An Identity-Based Signature from Gap Diffie-Hellman Groups

Jae Choon Cha[1] and Jung Hee Cheon[2]

[1] Department of Mathematics
Korea Advanced Institute of Science and Technology
Taejon, 305–701, Korea
jccha@knot.kaist.ac.kr
http://knot.kaist.ac.kr/~jccha
[2] Information and Communications University (ICU)
Taejon, 305–732, Korea
jhcheon@icu.ac.kr
http://vega.icu.ac.kr/~jhcheon

Abstract. In this paper we propose an identity(ID)-based signature scheme using gap Diffie-Hellman (GDH) groups. Our scheme is proved secure against existential forgery on adaptively chosen message and ID attack under the random oracle model. Using GDH groups obtained from bilinear pairings, as a special case of our scheme, we obtain an ID-based signature scheme that shares the same system parameters with the ID-based encryption scheme (BF-IBE) by Boneh and Franklin [BF01], and is as efficient as the BF-IBE. Combining our signature scheme with the BF-IBE yields a complete solution of an ID-based public key system. It can be an alternative for certificate-based public key infrastructures, especially when efficient key management and moderate security are required.

Keywords: ID-based signature, GDH group, Elliptic curve, Weil pairing

1 Introduction

In 1984, Shamir asked for identity(ID)-based encryption and signature schemes to simplify key management procedures of certificate-based public key infrastructures (PKIs) [Sha84]. Since then, several ID-based encryption schemes and signature schemes have been proposed based on the integer factorization problem (IFP) [DQ86, Tan87, TI89, MY91]. Recently, Boneh and Franklin [BF01] proposed an ID-based encryption scheme (BF-IBE) based on bilinear maps on an elliptic curve. BF-IBE scheme is considered as the first practical ID-based encryption, but it was not reported whether it is possible to design a signature version of BF-IBE in [BF01]. Actually no concrete ID-based signature scheme was proposed on elliptic curves. We remark that an ID-based signature scheme based on pairings was proposed in [SOK01] but no security argument was given.

Y.G. Desmedt (Ed.): PKC 2003, LNCS 2567, pp. 18–30, 2003.

In this paper, we propose an ID-based signature scheme using gap Diffie-Hellman (GDH) groups. Its security is based on the hardness of computational Diffie-Hellman problem (CDHP). More precisely, under the random oracle model, our scheme is proved to be secure against existential forgery on adaptively chosen message and ID attack, which is a natural ID-based version of the standard adaptively chosen message attack (see Section 3 for details), assuming CDHP is intractable.

Using GDH groups obtained from bilinear pairings, as a special case of our scheme, we obtain an ID-based signature scheme that shares the same system parameters with BF-IBE. It is as efficient as BF-IBE. We remark that BF-IDE is indistinguishably secure against adaptively chosen ciphertext attack, assuming the hardness of the bilinear Diffie-Hellman problem (BDHP), which is believed to be more difficult than CDHP that our scheme is based on. BF-IBE and our scheme form a provably secure system which fully enjoys the functionals originally suggested by Shamir [Sha84].[1] Our scheme can also be used to realize proxy signatures by using the whole ID-based scheme for a single user, in a similar way to delegation of duties on encryption [BF01].

A problem of ID-based signatures is the difficulty of providing non-repudiation property. In all previous schemes based on IFP, one private key generator (PKG) knows the whole secret and so can generate valid signatures of any user. Thus non-repudiation property is obtained only when the PKG is completed trusted. On the other hand, in our scheme the secret can be shared to several parties through a threshold scheme. If we apply an (n, k)-threshold scheme to our scheme, at least k-parties out of n PKG's should collude to generate a valid signature and the number k can be as large as we want. That is, our scheme provides stronger non-repudiation property than previous ID-based schemes.

The rest of the paper is organized as follows: In Section 2, we introduce related mathematical problems and describe our scheme. In Section 3, we present a natural attack model and security proof of our signature scheme. In Section 4, we discuss the implementation issues of BF-IBE and our scheme. We conclude in Section 5.

2 Our Identity-Based Signature Scheme

In this section we propose an ID-based signature scheme that can be built on any group whose computational Diffie-Hellman problem is hard but decisional Diffie-Hellman problem is solved. We start with a formal definition of such groups.

[1] After we had submitted an earlier version[CC01] of this paper, some other schemes were also announced as preprints. Paterson's scheme [Pat02] was proposed with a brief security arguments but no rigorous proof. Hess's scheme [Hes02] was claimed to be provably secure with a proof in the case of fixed ID. It is interesting that all of the schemes are different. In this version of our paper, the security proof of the earlier version is extended to the case of adaptively chosen ID and the base problems are clarified.

2.1 Gap Diffie-Hellman (GDH) Groups

Let G be a cyclic group generated by P, whose order is a prime ℓ. We assume that multiplication and inversion in G can be computed in a unit time. We are interested in the following mathematical problems. View G as an additive group, and let a, b, and c be elements of $\mathbb{Z}/\ell$.

