Paper

A New Short Signature Scheme with Random Oracle from Bilinear Pairings

Sedat Akleylek[a,b], Barış Bülent Kırlar[a,c], Ömer Sever[a], and Zaliha Yüce[a]

[a] *Institute of Applied Mathematics, Middle East Technical University, Ankara, Turkey*
[b] *Department of Computer Engineering, Ondokuz Mayıs University, Samsun, Turkey*
[c] *Department of Mathematics, Süleyman Demirel University, Isparta, Turkey*

Abstract—In this paper, we propose a new and efficient short signature scheme from the bilinear pairings. Our scheme is constructed by bilinear inverse-square Diffie-Hellman problem (BISDHP) and does not require any special hash function. The exact security proofs are also explained in the random Oracle model. We give the implementation and comparison results of our proposed signature scheme with the signature scheme proposed by Boneh, Lynn, Shacham (BLS) and Zhang, Safavi, Susilo (ZSS). Furthermore, we use this signature scheme to construct a ring signature scheme.

Keywords—bilinear Diffie-Hellman problem, bilinear pairings, ring signature, short signature.

1. Introduction

Digital signatures are one of the most important cryptographic primitive for the daily life. Short signatures are needed in environments with space and bandwidth constraints. Up to pairing-based cryptography, the best known shortest signature was obtained by using the digital signature algorithm (DSA) [1] over a finite field $\mathbb{F}_q$. The length of the signature is approximately $2\log q$. On the other hand, when the pairing-based cryptographic protocol is used the length of the signature is about $\rho \log r$, where $\rho = \log q / \log r$ and r is the largest prime divisor of the number of the points in the elliptic curve. For example, if one uses RSA signature 1024 bit modulus, the output of elliptic curve digital signature algorithm (ECDSA) is 320 bit long for the same security level. However, short signature provides the same security level only for 160 bits for the best choice.

In 2001 Boneh, Lynn and Shacham [2] proposed the idea of short signature scheme by using bilinear pairings. This scheme is based on Weil pairing and needs a special hash function [2], [3], [4]. Over the last years, there are various applications of bilinear pairings in short signature schemes to construct new efficient schemes [5], [6], [7]. The main improvement in short signature schemes is the use of cryptographic hash function such as MD5, SHA-1 [7] instead of special hash function called `MapToPoint` hash operation. It is known that short signature scheme with cryptographic hash function is more efficient than others since `MapToPoint` hash operation is still probabilistic.

In this study, we describe a new short signature scheme in a similar setting in ZSS scheme [7]. Our system is based on bilinear inverse-square Diffie-Hellman problem a combination of bilinear inverse Diffie-Hellman problem (BIDHP) and bilinear square Diffie-Hellman problem (BSDHP). The main advantage of our scheme is that it can be used with any cryptographic hash function such as MD5, SHA-1. To give the exact security proofs, we define a new problem called inverse square problem with k traitors ($k-$ISP). Then, the exact security proofs of proposed scheme are also explained in the random Oracle model. We give the comparison of our scheme with the BLS scheme and ZSS scheme. According to the comparison results, our scheme is more efficient than BLS scheme.

Furthermore, based on new proposed signature scheme, we construct a ring signature scheme.

This paper is organized as follows: Some preliminaries about bilinear pairings and some related problems to pairings are given in Section 2. Proposed short signature scheme and its security analysis are explained in Section 3. A construction of ring signature scheme is given in Section 4. We conclude in Section 5.

2. Pairing-Based Cryptography

In this section, we give some facts about bilinear pairings and define some new problems. The proposed short signature scheme uses bilinearity like others.

