# Practical and Exposure-resilient Hierarchical ID-based Authenticated Key Exchange without Random Oracles

Kazuki Yoneyama
*NTT Secure Platform Laboratories, 3-9-11 Midori-cho Musashino-shi Tokyo 180-8585, Japan*

Keywords: Authenticated Key Exchange, Hierarchical ID-based Authenticated Key Exchange, Exposure-resilience.

Abstract: ID-based authenticated key exchange (ID-AKE) is a cryptographic tool to establish a common session key between parties with authentication based on their IDs. If IDs contain some hierarchical structure such as an e-mail address, hierarchical ID-AKE (HID-AKE) is especially suitable because of scalability. However, most of existing HID-AKE schemes do not satisfy advanced security properties such as forward secrecy, and the only known strongly secure HID-AKE scheme is inefficient. In this paper, we propose a new HID-AKE scheme which achieves both strong security and efficiency. We prove that our scheme is eCK-secure (which ensures maximal-exposure-resilience including forward secrecy) without random oracles, while existing schemes is proved in the random oracle model. Moreover, the number of messages and pairing operations are independent of the hierarchy depth; that is, really scalable and practical for a large-system.

## 1 INTRODUCTION

Authenticated Key Exchange (AKE) is a cryptographic primitive to share a common *session key* among multiple parties through unauthenticated networks such as the Internet. In the ordinary PKI-based setting, each party locally keeps his own *static secret key* (SSK) and publish a *static public key* (SPK) corresponding to the SSK. Validity of SPKs is guaranteed by a certificate authority. In a key exchange session, each party generates an *ephemeral secret key* (ESK) and sends an *ephemeral public key* (EPK) corresponding to the ESK. A session key is derived from these keys with a *key derivation function*.

*ID-based AKE* (ID-AKE) is a variant of AKE, and the purpose is to remove the management of certificates. Similar to the basic scenario of ID-based encryption (IBE) such as (Boneh and Franklin, 2001; Boneh and Boyen, 2004; Waters, 2005), a trusted key generation center (KGC) generates a master key (MSK), and SSKs of all parties with the MSK according to their IDs. Various ID-AKE schemes have been studied (Chen et al., 2007; Huang and Cao, 2009; Fiore and Gennaro, 2010). ID-AKE enjoys the same merit as IBE: no need of PKI, and using IDs instead of SPKs. However, at the same time, a problem of *scalability* is inherited: the workload for a KGC becomes burdensome when running on a large system.

To resolve the scalability problem, *hierarchical ID-AKE* (HID-AKE) is useful. In HID-AKE, the key generation can be decentralized through a hierarchy where intermediate nodes in the hierarchy can derive the SSKs for each of its children. For example, the ID of a party $U$ at level $t$ is represented as $(ID_1, ID_2, \ldots, ID_t)$, and the party can generate the SSK of the party which ID is $(ID_1, ID_2, \ldots, ID_t, *)$, where $*$ means a wild-card. Thus, it is enough that KGC just generates a MSK and the SSK of the first level party. The situation is very close to the motivation of hierarchical IBE (HIBE) such as (Horwitz and Lynn, 2002; Gentry and Silverberg, 2002; Boneh and Boyen, 2004; Boneh et al., 2005; Gentry and Halevi, 2009). Various typical IDs contain hierarchical structures such as an e-mail address.

There are existing non-interactive HID-AKE schemes (Blundo et al., 1998; Eschenauer and Gligor, 2002; Ramkumar et al., 2005; Gennaro et al., 2008). Since these schemes can establish a session key without any interaction, efficiency in communication is optimal. However, non-interactive setting cannot avoid abandoning several important security properties such as *forward secrecy*. Forward secrecy means that an adversary cannot recover a session key even if the SSKs are compromised after the completion of the session. Also, in contrast with the ID-AKE setting, we have to consider *collusion resistance* in the HID-AKE setting. Collusion resistance means that disclosure of a party's SSK does not compromise SSKs of


Yoneyama K..
Practical and Exposure-resilient Hierarchical ID-based Authenticated Key Exchange without Random Oracles.
DOI: 10.5220/0004525705180523
In *Proceedings of the 10th International Conference on Security and Cryptography* (SECRYPT-2013), pages 518-523
ISBN: 978-989-8565-73-0

higher-level parties. Unfortunately, above schemes only partially satisfy collusion resistance; that is, if greater numbers of SSKs in a level than a threshold are compromised, a SSK of higher-level party is also compromised.

