Formalized Security Model of Identity-Based Multi-Proxy Signature

[1]Wang Tao and [2]Li Mingxiang

[1]Department of Information Management and Engineering
Hebei Finance University Baoding, Hebei Province, China

[2]Department of Information Management and Engineering
Hebei Finance University Baoding, Hebei Province, China

Abstract

Multi-proxy signature is an extension of the primitive basic proxy signature.. In a multi-proxy signature scheme, an original signer could delegate his signing rights to a group of proxy signers. And only the cooperation of all the proxy signers can generate the proxy signature on behalf of the original signer. Combining multi-proxy signature with identity-based cryptography, some identity-based multi-proxy signature schemes have been proposed. But to date, no formalized security model has been provided for identity-based multi-proxy signature schemes. In this paper, the authors give the formal definition of identity-based multi-proxy signature schemes and formalize a notion of security for them.

Keywords: *multi-proxy signature; identity-based signature; identity-based multi-proxy signature; security model*

1. Introduction

Mambo, Usuda and Okamoto [1] introduced the concept of proxy signature schemes in 1996. In a proxy signature scheme, an entity called original signer can delegate his signing rights to another entity called proxy signer, and the proxy signer can sign message on behalf of the original signer. Hwang and Shi [2] proposed the concept of multi-proxy signature schemes in 2000. In a multi-proxy signature scheme, an original signer can authorize a group of proxy signers, and only the cooperation of all the proxy signers can generate the proxy signatures on behalf of the original signer. It can be regarded as a special case of a (t,n) threshold proxy signature scheme for $t = n$.

Boldyreva, Palacio and Warinschi [3] provided a formal security model for proxy signatures in 2003. In the model of Boldyreva et al., an adversary is not allowed to gain access to any proxy keys. Malklin, Obana and Yung [4] developed a formal security model for multi-level proxy signatures in 2004. In the model of Malklin et al., an adversary is allowed to access to proxy keys which a user has obtained by self-delegation. Schuldt, Matsuura and Paterson [5] proposed an enhanced security model for multi-level proxy signatures in 2008. In the model of Schuldt et al., an adversary is allowed to gain access to any proxy keys. Obviously, the model of Schuldt et al. captures a more realistic set of attacks than the models of Boldyreva et al. and Malklin et al.

In 1984, Shamir [6] initially proposed the idea of identity-based cryptography to simplify certificate management. In 2001, Boneh and Franklin [7] proposed the first efficient identity-based encryption scheme. Since then, the identity-based cryptography has become a hotspot. Several identity-based multi-proxy signature (IBMPS) schemes [8, 9] have been proposed. But the security model for IBMPS has not been presented. Therefore, the IBMPS schemes [8, 9] are not provably secure. In this paper, we propose a formal definition of IBMPS schemes, and put forward a formal security notion for IBMPS schemes. Obviously, our contribution is necessary for constructing provably secure IBMPS schemes in the future.

2. Formal definition of IBMPS

An identity-based multi-proxy signature scheme is an extension of an ordinary identity-based signature (IBS) scheme. Let $\mathrm{IBS} = \{\mathrm{G}, \mathrm{E}, \mathrm{S}, \mathrm{V}\}$, then

$$\mathrm{IBMPS} = \{\mathrm{G}, \mathrm{E}, \mathrm{S}, \mathrm{V}, (\mathrm{D}, \mathrm{P}_1, \mathrm{P}_2, \mathrm{L}\ , \mathrm{P}_n), \mathrm{MPS}, \mathrm{MPV}\}.$$

G which on inputs a security parameter 1^k which generates the public parameters $params$ and the master secret msk . We assume that $params$ are made publicly available and will not write $params$ as an explicit argument to the functions defined below.

E which on input (msk, ID), where ID is a user's identity, generates the private key sk_{ID} of ID.

S which on input (sk_{ID}, m), where m is a message to be signed, generates the signature s of ID on m.

V which on input (ID, m, s) generates either accept or reject.

$(\mathrm{D}, \mathrm{P}_1, \mathrm{P}_2, \cdots, \mathrm{P}_n)$ is a delegation protocol, where D is owned by the original signer ID_A, and P_1, P_2, $\cdots$, and P_n are owned by the proxy signers ID_{B_1}, ID_{B_2}, $\cdots$, and ID_{B_n}, respectively.

D takes as input $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, sk_{\mathrm{ID}_A})$, where w is the delegation warrant. D will interact with P_1, P_2, $\cdots$, and P_n to perform the delegation, but will have no local output.

$\mathrm{P}_i\,(i \in \{1, 2, \cdots, n\})$ takes as input $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, sk_{\mathrm{ID}_{B_i}})$. Upon completion of the interaction with D, P_i returns the local output $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, psk_i)$, where psk_i is a partial proxy signing key.

