}$ can obtain the ephemeral secret value $r$ by the ESL attacks. Now, we assume that $P_{\text{pub}} = aP$ and $Q_{\text{ID},2} = bP$. Then, we have $U_1 = r \cdot P$ and $U_2 = r \cdot Q_{\text{ID},2} = r \cdot bP$. Therefore, if $\mathscr{A}$ could obtain the session key SK with a nonnegligible probability, it means that $\mathscr{A}$ has obtained $K_C$. In this case, given $(U_1, U_2, P_{\text{pub}}) = (r \cdot P, r \cdot bP, aP)$, $\mathscr{A}$ has obtained $K_C = r \cdot D_{\text{ID},2} = r \cdot abP$. Thus, $\mathscr{B}$ can evaluate $abP$ by computing $r^{-1} \cdot K_C$ so that $\mathscr{B}$ solves the CDH problem with a nonnegligible advantage. □

### 5.3. Server-to-Client Authentication

*5.3. Server-to-Client Authentication.* In the following theorem, we prove that an adversary cannot impersonate the server $S$ to communicate with the client $C$ under the CDH assumption.

**Theorem 10.** *If a PPT adversary $\mathscr{A}$ can violate the server-to-client authentication of our proposed protocol with a nonnegligible advantage, then there is a challenger $\mathscr{B}$ which can solve the CDH problem with a nonnegligible advantage.*

*Proof.* Here, we employ an algorithm $\mathscr{B}$ (called Simulation $\mathscr{B}$) to make the initialization and to respond to the adversary $\mathscr{A}$ according to our proposed protocol. As in the proof of Theorem 9, the Simulation $\mathscr{B}$ is perfectly indistinguishable unless the event $\text{Event}^{C2S}$ occurs. Since the probability $\text{Pr}_{C2S}$ is negligible by Theorem 8, we can assume that $\text{Event}^{C2S}$ does not occur.

Let the symbol $\text{Event}^{C2S}$ denote the event violating the server-to-client authentication. If the event $\text{Event}^{C2S}$ occurs, there is an instance $j$ of the client $C$ which has accepted the session with no legal partner. Namely, the oracle $\Pi_C^j$ has issued $(\text{ID}, U_1, U_2, V, Q_{\text{ID},1})$ and received $(N, \text{Auth}_S)$, where the latter is not generated by an oracle $\Pi_S^i$. Therefore, one of the following three cases must have happened. □

*Case 1.* The adversary $\mathscr{A}$ guessed the value $\text{Auth}_S$ correctly.

*Case 2.* The values $U_2$ occurred in other sessions.

*Case 3.* $\mathscr{A}$ asked $f_2$ for the tuple $(\text{ID}, U_1, U_2, V, N, K_S)$ with correct $K_S$.

Next, we discuss the probability for each of the three cases. It is obvious that the probability of Case 1 is less than $q_C/2^n$ and the probability of Case 2 is $(q_C/q)(q_C - 1)$, which is less than $q_C^2/q$. The probability of Case 3 can be denoted by $\Pr[(\mathrm{ID}, U_1, U_2, V, N, K_S) \mid U_1, U_2, P_{\mathrm{pub}} \in G_1, K_S = s \cdot U_2]$. Thus,

$$\begin{aligned}\Pr\left[\mathrm{Event}^{S2C} \mid \neg\mathrm{Event}^{C2S}\right] &\\ \leq \Pr\left[(\mathrm{ID}, U_1, U_2, V, N, K_S) \mid U_1, U_2 \in G_1, K_S = s \cdot U_2\right] &\\ + \frac{q_C}{2^n} + \frac{q_C^2}{q}. &\end{aligned} \tag{7}$$

