A Provably Secure Blind Signature Scheme

Xiaoming Hu and Shangteng Huang

Department of Computer Science and Engineering,
Shanghai JiaoTong University, Shanghai 200240, China
huxm@sjtu.edu.cn

Abstract. Some blind signature schemes have been constructed from some underlying signature schemes, which are efficient and provably secure in the random oracle. To the best of authors' knowledge, a problem still remains: does the security of the original signature scheme, by itself, imply the security of the blind version? In this paper, we answer the question. We show if the blind factors in the blind version come from hash functions, the design of blind signature scheme can be validated in random oracle model if the original scheme is provably secure. We propose a blind version of Schnorr signature scheme and reduce the security of the proposed scheme to the security of ECDLP. What's more, the complexity of this reduction is polynomial in all suitable parameters in the random oracle.

Keywords: Blind signature, Provably secure, Polynomial reduction, Security arguments.

1 Instruction

Blind signatures, introduced by [1], provide anonymity of users in application such as electronic voting and electronic payment systems. A blind signature scheme is an interactive two-party protocol between a user and a signer. It allows the user to obtain a signature of any given message, but the signer learns neither the message nor the resulting signature. Blind signature plays a central role in building anonymous electronic cash. A lot of work has been done in field of blind signature schemes since Chaum [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19, 20, 21, 22, 23]. Several blind signature schemes [1, 4, 6, 12] have been constructed from some underlying signature schemes [24, 25, 26]. These underlying signature schemes are efficient and have been validated in the so-called random oracle model. However, a problem still remains: the security of the original signature scheme does not, by itself, imply the security of the blind version.

The random oracle [27] model is a popular alternative of provable security. Some blind signature schemes were proposed and proven secure in the model [10, 11, 12, 13, 14, 28]. In [12, 28], Pointcheval and Stern prove the security of several blind digital signatures schemes, including blind variation of the [8], [25], and [5] signature schemes. However, their security proofs, while polynomial in the size of the keys, are poly-logarithmically bounded in the number of blind digital signatures. The authors leaves an open problem that whether

J.-Y. Cai, S.B. Cooper, and H. Zhu (Eds.): TAMC 2007, LNCS 4484, pp. 171–180, 2007.

one can achieve polynomial time both in the number of signatures obtained by the adversary and the size of the keys. Juels, et al. [29] gave a positive answer to show that security and blindness properties for blind signatures could be simultaneously defined and satisfied, assuming an arbitrary one-way trapdoor permutation family. However, their proof was based on complexity. As they had discussed, their schemes should be viewed merely as a proof of existence which pave the way for efficient future implementations. Pointcheval et al. [10] proposed a blind signature scheme based on factorization, unluckily it also need a user to make a poly-logarithmically bounded number of interactions with the signer. The less practical schemes of [11] are provably secure for a polynomial number of synchronized signer interactions, where the synchronization forces the completion of each step for all the different protocol invocations before the next step of any other invocation is started, so some restrictions apply. [13] requires a non-standard strong assumption - namely the RSA chosen-target inversion problem is hard. [14] proposed an efficient three-move blind signature scheme, which provides one more unforgeability with polynomially many signatures. However, the scheme is a specific blind signature scheme which prevents one-more unforgeability after polynomially many interactions with the signer, so it isn't a generic approach. In this paper, by using hash functions to generate the blind factors, we show that the design of blind signature scheme can be validated in random oracle model if the original scheme is provably secure in random oracle model. We propose a blind version of Schnorr signature scheme. Moreover, we show that the proposed blind signature is provably secure if ECDLP is hard, and the complexity of this reduction is polynomial in all suitable parameters.

The rest of the paper is organized as follows. In section 2 we recall some definitions for blind signatures. Section 3 proposes our blind signature scheme. Section 4 gives the proof of nonforgeablility of the proposed scheme. Section 5 concludes this paper.

2 Preliminaries

In this section we review the formal definition and the standard security notion of blind signature schemes [29].

