REMARKS ON IBE SCHEME OF WANG AND CAO

Sunder Lal and Priyam Sharma
Derpartment of Mathematics, Dr. B.R.A.(Agra), University,
Agra-282002(UP), India.
E-mail- sunder_lal2@rediffmail.com, priyam_sharma.ibs@rediffmail.com

Abstract: In this paper we analyze and find an anomaly in the security proof of the identity-based encryption (IBE) scheme *fullM-IBE* of Wang and Cao [8], which is based on mBDHP. Here we give another proof for *fullM*-IBE which is based on Bilinear Diffie-Hellman Problem (BDHP). We also obtain a tightness improvement using a stronger assumption, namely, the Bilinear Inverse Dicision Diffie-Hellman problem (BIDDHP).

Key-Words: Public-Key Encryption, Identity-Based Encryption (IBE), IND-ID-CCA attack, BDHP, BIDDHP, Random Oracle.

1. Introduction

In traditional (certificate based) public key cryptosystems, public keys are usually generated at random and secret keys are computed by the users. However, in 1984, Adi Shamir [6] introduced the concept of ID-based cryptosystems in which the public key of a user is derived from his identity and his private key is generated by a trusted third party called Private Key Generator (PKG). Main advantage of an ID-based cryptosystem is that it simplifies the key management process, which is a heavy burden in the traditional certificate based cryptosystem. The concept of ID-based encryption remained a concept till 2001 when Boneh and Franklin [1] proposed the first practical IBE scheme, *BasicIdent*. Using the padding technique of Fujisaki-Okamoto [3] they extended *BasicIdent* to *FullIdent* scheme, and proved that it is secure against chosen ciphertext attacks provided BDHP is hard. In 2003, Galindo [5] pointed out a flaw in the security proof of *FullIdent* [1], and provided another proof for the security of *FullIdent* without changing the scheme or the underlying assumption. In 2007, Wang [7] proposed another IBE scheme based on pairing which is more practical in multiple private key generator (PKG) environments than the IBE scheme *BasicIdent* of Boneh and Franklin [1]. In 2007, Sunder Lal and Priyam Sharma [9] analyzed the security of the IBE scheme by Wang [7] and proved that it relies on the BDHP for its security. In 2007, Wang and Cao [8] (an updated version of [7]) used the transformation technique of Fujisaki-Okamoto [4], and the transformation from [5], to transform the IBE scheme of [7] into *fullM-IBE* scheme, and proved that it is secure against chosen ciphertext attack. For security, they relied on the modified version of Bilinear Diffie-Hellman Problem (mBDHP) (which is weaker than BDHP).

In this paper we re-analyze the security of the IBE scheme of Wang and Cao [8]. In the security analysis of *fullM-IBE* Wang and Cao used a public-key encryption scheme-BasicPub, obtained from M-IBE (which is the IBE scheme of Wang [7]), but

BasicPub is not the public-key encryption scheme that we get from M-IBE, since the public parameters **params** contains more information that the public parameters **params** of M-IBE. This does not match with the general philosophy. In this paper, using another security proof, which matches with the general philosophy, we show that the scheme relies on the BDHP for its security. We also obtain an improved tightness using BIDDHP which is stronger than BDHP.

2. Preliminaries:

2.1 Identity-Based Encryption (IBE) Scheme:

An *identity-Based Encryption Scheme* consists of four randomized algorithms: **Setup**, **Extract**, **Encrypt**, **and Decrypt.**

Setup: It takes a security parameter k and returns system parameters **params** and **master-key.** The **params** which is known publically includes the description of a finite plaintext space $\mathcal{M}$ and the description of a finite ciphertext space $\mathcal{C}$. The master-key is known only to the private key generator (PKG).

Extract: This algorithm extracts private key from the given public key. It takes as input **params**, the **master-key** and an identity string $ID \in \{0, 1\}^*$, and returns key d_{ID}. String ID is used as public key, and d_{ID} as the corresponding private key.

Encrypt: Takes as input the **params,** an identity ID and a plaintext $M \in \mathcal{M}$ and returns a ciphertext $C \in \mathcal{C}$.

Decrypt: Takes as input **params**, a private key d_{ID}, and $C \in \mathcal{C}$. and returns $M \in \mathcal{M}$.

