A Novel Commutative Blinding Identity Based Encryption Scheme

Yu Chen Song Luo Jianbin Hu **Zhong Chen**

Information Security Lab, Peking University, China

FPS 2011
Paris, France

12 May 2011

Introduction

IBE schemes from pairings can be classified into three families: full domain hash family, exponent inversion family, and commutative blinding family.

Commutative blinding refers to the generation process of the session key that two secret coefficients and two blinding factors commute with each other under the pairing.

Boneh-Boyen IBE, Waters IBE belong to commutative blinding family.

- Naturally support high-level protocol, such as HIBE, IBE with wildcards, and ABE.

Boneh-Boyen IBE

- Fully secure in the standard model with an exponential loose reduction.
- Fully secure in the random oracle model with a polynomial loose reduction.

Waters IBE

- Fully secure in the standard model with a polynomial loose reduction.
- Suffer from large public parameters.

Motivation

For a cryptographic scheme, a tighter security reduction to some hard assumption not only means a better security guarantee, but also means a more efficient implementation.

- How to construct a new IBE scheme with commutative blinding structure with
  1. Tighter security reduction
  2. Smaller public parameters
- Investigate how to achieve CCA security.

Our Construction (1)

Setup. Run $\mathsf{GroupGen}(1^\kappa) \to (p, \mathbb{G}, \mathbb{G}_T, e)$, pick $x \xleftarrow{R} \mathbb{Z}_p$, $g, Y \xleftarrow{R} \mathbb{G}$, compute $X = g^x$. Pick a cryptographic hash function $H : \{0,1\}^* \to \mathbb{G}$. The public parameters are $mpk = (g, X, Y, H)$. The master secret is $msk = Y^x$.

KeyGen. To generate the private key d_{ID} for an identity $\mathsf{ID} \in \{0,1\}^*$, pick a random $r \in \mathbb{Z}_p$ and output

$$d_{\mathsf{ID}} = (d_1, d_2) = (Y^x Q^r, g^r) \in \mathbb{G} \times \mathbb{G}$$

where $Q = H(\mathsf{ID})$ is the public key of the identity ID.

Our Construction (2)

Encrypt. To encrypt a message $M \in \mathbb{G}_T$ under the identity ID, pick a random $z \in \mathbb{Z}_p$, compute $Q = H(\mathsf{ID})$. Then the ciphertext is constructed as

$$C = (g^z, Q^z, e(X,Y)^z M) \in \mathbb{G} \times \mathbb{G} \times \mathbb{G}_T$$

Decrypt. To decrypt a given ciphertext $C = (C_1, C_2, C_3)$ under ID using the private key $d_{\mathsf{ID}} = (d_1, d_2)$, output

$$\begin{aligned} C_3 \frac{e(d_2, C_2)}{e(d_1, C_1)} &= e(X,Y)^z M \frac{e(g^r, Q^z)}{e(Y^x Q^r, g^z)} \\ &= e(X,Y)^z M \frac{e(g, Q^{rz})}{e(X,Y)^z e(Q^{rz}, g)} = M \end{aligned}$$

