OTP-EKE : 원-타임-패스워드 기반의 키 교환 프로토콜

(OTP-EKE: A Key Exchange Protocol based on One-Time-Password)

서 승 현 † 조 태 남 †† 이 상 호 †††

(Seunghyun Seo) (Taenam Cho) (Sang-Ho Lee)

요 약 키 교환 프로토콜에서 상호 인증은 필수 요소이며, 사용자에게 편리하고 비용이 적게 드는 패스워드 기반의 인증 방식이 널리 사용되고 있다. 패스워드 기반의 프로토콜은 패스워드가 가지는 제약으로 인한 공격에 대해서 안전해야 할 뿐 아니라, 사용자의 작업량을 줄이기 위한 효율성도 매우 중요한 요건이다. 본 논문에서는 서버와 사용자간의 인증을 제공하고 세션키를 공유하기 위한 키 교환 프로토콜 OTP-EKE(One Time Password based Encrypted Key Exchange)를 제안하였다. 키 교환을 위한 사용자 인증방식으로 패스워드 방식을 채택하였다. 특히 서버 디렉토리에 대한 공격 등에 대해서 안전도를 높이기 위하여 원-타임-패스워드 확인자와 서버의 공개 패스워드를 이용하였다. 제안한 프로토콜은 모듈라 지수승 계산 횟수와 메시지 전송 횟수를 줄임으로써 효율성 향상을 보인다.

키워드 : 키 교환 프로토콜, 패스워드 확인자, 원-타임-패스워드, 인증

Abstract Mutual authentication is essential for key exchange protocols and password-based authentication scheme is used widely, which is convenient to users and executed on the cheap. Password-based protocol should be not only secure against attack but also efficient to reduce user's load. In this paper, we propose a new key exchange protocol, called OTP-EKE(One Time Password based Encrypted Key Exchange), to provide authentication and to share a session key between a server and a user. We choose a password-based scheme as a user authentication. Especially, we use a one-time-password verifier and server's public password to protect against attacks on server's directory. As for efficiency, we improve the performance by reducing the number of modular exponentiations and the number of rounds.

Key word : key exchange protocol, password-verifier, one-time-password, authentication

1. 서 론

통신 및 네트워크 기술의 발전은 전자 상거래, 원격지 사용자간의 통신, 응용서버와의 통신 등의 서비스를 창출하였다. 특히 인터넷과 같은 개방형 네트워크 상에서 통신 상대간에 안전한 통신을 하기 위해서는 전송될 정보를 암호화하여야 하며, 이를 위해서 통신 상대자간에 공통으로 사용할 수 있는 키의 공유가 우선되어야 한다. 이 때 통신 상대자간에는 통신을 통해 정보를 교환하고 있는 상대가 실제 의도한 상대인지를 확인하는 인증 과정이 반드시 필요한데 이러한 인증 기능은 사용자가 응용 서버로부터의 서비스를 받기 위해서도 필요하다.

따라서 암호화키를 공유하는 문제와 사용자 인증 문제는 안전한 정보 교환과 개인 정보보호를 위해 해결해야 할 중요한 문제이며, 이를 위해서는 보다 효율적인 프로토콜 개발이 절실히 필요하다.

사용자 인증 방식은 인증의 기반이 되는 요소가 무엇이냐에 따라 다음과 같이 세 가지로 분류된다. 첫째, 목소리 식별, 망막 검사 등과 같이 사용자의 물리적인 특징을 이용하는 인증 방법, 둘째, ID card나 smart card

† 학생회원 : 이화여자대학교 과학기술대학원 컴퓨터학과
happyday@ewha.ac.kr
†† 정 회 원 : 이화여자대학교 과학기술대학원 컴퓨터학과
tncho@ewha.ac.kr
††† 종신회원 : 이화여자대학교 컴퓨터학과 교수
shlee@ewha.ac.kr
논문접수 : 2001년 10월 10일
심사완료 : 2002년 1월 14일

등과 같이 사용자가 소유한 물건을 통한 인증 방법, 셋째, 패스워드와 같이 사용자가 알고 있는 지식을 통한 인증 방법이다. 첫 번째와 두 번째 방식은 강력한 보안을 위해 사용되기는 하지만 그에 따르는 부가적인 하드웨어 비용이 크다. 반면 세 번째 방식은 별도로 필요한 장비가 없기 때문에 큰 비용을 들이지 않고도 쉽게 사용될 수 있어 많이 이용되고 있으며 패스워드 프로토콜들이 이에 해당된다[11].

그러나 사용자들은 쉽게 기억할 수 있는 패스워드를 선택하는 경향이 있기 때문에, 패스워드 자체를 암호화하는 키로 이용한다면 패스워드 추측 공격을 당할 위험이 있고 암호 시스템 자체를 취약하게 만들 수도 있다. 이를 보완하고 안전한 키 공유도 동시에 하기 위해서 패스워드와 부수적인 매개변수를 이용하는 패스워드 기반 키 교환 프로토콜들이 인증 및 키 교환 프로토콜로 사용되고 있다[11][15].

