제20회 한국정보처리학회 추계학술발표대회 논문집 제10권 제2호 (2003. 11)

효율적인 패스워드 기반 키 분배 프로토콜

원동규*, 안기범*, 곽 진*, 원동호*
*성균관대학교 컴퓨터공학과
e-mail : {dkwon, gbahn, jkwak, dhwon}@dosan.skku.ac.kr

Efficient password-based key exchange protocol

Dongkyu Won*, Gibum Ahn*, Jin Kwak*, Dongho Won*
*Dept of Computer Engineering, Sungkyunkwan University

요 약

최근 키 분배 프로토콜과는 다르게 하드웨어에 암호키를 저장하여 사용하는 것과 달리, 사용자가 기억할 수 있는 길이의 패스워드(password)를 사용해 서버와의 인증과 키 교환을 동시에 수행하는 패스워드 기반 키 분배 프로토콜이 제안되고 있다. 본 논문에서는 이러한 패스워드 기반의 키 분배 프로토콜 중 BPKA(Balanced Password-authenticated Key Agreement)에 속하는 DH-EKE(Diffie-Hellman Encrypted Key Exchange), PAK(Password-Authenticated Key exchange), SPEKE(Simple Password Exponential Key Exchange) 프로토콜을 비교· 분석하고, 이를 바탕으로 기존의 BPKA 프로토콜에 비해 적은 연산량을 가지면서 사용자와 서버가 각기 다른 정보를 갖는 패스워드-검증자 기반 프로토콜을 제안한다. 본 논문에서 제안하는 패스워드 기반 키 분배 프로토콜의 안전성 분석을 위해 Active Impersonation 과 Forward Secrecy, Off-line dictionary attack, Man-in-the -middle Attack 등의 공격모델을 적용하였다.

1. 서론

개체 인증(entity authentication)은 네트워크상에서 통신하는 상대방의 신분확인을 위해 필요하다. 이러한 기능은 개체 사이에 키 동의(key agreement), 혹은 키 전송(key transport) 같은 세션 키 생성 스킴과 결합되어진다. 패스워드 기반 키 분배 프로토콜[1]도 개체 인증과 세션 키 생성을 제공하는 프로토콜로써, 대칭키나 비대칭키와 같이 긴 길이의 키를 사용하는 키 분배 프로토콜에 비해 사용자가 기억하기 용이한 길이의 비밀정보를 사용하기 때문에 하드웨어의 요구사항이 적고 편리하며, 간편하다는 장점 때문에 널리 사용되고 있다. 하지만 패스워드 기반 인증 방법은 정보량적인 측면에서 낮은 엔트로피(low entropy)를 가지기 때문에 패스워드에 대한 추측공격(Guessing Attack)에 취약하며, 서버에 저장되어 있는 패스워드 파일이 공격자에게 노출되었을 경우, 사전공격(Dictionary Attack)이 가능하다는 문제점을 가지고 있다.

패스워드 기반 프로토콜은 1989 년 M.Lomas, L.Gong, J.Saltzer, and R.Needham 에 의해 소개된 LGSN[2] 스킴 이후로 계속해서 제안되고 있다. 그 중에서 서버의 공개키를 사용하지 않고, 인증서(Certificate) 없이 인증기능을 제공하는 실질적인 패스워드 기반 키 분배 프로토콜인 EKE (Encrypted Key Exchange)[3]가 제안되었고, 이후 EKE 보다 안전한 DH-EKE 가 제안되었다. 사용자와 서버가 동일한 패스워드 정보를 가지는 패스워드-패스워드기반 프로토콜의 경우, 사용자와 서버가 동일한 비밀정보를 가지고 있기 때문에 문제가 발생하였을 경우 대칭키 암호 방식처럼 제 3 자가 분쟁을 해결을 할 수 없으며, 서버가 사용자로 위장할 수도 있는 문제점이 발생한다. 또한, 서버에서 패스워드 파일이 노출되었을 경우 프로토콜의 공격이 가능해지므로 패스워드 파일의 안전한 보관이 요구된다. 이에 Salt 와 검증자를 이용해 패스워드 파일의 노출에도 안전하고, 프로토콜에 참여한 사용자와 서버가 서로 다른 정보를 가지고 있기 때문에 위장이 불가능한 패스워드-검증자 기반 프로토콜이 제안되었다. 패스워드-검증자 기반 프로토콜로는 B-SPEKE[4], PAK-X[5] 등이 있으며, 이러한 프로토콜들은 패스워드-패스워드 기반 프로토콜의 경우보다 추가적인 지수연산이 요구되는 단점을 가지고 있다. 본 논문에서는 패스워드-패스워드 기반 키 분배 프로토콜과 같은 지수연산을 제공하는 효율적인 패스워드-검증자 기반 키 분배 프로토콜을 제안한다.

