# Direct Chosen Ciphertext Secure Hierarchical ID-Based Encryption Schemes in the Selective-ID Security Model[1)]

*Jong Hwan Park *Kyu Young Choi **Dong Hoon Lee
Graduate School of Information Security, CIST, Korea University, Korea
*{decartian, young}@cist.korea.ac.kr **donghlee@korea.ac.kr

## Abstract

It has been widely believed that one can obtain $l$-Hierarchical Identity Based Encryption (HIBE) scheme secure against chosen ciphertext attacks from $(l+1)$-HIBE scheme secure against chosen plaintext attacks. In this paper, however, we show that when applying two concrete HIBE schemes that Boneh et al. [1, 2] proposed, chosen ciphertext secure $l$-HIBE schemes are directly derived from chosen plaintext secure $l$-HIBE schemes. Our constructions are based on a one-time signature-based transformation that Canetti et al. [3] proposed. The security of our schemes is proved in the selective-ID security model without using random oracles.

## 1. Introduction

To prove the security for Identity-Based Encryption (IBE) and Hierarchical Identity-Based Encryuption (HIBE) schemes without random oracles, Canetti et al. [4, 3] proposed a weaker security model, called selective-ID security model. In this model the adversary is forced to commit ahead of time to the identity it wishes to attack before the setup stage. This model is weaker than the full security model (i.e., Boneh-Franklin security model [5]) in which the adversary is allowed to adaptively choose the identity that it wishes to attack. In 2004, Boneh and Boyen [1] provided a HIBE (denoted by $\mathbf{BB_1}$) scheme proven secure in the selective-ID security model without random oracles. Thereafter, Boneh, Boyen, and Goh [2] presented an improved HIBE (denoted by **BBG**) scheme where the number of ciphertext elements and pairing operations are independent of the hierarchy depth. Their scheme was also proved secure in the selective-ID security model without random oracles: the selective-ID security model for HIBE was further generalized in [6].

In both $\mathbf{BB_1}$ and **BBG** schemes, $l$-HIBE schemes secure against chosen ciphertext attacks was build from $(l+1)$-HIBE schemes secure against chosen plaintext attacks, using the signature-based method (so called "CHK transformation") of [3]. The CHK transformation, improved upon by Boneh and Katz [7] and further by Boyen, Mei, and Waters [8], requires one-time signature scheme to check the consistency of ciphertext. The important point is that the verification key associated with the one-time signature needs to be embedded into ciphertext in encryption procedure. For this, the authors [2, 3] add one level to the identity hierarchy and set the verification key as an identity at the bottom. Eventually, the authors [1, 2, 3] considered a $(l+1)$-HIBE scheme as a subroutine in constructing a $l$-HIBE scheme secure against chosen ciphertext attacks. Then, the identity and verification key act as a new identity for $(l+1)$-HIBE scheme.

In this paper, we present a new way of directly constructing $l$-HIBE scheme secure against chosen ciphertext attacks from $l$-HIBE (not from $(l+1)$-HIBE) scheme secure against chosen plaintext attacks in the selective-ID security model. Our method is based on the $\mathbf{BB_1}$ and **BBG** $l$-HIBE schemes, using the CHK transformation. In both the HIBE schemes, the size of ciphertext increases by one more element (plus one-time signature and a corresponding verification key) and decryption algorithm requires one more pairing computation than the original chosen plaintext secure HIBE schemes. The security of our constructions is based on the same assumptions used in the security proofs of the $\mathbf{BB_1}$ and **BBG** HIBE schemes.

## 2. Preliminaries

We briefly review the definition of security for HIBE. We also summarize the bilinear maps and the related security assumptions.

### 2.1 Selective-ID Security Model for HIBE

An Identity Based Encryption (IBE) scheme consists of four algorithms [9, 5]: Setup, KeyGen, Encrypt, Decrypt. The Setup algorithm generates system parameters **params** and a master key **master-key**. The KeyGen algorithm applies the **master-key** to an

1) This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Advancement) (IITA-2006-(C1090-0603-0025)).

identity to generate the private key for that identity. When encrypting messages, the Encrypt algorithm requires a receiver's identity (as a public key) and the system parameters. The Decrypt algorithm decrypts ciphertexts with a private key associated with the receiver's identity. In a Hierarchical Identity Based Encryption (HIBE) [1, 2], identities are considered as vectors. That is, an identity of depth $l$ is a tuple $ID=(I_1,\ldots,I_l)$. A HIBE scheme also consists of the above four algorithms, but the difference is that the KeyGen algorithm takes as input an identity $ID=(I_1,\ldots,I_l)$ at depth $l$ and the private key $d_{ID|l-1}$ of the parent identity $ID_{|l-1}=(I_1,\ldots,I_{l-1})$ at depth $l-1$.