There is the only existing HID-AKE scheme (Fujioka et al., 2010) which satisfies both forward secrecy and collusion resistance. They formulate a security model by extending the extended Canetti-Krawzcyk (eCK) security model (LaMacchia et al., 2007). We refer to their model as the HID-eCK model. The HID-eCK model captures *maximal-exposure-resilience* which means that an adversary is allowed to obtain any non-trivial[1] combination of MSK, SSKs, and ESKs individually. Thus, maximal-exposure-resilience implies forward secrecy and collusion resistance. Exposure of such secret keys may be usually caused in real-world applications. A MSK is exposed when the KGC is corrupted. A SSK is revealed if an implementer is pretend to generate SSKs in an insecure host machine in order to prevent the randomness generation mechanisms in a tamper-proof module such as a smart card. Also, if a pseudo-random number generator implemented in a system is poor, ESKs will be known to the adversary. Therefore, to consider such a *fail-safe security* is very important to apply a cryptographic scheme to practical systems.

Though the scheme (Fujioka et al., 2010) satisfies strong security, there are two drawbacks. One is the assumption. The security proof is given in the random oracle (RO) model. A strong negative result (Canetti et al., 1998; Canetti et al., 2004) is known for realizability of the RO. The other is efficiency. The number of messages and pairing operations increases with depending on the hierarchy depth. If we want to apply this scheme in a large system, it will be impractical.

## 1.1 Our Contribution

In this paper, we propose the first HID-AKE scheme resolving all problems of existing schemes. Our scheme has several advantages compared with existing schemes. We show a comparison in Table 1.

**Constant-size Overhead in Communication and Computation.** We construct our HID-AKE scheme to use HIBE as a main building block. Though the previous scheme (Fujioka et al., 2010) is also constructed from an HIBE scheme (Gentry and Silverberg, 2002), it inherits inefficiency of the HIBE scheme; that is, the number of messages and pairing operations depends on the hierarchy depth. On the other hand, we use another HIBE scheme (Boneh et al., 2005; Park and Lee, 2007) whose the number of messages and pairing operations are constant-size. Specifically, total messages sent by a party in a session are only two group elements, and a signature and a verification key of one-time signature. Total pairing operations are only four times. Amazingly, our scheme is more efficient in computation than (Fujioka et al., 2010) when the hierarchy depth is higher than 2, while (Fujioka et al., 2010) is proved in the RO model but our scheme can be proved without ROs. Moreover, our scheme also becomes more efficient in communication than (Fujioka et al., 2010) when the hierarchy depth is higher than 7.

**Maximal-Exposure-Resilience.** We prove the security of our scheme in the HID-eCK model (Fujioka et al., 2010). Since the HID-eCK model ensures maximal-exposure-resilience, our scheme satisfies such a strong security. A key technique to achieve the HID-eCK security is the twisted pseudo-random function (PRF) trick (Fujioka et al., 2012). This trick can neutralize the effect of exposure of ESKs if SSKs are not revealed. We can prevent an adversary to obtain any information about a session key from revealed ESKs with this trick. Moreover, we devise the session key derivation procedure to include a shared secret computed only from ESKs in the session as a countermeasure to exposure of the MSK or SSKs. If the MSK or all SSKs are exposed, the adversary cannot know such a shared secret because she does not know ESKs. For detailed discussion, please see Section 3.1.

**Security Proof without Random Oracles.** All (provably secure) existing schemes (Blundo et al., 1998; Eschenauer and Gligor, 2002; Ramkumar et al., 2005; Gennaro et al., 2008; Fujioka et al., 2010) use ROs for deriving a session key. It makes security proofs easy to understand because a simulator can arbitrarily manage the value of session keys thanks to the programmability of ROs in security reductions. Conversely, without ROs, we must exactly simulate session keys according to the protocol. Our solution is applying a technique to simulate decryption queries from the HIBE scheme (Park and Lee, 2007) with the decisional bilinear Diffie-Hellman exponent ($q$-DBDHE) assumption. We can manage session keys correctly with this technique.

