Two Efficient RSA Multisignature Schemes

Sangjoon Park[1], Sangwoo Park[1], Kwangjo Kim[1], Dongho Won[2]

[1] #0710, ETRI, Yusong P.O.BOX 106, Taejon, 305-600, Korea
E-mail : {sjpark, psw, kkj}@dingo.etri.re.kr
[2] Dept. of Information Engineering, Sung-Kyun-Kwan Univ.,
300 Chunchun-dong, Suwon, Kyunggi-do, 440-746, Korea
E-mail : dhwon@simsan.skku.ac.kr

Abstract. In this paper, we propose two efficient RSA multisignature schemes, one is an improved version of Okamoto's scheme [6] and the other is that of Kiesler-Harn's scheme [3]. The first one causes bit expansion in block size of a multisignature, but the bit length of the expansion is no more greater than the number of signers regardless of their RSA modulus. The second one has no bit expansion, in which all signers have a RSA modulus with the same bit size and the same most significant l bits pattern. An average number of the required exponentiations to obtain a multisignature is about $(1+\frac{1}{2^{l}-1})m$, where m denotes the number of signers. Futhermore, our schemes have no restriction in signing order and are claimed to be more efficient than Okamoto's scheme and Kiesler-Harn's scheme respectively.

1 Introduction

In 1978, Rivest, Shamir and Adleman proposed new type of public-key cryptosystem, so called "RSA cryptosystem", whose security is based on the difficulty of factoring a large integer [8]. The practical implementation of RSA cryptosystem for multiple operations of a given message causes bit expansion problem inherently. As early works to solve this problem, there are Kohnfelder's reblocking method[4] and Levine-Brewley's repeated exponentiation method[5].

Itakura and Nakamura first suggested a new notion of a multisignature scheme [2] in which multiple signers generate a digital signature for a given document. To solve the difficulty of bit expansion in a RSA multisignature, they allowed a signer to have a RSA modulus with a different bit size according to his position in a hierachical structure. Thus, the signing order is restricted.

On the other hand, Okamoto proposed a multisignature scheme with no restriction of the signing order [6]. In his scheme, if the length of intermediate signature exceeds a pre-determined threshold value, then the extra bits exceeding the threshold value are appended to a message. So, the length of expanded message depends on the number of signers and the bit size of each signer's RSA modulus.

Harn and Kiesler proposed two multisignature schemes with no bit expansion[1, 3]. In one of their schemes, based on Kohnfelder's method, the signing order is chosen according to the size of signers' public keys. The other scheme is based on

Levine and Brawley's re-encryption method. Even though their multisignature schemes have no bit expansion problem and the signing order is not restricted, all signers must have a modulus with the same size and the computational complexity of obtaining a multisignature is increased.

In this paper, we propose two efficient RSA multisignature schemes, one is an improved version of Okamoto's scheme [6] and the other is that of Kiesler-Harn's scheme [3]. The first one causes bit expansion in block size of a multisignature, but the bit length of the expansion is no more greater than the number of signers regardless of their RSA modulus. The second one has no bit expansion, in which all signers have a RSA modulus with the same bit size, and the same most significant l bits pattern. In this scheme, an average number of the required exponentiations to obtain a multisignature is about $(1+\frac{1}{2^{l-1}})m$, where m denotes the number of signers.

This paper is organized as follows : In Section 2, we propose new RSA multisignature schemes. In Section 3, we discuss the security of our proposed schemes. Finally, we state concluding remarks in Section 4.

2 Multisignature Schemes

In this section, we propose two efficient RSA multisignature schemes. The following notations are used in this section.

