Universal Undeniable Signatures

Huafei Zhu

Department of Information Science and Electronics Engineering, Zhejiang University,
Yuquan Campus, Hangzhou, 310027, PR. China
E-mail: zhuhf@zju.edu.cn

Abstract. In this paper, we provide a new approach to study undeniable signatures by translating secure digital signatures to secure undeniable signatures so that the existing algorithms can be used. Our mechanism is that any verifier without trapdoor information cannot distinguish whether a message is encoded from Diffie-Hellamn resource D or random resource R while a signer with trapdoor information can distinguish efficiently a codeword which is computed from D or R. We show how our mechanism can be efficiently achieved and provide proofs of security for our schemes in the standard complexity model. We also provide evidences to show that our approach can be applied to construct designated confirmer signatures, designated verifier signatures as well.

Keywords: Undeniable signatures, universal undeniable signatures, digital signatures

1 Introduction

Undeniable signature schemes, first introduced by Chaum and van Antwerpen [6], have various applications in cryptology. Such signatures are characterized by the property that verification can be only achieved by interacting with the legitimate singer through a confirmation protocol, on the other hand, the signer can prove that a forgery is such by engaging in a denial protocol. Since its introduction in 1989, undeniable signature schemes have received a significant attentions in the cryptographic research community [2], [7], [15], [16], [26], and [29]. These works have provided a variety of different schemes for undeniable signature schemes with variable degree of security, provability and additional features. All those works based on discrete logarithm problem can be viewed as variations of Chaum and van Antwerpen's scheme [6].

In [2], the problem of construction undeniable schemes based on RSA was suggested by Boyar, Chaum, Damgård, and Pedersen as a possible research direction. The first undeniable signature scheme based on the traditional RSA problem was presented by Gennaro, Krawczyk and Rabin [17]. Following Gennaro, Krawczyk and Rabin's works, Galbraith, Mao and Paterson [20], Galbraith and Mao [19] and Miyazaki [23] have already presented improved schemes. Those undeniable signature schemes can be viewed as variations of Gennaro, Krawczyk and Rabin's scheme.

As improvement of undeniable signature [6], [4], several interesting notions are introduced. The notion of convertible signatures was introduced in [2] and [14], and the notion of designated confirmer signatures, first introduced by Chaum [5], and formalized by Okamoto [28], are digital signatures that can be verified only by some help of either a signer or a semi-trusted designated confirmer. In designated confirmer signature schemes, if the signer is unavailable to confirm the signature, the confirmer, previously designated by the signer, can confirm the signature for the recipient. Practical constructions have been proposed by Chaum [5], Okamoto [28], Michels and Stadler [24] and by Camenisch and Michels [10]. Furthermore, Okamoto [28] shows that confirmer signatures exist if and only if public key encryption schemes exist. And finally, the conception of designated verifier signatures was introduced and implemented by Jakobsson, Sako, Impagliazzo [27].

Our contributions In this paper, we provide a new approach to study undeniable signatures by translating secure digital signatures to secure undeniable signatures so that the existing algorithms can be used. Our mechanism is that any verifier without trapdoor information cannot distinguish whether a message is encoded from Diffie-Hellamn resource D or random resource R while a signer with trapdoor information can distinguish efficiently a codeword which is computed from D or R. We show how our mechanism can be efficiently achieved and provide proofs of security for our schemes in the standard complexity model. We also provide evidences to show that our approach can be applied to construct designated confirmer signatures, designated verifier signatures as well.

The rest of paper is organized as follows. In Section 2, a new model on undeniable signature scheme is presented. In Section 3, a generic approach to construct undeniable signatures is presented, and we also provide proofs of security for our schemes. Concrete examples are presented in Section 4 together with possible extensions of our approaches to study variations of undeniable signatures such as designated confirmer signatures, designated verifier signatures. Finally, the conclusion of our research is presented in Section 5.

2 Definition and Notions

Our mechanism to construct undeniable signatures is from source hiding technique. That is a verifier without trapdoor information of source can not distinguish whether a message is encoded from D or R. However, the signer with trapdoor information can distinguish whether it is computed from D or R. In our model, an undeniable signature consists following algorithms and protocols:

- A key generation algorithm KG: This is a probabilistic polynomial time algorithm, which receives 1^k as input, where k is a security parameter, and generates as output a pair of keys (SK_s, PK_s) called secret and public key of signature algorithm, respectively. Also KG outputs a pair of keys (SK_c, PK_c) called secret, public key of confirm and disavowal protocol respectively. The

secret key SK_s is used by the signer to create ordinary signatures, while the keys SK_c is used to test the validity of a codeword.