1. **Computation Diffie-Hellman Problem (CDHP).** Given (P, aP, bP), compute abP.
2. **Decisional Diffie-Hellman Problem (DDHP).** Given (P, aP, bP, cP), decide whether $c = ab$ in $\mathbb{Z}/\ell$. (If so, (P, aP, bP, cP) is called a valid Diffie-Hellman tuple.)

We call G a *GDH group* if DDHP can be solved in polynomial time but no probabilistic algorithm can solve CDHP with non-negligible advantage within polynomial time [OP01, BLS01].

2.2 The Scheme

Let G be a group of prime order ℓ in which DDHP can be solved.

1. **Setup**. Choose a generator P of G, pick a ramdom $s \in \mathbb{Z}/\ell$, set $P_{pub} = sP$, and choose cryptographic hash functions $H_1\colon \{0,1\}^* \times G \to \mathbb{Z}/\ell$ and $H_2\colon \{0,1\}^* \to G$. The system parameter is (P, P_{pub}, H_1, H_2). The master key is s. We remark that H_1 and H_2 will be viewed as random oracles in our security proof.
2. **Extract**. Given an identity ID, the algorithm computes $D_{\mathrm{ID}} = sH_2(\mathrm{ID})$ and output it as the private key associated to ID. We remark that $Q_{\mathrm{ID}} = H_2(\mathrm{ID})$ plays the role of the associated public key.
3. **Sign**. Given a secret key D_{ID} and a message m, pick a random number $r \in \mathbb{Z}/\ell$ and output a signature $\sigma = (U, V)$ where $U = rQ_{\mathrm{ID}}$, $h = H_1(m, U)$, and $V = (r + h)D_{\mathrm{ID}}$.
4. **Verify**. To verify a signature $\sigma = (U, V)$ of a message m for an identity ID, check whether $(P, P_{pub}, U + hQ_{\mathrm{ID}}, V)$, where $h = H_1(m, U)$, is a valid Diffie-Hellman tuple.

This completes the description of our ID-based signature scheme. Consistency is easily proved as follows: If $\sigma = (U, V)$ is a valid signature of a message m for an identity ID, then $U = rQ_{\mathrm{ID}}$ and $V = (r + h)D_{\mathrm{ID}}$ for $r \in \mathbb{Z}/\ell$ and $h = H_1(m, U)$. Thus

$$\begin{aligned}(P, P_{pub}, U + hQ_{\mathrm{ID}}, V) &= (P, P_{pub}, (r+h)Q_{\mathrm{ID}}, (r+h)D_{\mathrm{ID}}) \\ &= (P, sP, (r+h)Q_{\mathrm{ID}}, s(r+h)Q_{\mathrm{ID}})\end{aligned}$$

as desired.

We will prove that if G is a GDH group, i.e., if CDHP is hard, then our signature scheme is secure against existential forgery on a natural generalization of the standard adaptively chosen message attack for ID-based schemes, in Section 3 (see Theorems 3 and 5).

2.3 Relationship with BF-IBE

Although we described our scheme as one that is built on a given GDH-group, this can be easily transformed into one that can share the setup algorithm and resulting system parameters with BF-IBE [BF01] in a formal manner. Indeed, we will describe a variant of the **Setup** algorithm of our scheme and observe that we can view that of BF-IBE as a special case of this variant. For this we need to introduce a notion of a parameter generator which outputs GDH groups.

GDH Parameter Generator. A polynomial time probabilistic algorithm $\mathcal{IG}_{GDH}$ is called a *GDH parameter generator* if for a given positive integer k, which plays the role of a security parameter, it outputs (descriptions of) a cyclic group G of prime order and a polynomial time algorithm $\mathcal{D}$ which solves DDHP in G. We will always view G as an additive group. We denote the output of $\mathcal{IG}_{GDH}$ by $\mathcal{IG}_{GDH}(1^k)$.

Gap Diffie-Hellman Assumption. Let $\mathcal{IG}_{GDH}$ be a GDH parameter generator, and let $\mathcal{A}$ be an algorithm whose input consists of a group G of prime order ℓ, an algorithm $\mathcal{D}$ solving DDHP, a generator P of G, aP and bP $(a, b \in \mathbb{Z}/\ell)$ and whose output is an element of G that is expected to be abP. As usual, the advantage of $\mathcal{A}$ with respect to $\mathcal{IG}_{GDH}$ is defined to be

$$\Pr\left[\mathcal{A}(G, \mathcal{D}, P, aP, bP) = abP \;\middle|\; (G, \mathcal{D}) \leftarrow \mathcal{IG}_{GDH}(1^k),\ P \overset{R}{\leftarrow} G^*,\ a, b \overset{R}{\leftarrow} \mathbb{Z}/\ell\right].$$

$\mathcal{IG}_{GDH}$ is said to satisfy the *GDH assumption* if any polynomial time algorithm $\mathcal{A}$ has advantage $\leq 1/f(k)$ for all polynomial f, that is, no polynomial time algorithm can solve CDHP with non-negligible advantage.

