대리인의 부정을 방지할 수 있는 안전한 Proxy-Signcryption에 관한 연구

김소진, 박지환
부경대학교 정보보호학과

A Study on Secure Proxy-Signcryption preventable Proxy Agent's dishonesty

So-Jin Kim, Ji-Hwan Park
Dept of Information Security, PuKyong University

요 약

디지털 서명과 암호화를 동시에 수행할 수 있는 Signcrytpion 방식이 Zheng에 의해 제안되었다[1]. 그 후, 서명과 암호화에 요구되는 작업을 상대적으로 성능이 뛰어난 서버에 의뢰 가능하도록 하는 Proxy-Signcrytpion을 이동통신 환경에 적합하도록 개선한 방식이 제안되었다[2]. 그러나 개선 방식은 효율성과 forword secrecy는 제공하지만 대리자의 부정은 방지할 수 없어 문제가 발생할 수 있다. 따라서 본 논문에서는 대리자가 부정하였을 경우 대리자의 개인키가 노출되도록 한 안전한 Proxy-Signcrytpion 방식을 제안한다.

1. 서론

최근 네트워크의 발전으로 E-mail 전송, 데이터 교환, 금융거래, 전자 상거래 등이 활발히 이루어지고 있다. 그러나 이러한 응용 서비스들은 보안상의 많은 문제점들을 가진다. 특히 보안 서비스 중 기밀성과 인증은 가장 중요한 부분이다. 그리고 네트워크를 통해 메시지를 전송할 경우, 메시지에 대한 서명과 암호화는 각각 따로 수행하기 때문에 요구되는 계산량이 많다.

이러한 문제점을 해결하기 위해 Y.Zheng에 의해 고안된 Signcryption 방식은 메시지의 서명과 암호화를 동시에 수행하도록 하여 기존의 서명-후-암호화하는 방식보다 효율적이다. 그 후, Sign cryption 방식에 요구되는 작업을 상대적으로 성능이 뛰어난 서버에 의뢰할 수 있도록 Proxy-Signcrytpion 방식이 제안되었다[3,4]. Proxy-Signcrytpion 방식은 기존의 대리 서명 방식[5,6]을 이용하므로 사용자의 계산량을 감소시킬 수 있는 장점이 있다. 제안된 대부분의 Proxy-Signcrytpion 방식들은 대리인 보호형 방식과 forward secrecy의 제공에 관한 것이며 대리자의 부정에 대해 고려되지 않았다[2,3,4].

따라서 본 논문에서는 적은 계산량으로 서명과 암호화를 수행하면서 기존의 Proxy-Signcryption 방식을 개선하여 대리자의 부정을 방지할 수 있도록 제안한다.

2. 기존 방식

본 장에서는 이동 통신 환경에 이용할 수 있는 수신자 지정 서명 방식과 대리인 보호형 서명방식을 적용하고 forward secrecy를 제공하는 Proxy-Signcrytpion 방식을 분석한다[2].

[시스템 설정]

- p : 512비트 이상의 큰 소수
- q : $q \mid p-1$인 큰 소수
- g : 위수가 q인 Z_p상의 원소
- $x_A \in Z_q$, Alice의 비밀키
- $y_A \equiv g^{x_A} \pmod{p}$, Alice의 공개키
- $x_B \in Z_q$, Bob의 비밀키
- $y_B \equiv g^{x_B} \pmod{p}$, Bob의 공개키
- $x_P \in Z_q$, Proxy agent의 비밀키
- $y_P \equiv g^{x_P} \pmod{p}$, proxy agent의 공개키
- $KH()$: Keyed 해쉬 함수
- $E()$ / $D()$: 관용암호/ 복호 알고리즘
- $h()$: 일방향 해쉬 함수

- S : 위임서명자의 일회용 위임키
- T_i : time stamp(실시간 값)
- T' : Proxy agent가 암호문을 받는 시간
- ΔT : 채널의 최대 지연시간
- m : 메시지

1) 위임키 생성

Alice는 proxy agent에게 서명생성을 위한 위임키와 관계된 값들을 미리 계산 (R, K, S)하여 휴대폰 단말기나 스마트 카드에 저장한다. 이는 연산 부하량과 시간을 단축시킬 수 있다.