2.1. Bilinear Pairings

Definition 1: Let G_1 and G_2 be additive cyclic groups of order n. Let G_3 be a multiplicative cyclic group of order n. A bilinear pairing is an efficiently computable map $e: G_1 \times G_2 \longrightarrow G_3$ which satisfies the following additional properties:

1. (bilinearity) For all $P,R \in G_1$ and all $Q,S \in G_2$, we have $e(P+R,Q) = e(P,Q)e(R,Q)$ and $e(P,Q+S) = e(P,Q)e(P,S)$.
2. (non-degeneracy) For all $P \in G_1$, with $P \neq Id_{G_1}$, there is some $Q \in G_2$ such that $e(P,Q) \neq Id_{G_3}$. For all $Q \in G_2$, with $Q \neq Id_{G_2}$, there is some $P \in G_1$ such that $e(P,Q) \neq Id_{G_3}$. When $G_1 = G_2$ and n is prime, $e(P,P)$ is a generator of G_3 for all $P \neq Id_{G_1}$

The following lemma which is related to the properties of bilinear pairings can be easily verified.

Lemma 1: Let $e : G_1 \times G_2 \longrightarrow G_3$ be a bilinear pairing. Let $P \in G_1$ and $Q \in G_2$. Then

1. $e(P, Id_{G_2}) = e(Id_{G_1}, Q) = Id_{G_3}$
2. $e(-P, Q) = e(P, -Q) = e(P, Q)^{-1}$
3. $e(kP, Q) = e(P, kQ) = e(P, Q)^k$ for all $k \in \mathbb{Z}$.
4. $e(kP, lP) = e(P, P)^{kl}$ for all $k, l \in \mathbb{Z}$.

2.2. Some Problems

We consider the following problems in the additive group $(G, +)$ of order n.

- **Discrete logarithm problem (DLP):** For $P, Q \in G$, find $k \in \mathbb{Z}_n^*$ such that $Q = kP$ whenever such k exists.
- **Decisional Diffie-Hellman Problem (DDHP):** For $a, b, c \in \mathbb{Z}_n^*$, given P, aP, bP, cP decide whether $c \equiv ab \pmod{n}$.
- **Computational Diffie-Hellman problem (CDHP):** For $a, b \in \mathbb{Z}_n^*$, given P, aP, bP compute abP.

There are two variations of CDHP:

- **Inverse computational Diffie-Hellman problem (ICDHP):** For $a \in \mathbb{Z}_n^*$, given P, aP, compute $a^{-1}P$.
- **Square computational Diffie-Hellman problem (SCDHP):** For $a \in \mathbb{Z}_n^*$, given P, aP, compute a^2P.

The following theorem shows the relation of these problems that the proof can be found in [8].

Theorem 1: CDHP, ICDHP and SCDHP are polynomial time equivalent.
The security of some applications of bilinear pairings in cryptography relies on the difficulty of bilinear Diffie-Hellman problem (BDHP) which was first stated in [4].

Definition 2: Let G be a finite additive cyclic group of order n with a generator P, let e be a bilinear pairing on G, and let a, b, c be integers. The BDHP is to compute the value of the bilinear pairing $e(abcP, P)$, whenever aP, bP and cP are given.

The well known pairing-based protocols are three-party key exchange in one round protocol proposed by Joux in [9], identity-based encryption scheme by Boneh-Franklin in [4] and short signature scheme by Boneh-Lynn-Shacham in [2] that the security of them depends on the BDHP. There are variants of BDHP:

- **Bilinear inverse Diffie-Hellman problem (BIDHP):** For $a, b \in \mathbb{Z}_n^*$, given P, aP, bP to compute $e(P, P)^{a^{-1}b}$.
- **Bilinear square Diffie-Hellman problem (BSDHP):** For $a, b \in \mathbb{Z}_n^*$, given P, aP, bP to compute $e(P, P)^{a^2b}$.

It is not hard to obtain bilinear inverse-square Diffie-Hellman Problem as a combination of BIDHP and BSDHP:

- **Bilinear inverse-square Diffie-Hellman problem (BISDHP):** For $a, b \in \mathbb{Z}_n^*$, given P, aP, bP to compute $e(P, P)^{a^{-2}b}$.

Theorem 2: BDHP, BIDHP, BSDHP and BISDHP are polynomial time equivalent.

