A New Designated Confirmer Signature Variant with Intended Recipient

Yong Li[*] Dingyi Pei
State Key Laboratory of Information Security
(Graduate School of Chinese Academy of Sciences)
BeiJing, 100039, P.R.China
http://www.is.ac.cn

November 4, 2004

Abstract

Previous designated confirmer signature schemes were less efficient because complex zero-knowledge proof employed in confirmation and disavowal protocol. In this paper, we propose a new efficient signature scheme which is recipient-specific and confirmer-specific. The new scheme is transformed from ID-based chameleon signature and inherits its advantage in simplicity and efficiency. The scheme's security relies on the underlying secure chameleon signature and public key encryption scheme. We also considers the case of confirmer as an adversary in security proof.

Keywords: Identity-based signature, Chameleon hashing, Designated confirmer signature

1 Introduction

Traditional digital signatures provides non-repudiation property required in case of possible disputes by *universal verifiability*, which can be verified by everyone. However, such universal verifiability is undesirable in some situations. For example, disclosing a sealed bid to a competitor can benefit one party but jeopardize the interests of the other in auction system. In other words, the abuse of disclosing bid is prevented in e-auction system. Meanwhile, when there is disputation between dealer and bidder, a trusted third party generally needed to give an arbitrament.

Let's consider another scenario that a person make will to declare how his possessions to be disposed of after death. Here, only the intended recipient

[*] Email: liy_bs@mails.gscas.ac.cn

(the inheritor) can be convinced about the validity of the signed document. At the same time, a specified third party (a lawyer) can convince the inheritor of the validity of signed will. This issue has some new characteristic to be discussed later and we prefer to call it “legal will problem”.

These examples show that sometimes the signature or commitment are to some extent commercially or personally sensitive and universal verifiability is not an expected property here. This is the conflict between authenticity(non-repudiation) and privacy(controlled verifiability) in the digital signatures.

For solving this problem, *undeniable signature* was firstly introduced by Chaum and van Antwerpen in 1989 [6]. Verification of such signature requires the collaboration of the signer, so that the signer can control to *whom* the signed document is being disclosed. Subsequently, plenty of undeniable signature schemes were proposed [2, 9, 12, 16, 19].

Another line of work that resolves the conflict between authenticity and privacy is *designated verifier signature* [14], introduced by M. Jakobsson *et al.* at EuroCrypto ’96. Designated verifier signature guarantees that only the specified verifier(recipient) can be convinced by the designated verifier proof. Furthermore, the designated verifier cannot transfer the conviction to others because he is able to forge a signature intended to himself that is indistinguishable from real signature. This non-transferability property is employed to construct a new scheme in the paper.

The main drawback of undeniable signature is that if the signer become unavailable, then the recipient cannot make use of the signature. The concept of *designated confirmer signature*(DCS) was introduced by Chaum [7] to solve this weakness. It involves three parties: the signer, recipient and the confirmer. If the signer is unavailable to confirm the signature σ, the confirmer, previous designated by the signer, can confirm σ for the recipient. The recipient of σ cannot convince anyone else of the validity of σ. “The technique works in essence by allowing the signer to prove to the signature’s recipient that designated parties can confirm the signature without the signer.” [7]. Along this line of work, a number of attempts have been made to design efficient and secure DCS schemes [8, 10, 11, 17, 18, 21].

Most of previous work on DCS use *zero-knowledge proof* in the confirmation protocol [8, 18]. In order for the verifier to be convinced of the validity of the DCS, the confirmer and verifier interact in a zero-knowledge proof in which confirmer proves to the verifier that what he got is indeed a valid confirmer signature, while the verifier is unable to transfer the convince to other party. However, complex zero-knowledge proof in confirmation protocol is unpractical and more efficient schemes are needed to be proposed.

