패스워드를 변경 가능한 효율적인 패스워드 기반의 키 교환 프로토콜

*이성운[O] **김현성 *유기영
*경북대학교 컴퓨터공학과, **경일대학교 컴퓨터공학과
{staroun[O], hskim, yook}@infosec.knu.ac.kr

Efficient Password-based Key Agreement Protocol with Password Changing

*Sungwoon Lee[O] **Hyunsung Kim *Keeyoung Yoo
*Dept. of Computer Engineering, Kyungpook Natl. Univ.,
**Dept. of Computer Engineering, Kyungil Univ.

요 약

본 논문에서는 사람이 기억할 수 있는 패스워드만을 이용하여 안전하지 않은 통신상에서 사용자와 서버 간에 서로를 인증하고 세션키를 공유하기 위한 새로운 키 교환 프로토콜을 제안한다. 제안된 프로토콜은 사용자가 자유롭게 자신의 패스워드를 변경할 수 있는 기능을 제공한다. 또한 여러 가지 다양한 공격들, 즉 패스워드 추측 공격, 중간 침입자 공격, Denning-Sacco 공격, Stolen-verifier 공격, 그리고 서비스 거부 공격에 안전하며, 완전한 전방향 보안성을 제공하도록 설계되었다.

1. 서 론

인터넷과 같은 개방된 통신 환경에서 안전한 통신을 하기 위해서는 인증과 메시지의 암호화가 중요하다. 또한 암호화를 위해서는 사용할 키의 공유가 선행되어야 한다. 사용자를 인증하는 방법은 여러 방식이 있지만 패스워드를 이용한 인증은 별도의 장비가 필요 없고 작은 패스워드만을 기억하면 되기 때문에 널리 이용되고 있다. 그러나 낮은 엔트로피를 가지는, 즉 사람이 기억할 수 있는 패스워드를 이용해야 하므로 패스워드 추측 공격에 취약할 수 있다.

패스워드를 이용하여 서로를 인증하고 키를 교환하는 프로토콜은 크게 두 종류로 분류될 수 있다[1]. 첫째는 동일 패스워드(Balanced password) 기반의 프로토콜로서 두 참여자는 같은 한 개의 패스워드를 사용하여 서로를 인증한다. 이 방식은 *Peer-to-Peer* 형태의 통신에는 효율적으로 사용될 수 있지만 클라이언트 서버 환경에 사용된다면 패스워드 파일이 공격자에게 노출될 경우 모든 사용자의 패스워드들이 드러나게 되어 프로토콜의 안전성이 크게 떨어질 수 있다. 둘째는 패스워드 검증자(Verifier) 기반의 프로토콜로서 클라이언트는 패스워드를 사용하고 서버는 패스워드로부터 유도된 값인 검증자를 패스워드 파일에 미리 저장해두고 프로토콜 수행 중에 해당 클라이언트에 대한 인증을 위한 검증 데이터로 사용한다. 이 방식은 서버의 패스워드 파일에 패스워드에 대한 검증자만을 저장하기 때문에 패스워드 파일이 노출되더라도 공격자는 이 검증자를 이용하여 직접 클라이언트로 위장하여 서버의 인증을 받을 수 없다. 이러한 검증자 기반의 프로토콜들로는 PAK-X[2], AMP[3], B-SPEKE[4], SRP[5], SNAPI-X[6], AuthA[7] 프로토콜 등이 있다.

본 논문에서는 사람이 기억할 수 있는 패스워드만을 이용하여 안전하지 않은 통신상에서 사용자와 서버간에 서로를 인증하고 세션키를 공유하기 위한 새로운 키 교환 프로토콜을 제안한다. 제안된 프로토콜은 사용자가 자유롭게 자신의 패스워드를 변경할 수 있는 기능을 제공한다. 또한 여러 가지 다양한 공격들, 즉 패스워드 추측 공격, 중간 침입자 공격, Denning-Sacco 공격, 그리고 stolen-verifier 공격에 안전하며, 완전한 전방향 보안성을 제공하도록 설계되었다. 더욱이 제안된 프로토콜은 구조가 간단하고 좋은 효율성을 제공한다.

2. 보안요구사항

본 장에서는 패스워드 기반의 키 교환 프로토콜들이 만족시켜야 할 보안 요구 사항들을 기술한다. 안전한 패스워드 기반의 키 교환 프로토콜을 설계하기 위해서는 다음과 같은 보안 요구 사항들이 고려되어야 한다.

