# 새로운 신원 위탁 방식에 관한 연구

황보성°, 이임영
순천향대학교 공과대학 정보기술공학부

# A Study On New Identity Escrow Scheme

Bo-Sung Hwang°, Im-Yeong Lee
Division of Information Technology Eng. Soonchunhyang Univ.

요약

인터넷 환경의 발달에 의해 사용자의 편리성은 증가하였지만, 사용자의 프라이버시가 노출되는 위험 또한 증가하였다. 그 중에서 사용자의 신원이 인증 절차에서 서비스 제공자에게 노출됨으로 사용자 프라이버시 침해 문제가 발생할 수 있는데, 본 논문에서는 인증 절차시 사용자의 익명성을 제공하고 유사시 제 3의 기관과 협력해서 익명성을 제거할 수 있는 새로운 신원 위탁 방식을 제안한다.

## 1. 서론

사용자들은 서비스 제공자에 접근하기 위해 자신을 증명해야 하는데 이 같은 개인식별은 사용자의 프라이버시 침해를 가져올 수 있기 때문에 사용자는 자신을 드러내지 않고 서비스를 받기를 원할 것이고 서비스 제공자는 사용자의 신원을 확인한 후 서비스를 제공하기를 원할 것이다. 두 가지 조건을 충족시켜줄 수 있는 것이 신원 위탁 방식(Identity escrow scheme)[1]이다. 사용자의 익명성을 제공하기 위해 사용자는 서비스 제공자에게 자신의 신원을 제공하지 않고, 제 3자에 의해서 발급된 사용자의 신원을 증명할 수 있는 정보를 줌으로써 사용자의 익명성을 제공할 수 있고 서비스 제공자의 신원 확인 요구를 만족시킬 수 있다.

하지만, 모든 사용자에게 완전한 익명성을 제공해야 하는 것인지는 생각해 보아야할 문제이다. 어떤 사용자가 서비스 제공자에게 접근해서 불법적인 행위를 하였을 경우에는 서비스 제공자 사용자의 익명성을 제거하고 정확한 신원을 알아야 할 것이다. 서비스 제공자는 사용자의 익명성을 제거하기 위해 사용자로부터 제공받은 정보를 Escrow agent에게 제공함으로써 그 정보에 대응되는 사용자의 정확한 신원을 확보할 수 있다.

* 본 연구는 1999학년도 한국무선국관리사업단 연구과제에 의해 수행되었음

신원 위탁 방식은 다음과 같은 잇점이 있다.

- 사용자가 서비스 제공자에게 접근할 때 익명성을 제공한다.
- 서비스 제공자들은 유사시 사용자의 정확한 신원을 알 수 있다.

### 1.1 신원 위탁 방식 구성요소

신원 위탁 방식은 Identifier, Issuer, Verifier, Escrow agent로 구성되며 기능은 다음과 같다.

- Identifier
  - Issuer에게 자신의 정확한 신원을 제공하고, 서비스 제공자에게 익명으로 자신을 증명할 수 있는 인증서를 가진다.
- Issuer
  - Identifier의 정확한 신원을 가지고, 유사시 Escrow agent와 협력해서 Identifier의 정확한 신원을 드러낸다.
- Verifier
  - 서비스 제공자로 Identifier의 인증서를 확인하고 유사시 Escrow agent에게 Identifier의 정확한 신원을 요구한다.
- Escrow agent
  - Verifier의 요구를 받아 Issuer와 협력하여 Identifier의 정확한 신원를 드러낸다.

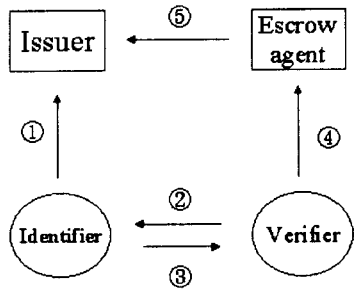
### 1.2 신원 위탁 방식의 기본단계

- 시스템 초기화 단계
  - 각 참여 개체는 시스템을 초기화하기 위해 자신의 파라메터를 공표한다.
- 신원 등록 단계
  - Identifier는 자신의 정확한 신원을 Issuer에게 전달한다.
- 인증 단계
  - Identifier는 서비스를 제공받기 위해 Verifier에게 자신의 인증서를 제공한다.
- 익명성 제거 단계
  - Verifier는 Escrow agent에게 인증서를 제공함으로써 사용자의 익명성을 제거한다.

## 2. 기존 신원 위탁 방식의 고찰

본 장에서는 Joe Kilian, Erez Petrank가 제안한 Group signature를 이용한 방식과 ZKIP을 이용한 방식[1]을 소개한다.