① $x \in Z_q$

$R \equiv g^x \bmod p$

$K \equiv y_P^x \bmod p$

② 위임키 생성

$S \equiv (x_A + x \cdot R) \bmod p-1$

③ 메시지와 위임키를 암호화

$C \equiv E_K(m \| S \| T_i)$

Alice는 메시지를 전송하고자 할 때, 사전 저장되어 있는 K, R, S 값을 이용하여 C를 계산한 후, Proxy agent에게 (R, C)를 전송한다. Alice는 위의 과정을 반복 수행할 필요 없이 필요한 경우 C만 수행하면 된다. 또한 K에서 g^x를 계산하기 위해서는 proxy의 비밀 서명정보 x_P를 알아야 하므로 지정된 수신자만이 이를 복호화할 수 있고 R을 이용하여 암호문 C를 복호해야 한다.

2) 위임정보의 확인 및 변환

Proxy agent는 수신된 정보 (R, C)로 세션키 K를 계산하고, 계산된 K로 암호문을 복호하여 time stamp(T_i)를 다음과 같이 검증한다.

① $K \equiv R^{x_P} \bmod p$

② $m \| S \| T_i \equiv D_K(C)$

$T' - T_i \leq \Delta T$

만약 검증이 만족되지 않으면 요청을 거절하고, 만족하면 위임 서명자의 정당성을 확인하고 다음과 같은 x_{AP}를 생성한다.

① $g^S \equiv (y_A \cdot R^R) \bmod p$

② $x_{AP} \equiv S + x_P y_P \bmod q$

3) Proxy agent에 의한 Signcryption 생성

① $x' \in Z_q$

$K_1 \equiv h(y_B^{x'} \bmod p)$

$K_2 \equiv h(g^{x'} \bmod p)$

② $H \equiv KH_{K_2}(m)$

③ $s \equiv x' - (x_P + x_{AP} \cdot r) \bmod q$

④ $c \equiv E_{K_1}(m)$

Signcryption 메시지 (c, r, s, R)을 Bob에게 전송한다.

4) Proxy-Signcryption 검증

Bob은 공개 위임값을 검증($y_{AP} \equiv y_A \cdot y_P^{y_P} \cdot R^R \bmod p$) 하고, 다음과 같이 메시지를 복호한다.

① $t_1 \equiv (y_{AP}^r \cdot y_P \cdot g^s) \bmod p$

② $t_2 \equiv t_1^{x_B} \bmod p$

③ $K' \equiv h(t_1)$

④ $k \equiv h(t_2)$

⑤ $m \equiv D_k(c)$

단, $KH_K(m) \equiv r$인 경우에만 정당한 Signcryption으로 받아들인다.

위의 방식은 수신자 Bob이 y_{AP}를 계산하는 과정에서 Alice의 공개키 y_A와 proxy agent의 공개키 y_P를 동시에 사용하므로 Alice의 위임에 의해 proxy agent가 생성한 Proxy-Signcryption임을 확인 가능하고, 대리 서명 위임자의 위조 공격으로부터 대리 서명자를 보호할 수 있는 forword secrecy를 제공한다. 그러나 위임 서명자의 일회용 비밀 정보인 S는 대리자에 의해 여러 번 반복 사용할 수 있다. 즉 S를 사전에 미리 계산하여 필요시 같은 값을 그대로 사용하기 때문에 대리자 자신의 부정은 막지 못한다. 따라서 대리자가 S값을 한번만 사용할 수 있도록 다음과 같이 제안한다.

3. 제안 Proxy-Signcrytpion 방식

제안 방식은 대리자의 부정을 방지할 수 있도록 일회용 대리 서명 방식[7]을 적용하였다. 그리하여 본 방식은 사용자의 익명성을 요구하는 전자 시스템 환경에도 적용 가능하며 이동 통신 환경으로도 확장 가

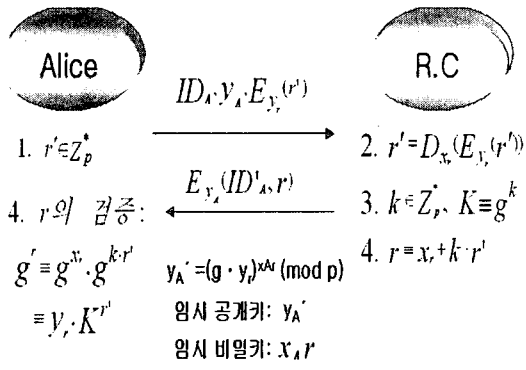
능하다.

