Identity Based Threshold Proxy Signature

Jing Xu, Zhenfeng Zhang, and Dengguo Feng

{xujing, zfzhang, feng}@is.iscas.ac.cn

Abstract. Identity-based (ID-based) public key cryptosystem can be a good alternative for certificate-based public key setting, especially when efficient key management and moderate security are required. In a (t, n) threshold proxy signature scheme, the original signer delegates the power of signing messages to a designated proxy group of n members. Any t or more proxy signers of the group can cooperatively issue a proxy signature on behalf of the original signer, but $t-1$ or less proxy signers cannot. In this paper, we present an ID-based threshold proxy signature scheme using bilinear pairings. We show the scheme satisfies all security requirements in the random oracle model. To the best of authors' knowledge, our scheme is the first ID-based threshold proxy signature scheme.

keywords: ID-based signatures, threshold proxy signatures, bilinear pairings.

1 Introduction

Proxy signatures are first introduced by Mambo, Usuda, and Okamoto in [1]. Such a scheme allows one user, called original signer, to delegate his/her signing capability to another user, called proxy signer. After that, the proxy signer can sign messages on behalf of the original signer. Upon receiving a proxy signature on some message, a verifier can validate its correctness by following a given verification procedure, and then is convinced of the original signer' agreement on the signed message if the validation is positive. Proxy signature schemes have been suggested for use in a number of applications, including e-cash systems, mobile agents, mobile communications, grid computing, global distribution networks, and distributed shared object systems etc.

Based on the ideas of secret sharing [2][3][4] and threshold cryptosystems , Zhang and Kim et al. independently constructed the first threshold proxy signatures in [5] and [6], respectively. In a (t, n) threshold proxy signature scheme, the original signer' signing power is delegated to a group of n proxy singers such that t or more of them can generate proxy signatures cooperatively, but $t-1$ or less of them cannot do the same thing. This technology not only allows the original signer to delegate the proxy signing power to a group of proxy signers instead of one single proxy signer, but also lets the original signer to set the threshold value t freely $(1 \leq t \leq n)$. Therefore, the threshold proxy signature approach is more practical, flexible and secure than standard proxy signature schemes.

In traditional public key infrastructure (PKI), a user must pre-enroll the PKI or he/she cannot enjoy the cryptographic services provided by the PKI, e.g. no one can send them any encrypted message. Identity-based (ID-based) cryptography [8] solves this problem: all users already have their corresponding public key before their enrollment since the public key can be derived via a public algorithm with input of a string that can uniquely identify each of them, such as an email address.

All previous threshold proxy signature constructions are non ID-based: the public key of each member of the group is required to be published by the underlying public key infrastructure before it can be used to generate the signature. Removing this pre-requisite requirement motivates the construction of our ID-based threshold proxy signature scheme, which provide a better alternative than non-ID based solutions.

The bilinear pairings, namely the weil-pairing and the tate-pairing of algebraic curves, are important tools for research on algebraic geometry. They have been found various applications in cryptography recently [9],[10],[11],[12]. More precisely, they can be used to construct ID-based cryptographic schemes.

To our knowledge, ID-based threshold proxy signature has not been treated in the literature. Our current work is aimed at filling this void. In this paper, we present an ID-based threshold proxy signature scheme using bilinear pairings. Our scheme satisfies proxy security requirements and uses simple Lagrange formula to share the proxy secret.

The rest of the paper is organized as follows. The next section contains some preliminaries about the formal definitions of bilinear pairing, Gap Diffie-Hellman group as well as ID-based threshold proxy signature scheme. In Sections 3 and 4, we present an ID-based threshold proxy signature scheme and analyze its security , respectively. And we end with concluding remarks in Section 5.

2 Definitions

2.1 The Bilinear Pairing

Let G be a cyclic additive group generated by P, whose order is a prime q, and V be a cyclic multiplicative group of the same order. Let $\hat{e} : G \times G \to V$ be a pairing which satisfies the following conditions:

1. Bilinearity: For any $P, Q, R \in G$, we have $\hat{e}(P + Q, R) = \hat{e}(P, R)\hat{e}(Q, R)$ and $\hat{e}(P, Q + R) = \hat{e}(P, Q)\hat{e}(P, R)$. In particular, for any $a, b \in \mathbf{Z}_q$,

$$\hat{e}(aP, bP) = \hat{e}(P, P)^{ab} = \hat{e}(P, abP) = \hat{e}(abP, P).$$

2. Non-degeneracy: There exists $P, Q \in G$, such that $\hat{e}(P, Q) \neq 1$.

3. Computability: There is an efficient algorithm to compute $\hat{e}(P, Q)$ for all $P, Q \in G$.

The typical way of obtaining such pairings is by deriving them from the weil-pairing or the tate-pairing on an elliptic curve over a finite field.

