사용자 결제정보보호를 위한 NTRU 기반 환 서명 기법+

박성욱, 고성종, 이해각, 이임영
순천향대학교 컴퓨터소프트웨어공학과
e-mail:swpark@sch.ac.kr, sjgo@sch.ac.kr, lhk7083@sch.ac.kr, imylee@sch.ac.kr

Ring Signature Scheme based on NTRU for the Protection of User Payment Information

Sung-Wook Park, Sung-Jong Go, Hae-Kag Lee, Im-Yeong Lee
Dept of Computer Software Engineering, Soonchunhyang University

요 약

최근 개인정보 노출에 의한 다양한 사건, 사고 발생에 의해 개인정보보호에 관련된 많은 이슈들이 문제시 되고 있다. 특히 과금, 결제에 활용되는 금융정보 노출 문제는 사용자들의 금전적인 피해를 발생시킬 수 있다. 이와 같은 문제점을 해결하기 위해 높은 암호학적 강도를 가진 암호알고리즘을 적용한다 하더라도 다양하고 끊임없는 공격에 의해 결국 사용자의 신원 또는 금융 결제 정보가 노출될 가능성을 가진다. 최근 한국인터넷진흥원에서 발표한 "NFC 개인정보보호 대책 최종보고서"에 따르면 개인정보 암호화를 부분적으로 미지원하거나 불필요한 개인정보의 과도한 수집 및 저장 등이 문제점으로 제기되었으며 Google사의 Google Wallet 서비스의 개인정보 유출 사고 또한 이러한 문제점을 뒷받침하는 근거가 되고 있다. 본 논문에서는 기존에 서비스되고 있는 NFC 모바일 결제 서비스 상에서 결제정보의 이동 경로 별 결제 기술을 분석한다. 또한 가장 높은 등급의 모호성을 제공하는 환 서명을 이용하여 결제정보를 직접적으로 사용하지 않고 결제자를 증명할 수 있는 NTRU기반 환 서명 인증 기법에 대해 제안한다.

1. 서론

최근 개인정보보호에 관련된 많은 이슈가 문제시되고 있다. 온라인으로 서비스를 제공하는 기업들도 서비스 제공에 필요한 수준의 정보만을 요구하고 있는 실정이며, 그 이상의 정보는 요구하지 않는 경우가 늘어나고 있다. 국내 일부 포털 사이트에서 회원가입 시 주민등록번호를 요구하지 않으며 기존에 수집한 주민등록번호를 폐기하는 등의 사례를 예로 들 수 있다. 이처럼 개인정보는 사회적으로 중요한 이슈로 자리 잡게 되었다. 개인정보보호가 중요시 되는 현시점을 고려할 때, 금융정보 노출 문제는 커다란 문제점을 가진다고 볼 수 있다. 이처럼 높은 암호학적 강도를 가진 암호알고리즘을 적용한다 하더라도 다양한 공격에 의해 사용자의 신원 또는 금융 결제 정보가 노출될 가능성을 가진다. 따라서 사용자 인증에 있어서도 본인의 정보를 노출시키지 않는 익명 인증기술의 적용을 생각해볼 수 있다. 익명 인증기술이란 당사자와 관련된 정보를 제공하지 않으면서 당사자가 합당한 자격을 가지고 있음을 증명하는 기술로, 기존의 사용자 인증에서 식별정보를 제외하여 사용자의 익명성을 높이고 제3자가 인터넷으로 전송되는 데이터를 보는 경우 해당 데이터와 사용자 간의 연결성(linkability)을 없애는 것이다. 초기의 익명 인증기술은 전자화폐나 전자투표에서의 활용을 위하여 주로 연구되었으나, 최근에는 개인정보를 보호하기 위한 인증방법으로 연구가 진행되고 있다. 익명 인증기술은 여러 기법이 있어 각 기법이 제공하는 익명성 수준에 따라 이를 구분할 수가 있다. 익명성 수준은 익명성과 관련된 정보의 노출 정도에 따라서 결정되는데, 관련 정보는 크게 두 가지로 신원 정보와 연결 정보로 나눌 수 있다. 신원 정보는 이름, 주민등록번호, 이메일 주소 등 스스로를 증명하려는 사람을 유일하게 하는 정보이며, 연결 정보는 당사자가 인증절차를 처리하였는지에 관한 정보이다. PKI 기반 전자서명을 예로 들면 인증서에 표시된 이름은 신원 정보라고