Chameleon signatures, introduced by Krawczyk and Rabin [15] , are based on well established hash-and-sign paradigm, where a chameleon hash function is used to compute the cryptographic message digest. One conspicuous property of chameleon signatures is it simultaneously provide non-repudiation and non-transferability for the signed message as undeniable signatures do, but the former allows for simpler and more efficient realization than the latter. Unlike traditional undeniable signatures and confirmer signatures, chameleon signatures are non-interactive and do not involve the design and complexity of zero-knowledge proofs. We give a brief description of chameleon hashing and chameleon signature in section 2.

Recall the "legal will problem" mentioned before, the pure designated verifier signature or designated confirmer signature can not satisfied its requirement because not only the recipient is designated by signer, but also the confirmer. To the best of our knowledge, there seems no corresponding signature scheme has been proposed to deal with such situation. We attempt to give our solution and our goal is to combine the advantage of chameleon signature with DCS scheme for constructing a new variant of DCS scheme without loss of security.

1.1 Related Work

Designated confirmer signature was introduced by Chaum [7]. Okamoto presented a formal model and definition of DCS and proved that secure designated confirmer signature are equivalent to secure public-key encryption [18]. However, it is shown that Okamoto's scheme is insecure because the confirmer can forge a signature [17]. Michels and Stadler proposed a solution to Okamoto's problem by introducing a new tool: *confirmer commitments* [17]. As pointed out in [8], these schemes are vulnerable to an *adaptive signature-transformation* attack (which is similar to security against adaptive chosen-ciphertext attacks for encryption schemes). Camenisch and Michels presented a generic construction for confirmer signature schemes that does not suffer from the adaptive signature-transformation attack.

Later, there are more other works discussing on DCS schemes. In [10], S.D. Galbraith and W. Mao give a precise definition of anonymity for confirmer signatures in the multi-user setting and show that anonymity and invisibility are closely related. In another work [21], Wang S.P. *et al.* attempt to construct a secure and efficient DCS scheme using standard DSA and RSA. However, their scheme was found several security flaws in non-transferability, invisibility and non zero knowledge proof by Wang G.L. *et al.* [20]. In most recent work [11], Shafi Goldwasser *et al.* present simple and efficient transformations of a large class of digital signature schemes, including the Cramer-Shoup, Goldwasser-Micali-Rivest and Gennaro-Halevi-Rabin signatures into secure designated confirmer signature schemes. The tool their transformation uses is *witness hiding proofs of knowledge* of a signature for the underlying signature scheme. More details may be refered to [11].

In recent years, the bilinear pairings have been found various applications in cryptography and have allowed us to construct some new cryptographic primitives. More precisely, they are basic tools for construction of ID-based cryptographic schemes. In [13], Han *et al.* proposed an ID-based confirmer signature scheme using pairings, but their scheme was not secure against the denial attack and the forge attack and it is actually an ID-based undeniable signature scheme [22]. In [1], Ateniese and Medeiros introduced the concept of ID-based chameleon hash function. Later, Zhang *et al.* proposed new construction of ID-based chameleon hash from bilinear pairings and chameleon signature based on these new hash function [23].

Our Contributions In this paper, we transform ID-based chameleon signature from bilinear pairing into a new variant of DCS scheme, which not only confirmer is designated, but also the verifier(intended signature recipient). However, the new scheme is different from traditional DCS, so we may call it designated verifier-designated confirmer signature(DV-DCS). The new scheme is efficient and simple while maintaining its security properties simultaneously. The security proof also handles the problem of confirmer as an adversary, which previous schemes didn't consider.

The rest of the paper is organized as follows: Some preliminary notions are given in Section 2. In Section 3 we describe the new signature model. The new proposed DV-DCS scheme is given in Section 4. In section 5, we present security analysis of the new scheme. In Section 6, efficiency analysis is given. Finally, conclusions is made in Section 7.

2 Preliminary Notions

The main difference between regular signature and chameleon signature is in the type of hash function. Chameleon signatures use a chameleon hash function. A chameleon hash function is a trapdoor one-way hash function, which prevents everyone except the holder of the trapdoor information from computing the collisions for a randomly given input. As shown in [15], chameleon signature has the following properties at the same time.

