

Efficient Multi-proxy Signature Scheme from Bilinear Pairing

Rajeev Anand Sahu and Sahadeo Padhye

Department of Mathematics, Motilal Nehru National

Institute of Technology, Allahabad-211004, India

E-mail: rajeevs1729@gmail.com; sahadeomathrsu@gmail.com

(Received November 15, 2010)

Abstract: Digital signatures are efficient tools for source authentication in cryptography. The concept of proxy signature enables a signer to delegate its signing capability to any other party, called proxy signer. In this paper, an efficient multi-proxy signature scheme is proposed from bilinear pairing, using the Boneh, Lynn and Shacham's (BLS) short signature¹ and ID-based signature of Xun Yi². The size of signature in our scheme is less than those of other existing schemes³⁻⁴. Furthermore, our scheme satisfies all the security requirements of a proxy signature, as given in⁵.

Keywords: Signature scheme, Bilinear pairing, Multi-proxy signature.

2000 Mathematics AMS Classification Number: 94A60

1. Introduction

In cryptography, a digital signature is used for the purpose of source authenticity and the data integrity assurance. In a signature scheme, the signer uses its private key to make signature on a message, and any verifier in public domain can verify that signature using signer's public key. The concept of bilinear pairing over elliptic curves has opened new dimensions in cryptography. In particular, the notion of pairing is quite helping in computational jobs in signature protocols. Specially, the pairing-based cryptosystems provide efficient tools for construction of special signature schemes as short signature, multi signature, aggregate signature etc⁷⁻⁸⁻⁹.

The concept of proxy signature was introduced by Mambo, Usuda and Okamoto¹⁰ in 1996. In a proxy signature scheme, an original signer is authorized to delegate its signing capability to a proxy signer. According to the number of signers in the original and proxy group, the proxy signature can be divided in three categories multi-proxy signature, proxy multi-signature and multi-proxy multi-signature. The multi-proxy signature is due to Hwang and Shi¹¹ in which, an original signer can authorize a group of

proxy signers, to generate the signature on his/her behalf. In a contrary scenario, a group of original signers may require a single proxy signer. To handle such situations proxy multi-signature was proposed by Yi et al.¹² in 2000. The third type of signature, called multi-proxy multi-signature is proposed by Hwang and Chen¹³ in 2004. A multi-proxy multi-signature is a signature, generated by a group of proxy signers on behalf of the group of original signers.

1.1. Our Contribution: In this paper, we have proposed an efficient multi-proxy signature scheme from bilinear pairing, using the BLS short signature¹ and ID-based signature of Xun Yi². Our scheme is computationally more efficient than other existing schemes^{3, 4}. Moreover, signature size in our scheme is less than those of schemes given in^{3, 4}. We also analyze the security properties of our scheme and show that the proposed scheme is secure.

1.2. Organization of this paper: The rest of this paper is organized as follows: In Section 2, we introduce the bilinear pairing, computational Diffie-Hellman problem and the security requirements for a proxy signature. In Section 3, we review the BLS short signature scheme. The proposed scheme is described in Section 4. Section 5, gives the correctness and investigates the security properties of our scheme. We compare the efficiency of our scheme with others in Section 6 and finally Section 7 concludes the paper.

2. Preliminaries

- (i) Bilinear Pairing: Given two groups G_1 and G_2 of prime order q , a map $e: G_1 \times G_1 \rightarrow G_2$ satisfying the following properties, is called bilinear pairing:
 - (a) Bilinearity: $e(aP, bQ) = e(P, Q)^{ab} \quad \forall a, b \in \mathbb{Z}_q^*$ and $P, Q \in G_1$.
 - (b) Non-Degeneracy: There exists $P, Q \in G_1$ such that $e(P, Q) \neq 1$.
 - (c) Computability: There is an efficient algorithm to compute $e(P, Q)$ for all $P, Q \in G_1$.
- (ii) Computational Diffie-Hellman Problem (CDHP): For given $P, aP, bP \in G_1$, to compute $abP \in G_1$, where $a, b \in \mathbb{Z}_q$.
- (iii) Security requirements for a proxy signature: A secure proxy signature should satisfy the following security requirements⁵:

Strong unforgeability: No one, other than the proxy signer, can generate a valid proxy signature.

Verifiability: The signature can be verified by anyone, and the signed message should confirm to the delegation warrant.

Strong identifiability: Identity of corresponding proxy signer can be determined by anyone.

Strong undeniability: The proxy signer cannot deny his signature made in earlier session.

Prevention of misuse: The proxy signer cannot sign any unauthorized message.