- o 중간 침입자 공격(Man-in-the-middle attack)에 안전해야 한다. 키 교환 프로토콜은 안전하지 않은 통신망에서 메시지 교환을 통하여 세션키를 공유하고 서로를 인증한다. 그래서 공격자는 통신 선로 중간에서 전송 메시지들을 도청(Eavesdropping), 변경(*Modifying*), 반송(*Reflecting*), 재전송(Replay), 또는 위장(Masquerading)하여 공격할 수 있다. 키 교환 프로토콜은 이러한 공격들에도 세션키에 관한 정보를 노출시켜서는 안되며 잘못된 세션키의 공유를 탐지할 수 있어야 한다.
- o 패스워드 추측 공격(Password guessing attack)에 안전해야 한다. 패스워드 추측 공격은 온라인 패스워드 추측 공격과 오프라인 패스워드 추측 공격으로 나뉠 수 있다. 온라인 패스워드 추측 공격은 패스워드 인증 실패 횟수를 누적함으로써 쉽게 탐지되고 시도 횟수를 제한함으로써 쉽게 조치될 수 있다. 그러나 공격자는 안전하지 않은 통신상의 메시지를 가로채거나 정당한 사용자로 가장하여 다른 사용자와 세션키를 공유하는 과정 중에 발생하는 정보들을 저장해두고 오프라인으로 패스워드에 관한 정보를 알아내려고 할 수 있다. 이러한 오프라인 패스워드 추측 공격은 패스워드를 사용하는 키 교환 프로토콜들에 있어서 가장 큰 위협이다. 그러므로 패스워드 기반의 키 교환 프로토콜은 패스워드 추측 공격에 안전하도록 설계되어야 한다.
- o Denning-Sacco 공격에 안전해야 한다. Denning-Sacco 공격은 세션키가 노출되었을 때 공격자가 그 동안 통신상에서 도

청한 메시지들을 이용하여 패스워드에 관한 정보를 얻고자 하는 공격이다. 패스워드 기반의 키 교환 프로토콜은 이러한 공격에 안전해야 한다.

o 완전한 전방향 보안성(Perfect forward secrecy)을 제공해야 한다. 공격자가 참여자의 패스워드를 알아내었다 할지라도 이전에 사용된 세션키에 관한 정보는 알 수 없어야 한다.

패스워드 변경 프로토콜은 인증된 사용자가 자신의 패스워드를 변경할 수 있는 프로토콜이다. 패스워드 변경 프로토콜은 위에 언급된 공격들 이외에 서비스 거부 공격에 취약할 수 있다.

o 서비스 거부 공격에 안전해야 한다. 서비스 거부 공격은 통신 시설의 정상적인 사용이나 관리를 불가능하게 하는 공격이다. 예를 들어 공격자는 이 공격을 수행하여 서버가 특정 사용자의 로긴 요청을 허용하지 못하게 할 수 있다. 패스워드 변경 프로토콜은 이러한 공격에 안전해야 한다.

3. 제안된 프로토콜

본 장에서는 클라이언트-서버 환경에서 사람이 기억할 수 있는 패스워드를 이용하여 서로를 인증하고 세션키를 공유할 수 있는 키 교환 프로토콜을 제안한다.

3.1 초기설정

제안된 프로토콜에서 사용할 기호들에 대한 설명은 다음과 같다.

S	서버의 식별자
id	클라이언트의 아이디
n	큰 소수
g	곱셈군(multiplicative group) Z_p^*의 생성자(generator)
π	패스워드
υ	서버에 저장되는 패스워드 검증자(verifier)
t^{-1}	Z_p^*상에서 t의 역수
a, b	A와 B에 의하여 각각 선택된 Z_p^*의 임의의 원소
$h()$	일방향 해쉬 함수 (one-way hash function)
$\oplus$	비트 XOR (exclusive-OR) 연산
K_A, K_B	각각 A와 B의 세션키
$[M]_{K_A}$	세션키 K_A로 대칭키 암호화 알고리즘을 사용하여 M을 암호화