[1] If both the SSK and the ESK of a party in the target session are revealed, the adversary trivially obtains the session key for any scheme. Similarly, if both the MSK and an ESK in the target session are revealed, the adversary also trivially wins. This condition is defined as freshness.

Table 1: Comparison of existing HID-AKE schemes and our scheme.

| | Exposure Resilient? | Model | Assumption | Computation [#parings+#regular-exp] | Communication complexity | |
|---|---|---|---|---|---|---|
| (Gennaro et al., 2008) | no | ROM | DBDH | $[1,\ell]$ | none | 0 |
| (Fujioka et al., 2010) | yes | ROM | GBDH | $[3\ell-1,\ell+2]$ | $2\ell\kappa$ | $256\ell$ |
| **Ours** | yes | StdM | $(q+1)$-DBDHE | $[4,\ell+14]$ | $13\kappa$ | 1664 |

DBDH means the Decisional Bilinear Diffie-Hellman assumption. GBDH means the gap Bilinear Diffie-Hellman assumption. We show an instantiation by the Mohassel signature (Mohassel, 2010) as a strongly unforgeable signature in our scheme. For concreteness the expected ciphertext overhead for a 128-bit implementation is also given. Note that computational costs are estimated without any pre-computation technique and any multi-exponentiation technique.

# 2 PRELIMINARIES

In this section, we recall definitions of building blocks. The HID-eCK security model is given in (Fujioka et al., 2010).

Throughout this paper we use the following notations. If $\mathsf{M}$ is a set, then by $m \in_R \mathsf{M}$ we denote that $m$ is sampled uniformly from $\mathsf{M}$. If $\mathcal{R}$ is an algorithm, then by $y \leftarrow \mathcal{R}(x;r)$ we denote that $y$ is output by $\mathcal{R}$ on input $x$ and randomness $r$ (if $\mathcal{R}$ is deterministic, $r$ is empty).

## 2.1 Bilinear Group

Let $G$ and $G_T$ be cyclic groups of prime order $p$ where $g$ is a generator of $G$. We say that $e : G \times G \to G_T$ is a bilinear map if for all $X, Y \in G$ and $a, b \in \mathbb{Z}_p$, $e(X^a, Y^b) = e(X,Y)^{ab}$, and $e(g,g) = g_T \neq 1$. We say that $G$ is a bilinear group if map $e$, and group operations in $G$ and $G_T$ can be computed efficiently.

## 2.2 Decisional Bilinear Diffie-Hellman Exponent Assumption

The $q$-DBDHE problem is as follows. A distinguisher $\mathcal{D}$ is given a $(2q+2)$-tuple $(g, h, g^x, \ldots, g^{x^q}, g^{x^{q+2}}, \ldots, g^{x^{2q}}, T)$, where $h \in_R G$ and $x \in_R \mathbb{Z}_p$. Let $\vec{g}_{x,q} = (g^x, \ldots, g^{x^q}, g^{x^{q+2}}, \ldots, g^{x^{2q}})$. For distinguisher $\mathcal{D}$, we define advantage

$$\mathsf{Adv}^{\mathrm{DBDHE}}(\mathcal{D}) = |\Pr[\mathcal{D}(g,h,\vec{g}_{x,q}, T = e(g,h)^{x^{q+1}}) = 1] - \Pr[\mathcal{D}(g,h,\vec{g}_{x,q}, T = R) = 1]|,$$

where $R \in_R G_T$, and the probability is taken over the choices of $(x,h,R)$ and the random tape of $\mathcal{D}$.

**Definition 2.1** (Decisional Bilinear Diffie-Hellman Exponent Assumption)**.** *We say that the q-DBDHE assumption in G and $G_T$ holds if for all PPT distinguisher $\mathcal{D}$, the advantage $\mathsf{Adv}^{\mathrm{DBDHE}}(\mathcal{D})$ is negligible in security parameter $\kappa$.*

The validity of the DBDHE assumption is proved in the generic group model in (Boneh et al., 2005).

