공학석사 학위논문

대리서명 방식의 확장에 관한 연구

지도교수 박 지 환

이 論文을 工學碩士 學位論文으로 提出함

2002년 8월

부경대학교 산업대학원

전 산 정 보 학 과

이 명 희

이 논문을 이명희의 공학석사

학위논문으로 인준함

2002년 6월 22일

주 심 공학박사 여 정 모 

위 원 이학박사 박 홍 복 

위 원 공학박사 박 지 환 

차 례

그림 차례

표 차례

기호 목록

A Original Signer

B Verifier

P Proxy Signer

p 512비트 이상의 큰 소수

q $q|p-1$ 인 큰 소수

g 위수가 q 인 Z_P 상의 원소

x_A 원 서명자(위임서명자)의 비밀키

y_A $y_A = g^{x_A} \bmod p$, 원 서명자(위임서명자)의 공개키

x_B 검증자의 비밀키

y_B $y_B = g^{x_B} \bmod p$, 검증자의 공개키

x_P 대리서명자의 비밀키

y_P $y_P = g^{x_P} \bmod p$, 대리서명자의 공개키

$h()$ 안전한 일방향 해쉬함수

X_{AP} 대리서명용 비밀키

Y_{AP} $Y_{AP} = g^{x_{AP}} \bmod p$, 대리서명용 공개키

m_w 보증서(원서명자의 ID, 대리서명자의 ID, 위임기간 등)

M 메시지

ID_A 사용자A의 신원(식별자)

$E()/D()$ 관용암호/복호 알고리즘

$x||y$ x와 y의 연결

TS 타임스탬프

U_0 0번째 서명자(원 서명자)

U_i i 번째 대리서명자

x_0 0 번째 서명자의 비밀키

y_0 $y_0 = g^{x_0} \bmod p$, 0번째 서명자의 공개키

x_i i 번째 대리서명자의 비밀키

y_i $y_i = g^{x_i} \bmod p$, 0번째 서명자의 공개키

m_{wi} i 번째 대리서명자의 보증서

A Study on Extension of Proxy Signature

Hee-Myoung Lee

Dept. of Computer Information Graduate School Pukyong National University

abstract

Mambo, Usuda, Okamoto proposed the proxy signatures in 1996. In their schemes an original signer can delegate to proxy signer the power of producing original signer's signatures. In 2000, Araki proposed a multi-level proxy signatures extending Mambo-Usuda-Okamoto schemes. It is possible for him/her to delegate the other person using the scheme that is the repeated Mambo-Usuda-Okamoto scheme .

But Araki's scheme has two problems. First a verifier can not confirm whether the valid signers sign or not. Second it is difficult to make a proxy key is suited to multi-level eletronic enviroments(for example, re-creation of delegation signature's period). Therefore, we will

propose a multi-level proxy signature extending a warrent-delegation proxy signature and a proxy-signcryption scheme for a secure and adaptable applications. we also proposed a new proxy signature scheme that can be applied to wireless network. Under wireless network environments, users(customer) can sign using mobile agent that has the excellent computation ability relatively. The proposed scheme provides non-repudiation of the signers and confidentiality.

1. 서 론

대리서명 방식은 원 서명자가 지정한 사람(이하 대리 서명자)이 원 서명자를 대신해서 서명을 하는 방법으로 위조가 불가능한 것과 검증 가능성의 조건을 갖는다[1]. 이와 같은 서명 방식은 하드웨어 장치의 도움 없이 메시지에 서명하는 능력을 제3자에게 전달하는데 사용될 수 있다. 예를 들면, 어떤 회사의 간부가 정보통신망에 접속할 수 없는 지역으로 출장을 간 경우를 생각해 보자. 그는 출장 기간동안 메일을 받아 그 메일에 응답을 하려면 출장 전에 다른 사람에게 메일을 받을 수 있도록 조치하고 그 메일에 응답할 수 있는 권한을 부여해야 한다. 이러한 경우 본인을 대신해서 사전에 문서에 대한 대리 서명을 할 수 있도록 조치를 취해야 한다.

Mambo, Usuda, Okamoto[1]는 본인이 부재 중 자신을 대리해서 서명을 할 수 있는 대리서명 방식을 최초로 제안하였다. 대리서명 방식은 원 서명자가 대리 서명자에게 제공하는 형태에 따라 완전 위임방식, 부분 위임방식, 보증 위임방식으로 나누어진다. 다시 앞의 예를 생각해보자. 만약에 대리서명을 위임받은 대리인도 출장을 가야 할 상황이 발생한다면 그도 역시 서명 생성 능력을 위임해야만 할 것이다.

이에 본 논문에서는 기존의 보증 부분 위임 대리서명 방식을 확장해서 원 서명자가 대리 서명자에서 서명을 위임했을 때 대리 서명자 역시 부재 시 다음 대리 서명자에게 위임해서 서명할 수

있는 다단계 대리서명 방식을 제안한다. 나아가 디지털 서명과 암호 시스템의 기능을 동시에 제공하는 proxy-signcryption 방식을 확장하고자 한다.

또한 본 논문에서는 무선 이동 통신상에서 사용자 이동 단말기가 적은 계산량으로 전자 서명을 수행하고 안전성을 제공하기 위해서 상대적으로 계산능력이 뛰어난 대리 Agent의 도움을 통해 사용자가 전자서명을 수행할 수 있는 대리서명 방식을 제안한다. 이를 통해 무선 이동 통신상에서 발생할 수 있는 사용자의 불법적 행위로부터 대리 Agent를 보호하기 위한 기법을 제공함으로서 안전성을 확보하고 있다

본 논문의 구성은 다음과 같다. 먼저 2장에서는 이와 관련된 대리서명 방식의 개념과 구조에 대해 소개하고, 기존의 Araki 등[3]이 제안한 Mambo 등[1]의 대리 서명방식을 바탕으로 다단계 대리 서명자를 가지는 대리서명 방식의 문제점을 살펴본다. 3장에서는 전자 환경에 더욱 안전하고 융통성 있는 응용을 위해 보증위임 대리 서명 방식과 Proxy-Signcryption방식을 확장한 다단계 대리 서명 방식을 제안하여 기존의 방식과 비교, 분석한다. 그리고 4장에서는 무선 이동 통신상에서 인증성과 안전성을 제공하는데 필요한 요구사항에 대해 소개하고, 사용자의 불법적 행위로부터 사용자 및 대리 서명 Agent를 보호하기 위한 기법과 forward secrecy를 제공하는 서명방식을 제안하고, 5장에서는 결론 및 향후 연구과제를 제시한다.

2. 관련 연구

대리서명은 원 서명자가 대리 서명자에게 서명 능력을 위탁하고 대리 서명자는 원 서명자를 대신하여 서명을 생성할 수 있는 변형 전자서명 기법이다. 검증자는 대리서명을 검증할 때 원 서명자가 인정한 대리 서명자에 의하여 서명되었음을 검증해야 한다. 대리서명의 기본적인 방법은 [그림1]과 같이 원 서명자가 위임정보(대리서명자의 ID,위임기간, 서명정보, 보증서 등)를 대리 서명자에게 비밀리에 전달한다. 이 때 대리 서명자는 위임정보를 이용하여 대리 서명 비밀키를 생성하고 그 키를 이용하여 대리서명을 수행한다. 그리고 대리 서명자는 서명문은 검증자에게 전달하여 정당하게 서명되었는지를 검증한다.

[그림 1] 대리서명의 시나리오

2.1 Mambo등이 제안한 대리서명 방식

Mambo가 처음 제안한 대리서명 방식[1][2]은 원 서명자의 서명 능력을 위임하는 형태에 따라 완전 위임, 부분 위임, 보증 위임 방식으로 분류된다. 각 위임 형태에 대한 정의는 다음과 같다.

(1) 완전 위임(full delegation)

원 서명자가 대리 서명자에게 자신의 서명용 비밀키를 주는 경우를 말한다. 따라서 대리 서명자에 의한 서명과 원 서명자에 의한 서명은 구분되지 않는다.

(2) 부분 위임(partial delegation)

원 서명자가 대리 서명용 비밀키를 자신의 비밀키를 이용하여 생성한다. 이 때 자신의 비밀키는 대리 서명용 비밀키로부터 계산 불가능하여야 한다. 부분 위임의 형태는 보호형, 비 보호형으로 분류한다.

