서명 순서 지정 가능한 다중 서명 방식에 관한 연구

°이임영*, 강창구**, 김대호**

* 순천향 대학교 전산학과
** 한국전자통신연구소

A study on multisignature scheme with specified order

°Im-yeong LEE*, Chang-goo KANG**, Dae-ho KIM**

* Department of Computer Science, Soonchunhyang University
** Electronics and Telecommunications Research Institute

요 약

다수의 사람들이 동일한 문서에 서명하여야할 경우 디지탈 다중 서명 방식 중의 하나인 RSA 암호 방식을 이용한 서명 순서 지정 가능한 다중 서명 방식에 대하여 고찰 하였다. 그리고 이 방식에서는 전 서명자의 결탁에 의한 센터의 비밀키가 노출되며, 그룹간의 다중 서명문을 확인하지 못하는 문제점이 있음을 밝히고, 그 개선 방식을 제안하였다.

1. 서론

컴퓨터 산업의 급속한 발전과 정보화 사회 구현을 위한 각종 전산망이 확대 보급됨에 따라 정보의 불법적인 도청이나 내용의 변조를 방지하기 위한 보안 대책으로 디지탈 서명 기술의 중요성이 점차 크게 인식되어가고 있는 실정이다.

이러한 디지탈 서명 방식에는 단순 서명 방식과 다중 서명 방식이 있다. 향후 정보화 사회에서는 디지탈화된 전자 문서는 디지탈 통신망을 통하여 전달되고 처리되어질 것이며 전자 결재 시스템, 전자 계약 시스템 등에서는 동일한 전자 문서에 여러사람에 의한 서명이 필요할 것이다. 이와 같이 다수의 사람들이 동일한 문서에 서명하여야할 경우 디지탈 다중 서명 방식이 요구되어지며, 이러한 다중 서명 방식에 대하여 많은 연구가 행하여지고 있다.[1]~[6]

본고에서는 Doi등이 최근에 제안한 RSA암호방식을 이용한 서명 순서 지정 가능한 다중 서명 방식[8]에 대하여 논하고, 이 방식의 문제점에 대하여 언급하고, 그 개선 방안을 제안하고자 한다.

2. 기존 방식[8]

2.1 전제 조건

정의 1 (이산 대수 문제) RSA 암호에 있어서 n에 대하여, 비밀의 정수 d를 가지고서 다음식을 만족하는 M, C가 주어진다고 한다.

$$M \equiv C^d \bmod n$$

여기서 n이 충분히 클 경우 M, C로부터 d를 구하는 것은 곤란하다.

정의 2 (소인수 분해 문제) n을 큰 소수 p, q의 곱으로 한다. p, q가 충분히 클 경우 n을 소인수 분해하는 것은 곤란하다.

정의 3 (RSA 암호) p, q를 비밀의 큰 소수로 하고 $n = p \cdot q$를 공개한다. 또 공개키 e, 비밀키 d를 다음 식을 만족하도록 생성한다.

$$e \cdot d \equiv 1 \mod \lambda(n)$$

여기서 $\lambda(n)$은 LCM(p-1, q-1)이다. 그러면 임의의 정수 M(M < n)에 대하여 다음 식이 성립한다.

$$(M^e)^d \equiv M \mod n$$

또 p, q가 충분히 큰경우 공개 정보 n, e로부터 비밀 정보 p, q, $\lambda(n)$, d를 구하는 것은 곤란하다.

또한 여기서 사용하는 용어 및 기호를 다음과 같이 정의한다.

- n은 RSA암호의 비밀 키 p, q의 곱이다. p, q는 큰 소수이고 n의 소인수 분해는 곤란하다. p, q는 RSA암호의 조건을 만족하고 있다. 이와 같은 n을 여기서는 RSA 암호를 만족하는 공개키 n이라 한다.
- M은 서명 하여야 할 메세지를 나타내는 기호이고, X, Y, Z는 M에 대한 다중 서명 또는 작성 도중에의 서명을 나타내는 기호로 사용한다.
- 영 대문자 A, B, ..., U 는 사용자를 나타낸다. 대응하는 소문자(예를 들어 사용자 U에 대하여 u, u_1, u_2등)은 사용자와 센터만이 알고 있는 비밀키(임의의 정수)이다.
- 여기서 사용하는 가감승제 연산은 n을 법으로 하는 연산이다.