If **params** is the system parameters produced by the **Setup** algorithm, d_{ID} is the private key, corresponding to ID, which is generated by the algorithm **Extract,** then

$$\forall M \in \mathcal{M}, \quad \textbf{Decrypt (params, } d_{ID}\textbf{, Encrypt (params, ID, M)) = M.}$$

2.2 Adaptive Chosen Ciphertext Attack:

Semantic security against adaptive chosen ciphertext attack for an identity-based encryption scheme (IND-ID-CCA) is defined through the following game between challenger and adversary.

Setup: The challenger takes a security parameter k and runs the **Setup** algorithm. She then returns public system parameters **params** to the adversary and keeps the **master-key** to itself.

Phase1: The adversary issues queries $q_1, q_2, \ldots\ldots\ldots, q_n$ which is one of the following:

-Extraction query <ID_j>: The challenger responds by running the algorithm **extract** to generate the private-key d_j corresponding to the public-key ID_j and returns to the adversary.

-Decryption query <ID_j, C_j>: The challenger responds by running the algorithm **extract** to generate the private-key d_j corresponding to the public-key ID_j, uses this private key to decrypt the ciphertext C_j and returns the resulting plaintext to the adversary.

Challenge: The adversary outputs two equal length plaintext M_0, $M_1 \in \mathcal{M}$, with the only constrain that ID must not have appeared in any of the extraction query in **Phase1**. The challenger picks a random bit $b \in \{0, 1\}$ and sends the challenge C = Encrypt(params, ID, M_b) to the adversary.

Phase2: The adversary issues queries q_{n+1}, q_{n+1}, ….., q_t which is one of:

-Extraction query <ID_j> where $ID_j \neq ID$: The challenger responds as in **Phase1**.

-Decryption query <ID_j, C_j> $\neq$ <ID, C>: The challenger responds as in **Phase1**.

Guess: The adversary outputs a guess $b' \in \{0, 1\}$. He wins the game if b' = b.

Such an adversary is called an IND-ID-CCA attacker. The advantage of an IND-ID-CCA attacker $\mathcal{A}$ against the scheme is defined to be:

$$\mathrm{Adv}_A(\kappa) = \left|\Pr\left[b' = b\right] - \frac{1}{2}\right|$$

where the probability is over the random choices made by the challenger and the adversary. An identity-based encryption scheme is said to be semantically secure against adaptive chosen ciphertext attack (IND-ID-CCA) if no polynomially bounded adversary has non-negligible advantage in the game described above.

2.3 Bilinear Pairings:

Let G_1 be an additive group of order p, a prime and let P be a generator of G_1. Let G_2 be a multiplicative group of the same order p. A map $e : G_1 \times G_1 \to G_2$ is said to be a bilinear pairing if it satisfies the following properties:

(Bilinearity): For all P, Q $\in G_1$ and a, b $\in Z_p^*$, $e(aP, bP) = e(P, P)^{ab}$.

(Non-Degeneracy): For a given R $\in G_1$, $e(Q, R) = 1$, for all Q $\in G_1$ if and only if $R = 0$, where 1 is the identity of G_2 and 0 is the identity of G_1.

(Computability): For all P, Q $\in G_1$, there is an efficient algorithm to compute $e(P, Q)$ in polynomial time.

Following are some of the mathematical problems in G_1, G_2:

- **Computational Diffie-Hellman Problem (CDHP):** Given P, aP, bP in G_1, for some (unknown) a, $b \in Z_p^*$, compute abP in G_1.
- **Bilinear Diffie-Hellman Problem (BDHP):** Given P, aP, bP, cP in G_1, for some (unknown) a, b, $c \in Z_p^*$, compute $e(P, P)^{abc}$ in G_2.
- **Bilinear Inverse Diffie-Hellman Problem (BIDHP):** Given P, aP, bP in G_1, for some (unknown) a, $b \in Z_p^*$, compute $e(P,P)^{a^{-1}b}$ in G_2.
- **Bilinear Square Diffie-Hellman Problem (BSDHP):** Given P, aP, bP in G_1 for some (unknown) a, $b \in Z_p^*$, compute $e(P,P)^{a^2 b}$ in G_2.
- **Modified Bilinear Diffie-Hellman Problem (mBDHP):** Given P, aP, $a^{-1}P$, bP, cP in G_1, for some (unknown) a, $b \in Z_p^*$ compute $e(P, P)^{abc}$ in G_2.
- **Bilinear Decision Diffie-Hellman Problem (BDDHP):** Given P, aP, bP, cP in G_1 and $T \in G_2$, for some (unknown) a, b, $c \in Z_p^*$, decide whether $T = e(P,P)^{abc}$.
- **Modified Bilinear Decision Diffie-Hellman Problem (mBDDHP):** Given P, aP, $a^{-1}P$, bP, cP in G_1 and $T \in G_2$, for some (unknown) a, b, $c \in Z_p^*$, decide whether $T = e(P,P)^{abc}$.
- **Bilinear Inverse Decision Diffie-Hellman Problem (BIDDHP):** Given P, aP, bP, cP in G_1 and $T \in G_2$, for some (unknown) a, b, $c \in Z_p^*$, decide whether $T = e(P,P)^{a^{-1}bc}$.