익명성 수준	기준 정보		해당 익명 인증기술
	신원 정보	연결 정보	
0	완전노출	-	PKI-전자서명 인증
1	노출없음	완전노출	가명 기반 익명인증(I-PIN 등)
2	조건부노출	조건부노출	ETRI 다수준 인증
3	조건부노출	노출없음	그룹서명 기반 인증
4	노출없음	조건부노출	신용장(credential) 기반 인증
5	노출없음	노출없음	영지식 증명, 환서명 기반 인증

(그림 1) 익명성 수준 및 관련 인증기술

+ 이 논문은 2012년도 한국산업기술재단의 재원으로 지역산업기술개발사업의 지원을 받아 수행된 연구임 (과제번호:A001100086)

할 수 있으며, 인증서를 이용하여 생성된 서명값은 자체만으로 사용자에 대하여 알기 힘든 대신 서명값이 동일한 키로 생성되었는지에 대하여 알 수 있는 연결 정보라고 할 수 있다. 기주희 등은 두 정보를 완전노출, 조건부노출, 노출 없음으로 구분하여 익명성 수준을 6단계로 구성하였다[1]. 본 논문에서는 이 중 가장 높은 등급의 정보 모호성을 제공하는 환 서명 기반의 인증기법에 대하여 분석하고 NFC 환경에서 개인정보보호를 위한 새로운 사용자 인증기법을 제안한다. 본 논문의 구성은 다음과 같다. 2장에서 본 논문에서 제안하는 기법의 이해를 돕기 위한 관련 기술의 소개와 관련 연구를 분석하고 3장에서는 기존연구를 기반으로 제안방식에서 요구되는 보안요구사항에 대하여 분석한다. 4장에서는 보안요구사항을 만족하는 제안방식을 기술하며, 5장에서는 보안요구사항에 의한 제안방식을 분석한다. 마지막으로 6장에서 결론을 맺는다.

2. 관련연구

본 장에서는 최근 국외에서 상용화된 Google wallet 서비스에 대해 설명하고, 기존의 위임 서명 방식 분석과 이를 NFC 결제 환경에 적용했을 경우 발생할 수 있는 취약점을 분석한다.

2.1 Google Wallet

Google Wallet 서비스[2]는 Google 사에서 제공하는 NFC 스마트폰 기반의 결제 서비스로 현재 미국 일부지역, 일부 상점에서 시범 서비스가 진행 중이다. 삼성의 '넥서스S'를 시작으로 서비스 이용 가능 단말은 추후 지원 기종 및 통신사가 확대 될 예정이며, 이를 활용한 다양한 서비스가 예상된다. 결제에 관련하여 Citi Bank가 참여하고 사용자에 대한 다양한 개인정보를 각 금융·카드사, 이통사, 상점 등에서 수집을 수행한다. Google사 측에서 모든 신용카드 정보가 NFC 통신 결제 규격인 M/Chip 4(Mobile MasterCard(R) PayPass™ M/Chip 4)[5]에 의해 암호화 되고 이중 삼중의 보안 절차를 거치므로 물리적인 지갑보다 안전하다고 주장하고 있으나 검증된바 없으며 최근 '탈옥'하지 않은 정상적인 기기에서도 개인정보가 새나가는 문제가 수차례 발견됨에 따라 서비스의 일시적인 중단>패치>재개를 반복하고 있는 실정이다. 현재 Google Wallet 서비스에 적용된 PayPass의 표준 기술은 신용카드를 위한 표준 기술로서 NFC기반의 모바일 결제 흐름과 기존 신용카드 결제 흐름 자체가 유사하다. 하지만 NFC와 모바일이 융합된 NFC기반 결제환경에서는 기존의 신용카드 기반 서비스에서 제공할 수 없는 다양한 형태의 서비스 제공이 가능하므로 그 특성에 맞는 새로운 형태의 보안 기술들이 부가적으로 필요할 것으로 예상된다.