Proof:
BDHP ⇒ BIDHP is trivial.
BIDHP ⇒ BSDHP:
Given P, aP, bP, set the input of BIDHP as

$$Q = aP,\ Q_1 = P = a^{-1}Q,\ Q_2 = bP = ba^{-1}Q,$$

then BIDHP outputs

$$e(Q_1, Q_2) = e(Q, Q)^{(a^{-1})^{-1}ba^{-1}} = e(aP, aP)^b = e(P, P)^{a^2b}$$

BSDHP ⇒ BISDHP:
Given P, a^2P, bP, set the input of BSDHP as

$$Q = a^2P,\ Q_1 = a^{-2}Q = P,\ Q_2 = a^{-2}bQ = bP,$$

then BSDHP outputs

$$e(Q_1, Q_2) = e(Q, Q)^{(a^{-2})^2ba^{-2}} = e(P, P)^{a^{-2}b}$$

BISDHP ⇒ BDHP:
Given P, aP, bP, cP, set the input of BSDHP as the triples

$$(P, aP, cP),\ (P, bP, cP),\ (P, aP + bP, cP),$$

then we have $e(P, P)^{a^{-2}c}$, $e(P, P)^{b^{-2}c}$ and $e(P, P)^{(a+b)^{-2}c}$, respectively. Therefore, we obtain

$$e(P, P)^{abc} = \left(\frac{e(P,P)^{a^{-2}c} \cdot e(P,P)^{b^{-2}c}}{e(P,P)^{(a+b)^{-2}c}}\right)^{1/2}.$$

■

3. New Short Signature Scheme From Bilinear Pairings

In this section, we propose our signature scheme, and then explain its security. Moreover, we compare our scheme with BLS and ZSS schemes from the implementation point of view.

3.1. Signature Scheme

A signature scheme consists of four steps: a parameter generation algorithm `ParamGen`, a key generation algorithm `KeyGen`, a signature generation algorithm `Sign` and a signature verification algorithm `Verify`.
We describe the new signature scheme as follows:
Let $(G_1, +)$ and $(G_2, \cdot)$ be cyclic groups of prime order n, $P \in G_1$, $G_1 =< P >$ and $e : G_1 \times G_1 \rightarrow G_2$ be a bilinear map. Let $H : Z_2^{\infty} \rightarrow Z_2^{\lambda}$, where $160 \leq \lambda \leq \log(n)$ be a cryptographic hash function such as SHA1 or MD5. Suppose that $\mathcal{A}$ wants to send a signed message to $\mathcal{B}$.

- `ParamGen`: $\{G_1, G_2, e, n, P, H\}$
- `KeyGen`: $\mathcal{A}$ randomly selects $x \in \mathbb{Z}_n$ and computes $P_{pub1} = x^2P$ and $P_{pub2} = 2xP$. In this structure, P, P_{pub1} and P_{pub2} are the public keys, x is the secret key.
- `Sign`: Given a secret key x and a message m, $\mathcal{A}$ computes the signature, $s = (H(m)+x)^{-2}P$.
- `Verify`: Given the public keys P, P_{pub1} and P_{pub2}, a message m and a signature s, $\mathcal{B}$ verifies the signature if

$$e(H(m)^2P + P_{pub1} + P_{pub2}H(m), s) = e(P,P) \text{ holds.}$$

The verification is done by using bilinearity in the following equations:

$$e((H(m)+x)^2P, (H(m)+x)^{-2}P) = \\ e(P,P)^{(H(m)+x)^2(H(m)+x)^{-2}} = e(P,P).$$

3.2. Signature Security

The well-known attacks against signature schemes are without message attack and chosen-message attack. The strongest version of these attacks is an adaptive chosen-message attack. In this scenario, the attacker can ask the signer to sign any message that he/she chooses. He also knows the public key of the signer. Then, he can customize his queries according to the previous message and chosen signature pairs.