(3) 보증 위임(delegation by warrant)

원 서명자가 대리 서명자에게 보증서(warrant)를 발행함으로서 대리 서명을 구현하는 방식을 말한다. 그리고 그 형태에 따라 지정한 사람을 대리 서명자로 선언하는 서류에 원 서명자가 일반적인 디지털 서명을 통하여 서명한 후, 그 서명된 보증서를 이용하여 대리서명을 실행하는 보증서 기

반 대리 서명 방식과 지정한 서명자를 위한 비밀키와 공개키를 생성하고 생성된 공개키에 대하여 원 서명자가 보증서를 만들어 지정된 대리 서명자에게 주는 소지자 기반 대리 서명 방식으로 구분한다.

세 가지의 대리 서명방식 중 실용성과 응용성이 뛰어난 방식이 부분위임 방식이다. 이 방식은 서명자가 대리 서명자를 통제할 수 있을 뿐 아니라 안전성이 우수하다. 이러한 대리서명 방식도 이산대수 문제를 기반으로 구성된다[3].

다음은 원 서명자가 대리인을 가장하여 대리서명을 할 수 있다. 따라서 제3자는 원 서명자가 서명한 것을 대리 서명자가 서명한 것으로 오인할 수 있다. 그러므로 대리 서명자를 보호하기 위한 방안이 요구된다. 다음은 부분 위임 방식중 대리인을 보호하지 않는 서명 기법과 대리인을 보호하는 서명 기법을 설명하도록 한다. 본 논문에서는 512비트 이상의 큰 소수 p , $q|p-1$인 큰 소수 q, 위수가 q인 Z_q상의 원소를 g 로 정의한다.

2.1.1 대리인 비보호형 대리 서명방식

대리 서명자는 원 서명자를 대신하여 서명할 수 있으나, 대리 서명자 이외에 원 서명자 또는 정당한 대리 서명자를 가장하여 대리서명할 수 있다. 그러므로 원 서명자가 대리 서명자에게 대리 서명정보를 위임할 때 서명 능력에 제한을 주지 않기 때문에 어떤 메시지도 서명이 가능하다. 그리고 대리 서명자로부터 수신된 서명 정보를 다른 서명자에게 주었을 때 그 서명자 역시 대리서명이 가능하므로 누가 정당한 대리 서명자인지 알 수가 없는 단

점이 있다[8]. 대리인 비보호형 프로토콜은 다음과 같다.

(1) 원 서명자 (Original Signer), **A**

원 서명자는 $k_0(\in Z_q)$ 를 선택하여 다음과 같이 K와 대리 서명자의 비밀 서명키 X_{AP}를 생성하여 (X_{AP}, K)를 대리 서명자에게 비밀리에 전송한다.

$$K \equiv g^{k_0} \bmod p \tag{2.1}$$

$$X_{AP} \equiv x_A + k_0 \cdot K \bmod q \tag{2.2}$$

(2) 대리 서명자 (Proxy Signer), **P**

대리 서명자는 자신이 받은 (X_{AP}, K)이 정당한 키인지 다음의 관계식에 의하여 확인한다.

$$g^{X_{AP}} \equiv y_A \cdot K^K \bmod p \tag{2.3}$$

맞으면 정당한 대리 서명용 키로 받아들이고 아니면 다시 요구하던지 이 프로토콜을 멈춘다. 정당하면 대리 서명자는 비밀 랜덤수 $k_1(\in Z_q)$ 를 선택하여 다음과 같이 중간값 r 과 서명문 M 을 압축하기 위해 해쉬 함수를 계산하고 서명 $S_{X_{AP}}$를 생성하여 검증자에게 $(K, S_{X_{AP}}, r, M)$ 를 전송한다. 대리 서명자는 일반 디지털 서명방식에 따라 서명할 수 있다. 여기에서는 Nyberg-Rueppel의 디지털 서명 방식을 이용하여 대리서명 과정을 설명하고 있다[4].

$$r = g^{k_1} \bmod p \bmod q \tag{2.4}$$

$$H = h(M) \tag{2.5}$$

$$S_{X_{AP}} \equiv (k_1 - r \cdot X_{AP} \cdot H) \bmod q \tag{2.6}$$

(3) 검증자(Verifier), B

검증자는 위임서명 $S_{X_{AP}}$ 를 검증하기 위해 먼저 다음과 같이 계산한다.

$$Y_{AP} \equiv y_A \cdot K^K \bmod p \tag{2.7}$$

$$H = h(M) \tag{2.8}$$

검증 과정은 다음과 같은 식의 성립여부로 대리서명의 정당성을 확인하게 된다.

$$r \equiv g^{S_{X_{AP}}} y_{AP}{}^{r\ H} \bmod p \bmod q \tag{2.9}$$

2.1.2 대리인 보호형 대리서명 방식

정당한 대리 서명자만이 대리 서명이 가능하다. 따라서, 원 서명자 또한 정당한 대리 서명자를 가장하여 대리 서명을 할 수 없다. 그러므로 앞에서와 같은 단점을 보완하여 대리인을 보호할 수 있는 방식이다. 대리인 보호형 프로토콜은 다음과 같다.

(1) 원 서명자, A

원 서명자는 $k_0(\in Z_q)$ 를 선택하여 다음과 같이 K와 대리서명자의 비밀 서명키 X_{AP}를 생성하여 (X_{AP}, K)를 대리 서명자에게 비밀리에 전송한다.

$$K \equiv g^{k_0} \bmod p \tag{2.10}$$

$$X_{AP} \equiv x_A + k_0 \cdot K \bmod q \tag{2.11}$$

(2) 대리 서명자, P

대리 서명자는 자신이 받은 (X_{AP}, K)이 정당한 키인지 확인하기 위해 원 서명자의 공개키와 위임 정보를 이용하여 다음의 관계식에 의하여 확인한다.

$$g^{X_{AP}} \equiv y_A \cdot K^K \bmod p \tag{2.12}$$

맞으면 정당한 대리서명용 키로 받아들이고 아니면 다시 요구하던지 이 프로토콜을 멈춘다. 정당하면 대리 서명자는 원 서명자가 서명용 키를 알 수 없도록 다음과 같이 변형하여 대리 서명용 키를 생성한다.

$$X_{AP}' \equiv X_{AP} + x_P y_P \bmod q \tag{2.13}$$

비밀 랜덤수 k_1 $(\in Z_q)$ 를 선택하여 다음과 같이 중간값

r 과 서명문 M 을 압축하기 위해 해쉬 함수를 계산하고 서명 $S_{X_{AP}}$를 생성하여 검증자에게 $(K, S_{X_{AP}}, r, M)$ 를 전송한다.

$$r = g^{k_1} \bmod p \bmod q \tag{2.14}$$

$$H = h(M) \tag{2.15}$$

$$S_{X_{AP}} \equiv (k_1 - r \cdot X_{AP}' \cdot H) \bmod q \tag{2.16}$$

(3) 검증자 B

검증자는 위임서명 $S_{X_{AP}}$ 를 검증하기 위해 먼저 다음과 같이 계산한다.

$$Y_{AP}' \equiv (y_A \ K^K) y_P^{y_P} \bmod p \tag{2.17}$$

$$H = h(M) \tag{2.18}$$

검증 과정은 다음과 같은 식의 성립 여부로 대리서명의 정당성을 확인하게 된다.

$$r \equiv g^{S_{X_{AP}}} \ y_{AP}{}^{r \ H} \bmod p \bmod q \tag{2.19}$$

2.2 보증 부분 위임 대리서명 방식

Kim등이 제안한 대리서명 방식[5]는 대리서명 정보와 대리 서명용 키의 유효기간을 포함하는 메시지에 원 서명자가 서명함으로서 만들어진 보증서를 사용하여 부분 위임을 실행시키는 방법이다. 이 제안 방식은 Mambo가 제안한 서명방식의 단점을 보완한 방식으로 보증서가 있기 때문에 검증자는 보증서를 바탕으로 대리서명이 되었는지를 확인해야 한다. 보증 부분 위임 대리서명프로토콜은 다음과 같다.

