A Practical Identity-Based Signature Scheme from Bilinear Map*

Zhu Wang[1] and Huiyan Chen[2]

[1] State Key Laboratory of Information Security Graduate School of Chinese Academy of Sciences, Beijing, 100049
[2] Beijing Electronic Science and Technology Institute, Beijing 100070
chenhy2003@gmail.com

Abstract. In this paper, we present a new identity-based signature scheme with message recovery based on bilinear map. Our scheme is proved secure against existential forgery on adaptive chosen message and ID attack under the random oracle model. This new scheme shortens the total length of the original message and the appended signature and adapts to the ubiquitous network scenario very well.

Keywords: Identity-based signature, bilinear map, ID reduction, message recovery.

1 Introduction

In ubiquitous computing, the bandwidth of ubiquitous network is usually constrained, so it is desirable to shorten the total length of the original message M and the appended signature x. In this research area, there are two kinds of ideas adopted: one which is to directly produce short signature for message M, the other which is to "fold"part of message into the signature in such a way that it is "recoverable "by the verifier (i.e, signature scheme has the partial message recovery property). The former is taken by schemes proposed by D. Boneh et al. [10], F. Zhang et al. [11], D. Boneh and X. Boyen [12] and so on. In this paper, we mainly focus on the latter. On the whole, the existing digital signatures with message recovery may be classified into two types: RSA-based schemes and discrete-logarithm-based schemes. PSS–R [3] and ISO/IEC 9796-1,9796-2 are signature schemes with message recovery in the RSA type. The Nyberg-Rueppel [4,5,6], Miyaji [7] and Okamoto et al. [9] schemes are in DL (discrete logrithm) type.

The concept of identity-based cryptography was proposed in 1984 by Shamir [14]. The idea behind identity-based cryptography is that the user's public key can be derived from arbitrary string (e-mail address, IP address combined to a user name, social security number,...) which identifies him in a non ambiguous way. This greatly reduces the problems with key management. This kind of system needs trusted authority called private key generator (PKG) whose task

* This work is supported by the National Natural Science Foundation of China (No. 60577039).

M. Denko et al. (Eds.): EUC Workshops 2007, LNCS 4809, pp. 704–715, 2007.

is to compute user's private key from user's identity information (users do not generate their key pairs themselves). Several practical identity-based signature schemes [1,2,13] have been devised since 1984, but no identity-based signature scheme with message recovery goes out into the world until the scheme proposed by F. Zhang et al. [15] in 2005. F. Zhang et al. didn't quantify the security of their signature schemes in [15]. In addition, there are some problems occur in F. Zhang et al.'s schemes (see section 3).

In this paper, we present a new identity-based signature scheme with message recovery based on bilinear map, referred to as IDSMR. Its security is based on Computational Diffie-Hellman Assumption, CDH for short. IDSMR can deal with any message with arbitrary length.

Signature schemes from three move identification schemes such as Fiat-Shamir [1] are a typical class of practical signature schemes. To prove the security of such a class of signature schemes, K. Ohta and T. Okamoto presented a new key technique "*ID reduction*", in which the identification scheme corresponding to the signature scheme was used. In [8], K. Ohta and T. Okamoto thought that *ID reduction* technique had advantage over the previous technique, "*forking lemma*", by Pointcheval and Stern [16], and partly owed the advantage of *ID reduction* technique over *forking lemma* to the case that analyzing the identification scheme corresponding to the signature scheme was easier than analyzing the signature scheme. To prove that IDSMR is existentially unforgeable against adaptive chosen message and ID attack under the random oracle model, we make use of the *ID reduction* Technique and the results in [8,9].

The paper will proceed as follows. In section 2, we review some preliminaries used throughout this paper. In section 3, we review and analyse F. Zhang et al.'s schemes. In section 4, we present our signature scheme with message recovery. In section 5, we give security analysis of IDSMR. In section 6, we compare our scheme with other schemes. Section 7 concludes this paper.

2 Preliminaries

2.1 Notations

Throughout this paper, we will use the following notations. $|q|$ denotes the length of q in bit. If $|q| = 0$, q is denoted as $\emptyset$. Z^+ denotes the set of natural numbers and $\{0,1\}^*$ denotes the the space of finite binary strings. Let $[m]^{l_1}$ denote the most significant l_1 bits of m and $[m]_{l_2}$ denote the least significant l_2 bits of m. We denote by $a||b$ the string which is the concatenation of strings a and b. We also denote $[x]= y$ if $y \leq x < y+1$ and $y \in Z^+$. $a \bigoplus b$ denotes the bitwise XOR of bit strings a and b. If G is a group and $P \in G$, $(P)_2$ denotes the binary string representation of P.