- U_i : one of m signers, $U_1, \ldots, U_m$.
- n_i : RSA modulus of U_i.
- (e_i, n_i) : public key of U_i, (d_i, n_i) : secret key of U_i $(e_i \cdot d_i = 1 \pmod{\phi(n_i)})$.
- $|n_i|$: bit size of n_i.
- $A||B$: concatenation of A and B
- $h(\cdot)$: a secure hash function

<u>Scheme 1</u>

First, we introduce a new reblocking method in which the size of an enciphering block varies with the size of a message block. Let n be a RSA modulus and e a public key with $\gcd(e, \phi(n)) = 1$. Assume an odd M with $0 < M < 2^l n$. Then, $\phi(2^l n) = 2^{l-1}\phi(n)$ and $\gcd(e, 2^{l-1}\phi(n)) = 1$. If $e \cdot d = 1 \pmod{2^{l-1}\phi(n)}$, then $M^{e \cdot d} = M \pmod{2^l n}$. So, l varies with the size of a message M and d varies with l. If $C = M^e \pmod{2^l n}$ and $e \cdot d_1 = 1 \pmod{2^{l-1}}$, then $C \pmod{2^l} = M^e \pmod{2^l}$ and $M \pmod{2^l} = C^{d_1} \pmod{2^l}$. Thus, the proposed reblocking method can't be directly used for enciphering M with large block size.

Now, we show that this new reblocking method can be applied to a multisignature scheme. First, each user computes l_i from n_i as followings.

$$l_i = \begin{cases} 1 & \text{if } i = 1 \text{ or } 2^{l_{i-1}} n_{i-1} < 2n_i \\ 2^{l_i - 1} n_i < 2^{l_{i-1}} n_{i-1} < 2^{l_i} n_i & \text{otherwise.} \end{cases}$$

The generation and verification of a multisignature is done as follows :

- Signing by $U_1 : S_1 = (2h(M)+1)^{d_1} \pmod{2n_1}$ and he sends a message M and S_1 to the next signer U_2.
- Signing by U_i $(i = 2,\ldots,m)$: $S_i = S_{i-1}^{d_i} \pmod{2^{l_i} n_i}$, where $e_i \cdot d_i =$ $(\mathrm{mod}\ 2^{l_i - 1} \cdot \phi(n_i))$ and he sends M and S_i to the next signer.

Now, a receiver verifies S_m to be a multisignature of M by signers $U_1,\ldots,U_m$.

$$\begin{cases} S_{j-1} = S_j^{e_j} \pmod{2^{l_j} n_j} \ (j = m, m-1, \ldots, 2) \\ 2h(M)+1 = S_1^{e_1} \pmod{2n_1} \end{cases}$$

If $e_{i+1} \cdot d' = 1 \pmod{2^{l_{i+1}-1}}$ and $e_i \cdot d'' = 1 \pmod{2^{l_i - 1}}$, then $S_i = C_i^{d'}$ $(\mathrm{mod}\ 2^{l_{i+1}})$ and $S_i = S_{i-1}^{d''} \pmod{2^{l_i}}$. However, we can't obtain the most significant $|n_{i+1}|$ bits of S_i from C_i and the most significant $|n_i|$ bits of S_i from S_{i-1}.

If $L = \max(|n_1|, |n_2|, \ldots, |n_m|)$, then the bit length of the multisignature S_m is less than or equal to $L+m$. So, the length expanded by the proposed scheme is not greater than the number of signers. For example, if $|n_1| = |n_3| = |n_5| = 768$ and $|n_2| = |n_4| = |n_6| = 512$, then $|S_m| \leq 774$. So, the expanded bit length is 6. But, in this case, Okamoto'scheme has an expansion of 509 bits.

Scheme 2

Now, we propose another RSA multisignature scheme, which is a generalized version of Kiesler-Harn's scheme[3]. All users must choose a RSA modulus of the same number of bits - say m bits and the same most significant l bits pattern of all users' modulus must be the same. Let C be the l bits pattern which is pre-determined. Then the modulus of an user i can be represented as follows :

$$n_i = C \cdot 2^{k-l} + R_i (0 \leq R_i < 2^{k-l}). \tag{1}$$

Let $C \cdot 2^{k-l}$ be a threshold value u, and e_i and d_i be the RSA public key and secret key of user i, respectively. A multisignature by m signers is generated as follows :

