Identity-Based Proxy Signature from Pairings

Wei Wu, Yi Mu, Willy Susilo, Jennifer Seberry, and Xinyi Huang

Centre for Computer and Information Security Research
School of Computer Science & Software Engineering
University of Wollongong, Australia
weiwu81@gmail.com,{ymu,wsusilo,j.seberry,xh068}@uow.edu.au

Abstract. A proxy signature scheme allows an entity to delegate its signing capability to another entity (proxy) in such a way that the proxy can sign messages on behalf of the delegator. Proxy signatures have found numerous practical applications such as distributed systems, mobile agent applications, etc. Recently, Xu, Zhang and Feng proposed the first formal models of identity based proxy signature. Unfortunately, their model does not capture the notion of adaptively chosen message and chosen identity attacker in identity based system. In this paper, we redefine the security models of identity based proxy signature to capture the most stringent attacks against adaptively chosen message and chosen identity attacker. We also propose a new provably secure identity basad proxy signature scheme whose security is based on the hardness of Computational Diffie-Hellman problem in the random oracle model.

1 Introduction

Traditional public-key cryptography (PKC) has many applications; however, PKC seems less attractable in distributed and ad hoc systems, since the requirement of public-key infrastructure prevents its applications in this field. The notion of identity-based cryptosystem was introduced by Shamir in his seminal paper in [22]. The main essence of identity-based cryptosystem is to remove the need of certification of the public keys. The public key of each party is obtained from his/her public identity, such as the IP address in the ad hoc system, which can uniquely identify the party. Since its introduction in [22], many identity based schemes have been proposed (e.g., [2,15,20,25]).

On the other hand, permanent connections between customers and servers in this kind of system are unnecessary and infeasible. In order to ensure service availability to the customers distributed in the whole networks, the server must delegate his rights to some other parties in the systems, such as the mobile agents. This way, replication can be achieved and there is no need to count on a single server. A proxy signature scheme is a variation of the standard signature schemes, in which an original signer (say, Alice) can delegate his signing right to another signer, called the proxy signer (say, Bob), for signing messages. The notion of proxy signature was introduced by Mambo, Usuda and Okamoto [17]. Since then, proxy signature schemes have attracted a considerable amount of interest from the cryptographic research community. Based on the delegation type,

B. Xiao et al. (Eds.): ATC 2007, LNCS 4610, pp. 22–31, 2007.

there are three types of proxy signatures: *full delegation*, *partial delegation*, and *delegation by warrant.* In the full delegation system, Alice's secret key is given to Bob directly so that Bob can have the same signing capability as Alice. In practice, such schemes are obviously impractical and insecure. In a partial delegation proxy signature scheme, a proxy signer possesses a key, called private proxy key, which is different from Alice's private key. Hence, proxy signatures generated by using the proxy private key are different from Alice's signatures. However, in such schemes, the messages a proxy signer can sign are *not* limited. This weakness is eliminated in delegation by a warrant that specifies what kinds of messages are delegated. Here, the original signer uses the signing algorithm of a standard signature scheme and its secret key to sign a warrant and generate a signature on the warrant which is called as delegation. The proxy signer uses the delegation and his secret key to create a proxy signature on behalf of the original signer. According to whether the original signer can generate a valid proxy signature or not, proxy signatures can be classified into *proxy-unprotected* and *proxy-protected* schemes. In a proxy-protected scheme only the proxy signer can generate proxy signatures, while in a proxy-unprotected scheme either the proxy signer or the original signer can generate proxy signatures. In many applications, proxy-protected schemes are required to avoid the potential disputes between the original signer and the proxy signer. Though there exist many proxy signature schemes, most of them are insecure [14,11,13,18,19,23]. Recently, based on the work of [4,16], Xu, Zhang and Feng formalized the notion of security for ID-based proxy signature schemes and proposed a scheme based on the bilinear pairings [26]. However, as we will show later, the model defined in their paper does not capture the definitions of adaptively chosen message and chosen identity attacker in identity based system.