(1) 원 서명자, **A**

원 서명자는 $k_0(\in Z_q)$를 선택하여 다음과 같이 K와 대리서명자의 비밀 서명키 X_{AP}를 생성하여 (m_w, X_{AP}, K)를 대리 서명자에게 비밀리에 전송한다. 이 때, m_w 에는 원 서명자의 식별자(ID), 대리 서명자의 ID, 위임 기간등이 명시된다. 따라서 X_{AP}는 m_w와 K에 대한 원 서명자의 서명이다.

$$K \equiv g^{k_0} \bmod p \tag{2.20}$$

$$e_0 \equiv h(m_w, K) \tag{2.21}$$

$$X_{AP} \equiv x_A \cdot e_0 + k_0 \bmod q \tag{2.22}$$

(2) 대리 서명자, **P**

대리 서명자는 자신이 받은 (m_w, X_{AP}, K)이 정당한 키인지 다음의 관계식에 의하여 확인한다.

$$e_0 \equiv h(m_w, K) \tag{2.23}$$

$$g^{X_{AP}} \equiv y_A{}^{e_0} \cdot K \bmod p \tag{2.24}$$

맞으면 정당한 대리서명용 키로 받아들이고 아니면 다시 요구하던지 이 프로토콜을 멈춘다. 정당하면 비밀 랜덤수 k_1 $(\in Z_q)$ 를 선택하여 다음과 같이 중간값 r 과 서명문 M 을 압축하기 위해 해쉬 함수를 계산하고 서명 $S_{X_{AP}}$를 생성하여 검증자에게 $(K,\ S_{X_{AP}}, r, M, m_w)$를 전송한다.

$$r = g^{k_1} \bmod p \bmod q \tag{2.25}$$

$$H = h(M) \tag{2.26}$$

$$S_{X_{AP}} \equiv (k_1 - r \cdot X_{AP} \cdot H) \bmod q \tag{2.27}$$

(3) 검증자, **B**

검증자는 위임서명 $S_{X_{AP}}$ 를 검증하기 위해 먼저 다음과 같이 계산한다.

$$e_0 \equiv h(m_w, K) \tag{2.28}$$

$$Y_{AP} \equiv y_A{}^{e_0} \cdot K \bmod p \tag{2.29}$$

$$H = h(M) \tag{2.30}$$

검증 과정은 다음과 같은 식의 성립 여부로 대리서명의 정당성을 확인하게 된다.

$$r \equiv g^{S_{X_{AP}}} y_{AP}^{\ r\ H} \bmod p \bmod q \qquad (2.31)$$

2.3 대리인 보호형 Proxy-signcryption방식

C.Gamage등[7]은 M.Mambo가 제안한 부분 위임 대리서명 방식과 Y.Zheng의 signcryption 방식의 장점을 이용하여 proxy-sign-cryption 방식[6]을 제안하였다.

Proxy-signcryption이란 사용자가 지정한 대리인이 자신을 대신하여 정당한 signcryption 메시지를 생성할 수 있도록 하는 방식으로 signcryption을 생성하는데 요구되는 계산을 상대적으로 계산 능력이 뛰어난 proxy agent에 의존하는 것이다.

그러나 그들이 제안한 방식을 실제 응용에 적용할 경우 사용자가 proxy agent를 대신하여 정당한 서명을 생성할 수 있을 뿐만 아니라 자신이 전송한 메시지에 대해 부인할 수 있는 문제점이 있다. 이를 오수현 등[6]이 대리인 보호형 proxy-signcryption을 제안함으로 해결하였다.

이 방식은 alternative proxy X_{AP}'를 생성하는데 대리 서명자의 비밀키 x_P 가 사용되므로 이 값을 모르는 원 서명자는 정당한 proxy-signcryption을 생성할 수 없게 된다. 그리고 proxy-signcryption이 수신자는 Y_{AP}'를 계산하는 과정에 송신자의

공개키 y_A와 대리 Agent의 공개키 y_P를 동시에 사용하므로 원 서명자의 위임에 의해 대리 서명자가가 생성한 proxy-signcryption임을 확인할 수 있게 된다. 그러나, 송신자와 대리 서명자사이의 비밀키인 X_{AP}'가 드러날 경우 수신자 이외의 다른 사람이 키 K' 값을 계산할 수 있으며, 따라서 sincrypt된 문서를 복구할 수 있게 된다. 즉, 이 기법은 대리 서명키인 X_{AP}'에 대한 Forward Secrecy를 제공하지 못한다. X_{AP}'는 특정인의 개인키가 아니므로 대리서명인의 부주의로 각 개인의 비밀키보다 노출될 확률이 높다[9-10].

$$(y_{AP}' \cdot g^{H})^{S \cdot X_B} = (y_B^{x_{AP}' + H})^{S} \tag{2.32}$$

다음에서는 대리인 보호형 Proxy-Signcryption방식의 프로토콜을 기술한다.

(1) 원 서명자, **A**

원 서명자는 $k_0(\in Z_q)$를 선택하여 다음과 같이 K와 대리서명자의 비밀 서명키 X_{AP}를 생성하여 (X_{AP}, K)를 대리서명자에게 비밀리에 전송한다.

$$K \equiv g^{k_0} \bmod p \tag{2.33}$$

$$X_{AP} \equiv x_A + k_0 \cdot K \bmod q \tag{2.34}$$

(2) 대리 서명자, **P**

대리서명자는 자신이 받은 (X_{AP}, K)이 정당한 키인지 다음의 관계식에 의하여 확인한다.

$$g^{X_{AP}} \equiv y_A \cdot K^K \bmod p \tag{2.35}$$

정당하면 원 서명자의 부정행위를 막기 위해 X_{AP}'를 계산한다.

$$X_{AP}' \equiv X_{AP} + x_P y_P \bmod q \tag{2.36}$$

그리고 대리서명자는 비밀 랜덤수 $k_1(\in Z_q)$ 를 선택하고 다음과 같이 계산하여 검증자에게 (c, S, H, K)를 전송한다.

$$K' \equiv Y_B^{k_1} \bmod p \tag{2.37}$$

$$K' = K_1 \| K_2 \tag{2.38}$$

$$H = h_{K_2}(M) \tag{2.39}$$

$$S \equiv (k_1 / H + X_{AP}') \bmod q \tag{2.40}$$

$$c \equiv E_{K_1}(M) \tag{2.41}$$

(3) 검증자, **B**

검증자는 (2.42)를 계산한 후, 자신의 비밀키를 이용하여 다음을 구하고 메시지를 복호한다.

$$Y_{AP}' \equiv y_A \cdot K^K \bmod p \tag{2.42}$$

$$K' \equiv (y_{AP} \cdot g^H)^{S \cdot X_B} \bmod p \tag{2.43}$$

$$K' = K_1 \| K_2 \tag{2.44}$$

$$M \equiv D_{K_1}(c) \tag{2.45}$$

단, $h_{K_2}(M) = H$ 인 경우에만 정확하게 서명된 것으로 받아 들인다.

2.4 Araki의 대리서명 방식의 확장

Araki등이 제안한 대리 서명방식[11]은 Mambo등이 제안한 대리 서명방식[1]을 확장한 다단계 대리 서명 방식으로 대리인 보호용 대리 서명방식을 이용하여 서명용 키를 생성할 때 자신의 비밀키와 공개키를 사용하여 다시 서명용 키를 생성하는 것으로 이와 같은 방식을 proxy signer에서 sub-proxy signer로 위임하는 방식으로 제시하였다. 그러나 이러한 방식은 다음과 같은 두 가지의 문제점을 가지고 있었다.

