이동성 속성기반 스마트카드 인증스킴

서화정*, 김호원*
*부산대학교 컴퓨터공학과
e-mail:[hwajeong,howonkim]@pusan.ac.kr

An Attribute-Based Scheme for Mobile Authentication

Hwa-Joeng Seo*, Ho-Won Kim*
*Dept of Computer Engineering, Pusan University

요 약

급속한 기술의 발전과 더불어 스마트카드 하나만으로도 서비스 제공에 필요한 인증이 가능한 기술이 제시되고 있다. 우리는 편리함과 간편함을 제공받는 대신에 보안취약성이라는 위험에 노출되게 된다. 스마트카드는 사용자의 정보와 같은 민감한 정보를 포함하기 때문에 이를 보호하기 위한 다양한 연산(사칙연산, 지수연산, 해시연산 그리고 암호화연산)들을 통해 메시지 인증 및 암호화를 수행하게 된다. 현재 산학연을 중심으로 다양한 암호화 프로토콜들이 제시되고 있다. 그 중에서도 2008년 Rhee. et. al에 의해 제시된 속성기반 스마트카드 인증기법은 사용자의 특성에 따라 다양한 서비스 제공이 가능하다는 장점을 가진다. 하지만 이동성을 제공하지 못하는 단점이 있어 자신의 스마트카드를 통한 인증범위가 제한되는 단점을 가진다. 본 논문에서는 이동성을 보장하는 개선된 속성기반 암호화 기법을 제시한다.

1. 서론

사용자가 원하는 네트워크 서비스를 제공하는데 있어 사용자 인증은 중요한 요건이다. 모든 사용자는 서비스 요청 시 인증 서버와 상호인증이 일어난 후 서비스를 제공받게 된다. 하지만 인증과정에서 사용되는 개인정보는 공격자에게 노출 시 악의적인 목적으로 사용 및 오용되어 사용자에게 피해를 줄 수 있다. 따라서 사용자의 정보를 안전하게 보호하는 안전한 인증 및 키교환 기법의 연구가 시급하다.

최근 2008년에는 스마트카드에 속성기반암호화[3]를 적용한 인증 스킴인 Rhee et al.[4]에 의해 제안되었다. 해당 스킴은 인증이 필요한 서비스에 대한 권한을 속성에 따라 설정가능하다. 따라서 기존의 인증기법에 비해 보다 다양한 응용 및 적용이 가능하다는 장점을 가진다. 하지만 스마트카드는 사용자를 대표하는 하나의 인증문서로써 한정된 영역이 아닌 다양한 환경 및 서버에서의 인증 또한 보장이 되어야 한다.

따라서 본 논문에서는 스마트카드의 인증이 기존의 서버가 아닌 다른 서버에서도 키교환과 같은 부과적인 연산 없이 가능하도록 프로토콜을 제안하였다.

본 논문은 제 2장에서 논문과 관련된 연구인 속성기반 암호화와 이를 이용한 Rhee et al.에 대해 소개하고 제 3장에서는 이동성을 고려한 스마트카드 인증스킴에 대해 제안한다. 제 4장에서는 제안한 기법의 성능 및 안전성을 분석하며 마지막으로 제 5장에서는 제안된 스킴에 대한 결론을 내린다.

"이 논문 또는 저서는 2011년 교육과학기술부로부터 지원받아 수행된 연구임" (지역거점연구단육성사업/차세대물류IT기술연구사업단).

2. 관련연구

본 논문에서 사용되는 용어와 제안된 스킴과 관련된 연구결과에 대해 설명한다.

2.1. 용어

[표 1]는 Rhee et al.과 제안된 스킴에 사용된 용어에 대한 설명이다.

<표 1> Rhee et al.과 제안된 스킴에 사용되는 용어

기호	설명
ID	사용자의 아이디
pw	사용자의 패스워드
r, r_p, e	스마트카드와 서버에서 생성되는 랜덤 값

x_s	서버의 비밀 키
C	인증에 필요한 로그인 메시지
G	서버가 정의한 속성 값들의 집합
A, A^*	G의 부분 집합
U_i	사용자 i를 나타내는 기호
U_a	사용자 a를 나타내는 기호
S	인증 서버를 나타내는 기호
S_{new}	다른 인증 서버를 나타내는 기호
$a_{i,j}$	사용자 U_i에 해당하는 속성 값
G^*	위수가 소수 p인 군
$h()$	$\{0,1\}^* \rightarrow \{0,1\}^l$ 일방향 해쉬 함수
$T, \triangle T$	타임스탬프, 타임스탬프 허용제한시간
g	Diffie-Hellman 키교환 프로토콜의 파라미터
$\oplus$	bitwise exclusive-or 연산
=>	안전한 회선을 통한 전송
$\rightarrow$	일반 회선을 통한 전송