Definition 1. *A blind digital signature scheme is a four-tuple* $(Signer, User, Gen, Verify)$*.*
- $Gen(1^k)$ *is a probabilistic polynomial-time key-generation algorithm that takes security parameter* k *and outputs a public and secret key pair* (pk, sk)*.*
- $Signer(pk, sk)$ *and* $User(pk, m)$ *are a pair of polynomially-bounded probabilistic Interactive Turing machines, each of which has a public input tape, a private random tape, a private work tape, a private output tape, a public output tape, and input and output communication tapes. The random tape and the input tapes are read-only, and the output tapes are write-only. The private work tape is*

read-write. They are both given pk generated by $Gen(1^k)$ on theirs public input tapes. Additionally, the private input tape of Signer contains the private key sk, and that for User contains message m. User and Signer engage in the signature issuing protocol and stop in polynomial-time in k. At the end of this protocol, Signer outputs either completed or not-completed on his public output tap, and User outputs either $\perp$ or $\sigma(m)$ on his private output tap.
- Verify$(pk, m, \sigma(m))$ is a deterministic polynomial-time algorithm. On input $(pk, m, \sigma(m))$ and outputs accept/reject. with the requirement that for any message m, and for all random choices of key generation algorithm, if both Signer and User follow the protocol then the Signer always outputs completed, and the output of the user is always accepted by the verification algorithm.

Definition 2. *If Signer and User follow the signature issuing protocol, then with probability of at least $1 - 1/k^c$ for every constant c and sufficiently large k, Signer outputs completed and User outputs $(m, \sigma(m))$ that satisfies Verify $(pk, m, \sigma(m))$ =accepted. The probability is taken over the coin flips of Gen, Signer and User.*
A blind digital signature scheme is secure if it holds the following two properties:

Definition 3. *Let S^* be adversarial signer, and u_0 and u_1 are two honest users.*
- $(pk, sk) \leftarrow Gen(1^k)$.
- $m_0, m_1 \leftarrow S^*(1^k, pk, sk)$.
- Set the input tap of u_0 and u_1 as follows : Let $b \in_R \{0, 1\}, put\{m_b, m_{1-b}\}$ on the private input tap of u_0 and u_1, respectively;Put pk on the public input taps of u_0 and u_1, respectively;Randomly select the contents of the private random tapes.
- S^* engages in the signature issuing protocol with u_0 and u_1.
- If u_0 and u_1 output valid signature $(m_b, \sigma(m_b))$ and $(m_{1-b}, \sigma(m_{1-b}))$, respectively, then send $(\sigma(m_b), \sigma(m_{1-b}))$ to S^*. Give $\perp$ to S^* otherwise.
- S^* outputs $b' \in \{0, 1\}$. If b'= b, then S^* wins.
A blind signature scheme is blind if all probabilistic polynomial-time algorithm S^*, S^* outputs b'= b with probability at most $1/2 + 1/k^c$ for some constant c and sufficiently large k. The probability is taken over the flips of Gen, S^*, u_0 and u_1.

Definition 4. *Let U^* be adversarial user and S be an honest signer.*
- $(Step1) : (pk, sk) \leftarrow Gen(1^k)$.
- $(Step2)$: $U^*(pk)$ engages in the signature issuing protocol with S in adaptive, parallel and arbitrarily interleaved way. Let L denote the number of executions, where S outputted completed in the end of Step 2.
- $(Step3)$: U^* outputs a collection $\{(m_1, \sigma(m_1)), (m_j, (m_j))\}$ subject to the constraint the all $(m_i, \sigma(m_i))$ for $1 \leq i \leq j$ are all accepted by verify $(pk, m_i, \sigma(m_i))$.
A blind signature scheme is nonforgeable if the probability, taken over the coin flips of Gen, U^* and S, that $j > l$ is at most $1/k^c$ for some constant c and sufficiently large k.

3 The Proposed Scheme

In this section, we propose s blind signature scheme. It can be seen as a slight modification of Schnorr Blind Signature Scheme.