- Non-repudiation: The signer S cannot deny a signature he generated since he cannot find collisions in the hash.

- Non-transferability: The recipient R cannot prove to any third party that a given signature of S corresponds to a certain message since R could "open" the signed message in any way he wants using the trapdoor hashing.

Definition 1 *(ID-based Chameleon hashing [1]). An ID-based chameleon hashing scheme is defined by a family of efficiently computable algorithms:*

- ***Setup:*** A trusted party, called Private Key Generator (PKG), runs this probabilistic algorithm to generate a pair of keys SK and PK defining the scheme. It publishes PK and keeps SK secret.

- ***Extract:*** A deterministic algorithm that, on inputs SK and an identity string ID, outputs the trapdoor information S_{ID} associated to the identity.

- ***Hash:*** A probabilistic algorithm that, on inputs PK, an identity string ID, and a message m, outputs a hash value h.

- ***Forge:*** An algorithm that, on inputs PK, an identity string ID, the trapdoor information S_{ID} associated with ID, a message m', and a hash value h of a message m, outputs a sequence of random bits that correspond to a valid computation of $Hash(PK, ID, m')$ yielding the target value h.

Let $\mathbb{G}_1$ be a cyclic additive group generated by P, whose order is a prime q, and $\mathbb{G}_2$ be a cyclic multiplicative group of the same order q. A bilinear pairing is a map $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ with the following properties:
P1 *Bilinearity*: $e(aP, bQ) = e(P, Q)^{ab}$;
P2 *Non-degeneracy*: There exists $P, Q \in \mathbb{G}_1$ such that $e(P, Q) \neq 1$;
P3 *Computability*: There is an efficient algorithm to compute $e(P, Q)$ for all $P, Q \in \mathbb{G}_1$.

Throughout this paper, we define the system parameters are as follows: Let P be a generator of $\mathbb{G}$ with order q, the bilinear pairing is given by $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$. These system parameter can be obtained using a **GDH Parameter Generator** $\mathcal{IG}$ [4]. Define a cryptographic hash function $H_0 : \{0, 1\}^* \rightarrow \mathbb{G}_1^*$. Denote *params* $= \{\mathbb{G}_1, \mathbb{G}_2, e, q, P, H_0\}$.

3 The New Signature Model

This section provides a formal definition of DV-DCS and it's security requirements. The players in a DV-DCS scheme are signers S, confirmers C, verifiers V.

Definition 2 *A secure DV-DCS scheme consists of the following components:*

Key generation: *Let* $CKGS(1^n) \rightarrow (SK_S, PK_S)$*,* $CKGC(1^n) \rightarrow (SK_C, PK_C)$ *and* $CKGV(1^n) \rightarrow (SK_V, PK_V)$ *be three probabilistic polynomial algorithms. The parameter* n *is a security parameter,* (SK_i, PK_i) *(*$i \in \{S, C, V\}$*) is a secret/public key pair for the signer, confirmer and verifier respectively.*

Signing: *A probabilistic signature generation algorithm* $CSig(m, SK_S, PK_S, PK_C) \rightarrow \sigma$ *that generates a signature* σ *for signing a message* $m \in \{0, 1\}^*$*.*

Confirmation and disavowal: *A signature confirmation protocol* $Conf_{(C,V)}$ *and disavowal protocol* $Disavowal_{(C,V)}$ *between a confirmer and a verifier. The private input of the confirmer is* SK_C *and their common input consists* *of m, σ, PK_S and PK_C. The output of the verifier is either 1 (true) or 0 (false).*

Partially convertibility: *An algorithm $CConv(m, \sigma, PK_S, SK_C, PK_C) \rightarrow s$ that allows a confirmer to turn a DV-DCS signature σ into an designated verifier signature s. If the conversion fails, the algorithm's output is $\perp$.*