[시스템 설정]

- x_A, y_A: Alice의 비밀키, 공개키
  $x_A \in Z_q^*$, $y_A \equiv (g \cdot y_r)^{x_A} \pmod{p}$
- x_B, y_B: Bob의 비밀키, 공개키
  $x_B \in Z_q^*$, $y_B \equiv (g \cdot y_r)^{x_B} \pmod{p}$
- x_{P_1}, x_{P_2}, y_{P_1}, y_{P_2}: Proxy agent의 비밀키, 공개키
  $x_{P_i} \in Z_q^*$, $i \in \{1,2\}$
  $y_{P_i} \equiv (g \cdot y_r)^{x_{P_i}} \pmod{p}$
- ID_A, ID_P, ID_B: Alice, Proxy agent, Bob의 ID
- ID'_A: Alice의 임시 ID
- x_r, y_r: 등록센터의 비밀키, 공개키
  $x_r \in Z_q^*$, $y_r \equiv g^{x_r} \pmod{p}$

1) 등록

Alice는 임시 비밀키/공개키 쌍을 얻기 위해 그림1과 같은 과정을 수행하여 임시 ID인 ID'_A를 얻고, r 값을 검사하여 $g^r \equiv y_r \cdot K^{r'}$이 검증되면 $x_A r$, y_A'를 생성한다. 이때 r의 유효 기간에 따라 등록 횟수는 달라진다. 만약 r의 유효기간을 설정한다면, r을 생성할 때 유효기간 정보 값도 함께 포함시킨다. 그리고 r의 유효 기간을 설정하지 않는다면, 등록은 한번만 하면 된다. 그러면 다음 등록 단계는 생략하고, 대리 서명 위임 단계를 수행하면 된다. 끝으로 등록 센터는 자신의 비밀 DB에 ID_A, y_A, $E_{y_r}(r)$를 저장하고, 모든 임시 공개키는 공개한다.



(그림1) 임시 비밀키/공개키 생성 과정

2) 위임키 생성

Alice는 등록 단계에서 생성한 ID'_A, $x_A r$, y_A'를 가지고 서명을 위임하기 위해 다음과 같이 위임키 S를 생성하고 메시지는 세션키로 암호화하여 proxy agent에게 전달한다. 이때 세션키는 미리 계산하여 비밀로 저장 가능하고 Dffie-Hellman의 키 사전 분배 방식을 이용한다.

① 세션키 계산(사전 작업)

$sk \equiv (y_{p_1} \cdot y_{p_2})^{x_A r} \pmod{p}$

② $k \in Z_{p-1}$

$K \equiv (g \cdot y_r)^k \pmod{p}$

③ ID'_A, ID_P, K로 다음을 계산(송/수신자 지정)

$e = h(ID'_A, ID_P, K)$

④ 임시 비밀키 $x_A r$로 위임키 생성

$(S \equiv x_A r \cdot e + k \bmod p) \bmod q$

⑤ 메시지 암호화

$C \equiv E_{sk}(m)$

⑥ Proxy agent에게 ID'_A, K, S, C을 전송

이때 ID'_A, K, S은 암호화하지 않는다. S의 값은 송/수신자 정보를 포함하여 다른 제 3자는 사용할 수 없기 때문이다.

3) 위임정보의 확인 및 변환

Proxy agent는 세션키 sk를 계산하여 sk로 암호문(C)을 복호화하고 수신된 정보(ID'_A, K, S)로 정당한 위임임을 검증한다.

① 세션키 생성(DH의 키 사전 분배 방식)

$sk \equiv y'^{(x_{p_1} + x_{p_2})}_A \pmod{p}$

② 위임키 검증(검증되지 않으면 수행중단)

$(g \cdot y_r)^S \equiv y_A'^{\,h(ID'_A, ID_P, K)} \cdot K \pmod{p}$

(y_A'의 정당성은 등록 센터에서 확인 가능)

③ 대리 서명키 S' 생성

$S' \equiv S + x_{P_1} + x_{P_2} \pmod{q}$

④ 대리 서명에 대한 공개키 β 계산

$\beta \equiv (g \cdot y_r)^S \pmod{p}$

4) Proxy agent에 의한 Signcrytpion 생성

① Signcryption을 위한 대리키 생성

$K_1 \equiv h(y_B^S \bmod p)$

$K_2 \equiv h(\beta \bmod p)$

② $H \equiv KH_{K_2}(m)$

③ 메시지 서명

$\sigma \equiv S' \cdot H + x_{P_1} - x_{P_2} \pmod{q}$

④ 메시지 암호화

$c \equiv E_{K_1}(m)$

Signcryption 메시지 정보 $(ID'_A, ID_P, C, \sigma, H, K, \beta)$을 Bob에게 전송한다.

