Copyright ⓒ 2015 KSII

An Efficient Identity-Based Deniable Authenticated Encryption Scheme

Weifeng Wu[1] and Fagen Li[2]
School of Computer Science and Engineering
University of Electronic Science and Technology of China
Chengdu, 611731, China
[e-mail: [1]asd69.good@163.com; [2]fagenli@uestc.edu.cn]
*Corresponding author: Weifeng Wu

Received January 7, 2015; revised April 7, 2015; accepted May 3, 2015; published May 31, 2015

Abstract

Deniable authentication protocol allows a sender to deny his/her involvement after the protocol run and a receiver can identify the true source of a given message. Meanwhile, the receiver has no ability to convince any third party of the fact that the message was sent by the specific sender. However, most of the proposed protocols didn't achieve confidentiality of the transmitted message. But, in some special application scenarios such as e-mail system, electronic voting and Internet negotiations, not only the property of deniable authentication but also message confidentiality are needed. To settle this problem, in this paper, we present a non-interactive identity-based deniable authenticated encryption (IBDAE) scheme using pairings. We give the security model and formal proof of the presented IBDAE scheme in the random oracle model under bilinear Diffie-Hellman (BDH) assumption.

Keywords: Bilinear pairings, Deniable authentication, ID-based cryptography, Random oracle model

This work was supported by the National Natural Science Foundation of China under Grant No. 61272525 and the Fundamental Research Funds for the Central Universities under Grant No. ZYGX2013J069.

http://dx.doi.org/10.3837/tiis.2015.05.020 **ISSN : 1976-7277**

1. Introduction

Deniable authentication protocol plays an important role in practice and it is very useful in some special scenarios such as e-mail system, electronic bidding, electronic voting and negotiations over the Internet [1]. Compared with traditional authentication protocol, it has the following two characters: (1) The protocol principals can deny their involvement after the protocol run and the intended receiver can verify the source of a given message. (2) However, the intended receiver cannot convince any third party of the fact that the message was sent by the specific sender, even if fully-cooperate with the third party. Consider the following application scenario.

Electronic voting system: In an electronic voting system, let V be a voter and T be a tallying authority. When V submits his/her ballot *m* to T, he/she should send *m* together with the authenticator of *m* to T, so that T can make sure this ballot really comes from V but not from anyone else. Suppose a third party F compels V to elect a predetermined candidate but V does not want to do so. When V submits his/her ballot *m* to T in this situation, it is desirable for V to assure that T has no ability to prove the true source of the ballot *m* to F even if T and F co-operates fully. That is because even if there is full co-operation between T and F, F may also be skeptical of the evidence provided by T. Thus, F cannot force the voter V to elect the candidate predetermined by him/her. And in this way the property of fairness which is very important for electronic voting system is achieved perfectly. Therefore, such a deniable authentication protocol is needed to protect the voter V from coercion in electronic voting system.

However, in some cases, the content of the transmitted message (such as the content of ballot in electronic voting system) in a deniable authentication protocol should only be shared between the sender and intended receiver, any other third party should not have the ability to obtain the transmitted transcripts. That is, we should provide confidentiality to protect confidential data and sensitive information from eavesdropping and network-sniffing attack. We call a cryptographic scheme that achieves simultaneously confidentiality and deniable authentication deniable authenticated encryption (DAE) scheme.

1.1 Related Work

Several deniable authentication protocols have been proposed over the past few years. In 1998, Dwork et al. [2] first proposed a deniable authentication protocol based on zero-knowledge. However, the protocol requires a timing constraint, and the proof of knowledge is time-consuming [3]. Later, Aumann and Rabin [4] proposed another scheme based on factoring which needs a public directory trusted by the two communication parties. In 2001, Deng et al. [1] proposed two deniable authentication protocol based on factoring and discrete logarithm problem, respectively, under the

communication model defined by Aumann and Rabin in 1998. However, in 2006, Zhu et al. [5] proved that both of the protocols were vulnerable to the person-in-middle (PIM) attack. In 2005, Cao et al. [6] proposed an efficient non-interactive identity-based deniable authentication protocol from pairings. What's more, the scheme achieves confidentiality by employing a symmetric encryption algorithm. However, in 2006, Chou et al. [7] pointed out that Cao et al.'s scheme suffered from key compromise impersonation (KCI) attack. Then Chou et al. presented a new identity-based deniable authentication protocol from pairings, and claimed that the protocol is secure. But in 2007, Lim et al. [8] proved that Chou et al.'s scheme remains insecure due to the vulnerability to the KCI attack. Moreover, they presented an enhanced scheme. But later in 2008, they found that their enhanced scheme suffered from the insider KCI attack and key replicating attack, and then they repaired the secure flaw in [9]. Unfortunately, in 2011, Tian et al. [10] pointed out that the repaired protocol by Lim et al. was still not secure under a special KCI attack. Besides, in 2005, Shi and Li [11] proposed an identity-based non-interactive deniable authentication protocol in which a signature scheme is needed. And this results in less efficient than scheme in [6]. In 2013, Li et al. [12] proposed an efficient identity-based deniable authentication protocol for ad hoc networks. What's more, they gave the security model and formal proof of their protocol, and claimed that their protocol met the requirements of batch verification and was faster than all known identity-based deniable authentication protocols.