Our contribution

Firstly, we redefine the security notion of ID-based proxy signature schemes to capture the most stringent attacks in this model, namely the adaptively chosen message and chosen identity attacks. Compared with the model proposed in [26], our model captures a stronger security notion of the proxy signature by allowing the adversaries to behave more adaptively in oracle accessing. The adversary can freely choose the identities of the original signer and the proxy signer. We proceed by proposing a new identity based proxy signature scheme whose security is based on the hardness of the Computational Diffie-Hellman problem in the random oracle model. Compared with the scheme proposed in [26], the new proposed scheme enjoys less operation cost.

Roadmap

The rest of this paper is arranged as follows. In next section, we provide the preliminaries of our scheme including bilinear pairings and security assumptions. In Section 3, we describe the formal models of our ID-based proxy signature scheme. We present our ID-based proxy signature scheme with its security analysis in Section 4. Finally, we conclude our paper in Section 5.

2 Preliminaries

2.1 Bilinear Pairing

Let $\mathbb{G}_1$ be a cyclic additive group of prime order q, P is the generator of $\mathbb{G}_1$, $\mathbb{G}_T$ denotes a multiplicative group of the same order. Let $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_T$ be a bilinear pairing with the following properties:

- e is bilinear: $e(aP, bP) = e(P, P)^{ab}$, for all $a, b \in \mathbb{Z}_q$.
- e is non-degenerate: $e(P, P) \neq 1_{\mathbb{G}_T}$.
- e is efficiently computable.

We say $\mathbb{G}_1$ is a bilinear group if there exists a group $\mathbb{G}_T$, and a bilinear pairing $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_T$ as above, and e, and the group action in $\mathbb{G}_1$ and $\mathbb{G}_T$ can be computed efficiently. Using the bilinear pairing on certain elliptic curves over a finite field of characteristic, the elements in $\mathbb{G}_1$ can have short representation [3,9].

2.2 Security Definitions

Definition 1. Computational Diffie-Hellman (CDH) on $\mathbb{G}_1$,
Given $P, aP, bP \in \mathbb{G}_1$, *for some unknown* $a, b \in_R \mathbb{Z}_q$, *compute* $abP \in \mathbb{G}_1$.

The success probability of an algorithm $\mathcal{A}$ in solving the CDH problem on $\mathbb{G}_1$ is denoted as $\mathsf{Succ}_{\mathcal{A},\ \mathbb{G}_1}^{CDH} = Pr[\mathcal{A}(P, aP, bP) = abP : a, b \in_R \mathbb{Z}_q]$.

Definition 2. Computational Diffie-Hellman (CDH) Assumption on $\mathbb{G}_1$,
Given $P, aP, bP \in_R \mathbb{G}_1$, *for some unknown* $a, b \in \mathbb{Z}_q$, $\mathsf{Succ}_{\mathcal{A},\ \mathbb{G}_1}^{CDH}$ *is negligible for any polynomially bounded algorithm.*

3 Formal Models of ID-Based Proxy Signatures

Let Alice denote the original signer and Bob the proxy signer. The ID-based proxy signature scheme consists of the following algorithms: ParaGen, KeyExtract, StandardSign, StandardVer, DelegationGen, ProxySign and ProxyVer.

1. ParaGen: Taking as input the system security parameter ℓ, this algorithm outputs system's parameters Para and the system's master key s. That is: $(\mathsf{Para}, s) \leftarrow \mathsf{ParaGen}(\ell)$.
2. KeyExtract: Taking as input system's parameter Para and an identity ID_i where $i \in \{a, b\}$ denotes the identities of Alice and Bob, respectively, this algorithm generates a secret key sk_{ID_i} for them. That is: $sk_{ID_i} \leftarrow \mathsf{KeyExtract}$ (Para, ID_i, s).
3. StandardSign: Input system's parameter Para, the signer's secret key sk_{ID} and the message M to be signed, this algorithm outputs the standard signature σ_S. That is: $\sigma_S \leftarrow \mathsf{StandardSign}(\mathsf{Para}, M, sk_{ID})$.