5) Proxy-Signcryption 검증

Bob은 전달받은 값들을 검증하여 proxy agent가 생성한 값들에 대한 인증을 수행한다.

① 서명 공개키를 검증하여 proxy agent가 서명키를 정당하게 생성했는지 검증

$e = h(ID'_A, ID_P, K)$

$\beta \equiv y_A'^{e} \cdot K \cdot y_{P_1} \cdot y_{P_2}$

② Proxy-Signcrytpion 검증

다음과 같이 키를 복구하여 메시지를 복호한다.

$K_2 = h(\beta \bmod p)$

$K_1 = h(\beta^{x_b} \bmod p)$

$m \equiv D_{K_1}(C)$

단, $H \equiv KH_{K_2}(m)$인 경우에만 정당한 Signcryption으로 받아들인다.

제안 방식은 서명 공개키 β의 검증에서 Alice와 proxy agent의 키가 사용됨으로 Alice의 위임에 의해 proxy agent가 proxy-signcryption을 생성한 것임을 확인 가능하다.

4. 제안 방식의 고찰

본 논문에서 제안한 proxy-signcryption을 이동 통신 환경과 익명성을 요구하는 응용 서비스 환경에 적용할 수 있게 다음과 같은 요구사항들을 기준하여 고찰한다.

(1) **익명성** - 임시 비밀키/공개키 쌍으로 사용자의 신분을 보호한다.

(2) **인증성** - 수신자는 서명 공개키를 검증하고, 대리자는 위임키를 검증하여 정당성을 확인한다.

(3) **기밀성** - 제안 방식은 지정된 수신자만이 메시지를 확인 할 수 있고, 정당한 제 3자(중재자)에게 송신자의 서명을 검증하여도 메시지의 기밀성을 확보한다.

(4) **효율성** - 상대적으로 계산능력이 뛰어난 대리자가 송신자를 대신하여 signcrytpyion 작업을 수행함으로 송신자측의 계산량을 줄여 효율성을 확보한다.

(5) **부인봉쇄** - 대리서명 생성시 송신자와 대리자의 비밀정보를 포함하므로 서명생성에 대한 서로의 부인방지가 가능하다.

(6) **안전성** - 제안 방식은 전체적으로 이산대수 문제의 어려움에 기반하여 안전성을 확보한다.

▶ 안전성 분석

메시지 송/수신에 참여하는 송신자, 대리자, 수신자 및 공격자는 서로에 대한 위조 및 변경 등의 부정을 할 수 없다.

- 송신자만이 위임키를 생성할 수 있다. 위임키에 대한 안전성은 이산대수 문제의 어려움에 기반하여 원 서명자의 비밀키를 모르면 생성할 수 없다.
- 원 대리서명자만이 대리서명을 할 수 있다. 서명키의 안전성도 이산대수 문제의 어려움에 기반함으로 대리자의 비밀키를 모르면 키를 생성할 수 없으며, 서명도 불가능하다.
- 2번 이상 사용하면 다음과 같이 비밀키가 노출되기 때문에 대리 서명자의 서명이 1회성임을 보장한다.

$\sigma \equiv s' \cdot H + x_{P_1} - x_{P_2}$ ------①

$\sigma' \equiv s' \cdot H' + x_{P_1} - x_{P_2}$ -----②

- 식 ① - ②를 하면 다음과 같다.

$(\sigma - \sigma') \equiv s' \cdot (H - H')$

여기서 서명키 s'가 노출된다. 그리하여 식 ①에서 $x_{P_1} - x_{P_2}$의 값도 알 수 있고, $s' \equiv s + x_{P_1} + x_{P_2}$에서 $x_{P_1} + x_{P_2}$도 알 수 있어 호스트의 비밀키 x_{P_1}, x_{P_2}도 노출된다. 따라서 대리자의 부정을 방지할 수 있다.

- 서명단계는 실패-중단 서명기법의 안전성과 동일하다.
- 이산대수 문제의 어려움에 기반함으로 공격자의 서명의 위조는 등록 센터의 비밀키 x_r를 알아야 한다. 즉, 다음의 계산 가능성은 $\log_g y_r$를 계산할 수 있어야 한다.