첫 번째는 검증자가 아래와 같이 마지막 대리 서명자(U_i)에 의해서 생성 된 대리 서명용 공개키가 위임 순서에 의해 생성된 것인지를 확인할 수 없다.

$$\rho^i \equiv g^{\sigma_i} \tag{2.46}$$

$$\equiv (y_0 \cdot K_0^{K_0}) y_1^{y_1} \cdots y_j^{y_j} y_{j+1}^{y_{j+1}} \cdots y_i^{y_i} \bmod p \tag{2.47}$$

$$\equiv (y_0 \cdot K_0^{K_0}) y_1^{y_1} \cdots y_{j+1}^{y_{j+1}} y_j^{y_j} \cdots y_i^{y_i} \bmod p \qquad (2.48)$$

두 번째는 만약, U_{j+1}이 U_{j-1}과 협력한다면(where $0<j<i$), U_{j-1}은 U_j에게 비밀 서명 정보 (σ_{i-1}, K_0)를 비밀리에 전달한다. 그리고 U_j는 $\sigma_j(\equiv \sigma_{j-1} + x_j y_j \bmod q)$인 대리 서명키를 다시 생성해서 U_{j+1}에게 비밀리에 전달한다. 이 때 U_{j+1}가 U_{j-1}에게 σ_j를 주게 되면 U_{j-1}는 U_j의 비밀키 x_j를 계산할 수 있게 된다[11].

위와 같은 문제를 해결하기 위해 Araki등[11]은 위임 정보에 순서를 가질 수 있는 다단계 대리 서명방식을 제안하였다. 여기에서 검증자는 i-th 대리서명용 공개키가 대리 서명자 U_0에서 U_i까지의 공개정보 K_i 와 공개키 y_i 를 이용하여 위임 정보를 생성해야만 하므로 검증자는 대리 서명용 키 ρ_i를 사용하여 서명되었음을 확인할 수 있다. 그리고 만약 U_{j+1}이 U_{j-1}과 협력하여, U_{j+1}가 U_j의 비밀 서명 정보 대리 서명키(σ_i)를 U_{j-1}에게 주더라도 U_j의 랜덤수 k_j 를 알 수 없으므로 U_j의 비밀키 x_j를 알 수 없다. 그러므로 두 대리 서명자간의 부정 행위를 막을 수 있다.

다음은 위임정보 순서가 부여된 확장된 대리서명 프로토콜이다.

(1) 대리 서명용 키 생성

· 원 서명자 U_0는 아래와 같이 대리 서명용 키 σ_0를 생성하여 안전한 채널을 통해 대리 서명자 U_1에게 전송하고,

K는 신뢰센터에 보낸다.

$$k_0 \in Z_q \tag{2.49}$$

$$K \equiv g^{k_0} \mod p \tag{2.50}$$

$$\sigma_0 \equiv x_o + k_0 K_0 \mod q \tag{2.51}$$

· i번째 대리 서명자 U_i $(i>0)$가 다른 대리 서명자 U_{i+1}에게 원 서명자 U_0의 서명 생성능력을 위임하고자 한다면 다음의 단계를 수행하여 U_{i+1}에게 전송하고, K_i는 신뢰센터에 등록한다.

$$\lambda_i \equiv \sigma_i + x_i y_i \mod q \tag{2.52}$$

$$k_i \in Z_q \tag{2.53}$$

$$K_i \equiv g^{k_i} \mod p \tag{2.54}$$

$$\sigma_i \equiv (\sigma_{i-1} + x_i) y_i + k_i \mod q \tag{2.55}$$

(2) 대리 서명용 키 검증

대리 서명자 U_i는 U_{i-1}에게 받은 σ_i와 U_{i-1}의 공개키 y_{i-1}와 대리 서명용 공개키 σ_{i-1}를 이용하여 (2.56)를 계산, 대리 서명용 키를 검증한다.

$$g^{\sigma_{i-1}} \equiv (((y_0 K_0^{K_0} y_1)^{y_1} K_1) \cdots y_{i-1})^{y_{i-1}} K_{i-1} \mod p \tag{2.56}$$

위 식이 검증되면, U_i는 U_0의 대리 서명용 키 λ_i, σ_i 를 생성할 수 있다. 여기에서 λ_i는 서명키이고, σ_i는 다른 대리인에

게 보내는 대리 서명용 키이다.

(3) 서명 생성 및 검증

U_i는 일반적인 서명 방식을 이용하여 $SIG_{U_i}(m, \lambda_i)$ 대리 서명을 생성할 수 있다. 또한 이 서명을 받은 검증자도 다음 식과 같이 대리 서명 공개키를 검증할 수 있다.

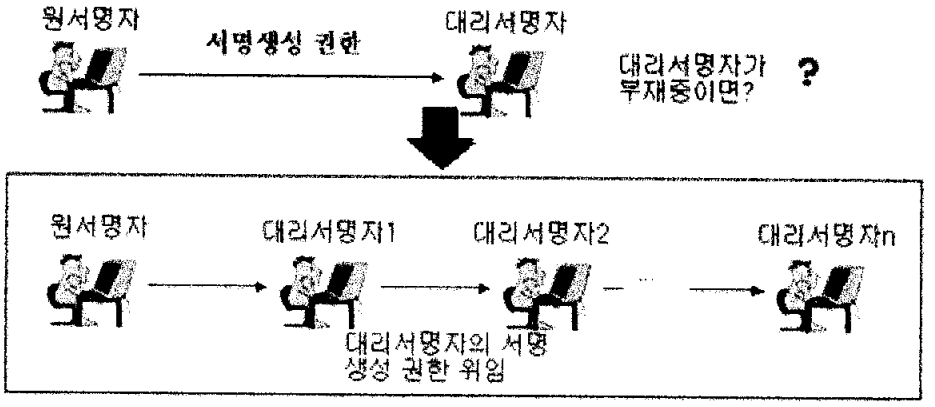
$$\rho_i \equiv g^{\lambda_i} \mod p \tag{2.57}$$

$$\equiv (((y_0 K_0^{K_0} y_1)^{y_1} K_1 \cdots y_{i-1})^{y_{i-1}} K_{i-1}) y_i^{y_i} \tag{2.58}$$

그리고 $Ver(Sig_{U_i}(m, \lambda_i), \rho_i)$을 이용하여 대리 서명을 검증할 수 있다.

3. 대리서명 방식의 일반적인 확장

예를 들어 회사에서 사장이 출장을 갔을 때 사장은 부사장에게 결재권은 위임하고 갈 것이다. 이 때 부사장 역시 갑자기 급한 일로 결재를 할 수 없을 때 부사장은 다음 사람에게 대신 결재권을 넘길 수 가 있다. 이러한 시나리오를 [그림2]와 같이 대리 서명방식에 적용해서 생각해 볼 수 있다.



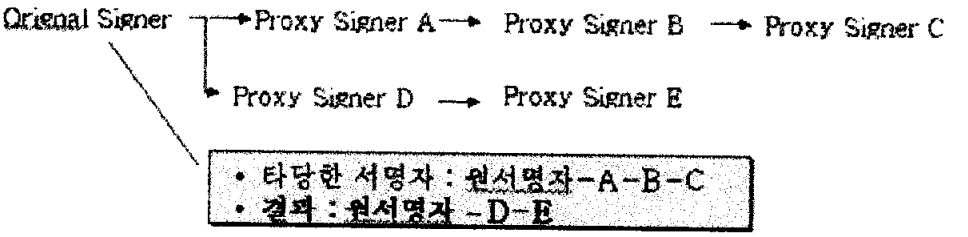
[그림 2] 확장 대리서명 방식의 시나리오

원 서명자가 대리 서명자에게 위임정보를 전달할 때 대리 서명자 역시 부재중이면 원 서명자는 다시 위임 정보를 생성해서 다른 대리 서명자에 전송해야 한다. 또한, 대리 서명자가 있더라고 대리 서명자가 많은 작업을 하고 있다면 서명 생성 또한 많은 시간을 소요하게 된다.

Araki등은 Mambo의 대리서명 방식을 변형, 확장한 다단계 대

리 서명방식[11]을 제안하였다. 그러나 이들이 제안한 방식은 검증자가 전송 받은 서명이 타당한 서명자들로부터 생성된 것인지 확신할 수 없는 문제와 다단계 환경(예를 들어 위임 서명 유효기간 재 설정)에 적합한 위임 서명 키 생성이 어렵다는 문제가 있다.

따라서 본 제안 방식은 전자환경에 좀 더 안전하고 융통성있는 응용을 위해 보증 위임 대리서명 방식과 Proxy-signcryption 방식을 확장한 다단계 대리 서명 방식을 제안하고자 한다.