4. $\mathsf{StandardVer}$: Input system's parameter Para, the signer's identity ID, the signed message M and the standard signature σ_S, this algorithm outputs True if σ_S is a valid standard signing of the message M under the identity ID and outputs $\perp$ otherwise. That is: $\{\mathsf{Ture}, \perp\} \leftarrow \mathsf{StandardVer}(\mathsf{Para}, ID, M, \sigma_S)$.
5. $\mathsf{DelegationGen}$: Input system's parameter Para, the original signer's secret key sk_{ID_a} and the warrant W to be signed, this algorithm uses the $\mathsf{StandardSign}$ algorithm to generate the delegation σ_w. That is: $\sigma_w \leftarrow \mathsf{DelegationGen}(\mathsf{Para}, W, sk_{ID_a})$.
6. $\mathsf{ProxySign}$: Input system's parameter Para, the warrant W, the delegation σ_w, the secret key sk_{ID_b} of the proxy signer and the message M to be signed, this algorithm generates the proxy signature σ. That is: $\sigma \leftarrow \mathsf{ProxySign}(\mathsf{Para}, W, \sigma_w, sk_{ID_b}, M)$.
7. $\mathsf{ProxyVer}$: Input system's parameter Para, original signer's identity ID_a, proxy signer's identity ID_b, the warrant W, the signed message M and the signature σ, this algorithm outputs True if σ is a valid proxy signature of the message M and outputs $\perp$ otherwise. That is: $\{\mathsf{True}, \perp\} \leftarrow \mathsf{ProxyVer}(\mathsf{Para}, ID_a, ID_b, W, M, \sigma)$.

3.1 Security Models

In [26], Xu, Zhang and Feng proposed the first formal security model of identity-based proxy signature. Actually, this model is a variation of the security model in the traditional public key system which is proposed in [4,16]. While their model provides some properties that an identity-based proxy signature schemes should capture, there are two weaknesses of the models defined in [26].

1. The attacker $\mathcal{A}$'s target identity ID_1 is given to $\mathcal{A}$ before $\mathcal{A}$ submits queries to the challenger. However, we normally allow $\mathcal{A}$ to choose the target identity adaptively after he received responses of all the queries.
2. When $\mathcal{A}$ outputs a forgery signature $(m, psign)$ under the proxy signing key skp (which is defined in their scheme), with original signer ID_1 and the proxy signer ID_i, $\mathcal{A}$ cannot request the proxy signatures of other messages under this proxy signing key. However, we normally allow $\mathcal{A}$ to obtain the signatures of the message m' under this signing key with the only restriction that $m \neq m'$.

Similarly to the model defined in [10], we divide the potential adversary into the following three types:

1. **Type I:** This type adversary $\mathcal{A}_I$ only has the public keys (identities) of Alice and Bob.
2. **Type II:** This type adversary $\mathcal{A}_{II}$ has the public keys (identities) of Alice and Bob, and also can have the secret key of the proxy signer Bob.
3. **Type III:** This type adversary $\mathcal{A}_{III}$ has the public keys (identities) of Alice and Bob, and also can have the secret key of the original signer Alice.

One can find that if an ID-based proxy signature scheme is secure against Type II (or Type III) adversary, the scheme is also secure against Type I adversary. We note the above classification helps to make the security model clearer; therefore, we will use this classification to improve the security model proposed in [26]. In a warrant based proxy signature, the delegation is the original signer's standard signature on the warrant which contains information regarding the proxy signer such as the proxy signer's ID, a period of validity, the restriction on the class of messages for which the warrant is valid. Therefore, this kind of proxy signature can prevent the misuse of the delegation. Here after, we only focus on the unforgeability of the proxy signature.