A Variant of Setup. Let $\mathcal{IG}_{GDH}$ be a GDH parameter generator. We describe another setup algorithm for our scheme as follows.

Setup′. Given a security parameter k, it works as follows:

1. Run $\mathcal{IG}_{GDH}$ on input k and let $(G, \mathcal{D})$ be the output.
2. Choose P, s, H_1 and H_2 as in the **Setup** algorithm described above, and let $P_{pub} = sP$. The system parameter is $(G, \mathcal{D}, P, P_{pub}, H_1, H_2)$. The master key is s.

In [BF01], Boneh and Franklin used a *BDH parameter generator* to build an ID-based public key cryptosystem, which is defined to be an algorithm that runs in polynomial time in a given security parameter k, and outputs (descriptions of) two groups G_1, G_2 of prime order ℓ and a computable non-degenerated bilinear map $\hat{e}\colon G_1 \times G_1 \to G_2$. The scheme in [BF01] is proved to be secure if the *bilinear Diffie-Hellman problem (BDHP)*, which asks to compute $\hat{e}(P, P)^{abc}$ for a given (P, aP, bP, cP), is infeasible. Formally speaking, a BDH parameter generator

$\mathcal{IG}_{BDH}$ is said to satisfy the *bilinear Diffie-Hellman (BDH) assumption* if the advantage

$$\Pr\left[\mathcal{A}(G_1, G_2, \hat{e}, P, aP, bP, cP) = \hat{e}(P, P)^{abc} \;\middle|\; \begin{array}{l} (G_1, G_2, \hat{e}) \leftarrow \mathcal{IG}_{BDH}(1^k), \\ P \overset{R}{\longleftarrow} G_1 - \{0\}, \\ a, b, c \overset{R}{\longleftarrow} \mathbb{Z}/\ell \end{array}\right]$$

is negligible for any polynomial time algorithm $\mathcal{A}$.

We recall two well-known facts: (1) An algorithm $\mathcal{D}$ that solves DDHP in G_1 can be obtained using the non-degenerated bilinear map $\hat{e}$, since $\hat{e}(aP, bP) = \hat{e}(P, abP)$ implies that (P, aP, bP, cP) is a valid Diffie-Hellman tuple. (2) BDHP is solved if so is CDHP. An immediate consequence is that a BDH parameter generator $\mathcal{IG}_{BDH}$ satisfying the BDH assumption can also viewed as a GDH parameter generator $\mathcal{IG}_{GDH}$ satisfying the GDH assumption; $\mathcal{IG}_{GDH}$ runs $\mathcal{IG}_{BDH}$ on the same security parameter k and outputs $G(= G_1)$ and $\mathcal{D}$.

This shows that the setup algorithm of the ID-based encryption scheme described in [BF01] can be shared with our scheme; the system parameters G_1, P, P_{pub}, H_1, and H_2 generated by the setup algorithm of the scheme in [BF01] can also be used for our scheme without any loss of security.

3 Security Proof

In this section we prove the security of our signature scheme, assuming the hardness of CDHP.

3.1 Attack Model for ID-based Signature Schemes

The most general known notion of security of a non-ID-based signature scheme is security against existential forgery on adaptively chosen message attacks; in this model, an adversary wins the game if he outputs a valid pair of a message and a signature, where he is allowed to ask the signer to sign any message except the output. We consider the following natural generalization of this notion, which is acceptable as a standard model of security for ID-based signature schemes. We say that an ID-based signature scheme, which consists of four algorithms **Setup**, **Extract**, **Sign**, and **Verify** playing the same role as ours, is *secure against existential forgery on adaptively chosen message and* ID *attacks* if no polynomial time algorithm $\mathcal{A}$ has a non-negligible advantage against a challenger $\mathcal{C}$ in the following game:

1. $\mathcal{C}$ runs **Setup** of the scheme. The resulting system parameters are given to $\mathcal{A}$.
2. $\mathcal{A}$ issues the following queries as he wants:
   (a) Hash function query. $\mathcal{C}$ computes the value of the hash function for the requested input and sends the value to $\mathcal{A}$.
   (b) **Extract** query. Given an identity ID, $\mathcal{C}$ returns the private key corresponding to ID which is obtained by running **Extract**.

   (c) **Sign** query. Given an identity ID and a message m, $\mathcal{C}$ returns a signature which is obtained by running **Sign**.

3. $\mathcal{A}$ outputs (ID, m, σ), where ID is an identity, m is a message, and σ is a signature, such that ID and (ID, m) are not equal to the inputs of any query to **Extract** and **Sign**, respectively. $\mathcal{A}$ wins the game if σ is a valid signature of m for ID.

For notational purposes, in the proof of the security of our scheme, the result of the **Sign** query (asked by $\mathcal{A}$ in Step 2) will be denoted by $(\mathrm{ID}, m, U, h, V)$ where (U, V) is the output of the signing algorithm of our scheme and $h = H_2(m, U)$, similarly to the convention of [PS00].