2.2 NTRU

1996년 Crypto의 럼프세션에서 Jeffrey Hoffstein 등에 의해 소개된 NTRU는 격자 문제를 기반으로 하는 공개키 암호 체계로 기본 연산은 다항식 환(Polynomial rings)상에서 이루어진다. 현재 IEEE에서 P1363.1로 격자 문제를 기반으로 하는 공개키 암호 표준으로 고려되고 있는 NTRU는 기존 공개키 암호 RSA, ECC(Elliptic Curve Cryptography) 등과 비교하여 동일한 안전성을 제공하면서 암·복호화 속도가 빠르다는 이점을 갖는다. NTRU는 이동 통신 사용자의 수가 점점 증가하고 전자 거래와 주식투자와 같은 높은 보안성을 요구하는 서비스가 환경에서 적합한 암호 시스템이라고 할 수 있다[3].

2.3 환 서명

환 서명(Ring Signature)은 2001년 Ron Rivest, Adi Shamir, and Yael Tauman에 의해 처음 소개되었는데, Ring Signature라는 용어는 서명 알고리듬의 구조가 "ring"의 구조와 유사하다는 점에서 나온 것이다[4]. 즉, 모든 그룹 멤버들이 동일한 위치에 있고, 중심이 되는 멤버가 따로 존재하지 않는다. 그룹 구성원 각각이 공개키/비밀키 쌍(PK1, SK1), (PK2, SK2), …, (PKn, SKn)을 가지고 있다고 가정하자. 환 서명 σ는 그룹 구성원 모두의 공개키와 서명자의 비밀키, 그리고 서명할 메시지를 이용하여 계산될 수 있다. 그리고, 그룹의 모든 멤버들은 서명값 σ, 메시지 m, 그리고 포함된 공개키 쌍 PK1, …, PKn이 주어졌을 때, 서명의 유효성을 확인할 수 있다. 환 서명이 적절히 계산되었다면 서명 값이 제대로 확인되어야 한다. 반면에, 그 그룹 멤버들의 비밀키들을 알지 못한다면 누가 서명을 생성하였는지를 알 수 없다. 환 서명의 구조는 ring-sign과 ring-verify의 두 가지 과정으로 구성되어 있다[6].

- Ring-sign(m, PK1, PK2, …, PKn, i, SKi): n명의 그룹 멤버들의 공개키와 i번째 그룹 멤버의 비밀키 SKi를 가지고 메시지 m에 대한 서명 값 σ를 생성하는 과정
- Ring-verify(m, σ): 메시지 m과 모든 그룹 멤버들의 공개키가 담긴 서명 σ를 이용하여 서명을 검증하는 과정

키를 가진 그룹 멤버들 중 어떤 사람에 의해서도 서명할 수 있고, 따라서 환 서명을 이용하여 서명된 메시지는 특정 그룹 멤버들 중 누군가에 의해서 보증된다. 환 서명은 그룹 내의 어떤 멤버의 키를 이용하여 서명되었는지를 알기가 어렵다는 특징이 있다. 환 서명은 그룹 서명과 비슷하지만 다음 특성들에서 차이가 있다.

- 그룹 매니저가 없으며, 서명자가 공개하기 전까지 익명성을 폐기할 수 없다.
- 어떤 그룹의 사용자든 추가적인 셋업 과정 없이 다른 그룹의 멤버가 될 수 있다.

2.4 Rivest-Shamir 환 서명 방식

Rivest-Shamir(RS) 환 서명 방식은 제 3 기관이 관여하는 부분이 전혀 없으므로 익명성을 제 3 기관이 제공하

지 않고 서명구조 자체에서 익명성을 만들어낸다. 이로 인해 서명에 이용되는 인증서의 유효성은 판단해줄 수 있으나 해당 서명을 누가 생성한지는 알아낼 수가 없다. 또한 서명자가 그룹에 속한 사람들의 공개키만 알면 서명할 수 있으므로 그룹 멤버를 추가하거나 삭제하는 과정이 필요 없지만, 서명 값에 그룹에 속한 멤버들의 공개키가 포함되어야 하므로 멤버의 수가 많아지면 서명 값도 이에 비례하여 길어진다는 단점이 있다. 즉, 트랩도어를 이용한 결합 함수 생성 시 멤버의 수만큼의 지수승 연산이 필요하다.

