

Proxy signature scheme for ID-based original signers and certificate-based proxy signers

Xin Xiangjun¹ Sun Lei²

(¹Department of Mathematics and Information Science, Zhengzhou University of Light Industry, Zhengzhou 450002, China)

(²School of Mathematics and Information Science, Henan Polytechnic University, Jiaozuo 454000, China)

Abstract: To realize delegation between different users in a mixed cryptosystem, a proxy signature scheme for ID-based original signers and certificated-based proxy signers (PSS-ID-CER) is defined. Using the bilinear properties of the pairings and the characters of key generations of certificate-based cryptosystems and ID-based cryptosystems, a construction for such a scheme is also presented. To prove the security of the proposed scheme, a general security model for this scheme under adaptive chosen-PKG, chosen-ID, chosen-delegation, chosen-ProxySigner-public-key, chosen-proxy-key and chosen-message attack is defined. The proposed scheme is provably secure under the random oracle model and the hardness assumption of computational Diffie-Hellman problem.

Key words: proxy signature; ID-based cryptosystem; bilinear pairings

An ID-based proxy signature is the combination of the ID-based cryptosystem^[1] with a proxy signature scheme^[2]. Because the ID-based proxy signatures have the advantages of the ID-based cryptosystem, many ID-based proxy signature schemes have been proposed^[3–5]. However, in many of these schemes, the original signer A and the proxy signer B belong to the same closed group. That is, both of the private keys of A and B are generated by the same private key generation center (PKG). But, in practice, it is impossible for every one to share one PKG. It is true that there are many entities that are independent of any group. This means that an independent entity may use a public-private key pair based on a certificate. In many cases, a branch (or a member) of a closed group (e. g. a multinational company) (in which an ID-based cryptosystem is used) may delegate its signing power to an independent entity that uses a certificate-based cryptosystem. The aim of this paper is to define and construct a scheme to realize this kind of delegation. To construct such a scheme, we first review some basic knowledge. Then, we define and construct algorithms for our scheme, whose security is analyzed.

1 Preliminaries—Bilinear Pairings

The pairing is defined as $e: G_1 \times G_1 \rightarrow G_2$, where G_1 is an additive cyclic group of prime order q ; G_2 is a multiplicative

cyclic group of the same order and P is an arbitrary generator of G_1 . A cryptographic bilinear pairing has the following properties:

1) Bilinear: For any $R, S \in G_1$, $a, b \in \mathbb{Z}_q^*$, $e(aR, bS) = e(R, S)^{ab}$;

2) Non-degenerate: There exists $R, S \in G_1$ such that $e(R, S) \neq I$, where I denotes the identity element of the group G_2 ;

3) Computable: For any $R, S \in G_1$, there exists an efficient algorithm to compute $e(R, S)$.

Definition 1 The discrete logarithm problem (DLP) in G_1 is defined as: Given the generator P of G_1 and $Q \in G_1^*$, compute a such that $Q = aP$ for some $a \in \mathbb{Z}_q^*$.

Definition 2 The computing Diffie-Hellman problem (CDHP) is defined as: Given a generator P of a group G and a random triple (P, aP, bP) , compute abP .

Assumption We assume that DLP and CDHP are hard on G_1 , which means that there is no polynomial time algorithm to solve either of them with a non-negligible probability.

2 PSS-ID-CER

In this paper, if there is no special statement, let A denote the original signer with identity ID_A and private key D_A . A belongs to a closed group GR which has a PKG. PKG sets up the ID-based cryptosystem in the closed group. A extracts its private key from PKG through a secure channel. A will delegate its signing power to the proxy signer B, who is independent of any group and would rather use a certificate-based cryptosystem. Let (P_B, s_B) denote B's public-private key pair.

2.1 Definition of PSS-ID-CER

A PSS-ID-CER scheme consists of the algorithms as follows:

● **Setup** The system parameters generation algorithm takes a security parameter 1^k as an input, and returns a closed group GR (which consists of some users and has a PKG), a master secret key s_{PKG} (which is used as the master key for all the members in GR) and public parameters Ω . That is, $(GR, s_{PKG}, \Omega) \leftarrow \text{Setup}(1^k)$. The PKG's public key is P_{PKG} , where $P_{PKG} \in \Omega$.