3. BLS Short Signature Scheme

In this section, we briefly review Boneh et al's short signature from the Weil pairing¹. We denote this scheme by BLS short signature scheme. System parameters in our scheme are same as BLS short signature scheme, with an extra hash function; we will define in our setup phase.

Setup: Let G_1 be a cyclic additive group and G_2 be a cyclic multiplicative group of order q , where q is a prime number. Let P be the generator of G_1 and $e : G_1 \times G_1 \rightarrow G_2$ be a bilinear pairing as defined above. Define a map to point hash function $H : \{0,1\}^* \rightarrow G_1$.

Key generation: Choose a random number $s \in {}_R Z_q^*$ as a secret key and compute the public key $P_{pub} = sP$. The system's public parameters $params = \{G_1, G_2, e, q, P, P_{pub}, H\}$.

Signing: To sign a message $M \in \{0,1\}^*$, the signer computes $P_M = H(M) \in G_1$ firstly then computes $S_M = sP_M$. The signature on message M is S_M .

Verification: The verifier accepts the signature S_M on message M , if the equation $e(S_M, P) = e(H(M), P_{pub})$ holds.

Remark: The signature size is almost half of the digital signature generated by DSA, with the same level of security. Hence, it helps to reduce the signature storage and communication cost by half. In this way, it is very

useful to transmit the signature over low bandwidth channels. The scheme is proven to be secure against existential forgery on adaptive chosen message attack in the random oracle model assuming the hardness of CDHP.

4. Proposed Scheme

In this section, we propose an efficient multi-proxy signature scheme using the BLS short signature scheme¹ and ID-based signature scheme of Xun Yi². Our scheme is divided into four phases: System setup phase, Proxy key generation phase, Multi-proxy signature generation phase and Verification phase.

Setup: For a given security parameter k , generate the system's parameters $params = \{G_1, G_2, e, q, P, P_{pub}, H_1, H_2\}$.

where $H_1 : \{0,1\}^* \rightarrow \{0,1\}^n$ for some integer n , and $H_2 : \{0,1\}^* \rightarrow G_1$. The original signer's private key is $s_0 \in {}_R Z_q^*$ and the public key is $PK_0 = s_0 P$.

Similarly, the public-private key pairs for the l proxy signers are $\{PK_1, s_1\}, \{PK_2, s_2\}, \dots, \{PK_l, s_l\}$.

where $PK_i = s_i P$ for $s_i \in Z_q^*$, $(i = 1, 2, \dots, l)$.

Proxy key generation: To delegate the signing capability to a group of proxy signers, the original signer does the following to make the signed warrant w . The warrant w specifies the delegation period, kind of delegated messages, the identity information of original and proxy signers, etc. If the following process is finished successfully, each proxy signer B_i gets a proxy key S_{P_i} for $(i = 1, 2, \dots, l)$.

- The original signer computes $S_w = s_0 H_2(w) \in G_1$, and sends w and S_w to the proxy group.
- Receiving the signature S_w and w , each proxy signer B_i for $(i = 1, 2, \dots, l)$, verifies the validity of signature S_w by checking:

$$e(S_w, P) = e(H_2(w), PK_0).$$
- If the above equality holds, each proxy signer B_i , for $(i = 1, 2, \dots, l)$, accepts the signature S_w and computes their proxy key

$$S_{P_i} = S_w + s_i H_2(w)$$

Multi-proxy signature generation: To generate a multi-proxy signature on a message m with warrant w on behalf of the original signer, each proxy signer B_i generates its partial proxy signature. One proxy signer in the proxy group is designated as a clerk, who combines all the partial proxy signatures to generate the final multi-proxy signature. The pairing computation using the X-coordinate only is due to Galbraith and Lin¹⁴.

- For $(i = 1, 2, \dots, l)$, each proxy signer B_i chooses $r_i \in Z_q^*$
 computes $R_i = r_i P$
 each B_i broadcasts his R_i to the other $(l-1)$ proxy signers
 each proxy signer B_i computes $R = \sum_{i=1}^l R_i$
 and $T_i = r_i PK_i + H(m \| R) S_{P_i}$
- For $(i = 1, 2, \dots, l)$, the pair $((R_i)_X, (T_i)_X)$ is the partial proxy signature by B_i on message m with warrant w , where $(R_i)_X, (T_i)_X$ are the X-coordinates of the point R_i and T_i respectively.
- For $(i = 1, 2, \dots, l)$, each proxy signer sends the tuple $((R_i)_X, (T_i)_X)$ as their partial proxy signature to the clerk in proxy group.
- The clerk verifies the validity of each partial proxy signature $((R_i)_X, (T_i)_X)$, by checking

$$e((T_i)_X, P) = e((R_i)_X + H_1(m \| R) H_2(w), PK_i) e(H_1(m \| R) H_2(w), PK_0)$$