To prove chosen ciphertext security for HIBE schemes without random oracles, we are interested in the selective-ID model which was defined by Boenh et al. [1, 2, 3]. Selective-ID security model for HIBE scheme is described via the following game between an adversary $A$ and a challenger.

**Init:** $A$ outputs an identity $ID^*$ that it intends to be challenged.

**Setup:** The challenger runs Setup algorithm. It gives $A$ the resulting system parameters **params**. It keeps the **master-key** to itself.

**Phase 1:** $A$ issues queries $q_1,\ldots,q_m$ adaptively where $q_i$ is one of:

- Private key query on $ID_i$ where $ID_i \neq ID^*$ and $ID_i$ is not a prefix of $ID^*$. The challenger responds by running KeyGen algorithm to generate the private key $d_i$ corresponding to the public key $ID_i$. It sends $d_i$ to $A$.
- Decryption query $CT_i$ on $ID^*$ or any prefix of $ID^*$. The challenger responds by running KeyGen algorithm to generate the private key $d$ corresponding to $ID^*$. It then runs Decrypt algorithm to decrypt the ciphertext $CT_i$ using the private key $d$ and sends the resulting plaintext to $A$.

**Challenge:** Once $A$ decides that Phase 1 is over, it outputs two equal length plaintexts $M_0, M_1 \in M$ on which it wishes to be challenged. The challenger picks a random bit $b \in \{0,1\}$ and computes the ciphertext $CT = Encrypt(M_b, params, ID^*)$. It sends $CT$ as the challenge to $A$.

**Phase 2:** $A$ issues more queries $q_{m+1},\ldots,q_n$ adaptively where $q_i$ is one of:

- Private key query on $ID_i$ where $ID_i \neq ID^*$ and $ID_i$ is not a prefix of $ID^*$. The challenger responds as in Phase 1.
- Decryption query $CT_i \neq CT$ on $ID^*$ or any prefix of $ID^*$. The challenger responds as in Phase 1.

**Guess:** Finally, $A$ outputs a guess $b' \in \{0,1\}$. $A$ wins if $b'=b$.

We refer to such an adversary $A$ as an IND-sID-CCA adversary. The advantage of $A$ in breaking the HIBE scheme $E$ is defined as

$$Adv_{E,A} = \left|\Pr[b=b'] - \frac{1}{2}\right|.$$

**Definition 1.** *We say that a HIBE scheme $E$ is $(t, q_{ID}, q_C, \epsilon)$-selective-ID, adaptive chosen ciphertext secure if for any $t$-time IND-sID-CCA adversary $A$ that makes at most $q_{ID}$ chosen private key queries, at most $q_C$ chosen decryption queries we have that $Adv_{E,A} < \epsilon$.*

## 2.2 Complexity Assumptions

Recall that a pairing is an efficiently computable, non-degenerate function, $e: G \times G \rightarrow G_1$, with the bilinearity property that $e(g^r, g^s) = e(g,g)^{rs}$. Here, $G$ and $G_T$ are all multiplicative groups of prime order $p$, respectively generated by $g$ and $e(g,g)$.

Next, we briefly summarize the bilinear maps and bilinear groups. We review the Bilinear Diffie-Hellman (BDH) assumption and the Bilinear Diffie-Hellman Exponent (BDHE) assumption.

**Decisional Bilinear Diffie-Hellman Assumption:** The BDH problem in $G$ is defined as follows: given a tuple $(g, g^a, g^b, g^c)$ as input, compute $e(g,g)^{abc} \in G_1$. The decisional version of BDH problem is stated as follows: given a tuple $(g, g^a, g^b, g^c, T)$ for random $a, b, c \in Z_p$, decide whether $T = e(g,g)^{abc}$.