[그림3] Ariki가 제안한 확장 대리서명 방식의 문제점

이 문제의 해결을 위해 본 논문에서는 대리 서명자와 유효 기간을 지정하는 보증 위임 대리 서명 방식을 확장한 방식과 대리인 보호형 proxy-signcryption을 확장한 방식을 제안하고자 한다.

3.1 보증 부분 위임 대리서명 방식의 확장

보증 부분 위임 대리서명 방식은 보증서를 사용하여 부분 위임을 실행하는 방식으로 이를 확장하여 원 서명자는 위임정보에 자신의 보증서와 함께 대리 서명자에게 전달하고 대리 서명자 역시 다른 대리 서명자에게 자신의 보증서를 포함하여 서명 정보를 위

임하므로 검증자는 이 대리서명이 타당한 위임자의 동의에 의해 정당하게 서명된 것임을 알 수 있고, Araki[11]가 제안한 확장 대리서명 방식의 장점도 포함하고 있다. 다음은 제안하는 방식의 프로토콜이다.

(1) 대리 서명용 키 생성

· 원 서명자 U_0는 아래와 같이 대리 서명용 키 s_0와 e_0를 계산하여 m_0(자신 ID와 대리 서명자 ID, 유효기간 등을 포함한 보증서)와 함께 안전한 채널을 통해 대리 서명자 U_1에게 전송한다.

$$k_0 \in Z_q \tag{3.1}$$

$$K \equiv g^{k_0} \mod p \tag{3.2}$$

$$e_0 \equiv h(m_o, K_0) \mod q \tag{3.3}$$

$$s_0 \equiv x_o e_0 + k_0 \mod q \tag{3.4}$$

· i번째 대리 서명자 U_i $(i>0)$가 다른 대리 서명자 U_{i+1}에게 원 서명자 U_0의 서명 생성 능력을 위임하고자 한다면 다음 같이 계산하여 U_{i+1}에게 $(m_0, m_1, \cdots, m_i)$와 $(e_0, e_1, \cdots, e_i)$ 및 대리 서명용 키 s_i를 전송한다. 이 때 이전에 받은 $m_0, m_1, \cdots, m_{i-1}$의 내용과 m_i (다음 대리 서명자의 ID와 유효기간 명

시)을 연결하여 해쉬 함수를 적용한다.

$$k_i \in Z_q \tag{3.5}$$

$$K_i \equiv g^{k_i} \mod p \tag{3.6}$$

$$e_i \equiv h(m_0 \| m_1 \| \cdots \| m_i, K_i) \mod q \tag{3.7}$$

$$s_i \equiv s_{i-1} + x_i e_i + k_i \mod q \tag{3.8}$$

(2) 대리 서명용 키 검증

대리 서명자 U_i는 U_{i-1}에게 받은 정보와 U_{i-1}의 공개키 y_{i-1}와 대리 서명용 공개키를 이용하여 (3.9)와 같이 계산한 후 e_{i-1}과 같은지 확인하고, (3.9)이 성립하면 (3.10)을 확인하여, 대리 서명용 키를 검증한다.

$$e_{i-1} \equiv h(m_0 \| m_1 \| \cdots \| m_{i-1}, K_{i-1}) \tag{3.9}$$

$$g^{s_{i-1}} \equiv (y_0^{e_0} y_1^{e_1} \cdots y_{i-1}^{e_{i-1}} K_0 K_1 \cdots K_{i-1}) \mod p \tag{3.10}$$

위 식이 검증되면, U_i는 U_0의 대리 서명용 키 s_i, r_i 를 생성할 수 있다. 여기에서 r_i는 서명키이고, s_i는 다른 대리인에게 보내는 대리 서명용 키이다.

$$r_i \equiv (s_{i-1} + e_{i-1} x_i) \mod q \tag{3.11}$$

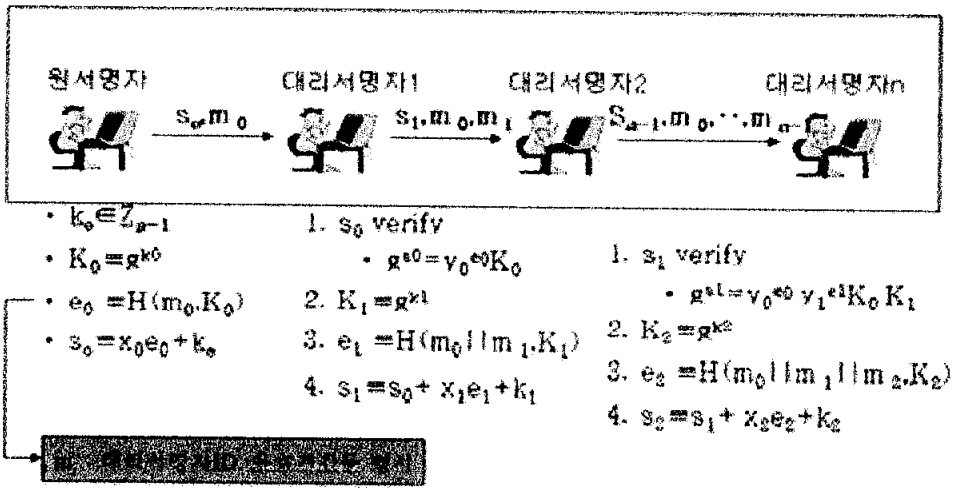
(3) 서명 생성 및 검증

U_i는 일반적인 서명 방식을 이용하여 $SIG_{U_i}(m, r_i)$대리 서명을 생성할 수 있다. 또한 이 서명을 받은 검증자도 다음 식과 같이 대리 서명 공개키를 검증할 수 있다.

$$R_i \equiv g^{r_i} \bmod p \tag{3.12}$$

$$\equiv (y_0^{e_0} y_1^{e_1} \cdots y_{i-1}^{e_{i-1}} y_i^{e_{i-1}} K_0 K_1 \cdots K_{i-1}) \bmod p \tag{3.13}$$

그리고 $Ver(Sig_{U_i}(m, r_i), R_i)$을 이용하여 대리 서명을 검증할 수 있다. [그림4]는 제안하는 방식에서 원 서명자와 대리 서명자들 사이의 흐름도를 나타낸 것이다.



[그림4] 제안하는 방식의 흐름도

3.2 대리인 보호형 Proxy-signcryption의 확장

대리인 보호형 Proxy-signcryption은 대리 Agent를 보호하기 위한 방식으로 원 서명자가 전송한 비밀 서명용 키에 자신의 비밀키를 이용하여 다시 비밀 서명용 키를 계산, 서명하는 방법이다. 만약, 이 때 대리 Agent가 많은 작업을 하고 있다면 서명 생성 및 암호 또한 많은 시간을 소요하게 된다. 그리고 대리 Agent가 작업을 수행할 수 없다면 원 서명자는 다시 대리 서명자를 지정하여 그 위임 환경에 맞는 서명정보를 생성해야 한다. 그러므로 Proxy-signcryption을 변형, 확장하여 다단계 Proxy-signcryption을 제안하고자 한다.

이 방식 또한 서명용 키 s_{i-1} 를 생성하는데 i-th대리 서명자의 비밀키 x_i 가 사용되므로 이 값을 모르는 원 서명자는 정당한 proxy-signcryption을 생성 할 수 없게 된다. 그리고 송신자와 여러 대리 Agent 사이의 비밀키인 s_{i-1}가 노출되더라도 마지막 대리서명자의 x_i를 알지 못하면 수신자 이외의 다른 사람이 키 IK값을 계산할 수 없으므로 정당성을 확인할 수 있다.

$$IK \equiv (y_0^{e_0} y_1^{e_1} \cdots y_{i-1}^{e_{i-1}} y_i K_0 K_1 \cdots K_{i-1} g^H)^{r x_B} \mod p \quad (3.14)$$

$$IK \equiv (g^{s_{i-1}} y_i g^H)^{r x_B} \mod p \quad (3.15)$$

$$IK \equiv (y_B^{s_{i-1}} y_B^{x_i} y_v^H)^r \mod p \quad (3.16)$$

s_{i-1}와 x_i 가 모두 드러나면 키를 계산할 수 있지만, 두 키가

모두 드러날 경우는 s_{i-1} 만 드러날 경우에 비하여 매우 낮은 확률로 발생한다.

다음은 제안하는 방식의 프로토콜이다.