프로토콜의 두 참여자 클라이언트(A)와 서버(B)는 합법적인 참여자들이다. A와 B는 안전하게 Z_p^*상의 생성자인 g와 큰 소수인 n을 미리 공유하고 있다고 가정한다. $h()$는 $\{0,1\}^* \rightarrow \{0,1\}^k$인 collision-free한 성질을 만족하는 암호학적으로 강한 일방향 해쉬 함수(one-way hash function)[9]이다. 또한 A는 패스워드 π를 소유하고 있고 B는 A의 id와 패스워드의 검증자인 $\upsilon = g^{h(id,S,\pi)}$를 패스워드 파일에 저장하고 있다고 하자. 패스워드에 대한 검증자 υ는 패스워드 π를 이용하여 쉽게 계산될 수 있지만 그 역은 다항식 시간에 계산하기가 불가능하다. 제안된 프로토콜이 성공적으로 완료하면 A와 B는 $K_A = K_B = g^{ab}$를 세션키로 공유하게 된다. 프로토콜의 효율성을 높이기 위하여 A는 프로토콜이 시작하기 전에 미리 $\upsilon = g^{h(id,S,\pi)}$와 $h(id,S,\pi)^{-1}$을 계산할 수 있다. 'mod n' 연산 표기는 생략하기로 한다.

3.2 프로토콜의 수행

제안된 프로토콜은 다음과 같이 수행한다.

단계 1. A는 임의의 정수 a를 선택하고 $X_A = g^a \oplus \upsilon$를 계산하여 자신의 id와 함께 B에게 전송한다.

단계 2. B는 패스워드 파일로부터 A의 검증자 υ를 검색한다. 그리고 임의의 정수 b를 선택하여 $X_B = (\upsilon)^b \oplus \upsilon = g^{b \cdot h(id,S,\pi)} \oplus g^{h(id,S,\pi)}$를 계산하여 A에게 전송한다. 그리고 B는 A의 응답을 기다리는 동안 다음과 같이 $K_B = (X_A \oplus \upsilon)^b = g^{ab}$와 $V_B = [S, X_A]_{K_B}$를 계산한다.

단계 3. A는 B로부터 X_B를 받은 후에 다음과 같이 $K_A = (X_B \oplus \upsilon)^{a \cdot h(id,S,\pi)^{-1}} = g^{ab}$와 $V_A = [id, X_B]_{K_A}$를 계산하여 B에게 전송한다.

단계 4. B는 A로부터 V_A를 받은 후에 K_B를 사용함으로 V_A를 복호화하여 메시지의 송신자와 X_B가 정확한지를 검사한다. 그리고 V_B를 A에게 전송한다.

단계 5. A는 B로부터 V_B를 받은 후에 K_A를 사용함으로 V_B를 복호화하여 메시지의 송신자와 X_A가 정확한지를 검사한다.

제안한 프로토콜의 수행을 간단히 요약하면 다음과 같다.

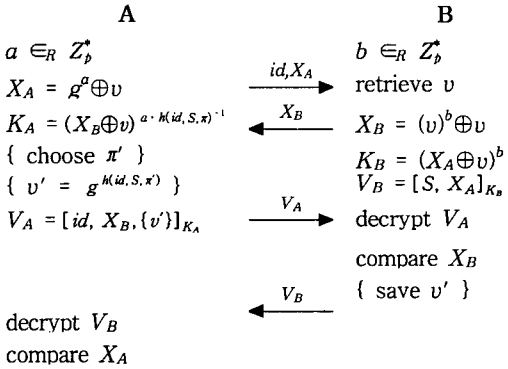

그림 2 제안한 프로토콜

그림2에 있는 '{ }'안의 내용은 클라이언트인 A가 자신의 패스워드를 변경하는 과정이다. 즉 A가 자신의 패스워드를 변경하고자 한다면 다음과 같이 프로토콜을 수행한다.

단계 3. A는 B로부터 X_B를 받은 후에 아래와 같이 $K_A = (X_B \oplus \upsilon)^{a \cdot h(id,S,\pi)^{-1}} = g^{ab}$를 계산한다. A는 새로운 패스워드 π'를 선택하고 새로운 검증자 $\upsilon' = g^{h(id,S,\pi')}$를 계산한다. 그리고 $V_A = [id, X_B, \upsilon']_{K_A}$를 계산하여 B에게 전송한다.