2.2 Gap Diffie-Hellman (GDH) Groups

Let G be a cyclic group of prime order q and P be a generator of G.

1. The decisional Diffie-Hellman (DDH) problem is to decide whether $c = ab$ in Z/qZ for given $P, aP, bP, cP \in G$. If so, (P, aP, bP, cP) is called a valid Diffie-Hellman (DH) tuple.

2. The computational Diffie-Hellman (CDH) problem is to compute abP for given $P, aP, bP \in G$.

Definition 2.1 The advantage of an algorithm $\mathcal{F}$ in solving the computational Diffie-Hellman problem on group G is

$$AdvCDH_{\mathcal{F}} = Pr[\mathcal{F}(P, aP, bP) = abP : \forall a, b \in Z_q]$$

The probability is taken over the choice of a, b and $\mathcal{F}'$s coin tosses. An algorithm $\mathcal{F}$ is said (t, ε)-breaks the computational Diffie-Hellman problem on G if $\mathcal{F}$ runs in time at most t, and $AdvCDH_{\mathcal{F}}$ is at least ε.

Now we present a definition for a gap Diffie-Hellman (GDH) group.

Definition 2.2 A group G is a (t, ε)-gap Diffie-Hellman (GDH) group if the decisional Diffie-Hellman problem in G can be efficiently computable and there exists no algorithm (t, ε)-breaks computational Diffie-Hellman on G.

If we have an admissible bilinear pairing $\hat{e}$ in G, we can solve the DDH problem in G efficiently as follows:

(P, aP, bP, cP) is a valid DH tuple $\Leftrightarrow \hat{e}(aP, bP) = \hat{e}(P, cP)$

Hence an elliptic curve becomes an instance of a GDH group if the Weil (or the Tate) pairing is efficiently computable and the CDH is sufficiently hard on the curve.

2.3 ID-Based Setting from Bilinear Pairings

The ID-based public key systems allow some public information of the user such as name, address and email *etc.*, rather than an arbitrary string to be used as his public key. The private key of the user is calculated by a trusted party, called PKG and sent to the user via a secure channel.

ID-based public key setting from bilinear pairings can be implemented as follows:

Let G be a cyclic additive group generated by P, whose order is a prime q, and V be a cyclic multiplicative group of the same order.A bilinear pairing is the map $\hat{e} : G \times G \to V$. Define cryptographic hash function $H : \{0, 1\}^* \to G$.

– $\mathcal{G}$**:** PKG chooses a random number $s \in Z_q^*$ and sets $P_{pub} = sP$. He publishes system parameters $params = \{G, V, \hat{e}, q, P, P_{pub}, H\}$; and keeps s secretly as the *master-key*.

– $\mathcal{K}$**:** A user submits his/her identity information ID and authenticates him to PKG. PKG computes the user′s private key $d_{ID} = sQ_{ID} = sH(ID)$ and sends it to the user via a secure channel.

2.4 Security Requirements of ID-Based Threshold Proxy Signature

Like the general threshold proxy signature, an ID-based threshold proxy signature scheme should satisfy the following requirements [13][14]:

- **Distinguishability:** Proxy signatures are distinguishable from normal signatures by everyone.
- **Secrecy:** The original signer's private key cannot be derived from any information, such as the shares of the proxy signing key, proxy signatures etc. Particularly, even all proxy signers collude together, they cannot derive the original signer's private key.
- **Proxy Protected:** Only the delegated proxy signer can generate valid partial proxy signatures. Even the original signer cannot masquerade as a proxy signer to create partial signatures.
- **Unforgeability:** A valid proxy signature can only be cooperatively generated by t or more proxy signers. This means that valid proxy signatures cannot be created by $(t-1)$ or less proxy signers, or any third parties who are not designated as proxy signers.
- **Nonrepudiation:** Any valid proxy signature must be generated by t or more proxy signers. Therefore, proxy signers cannot deny that they have signed the message. In addition, the original signer cannot deny having delegated the power of signing messages to the proxy signers.

3 Our (t, n) threshold Proxy Signature Scheme

Our proxy signature scheme is based on SOK-IBS (Sakai-Ogishi-Kasahara Identity Based Signature)[15]. It can be divided into six stages: *Param-generation*, *Key-generation*, *Secret-share-generation*, *Proxy-share-generation*, *Proxy-signature -generation* and *Proxy-signature-verification*.