본 논문의 구성은 다음과 같다. 2 절에서는 기존의 패스워드 기반 키 분배 프로토콜에 대해 분석하고, 3 절에서는 제안하는 키 분배 프로토콜에 대해서 기술한다. 4 절에서는 제안한 프로토콜의 안전성과 특징을 분석하고, 5 절에서는 기존 프로토콜과의 계산량에 대한 효율성을 비교한다. 마지막으로 6 절에서는 결론과 향후 연구 방향에 대해 서술한다.

2. 관련연구

본 장에서는 기존의 대표적인 프로토콜인 SPEKE [6], DH-EKE, B-SPEKE, PAK[5], PAK-X 에 대해 살펴본다. SPEKE 와 DH-EKE 는 사용자와 서버사이에 인증을 위해 공통 비밀정보 S 를 공유하는데 공통 비밀정보 S 는 패스워드를 해쉬한 값이며, 프로토콜의 수행 이전에 공유하고 있음을 가정한다. SPEKE 를 기반으로 발전한 B-SPEKE 와 해쉬함수를 이용한 PAK-X 는 패스워드-검증자 기반 프로토콜로써 프로토콜 수행이전에 사용자는 패스워드를 가지고 있고, 서버는 패스워드를 이용하여 생성한 검증자를 사용자로부터 전송받아 비밀리에 보관한다.

2.1 SPEKE

SPEKE 는 이산대수 문제와 Diffie-Hellman 문제의 어려움에 기반하고 있다. SPEKE 는 두 단계로 구성되는데, 첫 단계는 Diffie-Hellman 키 교환 방식으로 키를 생성하는 것이고, 두 번째 단계는 키를 확인하는 과정이다. Diffie-Hellman 키 분배 과정의 지수연산에서 원시원소 g 를 사용하는 대신에 패스워드를 해쉬한 값인 공통 비밀정보 S 를 사용한다. SPEKE 프로토콜은 패스워드-패스워드 기반으로 수행되기 때문에 서버는 사용자로 위장할 수 있으며, 서버에서 패스워드 파일이 노출이 될 경우 프로토콜에 대한 공격이 가능하다.

2.2 DH-EKE

DH-EKE 는 SPEKE 와 마찬가지로 이산대수 문제와 Diffie-Hellman 문제의 어려움에 기반하고 있으며 두 단계로 나누어 수행된다. SPEKE 와는 다르게 지수연산에 원시원소 g 를 사용하며, 패스워드는 키 분배 프로토콜 중 전송 과정에서 세션키 생성 비밀정보를 대칭 암호 방식으로 암호화 하는데 사용된다. 대칭 암호 방식은 간단히 XOR(Exlusive-OR)의 수행으로도 가능하다. 패스워드-패스워드 기반 프로토콜이므로 SPEKE 의 문제점과 동일하다.

2.3 B-SPEKE

B-SPEKE 는 SPEKE 방식의 패스워드-패스워드 기반 방식을 패스워드-검증자 기반 방식으로 변형한 프로토콜이다. B-SPEKE 는 검증자와 패스워드 해쉬값을 통해 SPEKE 의 문제점을 해결하였고, SPEKE 의 해쉬값을 이용한 패스워드 검증 과정에 패스워드 자체를 이용한 검증과정을 더해 두 번의 검증을 수행한다. 하지만 기존의 프로토콜에 비해 추가적인 연산량을 요구한다는 단점을 가지고 있다.

2.4 PAK, PAK-X

이 프로토콜들은 Diffie-Hellman 기반 패스워드 키 분배 프로토콜로써 이산대수문제와 해쉬함수의 안전성에 기반한다. 패스워드-패스워드 기반 PAK 와 패스워드-검증자 기반인 PAK-X 로 나누어 지며, 프로토콜은 지수연산과 구분된 3 번의 해쉬함수로 수행된다.

3. 제안하는 키 분배 프로토콜

본 절에서는 제안하는 패스워드 기반 프로토콜에서 사용되는 시스템 파라미터에 대해 정의하고, 패스워드와 검증자를 사용하여 사용자와 서버간의 세션키를 분배하는 키 분배 프로토콜을 기술한다.