MPS is the proxy signing algorithm. Each proxy signer $\mathrm{ID}_{B_i}\,(i \in \{1, 2, \cdots, n\})$ takes input $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, psk_i, m)$, then outputs $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, ps_i)$, where ps_i is the partial proxy signature, and $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, ps_i)$ is sent to the clerk. In accordance with $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, ps_i)(i = 1, 2, \cdots, n)$, the clerk outputs $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, ps)$, where ps is the multi-proxy signature.

MPV is the proxy verification algorithm. Any verifier takes input $(m, (\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, ps))$, and outputs either accept or reject.

We say that an IBMPS scheme is sound if it satisfies the following two conditions.

1) The basic IBS scheme is sound;

2) For private key sk_{ID_A} generated by $\mathrm{E}(msk, \mathrm{ID}_A)$, private keys $sk_{\mathrm{ID}_{B_i}}\,(i = 1, 2, \cdots, n)$ generated by $\mathrm{E}(msk, \mathrm{ID}_{B_i})(i = 1, 2, \cdots, n)$, proxy keys $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, psk_i)(i = 1, 2, \cdots, n)$ generated by $\mathrm{D}(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, sk_{\mathrm{ID}_A})$ and $\mathrm{P}_i(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, sk_{\mathrm{ID}_{B_i}})(i = 1, 2, \cdots, n)$, and all message m satisfying the warrant w, we have that

$$\Pr\left[\mathrm{MPV}\left(m, \mathrm{MPS}\left(\left(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, psk_i, m\right)(i = 1, 2, \cdots, n)\right)\right) = \mathrm{Accept}\right] = 1$$

where the above probability is taken over all random coins used by E, D, $\mathrm{P}_i\,(i = 1, 2, \cdots, n)$ and MPS algorithms.

3. Security model of IBMPS

We define existential unforgeability under an adaptive chosen message and identity attack (UF-CMIA) for IBMPS schemes. The security notion is based on the security game defined below, played between a challenger C and an adversary A.

Setup. The challenger C runs G with input 1^k and generates both the public parameters $params$ and the master secret msk. The adversary A is given $params$, but msk is kept by the challenger C.

Queries. A adaptively makes a number of different queries for C. Each query can be one of the following items.

1) Extract query. On input ID from A, C runs $\mathrm{E}(msk, \mathrm{ID})$ to obtain sk_{ID}, and forwards sk_{ID} to A.

2) Standard Sign query. On input (ID, m) from A , C runs $\mathrm{E}(msk, \mathrm{ID})$ to obtain sk_{ID} , then runs $\mathrm{S}(sk_{\mathrm{ID}}, m)$ to obtain s , which is forwarded to A .

3) Delegate query. We define an oracle for each of the three different types of delegation the adversary can request the challenger to perform.

① A outputs $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\})$, and C interacts with A through the delegation protocol by playing the role of proxy signer ID_{B_i} $(i \in \{1, 2, \cdots, n\})$. C runs $\mathrm{E}(msk, \mathrm{ID}_{B_i})$ to obtain $sk_{\mathrm{ID}_{B_i}}$, and runs $\mathrm{P}_i(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, sk_{\mathrm{ID}_{B_i}})$ to obtain $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, psk_i)$, which is forwarded to A . When the delegation is complete, C adds (ID_{B_i}, w) to *pskList* list.

② A outputs $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w)$, and C interacts with A through the delegation protocol by playing the role of original signer ID_A . C runs $\mathrm{E}(msk, \mathrm{ID}_A)$ to obtain sk_{ID_A} , and runs $\mathrm{D}(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, sk_{\mathrm{ID}_A})$. Upon completion of the delegation protocol, C adds (ID_A, w) to *delList* list.

③ A outputs $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w)$, where $\mathrm{ID}_A = \mathrm{ID}_{B_i}$ $(i \in \{1, 2, \cdots, n\})$, and C interacts with A through the delegation protocol by playing the role of original signer ID_A and proxy signer ID_{B_i} . C runs $\mathrm{E}(msk, \mathrm{ID}_{B_i})$ to obtain $sk_{\mathrm{ID}_{B_i}}$, and runs $\mathrm{D}(\mathrm{ID}_{B_i}, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, sk_{\mathrm{ID}_{B_i}})$ and $\mathrm{P}_i(\mathrm{ID}_{B_i}, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, sk_{\mathrm{ID}_{B_i}})$ to obtain $(\mathrm{ID}_{B_i}, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, psk_i)$, which is forwarded to A . When the delegation is complete, C adds (ID_{B_i}, w) to *pskList* list.

4) Proxy Sign query. There are two types of the multi-proxy signatures in accordance with the types of delegation. We define an oracle for each of them.