The concept of identity-based cryptography (IBC) was first introduced by Shamir [13] in 1984, and the original motivation of IBC was to simplify certificate management. In IBC systems, every user's public keys can be obtained from their public information such as e-mail address, name, telephone number and so on. The certificate is not necessary in an IBC system. However, the first practical identity-based encryption scheme was proposed until 2001 by Boneh and Franklin in [14] using Weil pairing. In an identity-based system, a trusted private key generator (PKG) is required to generate the private key corresponding to some public key of a user, and then PKG sends the private key to the user via a secure channel. However, as we know, most of the existed identity-based deniable authentication protocols transmit the message in plaintext form over an insecure public network which does not meet our requirements in practice. During the schemes mentioned above, only scheme in [6] achieved message confidentiality along with deniable authentication. It is desirable to design a new kind of protocol that achieves both properties of deniable authentication and confidentiality simultaneously within one logical step. Motivated by the advantages of IBC, in this paper, we present an IBDAE scheme with pairings that can meet the requirements in practice.

1.2 Contribution

In this paper, we first define a security model for IBDAE scheme and then present an IBDAE scheme using bilinear pairings. Next, we give the formal proof of our scheme in the random oracle model under the BDH assumption. After that, we compare the performance of our IBDAE scheme with the other three identity-based deniable

authentication protocols. Generally speaking, our IBDAE scheme achieves the following three properties simultaneously:

Confidentiality: This property assures that only the intended receiver can share the transmitted message with the sender, any third person has no ability to gain the transmitted transcripts.

Deniability: The sender of the protocol can later deny the authorship of the transmitted message and even deny having taken part in the communication run. At the same time, the intended receiver can identify the true source of a given message, but he/she cannot prove this fact to any other third party.

Authentication: This property assures that the protocol principal is actually the person he/she claims to be, rather than another third person or an adversary.

1.3 Organization

This paper is organized as follows. In Section 2, we introduce the preliminary work that will be used later in this paper. In Section 3, we define the security model of IBDAE, and our scheme is presented in Section 4. In Section 5, we provide the security proof of the proposed scheme and compare its performance with other schemes. Finally, we draw the conclusions in Section 6.

2. Preliminaries

In this section, we simply review the basic concept and some properties of bilinear pairings.

Let G_1 be a cyclic additive group generated by P, whose order is a prime q, and G_2 be a cyclic multiplicative group of the same order q. A bilinear pairings is a map $\hat{e}: G_1 \times G_1 \to G_2$, which satisfies the following three properties:

(1) Bilinear: $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$ for all $P, Q \in G_1$.

(2) Non-degeneracy: There exists $P, Q \in G_1$ such that $\hat{e}(P, Q) \neq 1$.

(3) Computability: There is an efficient algorithm to compute $\hat{e}(P, Q)$ for all $P, Q \in G_1$.

A bilinear pairings that satisfies the three properties above is said to be an admissible bilinear pairings, and the modified Weil pairing or Tate pairing are admissible maps of this kind. There is a more concrete description in [14]. The security of our IBDAE scheme relies on the following two related mathematical problems in G_1.

(1) Discrete Logarithm Problem (DLP): Given $P, Q \in G_1$, find an integer a, such that $Q=aP$, whenever such an integer exists.

(2)Bilinear Diffie-Hellman Problem (BDHP): Given $P, aP, bP, cP \in G_1$, computes $\hat{e}(P,P)^{abc} \in G_2$, where $a,b,c \in \mathbb{Z}_q^*$.

3. Formal Model for IBDAE

3.1 Syntax

A generic IBDAE scheme consists of four algorithms: system setup (**Setup**) algorithm, key extraction algorithm (**Extract**), identity-based deniable authenticated encryption (**IBDAE**) algorithm and identity-based deniable authenticated decryption (**IBDAD**) algorithm. Now, we describe these algorithms as follows.

Setup is a probabilistic algorithm run by PKG that takes as input a security parameter k, and outputs the system parameters *param*, a master public key P_{pub} and a master secret key s, In this paper, we simplify the input of other algorithms by assuming that the system parameters *param* are public, thus we do not need to contain them in other algorithms.

Extract is a key extraction algorithm run by PKG that takes as input an identity *ID* and the master secret key s and outputs the private key S_{ID} corresponding to *ID*. Then PKG transmits this private key to the user of identity *ID* via a secure channel.

IBDAE is a probabilistic algorithm run by a sender that takes as input a message m, a sender's private key S_S, the identities of a sender and receiver ID_S and ID_R, and outputs an IBDAE authenticator δ=IBDAE (m, S_S, ID_S, ID_R).

IBDAD is a deterministic decryption and verification algorithm run by a receiver that takes as input an IBDAE authenticator δ, a receiver's private key S_R, the identity ID_R of a receiver and outputs the recovered message m from IBDAD (δ, S_R, ID_R) if δ is a valid IBDAE authenticator of m between the sender and receiver. Otherwise, an error symbol ▲will be output.

The algorithms above must have the following consistency requirements. If δ=IBDAE (m, S_S, ID_S, ID_R), then we must have m= IBDAD (δ, S_R, ID_R) holds.