- Message encoding algorithm Enc: There are two kind of random source denoted by D and R. Given a message m, the signer encodes a message m to a codeword which is either in D or in R. We say a codeword is valid denoted by $Vf(Enc(m), SK_c) = 1$ if it is in D, otherwise it is invalid, denoted by $Vf(Enc(m), SK_c) = 0$.
- A signature algorithm $(Sign, Vf)$: This is a probabilistic polynomial time algorithm, which receives a secret key K_s and a m, it encodes the message m by running Enc, and then runs the signing algorithm with input $Enc(m)$, and finally outputs a signature $\sigma(m) : =(Enc(m), Sign(Enc(m)))$. A signature $\sigma(m)$ of message m is called universally valid if $Vf(Sign(Enc(m)) = 1$.
- A confirmation protocol (C, V_C): This is a pair of interactive Turing machines called the signer and the verifier. The common input consists of a message $m \in M$, a string $z \in S$, and a pair of public keys (PK_s, PK_c). The signer receives as private input a verification key SK_c. The output of the confirmation protocol is 1 or 0, where the output of 1 indicates that the signature is valid, i.e., the confirmation protocol outputs 1 if and only if both $Vf(Enc(m), SK_c) = 1$ and $Vf(Sign(Enc(m))) = 1$.
- The denial protocol (D, V_D): This is a pair of interactive Turing machines called the signer and the verifier. The common input consists of a message $m \in M$, a string $z \in Z$ and a public key (PK_s, PK_c). The signer receives as private input a verification key SK_c. The protocol is designed to convince V_D that z is invalid with w.r.t. m and (PK_s, PK_c). We say a signature is invalid if and only if either $Vf(Enc(m), SK_c)$=0 or $Vf(Sign(Enc(m)))$=0.

There are two basic requirements on undeniable signature schemes: the first is signature unforgeable, namely, without access to the private key of the signer, no one should be able to produce legitimate signatures by himself and the second requirement is non-transferability of the signature, namely, no attacker should be able to convince any other party, without the cooperation of the legitimate signer, of the validity or invalidity of a given message and signature.

To define the security against signature forgeable attack, we allow adversaries to play the following game:

Game 1 (Signature forgery attack)

An adversary $Adv^{Sign(SK_s)}$ is a probabilistic polynomial time algorithm, which can be used in the following type of experiment:

(G1.1) KG is executed on input 1^k, let the output be PK_s, SK_s, PK_c and SK_c. As input, $Adv^{Sign(SK_s)}$ gets PK_s, PK_c, SK_c and 1^k;

(G1.2) $Adv^{Sign(SK_s)}$ may make any polynomial size bound of signature request. $Adv^{Sign(SK_s)}$ produces m and receives the result of running $\sigma(m)$ on input (m, K_s);

(G1.3) Let M be a set of messages occurring in the signature requests done in the second step. Now $Adv^{Sign(SK_s)}$ outputs a message $m' \notin M$, and a string z'.

Let $p_{sig}(k)$ be the probability that $Adv^{Sign(SK_s)}$ outputs (m', z') such that $Vf(z') = 1$. This probability is taken over the random choices make by $Adv^{Sign(SK_s)}$, KG and $Sign$.

We say a function ν: $\mathsf{N} \rightarrow \mathsf{R}$ is a negligible function if for any $c > 0$, there exists $N_0 \in N$ such that $\nu(k) < 1/k^c$ for all $k > N_0$. We say that a function q: $\mathsf{N} \rightarrow \mathsf{R}$ is overwhelming if the function ν defined by $\nu(k) = 1 - q(k)$ is a negligible function.

$Adv^{Sign(SK_s)}$ is successful against the scheme (KG, $Sign$, (C, V_C), (D, V_D) if $p(k)$ is at most negligible amount.

Definition 1: An undeniable signature scheme is said to be unforgeable if no adversary $Adv^{Sign(SK_s)}$ has success against it.