3. 보안요구사항

NFC 모바일 결제환경에서 결제정보보호를 위한 환 서명 기법이 기본적으로 만족해야할 특성과 더불어 기존 방식의 문제점인 효율성을 개선해야 한다. 따라서 NFC 모바일 결제환경에서 NTRU기반 환 서명 기법의 보안요구사항은 다음과 같다.

- 위조 불가성 : 링 멤버가 아닌 비인가자가 링 서명을 위조하는 것이 불가능해야 한다.
- 서명자 모호성 : 링 멤버의 총 인원이 n일 경우, 검증자가 성공적으로 링 서명의 실제 서명자를 확인할 수 있는 확률은 1/n 이어야 한다.
- 안전성 : 기본적인 환 서명 프로토콜의 요구사항을 만족하며 비밀정보에 대해 높은 안전성을 유지할 수 있어야 한다.
- 효율성 : 제한된 디바이스 환경을 위해 연산량 측면에서 효율성이 높아야 한다.

4. 제안방식

이 장에서는 다항식의 컨볼루션 곱 연산의 특성을 이용하여 NTRU 기반 환 서명 기법을 제안한다. 이전 장에서 사용되는 NTRU 파라미터를 그대로 이용하되 사용자의 비밀키에 해당하는 f, g의 생성방법을 변경하여 사용한다. 기존 방식의 경우 잘려진 다항식 환 r에서 2개의 작은 다항식 f, g를 선택하며 이때, f만이 f의 역을 계산하였다. 하지만 본 제안에서 g의 역 또한 필요하므로 정밀한 작은 다항식 선택을 위해 사용되는 f의 파라미터 설정을 g 생성 시에도 적용한다.

4.1 시스템 계수

본 제안방식에서는 다음과 같은 시스템계수를 사용하여 프로토콜을 설계한다.

- $*$: 개체 (A : 증명자(User), B : 검증자(Bank))
- N : 잘려진 다항식 환 $R=Z[X]/(X^N-1)$의 차수를 정하는 차원 파라미터 값(N=소수)
- p, q : $\gcd(p,q)=1$을 만족하는 큰 소수
- f_*, g_* : $*$의 비밀키 다항식, $f_* \in L_f, g_* \in L_g$
- ${f_*}_p^{-1}, {f_*}_q^{-1}$: $*$의 f 역함수
- ${g_*}_p^{-1}, {g_*}_q^{-1}$: $*$의 g 역함수
- v_* : $*$의 공개키, $v_* = p{f_*}_q^{-1} \bullet g_* \in Z_q[X]/(X^N-1)$
- L_f, L_g : 잘려진 다항식 환 R의 부분집합
- g_* : $*$의 트랩도어 함수
- x_* : 임의로 생성되는 서명정보
- v : 임의로 생성되는 초기 값 또는 서명 검증 정보
- v_* : 결합함수에 의해 생성된 임의의 비트 스트링
- H : 해시 함수

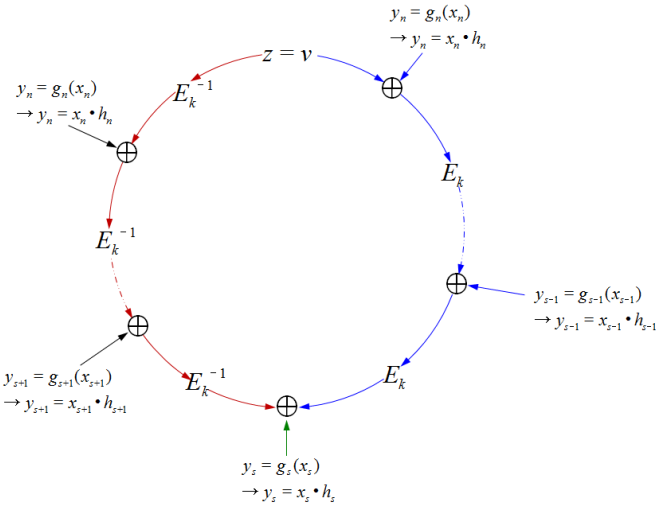
4.2 Polynomial Ring Trapdoor permutation