2.2 다중 서명형

기본적인 형태로서 서명순서를 생각하지 않는 "병행형 다중서명"과 서명순서를 지정 가능한 "순차형 다중 서명"의 방식을 나타낸다.

2.2.1 병행형 다중 서명

(그림 1)의 서명 구조와 같이 메세지 M에 대하여 사용자 A, B, C가 서명을 하고, 전 사용자가 공개키 y, z를 이용하여 A, B, C가 서명을 하였음을 확인하는 경우에 대하여 고찰한다. 여기서 사용자 A, B, C의 서명 순서는 무관하다.

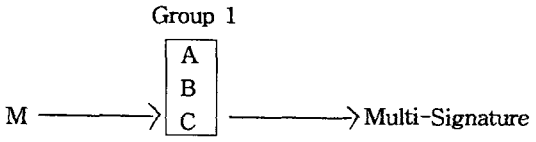

그림 1. 병행형 다중 서명

센터는 RSA 암호를 만족하는 공개키 n을 생성한 후 다음 식을 만족하는 y, z를 계산하여 공개한다.

$$(a+b+c+y)\cdot z \equiv 1 \mod \lambda(n)$$

각 사용자의 비밀키 a, b, c는 임의의 정수라도 무방하다. y를 변화 시키면 $a+b+c+y$ 와 $\lambda(n)$는 언제가는 서로소가 되기 때문에 $a, b, c, \lambda(n)$를 알고 있는 센터는 이러한 y, z를 생성 할 수 있다.

서명 순서

1. 센터는 n, y, z를 계산하고 공개한다.
2. 메세지 작성자는 M을 작성하고 그룹 1에 전달한다.
3. 그룹 1내의 서명을 하고자 하는 사용자 A, B, C는 비밀키에 의한 멱승연산을 하고 이것을 서명문으로 한다. 이러한 작업을 모든 사용자가 행한다.
4. 그룹 1의 전원의 서명을 중계자가 각각을 곱한 M^{a+b+c}를 생성하고 이것을 다중 서명문으로 한다.

확인 순서

확인자는 병행 다중 서명문 M^{a+b+c}의 정당성 확인으로서 다음 식이 성립함을 확인한다.

$$(M^{a+b+c}\cdot M^{y})^{z} \equiv M \mod n$$

이 식이 성립한 경우 확인자는 M에 대하여 A, B, C에 의한 병행 다중 서명을 확인할 수 있다. 본식에서 알 수 있듯이 서명 순서에 무관함을 알 수 있다.

2.2.2 순차형 다중 서명

순차형 다중 서명은 서명 순서를 지정할 수 있는 다중 서명 방식으로 서명 순서가 틀린 경우 다중 서명을 작성할 수 없는 방식이다. (그림 2)의 서명 구조와 같이 메세지 M에 대하여 사용자 A, B의 순서로 서명을 하고, 전 사용자가 공개키 y, z를 이용하여 A, B의 순서로 서명이 행하여 졌음을 확인하는 경우에 대하여 고찰한다. 여기서 C는 중계자이다.

그림 2. 순차형 다중 서명

센터는 RSA 암호를 만족하는 공개키 n을 생성한 후 다음 식을 만족하는 y, z를 계산하여 공개한다.

$$((a+1)\cdot b+y)\cdot z \equiv 1 \mod \lambda(n)$$

a, b는 임의의 정수라도 무방하다. y를 변화 시키면 $(a+1)\cdot b+y$ 와 $\lambda(n)$는 언제가는 서로소가 되기 때문에 $a, b, \lambda(n)$를 아는 센터는 이러한 y, z를 생성 할 수 있다.