2.2 Bilinear Map

Let G_1 be a cyclic additive group, whose order is a prime p, and G_2 be a cyclic multiplicative group with the same order p. Let $\widehat{e} : G_1 \times G_1 \rightarrow G_2$ be a bilinear map with the following properties:

(1) Bilinearity: $\widehat{e}(aP, bQ) = \widehat{e}(P, Q)^{ab}$ for all $P, Q \in G_1$, $a, b \in Z_p$
(2) Non-degeneracy: There exists $P, Q \in G_1$ such that $\widehat{e}(P, Q) \neq 1$, in other words, the map does not send all pairs in $G_1 \times G_1$ to the identity in G_2;
(3) Computability: There is an efficient algorithm to compute $\widehat{e}(P, Q)$ for all $P, Q \in G_1$.

The Weil and Tate pairings associated with supersingular elliptic curves can be modified to create such bilinear maps.

Definition 1. *CDH: Let G_1 be a cyclic additive group generated by P, whose order is a prime p. For $a, b \in Z_p$, given P, aP, bP, compute abP. An algorithm $\mathcal{A}$ has advantage ϵ in solving CDH in G_1 if*

$$Pr[\mathcal{A}(P,\ aP,\ bP)=abP] \geq \epsilon$$

where the probability is over the random choice of generator $P \in G_1$, the random choice of $a,b \in Z_p^*$ and the random bits consumed by $\mathcal{A}$.

Definition 2. *We say that the (t, ϵ)-CDH assumption holds in G_1 if no t-time algorithm has advantage at least ϵ in solving CDH in G_1.*

3 Analysis of F. Zhang et al.'s Scheme

Zhang et al. proposed two schemes in [15]: an ID-based message recovery signature scheme for messages of fixed length, and an ID-based partial message recovery signature scheme for messages of arbitrary length. Here we review their scheme for messages of fixed length and analyze its problems.

- **Setup:** The private key generator(PKG) chooses a random number $s \in Z_p^*$ and sets $P_{pub} = sP$. PKG also publishes system parameters $\{G_1, G_2, \widehat{e}, p, \lambda, P, P_{pub}, H_1, H_2, F_1, F_2, k_1, k_2\}$, and keeps s as the master-key, which is known only by itself. Here $|p| = k_1 + k_2$, $H_1 : \{0,1\}^* \rightarrow Z_p^*$, $H_2 : \{0,1\}^* \rightarrow G_1^*$, $F_1 : \{0,1\}^{k_2} \rightarrow \{0,1\}^{k_1}$, $F_2 : \{0,1\}^{k_1} \rightarrow \{0,1\}^{k_2}$ are four cryptographic hash functions
- **Extract:** A user submits his/her identity information ID to PKG. PKG computes the user's public key as $Q_{ID} = H_2(ID)$, and returns $S_{ID} = sQ_{ID}$ to the user as his/her private key.
- **Sign:** Let the message be $m \in \{0,1\}^{k_2}$
  1. Randomly choose $k \in Z_p^*$, and compute $v = \widehat{e}(P, P)^k$.
  2. Compute $f = F_1(m) || (F_2(F_1(m)) \bigoplus m)$.
  3. Compute $r = H_1(v) + f \mod p$
  4. Compute $U = kP - rS_{ID_A}$.

  The signature is (r, U).
- **Verify:** Given ID_A, a message m, and a signature (r, U), compute

$$r - H_1(\widehat{e}(U, P)\widehat{e}(Q_{ID_A}, P_{pub})^r) = f$$

and

$$m = [f]_{k_2} \bigoplus F_2([f]^{k_1})$$

Check whether $[f]^{k_1} = F_1(m)$ holds. If it is correct, then accept this signature and output true. Otherwise, output $\perp$.

In the above scheme, if $f \in Z_p$ and $|f| < |p|$, then, in the verification phase, we need padding $(|p| - |f|)0$s in the left of the binary string representation of f. Otherwise, the signature will be rejected. If $f > p$ and $|f| = |p|$, we say $f = p + f'$ then, in the verification phase, we get

$$r - H_1(\widehat{e}(U,P)\widehat{e}(Q_{ID_A}, P_{pub})^r) = f' \text{ and } m = [f']_{k_2} \bigoplus F_2([f']^{k_1})$$

With a large probability $[f']^{k_1} \neq F_1(m)$, so the signature will be rejected, although it is generated correctly. Zhang et al.'s second scheme for partial message recovery in [15] also suffers the similar problems.

In addition, their two schemes can't seem to deal with the message whose length in bits is less than some fixed length.

4 IDSMR Scheme

This section introduces our signature scheme with message recovery. It works as follows.