Setup of System Parameters. Before the whole scheme can be initialized, the following parameters over the elliptic curve domain must be known.
- A field size p, which is a large odd prime.
- Two parameters $a, b \in F_p$ to define the equation of the elliptic curve E over $F_p(y^2 = x^3 + ax + b(\bmod p)$ in the case $p > 3)$, where $4a^3 + 27b^2 \neq 0(\bmod p)$. The cardinality of E must be divisible by a large prime because of the issue of security raised by Pohlig and Hellman [30].
- A finite point P whose order is a large prime number in $E(F_p)$, where $P \neq O$, and O denotes infinity.
- The order of P is prime q.
- Two public cryptographically strong hash functions $H_1 : \{0,1\}^* \times E(F_p) \rightarrow Z_q$ and $H_2 : Z_q \times Z_q \rightarrow Z_q$. We remark that H_1 and H_2 will be viewed as random oracles in our security proof.

Setup of a Principal's public/private key. The signer picks a random number $x \in_R Z_q^*$ and computer $Q = xP$. His public key is Q and private key is x. The public keys Q must be certified by the CA.

Signature Generation
- The signer randomly chooses $d, e, r \in Z_q$, computes $U' = rP$, and sends (U', d, e) to the user.
- Blind. The user randomly chooses m_i, α'_i and $\beta'_i \in Z_q$, $1 \leq i \leq n$. He computes $\alpha_i = H_2(\alpha'_i, d)$ and $\beta_i = H_2(\beta'_i, e)$ as blinding factors, $U_i = U' - \alpha_i P - \beta_i Q$, $h_i = H_1(m_i, U_i)$ and $h'_i = h_i + \beta_i$, $1 \leq i \leq n$, sends to the signer n blinded candidates $h'_i, 1 \leq i \leq n$.
- The signer verifies if the user constructs h'_i with the blind factors α or β which are the outputs of the random oracle H_2.The signer randomly chooses a subset of n-1 blinded candidates indices $R = i_j$, 1≤ $i_j \leq n$ for 1≤ $j \leq n$-1 and sends R to the user.
- The user reveals (m_i, α'_i and β'_i) to the signer for all i in R.
- The signer verifies h'_i for all i in R. He computes $k_i = H_2(\alpha'_i, d)$, $\lambda_i = H_2(\beta'_i, e)$, $U_i = U' - k_i P - \lambda_i Q$, $h_i = H_1(m_i, U_i)$ and $H'_i = h_i + \lambda_i$ for all i in R. He accepts h'_i to be a valid blind message if $H'_i = h'_i$. If the signer accepts h'_i for all i in R, then the signer performs the following operations for the rest h'_i, i not belong to R, denote by h', and the corresponding parameters are $(m, \alpha, \beta, U, h, h_i)$. The signer stops otherwise.
- Blind Sign. The signer sends back s', where $s' = r - xh'$.
- Unblind. He outputs a signature $\sigma = (s, h)$ where $s = s' - \alpha$. Then σ is the signature of the message m.

Note: The method which the user prepares n blinded candidates but only 1 out of n is finally used, all other n-1 are opened, verified and thrown away, inevitably cause enormous computational and communication overhead. An alternative to cut down computational and communication overhead is as follows.

The user prepares n blinded candidates and sends them to the signer. The signer randomly chooses $n/2$ out of n to verify them. If all of them pass the verification, then the signer randomly chooses 1 out of the rest $n/2$ blinded candidates to perform blind signature. The signer stops otherwise. It is obvious that the signer's computational and communication overhead is almost half less than the original's, but we will show in Lemma3 that the probability of be caught as the user doesn't generate blind factors from the random oracle H_2, is almost the same with the original's.

Signature Verification. The verifier or recipient of the blind signature accepts the blind signature if and only if $H_1(m, sP + hQ) = h$. Signature verification works correctly because if (s, h) is a valid signature on the message m, then $sP + hQ = (s' - \alpha)P + hQ = (r - xh' - \alpha)P + hQ = rP - (h + \beta)Q - \alpha P + hQ = rP - \alpha P - \beta Q = U' - \alpha P - \beta Q = U$.