서명 순서

1. 센터는 n, y, z를 계산하고 공개한다.
2. 메세지 작성자는 M을 작성하고 그룹 1(사용자 A)에 전달한다.
3. 사용자 A는 M^a를 계산하고, 이것을 서명문으로 하여 중계자에게 전달한다.
4. 중계자는 병행 다중 서명 방식 처럼 직전 그룹의 서명을 각각 곱한다. 이 경우 A뿐임으로 M^a를 그룹 2(사용자 B)에게 보낸다. 중계자가 행하는 작업은 A혹은 B가 대행하여도 된다.
5. 사용자 B는 $(M^a \times M)^b$를 계산하고, 이것을 다중 서명문으로 한다.

확인 순서

확인자는 순차 다중 서명문 $M^{(a+1)\cdot b}$의 정당성 확인으로서 다음 식이 성립함을 확인한다.

$$(M^{(a+1)\cdot b} \cdot M^y)^z \equiv M \mod n$$

이 식이 성립한 경우 확인자는 M에 대하여 A, B에 의한 순차 다중 서명을 확인할 수 있다. B, A 순서로 서명을 하면 서명의 확인 식은 $M^{((b+1)\cdot a+y)z}$가 된다. 바르게 서명한 경우와의 M의 지수부의 차는 법 $\lambda(n)$에서 $(a-b)\cdot z$이다. 이것은 법 $\lambda(n)$에서 0이되는 확률은 $1/\lambda(n)$임으로 무시할 수 있다.

2.2.3 다중 서명의 일반형

앞절의 결과를 이용하여 그룹 단위로 서명 순서를 지정 가능한 다중 서명 방식의 일반형에 대하여 알아본다.

일반적으로 서명 하는 사용자 그룹이 $G_1, G_2, \cdots, G_k$으로 분할되어 있고 그룹 $G_1, G_2, \cdots, G_k$의 순서로 서명을 하여야 할 경우에 대하여 고찰한다. 단, 그룹 G_i에 속하는 사용자 $U_{i1}, U_{i2}, \cdots, U_{ii_i}$의 서명 순서는 무관하다. 대응하는 서명 구조는 (그림 3)과 같다.

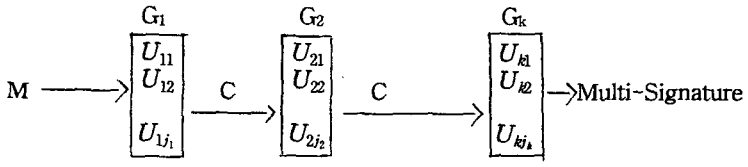

그림 3. 일반적인 다중 서명 구조

사용자 키의 공유

각 사용자 U는 비밀키 u를 임의로 선택하여 센터와 공유한다. u는 임의의 정수라도 무방하다.

서명 구조별 공개키 생성

사용자로 부터의 지정된 경로에 대한 다중 서명의 요구가 있을 경우 센터는 먼저 RSA암호를 만족하는 공개키 n을 생성한다. 다음으로 경로에 대응하는 다음의 방정식을 풀어 y, z를 계산하여 공개키로서 n, y, z를 공개한다. 또 u_{ij}는 사용자 U_{ij}의 비밀키로 한다.

$$((\cdots((2(\sum_{G_1} u_{1j})+1)(\sum_{G_2} u_{2j})+1)\cdots\cdots+1)(\sum_{G_k} u_{kj})+y)\cdot z \equiv 1 \mod \lambda(n)$$

메세지의 작성

메세지 작성자는 M을 작성하고 최초에 서명 할 사용자 그룹 G_1에게 초기 서명으로 M을 전달한다.

각 사용자의 다중 서명

각 사용자 U는 받아들인 서명 X에 대하여 자기의 비밀키 u를 사용하여 새롭게 $(X\times M)^u$를 계산하고 서명문으로 한다. 그러므로 최초 그룹의 서명문은 $(M\times M)^u = M^{2u}$가 된다.

그룹간의 이동

지정한 그룹 단위의 서명 순서에 따라 다중 서명이 그룹간에 이동할 때에는, 중계자는 받아들인 다중 서명 $(X\times M)^{u_1}, (X\times M)^{u_2}, \cdots$ 등을 각각 곱한다. 이것을 다중 서명으로 다음의 그룹에 전달한다. 전 서명자가 서명을 끝낸후에도 위와 같은 방법으로 모두 곱하여 이것을 다중 서명문으로 한다.