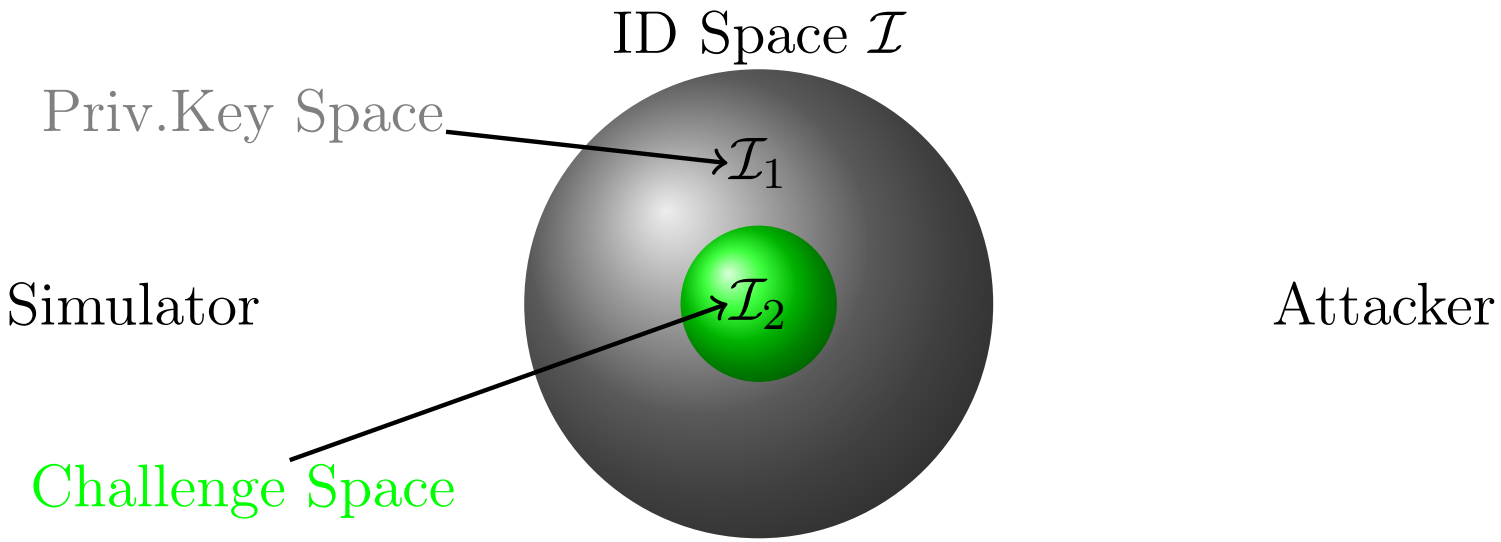
Identity Hash Function

Our scheme is similar to Waters-IBE and BB_1-IBE. The difference lies at the identity hash function (IHF).

- Boneh-Boyen IBE: $\mathsf{IHF}(\mathsf{ID}) = U'Y^{F(\mathsf{ID})}$, where $U', Y \in \mathbb{G}$, $F : \mathsf{ID} \to \mathbb{Z}_p$ is a hash function.
- Waters IBE: $\mathsf{IHF}(\mathsf{ID}) = U' \prod_{i=1}^{n} v_i U_i$, known as Waters hash, where $U', U_i \in \mathbb{G}$, v_i is the i-th bit of ID. Waters hash is essentially a $(1, \mathsf{poly})$- programmable hash function.
- Our scheme: $\mathsf{IHF}(\mathsf{ID}) = H(\mathsf{ID})$, where $H : \{0,1\}^* \to \mathbb{G}$ is a cryptographic hash function and be modeled as a random oracle. The IHF is an ideally programmable hash function.