(1) 대리 서명용 키 생성

· 원 서명자 U_0는 아래와 같이 대리 서명용 키 s_0와 e_0를 계산하여 m_0와 함께 안전한 채널을 통해 대리 서명자 U_1에게 전송한다.

$$k_0 \in Z_q \tag{3.17}$$

$$K_0 \equiv g^{k_0} \bmod p \tag{3.18}$$

$$e_0 \equiv h(m_o, K_0) \bmod q \tag{3.19}$$

$$s_0 \equiv x_o e_0 + k_0 \bmod q \tag{3.20}$$

· i번째 대리 서명자 $U_i\ (i>0)$가 다른 대리 서명자 U_{i+1}에게 원 서명자 U_0의 서명 생성 능력을 위임하고자 한다면 다음 같이 계산하여 U_{i+1}에게 $(m_0, m_1, \cdots, m_i)$ 와 $(e_0, e_1, \cdots, e_i)$ 및 대리 서명용 키 s_i를 전송한다. 이 때 이전에 받은 m_0, m_1, $\cdots$, m_{i-1}의 내용과 m_i (다음 대리 서명자의 ID와 유효기간 명시)을 연결하여 해쉬 함수를 적용한다.

$$k_i \in Z_q \tag{3.21}$$

$$K_i \equiv g^{k_i} \bmod p \tag{3.22}$$

$$e_i \equiv h(m_0 \| m_1 \| \cdots \| m_i, K_i) \bmod q \tag{3.23}$$

$$s_i \equiv s_{i-1} + x_i e_i + k_i \bmod q \tag{3.24}$$

(2) 대리 서명용 키 검증

대리 서명자 U_i는 U_{i-1}에게 받은 $s_i, e_i, (m_0, m_1, \cdots, m_i)$와 U_{i-1}의 공개키 y_{i-1}와 대리 서명용 공개키를 이용하여 다음과 같이 대리 서명용 키를 검증한다.

$$g^{s_{i-1}} \equiv y_0^{e_0} y_1^{e_1} \cdots y_{i-1}^{e_{i-1}} K_0 K_1 \cdots K_{i-1} \bmod p \tag{3.25}$$

위 식이 검증되면, U_i는 U_0의 대리 서명용 키 s_i 를 생성할 수 있다. 여기에서 s_i는 다른 대리인에게 보내는 대리 서명용 키이다.

(3) 서명 생성

U_i가 서명을 생성한다면, 다음과 같이 계산하고 메시지 m에 대한 signcryption을 생성, $(C, r, H, (e_0, e_1, \cdots, e_i))$를 검증자에게 전송한다.

$$k_i \in Z_q \tag{3.26}$$

$$IK \equiv y_B^{k_i} \mod p \tag{3.27}$$

$$IK = IK_1 \| IK_2 \tag{3.28}$$

$$H \equiv h_{IK_2}(m) \tag{3.29}$$

$$r \equiv \left(\frac{k_i}{H + x_i + s_{i-1}}\right) \tag{3.30}$$

$$C \equiv E_{IK_1}(m) \tag{3.31}$$

(4) proxy-signcryption의 검증

검증자는 자신의 비밀키를 이용하여, 식 (3.32)를 계산하고 그 값을 이용하여 메시지를 복호 및 검증한다.

$$IK \equiv (y_0^{e_0} y_1^{e_1} \cdots y_{i-1}^{e_{i-1}} y_i K_0 K_1 \cdots K_{i-1} g^H)^{x_B} \mod p \tag{3.32}$$

$$IK = IK_1 \| IK_2 \tag{3.33}$$

$$m = D_{IK_1}(C) \tag{3.34}$$

단, $h_{IK_2}(m) = H$인 경우에만 정당한 signcryption으로 받아들인다.

3.3 기존방식과의 비교 분석

제안하는 방식은 기존의 Araki등[11]이 제안한 방식과 같이 두 가지의 장점을 가지고 있다. 첫 번째로 검증자는 i-th 대리서명용 공개키가 대리 서명자 U_0에서 U_i 까지의 공개정보 K_i 와 공개키 y_i 를 이용하여 위임정보를 생성해야만 하므로 검증자는 대리 서명용 키 ρ_i를 사용하여 서명되었음을 확인할 수 있다. 그리고 두 번째로 만약, U_{j+1}이 U_{j-1}과 협력하여, U_{j+1}가 U_j의 비밀 서명 정보 대리 서명키(σ_i)를 U_{j-1}에게 주더라도 U_j의 랜덤수 k_j를 알 수 없으므로 U_j의 비밀키 x_j를 알 수 없다[11]. 더불어 제안하는 방식에서 전자 환경에 좀 더 안전하고 융통성 있는 응용을 위해 다음과 같은 부분에 특징을 가지고 있다.

(1) 위임 환경 재설정

제안 방식은 대리 서명 방식을 다단계로 확장한 대리 서명 방식이다. 이를 전자 환경에 적용하기 위해서는 각 단계마다 서명자의 상황에 맞는 환경을 재설정할 수 있어야 할 것이다. 예를 들어 원 서명자가 위임 서명 생성 기간을 10일이라고 지정해서 대리 서명자에게 위임했다고 하자. 만약 지정된 대리 서명자가 10일 중에 5일동안만 그 역할을 수행할 수 없다면, 다음 대리 서명자에게는 5일간만 위임 능력을 위임시키고, 나머지 5일간은 자신이 원 서명

자를 대신하면 된다.

이전의 방식[11]에서는 위임 정보 내용에 대리 서명용 키만 있으므로 이런 기능을 제공할 수가 없다. 본 제안 방식에서는 위임 정보에 유효 기간을 명시할 수 있을 뿐만 아니라, 각 단계마다 대리 서명자의 상황에 맞는 위임 내용을 추가할 수 있으므로, 좀 더 융통성있게 응용될 수 있을 것이다.

(2) 타당한 서명자 인증

만약 원 서명자(A)가 지정한 대리 서명자(B)가 다른 사람(C)에게 원 서명자가 전송해 준 정보를 준다고 가정해보자. 이전의 방식[11]에서는 C도 같은 방식으로 타당한 대리 서명키를 생성할 수 있으므로, 검증자는 C가 타당한 서명자인지 아닌지 결정할 수 없을 것이다. 제안 방식에서는 위임 정보에 대리 서명자의 신원을 지정함으로, 검증자는 대리 서명자의 흐름이 타당한 지 아닌지 확인할 수 있을 것이다.

4. 이동통신환경에 적합한 방식으로 확장

최근 무선 이동통신의 발전을 기반으로 많은 사용자들이 이동통신서비스를 통해 그 편리성과 유효성을 인지하고 있다. 이러한 유/무선 통신상에서 통신 상대방을 인증하고 메시지의 무결성을 보장하는 데 있어 가장 각광을 받고 있는 방식 중에 하나가 "전자서명"이다. 그러나 공개키 암호화 기법에 기초하는 전자 서명 방식을 무선 이동통신에 적용하는 것은 상대적으로 많은 시간을 필요로 한다.

또한, 이동통신에서 도청자나 그 밖의 신뢰되지 못한 요소들로부터 위조나 불법적 변경 등과 같은 위협들에 대해서는 취약성을 지니고 있고, 사용자 인증 및 부인봉쇄 등과 같은 문제는 이동통신에서 발생할 수 있는 안전성과 관련하여 여러 가지 문제를 발생시킬 수 있다.

4.1 무선 이동 통신상에서 필요한 요구사항

무선 이동통신 상에서 수행되는 다양한 응용분야에서 신뢰성과 효율성을 제공하기 위해 어떠한 요소들이 요구되며, 그 특징이 무엇인지 살펴본다[12].

무선 통신상의 정보교환을 위해서 필요한 요구사항을 기술한 것이다.

(1) 사용자 기밀성

무선 이동 통신을 통해 송신자가 메시지를 송신할 경우 제3자의 도청으로부터 자신의 신원을 보장하기 위하여 안전하고 정확한 방법으로 정당한 수신자에게 전송되어야 한다. 이를 위해서 사용자 기밀성이 요구되며 다양한 기법들을 적용할 수 있다.