It is straightforward to prove our scheme satisfies the blindness property [25]. So in this paper, we will only show that our scheme satisfies nonforgeability property.

4 Security Proofs

In this section, we first show that the adversary should following the protocol. Next, we prove that the security of our scheme can be reduced to security of Schnorr signature scheme and further the security of ECDLP, and the complexity of the reduction is fully polynomial in all suitable parameters.

Lemma 1. *If an adversary constructs h' with α or β which are not the outputs of H_2, the probability the signer accept h' is negligible when the signer verify h'.*

Proof. If signer verify h', the adversary should find a pair(α', β') that satisfies:

$$\alpha = H_2(\alpha', d) \ . \tag{1}$$

$$\beta = H_2(\beta', e) \ . \tag{2}$$

$$U = U' - \alpha P - \beta Q \ . \tag{3}$$

$$h' = H_1(m, U) + \beta \ . \tag{4}$$

Since H_2 is a hash function, the probability he succeeds is negligible. □

Lemma 2. *Let A be the adversary who tries to destroy the requirement that constructs blind candidates with blind factors α or β which are the outputs of the random oracle H2. If there exists 1 out of n blinded candidates $h'_i (1 \leq i \leq n)$, which blind factors α or β are not the outputs of H_2, then A is caught with probability 1-1/n; If there exists ≥ 2 out of n blinded candidates $h'_i (1 \leq i \leq n)$, which blind factors α or β are not the outputs of H_2, then A is caught with probability 1.*

Proof. There are n-1 blind candidates to satisfy the requirement of generating blind factors from H_2, so A isn't caught with probability $C_{n-1}^{n-1}/C_{n}^{n-1}$, namely $1/n$. Thus, the probability A is caught is 1-1/n. Similarly, we can get if there exists ≥ 2 out of n blinded candidates, which blind factors α or β are not the outputs of H_2, then A is caught with probability 1. □

Lemma 3. *Consider the above alternative, namely chooses n out of 2n to verify. Let A be the adversary who tries to destroy the requirement that constructs blind candidates with blind factors α or β which are the outputs of the random oracle H_2. Let ε be the probability of blinded candidates that blind factors α or β are not the outputs of the random oracle H_2, then the signer signs finally on a blind candidate, which blind factors α or β are the outputs of the random oracle H_2, with probability at least 1- $\varepsilon 2^{-2\varepsilon n}$.*

Proof. The number of blinded candidates which satisfy the requirement of generating blind factors from H_2 is $2(1-\varepsilon)n$, so A pass the verification with probability at most

$$C^n_{2(1-\varepsilon)n}/C^n_{2n} = ((2n-2\varepsilon n)!n!)/((n-2\varepsilon n)!(2n)!)\ . \tag{5}$$

$$C^n_{2(1-\varepsilon)n}/C^n_{2n} = n(n-1)(n-2\varepsilon n+1))/((2n)(2n-1)(2n-2\varepsilon n+1))\ . \tag{6}$$

$$C^n_{2(1-\varepsilon)n}/C^n_{2n} = 1/((1+n/n)(1+n/(n-1))(1+n/(n-2n+1)))\ . \tag{7}$$

Then, the signer randomly chooses a blind candidate from the rest n blind candidates. The probability of getting a blind candidate which blind factors α or β are not the outputs of the random oracle H_2, is 2n/n=. Thus, the signer signs finally on a blind candidates, which blind factors α or β are the outputs of the random oracle H_2, with probability at least $1-\varepsilon C^n_{2(1-\varepsilon)n}/C^n_{2n}$. It is obvious that $2^{-2\varepsilon n} = 1/(1+1)^{2n\varepsilon} > C^n_{2(1-\varepsilon)n}/C^n_{2n} > 1/(1+1/(1-2\varepsilon-n^{-1})^{2n\varepsilon}) \Rightarrow C^n_{2(1-\varepsilon)n}/C^n_{2n} < 2^{-2\varepsilon n} \Rightarrow 1-\varepsilon C^n_{2(1-\varepsilon)n}/C^n_{2n} > 1-\varepsilon 2^{-2\varepsilon n}$.