- signer U_1 : U_1 generates a signature $S_1 = h(M)^{d_1} \pmod{n_1}$ for the original message M. If $S_1 \geq u$, he applies the repeated exponentiation technique to S_1 until $S_1 < u$ and sends M and S_1 to the second signer.
- signers U_i $(i = 2,\ldots,m)$: U_i computes a signature $S_i = S_{i-1}^{d_i} \pmod{n_i}$. If $S_i \geq u$ then he computes $S_i = S_i^{d_i} \pmod{n_i}$, repeatedly, until $S_i < u$. He sends M and S_i to the next signer.

The final signature S_m is the multisignature of M by the signers $U_1,\ldots,U_m$. Note that the signing order is independent of signers' public keys. To verify that S_m is the multisignature of M, the receiver also applies repeated exponentiation technique : For $i = m, m-1, \ldots, 2$, he computes $S_{i-1} = S_i^{e_i} \pmod{n_i}$ and if $S_{i-1} \geq u$, then he repeates exponentiations $S_{i-1} = S_{i-1}^{e_i} \pmod{n_i}$ until $S_{i-1} < u$. Finally, the receiver confirms $h(M) \stackrel{?}{=} S_1^{e_1} \pmod{n_1}$.

Since each signer's modulus n_i is of the form as equation (1), the probability that a random number $x(0 \leq x < n_i)$ is less than $h{=}C{\cdot}2^{k-l}$ is greater than $1{-}2^{-l+1}$,

$$Pr[0{\le}x{<}u|0{\le}x{<}n_i] = \frac{C \cdot 2^{k-l}}{n_i} = 1{-}\frac{R_i}{n_i} > 1{-}\frac{2^{k-l}}{2^{k-1}} = 1{-}2^{-l+1}.$$

So, if l is sufficiently large, then the average number of exponentiations required for obtaining a multisignature is close to m. For example, if $l = 32$ and $m = 10$, the average number of exponentiations of Kiesler-Harn's scheme is $1.5{\times}10{=}15$, but that of our scheme is $(1 + 2^{-31}){\times}10 \approx 10$. Thus, our scheme is more efficient than Kiesler-Harn's scheme.

Now, to make our multisignature scheme practical, we propose a method for generating a RSA modulus [7] which is required for our multisignature scheme. First of all, the key management center opens the bit length of the modulus, k, and some(fixed) pattern of l bits, C, to all users. For the sake of convenience, we suppose k is even. Each user's RSA modulus n must be k bits long and its most significant l bits pattern must be C. And, we expect that n becomes the product of two primes p and q, where $p-1$ and $q-1$ have large prime factors. A RSA modulus for the multisignature is generated as follows :

Step 1 Generate a random number R of $k{-}l$ bits, and compute $N{=}C{\cdot}2^{k-l}{+}R$.

Step 2 Generate a random number P of $\frac{k}{2}$ bits, and two prime numbers p' and q' of $\frac{k}{2}{-}l{-}t$ bits. And, compute $s{=}\lfloor\frac{P}{2\cdot p'}\rfloor$.

Step 3 If $p{=}2{\cdot}p'{\cdot}s{+}1$ is not a prime, then $s{=}s{+}1$ and repeat step 3, until p becomes a prime.

Step 4 Compute $Q{=}\lfloor\frac{N}{p}\rfloor$ and $s{=}\lfloor\frac{Q}{2\cdot q'}\rfloor$.

Step 5 If $q{=}2{\cdot}q'{\cdot}s{+}1$ is not a prime, then $s{=}s{+}1$ and repeat step 5, until q becomes a prime.

Step 6 Compute $n{=}p{\cdot}q$, and if $\lfloor\frac{n}{2^{k-l}}\rfloor$ equals to C, then, n is a RSA modulus which is required. Otherwise, return to step 1.