(2) 인증성

메시지 송.수신시 출처가 누구이며, 전송 도중 불법적인 제3자로부터 위조 및 변경되지 않았음을 보증하는 것으로 전자 서명 기법이 적용된다.

(3) 부인 봉쇄

메시지의 송.수신 여부에 대하여 무선 이동통신 당사자간에 부인은 방지되어야 하며 이를 위해 전자 서명 기법을 사용한다.

무선 이동통신상에서 각 통신 주체의 역할 및 환경 구성에 있어 발생할 수 있는 위협에 대해 다음과 같은 보안요소가 고려되어야 한다.

(4) 유효성

무선 이동 통신은 메시지 송.수신을 위해서 일반 네트워크에 비해 상대적으로 계산 능력이 떨어지는 무선 단말기를 사

용한다. 따라서 사용자 측면에서도 충분히 사용 가능해야 한다.

(5) 안전성

무선 이동 통신에서 메시지 송.수신에 참여하는 개체들이라 할지라도 위조 및 변조가 불가능해야 한다.

대리 서명방식은 본인의 부재 중 자신을 대신하여 서명을 수행할 수 있도록 하는 방식이다. 이는 무선 통신상에서 계산능력이 부족한 사용자 단말기의 한계를 극복하기 위해 대리 Agent에서 서명을 수행할 수 있도록 확장될 수 있다. 동시에 이 방식에서 검증자는 대리 서명자가 위임 서명자의 위임 사실을 확인할 수 있다는 특징을 가지고 있다. 따라서 대리 서명방식을 무선 이동통신에 적용할 경우, 유효성을 높일 수 있다는 장점을 가지고 있다. 그러나 서명 관련 개체들의 부정이 발생할 경우 안전성 및 신뢰성 등에 문제가 발생할 수 있다.

4.2 제안 대리서명 방식

상기 요구 사항을 만족하는 전자 서명을 위하여 다음과 같은 해결책을 제시한다. 본 제안 방식은 무선 이동 통신상에서 유효성을 획득하기 위해 대리 Agent를 도입한다. 또한 기밀성과 인증성을 만족하기 위해 대리서명 메시지를 검증자의 공개키로 암호화하여 전송한다. 동시에 서명 메시지 생성 시 대리 Agent의 비밀 정

보와 송신자의 대리 서명 의뢰 정보를 부가함으로서 부인봉쇄 및 안전성을 확보하고 있다.

(1) 원 서명자, **A**

이동 단말기를 보유한 **A**는 대리 Agent에게 서명 생성을 위한 위임 서명 정보를 다음과 같이 생성하여 위임 서명 정보 (X_{AP}, TS, ID_A, K)를 비밀리에 대리 Agent에게 전송한다.

$$k_0 \in Z_q \tag{4.1}$$

$$K \equiv g^{k_0} \bmod p \tag{4.2}$$

$$e_0 \equiv h(ID_A \| TS, K) \tag{4.3}$$

$$X_{AP} \equiv x_A \cdot e_0 + k_0 \bmod q \tag{4.4}$$

A는 K를 계산하는 그 때의 타임스탬프 TS와 자신의 신원 ID_A를 연접하여 K와 함께 해쉬 함수 계산을 한다. 그리고 자신의 개인키를 사용하여 대리 서명용 키 X_{AP}를 (4.4)와 같이 계산하여 생성한다. 여기서 대리 서명용 키 생성시 일회성을 이용함으로서 대리 Agent가 임의로 서명을 생성하는 것을 방지하고 있다.

(2) 대리 Agent , **P**

· 대리 Agent는 위임 정보를 확인하기 위해 (4.6)와 같이

수신된 정보를 이용하여 원 서명자의 정당성을 확인하고 인증한다.

$$e_0 \equiv h(ID_A \| TS, K) \tag{4.5}$$

$$g^{X_{AP}} \equiv y_A^{e_0} \cdot K \bmod p \tag{4.6}$$

· 대리 Agent는 원 서명자의 부정행위를 막기 위해 X_{AP}'를 계산하고 다음과 같이 서명하여 $(M, K, K', ID_A, ID_B, TS, S(Z))$를 검증자B에게 전송한다. 이 때, Z는 B만이 서명을 확인할 수 있도록 하기 위함이다.

$$e_1 \equiv h(ID_A \| ID_B \| TS, K \| K') \tag{4.7}$$

$$X_{AP}' \equiv X_{AP} + x_P e_1 \bmod q \tag{4.8}$$

$$k_1 \in Z_q \tag{4.9}$$

$$K' = y_B^{x_1} \bmod p \tag{4.10}$$

$$Z = h(y_B \| K' \| M) \tag{4.11}$$

$$S(Z) \equiv (x_1 - x_P - X_{AP}' \cdot Z) \bmod q \tag{4.12}$$

(3) 검증자, B

B는 위임서명 $S(Z)$를 검증하기 위해 다음과 같이 계산한다. 검증 과정은 생성된 정보와 자신의 비밀키를 이용하여 다음과 같은 식의 성립여부로 대리서명의 정당성을 확인하게 된다.

$$h(y_B \| K' \| M) \equiv Z \tag{4.13}$$

$$e_0 \equiv h(ID_A \| TS, K) \tag{4.14}$$

$$e_1 \equiv h(ID_B \| ID_A \| TS, K \| K') \tag{4.15}$$

$$Y_{AP}' \equiv y_A^{\,e_0} \cdot y_P^{\,e_1} \cdot K \bmod p \tag{4.16}$$

$$(g^{S(Z)} \cdot y_{AP}^{\,Z} \cdot y_P)^{x_B} \bmod p \equiv K' \tag{4.17}$$

서명 검증 정보는 다음과 같은 과정을 통해 그 유효성을 입증할 수 있다.

$$(g^{S(Z)} \cdot Y_{AP}^{\,Z} \cdot y_P)^{x_B} \bmod p \tag{4.18}$$

$$\equiv (g^{x_1 - x_P - X_{AP}, Z} (y_A^{e_0} y_P^{e_1} K)^Z y_P)^{x_B} \bmod p \tag{4.19}$$

$$\equiv (g^{x_1 - x_P, - X_{AP}, Z} (g^{x_A\, e_0}\, g^{x_P, e_1} g^{x})^Z y_P)^{x_B} \bmod p \tag{4.20}$$

$$\equiv (g^{x_1 - x_P - X_{AP}, Z} (g^{x_A\, e_0 + x_P\, e_1 + x_0})^Z y_P)^{x_B} \bmod p \tag{4.21}$$

$$\equiv (g^{x_1 - x_P - X_{AP}, Z} (g^{X_{AP}'})^Z g^{x_P})^{x_B} \bmod p \tag{4.22}$$

$$\equiv (g^{x_1 - x_P - X_{AP}, Z + X_{AP}, Z + x_P})^{x_B} \bmod p \tag{4.23}$$

$$\equiv (g^{x_1})^{x_B} \bmod p \tag{4.24}$$

$$\equiv K' \tag{4.25}$$

4.3 제안 방식 고찰

(1) 서명자 기밀성 확보

검증자만이 서명자의 서명을 확인할 수 있으므로 제3자의 도청에 의한 서명자 기밀성을 확보할 수 있다

(2) 인증성 제공

이동 통신에서 전자상거래를 수행할 경우 수신자 지정 서명 방식을 이용함으로서 인증성을 제공하고 있다

(3) 유효성 제공

서명생성 시 계산능력이 뛰어난 대리서명Agent를 이용하므로 유효성을 확보하고 있고 (4.18)과 같이 입증하고 있다

(4) 부인 봉쇄 가능

서명 생성 시 자신의 비밀정보와 서명자의 위임정보를 함께 포함하여 수행하므로 위임서명자의 서명생성 의뢰에 대한 부인을 방지한다.

(5) 안전성 제공

서명 생성 시 대리 서명자는 자신의 비밀정보 키를 사용하여 비밀서명정보를 생성, 전송하고 원 서명자 역시 일회용 비밀서명정보를 제공함으로서 원 서명자 및 대리 서명 Agent는

불법적인 서명 생성은 불가능하므로 안정성을 제공한다.