**Decisional Bilinear Diffie-Hellman Assumption:** The BDHE problem in $G$ is defined as follows: given a $(2q+1)$-tuple $(g, h, g^x, \ldots, g^{x^{q-1}}, g^{x^{q+1}}, \ldots, g^{x^{2q}}) \in G^{2q+1}$ as input, compute $e(g,h)^{x^q} \in G_1$. Similar to the above, the decisional version of BDHE problem is described as follows: given a tuple $(g, h, g^x, \ldots, g^{x^{q-1}}, g^{x^{q+1}}, \ldots, g^{x^{2q}}, T)$ for random $x \in Z_p$, decide whether $T = e(g,h)^{x^q}$.

**Definition 2.** *We say that the decision $(t, q, \epsilon)$-BDH (or BDHE) assumption holds in $G$ if no $t$-time algorithm has advantage at least $\epsilon$ in solving the decision BDH (or $q$-BDHE) problem in $G$.*

# 3. Direct Chosen Ciphertext Secure HIBE from $BB_1$ HIBE

We transform the $BB_1$ scheme into a IND-sID-CCA secure HIBE scheme without random oracles, using the CHK transformation [3]. In this construction, we need a one-time signature scheme Sig=(SigKeyGen, Sign, Verify) which is strongly existentially unforgeable. Briefly speaking, a signature scheme is

$(t, q_S, \epsilon)$-strongly existentially unforgeable if no $t$-time forger who makes at most $q_S$ signature queries is able to generate a new signature on even a previously signed message with probability at least $\epsilon$ (see the complete definition in [10]). We note that instead of one-time signature-based method, we can use the Message Authentication Code (MAC)-based method. We assume that the verification keys are elements of $Z_p$.

### 3.1 Construction

**Setup(k)**: To generate HIBE system parameters for maximum depth of $l$, select a random $\alpha \in Z_p$ and set $g_1 = g^\alpha$. Next, pick random elements $h, h_1, \ldots, h_l \in G$ and a generator $g_2 \in G$. The public parameters **params** and the secret **master-key** are given by

$$params = (g, g_1, g_2, h, h_1, \ldots, h_l), \; masterkey = g_2^\alpha$$

For $j = 1, \ldots, l$, define $F_j : Z_p \rightarrow G$ to be the function: $F_j(x) = g_1^x h_j$.

**Extract**$(d_{ID|j-1}, ID)$: To create a private key $d_{ID}$ for a user $ID = (I_1, \ldots, I_j) \in Z_p^l$ of depth $j \leq l$, pick random $r_1, \ldots, r_j \in Z_p$ and output

$$d_{ID} = (g_2^\alpha \prod_{k=1}^{j} F_k(I_k)^{r_k}, g^{r_1}, \ldots, g^{r_j}).$$

The private key for $ID$ can be also generated from a private key for $d_{ID|j-1}$. Let $d_{ID|j-1} = (d_0, \ldots, d_{j-1})$ be the private key for $ID_{j-1}$. After selecting a random $r_j \in Z_p$, output

$$d_{ID} = (d_0 F_j(I_j)^{r_j}, d_1, \ldots, d_{j-1}, g^{r_j}).$$

**Encrypt(M, params, ID)**: To encrypt a message $M \in G_1$ under a public key $ID = (I_1, \ldots, I_j) \in Z_p^j$,

1. run the SigKeyGen to obtain a signing key $SigK$ and a verification key $VerK$.
2. pick a random $s \in Z_p$ and compute

$$C = (g^s, e(g_1, g_2)^s M, F_1(I_1)^s, \ldots, F_j(I_j)^s, (g_1^{VerK} h)^s).$$

3. output the ciphertext $CT = (C, Sign_{SigK}(C), VerK)$.

**Decrypt(CT, params, $d_{ID}$)**: To decrypt a ciphertext CT using the private key $d_{ID} = (d_0, \ldots, d_j)$,

1. verify that the signature of C is valid under the key $VerK$. If invalid, output $\perp$.
2. otherwise, let $C = (A, B, C_1, \ldots, C_{j+1})$. Pick a random $r_{j+1} \in Z_p$ and output

$$\frac{\prod_{k=1}^{j} e(C_k, d_k) e(C_{j+1}, g^{r_{j+1}})}{e(A, d_0 (g_1^{VerK} h)^{r_{j+1}})} B.$$