2.2. 선행연구

Rhee et al. scheme은 속성기반 암호화를 이용하여 새로운 인증 기법을 제안하였다. 해당 스킴은 속성에 따라 사용자에게 다양한 조건에 따른 인증이 가능하게 한다. 하지만 이동성이 제공되는 인증스킴은 제공하지 않는다. 따라서 본 논문에서는 이동성을 제공하는 속성 기반 인증 기법을 통해 다양한 환경에서도 효율적인 보안 통신이 가능한 인증 스킴을 제안한다.

2.2.1. 속성기반 암호화(ABE : Attribute-Based Encryption)

Sahai와 Waters에 의해 제안된 퍼지 아이디 기반 암호화(Fuzzy Identity-Based Encryption) 기법[3]은 신원(Identity)기반 암호화 기법[2]을 확장한 것으로 송신자가 선택한 속성 값을 암호 인자로 암호화하여 해당 속성값을 가지고 있는 수신자가 복호화 할 수 있도록 제안된 시스템이다. 해당 기법은 암호화 과정이 기존의 신원 기반 암호화 기법에서와 같이 1대1관계가 아닌 1대N관계이므로 분산 환경 시스템에서의 다양한 응용에 적용가능하다.

2.2.2. Rhee et al. scheme[4]

Rhee et al.의 scheme은 속성기반 암호화를 스마트카드 인증에 최초로 적용하였다. 속성값을 사용하여 상호간의 인증을 익명으로 수행이 가능하며 서버와 사용자간의 세션키분배도 Diffie-Hellman Protocol을 통해 안전하게 수행된다. 또한 Chien et al.[1]와 달리 대칭키 암호를 사용하지 않으므로 성능이 개선된다. 스킴은 [표 2]의 등록 단계, [표 3]의 로그인 단계, 그리고 [표 4]의 검증단계로 구성된다. 등록단계는 스마트카드를 부여받을 때 [Step 1]에서 자신의 ID와 password에 대한 정보를 서버쪽으로 전송한다. [Step 2~3]에서는 서버의 비밀키, 스마트카드의 비밀키 값 그리고 스마트카드의 아이디에 해쉬와 XOR연산을 수행하여 암호문을 생성한다. [Step 4]에서는 속성값과 서버의 비밀키값을 XOR연산을 한 후 해시를 하며 서버의 비밀키 값에 대한 해시값을 다시 XOR연산한다. [Step 5]에서는 생성된 메시지를 스마트카드로 안전한 회선을 통해 전송한다.

<표 2> 등록 단계

1 U_i => S : ID, pw를 전송한다.
2 S : $V = h(x_s) \oplus h(pw)$를 계산한다.
3 S : $I = h(x_s) \oplus h(ID \oplus pw)$를 계산한다.
4 S : $y_i = h(a_{i,j} \oplus x_s) \oplus h(x_s)$를 계산한다. 속성값은 $a_{i,j} \in A_i, 1 \leq j \leq n\,(A_i \subseteq G)$조건을 만족한다.
5 S=> U_i : $(A_i, h(), Y_i, V, I)$를 전송한다.

로그인과정은 스마트카드를 통한 인증이 필요한 시점에 수행되어 서버와의 인증에 필요한 메시지를 생성한다. [Step 1]에서는 스마트 카드의 아이디와 비밀번호 그리고 속성값을 입력한다. [Step 2~3]에서는 비밀키값을 통해 V, W 값을 생성한다. [Step 4]에서는 속성값과 W값을 XOR연산을 하며 [Step 5]에서는 타임 스탬프와 키교환에 사용되는 인자를 생성하여 해쉬 및 XOR연산을 수행한다. [Step 6]에서는 생성된 결과값을 인증 서버에게 보내게 된다.

<표 3> 로그인 단계

1 스마트카드를 terminal에 넣고 ID, pw 그리고 선택한 k개의 속성 값 $a_{i,j_t}, 1 \leq t \leq k$를 입력한다.
2 U_i : $V \oplus h(pw) \oplus h(ID \oplus pw)$를 계산하여 I값과 비교하여 동일한 경우 다음 단계를 진행한다.
3 U_i : $W = V \oplus h(pw)$를 계산한다.
4 U_i : $X_t = y_t \oplus W (1 \leq t \leq k), X = \prod_t^k X_t$를 계산한다.
5 U_i : $C_1 = (X \| T)^r, C_2 = W \oplus r, C_3 = g^{r'}$을 계산한다. $r, r' \in G^*$는 임의의 수, T는 타임스탬프 그리고 g는 G의 생성원이다.
6 $U_i \rightarrow S$: $C = [(a_{i,j_1}, a_{i,j_2}, \ldots, a_{i,j_k}), C_1, C_2, C_3, T]$을 전송한다.