- ***Param-generation*:** Assume k is a security parameter. G is a GDH group of prime order $q > 2^k$ generated by P, and $\hat{e} : G \times G \rightarrow V$ is a bilinear map. Pick a random master key $s \in Z_q^*$ and set $P_{pub} = sP$. Choose hash functions $H_1, H_2 : \{0,1\}^* \rightarrow G$. Let P_0 be the original signer and $PS = \{P_1, P_2, \cdots, P_n\}$ be the proxy group of n proxy signers. Each user P_i owns a secret key $d_i \in G$.
- ***Key-generation*:** Given a users identity ID, compute $Q_{ID} = H_1(ID) \in G$ and the associated private key $d_{ID} = sQ_{ID} \in G$.
- ***Secret-share-generation*:** The proxy group apply a (t, n) verifiable secret sharing scheme to generate secret shares for all proxy signers in PS as follows. Each $P_i \in PS$ randomly chooses a $(t-1)$-degree polynomial

$$f_i(x) = \sum_{l=1}^{t-1} a_{il} x^l + a_{i0} \tag{1}$$

with random coefficients $a_{il} \in Z_q^*$ and publishes $A_{il} = a_{il}P$ for $l = 0, 1 \cdots t-1$. Furthermore, P_i sends $f_i(j)$ to P_j via a secure channel for $j \neq i$. On receiving $f_i(j)$, P_j can validate it by checking the equality

$$f_i(j)P = \sum_{k=0}^{t-1} j^k \cdot A_{ik}.$$

Finally, each P_i computes his secret share $r_i = \sum_{k=1}^{n} f_k(i)$ and publishes $U_i = r_i P$.

– ***Proxy-share-generation*:** Let m_ω be the warrant consisting the identity of the original signer and the proxy signers of proxy group, the threshold parameter t, and the valid delegation time, etc. Every proxy signer P_i in PS gets their own proxy signing key share as follows:

1. The original signer P_0 first randomly chooses $r_\omega \in Z_q^*$ and computes $U_\omega = r_\omega P$. Let $H_\omega = H_2(ID_0, m_\omega, U_\omega)$. Then, he computes $V_\omega = d_0 + r_\omega H_\omega$. The signature on m_ω is $\omega = \langle U_\omega, V_\omega \rangle$. Finally, P_0 sends ω and m_ω to each $P_i \in PS$.

2. To verify a signature $\omega = \langle U_\omega, V_\omega \rangle$ on a message m_ω for an identity ID_0, the proxy signer P_i first takes $Q_0 = H_1(ID_0) \in G$ and $H_\omega = H_2(ID_0, m_\omega, U_\omega) \in G$. He then accepts the signature if

$$\hat{e}(P, V_\omega) = \hat{e}(P_{pub}, Q_0)\hat{e}(U_\omega, H_\omega) \tag{2}$$

and rejects it otherwise. If the signature ω is accepted, P_i computes $s_i = d_i + \frac{1}{n}V_\omega$ as his own proxy secret.

3. P_i randomly chooses a $(t-1)$-degree polynomial

$$g_i(x) = \sum_{l=1}^{t-1} b_{il}x^l + s_i \tag{3}$$

with random coefficients $b_{il} \in G$ and publishes $B_{il} = \hat{e}(P, b_{il})$ for $l = 1, 2 \cdots t-1$. B_{i0} can be got by each proxy signer as

$$\hat{e}(P, s_i) = \hat{e}(P_{pub}, H_1(ID_i))\hat{e}(P_{pub}, \frac{1}{n}Q_0)\hat{e}(U_\omega, \frac{1}{n}H_\omega).$$

Furthermore, P_i sends $g_i(j)$ to P_j via a secure channel for $j \neq i$.

4. On receiving $f_j(i)$, P_i can validate it by checking the equality

$$\hat{e}(P, g_j(i)) = \prod_{k=0}^{t-1} B_{jk}^{i^k}.$$

Finally, P_i computes his proxy signing key share $skp_i = \sum_{k=1}^{n} g_k(i)$ and publishes $\hat{e}(P, skp_i)$.

– ***Proxy-signature-generation*:** Let $D = \{P_1, P_2, \cdots, P_t\}$ be the actual proxy signers who want to sign m on behalf of the original signer P_0.

1. Apply the Lagrange interpolation formula to compute $U = \sum_{i=1}^{t} \eta_i U_i$. Here $\eta_i = \prod_{j \neq i}^{j \in \{1,2,\cdots,t\}} \frac{j}{j-i}$. Let $H = H_2(m, U)$.