### 2.1 Group signature를 이용한 신원 위탁 방식

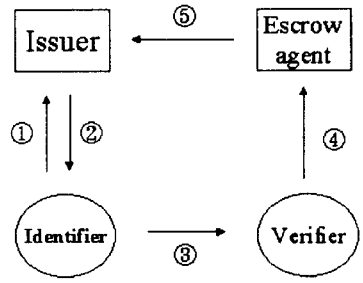



[그림 1] Group signature를 이용하는 방식

- 시스템 초기화 단계
  - Issuer
    n=pq, $e_1$, $e_2$를 생성하고 p, q는 비밀로 하고 n, $e_1$, $e_2$는 공개한다.
  - Escrow agent
    Group G=<g>와 element h∈G의 그룹에 관한 정보를 생성하고 ElGamal(P, $Y_R=h^P$)를 만들어 P는 비밀로 하고 G, g, h, $Y_R$는 공개한다.
  - Identifier
    $Y=x^{e1}$, $Z=g^Y$로 하고 x를 비밀로 한다.
- 신원 등록 단계
  ① Identifier는 Issuer에게 Z와 정확한 신원을 제출한다. ($h^r$, $Y_R{}^r g^Y$)
- 인증 단계
  ② Identifier는 Verifier에게 서비스를 요구하면 Verifier는 Identifier에게 Time과 Name을 포함하는 랜덤 메시지를 준다.
  ③ Identifier는 랜덤 메시지에 서명한다.
- 익명성 제거 단계
  ④ 유사시 Verifier는 Escrow agent에게 Identifier가 서명한 랜덤메시지를 준다.
  ⑤ Issuer와 Escrow agent의 협력에 의해 누가 서명했는지 드러낸다.

Group signature 방식은 Group을 감독하는 Group manager에 의해 통제되는데 Issuer와 Escrow agent로 구성된다. Group manager는 Group에 새로운 Identifier의 참여를 허락할 수 있고, Group에서 서명된 메시지를 증명할 수 있다. 이 방식은 Identifier가 Verifier에게 Group signature를 사용함[2]으로써 자신의 익명성을 유지시키고, 유사시 Group manager가 서명자를 확인함으로써 익명성을 제거할 수 있다. 시스템 파라메터 Z가 사용자의 정확한 신원과 연결되는 값으로 Issuer가 보관한다. 그리고 Issuer와 Escrow agent가 사용자의 비밀정보 x, Y를 알지 못함으로 메시지를 위조할 수 없다. 하지만, 이 방식의 큰 약점은 Group signature에 기반을 두고 있기 때문에 Issuer와 Escrow agent의 비독립성에 있다. Escrow agent는 보안상 익명성 제거시에만 호출되어야 하는데 이 방식에서는 시스템 초기화 때도 호출된다.

### 2.2 ZKIP을 이용한 신원 위탁 방식




[그림 2] ZKIP을 이용하는 방식

Joe Kilian, Erez Petrank은 Group Signature 방식이 Escrow agent와 잦은 접촉 때문에 보안상의 위험이 있다고 주장했다. 이를 해결하기 위해서는 Escrow agent가 다른 단계에서는 관여하지 않고 단지 익명성 제거시에만 접속하는 것이 바람직하다고 주장하고 이것을 ZKIP을 이용한 방법으로 설명하였다.

- 시스템 초기화 단계

- Issuer
  n=p*q, e, d와 랜덤수 δ을 생성하고 p, q, δ는 비밀로 하고 n, e, d는 공개한다.

· 신원 등록 단계
① Identifier는 자신의 정확한 신원을 보낸다.
② Issuer는 low bit에 해당하는 $a^e$을 생성하고 $a^e-b^e=\delta$를 만족하는 b를 계산해 인증서 (a,b)를 Identifier에게 제공한다.

· 인증 단계
③ Identifier는 Verifier에게 $a=a_1*a_2$ $b=b_1*b_2$, x, y를 선택하고 $a_1$, $a_2$, $b_1$, $b_2$, $(a_1)^e$, $(a_2)^e$, $(b_1)^e$, $(b_2)^e$, x, $x(a_1)^e$, $x(b_1)^e$, $x(a_1a_2)^e+y$, $x(b_1b_2)^e+y$를 제출한다. Escrow agents의 공개키를 이용해 $(a_1)^e$와 $(a_2)^e$는 제출하고 zero-knowledge를 이용해 다른 값들을 제출한다. Verifier는 $a = a_1*a_2$와 $b = b_1*b_2$가 $a^e - b^e = \delta$ 를 만족하는지 확인한다.