Existential unforgeability under adaptive A_{II} Adversary
Roughly speaking, a valid ID-based proxy signature σ of the message M under the warrant W shows that the original signer agrees on this warrant and has signed this warrant. Therefore, even the adversary can obtain the secret key of the proxy signer, he cannot create a valid ID-based proxy signature under the warrant W if he does not obtain the delegation of this warrant. It is defined using the following game between the challenger $\mathcal{C}$ and a type II adversary $\mathcal{A}_{II}$:

- Setup: $\mathcal{C}$ runs the ParaGen algorithm to obtain system's parameter para and the master key s.
- KeyExtract queries: Given an identity ID, $\mathcal{C}$ returns the private key sk_{ID} corresponding to ID.
- StandSign queries: $\mathcal{A}_{II}$ can request the signature of M under the identity ID. In response, $\mathcal{C}$ runs the StandSign algorithm to obtain σ_S and returns σ_S to the adversary $\mathcal{A}_{II}$. Especially, $\mathcal{A}_{II}$ can request the original signer ID_A's delegation (that is the standard signature) on (W, ID_A, ID_B) where W denotes the warrant, ID_B denotes the proxy signer's identity and W, ID_A, ID_B are chosen by $\mathcal{A}_{II}$ adaptively. In response, $\mathcal{C}$ runs the StandSign algorithm to sign the message (W, ID_A, ID_B) to generate σ_W. Then $\mathcal{C}$ returns σ_W to the adversary $\mathcal{A}_{II}$.
- ProxySign queries: Proceeding adaptively, $\mathcal{A}_{II}$ can request the proxy signature of $(W, M, ID_A,\ ID_B)$ where W is the warrant, M is the message to be signed, ID_A is the original signer's identity and ID_B is the proxy signer's identity. In response, $\mathcal{C}$ firstly runs the KeyExtract algorithm to obtain the secret keys of the original signer and proxy signer, respectively. Then $\mathcal{C}$ runs the StandSign algorithm to sign the message (W, ID_A, ID_B) and generates the delegation σ_W. At last, $\mathcal{C}$ runs the ProxySign algorithm and generates the proxy signature σ. Then $\mathcal{C}$ returns σ to the adversary $\mathcal{A}_{II}$ as response.
- Output: Finally, $\mathcal{A}_{II}$ outputs $(M^*, W_f, \overline{ID}_A, \overline{ID}_B, \sigma^*)$ where $\overline{ID}_A$ is the identity of original signer, $\overline{ID}_B$ is the identity of proxy signer, W_f is the warrant, M^* is the message and σ^* is the signature which satisfy that:
  1. $\overline{ID}_A$ has not been requested as one of the KeyExtract queries.
  2. $(W_f, \overline{ID}_A, \overline{ID}_B)$ has not been requested as one of the StandSign queries.
  3. $(M^*, W_f, \overline{ID}_A, \overline{ID}_B)$ has not been requested as one of the ProxySign queries.
  4. σ^* is a valid ID-based proxy signature of the message m^* under the warrant W_f, the original signer $\overline{ID}_A$ and the proxy signer $\overline{ID}_B$.

Remark: Compared with the model defined in [26], an important refinement is that we allow $\mathcal{A}_{II}$ to adaptively submit the ProxySign queries under the warrant whose delegation is unknown to him. The only restrictions are that when $\mathcal{A}_{II}$ outputs the forgery $(M^*, W_f, \overline{ID}_A, \overline{ID}_B, \sigma^*)$, he cannot submit $\overline{ID}_A$ as one of the KeyExtract queries, or $(W_f, \overline{ID}_A, \overline{ID}_B)$ as one of the StandSign queries, or submit $(M^*, W_f, \overline{ID}_A, \overline{ID}_B, \sigma^*)$ as one of the ProxySign queries. However, he can even submit $\overline{ID}_B$ to the KeyExtract queries, $(W_f', \overline{ID}_A, \overline{ID}_B)$ to the StandardSign queries where $W_f' \neq W_f$ and $(M', W_f, \overline{ID}_A, \overline{ID}_B)$ to the ProxySign queries where $M' \neq M^*$.