As before, $\mathscr{A}$ can obtain the ephemeral secret value $r$ by the ESL attacks. Now, we assume that $P_{\mathrm{pub}} = sP$ and $Q_{\mathrm{ID},2} = wP$. Then we have $U_1 = r \cdot P$ and $U_2 = r \cdot Q_{\mathrm{ID},2} = r \cdot wP$ with unknown $s, w \in Z_q^*$. In this case, given $(U_1, U_2, P_{\mathrm{pub}}) = (r \cdot P, r \cdot wP, sP)$, $\mathscr{A}$ can compute $K_S = s \cdot U_2 = r \cdot swP$ with a nonnegligible probability. Therefore, if $\mathscr{A}$ could obtain the session key $SK$, it would be able to compute $K_S$. In such a case, $\mathscr{B}$ can use $\mathscr{A}$ to obtain $swP$. Therefore, $\mathscr{B}$ can solve the CDH problem with a nonnegligible advantage.

Hence, by assuming that $\mathscr{A}$ can violate the server-to-client authentication with a nonnegligible advantage $\varepsilon$, $\mathscr{B}$ then solves the CDH problem with the advantage $\varepsilon' \geq \varepsilon - q_C/2^n - q_C^2/q$. Therefore, under the CDH assumption, our proposed protocol provides the server-to-client authentication.

### *5.4. Implicit Key Confirmation*

**Theorem 11.** *Under the CDH assumption, our proposed protocol offers implicit key confirmation in the random oracle model.*

*Proof.* We say that an ID-AKE protocol provides *implicit key confirmation* if the protocol assures that the server/client can compute a session key which no others can produce. By Theorems 8 and 10, the client and the server can authenticate each other in the random oracle model under the CDH assumption. In Theorem 9, we have proved that an adversary cannot compute the session key. Therefore, our proposed protocol provides implicit key confirmation. □

### *5.5. Partial Forward Secrecy*

**Theorem 12.** *Under the CDH assumption, our proposed protocol offers partial forward secrecy in the random oracle model.*

*Proof.* If the adversary $\mathscr{A}$ corrupts the secret key $s$ of the server, then all the previous session keys can be recovered (from the transcripts) since $\mathscr{A}$ can then compute $K_S = s \cdot U_2$ and $\mathrm{SK} = f_4(\mathrm{ID}, U_1, U_2, V, N, K_S, \mathrm{Auth}_S, \mathrm{Auth}_C)$. On the other hand, we show that the corruption of a client does not help $\mathscr{A}$ to recover previous session keys. In Theorem 9, we allow $\mathscr{A}$ to issue the *Corrupt* (ID) and obtain $D_{\mathrm{ID}} = (D_{\mathrm{ID},1}, D_{\mathrm{ID},2}, Q_{\mathrm{ID},1})$. Since, in a session, a *Test* query is required to occur before *Corrupt* query, Theorem 9 still holds under a *Corrupt* query to the client. Therefore, our proposed protocol offers partial forward secrecy. □

Table 1: Computational costs (in seconds) required for pairing-based operations.

| | $TG_e$ | $TG_{\mathrm{mul}}$ | $T_{\exp}$ |
|---|---|---|---|
| HiPersmart (36 MHz) | 0.38 s | 0.13 s | 0.07 s |

## 6. Performance Comparisons and Discussions

For convenience, the following notations are used to analyze the performance:

$TG_e$: the executing time of a bilinear pairing operation $\hat{e}: G_1 \times G_1 \rightarrow G_2$;

$TG_{\mathrm{mul}}$: the executing time of a scalar multiplication in $G_1$;

$T_{\exp}$: the executing time of a modular exponential operation in $G_2$;

$TG_H$: the executing time of a map-to-point hash function in $G_1$;

$TG_{\mathrm{add}}$: the executing time of an addition in $G_1$ or a multiplication in $G_2$;

$T_H$: the executing time of a hash function;

$|\sigma|$: the bit length of a transmission message $\sigma$.