The strongest notion of security for signature schemes that is existentially unforgeable under adaptive chosen-message attack was defined by Goldwasser, Micali and Rivest [10]. Here, we use the definition of exact secure signature schemes by Bellare and Rogaway [11] stated as follows:

Definition 3: A signature scheme S, defined by $S = <\texttt{ParamGen}, \texttt{KeyGen}, \texttt{Sign}, \texttt{Verify}>$, is $(t, q_H, q_S, \varepsilon)$-existentially unforgeable under adaptive chosen-message attack if for every probabilistic polynomial time forger algorithm $\mathcal{F}$ running in t processing time, at most q_H queries to the hash oracle and q_S signatures queries, there does not exist a non-negligible probability ε.

A signature scheme S is $(t, q_H, q_S, \varepsilon)$-secure if there is no forger who $(t, q_H, q_S, \varepsilon)$ breaks the scheme.

We introduce a new problem that was called **k-ISP** (inverse square problem with k traitors) to give the security proof of the new signature scheme. This problem is similar to **k-CAA** (collusion attack algorithm with k traitors) that was proposed by Mitsunari, Sakai and Kasahara in [12].

Definition 4: (**k-ISP**) For an integer k, and $x \in \mathbb{Z}_n$, $P \in G_1$, given

$$\{P, xP, H_1, H_2, \cdots, H_k, (H_1+x)^{-2}P, (H_2+x)^{-2}P, \cdots, (H_k+x)^{-2}P\},$$

compute $(H+x)^{-2}P$ for some $H \notin \{H_1, H_2, \cdots, H_k\}$.

k-ISP is (t, ε)-hard if for any t-time adversaries $\mathcal{A}$, we have

$$Pr\left[\begin{array}{l}\mathcal{A}\Big(P, xP, H_1, H_2, \cdots, H_k, (H_1+x)^{-2}P, (H_2+x)^{-2}P, \cdots, \\ \quad (H_k+x)^{-2}P) | x \in \mathbb{Z}_n, P \in G_1, H_1, H_2, \cdots, H_k \in \mathbb{Z}_n\Big) \\ \qquad = (H+x)^{-2}P, H \notin \{H_1, H_2, \cdots, H_k\}\end{array}\right] < \varepsilon$$

where ε is negligible.

The following theorem shows that proposed signature scheme is secure against the adaptive chosen-message attack.

Theorem 3: If there exists a $(t, q_H, q_S, \varepsilon)$-forger $\mathcal{F}$ using an adaptive chosen message attack for the signature scheme proposed in Section 3.1, then there exists a (t', ε')-algorithm $\mathcal{A}$ solving $q_S - ISP$, where $t' = t$ and $\varepsilon' \geq (\frac{q_S}{q_H})^{q_S} \cdot \varepsilon$.

Proof: Assume that the output of the hash function is uniformly distributed and the hash oracle will give a correct response for any hash query.

Suppose that a forger $\mathcal{F}$ $(t, q_H, q_S, \varepsilon)$-break the signature scheme using an adaptive chosen message attack. One needs an algorithm $\mathcal{A}$ to solve $q_s - ISP$. In this structure, the challenge is to compute $(H+x)^{-2}P$ for some $H \notin \{H_1, H_2, \cdots, H_k\}$ for given $P \in G_1$, $P_{pub1} = x^2P$, $P_{pub2} = 2xP$, $H_1, H_2, \cdots, H_{q_s} \in Z_n$ and $(H_1+x)^{-2}P$, $(H_2+x)^{-2}P, \cdots, (H_{q_s}+x)^{-2}P$

$\mathcal{A}$ is the signer and answers hash and signing queries by itself. Algorithm is as follows:

Step 1: $\{H_1, H_2, \cdots, H_{q_H}\}$ are the responses of the hash oracle queries for the corresponding messages $\{m_1, m_2, \cdots, m_{q_H}\}$.

Step 2: $\mathcal{F}$ makes a signature oracle query for each H_i for $1 \leq i \leq q_H$. If the hash oracle answers truely, $\mathcal{A}$ returns $(H_i+x)^{-2}P$ to $\mathcal{F}$ as the response. Otherwise, the process stops.