In our method, the most significant l bits of n is always C, but p and q are random. By the variable t in step 2, the most significant l bits of n in step 6 is not changed, even though s is incremented in step 3 and step 5. To generate efficiently n, we choose $t = 16$. By the proposed algorithm, $p-1$ and $q-1$ have large prime factors p' and q', respectively.

3 Security

First, we will discuss the security of the Scheme 1 which we proposed in Section 2.

Theorem 1 *If we can compute the secret key d with* $e \cdot d = 1 \pmod{2^{l-1} \cdot \phi(n))}$, *then a RSA signature of arbitrary message M can be obtained.*
(proof) If $d' = d \pmod{\phi(n))}$, *then* $C = M^d = M^{d'} \pmod{n}$ *and* $e \cdot d' = e \cdot d = 1 \pmod{\phi(n))}$. *So, C is a RSA signature of M.*

Theorem 2 *If, for any odd M* $(0 < M < 2^l \cdot n)$, *we can compute C with* $C = M^e \pmod{2^l \cdot n}$, *then the RSA signature of M can be computed.*
(proof) Let $C' = C \pmod{n}$. *Then,* $C' = M^e \pmod{n}$. *So, C' is a RSA signature.*

By Theorems 1 and 2, the security of the Scheme 1, based on the new reblocking method, depends on the security of a RSA signature scheme.

Now, we will discuss the security of the Scheme 2. Let n be a RSA modulus for the Scheme 2. Since n have large prime factors p and q, we can not factor it by any integer factoring algorithm. Moreover, $p-1$ and $q-1$ have large prime factors p' and q', respectively. Even if all users have n_i's, the most significant l bits of which are of the same value, the prime factors p_i and q_i of n_i are random. So, U_i can not guess the prime factors p_j and q_j of other user U_j.

4 Concluding Remarks

We have proposed two RSA multisignature schemes. First, we have suggested a new reblocking method in which the size of an enciphering block varies with the size of a message block and have applied the new reblocking method to a multisignature scheme. Each signer is allowed to have a RSA modulus with different bit size. It causes bit expansion which depends only on the number of signers regardless of the bit length of RSA modulus. The length of the expansion is less than or equal to the number of signers. If each signer has a RSA modulus with the same size, then our scheme and Okamoto's one have the same expansion. But, ours has smaller bit expansion than Okamoto's one.

The second multisignature scheme does not cause any bit expansion. All users must have a RSA modulus of a fixed length, k bits, the most significant l bits of which are the same. To obtain a multisignature, Kiesler-Harn's scheme requires an average exponentiation of $1.5m$, but our scheme requires about $(1+\frac{1}{2^{l-1}})m$. So, our scheme is said to be more efficient than Kiesler-Harn's one.

References

1. Harn, L. and Kiesler, T., "New scheme for digital signatures", *Electronics Letters*, 1989, 25, (22), pp.1527-1528
2. Itakura, K., and Kakamura, K., "A public-key cryptosystem suitable for digital signatures", NEC J. Res. Dev. 71 (Oct. 1983).
3. Kiesler, T. and Harn, L., "RSA blocking and multisignature schemes with no bit expansion", *Electronics Letters*, 1990, 26, (18), pp.1490-1491

4. Kohnfelder, L. M., "On the signature reblocking problem in public-key cryptography", *Commun. ACM*, 1978, 21, (2) pp.179
5. Levine, J. and Brawley, J. V., "Some cryptographic applications of permutation polynomials", *Cryptologia*, 1977, 1, pp. 76-92
6. Okamoto, T., "A digital multisignature scheme using bijective public-key cryptosystems", *ACM Trans. Computer Systems*, 1988, 6, (8), pp.432-441
7. Rivest, R. L., "Remarks on a proposed cryptanalytic attack on the M.I.T. public-key cryptosystem", *Cryptologia*, 1978, Vol.2, No. 1,pp. 62-65
8. Rivest, R. L., Shamir, A., and Adleman, L., "A method for obtaining digital signatures and public-key cryptosystem", *Commun. ACM*, 1978, 21, (2), pp. 120-126