So, the probability is at least $1-\varepsilon 2^{-2\varepsilon n}$. $\varepsilon 2^{-2\varepsilon n}$ is negligible when ε is sufficient large. Thus, the signer is assured that he is signing on a blind candidate that blind factors α or β are the outputs of the random oracle H_2, except a negligible probability. □

Lemma 4. *The proposed scheme is secure against one-more forgery assuming Schnorr signature scheme is secure. Concretely, suppose there is a one-more forgery adversary A that has advantage ε against our scheme within running time t. Let H_1, H_2 be random oracles. Assume that A makes at most $q_{H_1} > 0$ hash queries to H_1, $q_{H_2} > 0$ hash queries to H_2, and $q_s > 0$ signing queries to the signer. Then there is an algorithm B that performs a valid forgery against Schnorr signature scheme with probability ε in running time at most $t+t_{H_1}q_{H_1}+t_{H_2}q_{H_2}+(t_s+\tau)q_s$, where t_{H_1} is time for answering an H_1 query, t_{H_2} is time for answering an H_2 query, t_s is time for Schnorr signature scheme to generate a signature, τ and is time for answering an signing query to our scheme.*

Proof. Suppose C is the signer in Schnorr signature scheme. He keeps the secret key sk and publishes the public key pk. We show how to construct a simulator

B that uses A to forge a signature of Schnorr signature scheme for C with advantage ε. B first sends pk to A and claims that pk is his public key. Next, define the queries as follows:

- Sign query. From Lemma 1, Lemma 2 and Lemma 3, we know that A should follow the protocol. When A makes a signing query, B engages in blind signature protocol with A as follows:

B randomly chooses randomly message m' and makes a sign query to C for signature on message m'. C returns a signature (m', U, s, h). B randomly chooses $\alpha, \beta, d, e \in Z_q$ and computes $U = U' + \alpha P + \beta Q$, $s = s' + \alpha$, and sends (U', d, e) to A; A chooses α', $\beta' \in Z_q$, and queries to B for $H_2(\alpha', d)$ and $H_2(\beta', e)$. B returns (α, β) to the adversary A; A computes $U = U' - \alpha P - \beta Q$ and queries B for $H_1(m, U)$. B returns $h = H_1(m, U)$ to the adversary A; A computes $h' = h + \beta$ and sends it to B; B returns $s' = s + \alpha$; A outputs a signature $\sigma = (m, U, h, s)$ where $s = s' - \alpha$.

- Hash query. If A makes H_1 query to B, B sends the query to random oracle H_1 and returns the result from H_1 to A. If A makes H_2 query, B returns μ randomly chosen from Z_q.

It is straightforward to verify that signing query produce "valid" signatures. There is a collisions problem of the query result of H_1 query. In the sign query, B asks C to sign on the message m' which is randomly choose, C returns a valid signature (m', U, h, s). h is the random oracle H_1's answer to query $H_1(m', U)$. B simulates random oracle H_1 and cheats A that $h = H_1(m, U)$. But if A queries the same query $H_1(m', U)$ to B where the query does not come from the sign query, B returns the random oracle H_1's answer h. This may cause some "collision": a query result of H_1 query may produce a value of H_1 that is inconsistent with other query results of H_1. In this case, B just outputs fail and exits. However, since the message m' is randomly choose, the possibility of collisions is negligible. □

Lemma 5. *B simulates the signer C with an indistinguishable distribution.*

Proof. In the signing query, B simulates the signer without the secret key in the blind signature protocol. Furthermore, in the above queries, the answer to H_1 query comes from random oracle H_1 and the answer to H_2 query is randomly choose from Z_q, so they are uniformly random in their respective spaces. Therefore B simulates the signer with an indistinguishable distribution.

