자체인증 개인식별정보를 이용한 실용적인 일회용 패스워드 시스템

Practical one-time password system using the self-certified identity information

박 성 준 , 성 맹 희
한국정보보호센터
서울시 종로구 내수동 167 세종로 대우빌딩 복합동 2층
tel) 02) 398-0356, fax) 02) 398-0222

Sungjun Park , MeangHee Sung
KISA (Korea Information Security Agency)
2nd Fl., Sejongro Daewoo Bldg., 167 Naesu-Dong, Jongro-Gu, Seoul, Korea.
tel) 02) 398-0356, fax) 02) 398-0222

요 약

본 논문에서는 기존의 일회용 비밀번호 시스템의 사용 횟수의 제한 문제를 "서명 고리" 개념을 사용하여 해결한 횟수 제한이 없는 일회용 비밀번호 시스템을 제안한다.

제안하는 방식은, 역설적인 개인식별정보 방식인, 자체인증 개인식별정보에 기반을 둔 서명방식을 사용하였으며, 홈 뱅킹에 적합한 실용적인 일회용 비밀번호 시스템이다.

1. 개 요

현재 홈뱅킹 시스템은 사용자의 식별을 위해서 일반적으로 비밀번호 시스템을 사용하고 있으며, 공중통신망에 사용자의 비밀번호가 노출되는 위험한 특성을 가지고 있다. 이러한 위험 요소로 인해 이미 우리나라에서도 홈뱅킹의 안전성 문제가 발생하였으며, 이러한 안전성 문제가 해결되지 않고서는 건전한 전자상거래 달성에 많은 지장을 초래하게 된다.

일반적으로 공중통신망상에서의 사용자의 비밀번호의 노출위험성은 일회용 비밀번호 시스템을 이용하여 해결하고 있다. 그러나 기존의 일방향 함수를 사용한 일회용 비밀번호 시스템은 사용 횟수의 제한이라는 문제점을 가지고 있다.

본 논문에서는, 새로운 개념인, 서명 고리 (signature chain) 개념을 제안하고, 이 개념을 사용하여 사용 횟수에 제한이 없고, 매우 효율적인 일회용 비밀번호 시스템을 제안한다.

제안하는 일회용 비밀번호 시스템은 일방향 함수를 사용하는 기존의 시스템과는 달리 자체인증 특성을 지닌 개인식별정보에 기반을 둔 서명방식을 사용함으로써, 은행에서도 사용자의 비밀번호를 알 수 없는 특성으로 인하여 제안한 방식을 필요시 일반적인 전자서명 방식으로도 활용 가능하도록 하였다. 또한, 제안한 일회용 비밀번호 시스템은 홈뱅킹에 사용가능한 실용적인 일회용 비밀번호 시스템이다.

2. 기존의 일회용 비밀번호 시스템

기 제안된 일회용 비밀번호 시스템은 일방향 함수(one-way function)를 근간으로 구성된다.[1,3,4] 일방향 함수란 함수 값을 계산하기는 쉽지만, 역함수 값을 계산하는 문제는 매우 어려운 함수를 의미한다. 즉, 일방향 함수 f란 x값이 주어지면 f(x)를 계산 가능하나, 역으로 f(x)에서 $x = f^{-1}(x)$를 계산하는 것은 불가능한 함수이다.

먼저 일방향 함수를 사용하여 일회용 비밀번호 시스템을 구성하는 일반적인 방식을 알아본다.

사용자는 먼저 초기값 x_0, 일방향 함수 f, 그리고 사용 횟수 m을 결정한다. $f(x_0)$, $f^2(x_0)$, $f^3(x_0)$, $f^4(x_0)$, ..., $f^{m-1}(x_0)$, $f^m(x_0)$를 계산한 후 $f^m(x_0)$를 공개한다. 이 경우 첫 번째 비밀번호는 $f^{m-1}(x_0)$이 되고 공개 정보는 $f^m(x_0)$에서 $f^{m-1}(x_0)$로 변경된다. 첨자의 역순으로 비밀번호는 갱신된다. 그러므로 마지막 m번째 비밀번호는 x_0, 공개정보는 $f(x_0)$가 된다.