● **Set_Pub_Pri_Key** The public-private key pair generation algorithm is a probabilistic algorithm that takes a security parameter 1^k and the public parameters Ω as input. It returns a public-private key pair (pk, sk) . That is, $(pk, sk) \leftarrow \text{Set_Pub_Pri_Key}(1^k, \Omega)$. An independent user B (the proxy signer) runs this algorithm to obtain its own public-private key pair (P_B, s_B) .

Received 2008-04-15.

Biography: Xin Xiangjun (1974—), male, lecturer, xin_xiang_jun@126.com.

Foundation items: The National Natural Science Foundation of China (No. 60473028), the Natural Science Foundation of Zhengzhou University of Light Industry (No. 2006XXJ18), the Doctor Foundation of Zhengzhou University of Light Industry (No. 20080014).

Citation: Xin Xiangjun, Sun Lei. Proxy signature scheme for ID-based original signers and certificate-based proxy signers[J]. Journal of Southeast University (English Edition), 2008, 24(3): 318 – 321.

- **Member_Extract** The private key generation algorithm for the members of GR is a deterministic algorithm that takes GR, an identity ID, the master key s_{PKG} and the public parameters Ω as input. If $\text{ID} \in \text{GR}$ it returns the private key D corresponding to ID. That is, $D \leftarrow \text{Member_Extract}(\text{GR}, \text{ID}, s_{\text{PKG}})$. Because the original signer A belongs to GR, he extracts his private $D_A = \text{Member_Extract}(\text{GR}, \text{ID}_A, s_{\text{PKG}})$ from PKG through a secure channel.

- **Proxy_Delegate** The proxy-designation algorithm takes the original signer A's private key D_A and a warrant m_w as input, and outputs the delegation $(\text{ID}_A, m_w, W_{A \rightarrow B})$. That is, $(\text{ID}_A, m_w, W_{A \rightarrow B}) \leftarrow \text{Proxy_Delegate}(D_A, m_w)$. This algorithm is performed by the original signer A. The output of this algorithm is sent to the proxy signer B.

- **Delegation_Verify** The designation-verification algorithm takes the delegation $(\text{ID}_A, m_w, W_{A \rightarrow B})$ as input. If the delegation is valid it returns 1, or returns 0.

- **Proxy_Key_Generate** The proxy signing key generation algorithm takes $W_{A \rightarrow B}$ and the private key of the executor (proxy signer) as input, and outputs a proxy signing key D_p . This algorithm is performed by the proxy signer B who uses the certificate-based cryptosystem.

- **Proxy_Sign** The proxy signature generation algorithm takes a proxy signing key of the D_p and a message $m \in \{0, 1\}^*$ as input, and returns a proxy signature (m, σ) . This algorithm is performed by the proxy signer B. That is, $(m, \sigma) \leftarrow \text{Proxy_Sign}(D_p, m)$.

- **Proxy_Verify** The proxy signature verification algorithm takes the original signer's identity ID, the proxy signer's public key Y and the proxy signature (m, σ) as input. If (m, σ) is a valid proxy signature it returns 1, or it returns 0. That is, 0 or 1 $\leftarrow \text{Proxy_Verify}(\text{ID}, Y, m, \sigma)$.

2.2 A construction of PSS-ID-CER

- **Setup** The system sets up a closed group GR that consists of some members. Let PKG denote the private key generation center of GR. The security parameter is 1^k . $(G_1, +)$ and (G_2, \cdot) are two cyclic groups of order q , where q is a large prime (says, 160 bits). P is a generator of G_1 . $H_1, H_3: \{0, 1\}^* \rightarrow G_1$ and $H_2: \{0, 1\}^* \times G_1 \rightarrow G_2$ are three different hash functions. $e: G_1 \times G_1 \rightarrow G_2$ is a bilinear pairings map. PKG selects $s_{\text{PKG}} \in_{\mathbb{R}} Z_q^*$ and computes $P_{\text{PKG}} = s_{\text{PKG}} P$. P_{PKG} is the public key of PKG, while s_{PKG} is used as the master key for the closed group GR. s_{PKG} is kept secretly by PKG. PKG sets $\Omega = (G_1, G_2, P, q, H_1, H_2, H_3, e, P_{\text{PKG}})$ as the public parameters. That is, $(\text{GR}, s_{\text{PKG}}, \Omega) \leftarrow \text{Setup}(1^k)$. Let A be the original signer with identity $\text{ID}_A \in \text{GR}$ in our scheme.

