로밍 유저를 위한 패스워드-기반 인증 프로토콜에 관한 연구

이송원, 김광조

한국정보통신대학원대학교, 국제정보보호기술연구소

A Study on the Password-based Authentication Protocol for the Roaming User

Songwon Lee and Kwangjo Kim

International Research center for Information Security(IRIS),

Information and Communications Univ.(ICU), Korea

요 약

네트워크에 대한 접근 용이성이 높아짐에 따라 사용자는 자신의 클라이언트 터미널 이외의 터미널로부터 네트워크에 접속하고자 하며, 이러한 로밍 유저(Roaming User)는 패스워드-기반의 인증 프로토콜을 통하여 원격의 서버로부터 비밀키와 같은 자신의 비밀정보를 다운로드 할 수 있다. 본 논문에서는 그 동안 제시되어온 로밍 프로토콜들[7,8,9]을 간략하게 살펴봄으로써, 프로토콜의 구성 방법 그리고 사전 공격 등의 패스워드-기반 프로토콜에 대한 일반적인 공격으로부터 안전하기 위하여 어떠한 암호학적 기법들을 사용하였는지 이해하고자 한다. 그리고, 이러한 프로토콜들의 암호학적 특징들에 대한 차이점을 비교 분석한다.

I. 서론

Bellovin과 Merritt[6]은 사전 공격(Dictionary Attack)에 안전한 패스워드-기반의 EKE 프로토콜을 제안하였다. 본 프로토콜에서는 안전하지 못한 네트워크 상에서 두 당사자가 비밀 정보를 안전하게 교환하기 위하여 사전에 서로 공유된 패스워드를 이용한다. 유사한 목적의 다른 많은 일련의 프로토콜들이 제시되었는데, 예를 들면 A-EKE[2], SPEKE[3], B-SPEKE[4], 그리고 PAK[5] 등이 있다.

오늘날, 네트워크에 대한 접근 용이성이 높아짐에 따라 사용자는 자신의 클라이언트 터미널 이외의 터미널 또는 사용자 그룹이 공동으로 사용하는 터미널로부터 네트워크에 접속하게 된다. 이러한 사용자는 패스워드-기반의 인증 프로토콜을 통하여 원격의 서버로부터 비밀키와 같은 자신의 비밀정보를 다운로드 하여, 자신이 원하는 서비스 서버에 접속하기 위한 세션 키로 그 정보를 이용하고자 할 것이다[7,8,9]. Ford와 Kaliski[8]는 이러한 사용자를 로밍 유저(roaming user)라고 명명하였는데, 다른 클라이언트 터미널에서 네트워크에 접근하는 사용자를 말한다. 또한, 로밍 프로토콜(roaming protocol)[9]은 하나 또는 다중의 신용 서버(credential server)로부터 비밀키를 원격에서 검색하여 오기 위한 안전한 패스워드-기반의 프로토콜을 말한다. 이것은, 단지 기억하기 쉬운 패스워드만을 사용하여, 그리고 사용자(로밍 유저)의 정보를 터미널에 미리 저장하지 않고, 사용자가 신용 서버를 인증한 후, 해당 클라이언트 터미널에서 일시적으로 사용하기 위한 비밀키를 서버에서 다운로드 한다.

본 논문에서는 그 동안 제시되어온 로밍 프로토콜들[7,8,9]을 간략하게 살펴봄으로써, 프로토콜의 구성 방법 그리고 사전 공격 등의 패스워드-기반 프로토콜에 대한 일반적인 공격으로부터 안전하기 위하여 어떠한 암호학적 기법들을 사용하였는지 이해하고자 한다. 그리고, 이러한 프로토콜들의 암호학적 특징들에 대한 차이점을 비교 분석한다.

II. 기존 프로토콜

1. PK99[7]

EKE[6]와 SPEKE[3]에 기반 한 여러 개의 로밍 프로토콜들을 제시하고 있다. 패스워드만을 사용한 단순한 형태의 프로토콜일지라도 안전한 로밍 프로토콜로서 충분하다는 것을 보여주고 있다. 여기에서는 EKE-기반의 2메시지 프로토콜을 살펴본다.