검증과정의 [Step 1]에서는 스마트 카드로부터 받은 메시지의 타임스탬프를 확인하여 재전송 공격을 판단한다. [Step 2]에서는 전송된 속성값이 특정 속성을 만족하는지 확인한다. [Step 3~4]에서는 전달되어온 메시지로부터 인증에 필요한 값을 계산하기 위해 지수 인자와 밑수를 계

산한다. [Step 5]에서는 상대방과 키교환 및 상호인증에 필요한 메시지를 생성한다. [Step 6~7]에서는 스마트카드로 전송된 메시지를 확인하여 서버임을 확인하고 Diffie-Hellman기법을 사용하여 서버와 스마트카드간의 키교환을 하게 된다.

<표 4> 검증 단계

1 S : T를 계산하여 $\triangle T$를 만족하는지 확인한다.
2 S : 전송된 속성값 $a_{i,j_t} \in A, 1 \leq t \leq k$이 조건을 만족하는 경우 다음 단계를 진행한다.
3 S : $r'' = C_2 \oplus h(x_s)$와 $Z = \prod_{j_t}^{k} h(a_{i,j_t} \oplus x_s)$를 계산한다.
4 S : $(Z \| T)^{r''}$를 계산하여 C_1와 동일한 경우 다음 단계를 진행한다.
5 S : $C_4 = Z^{r''+1}$와 $C_5 = g^e$를 계산한다. 임의의 수 $e \in G^*$를 만족한다.
6 $S \rightarrow U_i$: $B = [C_4, C_5]$를 전송한다.
7 U_i : X^{r+1}를 계산하여 C_4와 동일한 경우 서버에 대한 인증이 되며 세션키 $C_5^r = g^{re}$를 공유하게 된다.

Ⅲ. 제안된 스킴

[Step1]에서 인증에 필요한 메시지를 새로운 서버에게 전송한다. 메시지에는 속성값과 자신이 이전에 속한 서버의 아이디 그리고 세션키 생성에 필요한 암호화키를 생성한 후 전송한다. [Step2]에서 새로운 서버는 기존서버에 해당 메시지를 전송한다. 이를 받은 서버에서는 메시지의 속성을 확인한 후 암호화값이 적합한 유저의 대칭키로 생성되었는지 확인한다. 확인이 되며 상호인증을 위한 메시지를 위해 키의 해시값을 메시지에 xor연산하여 새로운 암호화 값을 생성한다. 새롭게 생성된 암호화 값은 안전한 채널을 통해 다른 서버로 전송이 되며 이 값을 이용하여 새로운 세션키가 성립되게 된다. 세션키는 전송된 메시지에 서버에서 생성된 새로운 난수값을 지수승하여 생성할 수 있다.

<표 5> 다른 서버에 대한 로그인 단계

1 $U_i \rightarrow S_{new}$: $M = [(a_{i,j_1}, a_{i,j_2}, \ldots, a_{i,j_k}), S_{ID}, W \oplus g^{re}]$ 자신의 속성값, 이전 서버의 아이디, 암호화값을 전송한다.
2 $S_{new} \Rightarrow S$: 스마트카드로부터 전송받은 메시지에 난수값을 추가한 메시지 $M' = [(a_{i,j_1}, a_{i,j_2}, \ldots, a_{i,j_k}), W \oplus g^{re}, q)]$를 적합한 서버로 안전한 회선을 통해 전송한다.
3 S : 전송된 속성값 $a_{i,j_t} \in A, 1 \leq t \leq k$이 조건을 만족하는 경우 다음 단계를 진행한다.
4 S : $r'' = W \oplus g^{re} \oplus h(x_s)$를 복호화하여 속성값과 일치하는 세션키가 나오는 경우 다음단계를 진행한다.
5 S : 세션키의 해시값을 메시지 r''에 xor연산하여 메시지 $M' = g^{re} \oplus h(g^{re})$를 생성한다.
6 $S \Rightarrow S_{new}$: 새롭게 생성된 메시지 M''를 안전한 채널을 통해 전송한다.
6 S_{new} : 이전에 생성한 난수값r을 사용하여 새로운 세션키 $(g^{re} \| h(g^{re}))^q$를 생성하고 메시지 $M''' = q \oplus g^{re} \oplus h(g^{re})$를 생성한다.