To capture the definition of non-transferability of a signature, we allow adversaries to play the following game:

Game 2 (Codewords indistinguishable)

An adversary $Adv^{D(SK_s,SK_c)}$ is a probabilistic polynomial time algorithm, which can be used in the following type of experiment:

(G2.1) KG is executed on input 1^k, let the output be PK_s, SK_s, PK_c and SK_c. As input, $Adv^{Dist(SK_s,SK_c)}$ gets PK_s, PK_c and 1^k;

(G2.2) $Adv^{Dist(SK_s,SK_c)}$ may now make polynomial size bound of signature requests. Given a message m, the encoding algorithm flops a coin b and then chooses encodes the message according to b, if b, then the codeword is computed form D, otherwise it computed from R. Finally produces a signature of the codeword of m.

(G2.3) Let M be the set of messages occurring in signature requests done in step 2. Now, $Adv^{Dist(SK_s,SK_c)}$ outputs a message $m' \notin M$, and receives a string z', which is either valid codeword, or invalid codeword according to the coin flopping $b \in \{0, 1\}$.

(G2.4) $Adv^{Dist(SK_s,SK_c)}$ may now make polynomial size bound of signature requests and the adversary may also request to play role of V_C or V_D in the confirmation and the denial protocol, provided that m' does not occur as the message in any request, and $Enc(m')$ does not occur in any encoding request.

(G2.5) Finally, $Adv^{Dist(SK_c)}$ outputs b'.

Let p_k be the probability that $Adv^{Dist(SK_c)}$ outputs $b' = b$ in the game. The probability is taken over the random choices made by KG, $Sign$, (C, V_C), (D, V_D) and coin flopping b. $Adv^{Dist(SK_c)}$ is successful against the scheme defined by the tuple (KG, $Sign$, (C, V_C), (D, V_D), if $|p_k - 1/2|$ is at most negligible amount.

Definition 2: An undeniable signature scheme is said to be signatures indistinguishable if no adversary $Adv^{Dist(SK_s,SK_c)}$ has success against it.

3 Generic constructions

Given a signature algorithm $\sum$, which is secure in the sense of Goldwasser, Micali and Rivest's definition [22], we provide a generic approach to translate ordinary signatures to undeniable signatures Since our approach to construct undeniable signatures is independent on the structure of underlying signature scheme, therefore the method is universal. Thus undeniable signatures constructed form this approach are called universal undeniable signatures.

3.1 How to generate codewords

The indistinguishability of codewords follows from the hardness assumption of decisional Diffie-Hellman problem. Let G be a large cyclic group of prime order q, and let g be a generator of G. We consider the following two distributions:

- Given a Diffie-Hellman quadruple g, g^x, g^y and g^{xy}, where $x, y \in Z_q$, are random strings chosen uniformly at random;
- Given a random quadruple g, g^x, g^y and g^r, where $x, y, r \in Z_q$, are random strings chosen uniformly at random.

An algorithm that solves the Decisional Diffie-Hellman problem is a statistical test that can efficiently distinguish these two distributions. Decisional Diffie-Hellman assumption means that there is no such a polynomial statistical test. This assumption is believed to be true for many cyclic groups, such as the prime sub-group of the multiplicative group of finite fields.

Generalized Decisional Diffie-Hellman assumption: for any k, we consider the following distributions:

- The distribution R of any random tuple $(g_1, \cdots, g_k, u_1, \cdots, u_k) \in G^{2k}$, where $g_1, \cdots, g_k$, and $u_1, \cdots, u_k$ are uniformly distributed in G^{2k};
- The distribution D of tuples $(g_1, \cdots, g_k, u_1, \cdots, u_k) \in G^{2k}$, where $g_1, \cdots, g_k$ are uniformly distributed in G^k, and $u_1 = g_1^r, \cdots, u_k = g_k^r$ for random $r \in Z_q$ chosen at random.

An algorithm that solves the generalized decisional Diffie-Hellman problem is a statistical test that can efficiently distinguish these two distributions. Generalized decisional Diffie-Hellman assumption means that there is no such a polynomial statistical test.

It has been shown in the literature that the quadruple decisional Diffie-Hellman problem is equivalent to the polynomial tuple decisional Diffie-Hellman problem. We refer the readers to [1], [8], [9], [31], [34] and [35] for further reference.