3.2 Security Model

(1) Confidentiality of IBDAE: The security notion of confidentiality of an IBDAE scheme is called ciphertext indistinguishability against adaptive chosen ciphertext attacks (IND-IBDAE-CCA2). Here, we consider the following game played between a challenger X and an adversary A.

Initial: X runs **Setup** algorithm that takes as input a security parameter k and sends the system parameters *param* to A, while keeps the master secret key s secret.

Phase 1: A performs a polynomial bounded number of key extraction queries, deniable authenticated encryption queries and deniable authenticated decryption queries in an adaptive way as follows.

① In a key extraction query, A submits an identity *ID* to X, then X runs **Extract** algorithm to obtain the corresponding private key S_{ID} and sends it to the adversary A.

② In a deniable authenticated encryption query, A selects a sender's identity ID_i, a receiver's identity ID_j, a plaintext m and sends them to the challenger X. X first runs **Extract** algorithm to obtain the private key S_i corresponding to ID_i and then sends the result of δ=IBDAE (m, S_i, ID_i, ID_j) to A.

③ In a deniable authenticated decryption query, the adversary A submits an IBDAE authenticator δ and a receiver's identity ID_j to X, then X runs **Extract** algorithm to obtain the receiver's private key S_j. After that, X sends the result of IBDAD (δ, S_j, ID_j) to A.

Challenge: Once A decides that Phase 1 is over, he/she outputs two plaintexts m_0, $m_1$with equal length, and two identities ID_S, ID_R that haven't made key extraction queries. Then A sends them to the challenger X. X randomly chooses a bit $k \in \{0,1\}$, computes δ= IBDAE (m_k, S_S, ID_S, ID_R), and sends δ to A as his challenge.

Phase 2: A can issue more polynomial bounded queries like Phase 1, but in this phase, A can not make key extraction queries on identities ID_S and ID_R. Meanwhile he can't make IBDAD queries on his challenge δ either.

Guess: Finally, A outputs a bit $k_1 \in \{0,1\}$. If $k_1=k$, we say A wins the game.

We define the advantage of A as the probability $|\Pr[k_1=k]-1/2|$.

(2) Unforgeability of IBDAE: We borrow the concept of unforgeability against adaptive chosen messages attacks in digital signature to define this security notion. However, the security notion in IBDAE scheme is essentially different from that in a digital signature scheme. That is because only the sender with correct private key has the ability to generate a valid signature in digital signature, while in an IBDAE scheme, both the sender and the receiver have ability to generate a valid IBDAE authenticator. We call this security notion deniable authentication against adaptive chosen messages attacks (DA-IBDAE-CMA). Here, we consider the following game played between a challenger X and an adversary Φ.

Initial: X runs **Setup** algorithm that takes as input a security parameter k and sends the system parameters *param* to Φ, while keeps the master secret key s secret.

Attack: Φ performs a polynomial bounded number of key extraction queries, deniable authenticated encryption queries and deniable authenticated decryption queries as Phase 1 in the model of confidentiality above.

Forgery: Φ outputs an IBDAE authenticator δ^* and two identities ID_S and ID_R. Φ wins the game if the following three conditions hold:

① δ^* is a valid IBDAE authenticator with identities ID_S and ID_R, it means that the result of IBDAD (δ^*, S_R, ID_R) doesn't be an error symbol ▲.

② Φ has not made key extraction queries on identities ID_S and ID_R.

③ Φ has not made a deniable authenticated encryption query with a message m^* and identities ID_S, ID_R.

We define the advantage of Φ as the probability that it wins in the game. In order to achieve the property of deniability in an IBDAE scheme, we require that Φ should not have made key extraction queries on both identities ID_S and ID_R in the second step of the three conditions. It is because the receiver can also generate a valid IBDAE authenticator.

4. Proposed IBDAE Scheme

In this section, we give the precise algorithms of our IBDAE scheme with bilinear pairings. Our scheme consists of the following four algorithms.

Setup: Suppose k is a secure parameter, G_1 is a cyclic additive group generated by P and G_2 is a cyclic multiplicative group, both groups have the same prime order $q(q\geq 2^k)$ and the discrete logarithm problems in both G_1 and G_2 are hard when $k\geq 160$. A bilinear map is $\hat{e}: G_1 \times G_1 \to G_2$. Let H, H_1, H_2 be three security cryptographic hash functions where $H: \{0,1\}^* \to G_1$, $H_1: G_2 \to \{0,1\}^{l_m + l_{ID}}$ and $H_2: \{0,1\}^{l_m + 2l_{ID}} \times G_1 \to \mathbb{Z}_q^*$ in which l_m denotes the length of message and l_{ID} denotes the length of an identity string. The PKG randomly chooses a master secret key $s \in \mathbb{Z}_p^*$ and computes $P_{pub}=sP$. PKG publishes the system parameters $param = \{G_1, G_2, P, P_{pub}, \hat{e}, q, l_m, l_{ID}, H, H_1, H_2\}$, while keeps the master secret key s secret.

Extract: Given an identity ID, PKG computes the private key corresponding to ID as $S_{ID}=sQ_{ID}$, where $Q_{ID}=H(ID)$ is the public key of the identity ID. Then PKG sends the private key S_{ID} to the user via a security channel. Throughout this paper, we assume that the public/private key pairs of the sender and receiver are (Q_S, S_S) and (Q_R, S_R), respectively.