# 3 EXPOSURE-RESILIENT HIERARCHICAL ID-BASED AKE WITHOUT ROs

We construct a HID-AKE scheme based on HIBE schemes (Boneh et al., 2005; Park and Lee, 2007). By applying the twisted PRF trick (Fujioka et al., 2012), the proposed scheme can satisfy the HID-eCK security.

## 3.1 Design Principle

Problems to be solved are roughly classified into two. One is to resist exposure of ESKs, and the other is to resist exposure of the MSK and SSKs. We must solve these problems without the help of ROs.

For the first problem, we use the twisted PRF trick as described in Section 1.1. The twisted PRF means that two PRFs $(F, F')$ with reversing keys are used; that is, we choose two ESKs $(esk, esk')$ and compute $F(esk, ssk) \oplus F'(ssk, esk')$, where $ssk$ is a part of the SSK. It is especially effective in the following two scenarios: exposure of both ESKs of parties in a session, and exposure of the SSK of the session owner and the ESK of the session peer. If $(esk, esk')$ are revealed, $F(esk, ssk)$ cannot be computed without knowing $ssk$. Similarly, if $ssk$ is revealed, $F'(ssk, esk')$ cannot be computed without knowing $esk'$. In the construction, the outputs of the twisted PRF are used as randomness to generate EPKs. Therefore, we can prevent the adversary to obtain any information about randomness because both the SSK and the ESK of a party cannot be revealed according to the freshness definition.

For the second problem, we add a shared secret to derive a session key. The shared secret has the form $e(g,h)^{s_A s_B}$, where $g$ and $h$ are a part of the public parameter, $s_A$ and $s_B$ are a part of the outputs of the twisted PRF generated by $U_A$ and $U_B$ respectively. Since EPSs include $g^{s_A}$ and $g^{s_B}$, $e(g,h)^{s_A s_B}$ can be computed if $s_A$ or $s_B$ is known. On the other hand,

the adversary which does not know both $s_A$ and $s_B$ cannot be compute $e(g,h)^{s_A s_B}$ even if she can obtain the MSK and all SSKs. Note that the adversary cannot reveal both the SSK (the MSK) and the ESK of a party.

## 3.2 Construction

**Parameters.** Let $\kappa$ be the security parameter. Let $G, G_T$ be bilinear groups with pairing $e : G \times G \to G_T$ of order $\kappa$-bit prime $p$ with generators $g, g_T = e(g,g)$, respectively. Let $\ell$ be maximum depth of the hierarchy in the system. Let $(\mathsf{Gen}, \mathsf{Sign}, \mathsf{Ver})$ be a one-time signature scheme such that a verification key is an element of $\mathbb{Z}_p$. Let $F_{ke} : \{0,1\}^* \times \mathsf{FS} \to \mathbb{Z}_p^*$, $F_{gen} : \{0,1\}^* \times \mathsf{FS} \to \mathsf{RS}_{gen}$, $F_{sig} : \{0,1\}^* \times \mathsf{FS} \to \mathsf{RS}_{sig}$, and $F_{kdf} : \{0,1\}^* \times \mathsf{FS} \to \{0,1\}^\kappa$ be pseudo-random functions, where $\mathsf{FS}$ is the key space of PRFs (the length of keys is larger than $\kappa$), $\mathsf{RS}_{gen}$ is the randomness space of Gen, and $\mathsf{RS}_{sig}$ is the randomness space of Sign.

Public parameter *Params* is $(F_{ke}, F_{gen}, F_{sig}, F_{kdf}, G, G_T, g, g_T, g_1, g_2, g_3, g_4, h_1, \ldots, h_\ell)$, where $g_1 = g^z$ for $z \in_R \mathbb{Z}_p^*$, and $g_2, g_3, g_4, h_1, \ldots, h_\ell \in_R G$. Master secret key *MSK* is $g_2^z$.