Signature verification: *An algorithm $COVer(m, s, PK_S) \rightarrow \{0, 1\}$ that allows the designated verifier V to verify signature s and takes as input a message m, a signature s, and the public key PK_S of the signer.*

In the following, we define the security requirements for the signer and confirmer of DV-DCS scheme. Informally speaking, *security for the signer* guarantees that DV-DCS as well as partially converted signature are unforgeable under an adaptive chosen-message attack. In other words, it assures that no one except the signer can generate a valid DV-DCS signature. *Security for the confirmer* guarantees that the scheme is secure for the confirmer against adaptive chosen-confirmer-signature attacks (this is similar to security against chosen-ciphertext attacks for encryption schemes). This requirement assures that no one apart from the confirmer can distinguish between valid and invalid DV-DCS signatures.

Formally, we say that a DV-DCS scheme is secure if it meets the following requirements:

- **Security for signer:** Let $\mathcal{A}$ be a probabilistic polynomial time (PPT) forging algorithm which, on input strings $1^n, PK_S, PK_C$ (and possibly also the secret key SK_C of the confirmer)[1]. $\mathcal{A}$ is allowed oracle access to the signer S and receives S's signature of polynomially many adaptively messages $\{m_i\}$. Finally, $\mathcal{A}$ halts and outputs a pair of strings (m, s') where $m \neq m_i$ for all i. We require that for all such $\mathcal{A}$, all sufficiently large n and for any message $m \notin \{m_i\}$,

  $$Pr(COVer(m, s', PK_S) = accept) < negl(n)$$

  The probability is taken over the coin tosses of the $S, \mathcal{A}$ and the key generation algorithms $CKGS, CKGC$.

  Remark. We call a function $negl(n) : \mathbb{N} \rightarrow \mathbb{R}$ *negligible* if for all c and for all sufficiently large n, $negl(n) < \frac{1}{n^c}$.

- **Security for the confirmer:** Let $\mathcal{A}$ be a PPT attacking algorithm which, on input strings $1^n, PK_S, PK_C$ can make oracle queries to the signers and confirmer via $CSig$, $Conf_{(C,\mathcal{A})}$, $Disavowal_{(C,\mathcal{A})}$ for polynomially many inputs of his choice and finally, for a pair (m, σ') of his

[1] This security requirement holds also against confirmer and thus we allow even as input the SK_C. Considering the scenario of confirmer acting as an adversary.

choice, $\mathcal{A}$ executes $Conf_{(\mathcal{A},V)}(1^n, m, \sigma', PK_S, PK_C)$. For all such $\mathcal{A}$,

$$Pr(Conf_{(\mathcal{A},V)}(1^n, m, \sigma', PK_S, PK_C) = accept) < negl(n)$$

where the probability is taken over all possible coin tosses of the $S, C, V, \mathcal{A}$ and the key generation algorithms $CKGS, CKGC$.

4 Proposed DV-DCS scheme

Our scheme construction firstly sign a message m using ID-based chameleon signature and then encrypt the signature using the confirmer's public key. The resulting ciphertext would serve as the DV-DCS σ of m. We adopt Zhang *et al.*'s ID-based chameleon signature scheme [23] as our building block.

The scheme includes three parties: signer S, confirmer C, verifier V. $ENC = \{KG, Enc, Dec\}$ is a encryption scheme secure against adaptively chosen-ciphertext attack ($\mathcal{CCA}2$).

1. **Setup:** PKG chooses a random number $s \in Z_q^*$ and sets $P_{pub} = sP$. Define cryptographic hash function $H_1 : \{0,1\}^* \to Z_q^*$, $H_2 : \mathbb{G}_2 \times \mathbb{G}_1 \to Z_q^*$. The system parameters $params = \{\mathbb{G}_1, \mathbb{G}_2, e, q, P, H_0, H_1, H_2\}$.