- A: 사용자
- B: 서버
- w: 패스워드
- V: 검증자
- p: $GF(p)$ 를 정의하는 큰 소수
- g: Z_p 에서 위수 $p-1$ 를 갖는 원시원소
- SK: 세션키
- $h(\ )$: 일방향 해쉬함수
- r_A: 사용자 A 가 생성한 랜덤수
- r_B: 서버 B 가 생성한 랜덤수
- SK: 사용자 A 와 서버 B 사이에 생성한 세션키

사용자에 대한 인증과 세션키를 교환하기 위해 사용자 A 와 서버 B 는 다음과 같은 과정을 수행하며, 사용자 A 는 패스워드를, 서버 B 는 사용자 A 가 생성하여 전송해 준 검증자 $V = g^{w^{-1}}$ 를 사전에 공유하고 있음을 가정한다.

1. 사용자 A 는 Z_p^* 상에서 랜덤수 r_A 를 선택하고, 모듈러(modulo) p 상에서 원시원소 g 에 지수승한 값 R 을 계산하여 서버 B 에게 전송해 준다.

$$R = g^{r_A} \bmod p$$

2. 서버 B 는 수신한 R 과 Z_p^* 상에서 선택한 랜덤수 r_B 를 이용해 세션키 SK 를 생성하고 검증자 V 와 랜덤수 r_B 를 이용하여 중간값 U 를 계산한다. 키 생성을 위해 서버 B 는 사용자 A 에게 U 를 전송해 주고, 키 확인을 위해 세션키 SK 를 일방향 함수로 두 번 해쉬하여 전송해준다.

$$SK = R^{r_B} \bmod p$$

$$U = V^{r_B} \bmod p$$

$$H = h(h(SK))$$

3. 사용자 A 는 서버로부터 수신한 중간값 C, 자신의 패스워드 w, 랜덤수 r_A 를 사용하여 세션키 SK' 를 계산한다. 생성한 세션키가 서버 B 가

생성한 세션키 SK 와 동일한지를 확인하기 위해 SK' 를 두 번 해쉬하여 서버 B 로부터 수신한 $h(h(SK))$ 와 비교 · 확인한다. 서버 B 에게 키 확인을 위해 세션키를 한번 해쉬한 값 $h(SK')$ 를 전송해 준다.

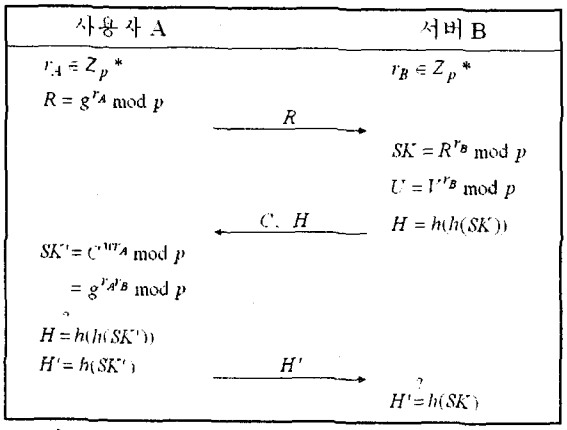
$$SK' = C^{w r_A} \bmod p = g^{r_A r_B} \bmod p$$

$$h(h(SK')) \stackrel{?}{=} H$$

$$H' = h(SK')$$

4. 서버 B 는 사용자 A 로부터 수신한 $h(SK')$ 와 자신이 생성한 세션키 SK 를 해쉬하여 비교함으로써 키 확인을 수행한다.

$$H' \stackrel{?}{=} h(SK)$$



[그림 1] 제안하는 패스워드 기반 키 분배 프로토콜

4. 제안한 프로토콜의 안전성 및 특징 분석

4.1 안전성 분석

본 논문에서 제안하고 있는 키 분배 프로토콜은 기본적으로 그 안전성이 이산대수 문제에 기반하므로 공개 정보와 전송 정보를 이용하여 세션키를 구하는 어려움은 이산대수 문제를 푸는 어려움과 동일하다.

- Active Impersonation 에 대한 안전성

공격자가 세션을 시작한 경우, 공격자가 사용자 A 로 위장하여 서버 B 와의 세션키를 설정하는 것은 이산대수문제를 푸는 어려움과 동일하다. 공격자는 패스워드 w 를 알지 못하면 정당한 세션키를 생성할 수 없으므로 위장이 불가능하고 서버 B 로 위장하였을 경우 검증자 V 를 알지 못하기 때문에 동일한 세션키를 생성하지 못한다. 이는 키 확인 과정에서 정당하지 않은 전송 정보 $h(SK'')$ 를 사용자 A 와 서버 B 가 확인했을 때 공격여부에 대해 알 수 있다.