**Key Generation.** There are two ways to generate a static secret key: from *MSK*, and from higher-level *SSK*. Static secret key $SSK_{ID}$ for $ID = (ID_1, \ldots, ID_i)$ $(i \leq \ell)$ is generated from *MSK* as $SSK_{ID} = \big(MSK \cdot (h_1^{ID_1} \cdots h_i^{ID_i} \cdot g_3)^r, g^r, g_4^r, h_{i+1}^r, \ldots, h_\ell^r, w_1, w_2, w_3, w_4, w_5, w_6\big)$, where $r \in_R \mathbb{Z}_P$ and $w_1, w_2, w_3, w_4, w_5, w_6 \in_R \mathsf{FS}$. Also, static secret key $SSK_{ID}$ for $ID = (ID_1, \ldots, ID_i)$ $(i \leq \ell)$ can be generated from $SSK_{ID'} = (u_0, u_1, u_2, v_i, \ldots, v_\ell, w_1, w_2, w_3, w_4, w_5, w_6)$ for $ID' = (ID_1, \ldots, ID_{i-1})$ as $SSK_{ID} = \Big(u_0 \cdot v_i^{ID_i} \cdot (h_1^{ID_1} \cdots h_i^{ID_i} \cdot g_3)^{r'}, u_1 \cdot g^{r'}, u_2 \cdot g_4^{r'}, v_{i+1} \cdot h_{i+1}^{r'}, \ldots, v_\ell \cdot h_\ell^{r'}, w_1 \oplus w_1', w_2 \oplus w_2', w_3 \oplus w_3', w_4 \oplus w_4', w_5 \oplus w_5', w_6 \oplus w_6'\Big)$, where $r' \in_R \mathbb{Z}_P$ and $w_1', w_2', w_3', w_4', w_5', w_6' \in_R \mathsf{FS}$.

**Key Exchange.** In the following description, user $U_A$ has static secret key $SSK_{ID_A} = (u_{A,0}, u_{A,1}, u_{A,2}, v_{A,\alpha+1}, \ldots, v_{A,\ell}, w_{A,1}, w_{A,2}, w_{A,3}, w_{A,4}, w_{A,5}, w_{A,6})$ corresponding to $ID_A = (ID_{A,1}, ..., ID_{A,\alpha})$ and user $U_B$ has static secret key $SSK_{ID_B} = (u_{B,0}, u_{B,1}, u_{B,2}, v_{B,\beta+1}, \ldots, v_{B,\ell}, w_{B,1}, w_{B,2}, w_{B,3}, w_{B,4}, w_{B,5}, w_{B,6})$ corresponding to $ID_B = (ID_{B,1}, ..., ID_{B,\beta})$.