2. **Key generation:** Let ID_S, ID_C, ID_V are identity of the signer, confirmer and verifier respectively. PKG generates $PK_S = H_0(ID_S)$, $SK_S = sPK_S$; $PK_C = H_0(ID_C)$, $SK_C = sPK_C$, $PK_V = H_0(ID_V)$, $SK_V = sPK_V$, where (PK_S, SK_S), (PK_C, SK_C) and $(PK_V, SK_V)$are signer, confirmer and verifier's public/secret key pair respectively.

3. **Signing:** Signer S chooses a random number $r \in_R Z_q^*$, and a random element $R \in_R \mathbb{G}_1$, computes $X = r \cdot PK_S$ and

$$z = Hash(ID_V, m, R) = e(R, P)e(H_1(m)H_0(ID_V), P_{pub}) \qquad (1)$$

Then, computes $h = H_2(z||X)$, $Y = (h+r)SK_S$ and $Z = Enc_{PK_C}(Y)$. The message-signature pair is $\{m, X, Z, R\}$ and be transferred to the verifier.

4. **Confirm and disavowal Protocol:** Given a signature $\sigma = \{X, Z, R\}$, confirmer first run partially conversion algorithm $CConv$ to get $s = \{X, Y, R\}$, then transfer $\{X, Y, R\}$ to verifier V. V verify the equation

$$e(Y, P) = e(X + H_2(Hash(ID_V, m, R)||X)PK_S, P_{pub}) \qquad (2)$$

If the equality holds, the verifier is convinced of the validity of signature σ and invalid otherwise.

5. **Partially convertibility:** The partially conversion algorithm $CConv$ converts a DV-DCS signature $\sigma = \{X, Z, R\}$ into an chameleon signature $s = \{X, Y, R\}$. Here the $CConv$ is a decryption operation: $Y = Dec_{SK_C}(Z)$.

6. **Signature verification :** The verification algorithm $COVer(m, s, PK_S)$ outputs 1 if equation (2) satisfied, and 0 otherwise.

Remark. The new DV-DCS scheme is different from traditional DCS because our scheme doesn't have *selective convertibility* property which confirmer signature can be converted into ordinary signature. For protecting privacy of signer and recipient, such as "Legal will problem", the converted signature needn't to be universally verifiable.

On the other hand, the new scheme is easily turned into signature with message recovery if we using tricks from PSS-R [5].

5 Security Analysis

Theorem 1 The above algorithms and protocols constitutes a secure DV-DCS scheme, given that chameleon signature is existentially unforgeable under chosen message attack and ENC is a public key encryption scheme secure against adaptively chosen-ciphertext attack .

Proof. (sketch)

- **Security for signer:** We first need to show that any PPT adversary $\mathcal{A}$, playing $CSig$ in the role of verifier on polynomially many adaptively messages $\{m_1, \ldots, m_k\}$ of his choice, cannot successfully run $COVer$ on a new message $m \notin \{m_1, \ldots, m_k\}$ with non-negligible probability. Suppose for contradiction that such an $\mathcal{A}$ exist. Assuming $\mathcal{A}$ choose a new message $m \notin \{m_1, \ldots, m_k\}$ previous not signed, and pass the verification $COVer$ with non-negligible probability. Passing the algorithm $COVer$ implies that $\mathcal{A}$ can extract chameleon signature s' from DV-DCS signature σ correctly. If $\mathcal{A}$ doesn't know confirmer's secret key SK_C, successfully extracting s' from σ means $\mathcal{A}$ can decrypt σ correctlly. It contradicts that ENC is a $\mathcal{CCA}2$ security encryption scheme. If $\mathcal{A}$ knows SK_C[2], the conversion from σ to s' is trivial. Assuming $\mathcal{A}$ knows SK_C, the proof is induction to $\mathcal{A}$ can forge σ with non-negligible success probability. Note that $\mathcal{A}$ does not know the trapdoor information SK_V, $\mathcal{A}$ cann't forge z in equation (1) by forging algorithm. (This is characteristic of chameleon hash function). It follows that $\mathcal{A}$ forges a confirmer signature σ on a new message m, requiring either find collision in hash function H_2 or to break the underlying chameleon signature scheme. It contradicts that the hash function H_2 is collision-resistance and the underlying ID-based chameleon signature is secure against chosen message attack. Therefore, for all such $\mathcal{A}$, all sufficiently large n and for any message $m \notin \{m_i\}$,

$$Pr(COVer(m, s', PK_S) = accept) < negl(n)$$

[2] In such case, we assume the confirmer acting as adversary $\mathcal{A}$.