$\beta^H \cdot y_{P_1} \cdot y_{P_2}^{-1} \equiv g^{\sigma} \cdot y_r^{\sigma} \equiv g^{\tau} \cdot y_r^{\tau}$

$\therefore g^{(\sigma - \tau)} \equiv g^{x_r(\tau - \sigma)} \quad (x_r \equiv (\sigma - \tau)(\tau - \sigma)^{-1})$

- 메시지의 복호화는 지정된 수신자만이 가능하다. 즉, 키를 복구하는 것은 수신자의 비밀키 x_b를 알아야 한다.

$K_1 \equiv h(y_B^S \bmod p) \equiv h(\beta^{x_A} \bmod p)$

- 원 서명자의 익명성을 보장한다. 이산대수 문제의 어려움에 기반하여 임시 공개키 $y'_A \equiv (g \cdot y_r)^{x_A r}$에서 $x_A r$은 오직 원 서명자만 알고, 원 공개키 y_A는 등록센터만 안다.
- 디지털 서명기법은 '서명자의 인증' 조건이 있다. 즉, 누구든지 검증할 수 있어야 한다는 것이다. 본 방식은 송신자의 익명성을 정당하게 제공함으로 송신자를 보호하면서 누구든지 서명을 검증할 수 있다.

표1은 제안 방식을 기존 proxy-signcrption 방식들과 비교한 것이다. Proxy-signcryption[3]은 서명자의 기밀성과 인증을 제공한 방식이고, 대리인 보호형[4]은 원 서명자가 직접 대리자의 대리키를 이용한 대리서명을 방지하는 부인봉쇄 기능을 추가한 방식이다. 그리고 forward secrecy 제공형[2]은 대리자와 수신자 사이에서 전송되는 메시지의 기밀성을 보장하여 안전성과 forward secrecy를 제공하고 있다. 그러나 이 방식들은 앞에서 지적했듯이 대리자의 부정에 대한 문제점이 있다.

5. 결론

본 논문에서는 기존의 proxy-signcryption 방식들을 분석하여 대리자의 부정 방지와 익명성의 기능을 추가하였다. 제안 방식은 일반적인 메시지 서명 후 암호화하는 방식보다 효율적이며 기존의 방식들과 비교해서 계산량도 크게 차이가 없다. 따라서 다양한 응용 서비스 환경에 적용 가능할 것이다.

<표1> 각 방식별 특성 비교(EXP: 모듈라 멱승, ENC: 관용암호방식, [ ] : 사전 작업(오프라인))

구 분	서명자 기밀성	인증성	부인봉쇄	효율성	안전성	forward secrecy	구성요소 개수	대리자의 부정방지	익명성 제공	Alice의 계산량
Proxy-signcryption[5]	O	O	×	O	×	×	3	×	×	ENC: 1 EXP: 1
대리인 보호형 [6]	O	O	O	O	×	×	4	×	×	ENC: 1 EXP: 1
forward secrecy 제공형[2]	O	O	O	O	O	O	3	×	×	ENC: 1 [EXP: 2]
제안 방식	O	O	O	O	O	3	3	O	O	ENC: 1 EXP: 1 [EXP: 1]

[참고문헌]

[1] Y.Zheng, "Digital Signcryption or How to Achieve Cost (Signature & Encryption) << Cost(Signature)+Cost(Encryption)", Advances in Cryptology - CRYPTO'97, Springer-verlag, LNCS 1294, pp.165-179, 1997

[2] 김동우, "이동통신 환경에 적합한 효율적인 Proxy-Signcryption 방식에 관한 연구", 2003년 5월 한국 멀티미디어학회 논문지 게재 예정.

[3] C.Gamage, J.Leiwo and Y.Zheng, "An Efficient Scheme for Secure Message Transmission Using Proxy-Signcryption", Proc. of the 22nd Australasian Computer Science Conference, Jan. 1999

[4] 오수현, 김현주, 원동호, "이동통신 환경에서의 전자상거래에 적용할 수 있는 Proxy-Signcryption 방식", 한국정보보호학회 논문지, 제10권 제2호, 2000.6

[5] 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

[6] M.Mambo, K. Usuda and E. Okamoto, "Proxy Signatures for Delegation Signing Operation", Proc. Third ACM Conference on Computer and Communications Security, pp48-57, 1996

[7] 김소진, 최재귀, 박지환, "익명성을 갖는 효율적인 1회용 대리서명", 한국정보처리학회 2002년 추계학술대회, 2002.11