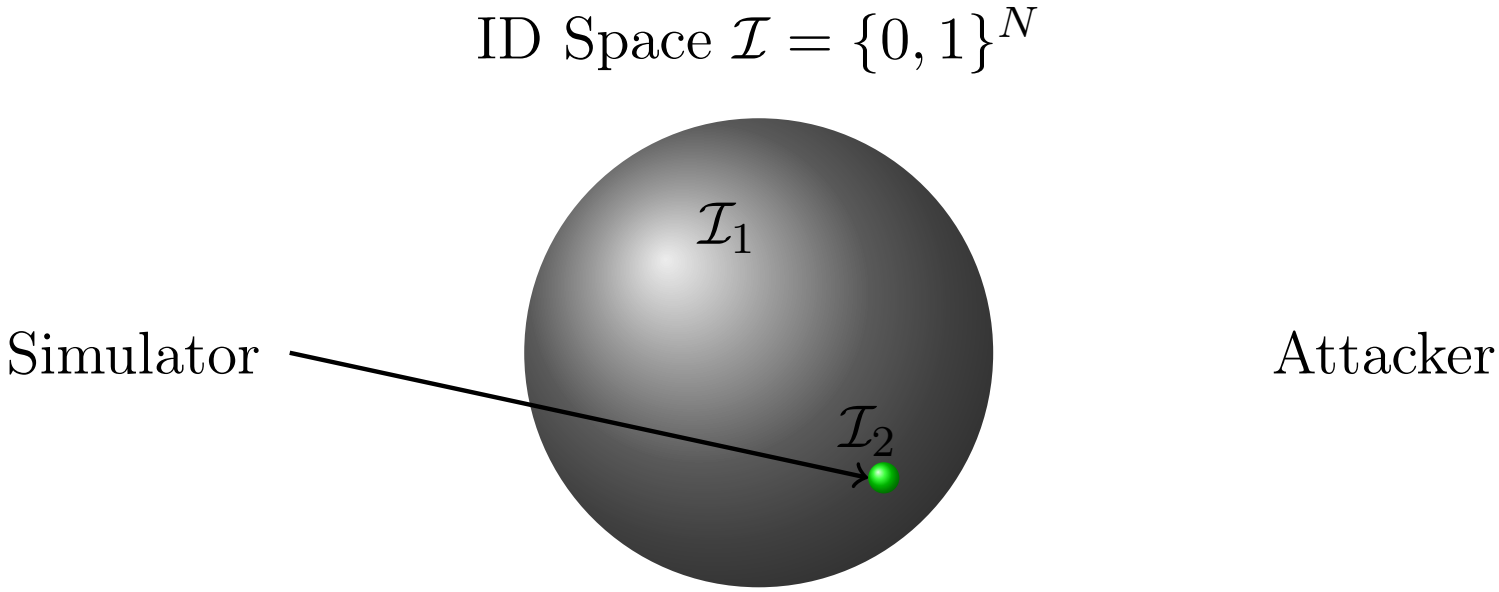
Partitioning Strategy

The simulator partitions the $\mathcal{I}$ into $\mathcal{I}_1$ and $\mathcal{I}_2$, this is done by either using the programmability of RO or using the trapdoor information in the public parameters.



The simulator expects all the identities of private key queries fall into $\mathcal{I}_1$, and the challenge identity fall into $\mathcal{I}_2$. Otherwise the simulator aborts.

Boneh-Boyen IBE



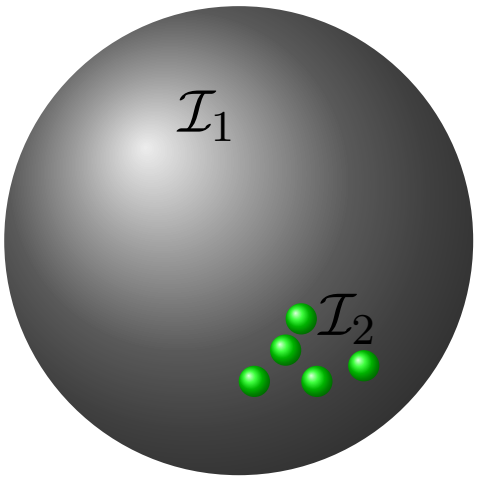
The challenge identity space $\mathcal{I}_2$ shrinks to a single point. The simulator has to guess it correctly. Known as “All-But-One" technique.

Security reduction factor is $1/2^N$, $N \geq 128$.