- **Setup:** Given a security parameter $l \in Z^+$, the private key generator(PKG) chooses two groups G_1 and G_2 of prime order p (*here,* l=$|p|$), a generator P of G_1, a bilinear map $\widehat{e}$:$G_1 \times G_1 \rightarrow G_2$. Then PKG picks a master-key $s \in Z_p^*$ and computes $P_{pub} = sP$ and w=$\widehat{e}(P,P)$. PKG also chooses cryptographic hash functions $H_1 : \{0,1\}^* \rightarrow G_1$, $H_2 : \{0,1\}^* \rightarrow \{0,1\}^l$, $F_1 : \{0,1\}^{[l/2]} \rightarrow \{0,1\}^{[(l+1)/2]}$, $F_2 : \{0,1\}^{[(l+1)/2]} \rightarrow \{0,1\}^{[l/2]}$. The system's public parameters are

$$Param = \{p, G_1, G_2, \widehat{e}, P, P_{pub}, w, H_1, H_2, F_1, F_2\}$$

- **Extract:** for an identity ID, the private key is d_{ID}=sQ_{ID}=$sH_1(ID)$.
- **Sign:** To sign a message $m = m_1||m_2$(If $|m| = [l/2]$, $m_2 = m$, $m_1 = \emptyset$; if $|m| > [l/2]$, $m_1 = [m]^{|m|-[l/2]}$, $m_2 = [m]_{[l/2]}$;), Alice follows the steps below
  1. Randomly choose $x \in Z_p^*$, and compute τ=w^x.
  2. Compute $f = F_1(m_2)||(F_2(F_1(m_2)) \bigoplus m_2)$.
  3. Compute $r = [(\tau)_2]_l \bigoplus f$
  4. Compute $r_0 = H_2(r||m_1)$
  5. Compute $S = xP - r_0 d_{ID_A}$.
  6. Alice sends $\sigma = (m_1, r, S)$ to verifier Bob.
- **Verify:** When receiving $\sigma = (m_1, r, S)$, Bob follows the steps below.
  1. Compute $r_0 = H_2(r||m_1)$
  2. Compute τ=$\widehat{e}(S,P)\widehat{e}(Q_{ID_A}, P_{pub})^{r_0}$
  3. Compute $f = r \bigoplus [(\tau)_2]_l$
  4. Compute $m_2 = [f]_{[l/2]} \bigoplus F_2([f]^{[(l+1)/2]})$
  5. Accept signature if and only if $[f]^{[(l+1)/2]} = F_1(m_2)$.
- **Remark 1:** If $|(\tau)_2|$<l, we need padding 0 in the left of $(\tau)_2$. If $|m| < [l/2]$, we need some redundancy to sign message m. We choose a hash function

$H : \{0,1\}^* \rightarrow \{0,1\}^{[l/2]}$ and set $m^{'} = m||H(m)$, then we sign message $m^{'}$ similar to message $m^{''}(|m^{''}| \geq [l/2])$. We don't discuss it any more here. Throughout this paper, we assume $|m| \geq [l/2]$ if message m need to be signed.

5 Security

In this section we prove the security of our signature scheme in the random oracle model, with CDH assumption. In order to prove the security of our signature scheme with *ID reduction* technique, we need to introduce an non-identity-based signature scheme, referred to as NIDS, and an identification scheme corresponding to NIDS, referred to as IFNIDS.

5.1 Attack Model for Identity-Based Signature Schemes

The most general known notion of security of a non-identity-based signature scheme is existential unforgery under adaptive chosen message attacks (EUF-ACMA); 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 identity-based signature schemes with message recovery.

Definition 3. *An identity-based signature scheme with message recovery, which consists of four algorithms* ***Setup***, ***Extract***, ***Sign***, *and* ***Verify*** *playing the same role as ours, has the existential unforgery for adaptive chosen message and ID attacks (EUF-ID-ACMA) property if no polynomial time algorithm $\mathcal{A}$ has a non-negligible succeed probability in the following game:*

1. *Challenger $\mathcal{C}$ 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. *$\mathcal{A}$ output (ID, σ, m_1), where ID is an identity, and σ is a signature of m ($m = m_1||m_2$. If $|m| > [l/2]$, $m_1 = [m]^{|m|-[(l+1)/2]}$, $m_2 = [m]^{[l/2]}$; if $|m| = [l/2]$, $m_2 = m$, $m_1 = \emptyset$), such that ID and (ID, m) are not equal to the input of any query to* ***Extract*** *and* ***Sign****, respectively. $\mathcal{A}$ wins the game if σ is a valid signature of m for ID.*

5.2 NIDS and IFNIDIS

Descriptions of NIDS Scheme. NIDS is described by three algorithms **Keygen**, **Sign** and **Verify**.