- Forward Secrecy 에 대한 안전성

사용자 A 의 비밀정보인 패스워드가 노출된 경우 공격자는 랜덤수 r_A, r_B 를 알지 못하기 때문에 과거의 세션키를 구할 수 없고, 서버 B 의 검증자 V 가 노출되었을 경우에도 사용자 A 의 패스워드와 랜덤수 r_A, r_B 를 알 수 없으므로 과거의 세션키의 안전성에는 문제가 없다.(Half Forward Secrecy)

또한 사용자 A, 서버 B 의 패스워드와 검증자가 모두 노출되었다 하더라도 과거의 랜덤수를 알 수 없으므로 세션키를 구할 수 없다.(Full Forward Secrecy)

- Man-in-the-middle Attack 에 대한 안전성

Man-in-the-middle Attack 은 공격자가 사용자 A 와 서버 B 각각에게 정당한 사용자로 위장하여 공격하는 방식이다. 사용자 A 가 랜덤수 r_A 를 서버 B 로 위장한 공격자에게 전송해 주면, 공격자는 랜덤수 r_C 와 중간값 R' 값을 계산하여 서버 B 에게 R' 를 전송해 준다. 하지만 이러한 공격방법의 경우 다음 과정에서 문제가 발생한다. 서버 B 가 H'' 와 C'' 를 계산하여 공격자에게 전송했을 경우 공격자는 패스워드 값 w 을 모르기 때문에 정당한 SK 를 계산해 낼 수가 없다. 또한 서버의 검증자 V 값 정보 또한 없기 때문에 사용자 A 에게 정당한 서버 B 로 위장할 수 없다.

- Off-line dictionary attack 에 대한 안전성

Off-line dictionary attack 은 키 교환 과정 동안 수행한 사용자들간에 전송정보를 이용하여 공격자가 패스워드 w 나 세션키 SK 를 구하는 방법이다. 첫 번째, 제안한 프로토콜에서 공격자가 패스워드 w 에 대해 off-line dictionary attack 을 수행하려는 경우, C 로부터 패스워드를 계산하기 위해서는 세션키 SK, 랜덤수 r_A 의 정보가 필요하다. 하지만 이 정보를 계산하는 것은 이산대수문제의 어려움과, 해쉬함수 역함수를 구하는 어려움과 동일하다. 두 번째 세션키를 계산하려는 경우에서, C 와 M 에 대해 off-line dictionary attack 을 수행하는 방법은 이산대수문제와 Diffie-Hellman 문제의 어려움과 동일하고, 전송 정보 H' 와 H 에 대해 off-line dictionary attack 을 수행하는 것은 해쉬 함수의 역함수를 구하는 어려움과 동일하다.

4.2 프로토콜의 특징 분석

제안한 프로토콜의 세션키 설정에 필요한 통신 횟수, 상호 개체 인증, 키 인증, Key freshness, 키 확인(Key confirmation)의 특징을 제공하는지에 대해 분석한다.

제안한 프로토콜은 세션키를 설정하는 과정에서 사용자 A 와 서버 B 가 각각 생성한 랜덤수 r_A, r_B 를 이용하여 계산하므로 키 동의와 Key freshness 을 보장한다. 그리고 사용자 A 는 서버 B 가 전송한 세션키의 해쉬 값 H 와 자신이 생성한 세션키의 해쉬 값을 비교하여 서버 B 임을 인증하고, 서버 B 는 사용자 A 가 전송해준 해쉬 값 H' 와 비교하여 사용자 A 임을 인증하므로 사용자 A 와 서버 B 모두 양방향 개

체 인증을 제공한다. 또한 제안한 프로토콜은 양방향 명시적 키 인증과 키 확인을 보장하는데, 이는 사용자 A 와 서버 B 는 패스워드와 검증자를 가져야만 키를 생성할 수 있고 해쉬값을 통한 확인과정을 수행하기 때문이다. 프로토콜의 특징에 대한 분석을 정리하면 아래 [표 1]과 같다.

[표 1] 제안한 프로토콜의 특징 분석

구 분	키 분배 프로토콜	
통신횟수	3회	
개체인증	양방향	
키인증	A	명시적 키 인증
	B	명시적 키 인증
키 확인	양방향	
Key freshness	양방향	

5. 효율성 분석

본 논문에서 제안하는 패스워드 기반 키 분배 방식은 계산량적 측면에서 장점을 갖는다. 이의 분석을 위해 키 분배 프로토콜의 분배 과정 및 검증과정에서 수행되는 지수연산과 해쉬함수를 기준으로 계산량을 비교한다. 지수연산을 기준으로 기존 프로토콜과 제안한 프로토콜의 계산량을 비교하고 지수연산이 비슷할 경우 해쉬함수의 연산을 이용한다.