2. Each $P_i \in D$ computes $V_i = skp_i + r_i H$ and $\sigma_i = (U_i, V_i)$ be his own proxy signature share.

3. Upon receiving σ_i, the designated clerk validates it by checking

$$\hat{e}(P, V_i) = \hat{e}(P, skp_i)\hat{e}(U_i, H).$$

If it holds, then σ_i is the valid individual proxy signature of m. If all individual proxy signatures for m are valid, then the clerk computes

$$V = \sum_{i=1}^{t} \eta_i V_i.$$

The proxy signature of m is $\langle m, U_\omega, m_\omega, (U, V)\rangle$.

– ***Proxy-signature-verification*:** To verify a proxy signature $\langle m, U_\omega, m_\omega, (U, V)\rangle$ for message m with the original designator's identity ID_0, the verifier first takes $Q_i = H_1(ID_i) \in G (i = 0, 1, \cdots, n)$, $H_\omega = H_2(ID_0, m_\omega, U_\omega)$ and $H = H_2(m, U) \in G$. He then accepts the signature if

$$\hat{e}(P, V) = \hat{e}(P_{pub}, \sum_{i=0}^{n} Q_i)\hat{e}(U, H)\hat{e}(U_\omega, H_\omega) \tag{4}$$

and rejects it otherwise.

4 Analysis of Our Scheme

4.1 Correctness

In the following, we show that the proposed scheme works correctly.

Theorem 4.1. *The proxy signers can verify the validity of* $\omega = (U_\omega, V_\omega)$ *sent by the original signer* P_0 *by Eq.(2).*

Proof. Eq.(2) is correct because of the following.

$$\begin{aligned} & \hat{e}(P, V_\omega) \\ = & \hat{e}(P, d_0 + r_\omega H_\omega) \\ = & \hat{e}(P, d_0)\hat{e}(P, r_\omega H_\omega) \\ = & \hat{e}(P_{pub}, Q_0)\hat{e}(U_\omega, H_\omega). \end{aligned}$$

Theorem 4.2. *If the proxy signature is constructed correctly, it will pass the verification of Eq.(4).*

Proof. Eq.(4) is correct because of the following.

$$\begin{aligned}
&\hat{e}(P, V) \\
&= \hat{e}(P, \sum_{i=1}^{t} \eta_i V_i) \\
&= \hat{e}(P, \sum_{i=1}^{t} \eta_i skp_i)\hat{e}(P, \sum_{i=1}^{t} \eta_i r_i H) \\
&= \hat{e}(P, \sum_{k=1}^{n}\sum_{i=1}^{t} \eta_i g_k(i))\hat{e}(P, \sum_{i=1}^{t} \eta_i r_i H) \\
&= \hat{e}(P, \sum_{k=1}^{n} g_k(0))\hat{e}(\sum_{i=1}^{t} \eta_i U_i, H) \\
&= \hat{e}(P, V_\omega + \sum_{k=1}^{n} d_k)\hat{e}(\sum_{i=1}^{t} \eta_i U_i, H) \\
&= \hat{e}(P_{pub}, \sum_{i=0}^{n} Q_i)\hat{e}(U, H)\hat{e}(U_\omega, H_\omega).
\end{aligned}$$

4.2 Security

We will show that our ID-based (n, t) threshold proxy signature scheme satisfies all the requirements stated in Section 2.

- **Distinguishability:** This is obvious, because there is a warrant m_ω in a valid proxy signature, at the same time, this warrant m_ω and the public keys of the original signer and the proxy signers must occur in the verification equation of threshold proxy signature.
- **Secrecy:** Our scheme is ID-based signature scheme, so any party's private key d_i must be kept secret. From the identity, no one can derive the corresponding private key except PKG. Furthermore, we also cannot compute the original signer's private key from the signature ω and the warrant m_ω since SOK-IBS (Sakai-Ogishi-Kasahara Identity Based Signature)[15] is secure. Even if t out of n proxy signers collaborate to deliver the proxy share, they can not calculate the original signer's private key d_0. Therefore, in our scheme, the original signer's private key can always be kept secret and used repeatedly.
- **Proxy Protected:** In our scheme, the original signer cannot generate a valid signature share on behalf of P_i since the original signer does not know P_i's private key d_i. The designated combiner does not accept the partial

proxy signature σ_i when the received signature is incorrect. Therefore, the original signer has no ability to substitute for proxy signers.