단계 4. B는 A로부터 V_A를 받은 후에 K_B를 사용함으로 V_A를 복호화하여 id와 X_B, 그리고 υ'를 구한다. 메시지의 송신자와 X_B가 정확한지를 검사하고 정확하다면 새로운 검증자 υ'를 패스워드 파일에 저장한다. 그리고 V_B를 A에게 전송한다.

단계 5. A는 B로부터 V_B를 받은 후에 K_A를 사용함으로 V_B를 복호화하여 메시지의 송신자와 X_A가 정확한지를 검사한다.

4. 안전성 분석

본 장에서는 2장에 기술된 각 공격들에 대하여 제안된 프로토콜이 안전함을 보이고자 한다.

4.1 중간 침입자 공격

공격자는 통신 선로 중간에서 전송 메시지들을 도청(Eavesdropping), 변경(Modifying), 반송(Reflecting), 재전송(Replay), 또는 위장(Masquerading)하여 공격할 수 있다.

첫째로, 수동적인 공격을 고려해보자. 공격자는 전송 메시지들을 도청하여 $X_A = g^a \oplus g^{h(id,S,\pi)}$, $X_B = g^{b \cdot h(id,S,\pi)} \oplus g^{h(id,S,\pi)}$, $V_A = [id, X_B, \{v'\}]_{K_A}$, $V_B = [S, X_A]_{K_B}$를 얻을 수 있다. 그러나 공격자가 이 값들을 획득한다 하더라도 패스워드 π와 세션키 K를 계산할 수 있는 방법은 없다.

둘째로, 적극적인 공격자의 수정 공격을 고려하자. 공격자가 X_A와 X_B를 중간에서 수정하여 상대에게 전송한다면, 이 위조된 값들은 A와 B에 의해 K_A와 K_B를 생성하는데 각각 사용되게 된다. 그러나 A는 임의의 정수 a를 사용하여 K_A를 계산하고 B는 임의의 정수 b를 사용하여 K_B를 계산하기 때문에 K_A와 K_B의 값이 같게 될 확률은 무시할만하다. 결국, 이 공격은 세션키 K_A와 K_B를 다르게 만들므로 검증 시에 정확한 X_A와 X_B를 얻을 수 없어 탐지될 수밖에 없다.

셋째로, 적극적인 공격자의 재전송 공격을 고려하자. 재전송 공격은 이전 세션의 전송 메시지들을 저장해 두었다가 이후 세션들에 이용하는 공격이다. 그러나 매 세션마다 각 참여자들은 새로운 임의의 난수 a와 b를 생성하여 사용한다. 공격자가 이 난수들을 알 수 있는 확률은 무시할 만하다.

넷째로, 공격자는 합법적인 참여자로 위장하여 정상적인 방법으로 다른 합법적인 참여자와 세션키를 공유하려고 할 수 있다. 그러나 이러한 위장 공격은 공격자가 패스워드를 알지 못하기 때문에 합법적인 참여자가 생성한 정확한 세션키를 생성할 수 없어 검증 시에 탐지될 수밖에 없다.

결국 제안한 프로토콜들은 이와 같은 중간 침입자 공격들에 안전하다.

4.2 패스워드 추측 공격

먼저 도청한 메시지만을 이용하는 수동적인 패스워드 추측 공격을 고려하자. 공격자는 메시지 X_A, X_B, V_A, V_B를 가로채 저장하고, 패스워드로 사용될 수 있는 π'를 추측한다. 그리고 π'를 도청한 값들에 적용하여 비교함으로 검증한다. 이를 모든 패스워드 범위에 대하여 반복 수행함으로써 추측한 π'가 참여자들이 사용하고 있는 정확한 π인지를 확인할 수 있어야 한다. 그러나 제안된 프로토콜에서는 전송 메시지인 X_A, X_B, V_A, V_B에 π'를 적용하여도 π'가 정확한지를 검증할 방법이 없다. 또한 공격자가 정당한 참여자로 위장한 적극적인 패스워드 추측 공격을 고려해 보자. 공격자가 A로 위장한다면 자신이 만든 a와 g^a, 그리고 B로부터 받은 $g^{b \cdot h(id,S,\pi)} \oplus g^{h(id,S,\pi)}$를 얻을 수 있다. 그러나 이들을 이용해서는 π'가 정확한지를 검증할 방법이 없다. 그리고, 공격자가 B로 위장한다면 자신이 생성한 b와 g^b, 그리고 A로부터 받은 $g^a \oplus g^{h(id,S,\pi)}$와 $[id, g^b, \{g^{h(id,S,\pi)}\}]_{(g^b \oplus g^{h(id,S,\pi)})^{a \cdot h(id,S,\pi)}}$ 값들을 얻을 수 있다. 그러나 이 값들을 이용해서도 π'가 정확한지를 검증할 방법이 없다. 그러므로 제안된 프로토콜들은 패스워드 추측 공격에 안전하다.