NTRU 기반 환 서명에서는 기존에 제안된 RSA Trapdoor permutation방식을 사용하지 않고 본 논문에서 제안하는 Polynomial Ring Trapdoor permutation 방식을 사용한다. 오직 사용자만이 효율적인 trapdoor정보를 통해 inverse permutation을 계산할 수 있는 아래 trapdoor permution을 사용하여 y, x′값을 생성한다. 공유된 polynomial ring trapdoor permutation과 사용자의 polynomial ring trapdoor permutation의 수식은 아래와 같다. 제안 방식은 서명 생성단계와 검증단계로 구성된다.

$$y_n = g_n(x_n) \rightarrow y_n = x_n \bullet h_n$$
$$h = pf_q \bullet g \bmod q$$
$$x_S = g_S(y_S) \rightarrow x = y \bullet e_{S=1} \bmod q$$
$$\rightarrow x = y \bullet g^{-1} \bullet f^{-1} \bmod q$$
$$\rightarrow x = x \bullet h \bullet g^{-1} \bullet f^{-1} \bmod q$$
$$\rightarrow x = x \bullet pf_q \bullet g \bullet g^{-1} \bullet f^{-1} \bmod q$$

4.3 서명생성 단계

Step 1 : 서명자는 임의로 생성된 그룹의 공개키와 전달할 메시지를 사용하여 메시지 서명에 사용될 대칭키를 생성한다.



(그림 2) Ring Signature based on Polynomial Ring

$$k = h(m, P_1, P_2, ..., P_r)$$

Step 2 : 서명자는 검증 데이터 생성 및 검증단계에서 사용자 인증 사용 될 임의 값 $v \in (0,1)^b$를 생성한다.

Step 3 : 서명자는 결합 함수에 사용될 각 링 멤버들의 임의 정보 x_i 집합을 생성하고 각 멤버들이 가진 Polynomial Ring Trapdoor permutation을 이용하여 y_i 집합을 생성한다.

$$y_i = g_i(x_i)$$

Step 4 : 서명자는 방정식의 좌변의 결과와 XOR 연산을 수행하고 K로 암호화 했을 때, 그 결과가 결합함수 $C_{k,v}(y_1, y_2, ..., y_r) = v$를 만족하는 y_s 값 계산한다.

$$\begin{aligned}
&C_{k,v}(y_1, y_2, ..., y_r) = v \\
&z = C_{k,v}(g_1(x_1), ..., g_n(x_n)) \\
&\quad = E_k(g_n(x_n) \oplus E_k(... \oplus E_1(g_1(x_1) \oplus v)) \\
&y_S = E_k(g_{S-1}(x_{S-1}) \oplus ... \oplus E_k(g_1(x_1) \oplus v)) \\
&\oplus E_k^{-1}(g_{S+1}(x_{S+1}) \oplus ... \oplus E_k^{-1}(g_n(x_n) \oplus E_k^{-1}(z))) \\
&y_S = E_k(y_{S-1} \oplus ... \oplus E_k(y_1 \oplus v)) \\
&\oplus E_k^{-1}(y_{S+1} \oplus ... \oplus E_k^{-1}(y_n \oplus E_k^{-1}(v)))
\end{aligned}$$

Step 5 : 사용자는 자신만이 알고 있는 Polynomial Ring Trapdoor permutation을 이용하여 서명자만이 생성할 수 있는 비밀값 x_s'를 생성한다.

$$x_S' = g_S^{-1}(y_S)$$

Step 6 : 사용자는 기존의 x_s를 제외한 링 멤버들의 임의 정보 x_i 집합와 변경된 $x_{s'}$값,서명 검증정보 v, 사용자 집합 정보 $(P_1, P_2, ..., P_r)$를 연접하여 검증자에게 전송한다.

$$(P_1, P_2, ..., P_r; v; x_1, x_2, ..., x_r)$$

4.4 검증단계

Step 1 : 검증자는 사용자가 전송한 서명 내에 x_i 값을 이용하여 y_i 값 계산한다.