(6) Forward Secrecy 제공

X_{AP}' 가 노출되더라도 대리서명자의 비밀키 x_P 를 알지 못하면 키를 계산할 수 없으므로 정당성을 확인할 수 있다.

$$(g^{S(Z)} \cdot y_{AP}{}^{Z} \cdot y_P)^{x_B} \bmod p \tag{2.26}$$

$$= y_B^{S(Z)} \cdot y_B{}^{x_{AP}} \cdot Z \cdot y_B^{x_P} \bmod p \tag{2.27}$$

X_{AP}'와 X_P 가 모두 드러나면 키를 계산할 수 있지만, 두 키가 모두 드러날 경우는 X_{AP} 만 드러날 경우에 비하여 매우 낮은 확률로 발생한다.

다음은 상기 사항을 고려하여 기존의 방식과 제안방식을 비교 분석한 것이다.

방식 / 특성	대리서명	Proxy-Signcryption	제안방식
서명자 기밀성	×	○	○
인증성	○	○	○
부인봉쇄	×	×	○
유효성	○	○	○
안전성	×	×	○
Forward Secrecy제공	×	×	○

[표1] 각 방식의 비교분석

5. 결 론

컴퓨터 네트워크 및 이동통신의 발전을 통해 향후 정보화 사회는 전자 상거래 서비스들을 비롯하여 더욱 다양한 응용 서비스들이 제공될 것이다. 이러한 전자 환경시스템 하에서 좀 더 안전하고 융통성있는 효율적인 전자 서명방식의 연구는 매우 중요한 주제가 되고 있다.

본 논문에서는 위임 정보 대리서명 방식과 대리인 보호형 Proxy-signcryption방식을 확장하였다. Ariki[11]가 제안한 기존의 확장된 대리서명 방식의 하나인 다단계 대리서명 방식은 위임 서명정보에 위임순서를 부여하여 대리 서명자의 부재 시 또 다른 대리서명자를 지정함으로써 원 서명자가 위임 환경을 재 설정하는 불편함을 덜었고 각각의 비밀키를 사용함으로써 두 대리 서명자의 공격에 대한 안전성과 원 서명자의 안전성 또한 해결하였지만 위임 내용에 위임자의 정보가 포함되어 있지 않음으로 검증자는 원 서명자가 지정한 타당한 위임자임을 확인할 수 없다.

이러한 문제를 해결하기 위해 본 제안 방식에서는 대리 서명자와 유효기간을 지정한 보증 위임 대리서명 방식을 확장한 방식과 대리인 보호형 proxy-signcryption을 확장한 방식을 제안하였다. 이는 원 서명자가 지정한 대리 서명자가 그 역할을 수행할 수 없을 때 또 다시 다른 사람에게 위임해서 서명할 수 있는 다단계 대리 서명방식으로 전자 환경시스템에 좀 더 안전하게 응용될 수 있을 것이다.

또한 이동통신 상에서 효율적인 전자 서명방식을 위해 대리 서명

방식을 확장하여 적용하면 유효성은 확보하고 있으나 기밀성 및 서명자 부인봉쇄가 불가능한 경우가 발생한다. 이에 본 논문의 제안 방식은 이러한 문제점들을 해결하고 동시에 forward secrecy를 제공하는 새로운 대리 서명 방식을 제안하였다. 이를 통해 제안 방식은 기밀성과 효율성을 획득하고 있으며 동시에 인증성, 부인봉쇄 , 안전성 및 forward secrecy를 만족하고 있다.

향후 기존에 제안된 많은 서명방식들을 통하여 더욱 효율적이고 안전한 대리 서명방식의 연구가 필요하리라 판단된다.

참고문헌

[1] M. Mambo, K. Usuda and E. Okamoto, "Proxy signature : Delegation of the power to sign message", IEICE Transaction on Fundamentals, E79-A(9):1338-1354, 1996

[2] M. Mambo, K. Usuda and E. Okamoto, "Proxy signatures for Delegation Signing Operation", Proc. Third ACM Conference on Computer and Communications Security, pp.48-57, 1996

[3] T. ElGamal, "A Public Key Cryptosystem and A Signature Scheme Based on Discrete Logarithms", IEEE Trans.,pp469-472, 1985

[4] K. Nyberg and R.A. Rueppel "Message recovery for signature schemes based on the discrete logarithm problem", Advances in Cryptology EUROCRYPT'94, Lectue Notes in Computer Science 950, Springer-Verlag, pp182-193, 1995

[5] S.J.Kim, S.J.Park and D.H.Won, "Proxy signatures, revisited", Proc. of ICICS'97. International Conference on Information and Communications Security, Springer, Lecture Notes in Computer Science, LNCS 1334, pp.223-232, 1997

[6] 오수현, 김현주, 원동호 , "이동 통신 환경에서의 전자 상거래에 적용할 수 있는 Proxy-signcryption방식", 통신정보보호

학회논문지. 제10권 제2호. 2000.6

[7] C.Gamage, J.Leiwo and Y.Zheng, "An Efficient scheme for Secure Message Transmission using Procy-Signcryption", Proceeding of the Twenty Second Australasian Comprter Science Conference, Auckland, New Zealand. January 18-21, 1999

[8] 박희운, 이임영 , "이동통신에서 적용 가능한 수신자 지정 대리서명 방식 "통신정보보호학회논문지, 제11권2호, pp27-34, 2001. 4

[9] D.Park, C.Boyd and S.Moon, "Forward Secrecy and its Application to Future Mobile Communication Security" Proc.of PKC 2000,

[10] 정희윤, 이동훈, 임종인, "Forward Secrecy를 제공하는 Signcryption" ITRC Forum 2001. pp (D1)11-14

[11] S.Araki and K.Imamura, "An application of Mambo,Usuda and Okamoto Proxy Signature Schemes", International Symposium on Information Theory and Its Applications, November 5-8, 2000

[12] H.U.Park and I.Y.Lee, "A 2-pass Key Agreement and authentication for mobile communication", Poceedings of The 2000 International Conference on Electronics. Information and Communications(ICEIC2000). pp115-118, 2000

감사의 글

내 자신을 위한다는 마음으로 무작정 시작했던 대학원 생활이 벌써 2년 반이 지났습니다. 쉽게 생각하지는 않았지만 학부 생활과는 또 다른 지식을 배우기에는 부족한 점이 많았습니다. 하지만 내가 여기까지 올 수 있도록 도움을 주신 많은 분들에게 지면으로나마 감사의 마음을 전합니다.

먼저, 부족함이 많은 저에게 언제나 좋은 말씀으로 지도해 주신 박지환 교수님께 진심으로 감사의 마음을 전합니다. 부족한 저의 논문을 위해서 많은 조언을 해 주신 여정모 교수님, 박흥복 교수님께 감사드립니다. 그리고 언제나 깊은 관심으로 지켜봐 주신 박만곤 교수님, 윤성대 교수님, 정순호 교수님, 박승섭 교수님, 김창수 교수님, 이경현 교수님, 김영봉 교수님께 깊은 감사를 드립니다.

언제나 자상하게 배려 해주신 안경모 선생님과 김문수 선생님, 뒤늦게 대학원 생활의 깊은 애정과 많은 즐거움, 좋은 추억을 가질 수 있도록 도와준 윤경씨와 미애씨, 무엇보다 이 논문을 쓸 수 있도록 많은 도움을 주고 많은 애정을 느낄 수 있도록 해 준 재귀씨, 그리고 연구실 가족들인 진흥씨, 현호씨, 소진씨, 산업대학원의 장영철씨, 조계영씨, 교육대학원의 임쌍학 선생님, 민정씨, 순혜씨, 현화씨에게 모두 고마움을 전합니다.

대학원 생활의 바쁜 나날동안 물심양면으로 도와주신 회사직원들, 그리고 나를 제일 많이 이해해주고 격려해준 순자언니와 재영이, 바쁘다는 핑계로 자주 만나지도 못하면서 짜증만 내는 나에게 우정으로 지켜봐 준 친구 옥희, 미동이, 동생 미정이 ,진실이, 언제나 내가 하는 일을 이해해주고 도와준 나의 동생 상희, 관형이, 우리 제부, 올 해 우리 가족에게 가장 큰 행복과 기쁨을 준 우리 오빠와 새언니 귀여운 나의 조카들, 마지막으로 나를 있게 해주신 부모님께 감사를 드리며 이 논문을 바치고자 합니다.