–**Keygen:** Given a security parameter $l \in Z^+$, signer $\mathcal{S}$ chooses the same system parameters as PKG of IDSMR except that it chooses its public key Q_{ID} and computes its private key d_{ID}=sQ_{ID}, doesn't choose hash function H_1. The system's public parameters are

$$Param=\{p,G_1,G_2,\widehat{e},P,P_{pub},\ Q_{ID},\ w,\ H_2,F_1,F_2\}$$

–**Sign:** To sign a message $m = m_1||m_2$(If $|m| = [l/2]$, $m_2 = m$, $m_1 = \emptyset$; if $|m| > [l/2]$, $m_1 = [m]^{|m|-[l/2]}$, $m_2 = [m]_{[l/2]}$), $\mathcal{S}$ follows the steps below

1. Randomly choose $x \in Z_p^*$, and compute τ=w^x.
2. Compute $f = F_1(m_2)||(F_2(F_1(m_2)) \bigoplus m_2)$.
3. Compute $r = [(\tau)_2]_l \bigoplus f$
4. Compute $r_0 = H_2(r||m_1)$
5. Compute $S = xP - r_0 d_{ID}$.
6. $\mathcal{S}$ sends $\sigma = (m_1, r, S)$ to verifier $\mathcal{V}$.

–**Verify:** When receiving $\sigma = (m_1, r, S)$, $\mathcal{V}$ follows the steps below:

1. Compute $r_0 = H_2(r||m_1)$
2. Compute τ=$\widehat{e}(S,P)\widehat{e}(Q_{ID},P_{pub})^{r_0}$
3. Compute $f = r \bigoplus [(\tau)_2]_l$
4. Compute $m_2 = [f]_{[l/2]} \bigoplus F_2([f]^{[(l+1)/2]})$
5. Accept signature if and only if $[f]^{[(l+1)/2]} = F_1(m_2)$.

Descriptions of IFNIDS Scheme. In IFNIDS, prover $\mathcal{P}$ publishes its public system parameters while keeping the corresponding secret key, and proves its identity to verifier $\mathcal{V}$. Here hash functions F_1, F_2 are shared by $\mathcal{P}$ and $\mathcal{V}$. IFNIDS works as follows.

–**Keygen:** Given a security parameter $l \in Z^+$, prover $\mathcal{P}$ chooses its public key Q_{ID}, computes its private key d_{ID}=sQ_{ID}, chooses the same system parameters as signer $\mathcal{S}$ of NIDS except that it doesn't choose hash function H_2. The system's public parameters are

$$Param=\{p,G_1,G_2,\widehat{e},P,P_{pub},Q_{ID},\ w,F_1,F_2\}$$

–**Identification Protocol:** $\mathcal{P}$ proves its identity and $\mathcal{V}$ checks the validity of $\mathcal{P}$' proof as follows:

(1) $\mathcal{P}$ chooses message m ($m = m_1||m_2$. If $|m| = [l/2]$, $m_2 = m$, $m_1 = \emptyset$; if $|m| > [l/2]$, $m_1 = [m]^{|m|-[l/2]}$, $m_2 = [m]_{[l/2]}$) and generates r as follows:

$$f = F_1(m_2)||(F_2(F_1(m_2)) \bigoplus m_2),\ \tau = w^x,\ r = f \bigoplus [(\tau)_2]_l$$

Here $x \in Z_p^*$ is uniformly selected. $\mathcal{P}$ sends $(r,\ m_1)$ to verifier $\mathcal{V}$.

(2) $\mathcal{V}$ generates random challenge $u \in Z_p^*$ and sends it to $\mathcal{P}$.

(3) $\mathcal{P}$ generates an answer S as follows and send it to $\mathcal{V}$.

$$S = xP - ud_{ID}$$

(4) $\mathcal{V}$ checks the validity of $\mathcal{P}$' proof through whether $[f]^{[(l+1)/2]} = F_1(m_2)$ holds or not, where

$\tau = \widehat{e}(S,P)\widehat{e}(Q_{ID}, P_{pub})^u, f = r \bigoplus [(\tau)_2]_l, m_2 = [f]_{[l/2]} \bigoplus F_2([f]^{[(l+1)/2]})$.

Security of NIDS and IFNIDS. In order to analyze the security of NIDS and IFNIDS, we firstly introduce the following notions similar to those [8,9]. Here we assume all hash functions are modeled as random oracles.