3.2 Our Signature Scheme and CDHP

Consider the following variant of the above game: First we fix an identity ID. In Step 1, $\mathcal{C}$ gives to $\mathcal{A}$ system parameters together with ID, and in Step 3, $\mathcal{A}$ must output the given ID (together with a message and a signature) as its final result. If no polynomial time algorithm $\mathcal{A}$ has non-negligible advantage in this game, we say that the signature scheme is secure under existential forgery on adaptively chosen message and *given* ID attacks. The first step of our proof is to reduce the problem to this case.

Lemma 1 *If there is an algorithm $\mathcal{A}_0$ for an adaptively chosen message and* ID *attack to our scheme with running time t_0 and advantage ϵ_0, then there is an algorithm $\mathcal{A}_1$ for an adaptively chosen message and given* ID *attack which has running time $t_1 \leq t_0$ and advantage $\epsilon_1 \leq \epsilon_0(1 - \frac{1}{\ell})/q_{H_2}$, where q_{H_2} is the maximum number of queries to H_2 asked by $\mathcal{A}_0$. In addition, the numbers of queries to hash functions, **Extract**, and **Sign** asked by $\mathcal{A}_1$ are the same as those of $\mathcal{A}_0$.*

Proof. We may assume that for any ID, $\mathcal{A}_0$ queries $G(\mathrm{ID})$ and **Extract**(ID) at most once, without any loss of generality. Our algorithm $\mathcal{A}_1$ is as follows:

1. Choose $r \in \{1, \ldots, q_{H_2}\}$ randomly. Denote by ID_i the input of the i-th query to H_2 asked by $\mathcal{A}_0$. Let ID'_i be ID if $i = r$, and ID_i otherwise. Define $H'_2(\mathrm{ID}_i)$, $\mathbf{Extract}'(\mathrm{ID}_i)$, $\mathbf{Sign}'(\mathrm{ID}_i, m)$ to be $H_2(\mathrm{ID}'_i)$, $\mathbf{Extract}(\mathrm{ID}'_i)$, $\mathbf{Sign}(\mathrm{ID}'_i, m)$, respectively.
2. Run $\mathcal{A}_0$ with the given system parameters. $\mathcal{A}_1$ responds to A_0's queries to H_1, H_2, **Extract**, and **Sign** by evaluating H_1, H'_2, $\mathbf{Extract}'$, and $\mathbf{Sign}'$, respectively. Let the output of $\mathcal{A}_0$ be $(\mathrm{ID}_{out}, m, \sigma)$.
3. If $\mathrm{ID}_{out} = \mathrm{ID}$ and (ID, m, σ) is valid, then output (ID, m, σ). Otherwise output `fail`.

Since the distributions produced by H'_2, $\mathbf{Extract}'$, and $\mathbf{Sign}'$ are indistinguishable from those produced by H_2, **Extract**, and **Sign** of our scheme, $\mathcal{A}_0$ learns nothing from query results, and hence

$$\Pr[(\mathrm{ID}_{out}, m, \sigma) \text{ is valid}] \geq \epsilon.$$

Since H_2 is a random oracle, the probability that the output $(\mathrm{ID}_{out}, m, \sigma)$ of $\mathcal{A}_0$ is valid without any query of $H_2'(\mathrm{ID}_{out})$ is negligible. Explicitly,

$$\Pr[\mathrm{ID}_{out} = \mathrm{ID}_i \text{ for some } i \mid (\mathrm{ID}_{out}, m, \sigma) \text{ is valid}] \geq 1 - \frac{1}{\ell}.$$

Since r is independently and randomly chosen, we have

$$\Pr[\mathrm{ID}_{out} = \mathrm{ID}_r \mid \mathrm{ID}_{out} = \mathrm{ID}_i \text{ for some } i] \geq \frac{1}{q_{H_2}}.$$

Combining these,

$$\Pr[\mathrm{ID}_{out} = \mathrm{ID}_r = \mathrm{ID} \text{ and } (\mathrm{ID}, m, \sigma) \text{ is valid}] \geq \epsilon \cdot \left(1 - \frac{1}{\ell}\right) \cdot \frac{1}{q_{H_2}}$$

as desired.

We remark that the algorithm $\mathcal{A}_1$ can be viewed as an adversary to the non-ID-based scheme obtained by fixing an ID in our ID-based scheme, which is allowed to access the extraction oracle to obtain secret keys associated to identities different from the fixed one, as well as the signing oracle and hash functions.

Now we are ready to construct an algorithm which solves CDHP, assuming the existence of $\mathcal{A}_1$.