By the simulation results in [44, 45], $TG_e$, $TG_{\mathrm{mul}}$, $T_{\exp}$, and $TG_H$ are more time-consuming than $TG_{\mathrm{add}}$ and $T_H$, in which $TG_e$ is the most time-consuming operation. Here, we list the simulation result of pairing-based operations with a resource-constrained mobile device. Scott et al. [44] gave the computational costs needed for various pairing-based operations under the Philips HiPersmart card with the processor of maximum clock speed 36 MHz. For the Ate pairing system in [44], a popular and valid choice would be to use a supersingular curve over a finite field $E(F_p)$, with $p = 512$ bits and a large prime order $q = 160$ bits. Table 1 lists the experimental data for related pairing-based operations on the Philips HiPersmart card.

In the following, we analyze the computational cost of the proposed protocol. In our protocol, the client side requires $4TG_{\mathrm{mul}} + TG_H$ and does not require any bilinear pairing operation. Furthermore, the client can perform offline computations in advance in Step 1 of the mutual authentication and key exchange phase described in Section 4.3. Hence, the mutual authentication and key exchange phase requires only $TG_{\mathrm{mul}}$ for online computation on the client side. On the other hand, the server side performs Steps 2 and 4 to authenticate a client with a session key. It requires $3TG_e + 2TG_{\mathrm{mul}} + 2TG_H$. As for the communicational cost, the bit length of communication between a client and the server is bounded by $4|q| + 4|G_1|$.

In Table 2, we demonstrate the comparisons among Ni et al.'s protocol [35], Chuang and Tseng's protocol [33], Islam's protocol [36], and ours in terms of the computational cost, communicational cost, and ESL security. As mentioned in Section 1, both the proposed protocols of Ni et al. and Islam fulfill all basic security properties including ESL resistance, while Chuang and Tseng's protocol cannot withstand ESL

Table 2: Comparisons between some recently proposed protocols and ours.

| | Ni et al.'s protocol [35] | Chuang and Tseng's protocol [33] | Islam's protocol [36] | Our protocol |
|---|---|---|---|---|
| Computational cost for each client (total) | $6TG_e + 2TG_{\text{mul}} + 2T_{\exp}$ | $4TG_{\text{mul}} + T_{\exp}$ | $2TG_e + 4TG_{\text{mul}}$ | $4TG_{\text{mul}} + TG_H$ |
| Computational cost for each client (online) | $2TG_e + TG_{\text{mul}} + 2T_{\exp}$<br>(1.03 seconds) | $TG_{\text{mul}}$<br>(0.13 seconds) | $2TG_e$<br>(0.76 seconds) | $TG_{\text{mul}}$<br>(0.13 seconds) |
| Computational cost for the server | $6TG_e + 2TG_{\text{mul}} + 2T_{\exp}$ | $2TG_e + 3TG_{\text{mul}} + T_{\exp}$ | $2TG_e + 4TG_{\text{mul}}$ | $3TG_e + 2TG_{\text{mul}} + 2TG_H$ |
| Bit length of communication | $2\|q\| + 2\|G_1\| + 2\|G_2\|$ | $4\|q\| + 3\|G_1\|$ | $5\|q\| + 6\|G_1\|$ | $4\|q\| + 4\|G_1\|$ |
| Against ESL attacks | Yes | No | Yes | Yes |

attacks. In Section 5, we have demonstrated that our protocol also fulfills all basic security properties including ESL resistance. Moreover, Chuang and Tseng's protocol and ours only require $TG_{\text{mul}}$ for online computation on the client side. However, both protocols of Ni et al. and Islam still require two bilinear pairing operations. Hence, according to Table 2, our protocol withstands ESL attacks and possesses better performance.

In the following, let us discuss the relationship between our protocol and the extensible authentication protocol (EAP) for wireless networks [40–43]. Typically, the EAP standard or framework [40, 41] is viewed as an authentication framework independent of the underlying authentication technology. Under the EAP framework, many authentication protocols have been proposed, and each of them has various advantages and weaknesses. As in [42], our ESL-secure ID-AKE protocol might be viewed as an authentication method of the EAP framework, without relying on PKI (public key infrastructure). In such a case, there is no need for the management of certificates and the deployment of certification authority (CA).