서명문의 확인

최종적으로 작성된 서명문 Z에 대하여 $(Z\cdot M^y)^z = M$을 확인하고, 이것이 성립한 경우 서명이 지정한 대로 행하여졌음을 확인한다.

본 방식에 있어서 동일 서명 구조로서 다중 서명을 작성할 때에 센터가 작성한 공개키를 몇번이라도 사용 가능하다. 또 각 사용자의 비밀키는 참가하는 다중 서명의 계열에 관계 없이 하나이다. 또한 사용자의 비밀키의 변경에 의한 정보의 변경은 그 사용자가 속하는 서명 구조의 공개키만이다.

3. 기존 방식의 문제점

3.1 중간에서의 확인 불가

다중 서명 방식에 있어서 최종 서명문이 작성된 후 확인을 행하도록 되어있기 때문에 중계자가 서명의 확인을 할 수가 없다. 즉 어느 그룹의 사용자가 잘못 서명을 하였더라도 최종 서명 확인자는 어느 그룹의 사용자인가를 검출하지 못한다. 2.2.2절의 식에서 B그룹의 서명자가 틀린 키(b′)로 서명을 했을 경우

$$((a+1)\cdot b' + y)\cdot z \equiv 1 \mod \lambda(n)$$

이 성립하게 되여, 최종 확인자는 A, B그룹 중 어는 그룹의 서명자가 틀린 키를 사용하였는지 검출할 수 없다.

3.2 전 사용자의 결탁에 의한 센터의 비밀키 노출

전서명자가 결탁을 하여 다음의 t 값을 얻는다고 하자.

$$(\cdots((2(\sum_{G_1} u_{1j})+1)(\sum_{G_2} u_{2j})+1)\cdots\cdots+1)(\sum_{G_k} u_{kj}) = t$$

그러면 $(t + y) \cdot z \equiv 1 \mod \lambda(n)$에서 t, y, z를 알고 있기 때문에 $\lambda(n)$ 이 $((t + y) \cdot z - 1)$의 약수임을 알게되어 센타의 비밀키를 추측할 수 있게 된다.

4. 개선 방안 제안

2장에서 논한 다중 서명 방식의 문제점은 서명 확인을 위하여 y, z의 공개로서 전 서명자의 결탁에 의한 센터의 비밀키 노출과 서명 중간에서의 다중 서명문 확인을 할 수 없다는 것이다. 본 장에서는 이러한 문제점을 개선한 방식에 대하여 논한다. 그 구성 방식은 2장에서와 마찬가지로 일반적으로 서명 하는 사용자 그룹이 $G_1, G_2, \cdots, G_k$으로 분할되어 있고 그룹 $G_1, G_2, \cdots, G_k$의 순서로 서명을 하여야 할 경우이다.

<u>사용자 키의 공유</u>

각 사용자 U는 비밀키 u를 임의로 선택하여 센터와 공유한다. u는 임의의 정수라도 무방하다.

<u>서명 구조별 공개키 생성</u>

사용자로 부터의 지정된 경로에 대한 다중 서명의 요구가 있을 경우 센터는 먼저 RSA암호를 만족하는 공개키 n을 생성한다. 다음으로 경로에 대응하는 다음의 방정식을 풀어 y_i, z_i $(i=1,2,\cdots,k)$를 계산하며, 서명 확인자의 비밀키로서 센터와 서명 확인자만이 보유한다. 또 u_{ij}는 사용자 U_{ij}의 비밀키로 한다.

$$(2(\sum_{G_1} u_{1j}) + y_1) \cdot z_1 \equiv 1 \mod \lambda(n)$$

$$((2(\sum_{G_1} u_{1j}) + y_1)(\sum_{G_2} u_{2j}) + y_2) \cdot z_2 \equiv 1 \mod \lambda(n)$$

$$((\cdots((2(\sum_{G_1} u_{1j}) + y_1)(\sum_{G_2} u_{2j}) + y_2)\cdots\cdots + y_{k-1})(\sum_{G_k} u_{kj}) + y_k) \cdot z_k \equiv 1 \mod \lambda(n)$$

위식에서 알수 있듯이 y_i, z_i의 생성 순서는 (y_1, z_1), (y_2, z_2), ... (y_k, z_k)의 순이다.