IBDAE: Given a message m, a sender's private key S_S, the identities of a sender and receiver ID_S and ID_R, this algorithm works as follows.

(1) Choose $r \in \mathbb{Z}_q^*$ randomly, compute $U=rQ_S$ and $T = \hat{e}(S_S, Q_R)^r$.

(2) Compute $C = H_1(T) \oplus (m \| D_S)$.

(3) Compute $h = H_2(m \| ID_S \| ID_R, U)$.

(4) Compute $V = \hat{e}((r+h)S_S, Q_R)$.

Then the sender sends the IBDAE authenticator $\delta=(U,C,V)$ to the receiver.

IBDAD: After receiving the IBDAE authenticator $\delta=(U,C,V)$, the receiver performs IBDAD algorithm with his/her private key S_R and identity ID_R as follows.

(1) Compute $(m \| ID_S) = H_1(\hat{e}(U, S_R)) \oplus C$ to recover plaintext and sender's identity $m \| ID_S$.

(2) Compute $h = H_2(m \| ID_S \| ID_R, U)$.

(3) Check if $V = \hat{e}(U + hQ_S, S_R)$ holds. If yes, accept the recovered message m, otherwise, this algorithm outputs an error symbol ▲.

5. Analysis of Scheme

5.1 Security

The consistency of our scheme is obvious, so we mainly analyze the deniability, confidentiality and deniable authentication of our scheme in this section.

5.1.1 Deniability

Proof: Our scheme achieves the property of deniability since the receiver can also generate a valid IBDAE authenticator that indistinguishable from that of a sender for the following reasons.

(1) After receiving the IBDAE authenticator δ=(U,C,V) from a sender, the receiver can obtain the recovered message m by running IBDAD algorithm. Then he/she performs the processes as follows.

① $C_1 = H_1(\hat{e}(U, S_R)) \oplus (m \| ID_S)$.

② $h = H_2(m \| ID_S \| ID_R, U)$.

③ $V_1 = \hat{e}(U + hQ_S, S_R)$.

It's obvious that even if for the same message m, δ_1=(U,C_1,V_1) is indistinguishable from δ=(U,C,V) for any third party.

(2) The receiver can also chooses a random number $r_1 \in \mathbb{Z}_q^*$ and computes U_1=r_1Q_S, then performs as (1) above. The output of this situation will be δ_1=(U_1,C_1,V_1) which also indistinguishable from δ=(U,C,V).

5.1.2 Confidentiality

Theorem 1 In random oracle model, if there exists an adversary A that wins IND-IBDAE-CCA2 game with advantage ε within time t for a security parameter k and consults at most q_H queries to oracles H, q_{H_i} queries to oracles H_i (i=1,2), q_K key extraction queries, q_E DAE queries and q_D DAD queries. Then there exists an algorithm

X that can solve the BDH problem in time $t+O(q_E+q_D)T_e$ (T_e denotes the computation time of bilinear map) with advantage

$$Adv(C) > \frac{\varepsilon(2^k - q_E - q_{H_2} - q_{H_1} - q_D) - (q_E + q_{H_2} + q_{H_1} + q_D)}{2^{k+1} q_H} - \frac{\varepsilon + 1}{q_H^2}.$$

Proof: We can prove the confidentiality of our scheme via the game defined in section three. If there is an adversary A that can break the scheme, we can use A to build a new algorithm X to solve the BDH problem. Next we show how X solves the BDH problem which means X takes as input (P,aP,bP,cP) and outputs $\hat{e}(P,P)^{abc}$ by interacting with A. In the following game, X acts as the challenger of A. A will consult X for the answers of the random oracles H, H_1 and H_2. To track the queries of H, H_1 and H_2, X maintains three lists L, L_1, L_2 to store the answers respectively. The three lists are all empty at first. We describe the processes as follows.

Initial: X runs **Setup** algorithm that takes as input a security parameter k and sends the system parameters *param* in which $P_{pub}=cP$ (c plays as the master secret key and X knows nothing about c) to A.

Phase 1: A performs a polynomial bounded number of the following queries.

(1) H queries: At first, the challenger X chooses two different numbers $N_1,N_2 \in \{1,2,\ldots,q_H\}$. At the N_1-th query of H, X answers $H(ID_{N_1}) = aP$, while at the N_2-th query of H, X answers $H(ID_{N_2}) = bP$. For an identity ID_i ($i \in \{1,2,\ldots,q_H\}$ and $i \neq N_1,N_2$) given by A, X first looks up the list L and checks if the value of $H(ID_i)$ was previously defined. If it was, the previous defined value will be the result of this query. Otherwise, X chooses a random number $n_i \in \mathbb{Z}_q^*$, inserts the 2- tuples (ID_i, n_i) into L, and returns $H(ID_i)=n_iP$ as the result of this query.

(2) H_1 queries: For a query of $H_1(T_i)$, X first looks up list L_1, and checks if there exists a previous defined value of $H_1(T_i)$. If it was, the existed value will be returned as the result of this query. Otherwise, X selects a random value $H_1^i \in \{0,1\}^{l_m+l_{ID}}$ and inserts 2-tuples (T_i, H_1^i) into list L_1. Then he/she sets $H_1(T_i)= H_1^i$ as the result of this query.