<표 2.1>에서 일회용 비밀번호 시스템의 비밀정보, 비밀번호의 변경 정보를 표현하였다.

<표 2.1> 공개정보와 비밀번호의 변경

사용 순서	전송 정보	비밀번호
1_{st}	$f^m(x_0)$	$f^{m-1}(x_0)$
2_{nd}	$f^{m-1}(x_0)$	$f^{m-2}(x_0)$
3_{rd}	$f^{m-2}(x_0)$	$f^{m-3}(x_0)$
⋮	⋮	⋮
$(m-1)_{th}$	$f^2(x_0)$	$f(x_0)$
m_{th}	$f(x_0)$	x_0

3. 자체인증 개인식별정보

Girault는 인증자에 기반을 둔 방식과 개인식별정보에 기반을 둔 방식의 중간 개념인 자체인증 공개키 (self-certified public key) 방식을 제안하였다. 자체인증 공개키 방식은 별도의 인증자를 요구하지 않고 공개키 자체가 인증자 역할을 하는 방식이다.[2]

공개키 암호시스템에서는 각 사용자는 자신의 개인식별정보 I, 비밀키 s와 공개키 PK로 구성된 (I, s, PK)를 갖게 된다. 비밀키 s는 사용자만이 알고 있으며 공개키 PK는 모든 사용자에게 알려주게 된다. 특히 모든 사용자가 알 수 있게 하기 위한 방법으로 모든 사용자가 액세스할 수 있는 공개키 디렉토리를 사용한다. 그러나 모든 사용자가 액세스할 수 있는 공개키 디렉토리에 의해 사용자 A의 공개키 PK_A에 대한 제3자의 능동적 공격(active attack)이 가능하게 되어 공개키 디렉토리내의 공개키에 대한 인증 문제가 야기된다. 즉, 능동적인 공격자는 공개키 디렉토리에 있는 사용자 A의 공개키 PK_A를 자신이 만든 거짓의 공개키 PK'_A로 대치함으로써 사용자 A를 흉내낼 수 가 있게 된다. 그러므로 공개키 PK_A가 사용자 A의 공개키 인지를 확인하는 절차를 요구하게 된다.

바로 이 공개키에 대한 인증 문제의 해결책으로 인증자에 기반을 둔 방식이 제안되고 있다. 인증자에 기반을 둔 방식에서는 각 사용자는 자신의 개인식별정보 I, 비밀키 s와 공개키 PK로 구성된 (I, s, PK)에 더불어 자신의 공개키 PK에 대한 인증을 위한 (I, PK)를 센터의 비밀키로 서명한 인증자 C를 필요로하게 된다. 즉, 모든 사용자는 (I, s, PK, C)를 갖게 된다.

반면에 개인식별정보에 기반을 둔 방식에서는 각 사용자의 공개키 PK가 바로 자신의 개인식별정보 I가 됨으로서 특별한 인증자 C를 요구하지 않는다. 즉, 이 방식에서의 인증자는 바로 자신만이 소유한 비밀키 s가 된다.

또한 자체인증 공개키 방식에서는 공개키 PK 자체가 인증자 C의 역할을 하게 된다. 즉, C = P

가 된다. 바로 이 이유에서 자체인증 공개키라고 명명된다.

박성준외 2인은 자체인증 공개키 방식을 개인식별정보에 기반을 둔 방식에 적용하여 만든 새로운 개념인 자체인증 개인식별정보 (self-certified identity information) 방식을 제안하였다.[6,7] 자체인증 개인식별정보 방식은 자체인증 공개키 방식에서 인증자의 역할을 하는 공개키가 바로 개인식별 정보인 경우이다. 즉, C = PK = I가 된다.

4. "서명 고리"를 이용한 일회용 패스워드 시스템

4.1 서명 고리 (signature chain)

본 절에서는 "서명 고리 (signature chain)" 개념을 사용하여 사용 횟수에 제한이 없고, 매우 효율적인 일회용 비밀번호 시스템 모델을 제안한다. 제안하는 모델은 다음과 같다.