· 익명성 제거 단계
④ 유사시 Verifier는 escrow agent에게 $(a_1)^e$와 $(a_2)^e$는 제출한다.
⑤ Issuer와 Escrow agent의 협력에 의해 a에 해당하는 신원을 드러낸다.

위와 같이 이 프로토콜은 시스템 초기화시 Escrow agent의 접촉을 제거할 수 있다. 하지만 이 프로토콜은 가장 큰 약점은 Issuer가 사용자의 모든 정보를 가지고 있기 때문에 Identifier를 사칭할 수 있는 것이다.

# 3. 제안 방식

## 3.1 기반 기술

본 논문에서는 신원 위탁 방식의 일반적 요구사항을 만족시키기 위해 양자간의 사용자 비밀키 생성과 Blind 서명을 이용한다.

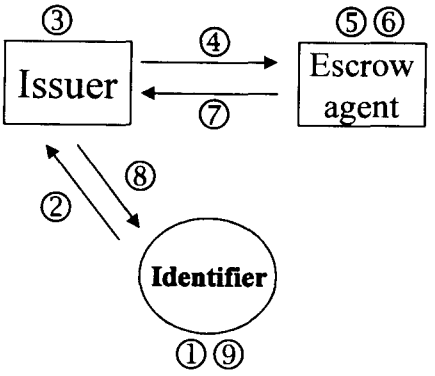
· 양자간의 사용자 비밀키 생성

사용자가 자신의 비밀키를 생성할 때, 사용자가 자신의 키의 반을 생성하고 나머지 반의 키는 제 3자(Issuer)가 생성함으로 해서 사용자는 제 3자가 자신의 비밀키 오용을 저지할 수 있다.

· Blind 서명

1982년 D.Chaum에 의하여 최초로 제안된 서명 방법으로 전자 현금의 익명성을 제공하기 위해 이용된다. 서명요구자가 서명자에게 서명되는 메시지의 내용을 숨기고자 할 때 이 프로토콜이 이용된다. 본 제안 방식에서는 Issuer가 서명하는 메시지에 대한 내용을 추적하지 못하게 하기 위해 Blind 서명을 이용한다.

## 3.2 제안 프로토콜




[그림 3] 신원등록단계

· 시스템 초기화 단계
- Issuer, Escrow agent
  자신의 공개키와 공개키에 해당하는 파라메터를 공개한다.

· 신원 등록 단계
① Identifier는 자신의 비밀키 중 일부가 되는 키쌍을 다음과 같이 생성한다.

$$P_{A_a} = g^{A_a} \bmod p$$

($P_{A_a}$ : 공개키, $A_a$ : 비밀키)

② Identifier은 자신의 공개키를 Escrow agent의 공개키로 암호화한 값($E_{KU_{esc}}[P_{A_a}]$)과 자신의 신원정보를 Issuer에게 제공한다.

③ Issuer는 Identifier의 신원을 저장하고 각 Identifier에 대한 식별자($ID_I$)와 나머지 키쌍 ($P_{A_b} = g^{A_b} \bmod p$)을 생성한다.

④ Issuer는 Escrow agent에게 다음을 제공한다.

$$E_{KU_{esc}} = [\ ID_I \ ||\ A_b \ ||\ E_{KU_{esc}}[P_{A_a}]]$$

⑤ Escrow agent는 $E_{KU_{esc}}[P_{A_a}]$를 복호하고 사용자의 최종공개키와 식별자($ID_E$)를 생성하고 $ID_E$와 $ID_I$를 저장한다.

$$P_A = (P_{A_a})^{A_b} \bmod p = g^{A_a * A_b} \bmod p$$

($P_A$ : Identifier의 최종공개키,
$A_a*A_b$ : Identifier의 최종비밀키)

⑥ Escrow agent는 $ID_E$를 자신의 공개키로 암호화한다.($E_{KU_{esc}}[ID_E]$) 이 값은 Identifier의 익명성을 제거할 때 이용된다. 그리고 다음과 같이 메시지를 생성한다.

$$M = P_A || E_{KU_{esc}}[ID_E] || H(P_A || E_{KU_{esc}}[ID_E])$$

⑦ Issuer에게 이 값을 블라인드 서명을 받기 위해 다음과 같은 메시지를 구성해 Issuer에게 전달한다.

$$M * P_{A_a}^{e} \quad (e : \text{Issuer의 공개키})$$

⑧ Issuer는 $A_b$와 다음과 같이 블라인드 서명한 값을 인증서로 Identifier에게 제공한다. Issuer는 $P_{A_a}$의 값을 알지 못하기 때문에 M의 내용을 알지 못하고 서명과정을 수행한다.

$$(M * P_{A_a}^{e})^{d} = M^{d} * P_{A_a} \quad (d : \text{Issuer의 비밀키})$$

⑨ Identifier는 $M^{d} * P_{A_a} / P_{A_a}$ 수행에 의해 서명된 인증서($M^{d}$)를 얻고 비밀키($A_a * A_b$)를 생성하고 공개키($P_A$)가 제대로 생성되었는지 검사한다. Issuer에 의해 서명된 인증서는 다음과 같다.

$$Sign_{Issuer}[P_A || E_{KU_{ac}}[ID_E]] || H(P_A || E_{KU_{ac}}[ID_E])]$$

- 인증 단계

① Identifier는 Verifier에게 접속시 인증서를 제공한다.