Waters IBE

ID Space $\mathcal{I} = \{0,1\}^N$

Simulator



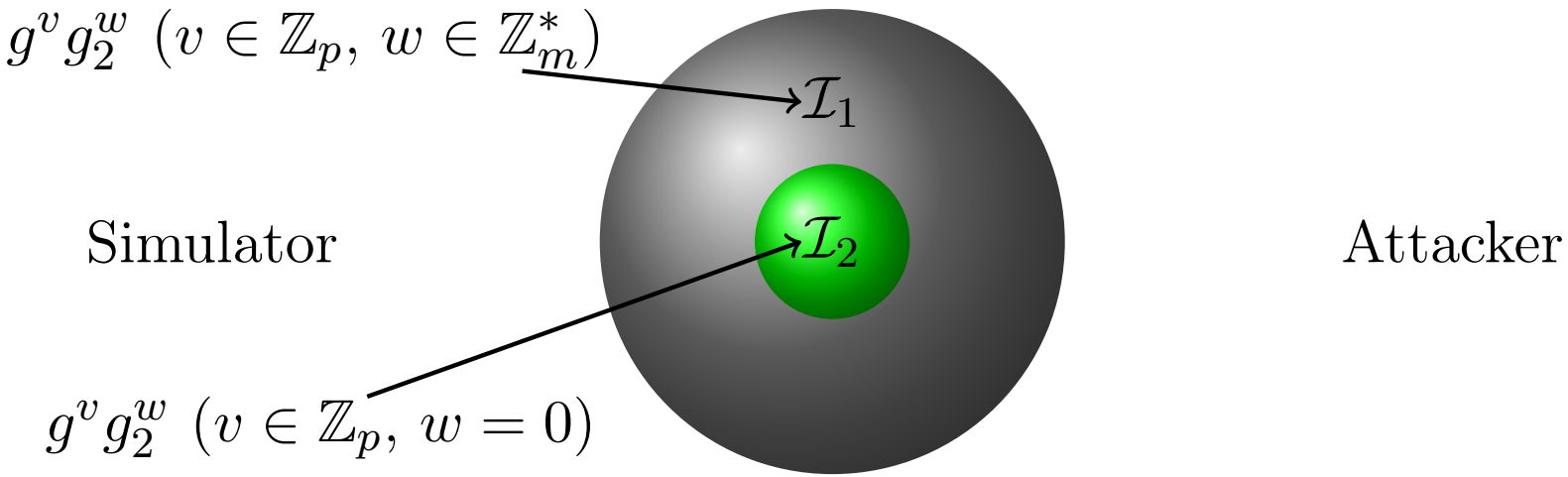
Attacker

The challenge identity space $\mathcal{I}_2$ contains a set of points. Can be summarized as “All-But-Several" technique.

Security reduction factor is $\frac{1}{32(n+1)Q_e}$, where $n \geq 128$.

Our Scheme

$$\mathcal{I} = g^v Y^w \ (v \in \mathbb{Z}_p,\ w \in \mathbb{Z}_m)$$



It is easy to see that $\mathcal{I}_2/\mathcal{I}_1 = 1/(m-1)$.

The reduction factor is $\frac{1}{eQ_e}$.

Twin BDH Problem

BDH Problem

Given $X, Y, Z \in \mathbb{G}$, compute $T = e(g,g)^{xyz}$, where $X = g^x$, $Y = g^y$, and $Z = g^z$.

Twin BDH Problem

Given $X_1, X_2, Y, Z \in \mathbb{G}$, compute $T_1 = e(g,g)^{x_1yz}$ and $T_2 = e(g,g)^{x_2yz}$, where $X_1 = g^{x_1}$ and $X_2 = g^{x_2}$.

Strong Twin BDH Problem

Solve the twin BDH problem with access to a decisional oracle: $\hat{T}_1 \stackrel{?}{=} e(g,g)^{x_1yz} \wedge \hat{T}_2 \stackrel{?}{=} e(g,g)^{x_2yz}$.

Cash, Kiltz, and Shoup prove that strong twin BDH problem is as hard as the usual BDH problem.

Benefits of the Strong Twin BDH Problem

- The strong twin BDH problem can easily be employed in schemes where one would use the usual BDH problem.
- For schemes based on strong the twin BDH problem, the simulator can use the decisional oracle to locate the final solution precisely, thus have tighter security reductions compared to the schemes based on usual BDH problem.
- Facilitate a kind of redundancy free KEM construction without making a stronger assumption.