- **Set_Pub_Pri_Key** Let B be an independent entity that uses the certificate-based cryptosystem and plays the role of proxy signer in our scheme. B runs this algorithm and gets his own public-private key pair (P_B, s_B) , where $s_B \in_{\mathbb{R}} Z_q^*$ and $P_B = s_B P$. That is, $(P_B, s_B) \leftarrow \text{Set_Pub_Pri_Key}(1^k, \Omega)$.

- **Member_Extract** A authenticates himself to GR's PKG. PKG computes A's private key $D_A = s_{\text{PKG}} H_1(\text{ID}_A)$ and sends it to A through a secure channel.

- **Proxy_Delegate** A selects $r \in_{\mathbb{R}} Z_q^*$ and computes $R =$

$rH_1(\text{ID}_A)$, $h = H_2(m_w, R)$ and $U = (r + h)D_A$, where m_w is a warrant. Finally, A outputs delegation $(\text{ID}_A, m_w, W_{A \rightarrow B})$, where $W_{A \rightarrow B} = (R, U)$.

- **Delegation_Verify** Once B receives the delegation (ID, m_w, R, U) , he computes $h = H_2(m_w, R)$. If $e(U, P) = e(R + hH_1(\text{ID}_A), P_{\text{PKG}})$ holds, B accepts this delegation.

- **Proxy_Key_Generate** If B accepts the delegation (ID, m_w, R, U) , he uses $D_p = (\text{ID}, m_w, R, U, s_B)$ as his proxy signing key, which is kept secretly by himself.

- **Proxy_Sign** To sign a message m , the proxy signer B computes $V = U + s_B H_3(m)$ and outputs (m, m_w, R, V) as the proxy signature.

- **Proxy_Verify** (m, m_w, R, V) is a valid proxy signature iff $e(V, P) = e(R + hH_1(\text{ID}_A), P_{\text{PKG}}) e(H_3(m), P_B)$, where $h = H_2(m_w, R)$.

3 Security Model and Analysis

The correctness of our scheme can be proved easily. In the following we discuss the security of our scheme. We present the security model for our scheme as follows.

Let PSS-ID-CER be the algorithm for our scheme. Consider an adversary Ad which is assumed to be a probabilistic Turing machine that takes the public parameters Ω , the public keys and a random tape as input. Ad can adaptively query for the master key of any PKG, the private keys of the members in GR, the delegation from the original signer, the private keys of the proxy signers, the proxy signing keys and the proxy signatures on messages. We call this attack as an adaptive chosen-PKG, chosen-ID, chosen-delegation, chosen-ProxySigner-public-key, chosen-proxy-key and/or chosen-message attack (A-C-P-ID-DE-PPK-PK-M). That is, under an A-C-P-ID-DE-PPK-PK-M attack, Ad can query the oracles O_Master_Key, O_Member_Extract, O_ProxySigner_Private_Key, O_Proxy_Delegate, O_Proxy_Key_Generate, O_Proxy_Sign as follows.

Definition 3 For a PSS-ID-CER scheme, the experiment $\text{Exp}_{\text{Ad}}^{\text{PSS-ID-CER}}(k)$ for adversary Ad, challenger C and the security parameter k is defined as follows:

- 1) A challenger C runs Setup and Set_Pub_Pri_Key, and gives Ad the closed groups, the public parameters Ω and the proxy signers' public keys.

- 2) O_Master_Key This oracle takes a public key P_{PKG} as input, and returns the master key s_{PKG} of the PKG. That is, $s_{\text{PKG}} \leftarrow \text{O_Master_Key}(P_{\text{PKG}})$.

- 3) O_Member_Extract This oracle takes a closed group GR (whose private key generation center is PKG) and identities ID_i as input. If ID_i belongs to GR, it returns the private key D_i of ID_i , or refuses to answer the query. That is, $D_i \leftarrow \text{O_Member_Extract}(\text{GR}, \text{ID}_i)$.

- 4) O_ProxySigner_Private_Key This oracle takes the public key P_B of the proxy signer as input and returns the private key s_B . That is, $s_B \leftarrow \text{O_ProxySigner_Private_Key}(P_B)$.