<사용자 i의 등록과정>

① 사용자 i는 은행에 등록을 신청한다.
② 은행은 사용자 i의 (공개키 P_i, 비밀키 s_i) 쌍을 계산하고, 랜덤 초기값 $seed_i$를 생성한다.
③ 은행은 사용자 i에 대응되는 (ID_i, P_i, $seed_i$)를 저장한 후, 사용자 i에게 (s_i, $seed_i$)를 분배한다.

<일회용 비밀번호 시스템>

(1) 첫번째

① 사용자 i는 $seed_i$에 대한 서명값, $sign(seed_i)$를 계산한 후, 서명값을 은행에 전송한다.
② 사용자 i는 자신의 비밀정보 (s_i, $seed_i$)를 (s_i, $sign(seed_i)$)로 변경한다.
③ 은행은 사용자 i의 공개키를 사용하여, 서명값을 확인한다.
④ 은행은 (ID_i, P_i, $seed_i$)를 (ID_i, P_i, $sign(seed_i)$)로 변경한다.

(2) 2번째

① 사용자 i는 $sign(seed_i)$에 대한 서명값, $sign(sign(seed_i))$를 계산한 후, 서명값을 은행에 전송한다.
② 사용자 i는 자신의 비밀정보 (s_i, $sign(seed_i)$)를 (s_i, $sign(sign(seed_i))$)로 변경한다.
③ 은행은 사용자 i의 공개키를 사용하여, 서명값을 확인한다.
④ 은행은 (ID_i, P_i, $sign(seed_i)$)를 (ID_i, P_i, $sign(sign(seed_i))$)로 변경한다.

(3) 3번째

⋮
⋮

(m) m번째

① 사용자 i는 $sign(seed_i)$에 대한 서명값, $sign^m(seed_i)$를 계산한 후, 서명값을 은행에 전송한다.
② 사용자 i는 자신의 비밀정보 (s_i, $sign^{m-1}(seed_i)$)를 (s_i, $sign^m(seed_i)$)로 변경한다.

③ 은행은 사용자 i의 공개키를 사용하여, 서명값을 확인한다.
④ 은행은 $(ID_i, P_i, sign^{m-1}(seed_i))$를 $(ID_i, P_i, sign^{m}(seed_i))$로 변경한다.

4.2 실용적인 일회용 패스워드 시스템

본 절에서는 역설적인 개인식별정보 방식인 자체인증 개인식별정보에 기반을 둔 서명방식을 사용한 홈뱅킹에 적용한 실용적인 일회용 비밀번호 시스템을 제안한다. 본 방식의 장점은 사용자 A에 대한 은행에서 저장하는 A의 정보 $(ID_A, C_A, seed_A)$에 대한 비밀성에 구애받지 않는다는 것이다. 즉, 본 시스템의 안전성은 합성수 n의 소인수 p, q의 비밀성과 사용자 A의 비밀정보 s_A에만 의존한다.

<은행의 시스템 구성[5]>

- n은 다음의 형태를 만족하는 2 소수 p, q의 곱이다. 즉, $n = p \cdot q$.
  [형태] $p = 2\gamma^d fp'+1$, $q = 2fq'+1$,
  여기서 f, p', q'는 서로 다른 소수이고, $\gcd(\gamma, q')=1$, $\gcd(\gamma, f)=1$.
- (n, γ, y) : acceptable triple
- 안전한 해쉬 함수 h

<사용자 A의 등록과정>

① 사용자 A는 비밀키 s_A를 랜덤하게 선택. 단, $0 < s_A < f$.
② A는 자신의 개인식별정보 ID_A와 $b^{s_A} \pmod n$를 은행에 등록 신청
③ 은행은 다음 수식을 만족하는 i_A, x_A를 계산하고, 랜덤 초기값 $seed_A$와 C_A를 생성한다.