Since A is a successful adversary against the proposed scheme and B simulates the signer with an indistinguishable distribution, A will forge a valid signature (m_0, U_0, h_0, s_0). Since (m_0, U_0, h_0, s_0) is not equal to the outputs of the signing query, h_0 must be the right answer to query $H_1(m_0, U_0)$. So (m_0, U_0, h_0, s_0) is a valid signature for C. Thus using A, B forges a valid signature of Schnorr signature scheme for C. Since A has advantage ε within running time t, B succeeds a forgery with advantage ε in running time at most $t + t_{H_1} q_{H_1} + t_{H_2} q_{H_2} + (t_s + \tau) q_s$. □

By the above proof, we know that B makes - query to H_1 and signing queries to Schnorr signature scheme. Again, B has probability ε against Schnorr signature scheme in running time at most $t + t_{H_1}q_{H_1} + t_{H_2}q_{H_2} + (t_s + \tau)q_s$. According to the Lemma 4 of [12], we obtain the following result:

Theorem 1. *The proposed scheme is secure against one-more forgery assume that ECDLP is hard in groups $E(F_p)$. Concretely, assume that there is an one-more forgery adversary A that has advantage ε against the proposed scheme within running time t. Let H_1, H_2 are random oracles. Assume that A makes at most q_{H_1} ¿ 0 hash queries to H_1, q_{H_2} ¿ 0 hash queries to H_2, and q_s ¿ 0 sign queries to Signer. If $\varepsilon \geq 10(q_s + 1)q_{H_1}/p$, then there is an algorithm C that solves the ECDLP problem in group $E(F_p)$ with probability $\varepsilon' \geq 1/9$ and at most time $t' \leq 23(q_{H_1} - q_s)(t + t_{H_1}q_{H_1} + t_{H_2}q_{H_2} + (t_s + \tau)q_s)/\varepsilon$, where t_{H_1} is time for answering an H_1 query, t_{H_2} is time for answering an H_2 query, t_s is time for Schnorr signature scheme to generate a signature, and τ is time for answering an signing query to the proposed scheme.* □

It is obvious that the complexity of this reduction is fully polynomial in all suitable parameters.

5 Conclusions

In this paper, we present an efficient blind signature scheme which prevents one-more forgery in the random oracle. The proof of security is fully polynomial in all suitable parameters in random oracle model. We also show that using hash functions to make the blind factors, the design of blind signature scheme can be proved secure in random oracle model assume that the original scheme is provably secure. The proposed security reduction can be an efficient technique in the proof of security for blind signature schemes.

Acknowledgment. We thank the anonymous reviewers for comprehensive comments.

References

1. Chaum, D.: Blind signatures for untraceable payments. Advances in Cryptology Crypto'82, LNCS, (1982) 199-203
2. Chaum, D.: Blind signature system. Proceedings of Crypto'83, Plenum, (1983) 153
3. Chaum, D., Fiat, A., Naor, M.: Untraceable electronic cash. Proceedings of Crypto'88, LNCS, (1988) 319-327
4. Chaum,D.: Security without identification. Transaction Systems to Make Big Brother Obsolete. Commun- ications of the ACM 28, (1985)
5. Guillou, L.C., Quisquater, J.J.: A practical zero-knowledge protocol fitter to security microprocessor minimizing both transmission and memory. EUROCRYPT, (1988)
6. Chaum, D.: Privacy protected payments: unconditional payer and/or payee untraceability. In Smartcard 2000, (1989) 69-93
7. Chaum, D., Boen, B., Heyst, E.: Efficient off-line electronic check. In Quisquater J, Vandewalle J, eds. Proceedings of the Eurocrypt'89, LNCS, **434** (1990) 294-301