- 5) O_Proxy_Delegate This oracle takes an original signer's identity $\text{ID}_A \in \text{GR}$ and a warrant m_w as input. If m_w is a valid warrant, it returns the delegation $(\text{ID}_A, m_w, W_{A \rightarrow B})$. That is, $(\text{ID}_A, m_w, W_{A \rightarrow B}) \leftarrow \text{O_Proxy_Delegate}(\text{ID}_A, m_w)$.

6) $O_Proxy_Key_Generate$ This oracle takes the proxy signer's public key P_B , a closed group GR and the identity $ID_A \in GR$ as input. It returns the proxy signer B's proxy signing key D_p . That is, $D_p \leftarrow O_Proxy_Key_Generate(P_B, ID_A, GR)$.

7) O_Proxy_Sign This oracle takes a message m , a warrant m_w , the independent proxy signer B's public key P_B and the original signer's identity $ID_A \in GR$ as input. It returns a proxy signature (m, m_w, σ) . That is, $(m, m_w, \sigma) \leftarrow O_Proxy_Sign(m, m_w, P_B, ID_A)$.

We use $E1, E2, E3, E4, E5$ and $E6$ to denote the events as follows: $E1$: Query $O_Master_Key(P_{PKG^*})$ has been asked; $E2$: Query $O_ProxySigner_Private_Key(P^*)$ has been asked; $E3$: Query $O_Member_Extract(GR^*, ID^*)$ has been asked; $E4$: (m^*, m_w^*, σ^*) has been output by the oracle O_Proxy_Sign ; $E5$: Query $O_Proxy_Key_Generate(P^*, ID^*, GR^*)$ has been asked; $E6$: (ID^*, m_w^*, W^*) has been output by oracle $O_Proxy_Delegate$.

Ad can query the above oracles adaptively as many times as it wants. Finally, if Ad 's output satisfies one of the following items, Ad 's attack is successful:

- Delegation forgery The outputs are P_{PKG^*} and (ID^*, m_w^*, W^*) , where P_{PKG^*} is the public key of PKG^* for the closed group GR^* , m_w^* is a valid warrant, and W^* is a valid delegation for warrant m_w^* and the original signer with identity $ID^* \in GR^*$. At the same time, P_{PKG^*} and (ID^*, m_w^*, W^*) satisfy the conditions as follows: 1) $E6$ does not occur; 2) $E1$ does not occur; 3) $E3$ does not occur. In this case, $Exp_{Ad}^{PSS-ID-CER}(k)$ returns 1.

- Proxy signature forgery The output are P_{PKG^*} and $(ID^*, P^*, m^*, m_w^*, \sigma^*)$, where P_{PKG^*} is the public key of PKG^* for the closed group GR^* ; ID^* belongs to GR^* ($ID^* \in GR^*$); (m^*, m_w^*, σ^*) is a valid proxy signature for the independent proxy signer's public key P^* . At the same time, P_{PKG^*} and $(ID^*, P^*, m^*, m_w^*, \sigma^*)$ satisfy the conditions as follows: 1) $E4$ does not occur; 2) Both $E1$ and $E2$ do not occur; 3) Both $E2$ and $E3$ do not occur; 4) Both $E2$ and $E6$ do not occur; 5) $E5$ does not occur. In this case, $Exp_{Ad}^{PSS-ID-CER}(k)$ returns 2.

Definition 4 A PSS-ID-CER signature scheme is said to be an existential delegation and an unforgeable signature under an A-C-P-ID-DE-PPK-PK-M attack, if for any polynomial time adversary Ad , any polynomial $p(\cdot)$ and a large enough k , $\Pr[Exp_{Ad}^{PSS-ID-CER}(k) = 1] < 1/p(k)$ and $\Pr[Exp_{Ad}^{PSS-ID-CER}(k) = 2] < 1/p(k)$.

Using the Forking lemma^[6] and the security proof techniques used in Refs. [6–7], theorem 1 can be proved easily.