  It may be noted here that, BIDDHP is termed as Decisional Modified BDHP in [2].

It is easy to show that, if we have an algorithm to solve the CDHP in G_1 or G_2, then using this algorithm we can solve BDHP in $\langle G_1, G_2, e\rangle$. In other words, *the BDHP in $\langle G_1, G_2, e\rangle$ is no harder than the CDHP in G_1 or G_2*. But, the problem that, *the CDHP in G_1 or G_2 is no harder than the BDHP* is still an open problem. Also, it is shown in [6] that BDHP, BIDHP, and BSDHP are all polynomial time equivalent. It is easy to see that *mBDHP is no harder than the BDHP*. *mBDDHP is no harder than BDDHP*. Also, *mBDDHP is no harder than BIDDHP*.

3. IBE Scheme by Wang and Cao (*fullM-IBE)*:

We first describe the IBE scheme *fullM-IBE* proposed by Wang and Cao [8]. The scheme consists of the following four algorithms:

Setup: The algorithm works as follows:

1. Runs $\mathcal{IG}$ on input k to generate two prime order groups G_1 and G_2 and a bilinear map $e: G_1 \times G_1 \to G_2$. Here $|G|_1$=$|G_2|$=p and G_1= < P >.
2. Chooses $s \in Z_p^*$ and computes $P_{Pub} = s^{-1}P \in G_1$.
3. For a suitable n and $k_0 \in \mathrm{N}$, chooses the plaintext space $\mathcal{M}=\{0,1\}^{n-k_0}$, the ciphertext space $\mathcal{C}= G_1^* x \{0,1\}^n$ and three cryptographic hash functions $H_1: \{0,1\}^* \to G_1$, $H_2: G_2 \to \{0,1\}^n$ and $H_3: \{0,1\}^{n-k_0} \times \{0,1\}^{k_0} \to Z^*_p$

The params is <G_1, G_2, e, n, p, P, P_{Pub}, H_1, H_2, H_3>, and the master-key is s.

Extract: For an identity $ID \in \{0,1\}^*$, PKG computes

1. $Q_{ID} = H_1(ID) \in G_1$ as the public key, and
2. $d_{ID} = sQ_{ID}$ as the corresponding private key.

Encrypt: To encrypt a plaintext $m \in \mathcal{M}$ for user with identity ID the sender

1. picks a random $\sigma \in \{0,1\}^{k_0}$ and compute $r=H_3(m,\sigma) \in Z_p^*$
2. computes $Q_{ID} = H_1(ID)$ and $g_{ID} = e(P, Q_{ID}) \in G_2$, and
3. sets the ciphertext C=< rP_{Pub}, $(m\|\sigma) \oplus H_2(g_{ID}^r)$ >.

Decrypt: To decrypt a ciphertext $C=<U, V> \in \mathcal{C}$, the receiver using the private key d_{ID}, and params < G_1, G_2, e, n, p, P, P_{Pub}, Q_{ID}, H_2, H_3>

1. computes $m' = V \oplus H_2(e(U, d_{ID})) = m\|\sigma$, and
2. parses $m\|\sigma$ and computes $r = H_3(m,\sigma) \in Z_p^*$. Accepts the ciphertext iff $U = rP_{Pub}$.
3. Outputs m.

The correctness follows because $e(U, d_{ID}) = e(rs^{-1}P, sQ_{ID}) = e(P, Q_{ID})^r$.