Encoding algorithm *Enc*: Let $P = 2Q + 1$ be a large safe prime. A group $G \subseteq Z_P^*$ of order Q is generated by public system parameter. We assume that the discrete logarithm problem defined over Z_P is hard. Let h be a generator of the group G. The singer chooses $x_1, x_2 \in Z_Q$ uniformly at random, and computes $h_1 = h^{x_1} \bmod P$ and $h_2 = h^{x_2} \bmod P$. The public key PK_c is $(h, h_1, h_2, P, Q, G, H)$, where H is a collision free hash function with suitable output length. The secret key SK_c is (x_1, x_2).

Given a message $m \in \{0,1\}^*$, the encoding algorithm computes a codeword as follows:

-**Valid codeword**: It chooses $r \in Z_Q^*$ uniformly at random, and then computes $u = h_1^r \bmod P$, $v = h_2^r \bmod P$, and $w = H(u, v, m)$. The codeword is defined by $Enc(m) = (u, v, w)$. A codeword is valid if (h_1, h_2, u, v) is Diffie-Hellman quadruple. The set of valid codewords is denoted by D.

-**Invalid codeword**: It chooses $r, r' \in Z_Q^*$ uniformly at random, and then computes $u = h_1^r \bmod P$, $v = h_2^{r'} \bmod P$, and $w = H(u, v, m)$. The codeword is defined by $Enc(m) = (u, v, w)$. A codeword is invalid if (h_1, h_2, u, v) is random quadruple. The set of invalid codewords is denoted by R.

This completes the description of encoding algorithm *Enc*.

3.2 How to universally translate signatures to undeniable signatures

Given signature schemes secure against adaptive chosen message attack in the sense of Goldwasser, Micali and Rivest's definition [22], we provide a method universally translating the ordinary signatures to undeniable signatures.

- Signing algorithm: on input m, the signer runs encoding algorithm *Enc* to get a codeword $Enc(m) := (u, v, w)$, and signs the codeword $Enc(m)$. The output of signature algorithm is $\sigma(m) := (Enc(m), Sign(w))$
- Verification algorithm: on input a putative signature $\sigma(m)$ of message m, the verification algorithm Vf tests the valid of $Sign(w)$.
  - if $Vf(Sign(w))$=0, then the verifier terminates the protocol, and outputs 0;
  - If $Vf(Sign(w))$=1, then the verifier further decides whether $w \in D$ or $w \in R$ by running interactive bi-proof of equality or inequality of two discrete logarithms with the signer (e.g., the protocols presented in [4], [16], [11] and [25]).
    - If $\log_{h_1}(u) = \log_{h_2}(v)$, then $(h_1, h_2, u, v) \in D$ thus $Enc(m)$ is a valid codeword, and the verification algorithm output 1;
    - Otherwise $Enc(m)$ is an invalid codeword, and the verification algorithm output 0.

The proof of security We want to show that universal undeniable signature scheme is secure for the signer and it is also codeword indistinguishable. Before we provide rigorous security proof of scheme, we first review the following well known bi-proof system of equality or inequality two discrete logarithms presented by Michael and Stadler [25] as an improvement of original protocols first studied by Chaum [4] and later extended by Fujioka, Okamoto and Ohta [16].

Suppose the prover knows the discrete logarithm $y = \alpha^x \bmod P$ (P is a large safe prime, i,e., P=2Q+1), and wants to allow the verifier to decide whether $\log_\beta(z) = \log_\alpha(y) \bmod P$ for given elements β, z, the prover and the verifier can execute following protocol.

- The verifier chooses random values $u, v \in Z_Q^*$, computes $a := \alpha^u y^v$ and send a to the prover.
- The prover chooses $k, \tilde{k}, w \in Z_Q^*$, computes $r_\alpha := \alpha^k$, $r_\beta := \beta^k$, $r\tilde{}_\alpha := \alpha^{\tilde{k}}$, $r\tilde{}_\beta := \beta^{\tilde{k}}$, and sends r_α, r_β, $r\tilde{}_\alpha$, $r\tilde{}_\beta$ and w to the verifier.
- The verifier opens its commitment a by sending u, v to the prover;
- If $a \neq \alpha^u y^v$, the prover halts, otherwise the prover computes $s := k-(v+w)x \bmod Q$, $\tilde{s} := \tilde{k} - (v+w)k \bmod Q$, and sends $s, \tilde{s}$ to the verifier;
- The verifier first checks whether $\alpha^s y^{v+w} = r_\alpha$, $\alpha^{\tilde{s}} y^{v+w} = \tilde{r}_\alpha$, and $\beta^{\tilde{s}} r_\beta^{v+w} = \tilde{r}_\beta$ and then concludes:
  - If $\beta^s z^{v+w} = r_\beta$, then $\log_\beta(z) = \log_\alpha(y)$;
  - If $\beta^s z^{v+w} \neq r_\beta$, then $\log_\beta(z) \neq \log_\alpha(y)$;