Theorem 1 In the random oracle, if there is an algorithm Ad for an A-C-P-ID-DE-PPK-PK-M attack to our scheme which queries oracles $H_1, H_2, H_3, O_Master_Key, O_Member_Extract, O_ProxySigner_Private_Key, O_Proxy_Delegate, O_Proxy_Key_Generate$ and O_Proxy_Sign

at most $q_{H1}, q_{H2}, q_{H3}, q_{MK}, q_{ME}, q_{PPK}, q_{PD}, q_{PKG}$ and q_{PS} times, respectively, and has running time t and a probability:

$$\varepsilon \geq 10l(q_{PS} + 1)(q_{MK} + q_{ME})^2(q_{PS} + q_{H1} + q_{H2} + q_{H3}) \cdot 2^{-k} \left(1 + \frac{1}{q_{MK} + q_{ME} + 1}\right)^{-q_{ME}}$$

to succeed in $Exp_{Ad}^{PSS-ID-CER}(k)$, where l is the base of the natural logarithm, then CDHP can be solved in expected time

$$t_1 \leq 120686l(q_{H1} + q_{H2} + q_{H3})(q_{MK} + q_{ME} + 1)^2 \cdot \varepsilon^{-1} \left(1 + \frac{1}{q_{MK} + q_{ME} + 1}\right)^{-q_{ME}} \cdot [t + (q_{H1} + q_{H2} + q_{H3} + q_{MK} + q_{ME} + q_{PPK} + q_{PD} + q_{PKG} + q_{PS})g]$$

where g is a constant which depends on G_1 .

By theorem 1, we can obtain the following theorem 2.

Theorem 2 In the random oracle model, our scheme is secure against an A-C-P-ID-DE-PPK-PK-M attack.

4 Conclusion

In practice, it is usual for a member or a branch in a corporation (in which the ID-based cryptosystem is used) to delegate his/her signing power to an independent entity (who uses a certificate-based cryptosystem). Then, we present a proxy signature scheme for the ID-based original signers and certificate-based proxy signers with the corresponding security model. Our scheme is secure under the hardness assumption of CDHP.

References

- [1] Shamir A. Identity-based cryptosystems and signature schemes [C]//*Proceedings of CRYPTO'84 on Advances in Cryptology*. Berlin: Springer-Verlag, 1985: 48–53.
- [2] Mambo M, Usuda K, Okamoto E. Proxy signature: delegation of the power to sign messages [J]. *IEICE Trans Foundations*, 1996, **E79-A**(9): 1338–1353.
- [3] Zhang F, Kim K. Efficient ID-based blind signature and proxy signature from bilinear pairings [C]//*ACISP 2003*. Berlin: Springer-Verlag, 2003: 312–323.
- [4] Qian H, Cao Z. A novel ID-based partial delegation with warrant proxy signature scheme [C]//*ISPA Workshops 2005*. Berlin: Springer-Verlag, 2005: 323–331.
- [5] Xu J, Zhang Z, Feng D. ID-based proxy signature using bilinear pairings [C]//*ISPA Workshops 2005*. Berlin: Springer-Verlag, 2005: 359–367.
- [6] Cha Jae Choon, Cheon Jung Hee. An identity-based signature from gap Diffie-Hellman groups [C]//*Proceedings of the 6th International Workshop on Theory and Practice in Public Key Cryptography: Public Key Cryptography*. Springer-Verlag, 2003: 18–30.
- [7] Boneh D, Lynn B, Shacham H. Short signatures from the Weil pairing [C]//*Proceedings of the 7th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology*. Springer-Verlag, 2001: 514–532.

一种基于身份原始签名者和基于证书代理签名者的代理签名体制

辛向军¹ 孙 垒²

(¹ 郑州轻工业学院数学与信息科学系, 郑州 450002)

(² 河南理工大学数学与信息科学学院, 焦作 454000)

摘要:为实现混合密码体制下不同用户之间的代理权,给出了基于身份的原始签名者和基于证书的代理签名者的代理签名体制(PSS-ID-CER)的定义,并利用双线性对的双线性及基于公钥证书的密码体制的密钥和基于身份的密码体制的密钥生成特点,给出了相应方案的构造.为证明方案的安全性,定义了此类方案在适应性选择PKG、ID、代理、代理签名者公钥、代理密钥和消息攻击下的一般化安全模型.在随机预言机模型和计算 Diffie-Hellman 问题困难假下,方案是可证明安全的.

关键词:代理签名;基于身份的密码系统;双线性对

中图分类号:TP309