Definition 4. *An EUF-ACMA adversary $\mathcal{A}$ breaks NIDS with (t, q_{sig}, q_{F_1}, q_{F_2}, q_{H_2}, ϵ) if and only if $\mathcal{A}$ queries **Sign** at most q_{sig} times, queries hash functions F_1, F_2, H_2 at most q_{F_1}, q_{F_2}, q_{H_2} times respectively, and can forge a signature of NIDS within time t with success probability greater than ϵ.*

Definition 5. *NIDS is (t, q_{sig}, q_{F_1}, q_{F_2}, q_{H_2}, ϵ)-secure if and only if no adversary can not break it with (t, q_{sig}, q_{F_1}, q_{F_2}, q_{H_2}, ϵ).*

Definition 6. *An adversary $\mathcal{A}$ breaks IFNIDS with (t, q_{F_1}, q_{F_2}, ϵ) if and only if $\mathcal{A}$ as a prover queries hash functions F_1, F_2 at most q_{F_1}, q_{F_2} times respectively, and can cheat honest verifier $\mathcal{V}$ within time t with success probability greater than ϵ.*

Definition 7. *IFNIDS is (t, q_{F_1}, q_{F_2}, ϵ)-secure if and only if no adversary can not break it with (t, q_{F_1}, q_{F_2}, ϵ).*

Using the *ID Reduction* Technique and the results in [9], we can straightforwardly obtain the following lemma.

Lemma 1. *ID Reduction Lemma*

(1) If $\mathcal{A}$ breaks NIDS with (t, q_{sig}, q_{F_1}, q_{F_2}, q_{H_2}, ϵ), there exists $\mathcal{A}_1$ which breaks NIDS with (t', 0, q_{F_1}, q_{F_2}, 1, ϵ'), where $\epsilon' = (1/q_{H_2} - q_{sig}/2^l)(\epsilon - 1/2^l)$, and $t' = t +$ (the simulation time of q_{sig} signatures).

(2) If $\mathcal{A}_1$ breaks NIDS with (t', 0, q_{F_1}, q_{F_2}, 1, ϵ'), there exists $\mathcal{A}_2$ which breaks IFNIDS with (t', q_{F_1}, q_{F_2}, ϵ')

(3) If $\mathcal{A}_2$ breaks IFNIDS with (t', q_{F_1}, q_{F_2}, ϵ'), there exists $\mathcal{A}_3$ which breaks IFNIDS with (t', 1, 1, ϵ''), Where $\epsilon'' = \frac{\epsilon' - 1/2^{[l/2]}}{q_{F_1}}$

Theorem 1. *Let $\epsilon \geq \frac{5}{p}$. Suppose CDH in G_1 is (t^*, ϵ^*)-secure, then IFNIDS is (t, 1, 1, ϵ)-secure, where*

$$t^* = \frac{6t'}{\epsilon - 2/p} + O(t_{pm}),\ \epsilon^* = \frac{1}{2}(1 - e^{-1})^2 > \frac{9}{50},\ t' = t + O(2t_p + t_e)$$

Here t_{pm} denotes the computation time of point multiplication over additive group G_1, t_p denotes the computation time of bilinear map, t_e denotes the computation time of exponentiation over G_2 and e is the base of the natural logarithm.

Due to lack of space, the proof of the above theorem is omitted in this version of the paper. The basic idea of proof is to use boolean matrix and heavy row introduced [9] and is similar to that of proof on Lemma 4 in [9].

5.3 Security of IDSMR

In order to analyze security of IDSMR, we introduce the following quantifiable notions.

Definition 8. *An EUF-ID-ACMA adversary $\mathcal{A}$ breaks IDSMR with (t, q_{sig}, q_{H_1}, q_{H_2}, q_{F_1}, q_{F_2}, ϵ) if and only if $\mathcal{A}$ queries **Extract** at most q_E times, queries **Sign** at most q_{sig} times, queries hash functions H_1, H_2, F_1, F_2 at most q_{H_1}, q_{H_2}, q_{F_1}, q_{F_2} times respectively, and can forge a signature of IDSMR within time t with success probability greater than ϵ.*

Definition 9. *IDSMR is (t, q_{sig}, q_{H_1}, q_{H_2}, q_{F_1}, q_{F_2}, ϵ)-secure if and only if no adversary can not break it with (t, q_{sig}, q_{H_1}, q_{H_2}, q_{F_1}, q_{F_2}, ϵ).*

The following theorem shows the relation between IDSMR and NIDS in security.