① A outputs $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, m)$, and C plays the role of proxy signer ID_{B_i} $(i \in \{1, 2, \cdots, n\})$. C runs $\mathrm{E}(msk, \mathrm{ID}_{B_i})$ to obtain $sk_{\mathrm{ID}_{B_i}}$, and C runs $\mathrm{P}_i(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, sk_{\mathrm{ID}_{B_i}})$ to interact with A . Upon completion, C will obtain $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, psk_i)$. Then C runs $\mathrm{MPS}(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, psk_i, m)$, and obtains $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, ps_i)$, which is forwarded to A . And C adds $(\mathrm{ID}_{B_i}, w, m)$ to *mpsList* list.

② A outputs $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w)$, where $\mathrm{ID}_A = \mathrm{ID}_{B_i}$ $(i \in \{1, 2, \cdots, n\})$, and C plays the role of original signer ID_A and proxy signer ID_{B_i} . C runs $\mathrm{E}(msk, \mathrm{ID}_{B_i})$ to obtain $sk_{\mathrm{ID}_{B_i}}$, and C runs $\mathrm{D}(\mathrm{ID}_{B_i}, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, sk_{\mathrm{ID}_{B_i}})$ and $\mathrm{P}_i(\mathrm{ID}_{B_i}, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, sk_{\mathrm{ID}_{B_i}})$ to interact with A . Upon completion, C will obtain $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, psk_i)$. Then C runs $\mathrm{MPS}(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, psk_i, m)$, and obtains $(\mathrm{ID}_A, \{\mathrm{ID}_{B_1}, \mathrm{ID}_{B_2}, \cdots, \mathrm{ID}_{B_n}\}, w, ps_i)$, which is forwarded to A . And C adds $(\mathrm{ID}_{B_i}, w, m)$ to *mpsList* list.

Forgery. The adversary $\mathcal{A}$ outputs a forgery and halts. The forgery can be one of the following forms.

1) $\mathcal{A}$ outputs $(m^*, \mathrm{ID}^*, s^*)$. This forgery is said to be valid if the following hold true.

① $\mathrm{V}(m^*, \mathrm{ID}^*, s^*) = \text{Accept}$;

② $\mathcal{A}$ has not made an extract query on ID^*;

③ $\mathcal{A}$ has not made a StandardSign query on (ID^*, m^*).

2) $\mathcal{A}$ outputs $\left(\mathrm{ID}_A^*, \{\mathrm{ID}_{B_1}^*, \mathrm{ID}_{B_2}^*, \ldots, \mathrm{ID}_{B_n}^*\}, w^*, ps^*\right)$. This forgery is said to be valid if the following hold true.

① $\mathrm{MPV}\left(m^*, \left(\mathrm{ID}_A^*, \{\mathrm{ID}_{B_1}^*, \ldots, \mathrm{ID}_{B_n}^*\}, w^*, ps^*\right)\right) = \text{Accept}$;

② $\mathcal{A}$ has not made an extract query on $\mathrm{ID}_{B_i}^*\ (i \in \{1, 2, \ldots, n\})$;

③ $(\mathrm{ID}_{B_i}^*, w^*) \notin pskList$;

④ $(\mathrm{ID}_{B_i}^*, w^*, m^*) \notin mpsList$.

3) $\mathcal{A}$ outputs $\left(\mathrm{ID}_A^*, \{\mathrm{ID}_{B_1}^*, \mathrm{ID}_{B_2}^*, \ldots, \mathrm{ID}_{B_n}^*\}, w^*, ps^*\right)$. This forgery is said to be valid if the following hold true.

① $\mathrm{MPV}\left(m^*, \left(\mathrm{ID}_A^*, \{\mathrm{ID}_{B_1}^*, \ldots, \mathrm{ID}_{B_n}^*\}, w^*, ps^*\right)\right) = \text{Accept}$;

② $\mathcal{A}$ has not made an extract query on ID_A^*;

③ $\mathrm{ID}_A^* \neq \mathrm{ID}_{B_i}^*\ (i \in \{1, 2, \ldots, n\})$;

④ $(\mathrm{ID}_A^*, w^*) \notin delList$.

If the forgery output is made by the adversary is valid, we say $\mathcal{A}$ succeeds. Otherwise, we say $\mathcal{A}$ fails.

The advantage of an adversary $\mathcal{A}$ in the above game is defined to be $\mathrm{Adv}_{\mathcal{A}} = \Pr[\mathcal{A}\ \text{succeeds}]$ where the probability is taken over all random coins tosses made by the adversary and the challenger.