The success probability of an algorithm $\mathcal{A}_{II}$ wins the above game is defined as Succ$\mathcal{A}_{II}$.

Definition 3. *We say a type II adversary $\mathcal{A}_{II}$ can $(t, q_H, q_{KE}, q_S, q_{PS}, \varepsilon)$ break a proxy signature scheme if A_{II} runs in time at most t, $\mathcal{A}_{II}$ makes at most q_H queries to the hash functions, at most q_{KE}* KeyExtract *queries, at most q_S* StandardSign *queries and at most q_{PS}* ProxySign *queries, and* $\mathsf{Succ}_{\mathcal{A}_{II}}$ *is at least ε.*

Existential unforgeability under adaptive A_{III} adversary
Roughly speaking, this property states that only the proxy signer can create a valid proxy signature, even the original signer can not. Given a valid ID-based proxy signature, the proxy signer cannot deny the fact that he has signed the message. The existential unforgeability of a proxy signature scheme under a type III attacker requires that it is difficult for the original signer to output a valid proxy signature of a message M^* which has not been signed by the proxy signer. It is defined using the games as same as those games between $\mathcal{A}_{II}$ and $\mathcal{C}$. After all the queries,

- Output: Finally, A_{III} outputs $(M^*, W_f, \overline{ID}_A, \overline{ID}_B, \sigma^*)$ where $\overline{ID}_A$ is the identity of original signer, $\overline{ID}_B$ is the identity of proxy signature, W_f is the warrant, M^* is the message to be signed and σ^* is the ID-based proxy signature which satisfy that:
  1. $\overline{ID}_B$ has not been requested as one of the KeyExtract queries.
  2. $(M^*, W_f, \overline{ID}_A, \overline{ID}_B, \sigma^*)$ has not been requested as one of the ProxySign queries.
  3. σ^* is a valid ID-based proxy signature of the message m^* under the warrant W_f, the original signer $\overline{ID}_A$ and the proxy signer $\overline{ID}_B$.

The success probability of an algorithm $\mathcal{A}_{III}$ wins the above game is defined as $\mathsf{Succ}_{\mathcal{A}_{III}}$.

Definition 4. *We say a type III adversary $\mathcal{A}_{III}$ can $(t, q_H, q_{KE}, q_S, q_{PS}, \varepsilon)$ break a proxy signature scheme if A_{III} runs in time at most t, $\mathcal{A}_{III}$ makes at most q_H queries to the hash functions, at most q_{KE}* KeyExtract *queries, at most q_S* StandardSign *queries and at most q_{PS}* ProxySign *queries, and* $\mathsf{Succ}_{\mathcal{A}_{III}}$ *is at least ε.*

4 Proposed ID-Based Proxy Signature Scheme

In this section, we present our construction of ID-based proxy signature scheme. The scheme consists of the following algorithms.