메시지를 전송받은 스마트카드에서는 대칭키를 통해 복호화를 수행하여 서버를 인증한다. 인증이 성공하면 랜덤키에 타임스탬프를 지수승하여 새로운 대칭키를 생성하게 된다.

<표 6> 다른 서버에 대한 검증 단계

1 $S_{new} \rightarrow U_i$: M'''을 스마트카드로 전송한다.
2 S_{new} : 전송받은 암호문을 복호하기 위해 현재 세션키를 해시연산한 값과 세션키를 M'''에 xor연산한다. 이를 통해 난수 q를 생성할 수 있다.
3 S_{new} : 랜덤값을 기존의 키와 해시키값에 지수승하여 새로운 대칭키 $(g^{re} \| h(g^{re}))^q$를 설립한다.

Ⅳ. 분석

4.1. 안전성 분석

로그인과 인증단계는 Rhee et al.스킴의 안전성과 동일하다. 따라서 안전성에 대한 분석은 추가된 재인증단계에 대해서만 다룬다. 스킴의 재인증 단계는 가장 공격, 오프라인 패스워드 공격, 그리고 재사용 공격에 안전하다. 단 스마트카드의 temper resistant한 성질을 이용한 안전성은 고려하지 않는다.

-은밀한 검증자 공격 : 전송되어지는 메시지는 연산을 통해 암호화된 값이므로 공격자는 메시지와 서버로부터 인증정보를 얻는 것이 불가능하다.

-사용자 가장 공격 : 공격자는 정당한 인증메시지에 사용된 $W \oplus g^{re}$를 생성하는 것이 불가능하다. 그 이유는 서버의 비밀키값 x_s와 이전의 세션키값 그리고 사용자의 속성값을 알 수 없기 때문이다.

-서버 가장 공격 : 공격자는 자신이 서버임을 증명하기 위해 세션키를 생성할 수 있어야 한다. 하지만 서버의 비밀키값 x_s와 세션키 g^{re}를 알 수 없으므로 불가능하다.

-오프라인 패스워드 공격 : 재인증 단계에서는 새로운 사용자의 password정보가 사용되지 않는다. 따라서 공격자는 암호화 메시지로부터 사용자의 패스워드 정보를 얻는 것이 불가능하다.

-사용자의 익명성 : 재인증 단계에 사용되는 암호화된

값들은 사용자의 속성값에 기반하였기 때문에 속성값을 통해 사용자를 확인하는 것은 불가능하다.

4.2. 연산량 및 정보보호 기능 분석

제안된 스킴은 Rhee et al.스킴을 기반으로 하였기 때문에 기존의 연산량에 재인증 과정에 따른 연산량만 추가되었다. 재인증 과정에는 해쉬연산 한번, 지수연산을 한번 수행하여 새로운 세션키를 분배 및 기존 서버와 유저간의 상호인증을 가능하게 한다. 이는 새로운 서버에서의 인증과정을 기존의 인증된 정보를 통해 보다 효율적으로 수행하도록 한다.

<표 7> 프로토콜의 연산량 비교

프로토콜	계산량			
	로그인	인증	재인증	총계
Our scheme	2H+1E	2H+3E	• 1H+1E	4H+4E 5H+5E
Rhee et al. scheme[4]	2H+1E	2H+3E	•	4H+4E
H : 해시함수, E : 지수 연산, S : 대칭키 연산				

V. 결론

본 논문에서 제안된 스킴은 기존의 스마트카드를 이용한 인증 스킴을 확장하여 다양한 환경에서의 인증이 가능한 인증 스킴을 설계하였다. 인증은 스마트카드에서 새로운 서버에 요청이 가면 해당정보가 기존 서버로 전송되어 인증이 일어나며 인증된 메시지는 다시 새로운 서버에게 전송되는 방식으로 수행된다. 따라서 다양한 환경 속에서도 안전한 인증이 가능하다.

참고문헌

[1] H.Y. Chien, C.H. Chen, "A remote authentication scheme preserving user anonymity," IEEE AINA'05, vol 2, pp. 245-248, 2005.

[2] D. Boneh, M. Franklin, "Identity-based encryption from the weil pairing," CRYPTO, pp. 213-229, 2001.

[3] A. Sahai, B. Waters, "Fuzzy identity-based encryption," Proc. of EUROCRYPT'05, LNCS3494, pp. 457-473, 2005.

[4] 이현숙, 유혜정, "스마트카드를 이용한 속성기반 사용자 인증 스킴," 정보보호학회논문지, 제 18권 제 5호, pp. 41-47, 2008.