The probability is taken over the coin tosses of the $S, \mathcal{A}$ and the key generation algorithms $CKGS, CKGC$.

- **Security for the confirmer:**

  Suppose a PPT adversary $\mathcal{A}$, on input strings $1^n, PK_s, PK_c$ make oracle queries to the signers and confirmer via $CSig$,$Conf_{(C,\mathcal{A})}$,$Disavowal_{(C,\mathcal{A})}$ for polynomially many inputs of his choice and finally, for a pair (m, σ') of his choice, $\mathcal{A}$ executes $Conf_{(\mathcal{A},V)}(1^n, m, \sigma', PK_S, PK_C)$ with non-negligible probability.

  Successfully running $Conf_{(\mathcal{A},V)}$ means that $\mathcal{A}$ can extract ordinary signature with non-negligible probability. It contradicts the underlying encryption scheme ENC is secure against adaptively chosen-ciphertext attack.

Remark. There exists distinction on non-transferability between confirmer signatures and chameleon signatures. In confirmer signatures, non-transferability means that given a confirmer signature σ generated by signer S, verifier V cannot convince a third party of it's validity because the verification of the signature needs the cooperation of the confirmer. In comparison with confirmer signatures, the recipient in chameleon signatures is fully capable of providing any indistinguishable chameleon hashing inputs to satisfy the signature, thus the third party can not trust the recipient's claim. Such is the assurance of non-transferability in chameleon signatures.

Furthermore, in our scheme, if a dishonest verifier disclose the signature σ and s to the third party after having interacted in a confirmation and disavowal protocol, the third party can not trust him because the verifier can forge z in equation (1) using his trapdoor information and sequentially forge verification equation (2). A dishonest confirmer also can't reveal σ and s to any other party because s can only be verified by the intended recipient V. So non-transferability is enhanced in our scheme.

Theorem 2 *The confirm and disavowal protocol have the following properties:*

- ***Completeness:*** *given a valid (invalid) signature σ on message m, if confirmer and verifier comply legally with the protocol, then the protocol always returns σ as a valid (invalid) signature.*

- ***Soundness:*** *given an invalid (valid) signature σ on message m, confirmer is unable to convince verifier of accepting it as valid (invalid) signature with non-negligible probability.*

Proof.

1. **Completeness:** Assuming the signature σ is valid. If confirmer and verifier follow the protocol, we have:

$$\begin{aligned}
&e(X + H_2(Hash(ID_V, m, R)||X)PK_S, P_{pub}) \\
&= e(r \cdot PK_S + H_2(Hash(ID_V, m, R)||X)PK_S, P_{pub}) \\
&= e((r + h)PK_S, P_{pub}) \\
&= e((r + h)PK_S, sP) \\
&= e((r + h)sPK_S, P) \\
&= e((r + h)SK_S, P) \\
&= e(Y, P)
\end{aligned}$$

Then the confirm protocol will return σ as a valid signature.

If the signature $\sigma' = \{X', Z', R'\}$ is invalid, confirmer first transforms it into signature $s' = \{X', Y', R'\}$ and sends it to signer. If s' is invalid, signer can always provide a collision $\{X, Y, R\}$ in the chameleon hash function, since $\{X, Y, R\}$ was originally generated by himself. Therefore, confirmer cannot assert the invalid σ' as a valid signature to the verifier.