CCA Construction from Twin Technique (1)

Setup. Pick $x_1, x_2 \xleftarrow{R} \mathbb{Z}_p$, $g, Y \xleftarrow{R} \mathbb{G}$, compute $X_1 = g^{x_1}$, $X_2 = g^{x_2}$. Pick two cryptographic hash functions $H : \{0,1\}^* \to \mathbb{G}$ and $K : \{0,1\}^* \times \mathbb{G} \times \mathbb{G}_T \times \mathbb{G}_T \to \{0,1\}^\lambda$. $mpk = (g, X_1, X_2, Y, H, K)$, $msk = (Y^{x_1}, Y^{x_2})$.

KeyGen. To generate the private key d_{ID} for an identity $\mathsf{ID} \in \{0,1\}^*$, pick random $r_1, r_2 \in \mathbb{Z}_p$ and output

$$d_{\mathsf{ID}} = (d_{11}, d_{12}, d_{21}, d_{22}) = (Y^{x_1} Q^{r_1}, g^{r_1}, Y^{x_2} Q^{r_2}, g^{r_2}) \in \mathbb{G}^4$$

where $Q = H(\mathsf{ID})$ can be viewed as the public key of the identity ID.

CCA Construction from Twin Technique (2)

Encrypt. To encrypt a message $M \in \{0,1\}^n$ under the identity ID, randomly pick $z \in \mathbb{Z}_p$, and set $k := K(\mathsf{ID}, g^z, e(X_1, Y)^z, e(X_2, Y)^z)$, the ciphertext is

$$C = (g^z, Q^z, \mathsf{Enc}(k, M)) \in \mathbb{G} \times \mathbb{G} \times \{0,1\}^n$$

Decrypt. To decrypt $C = (C_1, C_2, C_3)$ under ID, first check if $e(C_1, Q) = e(C_2, g)$ holds. If not, reject the ciphertext. Otherwise, use the private key $d_{\mathsf{ID}} = (d_1, d_2, d_3, d_4)$ to compute

$$\frac{e(d_{11}, C_1)}{e(d_{12}, C_2)} = \frac{e(Y^{x_1} Q^{r_1}, g^z)}{e(g^{r_1}, Q^z)} = e(X_1, Y)^z;$$
$$\frac{e(d_{21}, C_1)}{e(d_{22}, C_2)} = \frac{e(Y^{x_2} Q^{r_2}, g^z)}{e(g^{r_2}, Q^z)} = e(X_2, Y)^z.$$

$k := K(\mathsf{ID}, g^z, e(X_1, Y)^z, e(X_2, Y)^z)$, return $\mathsf{Dec}(k, C_3)$.

Comparison

Scheme	Assumption	Reduction	ROM	$\|mpk\|$
Boneh-Boyen IBE	DBDH	Q_h	yes	$4\|\mathbb{G}\|$
Waters IBE	DBDH	$32(n+1)Q_e$	no	$(n+4)\|\mathbb{G}\|$
Our scheme	DBDH	eQ_e	yes	$3\|\mathbb{G}\|$
Boneh-Boyen IBE+ FO-transformation	CBDH	eQ_eQ_h	yes	$3\|\mathbb{G}\|$
Our scheme+ Twin Technique	CBDH	eQ_e	yes	$4\|\mathbb{G}\|$

For security concern, n is suggested to be at least 128. $Q_e \approx 2^{30}$, $Q_h \approx 2^{60}$ refer to the maximum number of private key queries and the maximum number of random oracle queries, respectively. $e \approx 2.71$ is the base of the natural logarithm. The efficiency, ciphertext size of Boneh-Boyen IBE, Waters IBE, and our scheme are the same.

Any Questions?

Thanks for listening☺