Lemma 2 *If there is an algorithm $\mathcal{A}_1$ for an adaptively chosen message and given* ID *attack to our scheme which queries H_1, H_2, **Sign**, and **Extract** at most q_{H_1}, q_{H_2}, q_S, and q_E times, respectively, and has running time t_1 and advantage $\epsilon_1 \geq 10(q_S+1)(q_S+q_{H_1})/\ell$, then CDHP can be solved with probability $\epsilon_2 \geq 1/9$ within running time $t_2 \leq 23 q_{H_1} t_1/\epsilon_1$.*

Proof. We may assume that for any ID, $\mathcal{A}_1$ queries $H_2(\mathrm{ID})$ and **Extract**(ID) at most once as before, and $\mathcal{A}_1$ queries $H_2(\mathrm{ID})$ before ID is used as (part of) an input of any query to H_1, **Extract**, and **Sign**, by using a simple wrapper of $\mathcal{A}_1$. Our algorithm $\mathcal{A}_2$ described below computes abP for a randomly given instance (P, aP, bP) where P is a generator of G.

1. Fix an identity ID, and put $P_{pub} = aP$. Choose randomly $x_i \in \mathbb{Z}/\ell$, $y_j \in \mathbb{Z}/\ell$, and $h_j \in \mathbb{Z}/\ell$ for $i = 1, \ldots, q_G$, $j = 1, \ldots, q_S$. Denote by ID_i, ID_{i_k}, and (ID_{i_j}, m_j) the inputs of the i-th H_2 query, the k-th **Extract** query, and the j-th **Sign** query asked by $\mathcal{A}_1$, respectively. Define

$$H_2''(\mathrm{ID}_i) = \begin{cases} bP, & \text{if } \mathrm{ID}_i = \mathrm{ID} \\ x_i P, & \text{otherwise,} \end{cases}$$

$$\mathbf{Extract}''(\mathrm{ID}_{i_k}) = x_{i_k}(bP),$$

$$\mathbf{Sign}''(\mathrm{ID}_{i_j}, m_j) = (\mathrm{ID}_{i_j}, m_j, U_j, h_j, V_j)$$

$$\text{where } U_j = y_j P - h_j H_2''(\mathrm{ID}_{i_j}),\ V_j = y_j(bP).$$

2. We apply the oracle replay attack which was invented by Pointcheval and Stern in [PS96, PS00]. As done in [PS00, Lemma 4 and Theorem 3] for adaptively chosen message attacks to non-ID-based signature schemes, a collusion of $\mathcal{A}_1$, H''_2, **Extract**$''$, and **Sign**$''$ defines a machine $\mathcal{B}$ performing a "no-message attack" to the non-ID-based scheme obtained by fixing ID in the original scheme. ($\mathcal{B}$ is still allowed to ask queries to H_1.)
   We need to take care of a nasty problem of collisions of the query result of **Sign**$''$ and H_1, as mentioned in [PS00, Proof of Lemma 4]. Whenever **Sign**$''(\mathrm{ID}_{i_j}, m_j)$ is queried, $\mathcal{B}$ stores the output h_j as the value of $H_1(m_j, U_j)$. This may cause some "collision"; a query result of **Sign**$''$ may produce a value of H_1 that is inconsistent with other query results of **Sign**$''$ or H_1. In this case $\mathcal{B}$ just outputs `fail` and exits.
3. If no collisions have appeared, $\mathcal{B}$ outputs a valid message-signature pair, which is expected to be valid for the fixed ID, without accessing any oracles except H_1. Here P and P_{pub} are used as system parameters for $\mathcal{A}_1$. By replays of $\mathcal{B}$ with the same random tape but different choices of H, as done in the *forking lemma* [PS00, Lemma 2], we obtain signatures $(\mathrm{ID}, m, U, h, V))$ and $(\mathrm{ID}, m, U, h', V')$ which are expected to be valid ones with respect to hash functions H_1 and H'_1 having different values $h \neq h'$ on (m, U), respectively.
4. If both outputs are expected ones, then compute $(h - h')^{-1}(V - V')$ and output it. Otherwise, output `fail`.

It is straightforward to verify that **Extract**$''$ and **Sign**$''$ produce "valid" secret keys and signatures. Furthermore, since H''_2, **Extract**$''$, and **Sign**$''$ generate random distribution and are indistinguishable from H_2, **Extract**, and **Sign** of the original scheme, $\mathcal{A}_1$ learns nothing from query results. Therefore $\mathcal{B}$ works as expected if no collisions appear in Step 2. Intuitively, since U_j is random, the possibility of collisions is negligible; in [PS00, Proof of Lemma 4], this probability was computed explicitly, and furthermore, it was proved that the oracle replay in Step 3 produces valid signatures $(\mathrm{ID}, m, U, h, V))$ and $(\mathrm{ID}, m, U, h', V')$ with expected properties such that that $m = m'$, $U = U'$, and $h \neq h'$ with probability $\geq 1/9$.

Now a standard argument for outputs of the forking lemma can be applied as follows: since both are valid signatures, $(P, P_{pub}, U + hH''_2(\mathrm{ID}), V)$ and $(P, P_{pub}, U + h'H''_2(\mathrm{ID}), V')$ are valid Diffie-Hellman tuples. In other words, $V = a(U + hbP)$ and $V' = a(U + h'bP)$. Subtracting the equations, $V - V' = (h - h')abP$ and $abP = (h - h')^{-1}(V - V')$ as desired.