The correctness of decryption algorithm is verified as follows:

$$\begin{aligned}
&\frac{\prod_{k=1}^{j} e(C_k, d_k) e(C_{j+1}, g^{r_{j+1}})}{e(A, d_0 (g_1^{VerK} h)^{r_{j+1}})} B \\
&= \frac{\prod_{k=1}^{j} e(F_k(I_k), g)^{sr_k} e(g_1^{VerK} h, g)^{sr_{j+1}}}{e(g^s, g_2^\alpha \prod_{k=1}^{j} F_k(I_k)^{r_k} (g_1^{VerK} h)^{r_{j+1}})} B \\
&= \frac{B}{e(g_1, g_2)^s} \\
&= M.
\end{aligned}$$

At a first glance, the above scheme has a similar structure of $(l+1)$-HIBE scheme in that the additional element $h$ adds to the public parameters and the size of ciphertext increases by one more element. However, the private key for ID is still generated at level $(l-1)$ and is the same as that of IND-sID-CPA secure $l$-HIBE scheme.

### 3.2 Security

Due to the space limitation, we omit the security proof and state the result.

**Theorem 1.** *Suppose that the decision $(t, \epsilon_1)$-BDH assumption holds in $G$ and the signature scheme is $(t, 1, \epsilon_2)$-strongly existentially unforgeable. Then the previous $l$-HIBE scheme is $(t', q_{ID}, q_C, \epsilon)$-IND-sID-CCA secure for arbitrary $q_{ID}$, $q_C$, and $t' < t - o(t)$, where $\epsilon_1 + \epsilon_2 \geq \epsilon$.*

## 4. Direct Chosen Ciphertext Secure HIBE from BBG HIBE

The technique of the previous section can be easily applied to the BBG HIBE scheme. We present a IND-sID-CCA secure $l$-HIBE scheme based on the IND-sID-CPA secure $l$-HIBE BBG scheme.

### 4.1 Construction

**Setup(k)**: To generate HIBE system parameters for maximum depth of $l$, select a random $\alpha \in Z_p$ and set $g_1 = g^\alpha$. Next, pick random elements $g_2, g_3, h, h_1, \ldots, h_l \in G$. The public parameters **params** and the secret **master-key** are given by

$params = (g, g_1, g_2, g_3, h, h_1, \ldots, h_l)$, $masterkey = g_2^{\alpha}$

**Extract**$(d_{ID|j-1}, ID)$: To create a private key $d_{ID}$ for a user $ID = (I_1, \ldots, I_j) \in Z_p^l$ of depth $j \leq l$, pick random $r \in Z_p$ and output

$$d_{ID} = ( g_2^{\alpha} (h_1^{I_1} \cdots h_k^{I_k} g_3)^r,\ g^r,\ h_{k+1}^r, \ldots,\ h_l^r ).$$

The private key for $ID$ can be also generated from a private key for $d_{ID|j-1}$. Let

$$\begin{aligned} d_{ID|j-1} &= ( g_2^{\alpha} (h_1^{I_1} \cdots h_{k-1}^{I_{k-1}} g_3)^{r'},\ g^{r'},\ h_k^{r'}, \ldots,\ h_l^{r'} ) \\ &= ( a_0, a_1, b_k, \ldots, b_l ). \end{aligned}$$

be the private key for $ID_{j-1}$. After selecting a random $r^* \in Z_p$ and output $d_{ID}$ as

$$( a_0 b_k^{I_k} (h_1^{I_1} \cdots h_k^{I_k} g_3)^{r^*},\ a_1 g^{r^*},\ b_{k+1} h_{k+1}^{r^*}, \ldots,\ b_l h_l^{r^*} ).$$

Since $r = r' + r^*$, we see that this private key is a properly distributed private key for $ID = (I_1, \ldots, I_j) \in Z_p^l$.

**Encrypt(M, params, ID)**: To encrypt a message $M \in G_1$ under a public key $ID = (I_1, \ldots, I_j) \in Z_p^j$,

1. run the SigKeyGen to obtain a signing key $SigK$ and a verification key $VerK$.
2. pick a random $s \in Z_p$ and compute

$$C = (g^s,\ e(g_1, g_2)^s M,\ (h_1^{I_1} \cdots h_k^{I_k} g_3)^s,\ (g_1^{VerK} h)^s).$$

3. output the ciphertext $CT = (C,\ Sign_{SigK}(C),\ VerK)$.