<u>메세지의 작성</u>

메세지 작성자는 M을 작성하고 최초에 서명 할 사용자 그룹 G_1에게 초기 서명으로 M을 전달한다.

<u>각 사용자의 다중 서명</u>

각 사용자 U는 받아들인 서명 X에 대하여 자기의 비밀키 u를 사용하여 새롭게 $(X \times M)^u$를 계산하고 서명문으로 한다. 그러므로 최초 그룹의 서명문은 $(M \times M)^u = M^{2u}$가 된다.

<u>중간 확인자의 서명문 확인</u>

중간 확인자 이전 그룹 i에 의하여 작성된 서명문 Z에 대하여 $(Z \cdot M^{y_i})^{z_i} = M$을 확인하고, 이것이 성립한 경우 서명이 지정한 대로 행하여졌음을 확인한다.

<u>그룹간의 이동</u>

지정한 그룹 단위의 서명 순서에 따라 다중 서명이 그룹간에 이동할 때에는, 중계자는 받아들

인 다중 서명 $(X\times M)^{u_1}, (X\times M)^{u_2}, \cdots$ 등을 각각 곱한다. 이것을 다중 서명으로 다음의 그룹에 전달한다. 전 서명자가 서명을 끝낸후에도 위와 같은 방법으로 모두 곱하여 이것을 다중 서명문으로 한다.

서명문의 확인

최종적으로 작성된 서명문 Z에 대하여 $(Z\cdot M^{y_k})^{z_k} = M$을 확인하고, 이것이 성립한 경우 서명이 지정한 대로 행하여졌음을 확인한다.

5. 결론

본고에서는 다수의 사람들이 동일한 문서에 서명하여야할 경우 디지탈 다중 서명 방식의 하나인 Doi등이 제안한 RSA 암호 방식을 이용한 서명 순서 지정 가능한 다중 서명 방식에 대하여 고찰 하였다. 그리고 이 방식에서는 전 서명자의 결탁에 의한 센터의 비밀키가 노출되며, 그룹간의 다중 서명문을 확인하지 못하는 문제점이 있음을 밝히고, 그 개선 방식을 제안하였다.

표 1은 Doi 방식과 본 방식을 비교한 것이다.

표 1. 각 방식 비교

	DOI 방식	본 방식
서명자 비밀키의 노출 가능성	없음(O)	없음(O)
센터 비밀키의 노출 가능성	있음(X)	있음(O)
서명 중간 과정에서의 확인	불가(X)	가능(O)
센터의 비밀키 보유수	적음(O)	많음(X)
사후 부인 가능성	없음(O)	없음(O)

O : 장점
X : 단점

참고 문헌

[1] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithm", IEEE Trans. on Inform., Vol. IT-31, No. 4, pp.469-472, 1985

[2] S. G. Akl, "Digital signature : a tutorial survey", IEEE Computer, No. 16, pp. 27-35, 1983

[3] D. W. Davies, "Applying the RSA digital signature to electric mail",IEEE Computer, pp. 55-62, 1983.2

[4] 강창구, 김대영, "새로운 순차 및 동시 다중 서명 방식", 한국통신정보보호학회 논문지, 제2권, 제1호, pp. 36-44, 1992

[5] 강창구, 김대영, "디지탈 서명 다중 방식 비교", 한국통신정보보호학회지, 제2권, 제4호, pp.7-16, 1992

[6] K. Itakura and K. Nakamura, "A public key cryptosystem suitable for digital multisignature", NEC J. Res. Dev. 71, pp. 1-8, 1983

[7] R. L. Rivest, A.Shamir and L. Adliman, "A method for obtaining digital signature and public key cryptosystem", Comm. of the ACM, Vol. 21, No.2, pp. 120-126, 1978

[8] H. Doi, E. Okamoto, M. Mambo and T. Uyematsu, "Multisignature scheme with specified order", SCIS '94(Jan. 1994)