② Verifier는 정당한 사용자인지 알기 위해 시간과 Verifier 식별자를 포함하는 랜덤값([time||$ID_V$])을 생성해 Identifier에게 제공한다.

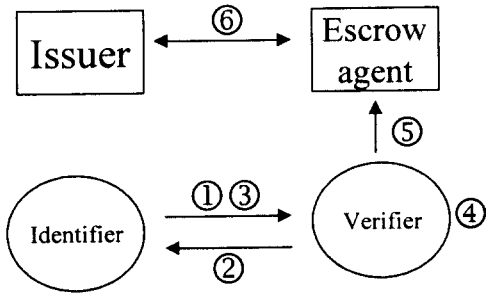
③ Identifier는 랜덤값에 자신의 비밀키로 서명해 Verifier에게 제공한다.

④ 인증서의 Issuer의 서명을 확인하고 Identifier의 공개키($P_A$)를 얻어 랜덤값의 서명을 확인한다.

- 익명성 제거 단계

⑤ Verifier는 부당한 Identifier의 익명성을 제거하기 위해 Escrow agent에게 $E_{KU_{ac}}[ID_E]$를 제공한다.

⑥ Escrow agent는 $ID_E$를 추출하고 그에 알맞은 $ID_I$을 Issuer에게 제공함으로써 Identifier의 신원을 드러낼 수 있다..




[그림 4] 인증 및 익명성 제거 단계

### 3.3 제안 방식 분석

제안 방식은 양자간의 사용자 비밀키 생성과 Blind 서명 기술을 이용해 신원 위탁 방식의 요구사항을 만족시키고자 하였다. 양자간의 사용자 비밀키 생성을 통해 사용자와 제 3자는 키쌍이 랜덤하다고 보장할 수 있고 사용자의 비밀키는 오직 사용자만이 알기 때문에 다른 개체가 사용자임을 위장할 수 없다. 또한 블라인드 서명은 Issuer가 서명하는 메시지에 대한 내용을 알지 못하게 하기 위해 이용된다. 이 방법을 이용해 Issuer와 Verifier와의 공모에 의한 Issuer의 신원이 드러나는 것을 방지할 수 있다. 또한, 인증서에는 Issuer가 사용자를 추적할 수 있는 정보가 없기 때문에 Issuer와 Escrow agent가 협동해야만 사용자의 신원을 알아 낼 수 있고 혼자서는 사용자의 신원을 알 수 없다. 기존의 방식과 새로운 제안 방식을 비교하면 다음과 같다.

[표 1] 기존 방식과 제안 방식의 비교

| | Escrow agent의 독립성 | 사용자 추적방지 | 신원 사칭 방지 | 시스템 효율성 |
|---|---|---|---|---|
| Group 서명을 이용하는 방식 | X | X | ○ | ○ |
| ZKIP을 이용한 방식 | O | X | X | X |
| 제안 방식 | X | O | ○ | △ |

## 4. 결론

지금까지 기존의 신원 위탁 방식을 소개하고 Blind 서명과 양자간의 사용자 키생성을 이용한 새로운 신원 위탁 방식을 살펴보았다.

사용자가 네트워크 상에서 서비스 제공자에게 접속할 때 자신의 신원을 정확히 알려주느냐, 그렇지 않느냐의 문제는 사용자와 서비스 제공자에게는 아주 민감한 문제이다. 사용자는 자신의 신원을 숨긴 채 서비스 받기를 원할 것이고, 서비스 제공자는 사용자의 정확한 신원확인 후 서비스를 제공하기를 원할 것이다. 이렇게 상충되는 의견은 신원 위탁 방식을 사용함으로써 사용자와 서비스 제공자 모두에게 소기의 목적을 달성할 수 있을 것이다.

## 참고문헌

[1] Joe Kilian and Erez Petrank, "Identity Escrow", Advances in Crytology-CRYPTO '98, pp.169-184, 1998

[2] Camenisch, "Efficient and generalized group signatures", Advances in Cryptology-EUTOCRYPT '97, pp.465-479, 1997