Once all the partial proxy signatures are correct, the clerk in proxy group generates the final multi-proxy signature on message m , with warrant w as

$$(w, R_X, T_X) \text{ by computing } R_X = \sum_{i=1}^l (R_i)_X \text{ and } T_X = \sum_{i=1}^l (T_i)_X.$$

Verification: Receiving the multi-proxy signature (w, R_X, T_X) and message m , the verifier checks the following:

- (1) Checks whether or not the message m confirms to the warrant w . If not, stop. Continue otherwise.
- (2) Checks whether or not the l proxy signers are authorized by the original signer in the warrant w . If not, stop. Continue otherwise.
- (3) Accepts the multi-proxy signature if and only if the following equality holds:

$$e((T)_X, P) = e((R)_X + lH_1(m\|R)H_2(w), \sum_{i=1}^l PK_i) e(lH_1(m\|R)H_2(w), lPK_0).$$

5. Analysis of proposed scheme

5.1. Correctness: In this section, we prove the correctness of verification, as follows:

$$\begin{aligned} e(T_X, P) &= e\left(\sum_{i=1}^l (T_i)_X, P\right) \\ &= e\left(\sum_{i=1}^l r_i PK_i + H_1(m\|R)S_P, P\right) \\ &= e\left(\sum_{i=1}^l [r_i s_i P + H_1(m\|R)(s_0 H_2(w) + s_i H_2(w))], P\right) \\ &= e\left(\sum_{i=1}^l [r_i P + H_1(m\|R)H_2(w)]s_i, P\right) e(lH_1(m\|R)s_0 H_2(w), lP) \\ &= e\left(\sum_{i=1}^l R_i + lH_1(m\|R)H_2(w), \sum_{i=1}^l PK_i\right) e(lH_1(m\|R)H_2(w), lPK_0) \\ &= e(R_X + lH_1(m\|R)H_2(w), \sum_{i=1}^l PK_i) e(lH_1(m\|R)H_2(w), lPK_0) \end{aligned}$$

5.2. Security analysis: In this section, we analyze the security properties of our scheme. We will show that our scheme satisfies all the security requirements of a proxy signature mentioned in second section ⁵.

(i) *Strong unforgeability:* There are mainly three attackers: The original signer, some proxy signers and any outsider.

Firstly, the original signer cannot generate a valid multi-proxy signature, because for this, he will be needed to generate all the partial proxy signatures, which involves the private keys of all proxy signers.

Secondly, If the clerk in proxy group who combines all the partial proxy signatures, wants to sign any unauthorized message m' from the help of all the partial proxy signatures on authorized message m , he can change its

R_i and hence R . But due to the secure signature scheme of Xun Yi² and one way hash function H_1 , it is impossible for the clerk to get T_i' and R_i' such that the tuple (w, R_X', T_X') satisfies the equality in verification phase. Since in our scheme, the clerk, who is unable to forge the multi-proxy signature, has more privilege than the other proxy signers, so no other proxy signers can forge a valid multi-proxy signature.

Finally, any third party who do not participate in the protocol cannot forge the signature even having signature of the original signer, because, to delegate the forge signed warrant, he will be needed private key of the original signer, getting of which is impossible due to the security of BLS short signature scheme¹, based on the hardness of CDHP.

Hence the proposed multi-proxy signature is strongly unforgeable.

- (ii) *Verifiability*: Any verifier can verify the signature and can check whether the signed message confirms to the delegation warrant or not, since the warrant contains identity information and nature of the delegated message. Also, correctness of the verification is discussed above.
- (iii) *Identifiability*: By warrant, anyone can determine the identity of proxy signers.
- (iv) *Strong undeniability*: No proxy signer can deny their signatures, they made in earlier session because the clerk validates all the partial proxy signatures by checking

$$e((T_i)_X, P) = e((R_i)_X + H_1(m \| R)H_2(w), PK_i) e(H_1(m \| R)H_2(w), PK_0).$$

Even, the original signer cannot deny his signature S_w on the warrant w .

- (v) *Prevention of misuse*: Since the warrant is attached with delegation period, nature of message, identity information of the original signer etc., the proxy signers cannot sign any message which does not confirms to the warrant and has not been authorized by the original signer.

6. Analysis of Efficiency and Signature Size

In this section, we compare the efficiency of our scheme with those of³⁻⁴ and will show that our scheme is computationally more efficient than those schemes. We also compare the signature size of our scheme with those of other existing schemes³⁻⁴ and will show that, we have a short signature size in this scheme compare to^{3,4}.