1. $\mathsf{ParaGen}$: Input the system's parameter ℓ, this algorithm generates a bilinear group $\mathbb{G}_1$ of prime order q ($q \geq 2^\ell$) such that CDH problem is hard in $\mathbb{G}_1$. Let $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_T$ be the bilinear pairing. The generator of $\mathbb{G}_1$ is P. Pick a random master key $s \in \mathbb{Z}_q^*$ and set $P_{pub} = sP$. It also chooses three distinct secure hash functions $H_0, H_1, H_2 : \{0,1\}^* \rightarrow \mathbb{G}_1$. Then the system's parameter is: $\mathsf{Para} = \{\ell, \mathbb{G}_1, \mathbb{G}_T, q, e, P, P_{pub}, H_0, H_1, H_2\}$.
2. $\mathsf{KeyExtract}$: Given a user's identity ID, compute $H_0(ID) \in \mathbb{G}_1$ and $sk_{ID} = sH_0(ID)$.
3. $\mathsf{StandardSign}$: Let M be the message to be signed, the standard signature is generated as: $\sigma_S = (sk_{ID} + rH_1(M), rP)$ where $r \in_R \mathbb{Z}_q^*$.
4. $\mathsf{StandardVer}$: Given the identity ID of the signer, the message M and a signature σ_S, verify whether $e(\sigma_S, P) \overset{?}{=} e(H_0(ID), P_{pub})e(H_1(M), rP)$.
5. $\mathsf{DelegationGen}$: Let W be the warrant to be signed by the original signer Alice with the identity ID_A who wants to delegate his signing rights to Bob with the identity ID_B, the delegation is generated as: $\sigma_W = (sk_{ID_A} + r_A H_1(W, ID_A, ID_B), r_A P)$ where $r_A \in_R \mathbb{Z}_q^*$. Then Alice sends the warrant W and delegation σ_W to the proxy signer Bob.
6. $\mathsf{ProxySign}$: Given the secret key sk_{ID_B}, the delegation $\sigma_W = (sk_{ID_A} + r_A H_1(W, ID_A, ID_B), r_A P)$ of the warrant W and a message M, the proxy signer chooses $r_B \in_R \mathbb{Z}_q^*$ and computes $\sigma = (\sigma_1, \sigma_2, \sigma_3)$ where $\sigma_1 = sk_{ID_A} + r_A H_1(W, ID_A, ID_B) + sk_{ID_B} + r_B H_2(M, W, ID_A, ID_B)$, $\sigma_2 = r_A P, \sigma_3 = r_B P$.
7. $\mathsf{ProxyVer}$: Given the identities (ID_A, ID_B) of original signer and proxy signer, a warrant $W \in \{0,1\}^*$, a message $M \in \{0,1\}^*$, and a signature $\sigma = (\sigma_1, \sigma_2, \sigma_3)$, verify whether $e(\sigma_1, P) \overset{?}{=} e(H_0(ID_A), P_{pub})e(H_0(ID_B), P_{pub})\ e(H_1(W, ID_A, ID_B), \sigma_2)\ e(H_2(M, W, ID_A, ID_B), \sigma_3)$. If the equality holds the result is True; otherwise the result is $\perp$.

4.1 Unforgeability Against Type II Adversary

Theorem 1. *If there exists a type II adversary $\mathcal{A}_{II}$ who can $(t, q_H, q_{KE}, q_S, q_{PS}, \varepsilon)$ break the proposed proxy signature scheme then there exists another algorithm $\mathcal{B}$ who can use $\mathcal{A}_{II}$ to solve an instance of the CDH problem in $\mathbb{G}_1$ with probability*

$$\mathsf{Succ}_{\mathcal{B},\ \mathbb{G}_1}^{CDH} \geq \left(\frac{3}{q_{KE} + q_S + q_{PS} + 1}\right)^3 \left(1 - \frac{3}{q_{KE} + q_S + q_{PS} + 4}\right)^{q_{KE}+q_S+q_{PS}+4} \varepsilon$$

in time $t + c_1(q_H + q_{KE} + 3q_S + 9q_{PS}) + c_2(q_H + 2q_S + 8q_{PS})$. Here c_1, c_2 are two constants that depends on $\mathbb{G}_1$.

Proof. We are forced to omit it due to page limitation.

4.2 Unforgeability Against Type III Adversary

Theorem 2. *If there exists a type III adversary $\mathcal{A}_{III}$ who can $(t, q_H, q_{KE}, q_S, q_{PS}, \varepsilon)$ break the proposed proxy signature scheme then there exists another algorithm $\mathcal{B}$ who can use $\mathcal{A}_{III}$ to solve an instance of the CDH problem in $\mathbb{G}_1$ with probability*

$$\mathsf{Succ}_{\mathcal{B},\ \mathbb{G}_1}^{CDH} \geq (\frac{3}{q_{KE}+q_S+q_{PS}+1})^3(1-\frac{3}{q_{KE}+q_S+q_{PS}+4})^{q_{KE}+q_S+q_{PS}+4}\varepsilon$$

in time $t + c_1(q_H + q_{KE} + 3q_S + 9q_{PS}) + c_2(q_H + 2q_S + 8q_{PS})$. Here c_1, c_2 are two constants that depends on $\mathbb{G}_1$.