2. **Soundness:** If the signature σ is invalid, the proof is similar to the second part proof of completeness. If σ is valid, suppose for contradiction that confirmer is able to deny the σ with non-negligible probability. Therefore, such confirmer could provide $\sigma' = \{X', Z', R'\}$ different from $\sigma = \{X, Z, R\}$ and let verifier pass the signature verification. It shows that another $Z' \neq Z$ is $Y's$ ciphertext too. This constitutes a malleability attack on the ENC encryption scheme. Since we are using a secure against adaptively chosen-ciphertext attack encryption scheme and $\mathcal{CCA}2$ security implies non-malleability security([3]). It induces a contradiction.

Remark. Theorem 2 guarantee the non-repudiation property of our DV-DCS scheme.

6 Efficiency

In the signing stage of our DV-DCS scheme, signer need to computer one point scalar multiplication of $\mathbb{G}_1$ and two pairing operations. The computation of pairing requires high cost compared with the computation cost for power operation over the finite fields or on the elliptic curve. By using precomputation technique, ie. pre-compute pairing operation and then use the mid value in later computation, we could implement the scheme efficiently.

Our scheme comply with hash-then-sign paradigm and do not employ zero-knowledge proof in confirm and disavowal protocol. The scheme combines simplicity and efficiency properties of ID-based chameleon signature.

7 Conclusions

Designated confirmer signature have many applications in real world such as fair contact signing, auctions etc., where confirmer plays semi-trusted third party role. In this paper, we proposed a new DV-DCS scheme transformed from ID-based chameleon signature. The new scheme is recipient-specific and confirmer-specific. To our knowledge, such signature scheme hasn't been discussed in literature. Furthermore, the scheme provides a solution to "legal will problem".

Our new scheme is non-interactive and combines the advantage of ID-based chameleon signature, efficiency and simplicity. The new scheme is efficient with respect to traditional confirmer signatures and with a security analysis considering the confirmer as an adversary.

Further work may focus on new scheme in the multi-user setting (e.g. designated at least two recipients).

References

[1] G. Ateniese and B. de Medeiros, Identity-based chameleon hash and applications, Cryptology ePrint Archive, Report 2003/167/, 2003. Available at http://eprint.iacr.org/2003/167/.

[2] J. Boyar, D. Chaum, I. Damgård, T. Pedersen, Convertible Undeniable Signatures, In: Alfred J. Menezes *et al.* ed. Proceedings of the Advances in Cryptology-Crypto '90, LNCS 537, Springer-Verlag, Berlin, 1991, 189-205.

[3] Mihir Bellare, Anand Desai, David Pointcheval, and Phillip Rogaway, Relations Among Notions of Security for Public-Key Encryption Schemes, In: H. Krawczyk ed. Proceedings of the Advances in Cryptology-Crypto '98, LNCS 1462, Springer-Verlag, Berlin, 1998, 26-46.

[4] D. Boneh and M. Franklin, Identity-based encryption from the Weil pairing, In: G. Goos *et al.* ed. Proceedings of the Advances in Cryptology- Crypto 2001, LNCS 2139, Springer-Verlag, Berlin, 2001, 213-229.

[5] M. Bellare and P. Rogaway, The exact security of digital signatures: How to sign with RSA and Rabin, Advances in Cryptology - Eurocrypt 96 Proceedings, Lecture Notes in Computer Science Vol. 1070, U. Maurer ed., Springer-Verlag, 1996.

[6] David Chaum, Hans Van Antwerpen, Undeniable signatures, In: G. Brassard, ed. Proceedings of the Advances in Cryptology- Crypto '89. LNCS 435, Springer-Verlag, Berlin, 1990. 212-216.

[7] David Chaum, Designated Confirmer Signatures, In: De Santis A, ed. Proceedings of the Advances in Cryptology- EUROCRYPT '94. LNCS 950, Springer-Verlag, Berlin, 1994. 86-89.