4. Security Analysis:

Regarding the security of *fullM-IBE*, Wang and Cao [8] proved the following:

Theorem: The *fullM-IBE* scheme is (t, q_H, q_D, ε)-secure if the mBDHP on (G_1,G_2, e) is

$$(t + c_{G_1}(2q_D + q_H) + q_H O(\log^3 p + \log p), \varepsilon / q_H^2) \text{ secure.}$$

In the above proof Wang and Cao reduce the *fullM-IBE* to a scheme called BasicPub much the same way as is done in Boneh-Franklin [1] and Galindo [5]. However, contrary to the reduced form by Boneh-Franklin and Galindo, the reduced BasicPub of Wang and Cao need more public information than is available in the full scheme. Information $P_{Pub}' = sP \in G_1$ is not a part of **params** in *fullM-IBE*, but it is a part of **params** in BasicPub of Wang and Cao. We feel it is an anomaly in the security proof of *fullM-IBE*. Here we provide a security proof which is free from this anomaly. Moreover, security proof is based on BDHP as against mBDHP of Wang and Cao. We prove the following theorem:

Theorem1: The *fullM-IBE* scheme is (t, q_H, q_D, ε)-secure if the BDHP on (G_1,G_2, e) is

$$(t + c_{G_1}(2q_D + q_H) + q_H O(\log^3 p + \log p), \varepsilon / q_H^2) \text{ secure.}$$

To prove the above theorem we make use of a public-key encryption scheme called as BasicPub^{Hy} which is obtained by applying Fujisaki-Okamoto transformation [4] to the public-key encryption scheme BasicPub-Wang in [9]. In the next subsection we describe the BasicPub-Wang.

4.1 BasicPub-Wang:

The scheme has three algorithms: **Keygen, Encrypt**, and **Decrypt**. Algorithms **Encrypt** and **Decrypt** are same as that of IBE scheme of Wang [7] (which is called M-IBE in [8]). The scheme is as follows:

Keygen: The algorithm works as follows:

1. As in the **Setup** algorithm of IBE scheme of Wang (M-IBE), $\mathcal{IG}$ generates two prime order groups G_1, G_2 and a bilinear map $e : G_1 \times G_1 \to G_2$. Also, the PKG computes its public key P_{Pub} and secret key s in the same way.
2. The plaintext space $\mathcal{M} = \{0,1\}^n$, the ciphertext space $\mathcal{C} = G_1^* \text{x } \{0,1\}^n$ and a cryptographic hash function H_2: $G_2 \to \{0, 1\}^n$ are chosen in the same way.
3. The algorithm now picks a random point $Q_{ID} \neq 0$ in G_1, the group generated by P.

4. The public key is <G_1, G_2, e, n, p, P, P_{Pub}, Q_{ID}, H_2>, the private key is $d_{ID}=sQ_{ID} \in G_1$.

Encrypt: To encrypt $m \in \{0, 1\}^n$, the algorithm chooses random $r \in Z_p^*$ and computes $C=< rP_{Pub}, m \oplus H_2(g_{ID}^r) >$, where $g_{ID} = e(P, Q_{ID}) \in G_2$.

Decrypt: To decrypt C=<U, V> the algorithm takes the public key < G_1, G_2, e, n, p, P, P_{Pub}, Q_{ID}, H_2> and private key d_{ID} as input,

1. computes $m = V \oplus H_2(e(U, d_{ID}))$, and

2. returns m.

4.2BasicPub^{Hy}:

The scheme is obtained by applying the Fujisaki-Okamoto transformation [4] to BasicPub-Wang. The scheme has three algorithms: **Keygen, Encrypt,** and **Decrypt**. Algorithms **Encrypt** and **Decrypt** are same as that of *fullM-IBE*.
The scheme is as follows:

Keygen: The algorithm works as follows:

1. Runs $\mathcal{IG}$ on input k to generate two prime order groups G_1 and G_2 and a bilinear map $e: G_1 \times G_1 \rightarrow G_2$. Here $|G|_1=|G_2|=p$ and $G_1= < P >$.

2. Chooses $s \in Z_p^*$ and computes $P_{Pub} = s^{-1}P \in G_1$.

3. For a suitable n and $k_0 \in \mathbb{N}$, chooses the plaintext space $\mathcal{M}=\{0,1\}^{n-k_0}$, the ciphertext space $\mathcal{C}= G_1^* x \{0,1\}^n$ and two cryptographic hash functions $H_1:G_2 \rightarrow \{0,1\}^n$ and $H_3: \{0,1\}^{n-k_0} \times \{0,1\}^{k_0} \rightarrow Z_p^*$.