- **Unforgeability:** Each member P_i only knows his own secret share skp_i and r_i. Therefore, only the designated group of n proxy signers can sign messages. Without the private key d_i, no one can forge the proxy signer P_i to create σ_i and pass the verification. Furthermore, in the Lagrange interpolation polynomial, $t-1$ or less proxy signers cannot reconstruct secrets. Our scheme is based on SOK-IBS (Sakai-Ogishi-Kasahara Identity Based Signature)[15] whose security is tightly related to Computational Diffie-Hellman (CDH) problem in the Random Oracle model. Therefore, our scheme ensures that a valid proxy signature can be generated only when t or more proxy signers cooperatively sign the message.
- **Nonrepudiation:** In the property of nonrepudiation, both the original signer and proxy signers cannot deny having signed the proxy signature. As the valid proxy signature contains the warrant m_ω, which must be verified in the verification phase, it cannot be modified by the proxy signer. Thus once proxy signers create valid proxy signatures of an original signer, he cannot repudiate the signatures creation. Furthermore, the verifier must use the identity of proxy signers from Eq.(4), so proxy signers cannot deny having signed the proxy signature.

5 Conclusion

In this paper we proposed an ID-based threshold proxy signature scheme from bilinear pairings. We prove the security of our scheme in the random oracle model. To the best of authors' knowledge, our scheme is the first ID-based threshold proxy signature scheme. Due to the elegancy of bilinear pairing, signatures generated by our scheme are short and simple.

References

1. M. Mambo, K. Usuda, and E. Okamoto. Proxy signatures for delegating signing operation. In Proceedings of the 3rd ACM Conference on Computer and Communications Security (CCS), 48C57. ACM, 1996.
2. A. Shamir, How to Share a Secret, Communications of the ACM, vol. 22, no. 11, 612-613, 1979.
3. T. P. Pedersen, Distributed Provers with Applications to Undeniable Signatures, Advance in Cryptology - EUROCRYPT91, LNCS 547, Springer-Verlag, 221-242, 1991.
4. R. Gennaro, S. Jarecki, H. Krawczyk and T. Rabin. Secure distributed key generation for discrete-log based cryptosystems, Advance in Cryptology - EUROCRYPT99, LNCS 1592, Springer-Verlag, 295-310, 1999.
5. K. Zhang, Threshold Proxy Signature Schemes, Proc. Information Security Workshop (ISW97), LNCS 1396, Springer-Verlag, 282-290, 1997.
6. S. Kim, S. Park, and D. Won, Proxy Signatures, Revisited, Proc. Information and Communications Security (ICICS97), LNCS 1334, Springer-Verlag, 223-232, 1997.

7. J. Herranz and G. Sez. Verifiable secret sharing for general access structures, with application to fully distributed proxy signatures. In Proceedings of Financial Cryptography 2003, LNCS. Springer-Verlag, 2003.
8. A. Shamir, Identity-based cryptosystems and signature schemes, Advances in Cryptology-Crypto 1984, LNCS 196, 47-53, Springer-Verlag, 1984.
9. D. Boneh and M. Franklin, Identity-based encryption from the Weil pairing, Advances in Cryptology-Crypto 2001, LNCS 2139, 213-229, Springer-Verlag, 2001.
10. D. Boneh, B. Lynn, and H. Shacham, Short signatures from the Weil pairing, Advances in Cryptology-Asiacrypt 2001, LNCS 2248, 514-532, Springer-Verlag, 2001.
11. A. Joux, The Weil and Tate Pairings as Building Blocks for Public Key Cryptosystems, ANTS 2002, LNCS 2369, 20-32, Springer-Verlag, 2002.
12. D.Boneh and X.Boyen, Short Signatures Without Random Oracles. Eurocrypt 2004, LNCS 3027, 56-73, Springer-Verlag, 2004.
13. M.-S. Hwang, E. J.-L. Lu, and I.-C. Lin, A Practical (t, n) Threshold Proxy Signature Scheme Based on the RSA Cryptosystem. IEEE Trans. Knowledge and Data Engineering, vol. 15, no. 6, 1552-1560, 2003.
14. J.Y. Lee, J.H. Cheon and S. Kim, An analysis of proxy signatures: Is a secure channel necessary? CT-RSA 2003, LNCS 2612, 68-79, Springer-Verlag, 2003.
15. M.Bellare, C.Namprempre and G.Neven. Security Proofs for Identity-Based Identification and Signature Schemes, Advances in Cryptology-Eurocrypt 2004,LNCS 3027,268-286,Springer-Verlag, 2004.
16. B.Libert, and J.J.Quisquater. The Exact Security of an Identity Based Signature and Its Applications. Available from http://eprint.iacr.org/2004/102.