Step 2 : 검증자는 사용자와 같은 방법으로 서명 검증에 필요한 대칭키 k를 계산한다.

$$k = h(m, P_1, P_2, ..., P_r)$$

Step 3 : 검증자는 결합 함수를 계산하고 검증 값과 서명 값 σ 가 만족하는지 검증 후 사용자의 서명으로 인증한다.

$$C_{k,v}(y_1, y_2, ..., y_r) = v$$

5. 제안방식분석

본 제안방식은 3장에서 도출된 보안요구사항을 다음과 같이 만족한다.

· 위조 불가성 : 링 멤버가 아닌 비인가자가 inverse permutation 없이 서명정보를 생성하는 것은 불가능하다.

· 서명자 모호성 : 검증자가 서명자를 찾기 위해서는 각각 링멤버의 서명에 대한 inverse permutation을 계산해야 한다. 본 제안은 트랩도어 함수의 안전성에 기반하여 서명자를 찾아낼 확률이 1/n 이므로 서명자의 모호성을 제공한다.

· 안전성 : 제안 방식에서 공격자가 사용자의 비밀정보를 찾는 것은 큰 크기의 레티스(Lattice)에서 작은 벡터를 찾는 수학 문제와 등가이므로 계산 상 불가능하다. 그러므로 본 제안방식은 트랩도어 함수의 특성을 만족한다. 또한 g가 공개된다 하여도 비멤버가 서명을 생성하기 위해선 g 중의 하나의 역함수를 알아야 하며 결합함수를 통해 y값을 알아냈다 해도 그 값의 역을 알아야 서명 생성이 가능하기 때문에 안전하다.

· 효율성 : 기존방식에서 사용되는 이산 대수 문제에 근간을 둔 지수승 연산과는 달리 다항식 섞음 시스템의 풀기 어려움에 안전성을 둔 단순 덧셈, 곱셈, 쉬프트 연산만을 수행하므로 연산량 측면에서 매우 효율적이다.

6. 결론

본 논문에서는 개인의 금융결제정보를 활용한 NFC 결제 환경에서 결제정보가 중간에 노출되는 문제점을 해결하기 위해 결제에 활용되는 개인의 금융정보를 결제에 직접적으로 사용하지 않고 임의의 정보를 이용해 결제자 자신을 증명할 수 있는 NTRU기반 환 서명 기법을 제안하였다. 본 방식은 기존방식과 동일한 통신횟수를 가지지만 기존방식에서 사용되는 지수승 연산과는 달리 다항식 섞음 시스템의 풀기 어려움에 안전성을 둔 단순 덧셈, 곱셈, 쉬프트 연산만을 수행하므로 연산량 측면에서 매우 효율적이며 기존 방식과 동일한 안전성을 제공한다. 향후 연구로는 사용자 추적이 가능한 NTRU기반 환 서명 연구가 필요할 것으로 사료된다.

참고문헌

[1] 기주희, "개인정보보호를 위한 익명 인증 기법 도입 방안 연구", 정보보호학회논문지 제20권 제6호, 2010. 12

[2] "Google Wallet: Security", Google, 2012

[3] J. Hoffstein, J. Pipher, J. H. Silverman, "NTRU : A Ring Based Public Key Cryptosystem", in Algorithmic Number Theory(ANTS III), 1998.

[4] R.L. Rivest, A. Shamir, Y. Tauman, "How to Leak a Secret", Advances in Cryptology-ASIACRYPT, 2001.

[5] "MasterCard PayPass", MasterCard, 2011

[6] 이윤경, "익명 인증 기술과 동향", 전자통신동향분석 제 23권 제 4호, 2008. 8