4. The algorithm now picks a random point $Q_{ID} \neq 0$ in G_1, the group generated by P.

The public key is <G_1, G_2, e, n, p, P, P_{Pub}, Q_{ID}, H_2, H_3>, the private key is $d_{ID}=sQ_{ID} \in G_1$.

Encrypt: To encrypt $m \in \{0, 1\}^n$, the algorithm chooses random $\sigma \in \{0,1\}^{k_0}$, computes $r = H_3(m, \sigma)$ and $C=< rP_{Pub}, (m \| \sigma) \oplus H_2(g_{ID}^r) >$, where $g_{ID} = e(P, Q_{ID}) \in G_2$.

Decrypt: To decrypt C=<U, V> the algorithm takes < G_1, G_2, e, n, p, P, P_{Pub}, Q_{ID}, H_2, H_3> and private key d_{ID} as input,

1. computes m' = $V \oplus H_2(e(U, d_{ID})) = m \| \sigma$,

2. parses $m\|\sigma$ and computes $r = H_3(m, \sigma) \in Z_p^*$. Checks that $U = rP_{Pub}$. If not rejects the ciphertext.

3. returns m.

To prove Theorem1 we proceed in the following three steps:

- We show that IND-ID-CCA attack on *fullM-IBE* can be converted into IND-CCA attack on BasicPub^{Hy}. This will show that private key extraction queries do not help the adversary.

- We show that IND-CCA attack on BasicPub^{Hy} can be converted into an IND-CPA attack on BasicPub-Wang.

- We show that IND-CPA attack on BasicPub-Wang can be converted into an algorithm that can solve BIDHP.

Lemma1: Let $\mathcal{A}$ be a t time IND-ID-CCA adversary with advantage ε against the *fullM-IBE* scheme making at most q_E private key extraction queries, q_D decryption queries and q_1 hash queries. Then there is an IND-CCA adversary $\mathcal{B}$ that has advantage at least $\frac{\varepsilon}{q_1}\left(1 - \frac{q_1}{q_E}\right) \approx \frac{\varepsilon}{q_1}$ against BasicPub^{Hy}. Its running time is $t' \leq t + c_{G_1}(q_D + q_E + q_1)$, where c_{G_1} denotes the time of computing a random multiple in G_1.

Proof: The proof can be found in [8]. ♣

Lemma2: Let $\mathcal{A}$ be a t time IND-CCA adversary with advantage ε against BasicPub^{Hy} making at most q_D decryption queries and q_2 hash queries. Then there is an IND-CPA adversary $\mathcal{B}$ that has advantage at least $(\varepsilon - q_2 2^{-(k_0 - 1)})(1 - 1/p)^{q_D} \approx \varepsilon$ against BasicPub-Wang with the running time $t' \leq t + q_2(T_{BasicPub} + \log p)$, where $T_{BasicPub}$ is the running time of **Encrypt** algorithm in BasicPub-Wang.

Proof: This result is obtained by applying Fujisaki-Okamoto transformation. The proof can be found in [4]. ♣

Lemma1 and Lemma2 are the same as Lemma 1 and Lemma 2 respectively of [8].

Lemma3: Let $\mathcal{A}$ be a t time IND-CPA adversary with advantage ε against BasicPub-Wang making at most q_{H_2} queries to H_2. Then there is an algorithm $\mathcal{B}$ that has advantage at least $\frac{\left(\varepsilon - \frac{1}{2^n}\right)}{q_{H_2}} \approx \frac{\varepsilon}{q_{H_2}}$ in solving the BIDHP. Its running time is t' = O(t).

Proof: Algorithm $\mathcal{B}$ is given an input the BIDH parameters $<G_1, G_2, e>$ produced by $\mathcal{IG}$ and a random instance <P, aP, bP> of the BIDHP for these parameters i.e., $P \in_R G_1$ where a, $b \in_R \mathbb{Z}_p^*$. $|G_1| = p = |G_2|$. Let $D = e(P,P)^{a^{-1}b} \in G_2$ be the solution to this problem. Algorithm $\mathcal{B}$ finds D by interacting with algorithm A as follows:

Setup: Algorithm $\mathcal{B}$ creates the BasicPub public key $K_{Pub} = < G_1, G_2, e, n, P, P_{Pub}, Q_{ID}, H_2>$ by setting $P_{Pub} = aP$, $Q_{ID} = bP$.
Observe that, the private key associated to K_{Pub} is $d_{ID} = a^{-1} Q_{ID} = a^{-1}bP$.