Step 3: $\mathcal{F}$ outputs a message-signature pair (m, S). The hash value of m is some H and $H \notin \{H_1, H_2, \cdots, H_{q_H}\}$. It satisfies:

$$e(x^2P + 2xP + H^2P, S) = e(P,P)$$

So, $S = (H+x)^{-2}P$. $\mathcal{A}$ outputs (H, S) as a solution of challenge.

Since the operations are the same for $\mathcal{A}$ and $\mathcal{F}$, the running time of $\mathcal{A}$ and $\mathcal{F}$ is equal, $t = t'$. The success probability of $\mathcal{A}$ is $\frac{q_S}{q_H}$ is Step 2. $\mathcal{A}$ will not fail with probability $p \geq (\frac{q_S}{q_H})^{q_S}$. Then, the success probability of the algorithm, $\mathcal{A}$ for all steps is $\varepsilon' \geq (\frac{q_S}{q_H})^{q_S} \cdot \varepsilon$. This completes the proof. ■

Note that, one can obtain a good bound if q_S and q_H are very closed.

We now recall k-weak computational Diffie-Hellman problem (**k-wCDHP**) proposed by Mitsunari *et. al* [12].

Definition 5: (**k-wCDHP**) For an integer k, and $x, H \in \mathbb{Z}_n$, $P \in G_1$, given $k+1$ values

$$\{P, (H+x)P, (H+x)^2P, \cdots, (H+x)^kP\},$$

compute $(H+x)^{-1}P$.

We define a new problem that is called $k+1$ inverse exponent problem (**k+1-IEP**) to give a specific evaluation of the security of our proposed signature scheme.

Definition 6: (**k+1-IEP**) For an integer k, and $a \in \mathbb{Z}_n$, $P \in G_1$, given $k+1$ values

$$\{P, aP, a^{-2}P, \cdots, a^{-k}P\},$$

compute $a^{-(k+1)}P$.

Theorem 4: **k-wCDHP** and **k+1-IEP** are polynomial time equivalent.

Proof:
k-wCDHP $\Rightarrow$ **k+1-IEP**:
Given $k+1$ values $P, (H+x)^{-1}P, (H+x)^{-2}P, \cdots, (H+x)^{-k}P$, let $Q = (H+x)^{-k}P$, $tQ = (H+x)^{-(k-1)}P$, and so $t = (H+x)$.
Set the input of **k-wCDHP** to be

$$(H+x)^{-k}P = Q,\ (H+x)^{-(k-1)}P = tQ,$$
$$(H+x)^{-(k-2)}P = t^2Q,\ \cdots\ ,$$
$$(H+x)^{-1}P = t^{k-1}Q,\ P = t^kQ.$$

Then, **k-wCDHP** outputs

$$t^{-1}Q = (H+x)^{-1}(H+x)^{-k}P = (H+x)^{-(k+1)}.$$

k+1-IEP $\Rightarrow$ **k-wCDHP**:
Given $k+1$ values $P, (H+x)P, (H+x)^2P, \cdots, (H+x)^kP$, let $Q = (H+x)^kP$, $t^{-1}Q = (H+x)^{(k-1)}P$, and so $t = (H+x)$.
Set the input of **k+1-IEP** to be

$$(H+x)^kP = Q,\ (H+x)^{(k-1)}P = t^{-1}Q,$$
$$(H+x)^{(k-2)}P = t^{-2}Q,\ \cdots\ ,$$
$$(H+x)P = t^{-(k-1)}Q,\ P = t^{-k}Q.$$

Then, **k+1-IEP** outputs

$$t^{-(k+1)}Q = (H+x)^{-1}P.$$

■

We note that **k+1-IEP** and **k-wCDHP** are no harder than the CDHP. There is a special case that **k+1-IEP** or **k-wCDHP** can be easily solved :
Given

$$P_0 = P,\ P_1 = (H+x)^{-1}P,\ P_2 = (H+x)^{-2}P,\ \cdots\ ,$$
$$P_{(k-1)} = (H+x)^{-(k-1)}P,\ P_k = (H+x)^{-k}P,$$