$$ID_A = b^{-s_A} y^{-i_A} x_A^{-\gamma^d} \pmod n$$

$$C_A = ID_A y^{i_A} x_A^{\gamma^d} \pmod n$$

④ 은행은 사용자 A에 대응되는 $(ID_A, C_A, seed_A)$를 저장
⑤ 사용자 A에게 $seed_A$를 분배

<일회용 비밀번호 시스템>

(1) 첫번째

① 사용자 A는 [1, f-1]상의 랜덤수 r_1를 선택하고, v_1, e_1를 계산한다.

$$v_1 = b^{r_1} \pmod n,$$

$$e_1 = h(v_1, seed_A).$$

② A는 $z_1 = r_1 + s_A e_1 \pmod f$를 계산한 후, (z_1, e_1)를 은행에 전송한다.
③ 사용자 A는 자신의 비밀정보 $(s_A, seed_A)$를 (s_A, e_1)로 변경한다.
④ 은행은 다음의 수식을 확인한다.

$$C_A^{e_1} b^{z_1} = (ID_A y^{i_A} x_A^{\gamma^d})^{e_1} b^{z_1} = v_1 \pmod n,$$

$$e_1 = h(v_1, seed_A)$$

⑤ 은행은 $(ID_A, C_A, seed_A)$를 (ID_A, C_A, e_1)로 변경한다.

(2) 2번째

① 사용자 A는 [1, f-1]상의 랜덤수 r_2를 선택하고, v_2, e_2를 계산한다.

$$v_2 = b^{r_2} \pmod n,$$

$$e_2 = h(v_2, e_1).$$

② A는 $z_2 = r_2 + s_A e_2 \pmod f$를 계산한 후, (z_2, e_2)를 은행에 전송한다.

③ 사용자 A는 자신의 비밀정보 (s_A, e_1)를 (s_A, e_2)로 변경한다.

④ 은행은 다음의 수식을 확인한다.

$$C_A^{e_2} b^{z_2} = (ID_A y^{i_A} x_A^{j_A})^{e_2} b^{z_2} = v_2 \pmod n$$

$$e_2 = h(v_2, e_1)$$

⑤ 은행은 (ID_A, C_A, e_1)를 (ID_A, C_A, e_2)로 변경한다.

(3) 3번째

……

⋮

⋮

(m-1) m-1번째

……

(m) m번째

① 사용자 A는 [1, f-1]상의 랜덤수 r_m를 선택하고, v_m, e_m를 계산한다.

$$v_m = b^{r_m} \pmod n,$$

$$e_m = h(v_m, e_{m-1}).$$

② A는 $z_m = r_m + s_A e_m \pmod f$를 계산한 후, (z_m, e_m)를 은행에 전송한다.

③ 사용자 A는 자신의 비밀정보 (s_A, e_{m-1})를 (s_A, e_m)로 변경한다.

④ 은행은 다음의 수식을 확인한다.

$$C_A^{e_m} b^{z_m} = (ID_A y^{i_A} x_A^{j_A})^{e_m} b^{z_m} = v_m \pmod n$$

$$e_m = h(v_m, e_{m-1})$$

⑤ 은행은 (ID_A, C_A, e_{m-1})를 (ID_A, C_A, e_m)로 변경한다.

즉, 사용자 A의 m번째 일회용 패스워드는 (z_m, e_m)이 된다. 여기서,

$e_m = h(v_m, e_{m-1}) = h(v_m, h(v_m, e_{m-2})) = \cdots = h(v_m, h(v_m, \cdots, h(v_1, seed_A)) \cdots)$

<표 4.1>에서 이러한 과정을 나타낸다.

5. 결론

본 논문에서는 현재 안전성 문제가 야기된, 공중통신망에 사용자의 비밀번호가 노출되는 위험한 특성을 지닌 홈뱅킹 시스템에서 현재 사용중인 비밀번호 시스템의 문제점을 해결하는 일회용 비밀번호 시스템을 제안하였다.

제안한 일회용 비밀번호 시스템은 기존의 방식과는 달리 역설적인 개인식별정보 방식인 자체인증 개인식별정보에 기반을 둔 서명방식을 사용하여 사용 횟수의 제한이라는 문제를 해결하였다.