Michael and Stadler [25] prove that the above protocol is complete and sound. It is zero-knowledge under the assumption that there exists no algorithm running in expected polynomial time which decides with non-negligible probability better than guessing whether two discrete logarithms are equal.

Theorem 1: If underlying signature algorithm $\sum$ is secure against in the sense of Goldwasser, Micali and Rivest's definition [22], then the constructed universal undeniable signature scheme $\prod$ is secure in the sense of definition 1.

Proof: Since the adversary is given SK_c in forgery game 1, we can ignore the algorithm Enc. Therefore the signature algorithm $\prod$ is the same as the signature algorithm $\sum$, and thus theorem 1 is correct.

Theorem 2: Under the hardness assumption of decisional Diffie-Hellman problem over Z_P^*, the universal undeniable signature scheme is codeword indistinguishable.

Proof: Given a quadruple (h_1, h_2, u, v) which is either from D or R, we want to distinguish whether $(h_1, h_2, u, v) \in D$ or in R with the help of adversary Adv, who is assumed to be able to distinguish a codeword in D or in R with non-negligible advantage.

We construct a simulator as follows. The input to simulator is (h_1, h_2, u, v) and a secure signature scheme $\sum$ as well as secret signing key SK_s, PK_s and PK_c, where PK_c is $(h, h_1, h_2, P, Q, G, H)$, where H is a collision free hash function with suitable output length. For $1 \leq i \leq poly(k)$, on response the ith signing query for a message m_i from the adversary, the simulator chooses a bit b_i uniformly at random. If $b_i = 1$, then the simulator chooses $r_i \in Z_Q^*$ uniformly at random and finally computes $u_i = h_1^{r_i}$, $v_i = h_2^{r_i}$ and $w_i = H(u_i, v_i, m_i)$; Otherwise, the simulator chooses $r_{i,1}, r_{i,2} \in Z_Q^*$ uniformly at random and computes $u_i = h_1^{r_{i,1}}$, $v_i = h_2^{r_{i,2}}$, $w_i := H(u_i, v_i, m_i)$, where H is collision free hash function. The coin tosses r_i or $(r_{i,1}, r_{i,2})$ and b_i are kept secret. Finally the simulator computes the signature $Sign(w_i)$; The output of signature algorithm is $Enc(m_i)$=(u_i, v_i, w_i) and $Sign(w_i)$.

Since $Sign(w_i)$ is always valid, i.e., $Vf(Sign(w_i)) = 1$, it follows that the signature is generated by the simulator itself, therefore the simulator knows the exact value r_i or $(r_{i,1}, r_{i,2})$ as well as b_i (otherwise this signature is a forgery signature of m_i, thus contracts the assumption of $\sum$ is secure against adaptive chosen message attack). To request the confirmation and deniable of a codeword $Enc(m_i)$, the simulator can run the bi-proof for equality or inequality of two logarithm $\log_{h_1}(u_i)$ and $\log_{h_2}(v_i)$ described above as the simulator knows the random strings r_i or $(r_{i,1}, r_{i,2})$.

We now translate the quadruple (h_1, h_2, u, v) to a signature of message m as follows: we compute $w = H(u, v, m)$ and $Sign(w)$. The codeword of message m is $Enc(m) = (u, v, w)$, the signature of message m is $\sigma := (Enc(m), Sign(w))$. Eventually, the adversary will output a bit b', at which point, a distinguisher $Dist$ to tell Diffie-Hellman quadruple from random quadruple can be easily constructed: the input to $Dist$ is (h_1, h_2, u, v), the output of $Dist$ is $b' \in \{0, 1\}$, a copy of adversary's output. Since the adversary is assumed to be able to guess the correct value b' with non-negligible probability, so is $Dist$.