Most EAP authentication protocols lack identity protection or user anonymity. In this paper, we focus on optimizing the authentication process but do not address the issue of user anonymity. Indeed, as mentioned in Section 1.2, Chuang and Tseng's ID-AKA protocol [33] is suitable for general users (with a long validity period) and anonymous users (with a short validity period) as well. Generally, ID-based or certificate-based authentication protocols must rely on other techniques (e.g., Universal Subscriber Identity Module of cellular networks) to provide user anonymity [42]. In addition, the reader can refer to [43] for the privacy protection issue of authentication protocols in the EAP framework.

## 7. Conclusions

In the paper, we proposed an efficient ESL-secure ID-AKE protocol for mobile client-server environments. Under the CDH assumption, our protocol is provably secure to provide mutual authentication, key agreement, implicit key confirmation, partial forward secrecy, and resistance to the ESL attacks in the random oracle model. We adopt the imbalanced computation to reduce the computational cost required by a mobile client. In addition, a mobile client may perform offline precomputation to reduce the online computational cost. When compared with previously proposed ID-AKE protocols for mobile client-server environments, our protocol has higher security and better computational performance.

## Conflict of Interests

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

## Acknowledgment

This research was partially supported by the Ministry of Science and Technology, Taiwan, under Contract no. MOST103-2221-E-018-022-MY2.

## References

[1] A. Menezes, M. Qu, and S. Vanstone, "Some new key agreement protocols providing mutual implicit authentication," in *Proceedings of the 2nd Workshop on Selected Areas in Cryptography (SAC '95)*, pp. 22–32, 1995.

[2] M. Bellare, D. Pointcheval, and P. Rogaway, "Authenticated key exchange secure against dictionary attacks," in *Advances in Cryptology—Eurocrypt 2000*, pp. 139–155, Springer, 2000.

[3] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *Proceedings of Advances in Cryptology—Eurocrypt2001*, vol. 2045, pp. 453–474, Springer, Berlin, Germany, 2001.

[4] 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.

[5] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[6] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in Cryptology: Proceedings of CRYPTO 84*, vol. 196 of *Lecture Notes in Computer Science*, pp. 47–53, 1985.

[7] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Advances in Cryptology—CRYPTO 2001*, vol. 2139, pp. 213–229, Springer, Berlin, Germany, 2001.

[8] B. Waters, "Efficient identity-based encryption without random oracles," in *Advances in Cryptology—EUROCRYPT 2005*, vol. 3494 of *Lecture Notes in Computer Science*, pp. 114–127, Springer, Berlin, Germany, 2005.

[9] D. Boneh and M. Hamburg, "Generalized identity based and broadcast encryption schemes," in *Advances in Cryptology—ASIACRYPT 2008*, vol. 5350, pp. 455–470, Springer, Berlin, Germany, 2008.

[10] 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.

[11] T.-T. Tsai, Y.-M. Tseng, and T.-Y. Wu, "RHIBE: constructing revocable hierarchical ID-based encryption from HIBE," *Informatica*, vol. 25, no. 2, pp. 299–326, 2014.

[12] J. C. Cha and J. H. Cheon, "An identity-based signature from gap Diffie-Hellman groups," in *Public Key Cryptography—PKC 2003*, vol. 2567 of *Lecture Notes in Computer Science*, pp. 18–30, Springer, Berlin, Germany, 2002.

[13] K. G. Paterson and J. C. N. Schuldt, "Efficient identity-based signatures secure in the standard model," in *Information Security and Privacy*, vol. 4058, pp. 207–222, Springer, Berlin, Germany, 2006.

[14] 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.

[15] 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. 91, no. 7, pp. 1828–1830, 2008.

[16] 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.

[17] 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.

[18] S. K. H. Islam and G. P. Biswas, "A pairing-free identity-based authenticated group key agreement protocol for imbalanced mobile networks," *Annals of Telecommunications*, vol. 67, no. 11-12, pp. 547–558, 2012.

