中華民國八十六年全國計算機會議

植基於 RSA 之聯合簽章法
On digital multisignature schemes based on RSA

洪國寶
Gwoboa Horng

國立中興大學 資訊科學研究所
Institute of Computer Science, National Chung Hsing University

摘要

聯合簽章是一種允許許多人共同簽署一份文件的密碼協定法。 RSA 密碼系統，既可加密又可簽章，是最適合用來設計聯合簽章，但是須先解決 reblocking problem 。本文探討 RSA 系統之 reblocking problem 的各種解法，及值基於它們的聯合簽章法。然後提出一新的解法，及植基於它的聯合簽章法。我們所提的新法沒有簽署次序及模數大小之限制，但會增加些微的位元。

關鍵字: 聯合簽章、 RSA 密碼系統、 Reblocking problem

Abstract

In this paper, a solution to the RSA reblocking problem and a digital multisignature scheme based on RSA scheme are presented. Our multisignature scheme has the following advantages: No initial setup is necessary. It allows the signatories to have full freedom in choosing their moduli. Furthermore, the signing order is not restricted. However, there is a slight bit expansion with our scheme.

Keywords: Digital multisignature, RSA Cryptosystem, Reblocking problem.

1 Introduction

The concept of public key cryptography was first introduced by Diffie and Hellman in 1976 [1]. The invention of public key cryptography solved the problem of how to share secret keys over open network in conventional private key cryptosystems. Furthermore, it also provided a method for generating digital signatures. A digital signature is a protocol that produces the same effect as a real signature [2]. There exists many applications where it is required that a message is to signed by more than one signatories. Of course each signatory can generate a separate signature for the message. However, an important disadvantage is that the bandwidth overhead will increase. Therefore, we need to develop new schemes.

A digital multisignature scheme [3] is a multi-party cryptographic protocol. It allows multi-signatories to sign a document. Assume there are k signatories $S_1, S_2, \ldots, S_k$, the order can be pre-defined or determined during the signing process. Let the signing algorithm and verification algorithm of signatory S_i, $i = 1, \ldots, k$, be D_i and E_i, respectively. The multisignature of the k signatories on a document M can be created sequentially: S_1 verifies the document M, creates his signature $C_1 = D_1(M)$, and passes C_1 to the next signatory S_2. S_2 verifies the signature C_1 to see if $M = E_1(C_1)$, if it is, then S_2 creates his signature $C_2 = D_2(C_1)$ on C_1 and passes C_2 to the next signatory S_3. This process continues. That is, S_i verifies the partial multisignature C_{i-1} to see if $M = E_1(E_2(\cdots(E_{i-1}(C_{i-1})\cdots))$, if it is, then S_i creates his signature $C_i = D_i(C_{i-1})$ on C_{i-1}, and passes C_i to the next signatory S_{i+1}. The result, C_k, generated by the final signatory, S_k, is the multisignature of the initial document M.

The rest of the paper is organized as follows: Section 2 describes the RSA reblocking problem and some prior solutions. Section 3 proposes a new solution to the reblocking problem and a new scheme for digital multisignature. Section 4 analyzes the new scheme. Section 5 contains the conclusions.

2 RSA reblocking problem and solutions

The most popular digital signature scheme is the RSA scheme [4]. To set up RSA scheme, a user, say A, chooses two large primes P_A and Q_A at random. Let $N_A = P_A Q_A$ and $\phi(N_A) = (P_A - 1)(Q_A - 1)$. Two more integers e_A, called the public exponent, and d_A, called the secret exponent, are chosen such that $e_A d_A \equiv 1 \bmod \phi(N_A)$. The pair (e_A, N_A) is made publicly known. However, P_A, Q_A and d_A are kept secret. The digital signature created by user A for a message M, $0 < M < N_A$ and $(M, N_A) = 1$, is the integer C such that $C \equiv M^{d_A} \bmod N_A$. Anyone can verify the correctness of the signature by checking $C^{e_A} \equiv M \bmod N_A$. That is, $D_A(M) \equiv M^{d_A} \bmod N_A$ is the signing algorithm and $E_A(C) \equiv C^{e_A} \bmod N_A$ is the corresponding verification algorithm.