- $U_A$ chooses ephemeral secret key $ESK_A = (esk_{A,ke}, esk'_{A,ke}, esk_{A,gen}, esk'_{A,gen}, esk_{A,sig}, esk'_{A,sig}) \in_R \mathsf{FS}^6$, and computes ephemeral public key $EPK_A$ as follows:
  1. compute $s_A = F_{ke}(w_{A_1}, esk_{A,ke}) \oplus F_{ke}(esk'_{A,ke}, w_{A_2})$, $rand_{A,gen} = F_{gen}(w_{A_3}, esk_{A,gen}) \oplus F_{gen}(esk'_{A,gen}, w_{A_4})$, and $rand_{A,sig} = F_{sig}(w_{A_5}, esk_{A,sig}) \oplus F_{sig}(esk'_{A,sig}, w_{A_6})$.
  2. run $\mathsf{Gen}(1^\kappa; rand_{A,gen})$, and obtain signing key $sk_A$ and verification key $vk_A$.
  3. compute $C_{A,1} = g^{s_A}$ and $C_{A,2} = (h_1^{ID_{B,1}} \cdots h_\beta^{ID_{B,\beta}} \cdot g_4^{vk_A} \cdot g_3)^{s_A}$.
  4. run $\mathsf{Sign}_{sk_A}(C_{A,1}, C_{A,2}; rand_{A,sig})$, and obtains signature $\sigma_A$.
  5. send ephemeral public key $EPK_A = (C_{A,1}, C_{A,2}, \sigma_A, vk_A)$, $ID_A$ and $ID_B$ to user $U_B$.
- $U_B$ chooses ephemeral secret key $ESK_B = (esk_{B,ke}, esk'_{B,ke}, esk_{B,gen}, esk'_{B,gen}, esk_{B,sig}, esk'_{B,sig}) \in_R \mathsf{FS}^6$, and computes ephemeral public key $EPK_B$ as follows:
  1. compute $s_B = F_{ke}(w_{B_1}, esk_{B,ke}) \oplus F_{ke}(esk'_{B,ke}, w_{B_2})$, $rand_{B,gen} = F_{gen}(w_{B_3}, esk_{B,gen}) \oplus F_{gen}(esk'_{B,gen}, w_{B_4})$, and $rand_{B,sig} = F_{sig}(w_{B_5}, esk_{B,sig}) \oplus F_{sig}(esk'_{B,sig}, w_{B_6})$.
  2. run $\mathsf{Gen}(1^\kappa; rand_{B,gen})$, and obtain signing key $sk_B$ and verification key $vk_B$.
  3. compute $C_{B,1} = g^{s_B}$, and $C_{B,2} = (h_1^{ID_{A,1}} \cdots h_\alpha^{ID_{A,\alpha}} \cdot g_4^{vk_B} \cdot g_3)^{s_B}$.
  4. run $\mathsf{Sign}_{sk_B}(C_{B,1}, C_{B,2}; rand_{B,sig})$, and obtains signature $\sigma_B$.
  5. send ephemeral public key $EPK_B = (C_{B,1}, C_{B,2}, \sigma_B, vk_B)$, $ID_B$ and $ID_A$ to user $U_A$.
- Upon receiving $EPK_B$, $U_A$ checks whether $1 \leftarrow \mathsf{Ver}_{vk_B}((C_{B,1}, C_{B,2}), \sigma_B)$, and aborts if not. Otherwise, $U_A$ derives session key *SK* as follows:
  1. compute $s_A = F_{ke}(w_A, esk_{A,ke}) \oplus F_{ke}(esk'_{A,ke}, w_A)$, and shared secrets
     $$\sigma_1 = e(g_1, g_2)^{s_A},$$
     $$\sigma_2 = e(C_{B,1}, u_{A,0} \cdot u_{A,2}^{vk_B}) / e(C_{B,2}, u_{A,1}),$$
     $$\sigma_3 = e(C_{B,1}, g_3)^{s_A}.$$
  2. set session transcript $\mathsf{ST} = (ID_A, ID_B, EPK_A, EPK_B)$, and compute session key $SK = F_{kdf}(\mathsf{ST}, \sigma_1) \oplus F_{kdf}(\mathsf{ST}, \sigma_2) \oplus F_{kdf}(\mathsf{ST}, \sigma_3)$.

- Upon receiving $EPK_A$, $U_B$ checks whether $1 \leftarrow \mathsf{Ver}_{vk_A}((C_{A,1}, C_{A,2}), \sigma_A)$, and aborts if not. Otherwise, $U_B$ derives session key $SK$ as follows:
  1. compute $s_B = F_{ke}(w_B, esk_{B,ke}) \oplus F_{ke}(esk'_{B,ke}, w_B)$, and shared secrets
  $$\begin{aligned}\sigma_1 &= e(C_{A,1}, u_{B,0} \cdot u_{B,2}^{vk_A}) / e(C_{A,2}, u_{B,1}),\\ \sigma_2 &= e(g_1, g_2)^{s_B},\\ \sigma_3 &= e(C_{A,1}, g_3)^{s_B}.\end{aligned}$$
  2. set session transcript $\mathsf{ST} = (ID_A, ID_B, EPK_A, EPK_B)$, and compute session key $SK = F_{kdf}(\mathsf{ST}, \sigma_1) \oplus F_{kdf}(\mathsf{ST}, \sigma_2) \oplus F_{kdf}(\mathsf{ST}, \sigma_3)$.

**Correctness.** The shared secrets that both parties compute are

$$\begin{aligned}\sigma_1 &= e(g^{s_A}, g_2^z \cdot (h_1^{ID_{B,1}} \cdots h_\beta^{ID_{B,\beta}} \cdot g_3)^{r_B} \cdot g_4^{r_B vk_A})\\ &\quad / e((h_1^{ID_{B,1}} \cdots h_\beta^{ID_{B,\beta}} \cdot g_4^{vk_A} \cdot g_3)^{s_A}, g^{r_B})\\ &= e(g^{s_A}, g_2^z) = e(g_1, g_2)^{s_A},\\ \sigma_2 &= e(g^{s_B}, g_2^z \cdot (h_1^{ID_{A,1}} \cdots h_\beta^{ID_{A,\alpha}} \cdot g_3)^{r_A} \cdot g_4^{r_A vk_B})\\ &\quad / e((h_1^{ID_{A,1}} \cdots h_\alpha^{ID_{A,\alpha}} \cdot g_4^{vk_B} \cdot g_3)^{s_B}, g^{r_A})\\ &= e(g^{s_B}, g_2^z) = e(g_1, g_2)^{s_B},\\ \sigma_3 &= e(g^{s_B}, g_3)^{s_A} = e(g, g_3)^{s_A s_B} = e(g^{s_A}, g_3)^{s_B}.\end{aligned}$$