이러한 프로토콜들은 크게 다음과 같이 구분될 수 있다. 첫 번째는 서버가 사용자 패스워드의 복사본을 저장하고 있어야 하는 평문등가 프로토콜(plaintext-equivalent protocol)이고[3][4][11][14][17], 두 번째는 서버가 오직 사용자의 패스워드에 대한 확인자(verifier)만을 저장하는 확인자 기반 프로토콜(verifier-based protocol)이다[2][4][5][10][12][13][14][18].

평문등가(plaintext-equivalent) 방식은, 사용자의 패스워드를 서버에 저장해야 하기 때문에 공격자의 공격에 의해 서버에 저장된 사용자 패스워드 파일이 노출될 경우, 모든 사용자에 대한 정보가 노출되어서 안전성이 떨어진다. 반면 확인자 기반(verifier-based) 방식은 서버에 사용자 패스워드의 확인자만이 저장되어 있으므로, 서버가 공격을 당하더라도 패스워드 확인자만 공격자에게 노출된다. 패스워드 확인자만으로는 공격자가 서버에게 인증 받을 수 없으므로, 공격자는 비용이 큰 사전공격(dictionary attack)을 수행해서 사용자의 패스워드를 알아내는 수고를 해야만 한다. 따라서 확인자 기반 방식이 평문등가 방식에 비하여 더 안전하다고 할 수 있다.

본 논문에서 제안하는 패스워드 기반 키 교환 프로토콜 OTP-EKE(One Time Password based Encrypted Key Exchange)는 EKE(Encrypted Key Exchange)[3] 부류의 확인자 기반 방식의 프로토콜로서, 서버와 사용자간의 키 공유 이후에 사용자가 자신의 패스워드를 올바로 알고 있음을 서버에게 인증하는 증명(proof) 단계에서 원-타임-패스워드 S/Key(One-Time password S/Key)[8] 방식을 적용함으로써 효율성을 향상 시켰다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문에서 사용할 용어들을 정의하고, 패스워드 기반 키 교환 프로토콜들이 만족해야 할 보안 요구사항과 제한 사항을 기술한다. 3장에서는 제안 프로토콜과 비교대상이 되는 기존의 EKE(Encrypted Key Exchange)[3] 부류의 프로토콜들을 소개하고, 4장에서는 본 논문에서 제안한 프로토콜을 설명한다. 5장에서는 제안한 프로토콜의 안전성을 분석하고 기존의 확인자 기반 프로토콜들과 제안한 프로토콜의 효율성을 비교 분석한다. 마지막으로 6장에서는 결론을 맺는다.

2. 개 요

이 절에서는 본 논문에서 사용할 용어들을 정의하고, 패스워드 기반 키 교환 프로토콜(password-based key exchange protocol)들이 만족해야하는 기본적인 보안 요구사항들을 기술한다. 또한 패스워드 기반 키 교환 프로토콜들이 해결할 수 없는 제한 사항을 기술한다.

2.1 용어 정의

- A : 사용자
- B : 서버
- ID : 사용자 식별자(ID, identifier)
- pwd : 사용자의 패스워드(password)
- $H()$: 일방향 해쉬 함수(one-way hash function)
- $E_K()$: 키 K를 사용한 대칭키 암호화 알고리즘 (symmetric key encryption algorithm)
- C_A, C_B : 사용자와 서버의 임의 시도 값 (challenge value)
- g : 곱셈 군(multiplicative group) Z_p^*의 생성자 (generator), p의 원시 근(primitive root)
- p, q : 강한 소수(stong prime), $p=2*q+1$

2.2 보안 요구사항

패스워드 기반의 키 교환 프로토콜들이 고려해야 할 보안 특성과 요구조건은 다음과 같다[1][9][15]. 본 논문에서 제안하는 프로토콜은 이러한 요구조건들을 만족시키도록 설계되었다.

(1) 수동적인 공격자(passive adversary)의 도청 공격에 안전해야한다 :

o 도청 공격(eavesdropping) :

도청공격은 공격자가 온라인 상의 통신 내용을 도청하여 세션키의 정보를 알아내거나, 통신에서 사용되는 유용한 정보를 알아내는 공격이다.

(2) 능동적인 공격자(active adversary)의 재전송 공격과 중간 침입자 공격에 안전해야 한다:

o 재전송 공격(replay attack) :

재전송 공격은 합법적인 사용자가 과거에 통신했던 메시지를 공격자가 저장했다가 이후의 통신에 재전송하는 공격이다.

o 중간 침입자 공격(man-in-the-middle attack) :

중간 침입자 공격은 통신 선로상의 중간에 위치한 공격자가 서버와 사용자 사이에 전송되는 정보들을 불법으로 도청·변경하여 전송함으로써 합법적인 사용자들간의 세션키를 구해내는 공격이다.

(3) 오프라인 패스워드 추측 공격(off-line password guessing attack)에 안전해야 한다 :

오프라인 패스워드 추측 공격은 공격자가 사용자에 의해 자주 선택되는 패스워드들에 대한 사전(dictionary)을 가지고 있다고 할 때 수행되는 공격이다. 공격자가 사용자들간의 통신을 저장한 후, 패스워드 사전으로부터 과거 통신에 사용된 패스워드와 일치하는 값을 비교하여 찾아낸다.