Theorem 2. *In the random oracle model, suppose that an EUF-ID-ACMA adversary $\mathcal{A}_0$ exists which makes at most q_E **Extract** queries, at most q_{sig} **Sign** queries, and at most q_{H_1} queries to hash function H_1, and which succeeds within time t_0 of making an existential forgery of IDSMR signature with probability greater than ϵ_0, then there is an EUF-ACMA adversary $\mathcal{A}_1$ which succeeds within time $t = O(t_0)$ of making an existential forgery of NIDS signature with probability $\epsilon > \epsilon_0(1 - 1/p)/q_{H_1}$. In addition, the numbers of queries to other hash functions asked by $\mathcal{A}_1$ are the same as those of $\mathcal{A}_0$.*

Proof. We show how to construct an EUF-ACMA adversary $\mathcal{A}_1$ that uses $\mathcal{A}_0$ to gain advantage $\epsilon_0(1 - 1/p)/q_{H_1}$ against NIDSMR. The game between the challenger and $\mathcal{A}_1$ starts with the challenger first generating random public system parameters $Param$=$\{p, G_1, G_2, \widehat{e}, P, P_{pub}, Q_{ID}\ w, H_2, F_1, F_2\}$ (Here $P_{pub} = sP$,$Q_{ID} \in G_1$), and a private key $d_{ID} = sQ_{ID}$. The challenger gives $Param$ to algorithm $\mathcal{A}_1$. The algorithm $\mathcal{A}_1$ interacts with $\mathcal{A}_0$ as follows and maintains list L_1 that is initially empty and is used to keep track of answers to queries asked by $\mathcal{A}_0$ to oracle H_1, and challenger maintains lists L_2, L_3 and L_4 that are initially empty and are used to keep track of answers to queries asked by $\mathcal{A}_0$ to oracle H_2, F_1 and F_2.

– **Setup:** The algorithm $\mathcal{A}_1$ gives the algorithm $\mathcal{A}_0$ the system parameters $\{p, G_1, G_2, \widehat{e}, P, P_{pub}, w, H_1, H_2, F_1, F_2\}$ of IDSMR scheme. Here $p, G_1, G_2, \widehat{e}, P, P_{pub}, w, H_2, F_1, F_2$ are taken from $Param$.

– H_1 **queries:** When $\mathcal{A}_0$ asks queries on the hash values of identities, $\mathcal{A}_1$ checks the list L_1. If an entry for the query is found, the same answer will be given to $\mathcal{A}$; otherwise, a value d_j from Z_p^* will be randomly chosen and d_jP will be used as the answer, (ID_j, d_jP) will then be stored in the list L_1. The only exception is that $\mathcal{A}_1$ has to randomly choose one of the H_1 queries from $\mathcal{A}_0$, say the i^{th} query, and answers $H_1(ID_i)$=Q_{ID} for this query.

Note that we assume that $\mathcal{A}_0$ must ask for $H_1(ID)$ before ID is used in any **Sign** and **Extract** queries.

– **H_2, F_1 and F_2 queries:** When $\mathcal{A}_0$ asks queries on these hash functions, $\mathcal{A}_1$ relays these queries to Challenger. Challenger checks the corresponding list. If an entry for the query is found, the same answer will be given to $\mathcal{A}_1$; otherwise, a randomly generated value will be used as an answer to $\mathcal{A}_1$, the query and the answer will then be stored in the list. $\mathcal{A}_1$ relays challenger's responses to $\mathcal{A}_0$.
– **Key extraction queries:** When $\mathcal{A}_0$ asks a private key extraction to ID_j, if $j = i$, then $\mathcal{A}_1$ fails and stops. If $j \neq i$, then the list L_1 must contain (ID_j, d_jP). $\mathcal{A}_1$ sends (ID_j, d_jP) to challenger and relays this query to challenger. Challenger computes private key $d_{ID_j} = sd_jP$ which corresponds to ID_j, and sends d_{ID_j} to $\mathcal{A}_1$. $\mathcal{A}_1$ relays d_{ID_j} to $\mathcal{A}_0$.
– **Sign queries:** Given an identity ID and a message $m(= m_1||m_2)$, $\mathcal{A}_1$ works as follows.
(1) $\mathcal{A}_1$ gets $Q_{ID}^{'} = H_1(ID)$ by simulation for H_1.
(2) $\mathcal{A}_1$ sends $Q_{ID}^{'}$ to challenger and relays this signature query to challenger.
(3) Challenger randomly selects $x \in Z_p^*$, computes $d_{ID}^{'}=sQ_{ID}^{'}$ and $\tau=w^x$, gets the hash values by simulation for H_2, F_1 and F_2, computes signature $\sigma = (m_1, r, S)$ to the signature query (ID, m)(here $m = m_1||m_2$), and sends σ to $\mathcal{A}_1$. $\mathcal{A}_1$ relays this signature σ to $\mathcal{A}_0$.
– $\mathcal{A}_0$ outputs (ID_{out}, m_1, r, S), where ID_{out} is an identity, m_1 is part of message m, and (m_1, r, S) is a signature to m, such that ID_{out} and (ID_{out}, m) are not equal to the input of any query to **Extract** and **Sign**, respectively.
– If $ID_{out} = ID_i$ and (ID_{out}, m_1, r, S) is valid, then outputs (ID_{out}, m_1, r, S). Otherwise output fail.