4 Concrete examples and possible extensions

4.1 Concrete examples

We provide the following example to show that efficiency of our construction from an efficient signature scheme presented in [36]. Without any modification, this technique can be applied to other secure signature schemes such as OAEP RSA [3], and Schnorr signature scheme [30] as well.

- Key generation algorithm: Let p, q be two large primes such that $p-1 = 2p'$ and $q-1 = 2q'$, where p', q' are two $(l'+1)$-bit strings. Let $n = pq$ and QR_n be the quadratic residue of Z_n^*. Let g, h be two generators of QR_n chosen uniformly at random. The public key is (n, g, h, X, H), where $X \in QR_n$ and H is a collision free hash function with output length l. The private key is (p, q).
- Signature algorithm: To sign a message m, a $(l+1)$-bit prime e and a string $t \in \{0,1\}^l$ are chosen at random. The equation $y^e = Xg^t h^{H(m)} \mathrm{mod} n$ is solved for y. The corresponding signature of the message m is (e, t, y).
- Verification algorithm: Given a putative triple (e, t, y), the verifier first checks that e is an odd $(l+1)$-bit number. Second it checks the validation that $X = y^e g^{-t} h^{-H(m)} \mathrm{mod} n$. If the equation is valid, then the verifier accepts, otherwise, it rejects.

This signature scheme is proved secure under the joint assumptions of the strong RSA problem, the discrete logarithm problem defined over QR_n, as well as the existence of collision free hash function.

Concrete examples We now provide a concrete example based on the above signature scheme.

– Key generation algorithm: Let $P = 2Q + 1$ be a large safe prime. A group $G \subseteq Z_P^*$ of order Q is generated by public system parameter. We assume that the discrete logarithm problem defined over Z_P is hard. Let h be a generator of the group G. The singer chooses $x_1, x_2 \in Z_Q$ uniformly at random, and computes $h_1 = h^{x_1} \bmod P$ and $h_2 = h^{x_2} \bmod P$. The public key PK_c is $(h, h_1, h_2, P, Q, G, H)$, where H is a collision free hash function with suitable output length. The secret key SK_c is (x_1, x_2).
  Let p, q be two large safe primes i.e., $p-1 = 2p'$ and $q-1 = 2q'$, where p', q' are two l'-bit strings. Let $n = pq$ and QR_n be the quadratic residue of Z_n^*. Let X, g_1 and g_2 are generators of QR_n chosen uniformly at random. The public key of signer is (n, X, g_1, g_2, H) along with an appropriate description of G including s. The private key of signer is (p, q).
– Signing algorithm: The signer chooses $r \in Z_Q$ uniformly at random and computes $u = h_1^r \bmod P$, $v = h_2^r \bmod P$, and $w = H(u, v, m)$. Then the signer chooses $l+1$-bit prime e at random, the equation $y^e = Xg_1^t g_2^w \bmod n$ is solved for y. The signature of message m is denoted by $\sigma(m) = (e, y, t, u, v)$.
– The confirm protocol between the singer and verifier $Conf_{S,V}$. Given a putative signature of message $\sigma(m)=(e, y, t, u, v)$, the verifier first checks the equation $X = y^e g_1^{-t} g_2^{-w} \bmod n$, where $w = H(u, v, m)$. If $Vf(Sign(w)) = 0$, then the verifier terminates the protocol and output 0. Otherwise, the signer proves the equality of two discrete logarithms $\log_{h_1}(u) = \log_{h_2}(v)$ to the verifier with the auxiliary input r, where r is a random string which is used for generating the signature. $\sigma(m)$ is valid if and only if the verifier accepts the proof of equality of discrete logarithm $\log_{h_1}(u) = \log_{h_2}(v)$.
– The disavowal protocol between the signer and a verifier $Dis_{S,V}$. Given a putative signature of message $\sigma(m)=(e, y, t, u, v)$, the verifier first checks the equation $X = y^e g_1^{-t} g_2^{-w} \bmod n$, where $w = H(u, v, m)$. If $Vf(Sign(w)) = 0$, the verifier terminates the protocol, and outputs 0. Otherwise, the signer proves the inequality of two discrete logarithms $\log_{h_1}(u) \neq \log_{h_2}(v)$ to the verifier with the auxiliary input r. Notice that a signer always knows the random string r to be used to generate $\sigma(m)$ as the output of $Vf(Sign(w))$ is always 1.