(3) H_2 queries: For a query of $H_2(m_i \| ID_i \| ID_j, U_i)$, X first checks if the value of $(m_i \| ID_i \| ID_j, U_i)$ was previously defined in list L_2. If it was, the previous defined value will be returned as the result of the query. Otherwise, X returns a random number $h_i \in \mathbb{Z}_q^*$ as the answer of the query and inserts the tuples $(m_i \| ID_i \| ID_j, U_i, h_i)$ into list L_2.

(4) Key extraction queries: A submits an identity ID_i to X, X fails and terminates if $i=N_1$ or $i=N_2$. Otherwise, X first looks up the list L to obtain the value n_i where $H(ID_i)=n_iP$, computes the corresponding private key $S_i=n_iP_{pub}$ and sends it to the adversary A. The probability of fail in key extraction queries is at most $2/q_H$.

(5) Deniable authenticated encryption queries: A submits a sender's identity ID_i, a receiver's identity ID_j and a plaintext m to X. If $i \neq N_1, N_2$, X first obtains the private key $S_i = n_i P_{pub}$, runs **IBDAE** algorithm and sends the result δ=IBDAE (m, S_i, ID_i, ID_j) to A. If $i=N_1$ or $i=N_2$, but $j \neq N_1, N_2$, X obtains the private key $S_j = n_j P_{pub}$, then he/she chooses $r \in \mathbb{Z}_q^*$ and computes $U=rQ_i$, $C = H_1(\hat{e}(U, S_j)) \oplus (m \| ID_i)$. After that, X runs the H_2 simulation algorithm to obtain h, then computes $V = \hat{e}(U + hQ_i, S_j)$. At last, X sends $\delta=(U,C,V)$ to A. However, if $i=N_1$ and $j=N_2$ (or $j=N_1$ and $i=N_2$), X chooses $r, h \in \mathbb{Z}_q^*$ and sets $U=rP-hQ_i$, $W=rP_{pub}$, at the same time he defines $H_2(m \| ID_i \| ID_j, U) = h$ and inserts the tuple $(m \| ID_i \| ID_j, U, h)$ into list L_2. After that, X computes $V = \hat{e}(W, Q_j)$. Besides, X chooses a random value $C \in \{0,1\}^{l_m + l_{ID}}$ and computes $H_1(T) = C \oplus (m \| ID_i)$, $T = V^{\frac{r}{r+h}}$. At the same time, X inserts the 2- tuples (T, $H_1(T)$) into list L_1. Finally, X sends $\delta=(U,C,V)$ to A. X fails and terminates if the values of H_1, H_2 are already defined. And the probability of fail in this phase is at most $\frac{q_E + q_{H_2} + q_{H_1}}{2^k}$.

(6) Deniable authenticated decryption queries: A submits an IBDAE authenticator $\delta=(U,C,V)$ and a receiver's identity ID_j to X. If $j \neq N_1$ and $j \neq N_2$, X obtains the receiver's private key $S_j = n_j P_{pub}$, computes $T_j = \hat{e}(U, S_j)$ and runs the H_1 simulation algorithm to obtain $H_1(T_j)$. After that, X computes $(m \| ID_i) = H_1(T_j) \oplus C$ to recover $m \| ID_i$, runs the H_2 simulation algorithm to obtain $h = H_2(m \| ID_i \| ID_j, U)$, and then computes $V_1 = \hat{e}(U + hQ_i, S_j)$. X sends the recovered message m to A if $V_1=V$ holds. Otherwise X tells A that the IBDAE authenticator is invalid and returns an error symbol ▲ to A. However, if $j=N_1$ or $j=N_2$, X will always tell the adversary A that the IBDAE authenticator is invalid as X can't compute the private key of ID_j. The probability of fail in this situation is at most $q_D / 2^k$.

Challenge: Once A decides that Phase 1 is over, he outputs two plaintexts m_0, m_1 with equal length, and two identities ID_S, ID_R that didn't make key extraction query. X fails if A consults X for $H(ID_S)$ or $H(ID_R)$ during the game. However, if ID_S and ID_R are not identities ID_{N_1} and ID_{N_2} ($S \neq N_1, R \neq N_2$ or $S \neq N_2, R \neq N_1$), X fails either. To compute the IBDAE authenticator, X performs as follows.

(1) X chooses $r \in \mathbb{Z}_q^*, T_S \in G_2$ randomly and computes $U=raP$. Then X runs the H_1 simulation algorithm to obtain $H_1(T_s)$ and computes $C = H_1(T_S) \oplus (m_\gamma \| ID_S)$ where $\gamma \in \{0,1\}$ was chosen by X randomly.

(2) X runs the H_2 simulation algorithm to obtain $h = H_2(m_\gamma \,||\, ID_S \,||\, ID_R, U)$ and computes $V = (T_S)^{r^{-1}(r+h)}$. Finally, X sends the IBDAE authenticator δ=(U,C,V) to the adversary A as his challenge.

Phase 2: A can issue more polynomial bounded queries like Phase 1, but A can not make key extraction queries on identities ID_S and ID_R, meanwhile he can't make deniable authenticated decryption query on the challenge δ either.

Guess: Finally, A outputs a bit $\gamma' \in \{0,1\}$. If $\gamma' = \gamma$, X outputs $(T_S)^{r^{-1}} = \hat{e}(P,P)^{abc}$.