An adversary $\mathcal{A}$ is said to be an $(\varepsilon, t, q_e, q_d, q_s)$-forger of an IBMPS scheme if $\mathcal{A}$ has advantage at least ε in the above game, runs in time at most t, and makes at most q_e, q_d and q_s extract, delegate and sign queries, respectively. An IBMPS scheme is said to be $(\varepsilon, t, q_e, q_d, q_s)$-secure if no $(\varepsilon, t, q_e, q_d, q_s)$-forger exists.

It can be seen that in the forgery case 2, the adversary doesn't know a proxy signer's private key, but know the original signer's private key and the rest proxy signer's private key, and in the forgery case 3, the adversary doesn't know the original signer's private key, but know all the proxy signer's private key. Therefore, the above security notion conforms to the strong unforgeability of proxy signatures [10, 11]. Additionally, the security definition [5] is shown to be achieved through a generic construction involving sequential aggregate signatures. We can show our security definition which is achieved through a similar construction too. We are forced to omit the detail due to the page limitation.

4. Conclusion

Multi-proxy signature is an important proxy signature, identity-based cryptography gets a focus at present . Some identity-based multi-proxy signature schemes have been proposed, but the security model for identity-based multi-proxy signature schemes has not been presented. In this paper, we give the formal definition of identity-based multi-proxy signature schemes, and put forward a security notion for them. Our work is of great importance for constructing the provably secure identity-based multi-proxy signature schemes in the future.

Acknowledgements

The authors extend deep gratitude to Hongtao Wang, Feng Li, Wenguang An, and Xinfang Luo for helpful discussions about this work.

References

[1] M. Mambo, K. Usuda, E. Okamoto. Proxy signatures for delegating signing operation[C]. In: Proceedings of the 3rd ACM Conference on Computer and Communications Security (CCS'96). New York: ACM Press, 1996. 48-57.

[2] S. Hwang, C. Shi. A simple multi-proxy signature scheme[C]. In: Proceedings of the 10th National

Conference on Information Security, Hualien, Taiwan, 2000, 134-138.

[3] A.Boldyreva, A. Palacio, B. Warinschi. Secure proxy signature schemes for delegation of signing rights[C]. Cryptology ePrint Archive, Report 2003/096, 2003. http://eprint.iacr.org/.

[4] T. Malklin, S. Obana, M. Yung. The hierarchy of key evolving signatures and a characterization of proxy signatures[C]. In: Proceedings of Eurocrypt 2004, Lecture Notes in Computer Science 3027. Berlin: Springer-Verlag, 2004. 306-322.

[5] J.C.N. Schuldt, K. Matsuura, K.G. Paterson. Proxy signatures secure against proxy key exposure[C]. In: Proceedings of Public Key Cryptography 2008 (PKC'08), Lecture Notes in Computer Science 4939. Berlin: Springer-Verlag, 2008. 141-161.

[6] A.Shamir. Identity-based cryptosystems and signature schemes[C]. In: Proceedings of Crypto 1984, Lecture Notes in Computer Science196.Berlin:Springer-Verlag,1984.47-53

[7] D. Boneh, M. Franklin. Identity-based encryption from the Weil pairing[C]. In: Proceedings of Crypto 2001, Lecture Notes in Computer Science 2139. Berlin: Springer-Verlag, 2001. 213-229.

[8] X. Chen, F. Zhang, K. Kim. ID-based multi-proxy signature and blind multisignature from bilinear pairings[C]. In: Proceedings of KIISC (Korea Institute of Information Security and Cryptology) Conference 2003, Korea, 2003, 11-19.

[9] X. Li, K. Chen. ID-based multi-proxy signature, proxy multi-signature and multi-proxy multi-signature schemes from bilinear pairings[J]. Applied Mathematics And Computation. 2005, 169(1): 437-450.

[10] B. Lee, H. Kim, K. Kim. Strong proxy signature and its applications[C]. In: Proceedings of the 2001 Symposium on Cryptography and Information Security (SCIS'01). Oiso, Japan, 2001. 603-608.

[11] Lee B, Kim H, Kim K. Secure mobile agent using strong non-designated proxy signature[C]. In: Proceedings of the 6th Australasian Conference on Information Security and Privacy (ACISP'01), Lecture Notes in Computer Science 2119. Berlin: Springer-Verlag, 2001. 474-486.

Wang Tao, born in 1973, associate professor. He received his M.S. from the Beijing University of Technology in January 2008. And he is the head of the department of information management and engineering of the Hebei Finance University for the present. So far, he has published 5 research papers. His current research interests include network security and wireless sensor network.

Li Mingxiang, born in 1968, associate professor. He received his Ph.D. from the University of Science and Technology Beijing in January 2009. Nowadays, he is a researcher of the Hebei Science and Technology Financial Critical Laboratory. His current research interests include cryptography and cloud computing.