Proof. The proof is completely similar to the proof of Theorem 1.

4.3 Efficiency Analysis

In this section we compare our scheme with Xu *et al.*'s scheme [26] in the sense of signature length and operation cost of verification. The two schemes require the same operation cost in the delegation and proxy sign algorithms. In the following table, the notion $|\mathbb{G}_1|$ denotes the bit length of an element in $\mathbb{G}_1$.

Table 1. Comparison between Xu *et al.*'s scheme [26] and our scheme

Scheme	**Signature Length**	**Pairings in Verification**	**exp. in $\mathbb{G}_2$**
Xu et al.'s scheme	$3\lvert\mathbb{G}_1\rvert$	4 (2 can be precomputed)	1
Our scheme	$3\lvert\mathbb{G}_1\rvert$	4 (2 can be precomputed)	0

5 Conclusion

In this paper, we improve the security models of identity based proxy signature defined by Xu, Zhang and Feng [26] by allowing adversaries to behave more adaptively in oracle accessing. We then propose a new identity based proxy signature which is secure against adaptively chosen message and chosen identity attacker. Compared with the scheme proposed in [26], the new proposed scheme enjoys less operation cost, and hence, it outperforms the existing schemes in the literature. The security of the proposed scheme is based on the hardness of the Computational Diffie-Hellman problem in the random oracle model.

References

1. Boneh, D., Boyen, X.: Short Signatures without Random Oracles. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 56–73. Springer, Heidelberg (2004)

2. Boneh, D., Franklin, M.: Identity-based Encryption from the Weil Pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (2001)
3. Boneh, D., Lynn, B., Shacham, H.: Short Signatures from the Weil Pairing. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 514–532. Springer, Heidelberg (2001)
4. Boldyreva, A., Palacio, A., Warinschi, B.: Secure Proxy Signature Scheme for Delegation of Signing Rights. In: IACR ePrint Archive (2003) available at `http://eprint.iacr.org/2003/096/`
5. Bellare, M., Rogaway, P.: The Exact Security of Digital Signatures - How to Sign with RSA and Rabin. In: De Santis, A. (ed.) EUROCRYPT 1994. LNCS, vol. 950, pp. 399–416. Springer, Heidelberg (1995)
6. Cheon, J.H.: Security Analysis of the Strong Diffie-Hellman Problem. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 1–11. Springer, Heidelberg (2006)
7. Canetti, R., Goldreich, O., Halevi, S.: The Random Oracle Methodology, revisited. In: Proceedings of the 30th Annual Symposium on the Theory of Computing-STOC'98, pp. 209–218 (1998)
8. Goldwasser, S., Micali, S., Rivest, R.: A Digital Signature Scheme Secure Against Adaptively Chosen Message Attacks. SIAM Journal on Computing 17(2), 281–308 (1988)
9. Galbraith, S.D., Paterson, K.G., Smart, N.P.: Pairings for Cryptographers. In: IACR ePrint Archive (2006) available at `http://eprint.iacr.org/2006/165/`
10. Huang, X., Mu, Y., Susilo, W., Zhang, F., Chen, X.: A Short Proxy Signature Scheme: Efficient Authentication in the Ubiquitous World. In: Enokido, T., Yan, L., Xiao, B., Kim, D., Dai, Y., Yang, L.T. (eds.) Embedded and Ubiquitous Computing – EUC 2005 Workshops. LNCS, vol. 3823, pp. 480–489. Springer, Heidelberg (2005)
11. Lee, J.-Y., Cheon, J.H., Kim, S.: An Analysis of Proxy Signatures: Is a Secure Channel Necessary? In: Joye, M. (ed.) CT-RSA 2003. LNCS, vol. 2612, pp. 68–79. Springer, Heidelberg (2003)
12. Lee, B., Kim, H., Kim, K.: Strong Proxy Signature and Its Applications. In: Proc of SCIS'01, pp. 603–608 (2001)
13. Lee, B., Kim, H., Kim, K.: Secure Mobile Agent Using Strong Nondesignated Proxy Signature. In: Varadharajan, V., Mu, Y. (eds.) ACISP 2001. LNCS, vol. 2119, pp. 474–486. Springer, Heidelberg (2001)
14. Kim, S., Park, S., Won, D.: Proxy Signatures, revisited. In: Han, Y., Quing, S. (eds.) ICICS 1997. LNCS, vol. 1334, pp. 223–232. Springer, Heidelberg (1997)
15. Hess, F.: Efficient Identity Based Signature Schemes Based on Pairings. In: Nyberg, K., Heys, H.M. (eds.) SAC 2002. LNCS, vol. 2595, pp. 310–324. Springer, Heidelberg (2003)
16. Malkin, T., Obana, S., Yung, M.: The Hierarchy of Key Evolving Signatures and a Characterization of Proxy Signatures. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 306–322. Springer, Heidelberg (2004)
17. Mambo, M., Usuda, K., Okamoto, E.: Proxy Signature: Delegation Of the Power to Sign Messages. IEICE Trans. Fundamentals E79-A(9), 1338–1353 (1996)
18. Okamoto, T., Inomata, A., Okamoto, E.: A Proposal of Short Proxy Signature Using Pairing. In: International Conference on Information Technology (ITCC 2005), pp. 631–635. IEEE Computer Society Press, Los Alamitos (2005)
19. Okamoto, T., Tada, M., Okamoto, E.: Extended Proxy Signatures for Smart Cards. In: Zheng, Y., Mambo, M. (eds.) ISW 1999. LNCS, vol. 1729, pp. 247–258. Springer, Heidelberg (1999)