[8] Jan Camenisch, Markus Michels, Confirmer Signature Schemes Secure against Adaptive Adversaries. In: Preneel B, ed. Proceedings of the Advances in Cryptology- EUROCRYPT 2000. LNCS 1807, Springer-Verlag, Berlin, 2000. 243-258.

[9] D. Chaum, E. van Heijst, B. Pfitzmann, Cryptographically strong undeniable signatures, unconditionally secure for the signer, In: Joan Feigenbaum, ed. Proceedings of the Advances in Cryptology-Crypto '91, LNCS 576, Springer-Verlag, Berlin, 1992, 470-484.

[10] S. D. Galbraith, W. Mao, Invisibility and anonymity of undeniable and confirmer signatures, In: M. Joye, ed. Topics in Cryptology CT-RSA 2003, Springer, LNCS 2612, 2003, 80-97.

[11] S. Goldwasser, E. Waisbard, Transformation of Digital Signature Schemes into Designated Confirmer Signature Schemes, First Theory of Cryptography Conference, TCC 2004, LNCS 2951, Springer-Verlag, Heidelberg, 2004, 77-100.

[12] R. Gennaro,H. Krawczyk, T. Rabin. RSA-based Undeniable Signatures. In: Burt Kaliski ed. Proceedings of the Advances in Cryptology-Crypto '97, LNCS 1294, Springer-Verlag, Berlin, 1997, 132-149.

[13] S. Han, K. Y. Yeung and J. Wang, Identity-based confirmer signatures from pairings over elliptic curves, Proceedings of ACM conference on Electronic commerce citation 2003, San Diego, CA, USA, 2003, 262-263.

[14] M. Jakobsson, K. Sako, R. Impagliazzo, Designated Verifier Proofs and Their Applications, In: Ueli Maurer ed. Proceedings of the Advances in Cryptology- EUROCRYPT '96, LNCS 1070, Springer-Verlag, Berlin, 1996, 143-154.

[15] H. Krawczyk and T. Rabin, Chameleon hashing and signatures, Proceedings of NDSS 2000, 2000, 143-154.

[16] B. Libert and J.Quisquater, ID-based undeniable signatures, Advances in CT-RSA 2004, LNCS 2964, Springer-Verlag, Heidelberg, 2004. 112-125.

[17] Markus Michels, Markus Stadler, Generic constructions for secure and effcient confirmer signature schemes. In: Nyberg K, ed. Proceedings of the Advances in Cryptology- EUROCRYPT '98, LNCS 1403, Springer-Verlag, Berlin, 1998, 406-412.

[18] Tatsuaki Okamoto, Designated confirmer signatures and public-key encryption are equivalent. In: Desmendt YG, ed. Proceedings of the Ad-

vances in Cryptology- Crypto '94. LNCS 839, Springer-Verlag, Berlin, 1994. 61-74.

[19] Torben Pryds Pedersen: Distributed Provers with Applications to Undeniable Signatures (Extended abstract), In: Donald W. Davies ed. Proceedings of the Advances in Cryptology- EUROCRYPT '91, LNCS 547, Springer-Verlag, Berlin, 1991. 221-242.

[20] Wang GL, Qing SH. Security flaws in a confirmer signature scheme. Journal of Software, 2004, 15(5): 752-756. (in Chinese).

[21] Wang SP, Wang YM, Zhang YL. A confirmer signature scheme based on DSA and RSA. Journal of Software, 2003, 14(3): 588-593. (in Chinese).

[22] Fangguo Zhang, Reihaneh Safavi-Naini, and Willy Susilo. Attack on Han et al.'s ID-based Confirmer (Undeniable) Signature at ACM-EC 03, Cryptology ePrint Archive: Report 2003/129, Available at http://eprint.iacr.org/2003/129/.

[23] Fangguo Zhang, Reihaneh Safavi-Naini, and Willy Susilo. ID-Based Chameleon Hashes from Bilinear Pairings, Cryptology ePrint Archive: Report 2003/208, Available at http://eprint.iacr.org/2003/208/.