(4) Denning-Sacco 공격에 안전해야 한다 :

Denning-Sacco 공격은 세션키가 노출되었을 때, 공격자가 그 동안 도청한 정보들을 기반으로 사용자의 패스워드에 대한 정보나 앞으로 진행될 세션에서 사용될 세션키에 대한 정보를 얻고자 하는 공격이다.

(5) PFS(Perfect Forward Secrecy)를 만족해야 한다 :

Perfect forward secrecy란 공격자가 사용자의 패스워드나 서버의 장기(long-term) 패스워드 확인자(password verifier)를 알아냈다 할지라도, 이전에 사용되었던 세션키에 관한 정보는 알아낼 수 없는 성질이다.

2.3 제한 사항

패스워드 기반 키 교환 프로토콜은 사용자가 기억할 수 있을 만큼의 작은 비트 길이를 갖는 패스워드를 이용해서 세션키를 공유해야하기 때문에, 그 특성상 해결할 수 없는 다음과 같은 제한 사항들을 가지고 있다[1][9].

(1) 온라인 추측 공격이 항상 가능하다 :

온라인 추측공격은 공격자가 사용자에 의해 자주 선택되는 패스워드들에 대한 사전을 가지고 있을 때 수행되는 공격이다. 이 공격에서 공격자는 사전으로부터 임의로 선택된 패스워드를 사용하여 정당한 사용자로 가장해 본다. 만약 공격에서 실패하면 공격자는 선택한 패스워드를 사전에서 제거하고 다른 패스워드를 선택하여 로그인에 성공할 때까지 이를 반복한다. 이러한 공격의 위협은 서버가 올바른 패스워드 입력 시도의 실패 횟수를 제한함으로써 최소화할 수 있다. 그러나 공격자가 제한된 입력 시도 횟수 내에서 계속하여 온라인 추측공격을 수행하는 것은 막을 수 없다.

(2) 서버가 공격 당하여 서버 내에 저장되어 있는 패스워드 확인자가 공격자에게 노출되면, 확인자로부터 패스워드를 알아내려는 공격자의 사전공격은 막을 수 없다. 이 공격의 비용은 패스워드의 크기와 확인자를 만드는 함수의 계산 비용에 비례한다.

3. 기존 연구

이 장에서는 기존에 발표된 패스워드 기반 키 교환 프로토콜들 중에서 OTP-EKE와 비교 대상이 되는 확인자 기반 EKE(Encrypted Key Exchange)[3] 부류의 프로토콜들을 소개하고 각 방식의 차이점을 기술하겠다.

3.1 A-EKE(Augmented Encrypted Key Exchange)

A-EKE의[2] 경우, 서버는 사용자의 ID와 패스워드 확인자 $H(pwd)$를 저장하고 있고 사용자는 자신의 패스워드 pwd를 기억하고 있으며 그림 1과 같이 수행된다. 키 공유 단계에서 전송되는 각 메시지들은 사용자의 패스워드 확인자 $H(pwd)$로 암호화되고 Diffie-Hellman 키 공유 방식[6]으로 세션키가 설정된다. 사용자의 패스워드 소유를 증명하는 단계 5에서 $F()$함수를 사용하였는데 저자는 이 함수를 사용자의 패스워드로 유도한 개인키(private key)와 공개키(public key)를 사용한 서명함수로 대신할 수 있다고 기술하였다. 그러나 현실적으로 패스워드로부터 개인키·공개키쌍을 만들어내는 것은 어렵고, 이 키 쌍은 서버에게 패스워드에 대한 또 다른 정보를 주는 것이 되며 RSA와[16]같은 서명함수를 사용할 경우에는 패스워드로부터 개인키·공개키 쌍을 유도해 낸다는 것은 거의 불가능하다[2]. 또한 ElGamal[16]서명 알고리즘을 사용할 경우, 서명을 위해 부가적인 지수승(exponential) 연산이 필요하기 때문에 효율성이 떨어진다.

단계	A	메시지	B
1	choose $a \in_R [1, q-1]$	ID , $E_{H(pwd)}(g^a)$ →	choose $b \in_R [1, q-1]$
2	$K = g^{ab}$	$E_{H(pwd)}(g^b)$ $E_K(C_B)$ ←	$K = g^{ab}$
3		$E_K(C_A, C_B)$ →	
4		$E_K(C_A)$ ←	
5		$E_K[F(pwd, K)]$ →	$T(H(pwd), F(pwd, K), K)$

그림 1 A-EKE 프로토콜

3.2 B-EKE("B" extension Encrypted Key Exchange)

B-EKE의[10] 경우, 서버는 사용자의 ID와 패스워드 확인자 $H(pwd)$, g^{pwd}를 저장하고 있고 사용자는 자신의 패스워드 pwd를 기억하고 있으며 그림 2와 같이 수행된다. 키 공유 단계에서 전송되는 각 메시지들은 사용자의 패스워드 확인자 $H(pwd)$로 암호화되고 Diffie-Hellman 키 공유 방식으로[6] 세션키가 설정된다. 사용자의 패스워드 소유를 증명하는 단계 4에서 사용자는 서버로부터 받은 g^x에 패스워드 pwd를 지수승한 g^{x*pwd}과 세션키 $K_1=g^{ab}$을 $P()$함수에 적용하여 서버에게 보낸다. 서버는 사용자의 확인자 g^{pwd}에 자신이 선택한 난수 x를 지수승하여 계산한 후, 세션키와 함께 $P()$함수에 적용하여 사용자로부터 받은 값과 같은지 확인한다. 이 값이 같으면 서버는 사용자를 인증하게 된다.