Therefore, they can compute the same session key $SK$.

## 4 SECURITY

The proposed HID-AKE scheme is selective ID secure in the HID-eCK security model under the $(q+1)$-DBDHE assumption.

**Theorem 4.1.** *If the $(q+1)$-DBDHE assumption in $G$ and $G_T$ holds, and $(\mathsf{Gen}, sig, ver)$ is strongly unforgeable, then the proposed HID-AKE scheme is selective ID secure in the HID-eCK model.*

Proof of Theorem 4.1 will be given in the full version. Here, we provide an intuitive sketch of the proof.

*Proof (Sketch).* We have to consider the following four maximal exposure patterns in the HID-eCK model (matching cases):

**(a)** the SSK of $U_A$ and the ESK of $U_B$
**(b)** the SSK of $U_B$ and the ESK of $U_A$
**(c)** both ESKs
**(d)** both SSKs

In case (a), $\sigma_1$ is protected by the security of $C_{A,1}$ and $C_{A,2}$ because $esk'_{A,ke}$, $esk'_{A,gen}$ and $esk'_{A,sig}$ are not exposed; thus, $F_{ke}(esk'_{A,ke}, w_{A_2})$, $F_{gen}(esk'_{A,gen}, w_{A_4})$ and $F_{sig}(esk'_{A,sig}, w_{A_6})$ are hidden from the property of PRF, and $SSK_{ID_B}$ is not also exposed. In case (b), $\sigma_2$ is protected by the security of $C_{B,1}$ and $C_{B,2}$ because $esk'_{B,ke}$, $esk'_{B,gen}$ and $esk'_{B,sig}$ are not exposed; thus, $F_{ke}(esk'_{B,ke}, w_{B_2})$, $F_{gen}(esk'_{B,gen}, w_{B_4})$ and $F_{sig}(esk'_{B,sig}, w_{B_6})$ are hidden from the property of PRF, and $SSK_{ID_A}$ is not also exposed. In case (c), $\sigma_3$ is protected because $w_{A_1}$, $w_{A_3}$, $w_{A_5}$, $w_{B_1}$, $w_{B_3}$ and $w_{B_5}$ are not exposed; thus, $F_{ke}(w_{A_1}, esk_{A,ke})$, $F_{gen}(w_{A_3}, esk_{A,gen})$, $F_{sig}(w_{A_5}, esk_{A,sig})$, $F_{ke}(w_{B_1}, esk_{B,ke})$, $F_{gen}(w_{B_3}, esk_{B,gen})$ and $F_{sig}(w_{B_5}, esk_{B,sig})$ are hidden from the property of PRF. In case (d), $\sigma_3$ is protected because $esk'_{A,ke}$, $esk'_{A,gen}$, $esk'_{A,sig}$, $esk'_{B,ke}$, $esk'_{B,gen}$ and $esk'_{B,sig}$ are not exposed; thus, $F_{ke}(esk'_{A,ke}, w_{A_2})$, $F_{gen}(esk'_{A,gen}, w_{A_4})$, $F_{sig}(esk'_{A,sig}, w_{A_6})$, $F_{ke}(esk'_{B,ke}, w_{B_2})$, $F_{gen}(esk'_{B,gen}, w_{B_4})$ and $F_{sig}(esk'_{B,sig}, w_{B_6})$ are hidden from the property of PRF.