RSA system can also be used for preserving privacy. That is, $E_A(M)$ can be used as the enciphering algorithm and $D_A(C)$ can be used as the corresponding deciphering algorithm. In fact, RSA system is the first cryptosystem which can be adapted for both digital signature and secrecy at the same time. For example, assume there is another user B with the enciphering algorithm $E_B(M) \equiv M^{e_B} \bmod N_B$. Then user A can achieve both authentication and secrecy at the same time for a message M by sending $E_B(D_A(M))$ to user B, assuming $N_A < N_B$. However, if $N_A > N_B$ then $E_B(D_A(M))$ will cause a loss of accuracy. This is called the reblocking problem. A solution suggested by Kohnfelder [5] is to apply E_B before D_A. Other solutions include using threshold value [4] and repeated exponentiation [4, 6]:

- Using threshold value: A threshold value h is pre-defined. Then each user chooses two sets of public and secret keys. The first set has modulus less than h and is used for authentication. The second set has modulus greater than h and is used for secrecy.
- Using repeated exponentiation: Let h be a pre-defined threshold value. Each user has a single set of keys with modulus between h and $2h$. A message less than h is enciphered as the ordinary RSA scheme, except that if the ciphertext is greater than h, it is repeated re-enciphered until it is less than h. Similarly for decryption, a ciphertext is repeated deciphered to obtain a value less than h.

RSA system is suitable for generating multisignature sequentially. Let N_i is the modulus of the RSA system of the ith signatory. Several such schemes have been proposed [3, 7, 8]. However, due to the reblocking problem, most of them have limitations:

- Schemes based on threshold value: the signing order is pre-defined and $N_1 < N_2 < \cdots < N_k$.

- Schemes based on repeated exponentiation: all signers must choose moduli of the same size. That is, $|N_1| = |N_2| = \cdots = |N_k|$.

In this paper, we first propose a method to solve the reblocking problem and then show how it can be used for generating digital multisignature sequentially based on RSA system. In the following, we will assume that both time stamp and hash function have been applied to a message before it is signed.

3 Our solution

Let $N_A = P_A Q_A$, e_A, and d_A be the modulus, the public exponent, and the secret exponent of user A respectively. Similarly, let $N_B = P_B Q_B$, e_B, and d_B be the modulus, the public exponent, and the secret exponent of user B respectively. Assume $N_A > N_B$. To solve the reblocking problem, we will allow the modulus N_B to increase provided that the public exponent e_B remains unchanged. A simple approach is to multiply N_B by 2^l for some integer l such that $2^{l-1}N_B < 2N_A < 2^l N_B$. Let $N'_A = 2N_A$ and $N'_B = 2^l N_B$. Note that under these new moduli, the secret exponent d_A is not changed since $\phi(N'_A) = \phi(N_A)$. Nevertheless, the secret exponent d_B may need to change to a new value d'_B such that $e_B d'_B \equiv 1 \bmod \phi(N'_B)$ where $\phi(N'_B) = 2^{l-1}(\phi(N_B)$. Such d'_B exists since e_B and $\phi(N'_B)$ are relatively prime. Now the new encrypting algorithm of B is $E'_B(M) \equiv M^{e_B} \bmod N'_B$.

In order for the system to work properly, the initial message to be signed or encrypted must be an odd integer. This can be easily achieved by concatenating a bit 1 to the end of the initial message.

It is easy to see that the above new scheme reveals nothing about d_B, d'_B, $\phi(N_B)$, or $\phi(N'_B)$. Therefore, it is as secure as the original RSA scheme. However, since $2N_A < N'_B < 2^2 N_A$, the size of final result after applying D_A followed by E'_B is two bits more than that of the Kohnfelder's solution. The two-bit bit expansion is quite small comparing to 1024 bits, a typical size of RSA scheme.

Based on RSA cryptosystem, assume signatory S_i has modulus N_i and public exponent e_i. These values are known to everyone. Let M be the document to be co-signed. Assume $M < N_1$ and M is odd. The first signatory, S_1, creates his signature $C_1 = M^{d'_1} \bmod N'_1$ where $N'_1 = 2N_1$ and $d'_1 e_1 \equiv 1 \bmod N_1$. He then sends $(M, \{S_1\}, C_1)$ to the next signatory S_2.