단계	A	메시지	B
1	choose $a \in_R [1, q-1]$	$ID, E_{H(pwd)}(g^a)$ →	choose $b \in_R [1, q-1]$
2	$K_1 = g^{ab}$	$E_{H(pwd)}(g^b)$ ←	$K_1 = g^{ab}$
3		$P(K_1), g^x$ ←	
4	$K_2 = g^{x*pwd}$	$P(K_1, K_2)$ →	$K_2 = g^{x*pwd}$

그림 2 B-EKE 프로토콜

3.3 AuthA

AuthA의[5] 경우, 서버는 사용자의 ID와 패스워드 확인자 $g^{H(ID||B||pwd)}$를 저장하고 있고, 사용자는 자신의 패스워드 pwd를 기억하고 있으며 그림 3과 같이 수행된다. 그림 3의 단계 1, 2에서 보는 바와 같이 키 공유 단계에서 전송되는 메시지들은 사용자의 패스워드 확인자$g^{H(ID||B||pwd)}$로 암호화되고, Diffie-Hellman 키 공유 방식으로[6] 세션키가 설정된다. 사용자의 패스워드 소유를 증명하는 단계 3에서, 사용자가 패스워드와 식별자를 일방향 해쉬 함수에 적용시킨 값 $H(ID||B||pwd)$을 세션키 설정시 서버로부터 받은 난수 g^b에 지수승하여 $g^{b*H(ID||B||pwd)}$을 서버에게 보낸다. 서버는 사용자의 확인자 $g^{H(ID||B||pwd)}$에 자신이 선택한 난수 b를 지수승하여 계산한 후, 사용자로부터 받은 값과 같은지 확인한다. 이 값이 같으면 서버는 사용자를 인증하게 된다.

단계	A	메시지	B
1	choose $a \in_R [1, q-1]$	$ID, E_{g^{H(ID\|\|B\|\|pwd)}}(g^a)$ →	choose $b \in_R [1, q-1]$
2	$K = H(Mkey\|\|0)$	$E_{g^{H(ID\|\|B\|\|pwd)}}(g^b)$ ←	$K = H(Mkey\|\|0)$
3		$H(Mkey\|\| g^{b*H(ID\|\|B\|\|pwd)})$ →	

$Mkey = H(ID||B||g^a||g^b||g^{ab})$

그림 3 AuthA 프로토콜

4. OTP-EKE(One-Time-Password based Encrypted Key Exchange)

기존 프로토콜들은 2.2에서 기술한 보안 요구사항들을 만족하지만, 인증 방식에서 각각 다음과 같은 비효율성을 가진다. A-EKE는 사용자의 패스워드로부터 유도된 공개키와 비밀키를 이용하여 서명을 하는 방식을 취하고, B-EKE는 사용자가 서버로부터 받은 임의의 난수 g^x에 패스워드 pwd를 지수승한 g^{x*pwd}을 서버에게 보냄으로써 인증한다. AuthA는 사용자가 패스워드 pwd와 사용자 식별자 값을 해쉬 함수에 적용시킨 결과 $H(ID||B||pwd)$을 서버로부터 받은 임의의 난수 g^b에 지수승하여 $g^{b*H(ID||B||pwd)}$을 보냄으로써 인증한다. 이러한 인증 방법을 사용함으로써 수반되는 부수적인 전송횟수나, 지수승 계산 비용에 대하여 기존의 프로토콜들은 효율성 개선의 여지가 있다.

본 논문에서는 보안 요구사항들을 만족하면서도 원-타임-패스워드를 사용하여 인증 단계의 효율성을 증대시킨 OTP-EKE 프로토콜을 제안한다.

4.1 사용자 ID와 패스워드를 설정하는 초기 단계

사용자 ID와 패스워드를 설정하는 초기 단계는 사용자가 서버로부터 서비스를 받기 위하여 서버에 ID와 패스워드를 등록하는 단계로서, 이것에 대한 상세한 기술은 본 논문의 범위를 벗어나므로 간략히 기술하겠다. 사용자는 ID와 패스워드 확인자에 해당하는 $H^n(pwd)$를 안전한 채널(secure channel)을 통해 서버에게 전송한다. 서버는 사용자에게 받은 ID와 패스워드 확인자를 패스워드 디렉토리 내에 저장하고 서버를 인증할 수 있는 공개 패스워드(public password)[9] $H(g^s)$을 안전한 채널을 통해 사용자에게 전송한다. 여기서 패스워드 확인자로 사용하는 $H^n(pwd)$는 사용자 패스워드 pwd에 일방향 해쉬함수를 n번 적용하여 얻은 값이다. 제안한

프로토콜에서 확인자는 원-타임-패스워드[8] 방식처럼 패스워드를 설정한 후 i번째 통신에서는 $H^{n-i+1}(pwd)$를 패스워드의 확인자로 사용하며, 설정한 패스워드 확인자를 $n-1$번 사용한 후에는 새로운 패스워드를 설정한다.