To calculate the value of Adv(X), we take into consideration all the probabilities that X does not fail in the simulation above, the probability that the two challenged identities chosen by A are ID_{N_1} and ID_{N_2} and the probability that A wins the IND-IBDAE-CCA2 game. The value of Adv(X) is calculated as follows.

$$Adv(C) > (\frac{\varepsilon+1}{2}(1-\frac{q_E+q_{H_2}+q_{H_1}+q_D}{2^k}-\frac{2}{q_H})-\frac{1}{2})\frac{2}{q_H} = \frac{\varepsilon(2^k-q_E-q_{H_2}-q_{H_1}-q_D)-(q_E+q_{H_2}+q_{H_1}+q_D)}{2^{k+1}q_H}-\frac{\varepsilon+1}{q_H^2}$$

Thus, X solves the BDH problem under the help of A. However, as we assume that the BDH problem is hard and there is no efficient algorithm to solve the BDH problem at present, therefore, the adversary A doesn't actually exist and the confidentiality of our scheme is proved.

5.1.3 Deniable Authentication

Theorem 2 In random oracle model, if there is an adversary Φ can win the DA-IBDAE-CMA game with advantage $\varepsilon \geq 5(q_E+1)(q_E+q_{H_2})q_H / (2^k-1)$ within time t for a security parameter k and consult at most q_H queries to oracles H, q_{H_i} queries to oracles H_i (i=1,2), q_K key extraction queries, q_E DAE queries and q_D DAD queries. Then there exists an algorithm X that can solve BDH problem within an expected time $t' \leq 60343 q_{H_2} q_H 2^k t / \varepsilon(2^k-1)$.

Proof: We prove the deniable authentication of our scheme via the game defined in section three that played between a challenger X and an adversary Φ. If there exists an adversary Φ that can break the scheme, we can use Φ to build a new algorithm X to solve BDH problem. Just like the proof of confidentiality above, X maintains three lists L, L_1, L_2 to store the answers of the random oracles H, H_1 and H_2 during the game, respectively. The processes of the game are described as follows.

Initial: X runs **Setup** algorithm that takes as input a security parameter k and sends the system parameters *param* in which P_{pub}=cP (c plays as the master secret key and X knows nothing about c) to Φ.

Attack: Φ performs a polynomial bounded number of H queries, H_1 queries, H_2 queries, key extraction queries, deniable authenticated encryption queries and deniable authenticated decryption queries like those in Phase 1 of the proof of confidentiality above.

Forgery: Φ outputs an IBDAE authenticator $\delta^*=(U^*,C^*,V^*)$ and two identities ID_S and ID_R. From the forking lemma[15], if Φ is an efficient forger during the interaction above, we can construct a Las Vegas machine Φ_1 that outputs two results (ID_S, ID_R ,C^* ,h^*, V^*) and (ID_S, ID_R ,C^* , h_1^*,V_1^*) where $h^* \neq h_1^*$ with the same U^*. X solves the BDH problem by computing $\hat{e}(P,P)^{abc}=(\frac{V^*}{V_1^*})^{\frac{1}{(h^*-h_1^*)}}$ when S=N_1 and R=N_2 (or R=N_1 and S=N_2) in the game above.

Now we consider the advantage of X. From the forking lemma [15] and the lemma on the relationship between given identity attack and chosen-identity attack [16], if Φ succeeds in time t with probability $\varepsilon \geq 5(q_E+1)(q_E+q_{H_2})q_H/(2^k-1)$, then X can solve the BDH problem in expected time $t' \leq 60343 q_{H_2} q_H 2^k t/\varepsilon(2^k-1)$. However, since we assume that the BDH problem is hard and there is no efficient algorithm to solve the BDH problem at present, there doesn't actually exist such an adversary Φ. Thus we prove the deniable authentication of our scheme.

5.2 Analysis of Performance

In this section, we compare the performance of our scheme with those in [6, 7] and [12]. With regard to the major computational cost, we can see more details from **Table 1**, and we denote by AD the point addition in G_1, PM the point multiplication in G_1, EXP the exponentiation in G_2, P the pairing computation, CS the computational cost of the sender and CR the computational cost of the receiver in the table. The other operations are omitted for they are trivial. Besides, we assume that $|G_1|$=1024 bits, $|G_2|$=1024 bits, $|\square_q^*|$=160 bits, $|m|$=160 bits, $|ID|$=160 bits and hash value =160 bits. The total communication size of Cao et al. [6] is $|ID|+|\square_q^*|+|m|+|$hash value$|$ =640 bits, the total communication size of Chou et al. [7] is $2|ID|+|G_1|+|\square_q^*|+|m|+|$hash value$|$=1824 bits and both the communication sizes of Li et al. [12] and ours are $|ID|+|G_1|+|m|+|G_2|$=2368 bits. We can refer to **Fig.1** for the comparison of the communication size. The presented scheme in this paper has a little higher communication cost for the IBDAE authenticator contains an element in G_2. Although the scheme of Cao et al. [6] achieves confidentiality by employing a symmetric encryption algorithm, it suffers from the problem of key distribution in symmetric cryptography before a communication setups. However, our scheme overcomes this weakness by using identity-based encryption algorithm.