6.1. Efficiency comparison: Here, we compare the computational efficiency of our scheme with others ³⁻⁴.

(i) Proxy key generation phase:

S/N	Schemes	Pairing	Hashing	Exponent
1.	Li and Chen scheme	3	2	1
2.	(2005) ³	3	3	0
3.	Cao and Cao scheme (2009) ⁴	2	2	0
	Our scheme			

(ii) Multi-proxy signature generation phase :

S/N	Schemes	Pairing	Hashing	Exponent
1.	Li and Chen scheme	3	1	1
2.	(2005) ³	5	1	1
3.	Cao and Cao scheme (2009) ⁴	3	1	0
	Our scheme			

(iii) Verification phase:

S/N	Schemes	Pairing	Hashing	Exponent
1.	Li and Chen scheme	3	2	1
2.	(2005) ³	4	3	0
3.	Cao and Cao scheme (2009) ⁴	3	1	0
	Our scheme			

From the above tables, it is clear that our scheme is computationally more efficient than the schemes given in ³⁻⁴.

6.2. Signature size: The following table shows the comparison in signature size of our scheme with those of ^{3,4}. The message length in each scheme is assumed to be n .

S/N	Schemes	Signature size
1.	Li and Chen scheme (2005) ³	$6 \log q + n$
2.	Cao and Cao scheme (2009) ⁴	$6 \log q + n$
3.	Our scheme	$2 \log q + n$

It is clear by the above table that, signature size in our scheme is small than the other existing schemes^{3,4}. Hence, advantage of the proposed scheme is that, signatures can be sent over low bandwidth channels, which enhance the storage space.

7. Conclusion

In this paper, we have proposed a multi-proxy signature scheme based on the Boneh, Lynn and Shacham's (BLS) short signature scheme¹ and ID-based signature scheme of Xun Yi². Our scheme is efficient, and the signature size is less than signature size of other existing schemes^{3,4}. We have also analyzed the security notions of our scheme and have shown that the proposed signature scheme is secure.

References

1. D. Boneh, B. Lynn and H. Shacham, Short Signatures from the Weil Pairing, *In C. Boyd, Ed., Adv. Crypto.-Asiacrypt 2001*, LNCS 2248, gold Coast, Springer Verlag, (2001) 514-532.
2. X. Yi, An Identity-Based Signature Scheme from the Weil Pairing, *IEEE Communication Letters* **7** (2) (2003) 76-78.
3. F. Cao and Z. Cao, A Secure Identity-Based Multi-Proxy Signature Scheme. *Computers and Electrical Engineering*, **35** (2009) 86-95.
4. X. Li and K. Chen, ID-Based Multi-Proxy Signature, Proxy Multi-Signature and Multi-Proxy Multi-Signature Schemes from Bilinear Pairings., *Appl. Math. and Comput.*, **169** (2005) 437-450.
5. B. Lee, H. Kim and K. Kim, Strong Proxy Signature and its Applications. *In: Proc. SCIS.*, (2001) 603-608.
6. A. Shamir, *Identity-Based Cryptosystem and Signature Scheme*, In: *Proc. Crypto' 84*, LNCS **196** Springer Verlag (1984) 47-53.
7. D. Boneh, X. Boyen and H. Shacham, *Short Group Signatures*, *Adv. Crypto: Crypto'04*, LNCS 3152, Springer Verlag (2004) 41-55.
8. A. Boldyreva, Efficient Threshold Signature, Multisignature and Blind Signature Schemes Based on the Gap-Diffie-Hellman-Group Signature Scheme, *PKC'03*, LNCS 2567, Springer Verlag (2003) 31-46.
9. D. Boneh, C. Gentry, B. Lynn and H. Shacham, *Aggregate and Verifiably Encrypted Signatures from Bilinear Maps*, *Adv. Crypto: Eurocrypt' 03*, LNCS 2656, Springer Verlag (2003) 416-432.
10. M. Mambo, K. Usuda and E. Okamoto, Proxy Signatures: Delegation of the Power to Sign Message, *IEICE Trans. Functional E79-A*, **9** (1996) 1338-1354.
11. S. Hwang and C. Shi, A Simple Multi-Proxy Signature Scheme. In: *Proc. 10th national conference on information security, Hualien, Taiwan, ROC* (2000) 134-138.
12. L. Yi, G. Bai and G. Xiao, Proxy Multi-Signature Scheme: A New Type of Proxy Signature Scheme, *Electronics Letters*, **36** (6) (2000) 527-528.

13. S. Hwang and C. Chen, New Multi-Proxy Multi-Signature Schemes. *Appl. Math. Comput.*, **147**(2004) 57- 67.
14. S. D. Galbraith, X. Lin, *Computing Pairing Using X-Coordinate Only*, Des. Codes. Crypto. **50** Springer (2009) 305-324.