Then, we transform the HID-eCK security game as the session key in the test session is randomly distributed. First, we change part of the twisted PRF in the test session into a random function because the key of part of the twisted PRF is hidden from the adversary; therefore, the randomness for generating ciphertexts, the signature key pair and the signature can be randomly distributed. Next, we change shared information $\sigma$ into a random value for each pattern; therefore, the input of a PRF is randomly distributed and has sufficient min-entropy. Finally, we change one of the PRFs (corresponding to the replaced $\sigma$) into a random function. Therefore, the session key in the test session is randomly distributed; thus, there is no advantage to the adversary. We can show a similar proof in non-matching cases. □

## REFERENCES

Blundo, C., Santis, A. D., Herzberg, A., Kutten, S., Vaccaro, U., and Yung, M. (1998). Perfectly Secure Key Distribution for Dynamic Conferences. In *Inf. Comput. 146(1)*, pages 1–23.

Boneh, D. and Boyen, X. (2004). Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles. In *EUROCRYPT 2004*, pages 223–238.

Boneh, D., Boyen, X., and Goh, E.-J. (2005). Hierarchical Identity Based Encryption with Constant Size Ciphertext. In *EUROCRYPT 2005*, pages 440–456.

Boneh, D. and Franklin, M. K. (2001). Identity-Based En-

cryption from the Weil Pairing. In *CRYPTO 2001*, pages 213–229.

Canetti, R., Goldreich, O., and Halevi, S. (1998). The Random Oracle Methodology, Revisited (Preliminary Version). In *STOC 1998*, pages 209–218.

Canetti, R., Goldreich, O., and Halevi, S. (2004). The Random Oracle Methodology, Revisited. In *J. ACM 51(4)*, pages 557–594.

Chen, L., Cheng, Z., and Smart, N. P. (2007). Identity-based Key Agreement Protocols From Pairings. In *Int. J. Inf. Sec. 6(4)*, pages 213–241.

Eschenauer, L. and Gligor, V. D. (2002). A key-management scheme for distributed sensor networks. In *ACM Conference on Computer and Communications Security 2002*, pages 41–47.

Fiore, D. and Gennaro, R. (2010). Making the Diffie-Hellman Protocol Identity-Based. In *CT-RSA 2010*, pages 165–178.

Fujioka, A., Suzuki, K., Xagawa, K., and Yoneyama, K. (2012). Strongly Secure Authenticated Key Exchange from Factoring, Codes, and Lattices. In *Public Key Cryptography 2012*, pages 467–484.

Fujioka, A., Suzuki, K., and Yoneyama, K. (2010). Hierarchical ID-Based Authenticated Key Exchange Resilient to Ephemeral Key Leakage. In *IWSEC 2010*, pages 164–180.

Gennaro, R., Halevi, S., Krawczyk, H., Rabin, T., Reidt, S., and Wolthusen, S. D. (2008). Strongly-Resilient and Non-interactive Hierarchical Key-Agreement in MANETs. In *ESORICS 2008*, pages 49–65.

Gentry, C. and Halevi, S. (2009). Hierarchical Identity Based Encryption with Polynomially Many Levels. In *TCC 2009*, pages 437–456.

Gentry, C. and Silverberg, A. (2002). Hierarchical ID-Based Cryptography. In *ASIACRYPT 2002*, pages 548–566.

Horwitz, J. and Lynn, B. (2002). Toward Hierarchical Identity-Based Encryption. In *EUROCRYPT 2002*, pages 466–481.

Huang, H. and Cao, Z. (2009). An ID-based Authenticated Key Exchange Protocol Based on Bilinear Diffie-Hellman Problem. In *ASIACCS 2009*, pages 333–342.

LaMacchia, B., Lauter, K., and Mityagin, A. (2007). Stronger Security of Authenticated Key Exchange. In *ProvSec 2007*, pages 1–16.

Mohassel, P. (2010). One-Time Signatures and Chameleon Hash Functions. In *Selected Areas in Cryptography 2010*, pages 302–319.

Park, J. H. and Lee, D. H. (2007). Direct Chosen-Ciphertext Secure Hierarchical ID-Based Encryption Schemes. In *EuroPKI 2007*, pages 94–109.

Ramkumar, M., Memon, N. D., and Simha, R. (2005). A hierarchical key pre-distribution scheme. In *IEEE EIT 2005*.

Waters, B. (2005). Efficient Identity-Based Encryption Without Random Oracles. In *EUROCRYPT 2005*, pages 114–127.