Meanwhile, the schemes of Chou et al. [7] and Li et al. [12] transmit message in an unencrypted form which may cause a big security flaw in practice.

Table 1. Performance Comparison

	Cao et al. [6]	**Chou et al.** [7]	**Li et al.** [12]	**Ours**
CS	2AD+2PM+P	4PM+EXP+4P	2PM+P	2PM+EXP+2P
CR	2AD+2PM+P	2PM+EXP+4P	AD+PM+P	AD+PM+2P
Confidentiality	**Yes**	No	No	**Yes**
Non-interactive	**Yes**	No	**Yes**	**Yes**
Resist KCI	No	No	**Yes**	**Yes**
Formal proof	No	No	**Yes**	**Yes**

For non-interactive deniable authentication protocol, we should consider the weaken-key compromise impersonation [17] (W-KCI) instead of KCI attack. In a W-KCI attack, the compromising long-term private key of a receiver can make an adversary have the ability to masquerade other users to cheat the receiver, however, the adversary cannot masquerade other users to cheat a sender when the sender's long-term private key is compromised. In our presented scheme, the adversary can masquerade a sender with identity ID_s and generate a valid DAE authenticator δ as follows when the private key of a receiver S_R is compromised.

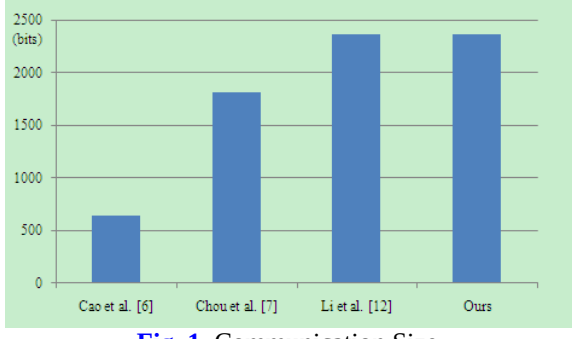


Fig. 1. Communication Size

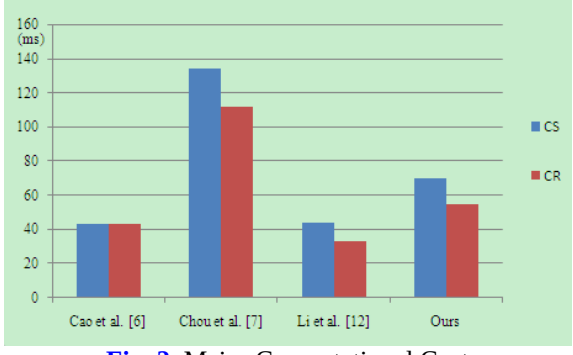


Fig. 2. Major Computational Cost

(1) Choose a random number $r \in \mathbb{Z}_q^*$ and compute $U=rQ_S$.

(2) $C = H_1(\hat{e}(U, S_R)) \oplus (m \| ID_S)$.

(3) $h = H_2(m \| ID_S \| ID_R, U)$.

(4) $V = \hat{e}(U + hQ_S, S_R)$.

Thus the adversary can cheat the receiver while the receiver cannot detect this attack. However, even if the private key of the sender S_S is compromised, our scheme is also secure against W-KCI attack for the following two reasons: (1) When the adversary received a DAE authenticator δ from the sender, he/she cannot recover the plaintext m from δ for he/she doesn't know any information about S_R. Besides, he/she knows nothing about r from $U = rQ_S$ under the assumption of DLP, so it is impossible for him/her to obtain the value $T = \hat{e}(S_S, Q_R)^r$ from U which means the adversary cannot recover m. (2) As the adversary knows nothing about the recovered plaintext m, he/she has no ability to compute $h = H_2(m \| ID_S \| ID_R, U)$ and $V = \hat{e}(U + hQ_S, S_R)$ even if he/she possesses the long-term private key S_S. After our analysis, we find that Li et al.'s scheme is secure against W-KCI attack, but Cao et al.'s scheme is insecurity against W-KCI and the interactive scheme of Chou et al. [7] is insecurity against KCI either.

We implement the four schemes using Type A pairing in PBC library [18] and obtain the precise computational cost that described in **Fig. 2**. Type A pairing is constructed on the curve $y^2 \equiv x^3 + x \pmod{p}$ for some prime $p \equiv 3 \bmod 4$. The embedding degree is 2. G_1 is a group of points on the curve over a base field $\mathbb{F}_p$ and G_2 is a subgroup of finite field $\mathbb{F}_{p^2}$. The group order q is some prime factor of $(p+1)$ and q is 160 bits long.

From **Fig. 2** we know that: CS is 43.174ms and CR is 43.369ms in [6]; CS is 134.33ms and CR is 112.288ms in [7]; CS is 43.937ms and CR is 33.021ms in [12], CS is 69.957ms and CR is 55.026ms in our scheme. The implement environment is: Pentium(R) Dual E5500 @2.80GHz of processor, memory of 2GB and OS with Microsoft Windows XP. Generally speaking, our scheme achieves all properties list in **Table 1** with an admissible computational cost and communication size.

6. Conclusion

In this paper, we present a non-interactive IBDAE scheme from bilinear pairings. We defined a formal security model of the IBDAE scheme and gave the security proof under the BDH assumption in the random oracle model. Our scheme can not only achieve the property of deniable authentication but also obtain the property of confidentiality. So it is suitable for some special application scenarios that need the requirement of message confidentiality while deniable authentication.