The total running time t_2 of $\mathcal{A}_2$ is equal to the running time of the forking lemma [PS00, Lemma 4] which is bounded by $23q_{H_1}t_1/\epsilon_1$, as desired.

Combining the above lemma, we have

Theorem 3 *If there is an algorithm $\mathcal{A}_0$ for an adaptively chosen message and* ID *attack to our scheme which queries H_1, H_2, **Sign**, and **Extract** at most q_{H_1}, q_{H_2}, q_S, and q_E times, respectively, and has running time t_0 and advantage $\epsilon_0 \geq 10(q_S + 1)(q_S + q_{H_1})q_{H_2}/(\ell - 1)$, then CDHP can be solved with probability $\geq 1/9$ and within running time $\leq \dfrac{23q_{H_1}q_{H_2}t_0}{\epsilon_0(1 - \frac{1}{\ell})}$.*

Remark. Using another variant of the forking lemma [PS00, Theorem 3] instead of [PS00, Lemma 4], we have the following results:

Lemma 4 *If there is an algorithm $\mathcal{A}_1$ for an adaptively chosen message and given* ID *attack to our scheme which queries H_1, H_2,* ***Sign****, and* ***Extract*** *at most q_{H_1}, q_{H_2}, q_S, and q_E times, respectively, and has running time t_1 and advantage $\epsilon_1 \geq 10(q_S+1)(q_S+q_{H_1})/\ell$, then CDHP can be solved within expected time $\leq 120686 q_{H_1} t_1/\epsilon_1$.*

Theorem 5 *If there is an algorithm $\mathcal{A}_0$ for an adaptively chosen message and* ID *attack to our scheme which queries H_1, H_2,* ***Sign****, and* ***Extract*** *at most q_{H_1}, q_{H_2}, q_S, and q_E times, respectively, and has running time t_0 and advantage $\epsilon_0 \geq 10(q_S+1)(q_S+q_{H_1})q_{H_2}/(\ell-1)$, then CDHP can be solved within expected time $\leq \dfrac{120686 q_{H_1} q_{H_2} t_0}{\epsilon_0(1-\frac{1}{\ell})}$.*

4 Implementation Issues

At the present time, no candidate for GDH group is known except some (hyper)elliptic curves, which are equipped with a bilinear map such as the Weil pairing or the Tate pairing. In this section, we discuss implementation issues for these groups.

4.1 Bilinear Maps

Let E be an elliptic curve over $\mathbb{F}_q$, $q = p^n$, p a prime. Let $E[\ell] = \{P \in E | \ell P = O\}$ denote the ℓ-torsion subgroup of E for a prime ℓ. The Weil pairing is a map $e\colon E[\ell] \times E[\ell] \to \mathbb{F}_{q^\alpha}^*$ for the least positive integer α, called an exponent, such that ℓ divides $q^\alpha - 1$. Assume ℓ divides $E(\mathbb{F}_q)$ with small cofactor. If we have a non-$\mathbb{F}_q$-rational map $\phi\colon E \to E$, then $G = E(\mathbb{F}_q)[\ell]$ is a group admitting an efficiently computable non-degenerated bilinear map $\hat{e}\colon G \times G \to \mathbb{F}_{q^\alpha}^*$, which is defined by $\hat{e}(P, Q) = e(P, \phi(Q))$. $\hat{e}$ is called a modified Weil pairing in [BF01]. The Tate pairing has similar properties (see [Gal01] for more details). DDHP in G can be solved using these pairings. In many cases, it is believed that CDHP is hard, i.e., G is a GDH group.

We summarize well-known classes of elliptic curves which may contain a GDH group in Table 1. Since the pairing computation becomes inefficient as the exponent α becomes large, we only consider supersingular curves with $\alpha \leq 6$. Note that the hardness of CDHP depends on the size of q^α due to MOV's attack as well as the largest prime divisor ℓ of $\#E(\mathbb{F}_q)$.

Any supersingular curve has the form $y^2 + y = x^3 + ax + b$ over a binary field. Due to the Weil descent attack, we consider only the case that m is odd. In this case, all supersingular elliptic curves are isomorphic to one of three curves [Men93]. The curves over trinary fields were introduced in [BLS01] and used for generation of short signatures. The reason they used is that the exponent is largest among supersingular curves. Over finite fields of characteristic