If algorithm $\mathcal{A}_1$ does not abort during simulation, algorithm $\mathcal{A}_0$'s view is identical to its view in the attack, furthermore

$$Pr[(ID_{out}, m_1, r, S) is\ valid | \mathcal{A}_1\ does\ not\ abort] > \epsilon_0$$

Let $\mathcal{E}_1$ be the event that algorithm $\mathcal{A}_1$ does not abort during simulation. Let $\mathcal{E}_2$ be the event that (ID_{out}, m_1, r, S) is valid. Since H_1 is a random oracle, the probability that the output (ID_{out}, m_1, r, S) of $\mathcal{A}_0$ is valid without any query of $H_1(ID_{out})$ is negligible. Explicitly,

$$Pr[ID_{out} = ID_j\ for\ some\ j, j \leq q_{H_1} | \mathcal{E}_1 \wedge \mathcal{E}_2] \geq 1 - 1/p$$

Since i is independently and randomly chosen, we have

$$Pr[ID_{out} = ID_i | (ID_{out} = ID_j\ for\ some\ j, j \leq q_{H_1}) \wedge \mathcal{E}_1 \wedge \mathcal{E}_2] \geq 1/(q_{H_1} - q_E)$$

$\mathcal{A}_1$'s failure during simulation is caused by $\mathcal{A}$ issuing a private query to ID_i, we have

$$Pr[\mathcal{E}_1] = (\tfrac{q_{H1}-1}{q_{H1}})(\tfrac{q_{H1}-2}{q_{H1}-1}) \ldots (\tfrac{q_{H1}-q_E}{q_{H1}-q_E+1}) = \tfrac{q_{H1}-q_E}{q_{H1}}$$

Therefore, we have

$$Pr[(ID_{out} = ID_i) \wedge ((ID_{out}, m_1, r, S)\ is\ valid) \wedge \mathcal{E}_1] > \epsilon_0(1 - 1/p)/q_{H_1}$$

Combing theorem 9 and 12 and lemma 8, we have

Theorem 3. (*Security of IDSMR*) *Let* $\epsilon \geq q_{H_1} \times \frac{p}{p-1} \times ((\frac{5q_{F_1}}{p} + \frac{1}{2^{[l/2]}})/(\frac{1}{q_{H_2}} - \frac{q_{sig}}{2^l}) + \frac{1}{2^l})$. *Suppose CDH is* (t^*, ϵ^*)*-secure, then IDSMR is* $(t, q_{sig}, q_{H_1}, q_{H_2}, q_{F_1}, q_{F_2}, \epsilon)$*-secure, where*

$$t^* = \frac{6t^{'}}{\epsilon^{'} - 2/p} + O(t_{pm}) \text{ and } \epsilon^* = \frac{1}{2}(1 - e^{-1})^2 > \frac{9}{50}$$

Here

$$t^{'} = O(t) + O(q_{sig}(t_e + 2t_{pm}) + t_e + 2t_p)$$
$$\epsilon^{'} = \frac{1}{q_{F_1}}((\frac{p-1}{pq_{H_1}}\epsilon - \frac{1}{2^l})(\frac{1}{q_{H_2}} - \frac{q_{sig}}{2^l}) - \frac{1}{2^{[l/2]}})$$

where t_{pm} *denotes the computation time of point multiplication over additive group* G_1, t_p *denotes the computation time of bilinear map,* t_e *denotes the computation time of exponentiation over* G_2 *and* e *is the base of the natural logarithm.*

6 Comparison of Schemes

In table 1 below, we compare our scheme with schemes [13,15,17,18] in terms of the total length of the original message and the appended signature, and the number of the dominant operations required by them. In table we use mls , exps, and pcs as abbreviations for point multiplications in G_1, exponentiations in G_2 and computations of bilinear map respectively.

Table 1. Comparison of Schemes

Schemes	Total Length*		Efficiency					
	$\|m\|=[l/2]$	$\|m\|>[l/2]$	Sign			Verify		
		$(m_1 = [m]^{\|m\|-[l/2]})$	mls	exps	pcs	mls	exps	pcs
F. Hess [13]	$\|m\| + \|p\| + \|G_1\|$	$\|m\| + \|p\| + \|G_1\|$	1	1	1		1	2
Cha-Cheon [17]	$\|m\| + 2\|G_1\|$	$\|m\| + 2\|G_1\|$	2			1		2
Libert et al.[18]	$\|m\| + \|p\| + \|G_1\|$	$\|m\| + \|p\| + \|G_1\|$	1	1		1	1	1
F. Zhang et al[15]	$\|p\| + \|G_1\|$	$\|m_1\| + \|p\| + \|G_1\|$	2	1	1		1	2
IDSMR	$\|p\| + \|G_1\|$	$\|m_1\| + \|p\| + \|G_1\|$	2	1			1	2

(∗) Total length is the length of the original message and the appended signature.