8. Okamoto, T.: Provably secure and practical identification schemes and corresponding signature schemes. CRYPTO, (1992)
9. Camenisch, J. L., Piveteau, J. M., Stadler, M. A.: Blind signatures based on the discrete logarithm problem. Lecture Notes in Computer Science, **950** (1995) 428-432
10. Pointcheval, D., Stern, J.: New blind signatures equivalent to factorization. In ACM SSS, ACM Press, (1997) 92-99
11. Pointcheval, D.: Strengthened security for blind signatures. Eurocrypt'98, LNCS, (1998) 391-405
12. Pointcheval,D., Stern, J.: Security arguments for digital signatures and blind signatures. Journal of Cryptology, **3(13)** (2000) 361-396
13. Bellare, M., Namprempre, C., Pointcheval, D.: The power of RSA inversion oracles and the security of Chaum's RSA-based blind signature scheme. In Proceedings of Financial Cryptography 01, Springer-Verlag, (2001)
14. Abe, M.: A secure three-move blind signature scheme for polynomially many signature. Eurocrypt'01, LNCS, **2045** (2001) 136-151
15. Zhang, F., Kim, K.: ID-based blind signature and ring signature from pairings. Proc. of Asiacrpt2002, LNCS, **2501** (2002) 533-547
16. Zhang, F., Kim, K.: Efficient ID-Based blind signature and proxy signature from bilinear pairings. ACISP'03, LNCS, **2727** (2003) 312-323
17. Sherman, S.M., Lucas, C.K., Yiu, S.M.: Two improved partially blind signature schemes from bilinear pairings. Available at: http://eprint.iacr.org/2004/108.pdf
18. Bellare, M., Namprempre, C., Neven, G.: Security proofs for identity-based identification and signature schemes. Christian Cachin and Jan Camenisch, editors: Advances in Cryptology EUROCRYPT 2004, Lecture Notes in Computer Science, **3027** (2004) 268-286
19. Camenisch, J., Koprowski, M., Warinschi, B.: Efficient blind signatures without random oracles. Blundo. Security in Communication Networks-SCN 2004, Lecture Notes in Computer Science, Berlin Heidelberg New York, **3352** (2005) 134-148
20. Liao, J., Qi, Y.H., Huang, P.W.: Pairing-based provable blind signature scheme without random oracles. CIS 2005, (2005) 161-166
21. Okamoto, T.: Efficient blind and partially blind signatures without random oracles. In Shai Halevi and Tal Rabin, editors, TCC 2006, 3rd Theory of Cryptography Conference, Lecture Notes in Computer Science, New York, NY, USA, March 4-7, **3876** (2006) 80-99
22. Fischlin, M.: Round-optimal composable blind signatures in the common reference string model. In Cynthia Dwork, editor, Advances in Cryptology, CRYPTO 2006, Lecture Notes in Computer Science, Santa Barbara, CA, USA, August 20-24, **4117** (2006) 60-77
23. Galindo, D., Herranz, J., Kiltz, K.: On the generic construction of identity-based signatures with additional properties. ASIACRYPT, (2006) 178-193
24. Rivest, R., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public key cryptosystems. Communications of the ACM, **2(21)** (1978) 120-126
25. Schnorr, C. P.: Efficient identification and signatures for smart cards. In Crypto '89, **435** (1990) 235-251
26. Schnorr, C. P.: Efficient signature generation by smart cards. Journal of Cryptology, **3(4)** (1991) 161-174
27. Bellare, M., Rogaway, P.: Random oracles are practical: a paradigm for designing efficient protocols. In Proc, of the 1st CSSS, ACM Press, (1993) 62-73
28. Pointcheval, D., Stern, J.: Provably secure blind signature schemes. Asiacrypt, (1996)

29. Juels, A., Luby, M., Ostrovsky, R.: Security of blind digital signatures. In Proceedings of Crypto'97, LNCS, **1294** (1997) 150-164
30. Pohlig, S., Hellman, M.: An improved algorithm for computing logarithms over GF(p) and its cryptographic significance. IEEE Transactions on Information Theory, **24** (1978) 106-110