[19] N. P. Smart, "Identity-based authenticated key agreement protocol based on Weil pairing," *Electronics Letters*, vol. 38, no. 13, pp. 630–632, 2002.

[20] K. Shim, "Efficient ID-based authenticated key agreement protocol based on Weil pairing," *Electronics Letters*, vol. 39, no. 8, pp. 653–654, 2003.

[21] L. Chen and C. Kudla, "Identity-based authenticated key agreement from pairings," in *Proceedings of the 16th IEEE Computer Security Foundations Workshop (CSFW '03)*, pp. 219–233, 2003.

[22] Y. J. Choie, E. Jeong, and E. Lee, "Efficient identity-based authenticated key agreement protocol from pairings," *Applied Mathematics and Computation*, vol. 162, no. 1, pp. 179–188, 2005.

[23] 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.

[24] M. Jakobsson and D. Pointcheval, "Mutual authentication for low-power mobile devices," in *Financial Cryptography: Proceedings of Financial Cryptography*, vol. 2339 of *Lecture Notes in Computer Science*, pp. 178–195, 2002.

[25] D. S. Wong and A. H. Chan, "Efficient and mutually authenticated key exchange for low power computing devices," in *Proceedings of Advances in Cryptology—Asiacrypt2001*, vol. 2248, pp. 272–289, Springer, Berlin, Germany, 2001.

[26] H.-A. Wen, C.-L. Lin, and T. Hwang, "Provably secure authenticated key exchange protocols for low power computing clients," *Computers & Security*, vol. 25, no. 2, pp. 106–113, 2006.

[27] K. Y. Choi, J. Y. Hwang, D. H. Lee, and I. S. Seo, "ID-based authenticated key agreement for low-power mobile devices," in *Proceedings of the 10th Australasian Conference on Information Security and Privacy (ACISP '05)*, pp. 494–505, Brisbane, Australia, July 2005.

[28] T.-Y. Wu and Y.-M. Tseng, "An efficient user authentication and key exchange protocol for mobile client-server environment," *Computer Networks*, vol. 54, no. 9, pp. 1520–1530, 2010.

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

[30] D. He, "An efficient remote user authentication and key agreement protocol for mobile client-server environment from pairings," *Ad Hoc Networks*, vol. 10, no. 6, pp. 1009–1016, 2012.

[31] S. H. Islam and G. P. Biswas, "A more efficient and secure ID-based remote mutual authentication with key agreement scheme for mobile devices on elliptic curve cryptosystem," *Journal of Systems and Software*, vol. 84, no. 11, pp. 1892–1898, 2011.

[32] D. He, C. Jianhua, and J. Hu, "An ID-based client authentication with key agreement protocol for mobile client-server environment on ECC with provable security," *Information Fusion*, vol. 13, no. 3, pp. 223–230, 2012.

[33] Y. H. Chuang and Y. M. Tseng, "Towards generalized ID-based user authentication for mobile multi-server environment," *International Journal of Communication Systems*, vol. 25, no. 4, pp. 447–460, 2012.

[34] B. LaMacchia, K. Lauter, and A. Mityagin, "Stronger security of authenticated key exchange," in *Proceedings of the 1st International Conference on Provable Security (ProvSec '07)*, pp. 1–16, 2007.

[35] L. Ni, G. Chen, J. Li, and Y. Hao, "Strongly secure identity-based authenticated key agreement protocols," *Computers and Electrical Engineering*, vol. 37, no. 2, pp. 205–217, 2011.

[36] S. K. H. Islam, "A provably secure ID-based mutual authentication and key agreement scheme for mobile multi-server environment without ESL attack," *Wireless Personal Communications*, 2014.

[37] Y. M. Tseng, T. T. Tsai, and S. S. Huang, "Leakage-free ID-based signature," *The Computer Journal*, 2013.

[38] 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*, pp. 62–73, 1993.

[39] D. Pointcheval and J. Stern, "Security arguments for digital signatures and blind signatures," *Journal of Cryptology*, vol. 13, no. 3, pp. 361–396, 2000.