7 Conclusion

This paper presented a signature scheme with message recovery. It is proved to be secure in the strongest sense (i.e., existentially unforgeable under adaptive chosen message and ID attacks) in the random oracle model under the CDH assumption. Furthermore, our scheme can deal with any message with arbitrary length and shortens the length of the original message and the appended signature by "folding" part of message into the signature.

Acknowledgement

The authors would like to thank anonymous referees for their helpful comments.

References

1. Fiat, A., Shamir, A.: How to Prove Yourself: Practical Solutions to Identification and Signature Problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (1987)
2. Guillou, L., Quisquater, J-J.: A "Paradoxical" Identity-Based Signature Scheme Resulting From Zero-Knowledge. In: Goldwasser, S. (ed.) CRYPTO 1988. LNCS, vol. 403, pp. 216–231. Springer, Heidelberg (1990)
3. Bellare, M., Rogaway, P.: The Exact Security of Digital Signatures –How to Sign with RSA and Rabin. In: Proc. of Eurocrypt's 1996. LNCS, pp. 399–416. Springer, Heidelberg (1996)
4. Nyberg, K., Rueppel, R.A., New, A.: Signature Scheme Based on the DSA Giving Message Recovery. In: Proc. of the First ACM Conference on Computer and Communications Security (1993)
5. Nyberg, K., Rueppel, R.A.: Message Recovery for Signature Schemes Based on the Discrete LogarithmProb lem. In: Proc. of Eurocrypt's 1994. LNCS, pp. 182–193. Springer, Heidelberg (1995)
6. Nyberg, K., Rueppel, R.A.: Message Recovery for Signature Schemes Based on the Discrete Logarithm Problem. Designs, Codes and Cryptography 7, 61–81 (1996)
7. Miyaji, A.: A Message Recovery Signature Scheme Equivalent to DSA over Elliptic Curves. In: Proc. of Asiacrypt's 1996. LNCS, pp. 1–14. Springer, Heidelberg (1996)
8. Ohta, K., Okamoto, T.: On the Concrete Security Treatment of Signatures Derived from Identification. In: RobVis 2001. LNCS, pp. 354–369. Springer, Heidelberg (1998)
9. Abe, M., Okamoto, T.: A Signature Scheme with Message Recovery as Secure as Discrete Logarithm. IEICE Trans. Fundamentals E84-A(1), 197–204 (2001)
10. Boneh, D., Lynn, B., Shacham, H.: Short signatures from the weil pairing. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 514–532. Springer, Heidelberg (2001)
11. Zhang, F., Safavi-Naini, R., Susilo, W.: An Efficient Signature Scheme from Bilinear Pairings and Its Applications. In: Bao, F., Deng, R., Zhou, J. (eds.) PKC 2004. LNCS, vol. 2947, pp. 277–290. Springer, Heidelberg (2004)
12. Boneh, D., Boyen, X.: Short Signatures Without Random Oracles. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 56–73. Springer, Heidelberg (2004)
13. Hess, F.: Efficient identity based signature schemes based on pairings. In: Nyberg, K., Heys, H.M. (eds.) SAC 2002. LNCS, vol. 2595, Springer, Heidelberg (to appear)
14. Shamir, A.: Identity Based Cryptosystems and Signature Schemes. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, Springer, Heidelberg (1985)
15. Zhang, F., Susilo, W., Mu, Y.: Identity-based Partial Message Recovery Signatures (or How to Shorten ID-based Signatures). In: Patrick, A.S., Yung, M. (eds.) FC 2005. LNCS, vol. 3570, pp. 47–59. Springer, Heidelberg (2005)

16. Pointcheval, D., Stern, J.: Security Proofs for Signature Schemes. In: Maurer, U.M. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 387–398. Springer, Heidelberg (1996)
17. Cha, J.C., Cheon, J.H.: An identity-based signature from gap Diffie-Hellman groups. In: PKC 2003. LNCS, vol. 2567, pp. 18–30. Springer, Heidelberg (2003)
18. Barreto, P.S.L.M., Libert, B., McCullagh, N., Quisquater, J.-J.: Efficient and Provably-Secure Identity-Based Signatures and Signcryption from Bilinear Maps. In: Roy, B. (ed.) ASIACRYPT 2005. LNCS, vol. 3788, pp. 515–532. Springer, Heidelberg (2005)