Table 1. Various curves and their properties

Char.	Ext. Deg.	Curve	Order	α	ϕ
$p = 2$	Odd m	$y^2 + y = x^3$	$p^m + 1$	2	ϕ_1
$p = 2$	$m \equiv \pm 1(8)$ $m \equiv \pm 3(8)$	$y^2 + y = x^3 + x$	$p^m + 1 + \sqrt{2p^m}$ $p^m + 1 - \sqrt{2p^m}$	4	ϕ_2
$p = 2$	$m \equiv \pm 1(8)$ $m \equiv \pm 3(8)$	$y^2 + y = x^3 + x + 1$	$p^m + 1 - \sqrt{2p^m}$ $p^m + 1 + \sqrt{2p^m}$	4	ϕ_2
$p = 3$	$m \equiv \pm 1(12)$ $m \equiv \pm 5(12)$	$y^2 = x^3 + 2x + 1$	$p^m + 1 + \sqrt{3p^m}$ $p^m + 1 - \sqrt{3p^m}$	6	ϕ_3
$p = 3$	$m \equiv \pm 1(12)$ $m \equiv \pm 5(12)$	$y^2 = x^3 + 2x - 1$	$p^m + 1 - \sqrt{3p^m}$ $p^m + 1 + \sqrt{3p^m}$	6	ϕ_4
$p > 3\ (p \equiv 2(3))$	$m = 1$	$y^2 = x^3 + 1$	$p + 1$	2	ϕ_5
$p > 3\ (p \equiv 2(3))$	$m = 1$	$y^2 = x^3 + x$	$p + 1$	2	ϕ_6

$$\begin{aligned}
&\phi_1(x, y) = (\zeta x, y), && \zeta^2 + \zeta + 1 = 0 \\
&\phi_2(x, y) = (\zeta^2 x + \xi + 1, y + \zeta^2 \xi x + \eta), && \zeta^2 + \zeta + 1 = 0,\ \xi^4 + \xi + 1 = 0,\ \eta^2 + \eta = \xi^3 \\
&\phi_3(x, y) = (-x + r, iy), && r^3 + 2r + 2 = 0,\ i^2 + 1 = 0 \\
&\phi_4(x, y) = (-x + r, iy), && r^3 + 2r - 2 = 0,\ i^2 + 1 = 0 \\
&\phi_5(x, y) = (ix, y), && i^2 + 1 = 0 \\
&\phi_6(x, y) = (-x, iy), && i^2 + 1 = 0.
\end{aligned}$$

> 3, we do not have a special form for supersingular elliptic curves. But certain curves are supersingular over almost half of primes (called a CM-curve). We have two well-known families that were suggested in [BF01]. A detailed discussion on the curves in Table 1 except the first and the last ones can be found in [Gal01].

4.2 Hash Functions

We used cryptographic hash functions H_1 and H_2 in our scheme and viewed them as random oracles in the security proof. Though it is a debating issue if currently-used cryptographic hash functions can be considered as random oracles, standard cryptographic hash functions onto fixed-length binary strings or a finite field are accepted as random oracles in general. For the case of H_2 whose range is a GDH group G, however, we need to be careful since the elements of the group might not be expressed uniformly as binary strings.

A known approach is to construct a hash function onto G from standard hash functions onto finite fields. In [BLS01], they constructed one called MapToGroup and showed that their short signature scheme with this hash function is secure provided so is the scheme with a cryptographic hash function onto G, for GDH groups given as subgroups of elliptic curves defined over a finite field with odd

Table 2. The number of operations for BF-IBE and our signature scheme

Algorithm	Bilinear map	Point mul.	Exp. in $\mathbb{F}_{p^2}$	Hash functions
Encrypt	1	1	1	2
Decrypt	1	0	0	1
Sign	0	2	0	1
Verify	2	1	0	2

characteristic. In [BF01, BKS02], similar results providing more efficiency were proved for some other hash functions onto G in more restricted cases.

The main technique of the proofs of these results is as follows: Given an adversary to a scheme with a (possibly non-cryptographic) hash function, to say H', from standard hash functions onto finite fields, an adversary to the scheme with a cryptographic hash function H can be constructed by simulating H' with the help of H. This argument is not specific to a particular scheme, and indeed the same conclusion can be drawn for our scheme: Our scheme with the hash functions described in [BLS01, BF01, BKS02] is secure provided so is the scheme with a cryptographic hash function.

4.3 Performance

We compare the performance of our scheme with BF-IBE in Table 2.

We can see that the verification is most expensive and the signing is least expensive assuming the pairing computation costs several times expensive than a point multiplication of $E(\mathbb{F}_q)$ or an exponentiation of an element of $\mathbb{F}_{q^\alpha}$. (Very recently, an efficient implementation of Tate pairing over an elliptic curve with odd characteristic was announced in [BKS02].)

Note that the security of the scheme depends on the size of q^α as long as ℓ has a small cofactor. Since the pairing computation is comparable to an exponentiation in $\mathbb{F}_{q^\alpha}$, the efficiency of all algorithms but Sign does not change as the curve changes. Since binary fields have more efficient implementations, we may expect that the curves over a binary field offer most efficient performance. Sign algorithm, that does not perform any pairing computation, is most efficient as the exponent is large. In this case, signature size also becomes small since the signature consists of two elliptic curve points.

5 Conclusion

In this paper, we proposed an ID-based signature scheme from gap Diffie-Hellman groups. Our scheme can share parameters with BF-IBE and is as efficient as BF-IBE. Our scheme is secure against existential forgery on adaptively chosen message and ID attacks, under the hardness assumption of CDHP, which is believed to be weaker than the BDH assumption of BF-IBE. Combining our scheme with BF-IBE gives a practical complete solution of an ID-based public key system.