20. Paterson, K.G., Schuldt1, J.C.N.: Efficient Identity-based Signatures Secure in the Standard Model. In: Batten, L.M., Safavi-Naini, R. (eds.) ACISP 2006. LNCS, vol. 4058, pp. 207–222. Springer, Heidelberg (2006)
21. Park, H.-U., Lee, I.-Y.A.: Digital Nominative Proxy Signature Scheme for Mobile Communications. In: Qing, S., Okamoto, T., Zhou, J. (eds.) ICICS 2001. LNCS, vol. 2229, pp. 451–455. Springer, Heidelberg (2001)
22. Shamir, A.: Identity-based Cryptosystems and Signature Schemes. In: Blakely, G.R., Chaum, D. (eds.) CRYPTO 1984. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (1985)
23. Wang, G., Bao, F., Zhou, J., Deng, R.H.: Security Analysis of Some Proxy Signatures. In: Lim, J.-I., Lee, D.-H. (eds.) ICISC 2003. LNCS, vol. 2971, pp. 305–319. Springer, Heidelberg (2004)
24. Zhang, F., Safavi-Naini, R., Susilo, W.: An Efficient Signature Scheme from Bilinear Pairings and Its Applications. In: Bao, F., Deng, R., Zhou, J. (eds.) PKC 2004. LNCS, vol. 2947, pp. 277–290. Springer, Heidelberg (2004)
25. Zhang, F., Susilo, W., Mu, Y.: Identity-Based Partial Message Recovery Signatures (or How to Shorten ID-Based Signatures). In: Patrick, A.S., Yung, M. (eds.) FC 2005. LNCS, vol. 3570, pp. 45–56. Springer, Heidelberg (2005)
26. Xu, J., Zhang, Z., Feng, D.: ID-based Proxy Signature Using Bilinear Pairings. In: Chen, G., Pan, Y., Guo, M., Lu, J. (eds.) Parallel and Distributed Processing and Applications - ISPA 2005 Workshops. LNCS, vol. 3759, pp. 359–367. Springer, Heidelberg (2005)