H_2-queries: At any time algorithm $\mathcal{A}$ may issue queries to H_2. To respond to these queries algorithm $\mathcal{B}$ maintains a list of pairs called the H_2-list. Each entry in the list is a pair of the form $< X_j, H_j >$. Initially the list is empty.

To respond to query X_j algorithm $\mathcal{B}$ does the following:

1. If the query X_j already appears on the H_2-list, then he responds with $H_2(X_j) = H_j$.
2. Otherwise, algorithm $\mathcal{B}$ just picks a random string $H_j \in \{0, 1\}^n$ and adds the tuple $<X_j, H_j>$ to the list. It responds to algorithm $\mathcal{A}$ with $H_2(X_j) = H_j$.

Challenge: Algorithm $\mathcal{A}$ outputs two equal length plaintext M_0, M_1 in which it wishes to be challenged. Algorithm $\mathcal{B}$ then picks a random string $R \in \{0, 1\}^n$ and defines C to be the ciphertext, C = <U, V> where U = P and V = R. Algorithm $\mathcal{B}$ picks a random bit $b \in \{0,1\}$ and gives C, encryption of M_b, as the challenge to algorithm $\mathcal{A}$.
Note that, the decryption of C is
$V \oplus H_2(e(u, d_{ID})) = V \oplus H_2(e(P, a^{-1}bP)) = V \oplus H_2(e(P,P)^{a^{-1}b}) = V \oplus H_2(D)$.
We set $M_b = V \oplus H_2(D)$.

Guess: Algorithm $\mathcal{A}$ outputs a guess b' $\in\{0,1\}$ for b.

It is easy to see that $\mathcal{A}$'s view is identical to its view in the real attack. The setup is as in the real attack. Since a and b are random in Z_p^* so is the challenge, as the challenged ciphertext C = <U,V> where U = P and $V\in\{0,1\}^n$. U = P imply U = $a^{-1}aP$ i.e. adversary $\mathcal{A}$ chooses r = $a^{-1}\in Z_p^*$, and his choice is justified as $\mathcal{A}$ sets the game in such a way that any response of $\mathcal{B}$ enables him to output the right solution of the problem given to him. Since, P is a random in G_1 and therefore the resulting encryption message, which is exclusive-or of two random strings in $\{0,1\}^n$, is also random plaintext.
Thus,

$$Adv_A(k) = \varepsilon = \left|\Pr[b' = b] - \frac{1}{2}\right|$$

It still remains to calculate the probability that algorithm $\mathcal{B}$ outputs the correct result.
The adversary $\mathcal{A}$ gains no advantage in distinguishing M_0, M_1 if it has not asked for $e(P,P)^{a^{-1}b}$,which is equal to D, to H_2. Let H denote the event that at the end of the simulation D appears in a pair on H_2-lists. Let $\Pr[H] = \delta$. If D does not appear in H_2-lists, then the decryption of C is independent of $\mathcal{A}$'s view, since H_2(D) is a random string in $\{0,1\}^n$ independent of $\mathcal{A}$'s view. Thus, $\Pr[M' = M \mid \neg H] \geq \frac{1}{2^n}$

Then,

$$\Pr[b' = b] = \frac{1}{2} \pm Adv_A(k) = \frac{1}{2} \pm \varepsilon$$

Now,

$$\Pr[b' = b] = \Pr[b' = b \mid H]\cdot\Pr[H] + \Pr[b' = b \mid \neg H]\cdot\Pr[\neg H]$$

$$\leq \Pr[H] + \Pr[b' = b \mid \neg H]\cdot\Pr[\neg H]$$

$$\leq \delta + \frac{1}{2^n}(1 - \delta)$$

$$\therefore \Pr[H] = \delta \geq \delta - \frac{\delta}{2^n} \geq \frac{1}{2} \pm \varepsilon - \frac{1}{2^n} \approx \varepsilon - \frac{1}{2^n}$$

$$\therefore \Pr[H] \geq \varepsilon - \frac{1}{2^n}$$

Also, since we pick a random element from H_2-list, the probability that algorithm $\mathcal{B}$ produces the right answer is at least

$$\Pr[H] \geq \frac{(\varepsilon - \frac{1}{2^n})}{q_{H_2}} \approx \frac{\varepsilon}{q_{H_2}} .$$