The *ID-based PKI* obtained by combining BF-IBE and our scheme may be considered as an alternative for certificate-based PKI. This ID-based PKI can be used when we have an existing hierarchy for each users to distribute the secret key securely and confidence on the key generation center, and offers advantages such as simple key management procedure [Sha84] and built-in key recovery [BF01]. Applications may include email systems, cellular phone services, and groupwares in private company where the key escrow is required.

Acknowledgements

The authors thank Dooho Choi and Seungjoo Kim for helpful discussions and the anonymous referees their valuable comments.

References

[BF01] D. Boneh and M. Franklin, *Identity Based Encryption from the Weil Pairing*, Proc. of Crypto '01, Lecture Notes in Computer Science, Vol. 2139, pp. 213-229, Springer-Verlag, 2001. (A full version is available from `http://crypto.stanford.edu/ dabo/pubs.html`) 18, 19, 21, 22, 26, 27, 28, 29

[BK02] P. Barreto and H. Kim, *Fast Hashing onto Elliptic Curves over Fields of Characteristic 3*, Available from `http://eprint.iacr.org`, 2002.

[BKS02] P. Barreto, H. Kim, and M. Scott, *Efficient Algorithms for Pairing-based Cryptosystems*, Available from `http://eprint.iacr.org`, 2002. 28

[BLS01] D. Boneh, B. Lynn, and H. Shacham, *Short Signatures from the Weil Pairing*, Proc. of Asiacrypt '01, Lecture Notes in Computer Sciences, Vol. 2248, pp. 514-532, Springer-Verlag, 2001. 20, 26, 27, 28

[CC01] J. Cheon and J. Cha, *Identity-based Signatures from the Weil Pairing*, Available from `http://vega.icu.ac.kr/ jhcheon/publications.html`, 2001. 19

[DQ86] Y. Desmedt and J. Quisquater, *Public-key Systems based on the Difficulty of Tampering*, Proc. of Crypto '86, Lecture Notes in Computer Sciences, Vol. 263, pp. 111-117, Springer-Verlag, 1987. 18

[FFS88] U. Feige, A. Fiat, and A. Shamir, *Zero-knowledge proofs of identity*, J. Cryptology, Vol. 1, pp. 77-94, 1988.

[FS86] A.Fiat and A. Shamir, *How to prove youself: Practical solutions to identification and signature problems*, Proc. of Crypto '86, Lecture Notes in Computer Sciences, Vol. 263, pp. 186-194, Springer-Verlag, 1987.

[Hes02] F. Hess, *Exponent group signature schemes and efficient identity based signature schemes based on pairings*, Available from `http://eprint.iacr.org`, 2002. 19

[Gal01] S. Galbraith, *Supersingular curves in cryptography*, Proc. of Asiacrypt '01, Lecture Nores in Computer Sciences, Vol. 2248, pp. 495-513, Springer-Verlag, 2001. 26, 27

[Men93] A. Menezes, *Elliptic curve public key cryptosystems*, Kluwer Academic Publishers, 1993. 26

[MY91] U. Maurer and Y. Yacobi, *Non-interective public-key cryptography*, Proc. of Eurocrypto '91, Lecture Nores in Computer Sciences, Vol. 547, pp. 498-507, Springer-Verlag, 1992. 18

[OP01] T. Okamoto and D. Pointcheval, *The gap-problems: a new class of problems for the security of cryptographic Schemes*, Proc. of PKC '01, Lecture Nores in Computer Sciences, Vol. 1992, pp. 104-118, Springer-Verlag, 2001. 20

[Pat02] K. Paterson, *ID-based signatures from pairings on elliptic curves*, Available from `http://eprint.iacr.org`, 2002. 19

[PS96] D. Pointcheval and J. Stern, *Security proofs for signature schemes*, Proc. of Eurocrypt '96, Lecture Notes in Computer Sciences, Vol. 1070, pp. 387–398, Springer-Verlag, 1996. 25

[PS00] D. Pointcheval and J. Stern, *Security arguments for digital signatures and blind signatures*, J. of Cryptology, Vol. 13, pp. 361-396, 2000. 23, 25, 26

[Sha84] A. Shamir, *Identity-base cryptosystems and signature schemes*, Proc. of Crypto '84, Lecture Notes in Computer Science, Vol. 196, pp. 47-53, Springer-Verlag, 1985. 18, 19, 29

[SOK01] R. Sakai, K. Ohgishi, and M. Kasahara, *Cryptosystems based on pairing*, Proc. of SCIS '00, Okinawa, Japan, Jan. pp. 26-28, 2001. 18

[Tan87] H. Tanaka, *A realization scheme for the identity-based cryptosystem*, Proc. of Crypto '87, Lecture Nores in Computer Sciences, Vol. 293, pp. 341-349, Springer-Verlag, 1987. 18

[TI89] S. Tsuji and T. Itoh, *An ID-based cryptosystem based on the discrete logarithm problem*, IEEE Journal of Selected Areas in Communications, Vol. 7, No. 4, pp. 467-473, 1989. 18