References

[1] X. Deng, C. H. Lee and H. Zhu. "Deniable authentication protocols," *IEEE Proceedings-Computers and Digital Techniques*, vol. 148, no. 2, pp. 101-104, 2001. Article (CrossRef Link)

[2] C. Dwork, M. Naor and A. Sahai. "Concurrent zero-knowledge," in *Proc. of the thirtieth annual ACM symposium on Theory of computing*, pp. 409-418, 1998. Article (CrossRef Link)

[3] M. H. Ibrahim. "Receiver-deniable public-key encryption," *International Journal of Network Security*, vol. 8, no. 2, pp. 159-165, 2009. Article (CrossRef Link)

[4] Y. Aumann and M. O. Rabin. "Efficient Deniable Authentication of Long Messages Int," in *Proc. of Conf. on Theoretical Computer Science in honour of Professor Manuel Blums 60th birthday*, pp. 20-24, 1998. Article (CrossRef Link)

[5] R. W. Zhu, D. S. Wong and C. H. Lee. "Cryptanalysis of a suite of deniable authentication protocols," *IEEE Communications Letters*, vol. 10, no. 6, pp. 504-506, 2006. Article (CrossRef Link)

[6] T. Cao, D. Lin and R. Xue. "An efficient ID-based deniable authentication protocol from pairings," in *Proc. of 19th International Conference on Advanced Information Networking and Applications*, pp. 388-391, 2005. Article (CrossRef Link)

[7] J. S. Chou, Y. Chen and J. C. Huang. "An ID-Based deniable authentication protocol on pairings," *IACR Cryptology ePrint Archive*, 2006. Article (CrossRef Link)

[8] M. H. Lim, S. Lee and Y. Park et al. "An enhanced ID-based deniable authentication protocol on pairings," in *Proc. of International Conference on Computational Science and Its Applications*, pp. 1008-1017, 2007. Article (CrossRef Link)

[9] M. H. Lim, S. Lee and H. Lee. "Cryptanalysis on improved Chou et al.'s ID-Based deniable authentication protocol," *IEEE International Conference on Information Science and Security*, pp. 87-93, 2008. Article (CrossRef Link)

[10] H. Tian, X. Chen and Y. Ding. "Analysis of two types deniable authentication protocols," *International Journal of Network Security*, vol. 9, no. 3, pp. 242-246, 2009. Article (CrossRef Link)

[11] Y. Shi and J. Li. "Identity-based deniable authentication protocol," *Electronics Letters*, vol. 41, no. 5, pp. 241-242, 2005. Article (CrossRef Link)

[12] F. Li, P. Xiong and C. Jin. "Identity-based deniable authentication for ad hoc networks," *Computing*, vol. 96, no. 9, pp. 843-853, 2014. Article (CrossRef Link)

[13] A. Shamir. "Identity-based cryptosystems and signature schemes," in *Proc. of CRYPTO 84 on Advances in cryptology*, pp. 47-53, 1985. Article (CrossRef Link)

[14] D. Boneh and M. Franklin. "Identity-based encryption from the Weil pairing," *21st Annual International Cryptology Conference on Advances in Cryptology—CRYPTO 2001*. pp. 213-229, 2001. Article (CrossRef Link)

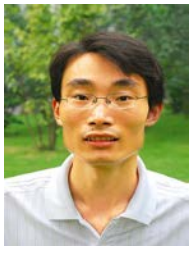
[15] D. Pointcheval and J. Stern. "Security arguments for digital signatures and blind signatures," *Journal of cryptology*, vol. 13, no. 3, pp. 361-396, 2000. Article (CrossRef Link)

[16] J. C. Choon and J. H. Cheon. "An identity-based signature from gap Diffie-Hellman groups," in *Proc. of 6th International Workshop on Practice and Theory in Public Key Cryptography*, pp. 18-30, 2002. Article (CrossRef Link)

[17] C. Fan, S. Zhou and F. Li. "An identity-based restricted deniable authentication protocol," *IEEE International Symposium on Parallel and Distributed Processing with Applications*, pp. 474-478, 2009. Article (CrossRef Link)

[18] http://crypto.stanford.edu/pbc/.

Weifeng Wu received his B.S. degree in College of Science from Henan Agricultural University, Zhengzhou, P.R. China, in 2008. Now, he is studying in School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, P.R. China, for M.S. degree. His current research interests include computer science and cryptography.



Fagen Li received his B.S. degree from Luoyang Institute of Technology, Luoyang, P.R. China, in 2001, M.S. degree from Hebei University of Technology, Tianjin, P.R. China, in 2004, and Ph.D. degree in Cryptography from Xidian University, Xi'an, P.R. China, in 2007. From 2008 to 2009, he was a postdoctoral fellow in Future University-Hakodate, Hokkaido, Japan, which is supported by the Japan Society for the Promotion of Science. He worked as a research fellow in the Institute of Mathematics for Industry, Kyushu University, Fukuoka, Japan from 2010 to 2012. He is now an associate professor in the School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu, P.R. China. His recent research interests include cryptography and network security. He has published more than 70 papers in the international journals and conferences.