Note that, if algorithm $\mathcal{A}$ answers correctly, then V $\oplus$ M'= H_2(D). So algorithm $\mathcal{B}$ could scan through the H_2-list, and pick a random pair <X_j, H_j> such that H_j = H_2(D), and output X_j instead of picking a random pair from the entire H_2-list. ♣

In order to come up with the total concrete security, we assume that $q_E = q_D$ (since extraction and decryption operations have almost same computational complexity). We also bound all hash queries q_i with q_H.

In Theorem 2 of [10], Zhang, Safavi-Naini and Susilo have shown that BDHP, BIDHP and BSDHP are all polynomial time equivalent. Using this result we infer, that *fullM-IBE* is secure so long as the BDHP is difficult. Therefore, from Lemma1, Lemma2 and Lemma3, we get:
If there exists an IND-ID-CCA adversary against algorithm $\mathcal{A}$ that has advantage ε against *full M-IBE,* then there is an algorithm $\mathcal{B}$ that can solve BDHP with advantage at least $\frac{\varepsilon}{{q_H}^2}$.

We now prove a theorem which improves the tightness in the above theorem. Here we rely on a stronger assumption, namely, the BIDDHP. To prove the theorem we work on the line of [5].

Theorem2: Let $\mathcal{A}$ be a t time IND-CPA adversary against BasicPub-Wang with advantage atmost ε making atmost q_{H_2} hash queries. Then there is an algorithm $\mathcal{B}$ that can solve BIDDHP with advantage ε and running time t' $\approx$ t.

Proof: Algorithm $\mathcal{B}$ is given an input the BIDDH parameters $<G_1, G_2, e>$ produced by $\mathcal{IG}$ and a random instance $<P, aP, bP, cP, T>$ of the BIDDH problem for these parameters i.e., $P \in_R G_1^*$ where a, b, $c \in_R \mathbb{Z}_p^*$. $|G_1| = p = |G_2|$. Algorithm $\mathcal{B}$ uses algorithm $\mathcal{A}$ to solve the BIDDHP as follows:

Setup: Algorithm $\mathcal{B}$ creates the BasicPub public key $K_{Pub} = < G_1, G_2, e, n, P, P_{Pub}, Q_{ID}, H_2>$ by setting $P_{Pub} = aP$, $Q_{ID} = bP$.
Observe that, the private key associated to K_{Pub} is $d_{ID} = a^{-1} Q_{ID} = a^{-1}bP$.

H_2-queries: At any time algorithm $\mathcal{A}$ may issue queries to H_2. To respond to these queries algorithm $\mathcal{B}$ maintains a list of pairs called the H_2-list. Each entry in the list is a pair of the form $< X_j, H_j >$. Initially the list is empty.

To respond to query X_j algorithm $\mathcal{B}$ does the following:

1. If the query X_j already appears on the H_2-list, then he responds with $H_2(X_j) = H_j$.

2. Otherwise, algorithm $\mathcal{B}$ just picks a random string $H_j \in \{0, 1\}^n$ and adds the tuple $<X_j, H_j>$ to the list. It responds to algorithm $\mathcal{A}$ with $H_2(X_j) = H_j$.

Challenge: $\mathcal{A}$ outputs two equal length plaintext M_0, M_1 in which it wishes to be challenged. Algorithm $\mathcal{B}$ returns as the challenge ciphertext $C = \langle cP, M_b \oplus H_2(T)\rangle$, where $b \in_R \{0,1\}$.

Note that, the decryption of C is $V \oplus H_2(e(U, d_{ID})) = V \oplus H_2(e(cP, a^{-1}bP)) = V \oplus H_2(e(P,P)^{a^{-1}bc})$

Guess: Algorithm $\mathcal{A}$ outputs its guess b' for b. Algorithm $\mathcal{B}$ returns 1 if b = b' and 0 otherwise.