패스워드-패스워드 기반프로토콜인 SPEKE, DH-EKE 는 사용자 2 번, 서버 2 번의 지수연산을 수행하고, PAK 은 사용자 3 번, 서버 2 번을 요구한다. 하지만 패스워드-검증자 기반 프로토콜인 B-SPEKE 는 사용자 3 번, 서버 2 번이고, PAK-X 는 사용자 5 번, 서버 4 번의 많은 연산을 요구한다. 제안한 프로토콜의 경우 패스워드-검증자 기반 프로토콜로써 패스워드-패스워드 기반 프로토콜과 같이 사용자 2 번, 서버 2 번의 적은 지수연산만을 요구하므로 효율적이다. 내용을 정리하면 위의 [표 2]와 같다.

[표 2] 계산량 비교

구 분	지수연산		해쉬연산	
	사용자	서버	사용자	서버
SPEKE/DH-EKE	2	2	3	3
PAK	3	3	4	4
B-SPEKE	3	4	-	-
PAK-X	5	4	5	5
제안한 프로토콜	2	2	3	3

6. 결론 및 향후 연구 방향

본 논문에서는 먼저, 기존의 패스워드 기반 키 분배 프로토콜중 BPAK 프로토콜에 대해 분석하고, 이를 기반한 새로운 프로토콜을 제안하였다. BPAK 는 패스워드-패스워드 기반 키 분배 프로토콜과 패스워드-검증자 기반 프로토콜이 있다. 패스워드-패스워드 기반 키 분배 프로토콜은 계산량이 패스워드-검증자에 비해 적지만 서버는 사용자로 위장이 가능하고, 서버에서 패스워드가 노출될 경우 프로토콜에 대한 공격이 가능하다는 문제점을 가진다. 이에 반해 패스워드-검증자 기반 프로토콜은 서버에 의해 검증자가 노출되어도 프로토콜은 안전하며, 서버가 사용자로 위장할 수 없는 장점을 가지지만 많은 계산량을 요구하므로 효율성이 떨어지는 문제점이 있다. 이에 패스워드-검증자의 안전성을 가지면서 패스워드-패스워드 기반 키 분배 프로토콜과 비슷한 계산량을 가진 프로토콜을 제안하였다. 또한 제안한 프로토콜은 명시적 키 인증과 양방향 키 확인, 개체인증, key freshness 의 보안 요구 사항을 만족한다.

본 논문에서 제안하는 프로토콜의 안전성은 이산대수문제와 Diffie-Hellman 문제에 의존하고 있으며, Active Impersonation, Forward Secrecy, Man in the middle attack, Off-line dictionary attack 에 대해 안전함을 분석하였다. 그러나 본 논문에서 기술한 안전성은 경험적 안전성에 기초하여 분석하였으므로 향후 formal 모델에서의 안전성 증명이 이루어져야 할 것이다. 또한 BPKA 프로토콜과는 다른 형태의 패스워드 기반인 APKA(Augmented Password-authenticated Key Agreement)에 대한 연구를 통해 보다 효율적인 프로토콜을 고려하여야 할 것이다.

참고문헌

[1] Bellare, Jablon, Krawczyk, MacKenzie, Rogaway, Swaminathan & Wu, "Proposal for P1363 Study Group on Password-Based Authenticated-Key-Exchange Methods", February 2000

[2] M.Lomas, L.Gong, J.Saltzer, and R.Needham, "Reducing risks from poorly chosen keys," Proceedings of the 12th ACM Symposium on Operating System Principles, ACM Operating Systems Review, 1989, pp.14-18

[3] Steven M. Bellovin, "Encrypted Key Exchange: Password-Based Protocols Secure Against Dictionary Attacks", Proceedings of the IEEE Symposium on Research in Security and Privacy, May 1992

[4] David P. Jablon, "Extended Password Key Exchange Protocols Immune to Dictionary Attack*", Proceedings of the WETICE , June, 1997

[5] Victor Boyko, Philip MacKenzie, Sarvar Patel, "Provably Secure Password-authenticated key Exchange Using Diffie-Hellman", July 2000

[6] David P. Jablon, "Strong Password-Only Authenticated Key Exchange", ACM Computer Communication, October 1996