특허 본 방식은 자체인증 개인식별정보 방식의 특성으로 인하여 필요시 일반적인 전자서명 방식으로도 활용가능하다.

<표 4.1> 홈뱅킹에 적용한 일회용 비밀번호 시스템

사용 순번	은행	일회용 패스워드	사용자 A
초기화	$(ID_A, C_A, seed_A)$		$(s_A, seed_A)$
1_{st}	① $C_A^{e_1} b^{z_1} = v_1 \pmod n$ ② $e_1 = h(v_1, seed_A)$ ③ $(ID_A, C_A, seed_A) \rightarrow (ID_A, C_A, e_1)$	(z_1, e_1) <------	① 랜덤수 r_1 ② $v_1 = b^{r_1} \pmod n$ ③ $e_1 = h(v_1, seed_A)$ $z_1 = r_1 + s_A e_1 \pmod f$ ④ $(s_A, seed_A) \rightarrow (s_A, e_1)$
2_{nd}	① $C_A^{e_2} b^{z_2} = v_2 \pmod n$ ② $e_2 = h(v_2, e_1)$ ③ $(ID_A, C_A, e_1) \rightarrow (ID_A, C_A, e_2)$	(z_2, e_2) <------	① 랜덤수 r_2 ② $v_2 = b^{r_2} \pmod n$ ③ $e_2 = h(v_2, e_1)$ $z_2 = r_2 + s_A e_2 \pmod f$ ④ $(s_A, e_1) \rightarrow (s_A, e_2)$
⋮	⋮	⋮	⋮
m_{th}	① $C_A^{e_m} b^{z_m} = v_m \pmod n$ ② $e_m = h(v_m, e_{m-1})$ ③ $(ID_A, C_A, e_{m-1}) \rightarrow (ID_A, C_A, e_m)$	(z_m, e_m) <------	① 랜덤수 r_m ② $v_m = b^{r_m} \pmod n$ ③ $e_m = h(v_m, e_{m-1})$ $z_m = r_m + s_A e_m \pmod f$ ④ $(s_A, e_{m-1}) \rightarrow (s_A, e_m)$
⋮	⋮	⋮	⋮

- **Remark**

본 방식은 사용자의 일회용 패스워드 생성을 사전계산(precomputation)에 의해 구할 수 있는 특성을 가지고 있다. 즉, m번째 일회용 패스워드 (z_m, e_m)를 생성하는 다음의 모든 절차는 사전계산이 가능하다는 것이다.

① 랜덤수 r_m

② $v_m = b^{r_m} \pmod n$

③ $e_m = h(v_m, e_{m-1})$

$z_m = r_m + s_A e_m \pmod f$

이러한 특성은 사용자가 다수의 일회용 패스워드 (z_1, e_1), (z_2, e_2), …, (z_m, e_m)를 미리 계산하여 저장한 뒤 필요시 사용할 수 있는 특성을 의미한다.

[참고 문헌]

[1] D. C. Feldmeier and P. R. Karn, "UNIX Password Security-Ten Years Later", CRYPTO'89.

[2] M. Girault,"Self-Certified Public Key", EUROCRYPT'91, pp. 490-497, 1991.

[3] Leslie Lamport, "Password Authentication with Insecure Communication", Communications of the ACM 24.11, pp. 770-772.

[4] Neil M. Haller and Philip R. Karn, "Description of the S/KEY One-time Password System", From Internet(ftp://thumper.bellcore.com/pub/nmh/skey)

[5] S. J. Park and D. H. Won, "A Generalized Public Key Residue Cryptosystem and Its Applications", IEEE GLOBECOM'95, pp. 1179-1182, 1995.

[6] 박성준, 양형규, 원동호,"자체인증 개인식별정보", 한국통신정보보호학회 종합학술발표회 논문집, pp. 9-13, 1994.

[7] 박성준, 원동호,"고차잉여류 문제와 이산대수 문제에 기반을 둔 역설적인 id-based 암호시스템", 한국통신정보보호학회 논문지 제4권 제2호, pp. 113-118, 1994.