By applying theorem 2, one knows that under joint assumptions of the strong RSA problem defined over Z_n^*, the discrete logarithm problem defined over Z_P^*, as well as the existence of collision free hash function, the undeniable signature scheme is secure.

4.2 Possible extensions

We consider possible extensions of our approach to construct convertible signatures, designated verifier signatures and designated confirmers signatures in this section.

– Convertible signature schemes can be easily constructed if the trapdoor information $x_1, x_2 \in Z_Q^*$ is revealed.

– Designated confirmer signature scheme can be easily constructed if the trapdoor information $x_1, x_2 \in Z_Q^*$ is chosen by a confirmer.
– Designated verifier signature scheme can be easily constructed if the trapdoor information $x_1, x_2 \in Z_Q^*$ is chosen by a verifier. Here we point out the difference between our approach and Steinfeld et.al's approach [32] and [33]. Our construction is pre-processing model, i.e., given a message, we first encode the message and sign the codeword. Steinfeld et.al's approach is post-precessing, i.e., after singer produces a signature of message m, the singer further translates the signature to designated-verifier signatures. Therefore two approaches are completely different.
– Since the quadruple decisional Diffie-Hellman problem is equivalent to any polynomial tuples of decisional Diffie-Hellman problem, we can construct a set of designated verifiers S (each player in S is a designated verifier) by the following approach: given a public known multiple group G with order Q. Let h be a publicly verifiable generator of G, each designated verifier can choose its secret key $x_{i,1}, x_{i,2}$ and then publishes its public key $h_{i,1} = h^{x_{i,1}}$ and $h_{i,2} = h^{x_{i,2}}$. We say a codeword is valid for the verifier i, if $(h_{i,1}, h_{i,2}, u_{i,2}, u_{i,2})$ is a Diffie-Hellman quadruple. Notice that Diffie-Hellman quadruple can be easily distinguished from random quadruple with the help of trapdoor information $(x_{i,1}, x_{i,2})$ which is generated and known only by the verifier i. Therefore, we provide a solution to the open problem suggested by Desmedt [13].

5 Conclusion

In this paper, we provide a universal approach to construct undeniable signatures from any secure signature scheme. We also provide evidences to show that our approach can be applied to construct designated confirmer signatures, designated verifier signatures as well.

References

1. Eli Biham, Dan Boneh, and Omer Reingold. Breaking generalized Diffie Hellman modulo a composite is no easier than factoring. Information Processing Letters,70:83–87, 1999.
2. J. Boyar, D. Chaum, I. Damgård, T P. Pedersen: Convertible Undeniable Signatures. CRYPTO 1990: 189-205
3. Mihir Bellare, Phillip Rogaway: Optimal Asymmetric Encryption. EUROCRYPT 1994: 92-111
4. David Chaum: Zero-Knowledge Undeniable Signatures. EUROCRYPT 1990: 458-464.
5. David Chaum: Designated Confirmer Signatures. EUROCRYPT 1994: 86-91
6. David Chaum, Hans Van Antwerpen: Undeniable Signatures. CRYPTO 1989: 212-216.
7. David Chaum, Eugene van Heijst, Birgit Pfitzmann: Cryptographically Strong Undeniable Signatures, Unconditionally Secure for the Signer. CRYPTO 1991: 470-484.