For $i = 2, 3, \ldots, k$, the signatory S_i receives $(M, \{S_1, \ldots, S_{i-1}\}, C_{i-1})$ from S_{i-1} and performs the following steps:

- Step 1: Compute $N'_1 = 2N_1$ and, for $j = 1, \ldots, i-1$, $N'_j = 2^{l_j} N_j$ where $l_j \geq 1$ and l_j is the smallest integer satisfying $N'_{j-1} < N'_j$.
- Step 2: Verify that $M = E'_1(E'_2(\cdots(E'_{i-1}(C_{i-1}))\cdots))$, where $E'_j(C) \equiv C^{e_j} \bmod N'_j$ for $j = 1, \ldots, i-1$. That is, check to see if C_{i-1} is the partial multisignature generated by $S_1, \ldots, S_{i-1}$ on document M. Continue to perform Step 3 to Step 5 if it is.
- Step 3: Compute the new signing algorithm $D'_i(C) \equiv C^{d'_i} \bmod N'_i$ where $d'_i e_i \equiv \bmod \phi(N'_i)$ and $N'_i = 2^{l_i} N_i$ where $l_i \geq 1$ and l_i is the smallest integer satisfying $N'_{i-1} < N'_i$.
- Step 4: Create the partial multisignature C_i such that $C_i = D'_i(C_{i-1})$.
- Step 5: Send $(M, \{S_1, \ldots, S_i\}, C_i)$ to the next signatory S_{i+1} if he is not the final signatory. Otherwise, stop and C_i is the digital multisignature.

4 Analysis and discussion

It is easy to see that the scheme works properly since C_i is odd and N_i' is even for $i = 1, 2, \ldots, k$. Note that Step 2 is optional. However, it is necessary to guard against chosen ciphertext attack if not all the signatories are trustworthy.

In the following, we analyze the size of the final multisignature. Let $N_j = \max_{1 \le i \le k} N_i$ and $|N_j| = n$. Then, by induction on k, $|C_k| \le n + k$, where k is the number of signatories. That is, the bit expansion of our digital multisignature scheme is at most k bits. In practice, k is relatively small comparing to the typical size of RSA scheme. Therefore, our scheme is quite practical.

5 Conclusion

Creating a digital multisignature sequentially has the advantage that it allows each signatory to verify the partial signature with respect to all the proceeding signatories. In this paper, we have proposed such a scheme based on RSA cryptosystem. Our scheme has the following advantages: No initial setup is necessary. It allows the signatories to have full freedom in choosing their moduli. Furthermore, the signing order is not restricted. However, there is a slight bit expansion with our scheme.

Acknowledgment. This work is supported by the National Science Council of Taiwan, Republic of China, under contract NSC86-2213-E-005-005.

References

[1] W. Diffie and M. Hellman, 'New directions in cryptography' *IEEE Trans. Information Theory*, 1976, **IT-22**, pp. 553-558.

[2] C. Pfleeger, *Security in computing*, Prentice-Hall, 1989.

[3] K. Itakura and K. Nakamura, 'A public-key cryptosystem suitable for digital multisignatures', *NEC Res. and Develop.*, 1983, **71**, pp. 1-8.

[4] R. Rivest, A. Shamir, and L. Adleman, 'A method for obtaining digital signatures and public-key cryptosystems', *Communications of the ACM*, 1978, **21**, pp. 120-126.

[5] L. Kohnfelder, 'On the signature reblocking problem in public-key cryptosystems', *Communications of ACM*, **21**, 1978, p. 179.

[6] T. Kielser and L. Harn, 'RSA blocking and multisignature schemes with no bit expansion', *Electr. Lett.*, 1990, **26**, pp. 1002-1003.

[7] T. Okamoto, 'A digital multisignature scheme using bijective public-key cryptosystems', *ACM Transactions on Computer System*, 1988, **6**, pp. 432-441.

[8] L. Harn and T. Kielser, 'New scheme for digital multisignature', *Electr. Lett.*, 1989, **25**, pp. 1002-1003.