if $P_i = P_j$ for $i \neq j$, this means that $(H+x)^{-i}P \equiv (H+x)^{-j}P \pmod{q}$, so the order of $(H+x)$ in $\mathbb{Z}_q$ is $j-i$. Then,

$$(H+x)^{-1}P = P_{j-i-1} \text{ or } (H+x)^{k+1}P = P_{k+1 \mod (j-i)}.$$

This case gives an attack on our proposed signature scheme. However, because of considering $(H+x)$ as a random element in $\mathbb{Z}_q^*$, we can show that the success probability of this attack is negligible.
Let q be a prime. Then, for any $a \in \mathbb{Z}_q^*$, the order of a, $ord(a)$, is a divisor of $q-1$. Given $k > 1$, assume that the number of element $a \in \mathbb{Z}_q^*$ such that $ord(a) \leq k$ is given by N. Since $\mathbb{Z}_q$ is a field, $N < k^2$ for $k > 1$. Let ρ be the probability that a randomly chosen element in $\mathbb{Z}_q^*$ has order less than k, then

$$\rho = \frac{N}{q} < \frac{k^2}{q}.$$

This gives us an opportunity to give a bound on k, such as, if $q \approx 2^{256}$, we limit $k \leq 2^{64}$, which means that the attacker has at most 2^{64} message-signature pairs. Therefore, using the above attack, the success probability is

$$\frac{(2^{64})^2}{2^{256}} = 2^{-128} \cdot 0.29387 \cdot 10^{-38}.$$

As a result, we have the following corollary.

Corollary 1: Assume that there is no polynomial time algorithm to solve the problem **k+1-IEP** with non-negligible probability, then the proposed signature scheme is secure under the random Oracle model.

3.3. Efficiency

We compare our signature scheme with the BLS scheme and ZSS scheme from the implementation point of view. PO, SM, PA, Squ, Inv, MTP and H denote the pairing operation, scalar multiplication in G_1, point addition in G_1, squaring in $\mathbb{Z}_n$, inversion in $\mathbb{Z}_n$, `MapToPoint` hash operation and hash operation in $\mathbb{Z}_n$, respectively. In the light of above, Table 1 summarizes the result.

Table 1
Comparison of our scheme with the BLS scheme and ZSS scheme

Scheme	BLS	ZSS	Proposed
Key generation	1 SM	1 SM	2 SM
Signing	1 MTP 1 SM	1 H 1 Inv 1 SM	1 H 1 Inv 1 SM 1 Squ
Verification	1 MTP 2 PO	1 H 1 SM 1 PO	1 H 1 SM 1 PO 2 PA 1 Squ

We implemented proposed signature scheme by using Pairing-Based Cryptography (PBC) Library [13] and The GNU Multiple Precision Arithmetic Library (GMP) [14]. Both libraries are installed as default installation. We run Cygwin as Linux simulator for GMP. The performance of signature schemas was measured on an Intel Core Duo

Table 2
Time comparison of our scheme with the BLS scheme and ZSS scheme

Scheme	BLS	ZSS	Proposed
All time including: key generation, signing, verification [s]	0.171000	0.098000	0.101000

1.6 GHz with 2 GB RAM, running Windows XP SP2. We have used standard functions of GMP 4.2.1/PBC 0.4.18 and compiled by GNU C Compiler. It should be noted that computation of pairing is the most time-consuming part in short signature schemes. According to the implementation result given in Table 2, our new scheme is more efficient than BLS scheme since it requires less pairing operation.

4. A Ring Signature Scheme

Ring signature schemes were proposed in [15]. Main purpose of a ring signature is to provide anonymity for the signer, by making it impossible to determine who among the possible signers is the actual one. By this way, the signature provides anonymity for the signer. Ring signature schemes satisfy signer ambiguity and security against an adaptive chosen message attack. A ring signature scheme is defined by:

- **ring signing** $(m, P_1, P_2, \cdots, P_r, x_i)$ produces a ring signature σ for the message m and a ring with r members, given the public keys $P_1, P_2, \cdots, P_r$ together with secret key of the signer x_i.
- **ring verifying** a signature pair (m, σ) includes the public keys of all the ring members i.e. possible signers.