4.3 Denning-Sacco 공격

Denning-Sacco 공격은 세션키가 노출되었을 때 공격자가 패스워드에 관한 정보를 얻고자 하는 공격이다. 제안된 프로토콜들에서 공격자가 임의의 세션에서 도청을 통해 X_A, X_B, V_A, V_B를 얻었고, 세션키 g^{ab}가 공격자에게 노출되었다고 가정하자. 그러나 이 값들로부터 패스워드 π를 구할 방법은 없다.

4.4 완전한 전방향 보안성

완전한 전방향 보안성을 제공하기 위해서는 패스워드가 공격자에게 노출되었다 할지라도 이전의 세션키들은 안전해야 한다. 제안된 프로토콜들에서 공격자에게 패스워드 π가 노출되었다고 하자. 공격자는 도청을 통해 X_A, X_B, V_A, V_B를 얻을 수 있다. 그러나 이 정보들로부터 세션키인 g^{ab}를 구할 방법은 없다.

4.5 Stolen-verifier 공격

Stolen-verifier 공격은 서버로부터 패스워드 검증자를 훔친 공격자가 직접적으로 합법적인 사용자를 가장하려는 공격을 의미한다. 제안된 프로토콜에서 서버에 저장된 패스워드 검증자는 $v = g^{h(id,S,\pi)}$ 이다. 이 자료를 훔친 공격자는 새로운 세션의 3단계에서 사용해야 하는 π를 알지 못하므로 직접적으로 사용자를 가장할 수 없다.

4.5 서비스 거부 (Denial of Service) 공격

제안된 프로토콜에서 공격자가 서버로 하여금 계속적으로 특정 사용자에 대한 인증을 거부하도록 하려면 사용자가 의도하지 않은 패스워드 검증자를 서버가 저장하도록 할 수 있어야 한다. 그러나 공격자는 세션키를 알지 못하기 때문에 패스워드 검증자를 자신이 생성한 값으로 대체시킬 수 없다. 그러므로 제안된 프로토콜은 서비스 거부 공격에 안전하다.

5. 결론

패스워드 기반의 프로토콜은 사람들이 패스워드와 같은 작은 지식만을 기억하면 되기 때문에 널리 이용되고 있다. 본 논문에서는 클라이언트-서버 환경에서 사용될 수 있는 상호 인증 가능하고 사용자가 패스워드를 자유롭게 변경할 수 있는 패스워드 기반의 키 교환 프로토콜을 제안하였다. 이 프로토콜은 중간 침입자 공격, 패스워드 추측 공격, Denning-Sacco 공격, Stolen-verifier 공격, 그리고 서비스 거부 공격에 안전하고 완전한 전방향 보안성을 제공하도록 설계되었다.

참고문헌

[1] IEEE. Standard Specifications for Public Key Cryptography, IEEE1363, 2002.

[2] V. Boyko, P. MacKenzie and S. Patel. "Provably Secure Password-Authenticated Key Exchange Using Diffie-Hellman," Advances in Cryptology-EUROCRYPT '2000, pp. 156-171, 2000.

[3] T. Kwon. "Ultimate Solution to Authentication via Memorable Password," Presented to IEEE P1363a, May 2000.

[4] D. Jablon. "Extended password key exchange protocols," WETICE Workshop on Enterprise Security, 1997.

[5] T. Wu. "Secure remote password protocol," Internet Society Symposium on Network and Distributed System Security, 1998.

[6] P. MacKenzie, S. Patel, and R. Swaminathan. "Password-authenticated key exchange based on RSA." In ASIACRYPT2000.

[7] M. Bellare and P. Rogaway, "The AuthA protocol for password-based authenticated key exchange," Presented to IEEE P1363a, March 2000.

[8] D. R. Stinson, Cryptography Theory and Practice, CRC, 1995.