8. Christian Cachin, Klaus Kursawe, and Victor Shoup. Random oracles in Constantinople: Practical asynchronous Byzantine agreement using cryptography. In Proceedings of the 19th Annual ACM Symposium on Principles of Distributed Computing, Portland, Oregon, July 2000.
9. Jan Camenisch, Ueli Maurer, and Markus Stadler. Digital payment systems with passive anonymity evoking trustees. In E. Bertino, H. Kurth, G. Martella, and E. Montolivo, editors, Proceedings of the Fourth European Symposium on Research in Computer Security (ESORICS), number 1146 in Lecture Notes in Computer Science, pages 33-43, Rome, Italy, September 1996. Springer Verlag, Berlin Germany.
10. J. Camenisch, M.Michels: Confirmer Signature Schemes Secure against Adaptive Adversaries. EUROCRYPT 2000: 243-258.
11. David Chaum, Torben P. Pedersen: Wallet Databases with Observers. CRYPTO 1992: 89-105.
12. R. Cramer and V. Shoup. Signature scheme based on the Strong RAS assumption. 6th ACM Conference on Computer and Communication Security, Singapore, ACM Press, November 1999.
13. Y.Desmedt. Lecture notes on verifier designated signatures. http:// www.cs.fsu.edu/ desmedt/ lectures/ verifier-designated-signatures.pdf.
14. Damgård, T P. Pedersen: New Convertible Undeniable Signature Schemes. EUROCRYPT 1996: 372-386.
15. Y. Desmedt, M. Yung: Weakness of Undeniable Signature Schemes (Extended Abstract). EUROCRYPT 1991: 205-220
16. A. Fujioka, T. Okamoto, K. Ohta: Interactive Bi-Proof Systems and Undeniable Signature Schemes. EUROCRYPT 1991: 243-256.
17. Rosario Gennaro, Hugo Krawczyk, Tal Rabin: RSA-Based Undeniable Signatures. CRYPTO 1997: 132-149.
18. Rosario Gennaro, Tal Rabin, Hugo Krawczyk: RSA-Based Undeniable Signatures. Journal of Cryptology 13(4): 397-416, 2000.
19. Steven D. Galbraith, Wenbo Mao: Invisibility and Anonymity of Undeniable and Confirmer Signatures. CT-RSA 2003: 80-97 2002.
20. Steven D. Galbraith, Wenbo Mao, Kenneth G. Paterson: RSA-Based Undeniable Signatures for General Moduli. CT-RSA 2002: 200-217.
21. L. Guillou, J. Quisquater. A practical zero-knowledge protocol fitted to security microprocessors minimizing both transmission and memory. *Eurocrypto'88*, 123-128, 1988.
22. S. Goldwasser, S. Micali, R. Rivest: A Digital Signature Scheme Secure Against Adaptive Chosen-Message Attacks. *SIAM J. Comput.* 17(2): 281-308, 1988.
23. T. Miyazaki: An Improved Scheme of the Gennaro-Krawczyk-Rabin Undeniable Signature System Based on RSA. ICISC 2000: 135-149.
24. M.Michels, M.Stadler: Generic Constructions for Secure and Efficient Confirmer Signature Schemes. EUROCRYPT 1998: 406-421.
25. M.Michels, M.Stadler: Efficient convertible undeniable signature schemes, 4th International workshop on selected areas in cryptology, SAC'97, 1997, 231-244.
26. M. Jakobsson: Blackmailing using Undeniable Signatures. EUROCRYPT 1994: 425-427
27. M. Jakobsson, K. Sako, R. Impagliazzo: Designated Verifier Proofs and Their Applications. EUROCRYPT 1996: 143-154.
28. T.Okamoto. Designated Confirmer Signatures and Public-Key Encryption are Equivalent. CRYPTO 1994: 61-74

29. T. Pedersen: Distributed Provers with Applications to Undeniable Signatures. EUROCRYPT 1991: 221-242.
30. C. Schnorr. Efficient identification and signature for smart card. Cryptology-Crypto'89, 235-251.
31. Ahmad-Reza Sadeghi, Michael Steiner: Assumptions Related to Discrete Logarithms: Why Subtleties Make a Real Difference; Eurocrypt 2001, LNCS 2045, Springer-Verlag, May 2001, 243-260.
32. Ron Steinfeld and Laurence Bull and Huaxiong Wang and Josef Pieprzyk, Universal Designated-Verifier Signatures, Asiacrypt'2003.
33. Ron Steinfeld and Huaxiong Wang and Josef Pieprzyk Efficient Extension of Standard Schnorr/RSA signatures into Universal Designated-Verifier Signatures. PKC2004, Singapore.
34. Michael Steiner, Gene Tsudik, and Michael Waidner. Key agreement in dynamic peer groups. IEEE Transactions on Parallel and Distributed Systems, 11(8):769-780, August 2000.
35. Stefan Wolf. Information theoretically and Computationally Secure Key Agreement in Cryptography. PhD thesis, ETH Zurich, 1999.
36. H. Zhu. Constructing Committed Signatures From Strong-RSA Assumption In The Standard Complexity Model, PKC2004, Singapore.