Note that, we say an algorithm $\mathcal{D}(t,\varepsilon)$ breaks BIDDHP on (G_1, G_2) if it runs in time at most t and

$$| \Pr[\mathcal{D}(P, aP, bP, cP, e(P,P)^{a^{-1}bc}) = 1] - \Pr[\mathcal{D}(P, aP, bP, cP, T) = 1] | \geq \varepsilon .$$

where the probability is computed over the random choices of the parameters, and the random bits of $\mathcal{D}$. The distribution on the left side is called BIDH distribution and is denoted by $\mathcal{P}_{\mathcal{BIDH}}$, while the distribution on the right is called random BIDH distribution and is denoted by $\mathcal{R}_{\mathcal{BIDH}}$.

In the above game, algorithm $\mathcal{B}$ is simulating a real attack environment for $\mathcal{A}$. If the random instance is from $\mathcal{R}_{\mathcal{BIDH}}$, then Pr[b' = b] = ½, since in this case the distribution of the ciphertext C is independent of the bit b. Otherwise, the instance comes from $\mathcal{P}_{\mathcal{BIDH}}$, and C is valid encryption of M_b. Therefore Pr[b' = b] = ½ + ε by definition of $\mathcal{A}$.
Therefore, $|\Pr[\mathcal{B}(\mathcal{P}_{\mathcal{BIDH}})=1 - \Pr[\mathcal{B}(\mathcal{R}_{\mathcal{BIDH}})=1]| = [1/2 + \varepsilon - 1/2] = \varepsilon$. ♣

With this second tightness improvement, we obtain that *fullM-IBE* scheme is $(t, q_H, q_D, \varepsilon)$ IND-ID-CCA secure if the BIDDHP problem on (G_1, G_2) is

$$\left(t + c_{G_1}(2q_D + q_H) + q_H O(\log^3 q + \log q), \frac{\varepsilon}{q_H}\right) \text{secure.}$$

Here, we got rid of a q_H factor in the security reduction at the cost of relying on a stronger assumption.

Conclusion: In this paper we present another proof that the IBE scheme *fullM-IBE* of Wang and Cao [8] is secure against chosen ciphertext attack. We remove an anomaly in the security proof by Wang and Cao which is based on mBDHP. We base our proof on the hardness of BDHP which is stronger than mBDHP. We also obtain a better tightness improvement using BIDDHP, which is a stronger assumption.

References:

1) D.Boneh and M.Franklin, “Identity-based encryption from weil pairing”, In Proc. Of CRYPTO 2001, LNCS # 2139, pp.213-229. Springer-Verlag, 2001.

2) M.Choudary Gorantla, Raju Gangishetti and Ashutosh Saxena, “A Survey On ID-Based Cryptographic Primitives”, http://eprint.iarc.org/2005/094.pdf.

3) E.Fujisaki and T.Okamoto, “Secure integration of asymmetric and symmetric encryption schemes”, In Advances in Cryptology-CRYPTO 1999, LNCS # 1666, pp. 537-554, Springer-Verlag, 1999.

4) E.Fujisaki and T.Okamoto, “How to enhance the security of public-key encryption at minimum cost”, IEICE Trans. Fundamentals, E83-9(1):24-32, 2000.

5) D.Galindo, “Boneh-Franklin identity based encryption revisited”, In proc. of ICALP 2005, LNCS # 3580, pp. 791-802. Springer-Verlag, 2003.

6) A.Shamir, “Identity-based cryptosystems and signature schemes”, In Proc. Of CRYPTO 1984, LNCS # 196, pp.47-53. Springer-Verlag, 1984. Also available on http://www.iseca.org/downloads/shamir47.pdf.

7) Shengbao Wang, “Practical Identity-Based Encryption (IBE) in Multiple PKG Environments and Its Applications”, http://eprint.iacr.org/2007/100.pdf.

8) Shengbao Wang and Zhenfu Cao, “Practical Identity-Based Encryption (IBE) in Multiple PKG Environments and Its Applications”, http://eprint.iacr.org/2007/100.pdf.

9) Sunder Lal and Priyam Sharma, “Security proof for Shengbao Wang’s identity-based encryption scheme”, http://eprint.iarc.org/2007/316.pdf.

10) F.Zhang, R.Safavi-Naini, and W.Susilo, “An efficient signature scheme from bilinear pairings and its applications”, In International Workshop on Practice and Theory in Public Key Cryptography-PKC’2004. LNCS # 2947, pp.277-290, Springer-Verlag, 2004.