1) 등록

사용자는 ID, 패스워드의 해쉬 값(W=h(pwd)), 그리고 패스워드로 암호화된 사용자의 비밀정보(Y)를 서버에 등록한다.

2) 비밀정보 조회

사용자는 그림1의 프로토콜을 통하여 서버로부터 비밀정보를 다운로드 한다.

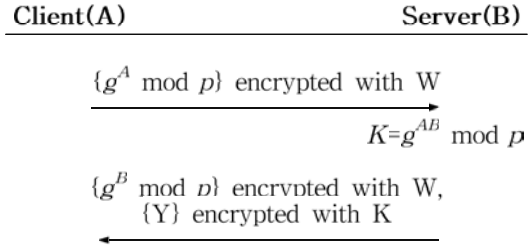

그림 1: [PK99] EKE-기반 2메시지 프로토콜.

프로토콜에서 사용자가 서버를 인증하는 것이 생략되었다. 이것은 본 프로토콜의 목적이 사용자의 비밀키를 단순히 다운로드 하는데 있기 때문이며, 누군가가 서버를 사칭한다할지라도 패스워드를 추측할 수는 없다.

2. FK00[8]

새로운 기법으로서 다중의 서버를 이용한 패스워드 경화(hardening) 프로토콜을 소개하고 있다. 다중의 서버들은 사용자의 패스워드로부터 강력한 비밀 정보를 생성하기 위하여 사용자와 프로토콜을 수행하지만, 서버는 사용자의 패스워드 및 경화된(hardened) 결과를 알 수는 없다.

1) 등록

사용자는 ID와 임의의 난수 값 y_i를 서버 B_i에 전송하고, $S_i = f(pwd)^{y_i}$를 구한다.

사용자는 KDF(Key Derivation Function)를 통하여 비밀정보인 $K_i = KDF(S_1, \ldots, S_n, i)$를 구한다. 사용자는 K_i를 이용하여 저장하고자 하는 비밀정보를 암호화하거나, 서버 B_i를 인증하기 위한 비밀키로 이를 사용할 수 있다. 각 S_i가 강력한 비밀 값인 것처럼 K_i도 그러함을 알 수 있다. 이것은 사용자의 패스워드인 pwd와 모든 공유키인 S_i를 통하여서 K_i를 도출할 수 있기 때문으로, 공격자가 n개의 공유키를 얻지 못한다면 pwd를 추측하여 이를 검증하는 것은 불가능하다.

2) 비밀정보 조회

사용자는 그림2의 프로토콜을 통하여 $\{S_1,...,S_n\}$을 구하고, 이로부터 비밀정보인 K_i를 얻는다.

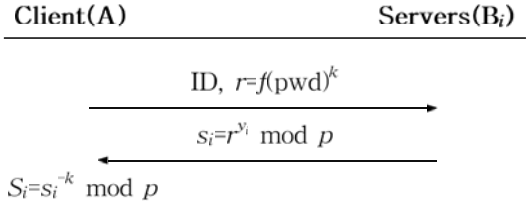

그림 2: [FK00] 패스워드 경화 프로토콜.

여기에서 임의의 난수 값 $k \in_R [1,q-1]$는 은닉 인자(blinding factor)로서, r로부터 패스워드를 도출할 수 없도록 한다. 또한, s_i는 공유키에 대한 은닉된 형태를 이룬다.

3. Jab01[9]

다중 서버 로밍 프로토콜로서 FK00[8]에 기초하여, 좀 더 개선된 모델을 제시하고 있다. 본 프로토콜은 안전하지 못한 네트워크 상에서의 인증을 지원하며, 인증 단계에서 정당한 사용자의 인증을 위하여 서명된 메시지를 사용한다. 오류 해제 프로토콜(Forgiveness protocol)을 제시한다. 이것은, 정당한 사용자로서 인증이 성공하였을 경우 이전에 실수에 의하여 인증에 실패한 것에 대한 징표를 송신토록 하여 서버에 있는 오류 기록을 삭제토록 한다. 다만, 여기에서는 이에 대한 설명을 생략한다.