**Decrypt(CT, params, $d_{ID}$)**: To decrypt a ciphertext CT using the private key $d_{ID} = (a_0, a_1, b_{k+1}, \ldots, b_l)$,

1. verify that the signature of C is valid under the key $VerK$. If invalid, output $\perp$.
2. otherwise, let $C = (A, B, C_1, C_2)$. Pick a random $t \in Z_p$ and output

$$\frac{e(C_1, a_1) e(C_2, g^t)}{e(A,\ a_0 (g_1^{VerK} h)^t)} B.$$

The correctness of decryption algorithm is checked as follows:

$$\begin{aligned} &\frac{e(C_1, a_1) e(C_2, g^t)}{e(A,\ a_0 (g_1^{VerK} h)^t)} B \\ &= \frac{e((h_1^{I_1} \cdots h_k^{I_k} g_3)^s,\ g^r)\ e((g_1^{VerK} h)^s,\ g^t)}{e(g^s,\ g_2^{\alpha} (h_1^{I_1} \cdots h_k^{I_k} g_3)^r (g_1^{VerK} h)^t)} B \\ &= \frac{B}{e(g_1, g_2)^s} \\ &= M. \end{aligned}$$

### 4.2 Security

**Theorem 2.** *Suppose that the decision* $(t, l+1, \epsilon_1)$*-BDHE assumption holds in* $G$ *and the signature scheme is* $(t, 1, \epsilon_2)$*-strongly existentially unforgeable. Then the previous* $l$*-HIBE scheme is* $(t', q_{ID}, q_C, \epsilon)$*-IND-sID-CCA secure for arbitrary* $q_{ID}$, $q_C$, *and* $t' < t - \Theta(\tau l q_{ID})$, *where* $\epsilon_1 + \epsilon_2 \geq \epsilon$ *and* $\tau$ *is the maximum time for an exponentiation in* $G$.

## 5. Conclusion

We presented two HIBE schemes that are secure against chosen ciphertext attacks, based on the $BB_1$ and BBG HIBE schemes, respectively. We obtain the chosen ciphertext security of the proposed $l$-HIBE schemes by directly using CHK transformation from $l$-HIBE schemes. which does not rely on the generic transformation based on $(l+1)$-HIBE scheme. Our security proofs were provided in the selective-ID security model and without random oracles.

## [References]


[1] D. Boneh and X. Boyen. Efficient selective-ID secure identity based encryption without random oracles. In Eurocrypt'04, volume 3027 of LNCS, pages 223-238. Springer, 2004.

[2] D. Boneh, X. Boyen, and E. Goh. Hierarchical identity based encryption with constant size ciphertext. In Eurocrypt'05, volume 3494 of LNCS, pages 440-456. Springer, 2005.

[3] C. Canetti, S. Halevi, and J. Katz. Chosen ciphertext security from identity-based encryption. In Eurocrypt'04, volume 3027 of LNCS, pages 207-222. Springer, 2004.

[4] C. Canetti, S. Halevi, and J. Katz. A forward-secure public-key encryption scheme. In Eurocrypt'03, volume 2656 of LNCS. Springer, 2003.

[5] D. Boneh and M. Franklin. Identity-based encryption from the Weil pairing. In Crypto'01, volume 2139 of LNCS, pages 213-229. Springer, 2001.

[6] S. Chatterjee and P. Sarkar. Generalization of the selective-ID security model for HIBE protocols. In PKC'06, volume 3958 of LNCS, pages 241-256. Springer, 2006.

[7] D. Boneh and J. Katz. Improved efficiency for CCA-secure cryptosystems built using identity-based encryption. In CT-RSA' 05, volume 3376 of LNCS, pages 87-103. Springer, 2005.

[8] X. Boyen, Q. Mei, and B. Waters. Direct chosen ciphertext security from identity-based techniques. In ACM Conference on Computer and Communications Security - CCS'05, pages 320-329. New-York: ACM Press, 2005.

[9] A. Shamir. Identity-based cryptosystems and signature schemes. In CRYPTO'84, volume 196 of LNCS, pages 47-53. Springer, 1984.

[10] D. Boneh and X. Boyen. Short signatures without random oracles. In Eurocrypt'04, volume 3027 of LNCS, pages 56-73. Springer, 2004.