The system parameters are $\{G_1, G_2, e, n, r, P, H\}$ which are defined in Section 3.1.

- **Sign:** Assume that the ith member of the ring sign the message. Let the public keys of the ring members be P_{pub1j} and P_{pub2j}, the secret key of the signer be x_i. Then,

$$\begin{aligned} S_i &= (H(m)+x_i)^{-2}P + (H(m) \sum_{j=1,i\neq j}^{r-1} 2x_jP \\ &+ \sum_{j=1,i\neq j}^{r-1} (x_j^2P + H(m)^2P)) \end{aligned}$$

- **Verify:**

$$\prod_{j=1}^{r} e((H(m)+x_j)^2P, S_i) = e(P,P).$$

Proof:

$$\begin{aligned} & \prod_{j=1}^{r} e((H(m)+x_j)^2P, S_i) \\ &= e(\sum_{j=1}^{r} (H(m)+x_j^2)P, S_i) \\ &= e(\sum_{j=1}^{r} (H(m)+x_j^2)P, (H(m)+x_i)^{-2}P \\ &+ (H(m) \sum_{j=1,i\neq j}^{r-1} 2x_jP + \sum_{j=1,i\neq j}^{r-1} (x_j^2P + H(m)^2P)) \\ &= e(P,P). \end{aligned}$$

■

The security of the proposed ring signature scheme is similar as given in Section 3.2 since it is based on the signature scheme described in Section 3.1.

5. Conclusion

In this paper, we propose a new short signature scheme not requiring any special hash function. The security of this signature scheme depends on a new problem called bilinear inverse-square Diffie-Hellman problem (BISDHP). It is shown that this problem and BDHP are polynomial time equivalent. We also propose a new complexity assumption called the $k+1$ inverse exponent problem. The exact security proofs are also explained in the random Oracle model. We give the implementation and comparison results of our proposed signature scheme with the BLS and ZSS schemes. According to the implementation results, our new scheme is more efficient than BLS scheme since it requires less pairing operation. Finally, we construct a ring signature scheme based on our proposed scheme.

Acknowledgments

A preliminary version of this work was presented at Information Systems Technology Panel Symposium, 2010 (IST-091/RSY-021). We would like to thank Professor Ersan Akyıldız for his valuable comments. This research is partially supported by ASELSAN A.Ş. with the project 2008.15.00.2.00.02.

References

[1] *Digital Signature Standard*, FIPS PUB 186. National Institute of Standards and Technology, 1994.

[2] D. Boneh, B. Lynn and H. Shacham, "Short signatures from the Weil pairing", in *Advances in Cryptology – Asiacrypt 2001*, Lecture Notes in Computer Science, vol. 2248. Berlin: Springer, 2001, pp. 514–532.

[3] P. S. L. M. Barreto and H. Y. Kim, "Fast hashing onto elliptic curves over fields of characteristic 3", Cryptology ePrint Archive, Report 2001/098 [Online]. Available: http://eprint.iacr.org/2001/098/

[4] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing", in *Advances in Cryptology – Crypto 2001*. Lecture Notes in Computer Science, vol. 2139, Berlin: Springer, 2001, pp. 213–229.

[5] D. Boneh and X. Boyen, "Short signatures without random Oracles", in *Advances in Cryptology – Eurocrypt 2004*, Lecture Notes in Computer Science, vol. 3027. Berlin: Springer, 2004, pp. 56–73.

[6] D. Boneh, X. Boyen and H. Shacham, "Short group signatures", in *Advances in Cryptology – Crypto 2004*, Lecture Notes in Computer Science, vol. 3152. Berlin: Springer, 2004, pp. 41–55.

[7] F. Zhang, R. Safavi-Naini and W. Susilo, "An efficient signature scheme from bilinear pairings and its applications", in *Advances in Cryptology – PKC 2004*, Lecture Notes in Computer Science, vol. 2947. Berlin: Springer, 2004, pp. 277–290.