1) 등록

사용자는 패스워드 P에 대하여 $g_P = h(P)^{2r}$을 구하고, 전자서명을 위한 개인키 U와 이에 대응하는 공개키 V를 생성한다. 임의의 난수 값 y_i로부터 n개의 키 공유 $S_i = g_P^{y_i}$를 생성하여, j-비트 대칭키 $K_m = h(S_1 \| \ldots \| S_n)$을 구하고, 암호화된 개인키

$U_K=_{Km}\{U\}$와 검증자 $proof_{PKm}=h(K_m \| g)$를 계산한다. 사용자는 {ID, y_i, V, U_K, $proof_{PKm}$}를 서버 B_i에 전송한다.

2) 비밀정보 조회

사용자는 인증된 비밀정보 조회를 위하여 그림 3의 프로토콜을 수행한다. 여기에서는 오류 해제 프로토콜을 위한 부분을 생략한다.

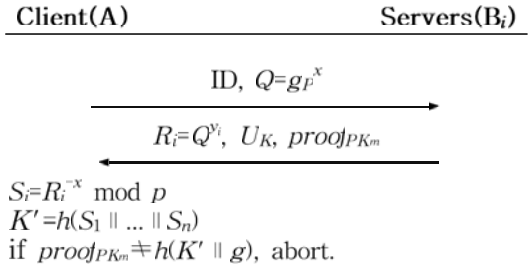

그림 3: [Jab01] 인증된 조회 프로토콜.

사용자는 임의의 난수 x에 의하여 은닉된 형태의 패스워드를 각 서버에게 전송한다. 수신된 정보로부터 마스터키 K_m을 구하고, $proof_{PKm}$와 패스워드를 사용하여 마스터키를 검증한다. 검증이 올바르지 않다면, 이것은 각 서버의 키 공유 중에서 적어도 하나는 올바르지 않음을 말한다.

III. 비교 분석

앞서 살펴본 3개의 프로토콜을 계산의 복잡성 측면에서 비교하면 표 1과 같다. 표 1은 등록단계를 제외한 프로토콜 실행단계만을 고려하였으며, C는 클라이언트를 S는 서버를 의미한다.

표 1: Comparison of efficiency.

		PK99	FK00	Jab01
# of servers		1	n	n
# of communications	C	2	$2n$	$2n$
	S	2	2	2
# of encryption /decryption	C	4	-	-
	S	3	-	-
# of hash functions	C	-	2	2
	S	-	-	-
# of exponents	C	2	$1+n$	$1+n$
	S	2	1	1
# of random numbers	C	1	1	1
	S	1	-	-

FK00과 Jab01은 다중 서버를 사용함에 따라, 클라이언트가 n개의 서버와 교신을 하여야 함으로써 PK99에 비하여 많은 연산을 필요로 한다. 그러나, 서버의 개수에 따른 암호학적 장단점이 상이하므로 단순히 연산의 복잡성 측면에서만 절대 비교할 수는 없을 것이다. Jab01의 경우, 다른 프로토콜과의 비교를 위하여, 오류해제(forgiveness) 프로토콜에 대한 연산은 제외하였다.

FK00에서 클라이언트가 2개의 해쉬 함수를 사용하는 것은 프로토콜에서 제시하고 있는 KDF (Key Derivation Function)와 MGF(Mask Generation Function)로서 각각 해쉬 함수를 이용하는 것으로 가정한 것이다[8,9].

다음은 각 프로토콜에 대한 암호학적 안전성에 대하여 살펴본다.

- 암호학적 Hard problem

PK99는 Diffie-Hellman 키 교환 프로토콜을 기반으로 하고 있다. FK00와 Jab01은 이산대수문제를 기반으로 하고 있으나, 타원곡선과 RSA를 이용할 수도 있음을 보이고 있다.

- 공격(attack)으로부터의 안전성

재전송(replay) 공격, 중간 침입자(man-in-the-middle) 공격, 그리고 사전(dictionary) 공격 또는 패스워드 추측(password guessing) 공격에 대하여 모두 각각 안전함을 보이고 있다. 공격자는 이러한 공격을 통하여 사용자의 패스워드와 비밀정보를 얻을 수 없다.