또한 서버가 초기 단계에서 사용자에게 보내주는 공개 패스워드 $H(g^s)$는 사용자에게 제공하는 서버의 공개키의 해쉬된 값으로서 사용자가 기억하거나 편리하게 가지고 다닐 수 있을 만큼 충분히 길이가 짧은 값(60~80 bit(6~8문자정도))이다[9]. 이 공개 패스워드는 기밀성(security)은 필요치 않으나 무결성(integrity)은 보장되어야 하며 OTP-EKE에서 이를 보장한다.

4.2 인증 및 세션키 설정 단계

본 논문에서 제안하는 프로토콜 OTP-EKE에서 서버와의 패스워드 설정 후 i번째 통신에 대한 단계별 수행 과정은 다음과 같으며 그림 4에 요약되어 있다.

(1) 사용자 A는 랜덤하게 $a \in_R [1, q-1]$를 선택해서 g^a을 계산하고 패스워드 확인자 $H^{n-i+1}(pwd)$로 암호화하여 ID와 함께 서버 B에게 전송한다.

(2) B는 $b \in_R [1, q-1]$를 선택해서 g^b을 계산하고, A로부터 받은 값과 자신의 랜덤 수들을 이용하여 $K = g^{ab}$, $K^* = g^{as}$을 계산한다. 또한 키 확인(key confirmation)을 위한 메시지 $H(K||K^*)$을 생성하여, 서버의 장기 공개키(long-term public key) g^s와 $H^{n-i+1}(pwd)$로 암호화한 g^b을 함께 A에게 전송한다. 사용자의 사용 환경에 따라 g^s은 저장될 수도 있는 값이고, 저장할 경우 서버가 매번 프로토콜 실행할 때마다 전송하지 않아도 된다.

(3) A는 B로부터 전송 받은 값을 $H^{n-i+1}(pwd)$를 이용하여 복호화하고, g^s에 일방향 해쉬함수(one-way hash function)를 적용하여 자신이 가지고 있는 공개 패스워드 $H(g^s)$값과 비교해서 같은지를 확인한다. 만약 같지 않으면 프로토콜 세션을 종료한다. 여기서, B가 서버의 장기 비밀키 s를 알고 있음은 B가 K^*를 사용하여 만든 키 확인 메시지를 통해 검증된다. A는 $a \in_R [1, q-1]$를 선택해서 g^a와 $K = g^{ab}$, $K^* = g^{as}$을 계산하고 $H(K||K^*)$값을 확인한다. A가 $H(K||K^*)$값이 자신이 계산한 것과 같음을 확인하고 나면, 다음 세션을 위한 패스워드 확인자 $H^{n-i}(pwd)$와 세션키 확인을 위한 메시지 $H(K)$를 K^*로 암호화해서 B에게 전송한다. 여기서 $K = g^{ab}$는 세션키이고, $K^* = g^{as}$은 사용자의 다음 번 패스워드 확인자를 서버에게 안전하게 전송하기 위해서 사용되는 암호화 키이다. K^*는 사용자와 서버만이 만들 수 있는 값이기 때문에 공격자는 이를 만들지 못하므로 K^*를 다음 번 사용자의 패스워드 확인자를 암호화하여 보내는데 사용된다.

(4) B는 A로부터 전송 받은 $H(K)$를 확인하여 A와 세션키 K를 공유했음을 확인한 후, 전송 받은 확인자 값에 일방향 해쉬함수를 적용하여 $H(H^{n-i}(pwd)) = H^{n-i+1}(pwd)$인지를 확인한다. 등식이 성립하면, 서버는 사용자를 인증하고 A의 패스워드 확인자를 $H^{n-i}(pwd)$로 교체해서 저장한다.

단계	A	메시지	B
1	choose $a \in_R [1, q-1]$	ID, $E_{H^{n-i+1}(pwd)}(g^a)$ →	choose $b \in_R [1, q-1]$
2	verify $H(g^s)$ $K = g^{ab}$ $K^* = g^{as}$	$E_{H^{n-i+1}(pwd)}(g^b), g^s$ $H(K\|K^*)$ ←	$K = g^{ab}$ $K^* = g^{as}$
3		$E_{K^*}(H^{n-i}(pwd),$ $H(K))$ →	verify $H(H^{n-i}(pwd)) = H^{n-i+1}(pwd)$

그림 4 OTP-EKE 프로토콜

5. 안전성 및 효율성 분석

이 절에서는 2.2에서 살펴본 기본 보안 요구 조건에 따라 OTP-EKE의 안전성을 분석하고, 효율성은 EKE[3] 부류의 프로토콜과 기존의 확인자 기반 프로토콜들과 비교 분석한다.

5.1 안전성 분석

(1) 도청 공격 :

제안한 프로토콜 상에서 전송되는 메시지들이 추측 불가능한 임의의 난수들이면서 모두 암호화되어 전송되기 때문에, 단순한 도청만으로는 유용한 정보를 얻을 수가 없다.

(2) 재전송 공격 :

프로토콜 단계별로 같은 메시지가 연속적으로 전송되지 않으므로, 한 세션 내에서 같은 메시지를 전송하는 공격을 수행할 수 없다. 또한 매 세션마다 새로 생성되는 랜덤 수와 패스워드 확인자를 사용하기 때문에, 공격자가 이전 세션에서 전송된 메시지를 가지고 있어도 다음 세션에서 그 메시지를 사용할 수 없으므로 재전송 공격을 수행 할 수 없다.

(3) 중간 침입자 공격 :

OTP-EKE 프로토콜을 통하여 사용자와 서버간에 생성되는 키는 K와 K^*이다. K와 K^*에 대하여 중간 침입자 공격을 하려면, 공격자가 g^a, g^b, g^s을 얻을 수 있

어야 하고 이들을 각각 $g^{a'}$, $g^{b'}$, $b^{s'}$으로 대치하여 전송할 수 있어야 한다[16]. 그러나 OTP-EKE 프로토콜에서 g^a, g^b은 패스워드 확인자로 암호화하여 전송되기 때문에 알아낼 수 없다. 만약 공격자가 패스워드 확인자를 알고 있다면 g^a, g^b은 알아내어 $g^{a'}$, $g^{b'}$로 대치할 수 있다. 그러나 사용자가 $H(g^s)$값을 알고 있으므로 g^s은 $g^{s'}$로 대치할 수 없으며 또한 Diffie-Hellman 문제의 어려움에[16] 근거하여 g^a, g^s로부터 $K^*=g^{as}$을 알아낼 수 없다. 따라서 OTP-EKE는 중간 침입자 공격에 대해 안전하다.

(4) 오프라인 패스워드 추측 공격 :

공격자가 과거의 통신기록과 패스워드 사전을 가지고서 사전 공격을 수행한다고 해도, 패스워드 확인자로 암호화된 메시지는 임의의 난수들로서 추정 가능문(verifiable-text)이나 기지 평문(known-plaintext)이 아니기 때문에 사전 공격을 수행해서 올바른 패스워드를 알아낸다는 것은 불가능하다[7]. 따라서 OTP-EKE는 오프라인 추측 공격에 대하여 안전하다.

(5) Dennig-Sacco 공격 :

제안한 프로토콜에서 세션키 K는 사용자의 패스워드에 대한 정보를 포함하고 있지 않기 때문에 세션키가 누출된다고 할지라도 사용자의 패스워드나 패스워드 정보를 알 수 없다. 또한 매번 랜덤하게 생성되는 난수들을 이용하여 세션키를 만들기 때문에 세션키가 누출되었다고 할지라도 그 이후의 세션키에 관한 정보는 알 수 없다.

(6) perfect forward secrecy:

제안한 프로토콜에서 세션키의 안전성은 이산대수 문제의 어려움과 Diffie-Hellman 문제의 어려움에[16] 기반하고 있으며 Diffie-Hellman 문제의 어려움에 근거하여 장기간 키(long-term key)로 사용되는 사용자의 패스워드가 노출되어도 과거에 사용되었던 세션키의 값들은 알 수 없다.

그 밖에, EKE[3] 부류가 취약할 수 있는 분할공격(partition attack)을[3] 방지하기 위해서 제안한 프로토콜 OTP-EKE는 키 교환 시에 사용되는 곱셈 군 Z_p^*의 생성자 g를 p의 원시 근으로 하였다. 생성자 g를 원시 근으로 한정하면, 암호화된 지수 계산값(encrypted exponentials)인 $E_{H^{n-i}(pwd)}(g^a)$, $E_{H^{n-i}(pwd)}(g^b)$이 Z_p^* 상에서 랜덤하게 분포(random distribution)함을 확신할 수 있으므로 분할공격을 막을 수 있다.

또한 서버 내의 패스워드 디렉토리가 노출되었을 때, 제안한 프로토콜 OTP-EKE의 경우에는 공격자가 서버의 비밀키 s를 알지 못하기 때문에 K^*를 만들어내지 못한다. 따라서 공격자는 그림 4의 단계 2를 수행하지 못하므로 서버를 가장하는 공격을 수행할 수 없고, 사전 공격을 수행하지 않는 한 사용자를 가장하는 공격도 성공할 수 없다.

따라서 OTP-EKE는 패스워드 디렉토리 노출(password directory compromising)에 대하여 안전하다. 만약, 서버 내의 패스워드 디렉토리와 서버의 비밀키 s가 노출되었을 경우라면 OTP-EKE는 공격자가 서버를 가장하여 공격할 수 있다. 또한 이 점은 현존하는 모든 패스워드 기반 키 교환 프로토콜의 문제이기도 하다. 그러나 OTP-EKE에서 서버의 모든 정보를 알고 있는 공격자라도 사용자를 계속적으로 가장하는 것은 거의 불가능하다.

만약 공격자가 $H^{n-i+1}(pwd)$와 s를 알고 있다고 가정하면, i번째 세션에서 서버를 가장하여 사용자로부터 $H^{n-i}(pwd)$를 알아낼 수 있다. 그러나 이를 이용하여 사용자를 가장하고 서버와 $i+1$번째 세션키를 설정하려면 마지막 단계에서 $H^{n-i-1}(pwd)$를 알고 있어야 한다. 즉 공격자가 서버의 패스워드 디렉토리와 s를 알아낸 후 서버와 사용자가 인식하지 못하게 계속적으로 사용자를 가장하려면, 이후의 사용자와 서버간의 모든 통신을 가로채어 저장해야만 한다. 즉, 사용자를 가장하여 j번째 ($j>i$) 세션에서 서버와 통신하기 위해서는 반드시 서버를 가장해서 사용자와 통신하여 $H^{n-l-1}(pwd)$, ($i<l \le k$, $k \ge j$)를 저장하고 있어야 한다. 따라서 패스워드 디렉토리와 서버의 비밀키를 가진 공격자가 사전공격을 수행하지 않고서, 사용자를 매번 가장하기는 매우 어렵다.

5.2 효율성 비교 분석

이 장에서는 제안한 프로토콜 OTP-EKE의 효율성 측면을 고찰하고, 기존의 확인자 기반 프로토콜과 비교 분석한다. 통신 오버헤드(communication overhead) 측면에서 보면 OTP-EKE는 온라인 상의 메시지 전송 회수(rounds)가 3회로 기존 프로토콜들에 비하여 통신 횟수가 가장 적고 간단하다.

또한 실행 시간(execution time)의 대부분을 차지하는 모듈러 지수승(modular exponent)에 대해서, A,B가 병렬적으로 수행하는 계산 횟수를 비교해 보면, OTP-EKE는 3회로서 기존 프로토콜들에 비하여 가장 효율적이며, 그림 4 단계 1에서 수행하는 g^a, g^b 계산을 온라인 통신 전에 미리 계산해(pre-computation) 놓으면, 병렬 지수승 계산(parallel exponentiations)을 2회만 수행하면 되므로 온라인의 실행시간을 줄일 수 있

표 1 확인자 기반방식 프로토콜의 비교분석

	Rounds	Exponentiations			Random numbers	
		User	Server	Parallel	User	Server
A-EKE	5	4	4	6	1	1
B-SPEKE	4	3	4	6	1	2
SRP	4	3	3	4	1	1
B-EKE	4	3	4	4	1	2
GXY	4	4	3	5	1	1
SNAPI-X	5	5	4	7	2	3
AuthA	3	4	3	6	1	1
PAK-X	3	4	4	8	1	2
AMP	4	2	4	5	1	1
OTP-EKE	3	3	3	3	1	1

표 2 EKE부류의 확인자 기반방식 프로토콜의 비교

	Rounds	Encryptions	Exponentiations			Random numbers	
			User	Server	Parallel	User	Server
A-EKE	5	6	4	4	6	1	1
B-EKE	4	2	3	4	4	1	2
AuthA	3	2	4	3	6	1	1
OTP-EKE	3	3	3	3	3	1	1

다.

난수 생성에서도 세션키를 만들기 위해서 서버와 사용자간에 각각 한번씩 난수를 생성하는 것 이외에 부가적인 난수생성이 필요 없다. 표 1은 확인자 기반의 프로토콜들[2][4][5][10][12][13][14][18]과 OTP-EKE를 라운드 수와 지수승 계산 횟수, 난수 생성 횟수 등으로 나누어 분석하고 효율성을 비교한 결과이다. A-EKE의 [2] 서명함수는 ElGamal로[16]가정하였다.

표 2는 OTP-EKE가 속한 EKE[3] 부류의 확인자 기반방식 프로토콜을 비교 분석한 결과이다. A-EKE의 [2] 경우는 사용자 패스워드 증명 단계에서 ElGamal [16] 서명함수를 이용하였고, 다른 것들에 비해 라운드 수나, 암호 알고리즘 사용 횟수 등이 많아 비효율적이라 할 수 있다. B-EKE와[10] AuthA의[5] 경우는 그에 비해서 라운드(rounds)수를 줄였지만, 패스워드 증명 단계에서 패스워드를 지수로 사용하여 계산함으로써 부가적인 지수승 계산이 수반되었다. 제안한 프로토콜 OTP-EKE의 경우는 라운드 수와 지수승 계산 횟수, 난수 생성 횟수 등이 EKE[3] 부류 중에서도 효율적이다.

6. 결 론

본 논문에서는 원-타임 패스워드 방식을 적용한 확인자 방식의 새로운 패스워드 기반 키 교환 프로토콜 OTP-EKE를 제안하였다.

제안한 프로토콜은 서버와 사용자간의 세션키 공유시에 패스워드 확인자로 암호화하여 보내는 EKE[3] 부류의 확인자 기반 프로토콜로서 패스워드 기반 키 교환 프로토콜이 가져야 하는 필수적인 보안 요구 사항에 맞추어 설계되었다.

특히 OTP-EKE는 사용자 패스워드 증명 단계에서 기존의 EKE 부류의 프로토콜들과 달리 원-타임-패스워드 방식을 적용함으로써 기존의 방식들보다 모듈러 지수승 계산을 적게 하여 효율성을 높였으며, 또한 메시지 전송회수를 3회로 감소시킴으로써 프로토콜 수행이 효율적으로 이루어지게 하였다.

본 논문에서 제안한 OTP-EKE는 안전도를 저하시키지 않으면서, 기존 확인자 기반 방식의 프로토콜들보다 개선된 효율성을 가진다. 따라서 본 프로토콜은 사용자

인증이나 세션키 공유 및 확인을 요하는 시스템에 유용하게 적용될 수 있을 것으로 기대된다.

참 고 문 헌

[1] M. Bellare, D. Jablon, H. Krawczyk, P. MacKenzie, P. Rogaway, R. Swaminathan and T. Wu," Proposal for P1363 study group on password-based authenticated -key-exchange methods," 2000.

[2] S. Bellovin and M. Merritt, "Augmented encrypted key exchange: a password-based protocol secure against dictionary attacks and password-file compromise," ACM Conference on Computer and Communications Security, 1993.

[3] S. Bellovin and M. Merrit, "Encrypted key exchange : password-based protocols secure against dictionary attacks," IEEE Symposium on Research in Security and Privacy, 1992.

[4] V. Boyko, P. MacKenzie and S. Patel, "Provably secure password authenticated key exchange using Diffie-Hellman," Eurocrypt 00, 2000.

[5] M. Bellare and P. Rogaway, "The AuthA protocol for password-based authenticated key exchange," 2000, available from http://www.cs.ucdavis.edu/rogaway/papers/autha.ps.

[6] W. Diffie and M. Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, Vol. 22, No. 6, 1976.

[7] L. Gong, T. M. A. Lomas, R. M. Needham and J. H. Saltzer, "Protecting poorly chosen from guessing attacks," IEEE Journal on Selected Areas in Communications, Vol. 11, No. 5, 1993.

[8] N. Haller, "The S/KEY one-time password system," RFC 1760, 1995.

[9] S. Halevi and H. Krawczyk, "Public-key cryptography and password protocols," ACM Transactions on Information and System Security (TISSEC), Vol. 2, 1999.

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

[11] D.Jablon, "Strong password-only authenticated key exchange," ACM Computer Communications Review, Vol. 26, No. 5, 1996.

[12] T. Kwon, "Authentication and key agreement via memorable password," NDSS 2001 Symposium Conference Proceedings, 2001.

[13] T. Kwon, J. Song, "Secure agreement scheme for g^{xy} via password authentication," Electronics Letters, Vol. 35, No. 11, 1999.

[14] P. MacKenzie, S. Patel and R. Swaminathan, "Password-authenticated key exchange based on RSA," ASIACRYPT, 2000.

[15] 박왕석, 정종필, 박창섭, 이동훈, "패스워드를 이용한 인증 프로토콜들에 대한 고찰," 통신정보보호학회 학술지 제9권 제4호, 1999.

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

[17] M. Steiner, G. Tsudik, M. Waidner, "Refinement and extension of encrypted key exchange," ACM Operating Systems Review, Vol. 29, No. 3, 1995.

[18] T. Wu, "Secure remote password protocol," NDSS, 1998.

서 승 현

2000년 2월 이화여자대학교 수학과 학사. 2002년 2월 이화여자대학교 과학기술 대학원 컴퓨터학과 석사. 2002년 3월 ~ 현재 이화여자대학교 과학기술대학원 컴퓨터학과 박사과정. 관심분야는 정보보호, 암호프로토콜, 암호알고리즘

조 태 남

1986년 2월 이화여자대학교 전자계산학과 학사. 1988년 2월 이화여자대학교 대학원 전자계산학과 석사. 1988년 3월 ~ 1996년 3월 한국전자통신연구원 선임연구원. 1998년 3월 ~ 현재 이화여자대학교 과학기술대학원 컴퓨터학과 박사과정. 관심분야는 정보보호, 암호프로토콜, 알고리즘 설계

이 상 호

1979년 2월 서울대학교 계산통계학과 학사. 1981년 2월 한국과학기술원 전산학과 석사. 1987년 8월 한국과학기술원 전산학과 박사. 1983년 9월 ~ 현재 이화여자대학교 컴퓨터학과 교수. 2000년 ~ 현재 한국정보과학회 총무이사, 정보보호연구회 부위원장. 관심분야는 정보보호, 암호프로토콜, 알고리즘 설계, 계산기하, 그래프 드로잉, 데이터 마이닝, Bioinformatics