[8] A. R. Sadeghi and M. Steiner, "Assumptions related to discrete logarithms: why subtleties make a real difference", in *Advances in Cryptology – Eurocrypt 2001*, Lecture Notes in Computer Science, vol. 2045. Berlin: Springer, 2001, pp. 243–260.

[9] A. Joux, "A one round protocol for tripartite Diffie-Hellman", in *Advances in Cryptology – ANTS 4*, Lecture Notes in Computer Science, vol. 1838. Berlin: Springer, 2000, pp. 385–394.

[10] S. Goldwasser, S. Micali and R. Rivest, "A digital signature scheme secure against adaptive chosen-message attacks", *SIAM J. Comput.*, vol. 17, no. 2, pp. 281–308, 1988.

[11] M. Bellare and P. Rogaway, "The exact security of digital signatures – how to sign with RSA and Rabin", in *Advances in Cryptology – Eurocrypt 1996*, Lecture Notes in Computer Science, vol. 1070. Berlin: Springer, 1996, pp. 399–416.

[12] S. Mitsunari, R. Sakai and M. Kasahara, "A new traitor tracing", *IEICE Trans. Fundamentals*, vol. E85-A, no. 2, pp. 481–484, 2002.

[13] T*he Pairing-Based Cryptography (PBC) Library* [Online]. Available: http://crypto.stanford.edu/pbc/

[14] *The GNU Multiple Precision Arithmetic Library (GMP)* [Online]. Available: http://gmplib.org/

[15] R. L. Rivest, A. Shamir and Y. Tauman, "How to leak a secret", in *Advances in Cryptology – Asiacrypt 2001*, Lecture Notes in Computer Science, vol. 2248. Berlin: Springer, 2001, pp. 552–565.

Sedat Akleylek received the B.Sc. degree in mathematics form Ege University, in Turkey, in 2004, the M.Sc. degree and the Ph.D. degree in cryptography both from Middle East Technical University (METU), in Turkey in 2008 and 2010, respectively. He has been with Institute of Applied Mathematics, METU and Department of Computer Engineering, Ondokuz Mayis University, Turkey since 2005. His research interest are in the areas of cryptography, algorithms and architectures for computations in Galois fields, computer algebra and e-learning.

e-mail: akleylek@metu.edu.tr
Institute of Applied Mathematics
Middle East Technical University
06531 Ankara, Turkey

Department of Computer Engineering
Ondokuz Mayıs University
55139 Samsun, Turkey

Barış Bülent Kırlar received his Ph.D. in cryptography in 2010 from the Institute of Applied Mathematics (IAM), Middle East Technical University (METU), Turkey. Since 2004 he has been with IAM, METU and Department of Mathematics, Süleyman Demirel University, Turkey where he is a research assistant. His research interests are number theory, finite fields and cryptography, with emphasis on elliptic curve cryptography.

e-mail: kirlar@metu.edu.tr
Institute of Applied Mathematics
Middle East Technical University
06531 Ankara, Turkey

Department of Mathematics
Süleyman Demirel University
32260 Isparta, Turkey

Ömer Sever received the B.S. degree in computer engineering in Middle East Technical University (METU), Turkey, and the M.Sc. degree in cryptography in METU, Turkey, in 2002 and 2007, respectively. Since 2002, he has been working in Turkish Navy as engineer officer. He is continuing Ph.D. in Cryptography in METU. His research interests are in the areas of computer security, cryptography and algorithms.

e-mail: severomer@yahoo.com
Institute of Applied Mathematics
Middle East Technical University
06531 Ankara, Turkey

Zaliha Yüce received the B.S. degree in computer engineering and the M.Sc. degree in cryptography in Middle East Technical University (METU), Turkey, in 2004 and 2007, respectively. Since 2008, she is working in STM A.Ț and still Ph.D. student in Cryptography Department of METU. Her research interests are software implementations of pairing-based cryptography protocols.

e-mail: zyuce@stm.com.tr
Institute of Applied Mathematics
Middle East Technical University
06531 Ankara, Turkey

Software engineer
STM A.Ş, 06800 Ankara, Turkey