- 완전한 전방향(perfect forward) 안전성

공격자가 참여자의 패스워드를 알아내었다 할지라도 세션 키와 같이 이전에 사용된 비밀키에 관한 정보를 알 수 없어야 한다. 각각의 프로토콜들은 Diffie-Hellman과 이산대수문제에 기반하고 있으며, 이것은 완전한 전방향 안전성을 제공한다. 패스워드를 알아낸 공격자가 이전의 비밀키를 얻기 위해서는 이러한 암호학적 문제를 풀 수 있어야 하기 때문이다.

IV. 결론

어느 곳에서든지 네트워크에 쉽게 접근할 수 있게 됨에 따라 사용자는 자신만의 클라이언트 단말기 이외의 곳에서 네트워크에 접속하게 되는 경우가 빈번하게 되었다. 그러나, 이러한 편리함은 새로운 보안위협에 직면하게 되는데, 단말을 사실상 공유하게 됨으로써 자신의 비밀정보가 다른 사람에게 노출될 수 있다는 것이다. 따라서, 사용자의 비밀정보를 클라이언트에 저장하지 않고 안전하게

네트워크에 접근할 수 있는 방법이 필요하다.

패스워드-기반의 로밍 프로토콜은 이러한 필요성을 충족시킬 수 있는 방법중의 하나로 제안되고 있다. 이것은 단지 쉽게 기억할 수 있는 패스워드만을 사용하여 서버로부터 개인키와 같은 비밀정보를 다운로드 하여 일시적으로 해당 터미널에서 사용하기 위한 것이다.

PK99는 EKE와 SPEKE에 기반한 패스워드-기반의 프로토콜로써, 비밀키를 서버에서 안전하게 다운로드 하기 위한 것이다. 그러나, 이들 프로토콜은 상호 인증을 제공하지 않으며 무엇보다도 다운로드 한 비밀키를 어떻게 검증할 것인지에 대한 방법을 제공하고 있지 않다.

FK00은 다중 서버를 사용한 패스워드-기반의 로밍 프로토콜을 제시하고 있는데, 공유키를 조회하기 위하여 패스워드 경화(hardening) 프로토콜을 소개한다. 그러나, 각 서버와의 인증된 통신 채널을 가정하고 있다.

Jab01은 안전한 채널을 필요로 하지 않으며, 다운로드 한 비밀정보의 검증방법을 제공하고 정당한 사용자의 실수에 의한 접속 실패를 다룰 수 있도록 한다. 그러나, 앞의 프로토콜과 마찬가지로 서버가 공격당하였을 경우 사용자는 자신의 비밀정보를 다운로드 할 수 없다. 이에 대한 하나의 방안으로 Threshold cryptosystem을 활용하는 방법을 고려해 볼 수 있을 것이다.

참고문헌

[1] T. Wu, "The Secure Remote Password Protocol", Internet Society Symposium on Network and Distributed System Security, pp. 97-111, 1998.

[2] S. Bellovin and M. Merritt, "Augmented Encrypted Key Exchange: a Password-Based Protocol Secure Against Dictionary Attacks and Password File Compromise", Technical report, AT&T Bell Laboratories, 1994.

[3] D. Jablon, "Strong Password-Only Authenticated Key Exchange", ACM Computer Communication Review, Vol.26, No.5, pp. 5-26, Oct. 1996.

[4] D. Jablon, "Extended Password Key Exchange Protocols Immune to Dictionary Attack", WETICE '97 Enterprise Security Workshop, Cambridge, MA, June 1997.

[5] V. Boyko, P. MacKenzie, and S. Patel, "Provably Secure Password-Authenticated Key Exchange Using Diffie-Hellman", Eurocrypt'00, pp. 156-171, 2000.

[6] S. Bellovin and M. Merritt, "Encrypted Key Exchange: Password-Based Protocols Secure Against Dictionary Attacks", Proc. IEEE Symposium on Research in Security and Privacy, Oakland, May 1992.

[7] R. Perlman and C. Kaufman, "Secure Password-Based Protocol for Downloading a Private Key", Proc. 1999 Network and Distributed System Security Symposium, Internet Society, 1999.

[8] W. Ford and B. Kaliski, "Server-Assisted Generation of a Strong Secret from a Password", Proc. 5th International Workshop on Enterprise Security, IEEE, 2000.

[9] D. Jablon, "Password Authentication Using Multiple Servers", CT-RSA 2001, LNCS 2020, pp. 344-360, 2001.