[40] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz, "Extensible authentication protocol (EAP)," IETF RFC3748, 2004.

[41] D. Simon, B. Aboba, and R. Hurst, "The EAP-TLS authentication protocol," IETF RFC 5216, March 2008.

[42] Y. M. Tseng, "USIM-based EAP-TLS authentication protocol for wireless local area networks," *Computer Standards and Interfaces*, vol. 31, no. 1, pp. 128–136, 2009.

[43] F. Pereñiguez-Garcia, G. Kambourakis, R. M. López, S. Gritzalis, and A. F. Gómez-Skarmeta, "Privacy-enhanced fast re-authentication for EAP-based next generation network," *Computer Communications*, vol. 33, no. 14, pp. 1682–1694, 2010.

[44] M. Scott, N. Costigan, and W. Abdulwahab, "Implementing cryptographic pairings on smartcards," in *Cryptographic Hardware and Embedded Systems—CHES 2006*, vol. 4249 of *Lecture Notes in Computer Science*, pp. 134–147, 2006.

[45] L. B. Oliveira, M. Scott, J. López, and R. Dahab, "TinyPBC: pairings for authenticated identity-based non-interactive key distribution in sensor networks," in *Proceedings of the 5th International Conference on Networked Sensing Systems (INSS '08)*, pp. 173–180, June 2008.

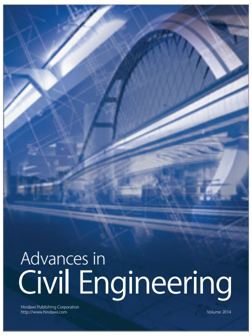
Journal of
Engineering

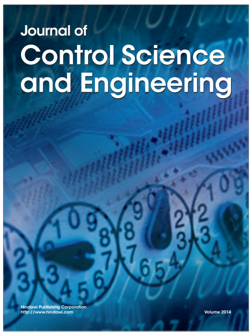
The Scientific
World Journal

International Journal of
Rotating
Machinery

Journal of
Sensors

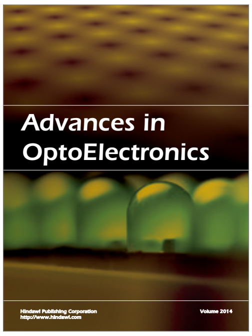
International Journal of
Distributed
Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science
and Engineering

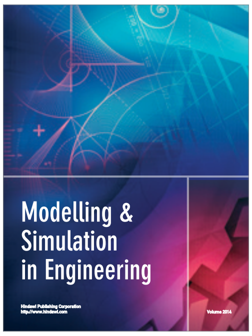
Journal of
Robotics

Journal of
Electrical and Computer
Engineering

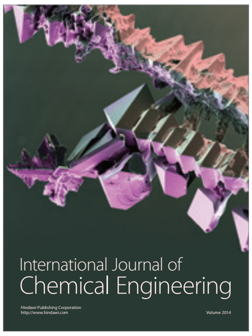

Advances in
OptoElectronics

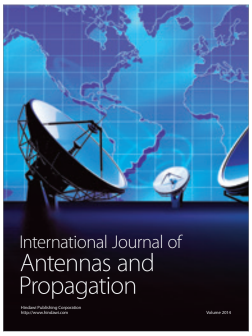
VLSI Design

International Journal of
Navigation and
Observation

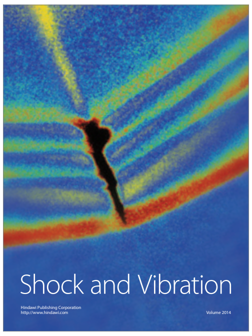

Modelling &
Simulation
in Engineering

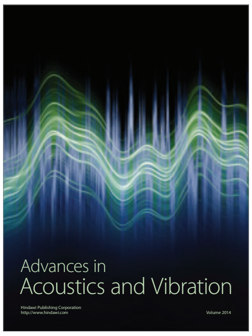
International Journal of
Aerospace
Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and
Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration