다중 서버를 이용한 패스워드 기반 키분배 방식에 관한 연구

안상만, 오수현, 원동호*

*성균관대학교, 정보통신공학부

A Study on Password-Based Key Exchange Method using Multiple Servers

Sang-Man Ahn, Soo-Hyun Oh, Dong-Ho Won*

*School of Information & Communications Engineering, Sungkyunkwan Univ.

요 약

패스워드 기반 키분배 프로토콜의 가장 중요한 요구사항 중 하나는 사전공격과 같은 패스워드 추측 공격에 대하여 안전하여야 한다. 그러나 지금까지 제안된 패스워드 추측 공격에 대한 안전성은 비밀 서버를 가정하고 있다. 즉, 검증자 기반 방식이더라도 서버에 저장되어 있는 패스워드 검증자가 비밀리에 보관되어야 한다는 단점이 있다.

본 논문에서는 새로운 방식의 다중 서버를 이용한 패스워드 기반 키분배 방식을 제안한다. 달리는 사용자에 대한 인증 및 검증자를 각 서버에 전송하는 일을 담당한다. 사용자는 특정 서버와 단독으로 세션키를 교환하지만, 서버는 세션키를 생성하기 위해서는 그룹 내에 있는 모든 서버와 비밀 복원 과정을 거쳐야만 하는 새로운 방식이다.

사용자와 키분배를 수행하는 특정한 서버는 그룹 내에 있는 다른 서버와 비밀 복원 과정을 거쳐야만 키분배 과정을 수행할 수 있으므로, 특정 서버의 패스워드 파일이 노출되어 패스워드 검증자가 공격자에게 노출되더라도 비밀 분산 과정을 수행하지 못하는 공격자는 패스워드 추측에 필요한 정보를 획득할 수 없다.

I. 서론

패스워드를 사용한 키분배 방식은 사용자가 단지 패스워드만을 사용하여 키분배을 수행할 수 있다는 점에서 최근에 매우 관심을 끌고 있는 프로토콜이다. 패스워드는 정보량적인 측면에서 낮은 엔트로피(불확실성)를 가지고 있기 때문에 패스워드에 대한 추측 공격에 약하다는 단점이 있다.

위와 같은 문제를 해결하기 위하여 Bellovin와 Merritt[1]는 서버의 공개키를 사용하지 않으며, 관용 암호방식과 공개키 암호방식을 혼합하여 방식으로 인증된 패스워드 기반 키분배를 하는 첫 번째 프로토콜인 EKE(Encrypted Key Exchange)를 제안하였다. EKE는 공격자가 패스워드에 대한 추측공격으로 암호문을 복호할 수 있지만, 키분배를 위해 사용된 공개키 암호 방식을 깰 수 없어, 추측한 값을 검증할 수 없다.

EKE가 제안된 이후, EKE보다 안전하고 새로운 특성을 만족하는 패스워드 인증 키분배 프로토콜들이 많이 제안되었다. 그러나 이전까지 제안된 패스워드 기반 키분배 프로토콜들은 비록 검증자 기반 방식이라고 하더라도, 패스워드에 대한 검증자를 "비밀 공개키(secret public key)"라고 하여 항상 비밀리에 보관되어야 한다. 즉, 서버에 저장되어 있는 패스워드 검증자가 공격자에게 노출되면, 공격자는 노출된 검증자를 이용하여 서버 위장 공격이 가능하며 또한, 공격자가 추측한 패스워드의 검증값으로 검증자를 사용하여 오프라인(off-line) 사전공격을 수행 할 수 있는 단점이 부각되었다.

이를 보완하기 위해 Ford와 Kaliski[2]는 공격자에 의한 추측공격을 방지하기 위하여 복수의 서버를 사용하는 프로토콜을 제안하였고, Jablon[3]은 [1]의 단점을 보완한 키 로밍 모델을 제안하였다. 그러나 이전의 방식들은 사용자 분산된 모든 서버와 등록 과정을 수행하며, 인증 및 키분배 과정을 모든 서버와 수행해야하는 단점으로 인하여 방대한 네트워크상에서 구현이 매우 어렵다.

MacKenzie[6]는 (k,n) threshold 비밀 분산 방식을 이용하는 멀티 서버 방식을 제안하였으나, 사용자가 패스워드 이외에 (n+1)개의 서버 공개키를 소유하고 있어야 하고, 또한 각 서버는 (4n+1)개의 공개키를 보유해야 한다는 단점이 있다. 단, n은 비밀 분산에 참여하는 서버의 개수이다.

본 논문에서는 이와 같은 단점을 보완하여, 사용자는 패스워드만을 이용하여 그룹 내에 존재하

는 서버 중, 특정 서버와 키분배 과정을 수행하고 또한, 서버에 저장되어 있는 패스워드 검증자가 노출되더라도 공격자가 패스워드 추측 공격을 수행할 수 없는 새로운 다중 서버를 이용한 패스워드 기반 키분배 방식을 제안한다.

II. 제안하는 다중 서버를 이용한 패스워드 기반 키분배 방식

1. 공개변수의 생성

ℓ은 제안하는 프로토콜의 안전 변수로써, 이산대수기반 공개키를 위한 안전 변수이며, 1024비트나 2048비트를 나타낸다.

본 논문에서 제안하는 방식에서 공개 변수의 분할 공격을 막기 위해 Z_p^*상의 원시원소를 사용해야 하나, 소수 위수 서브그룹(prime order subgroup)을 사용하기 때문에 소수의 형태가 매우 중요하다. 소수 위수 서브그룹은 기저 g의 위수인 $ord(g)$가 p-1의 소인수 q인 그룹 Z_p^*의 서브그룹을 의미하며, q의 크기는 160비트 이상일 때 안전하게 사용할 수 있다[5]. 따라서 소수 p는 $2q+1$($2^{\ell} > p > 2^{\ell-1}$, q : 160비트 이상의 큰 소수)의 형태를 선택한다. 여기에서 위수 q를 갖는 기저들로 나타낼 수 있는 그룹을 G_q로 표시한다.

$h(\pi)$는 사용자의 패스워드 π를 소수 위수 서브그룹을 만족하는 기저에 적합하도록 변환하는 함수(MGF-Mask Generation Function)로써, $h(\pi) \in G_q$를 만족하며, $g_\pi = h(\pi)^2 \bmod p$로 표시한다[2,3,4].

프로토콜에 대한 기술에 앞서 본 논문에서 사용하는 기호에 대한 정의는 표 1과 같다.

표 1 : 기호에 대한 정의

기호	정 의
p	: 큰 소수($2^{\ell} > p > 2^{\ell-1}$)이며, $p = 2q+1$
q	: 소수 위수 서브그룹을 갖는 g의 위수
g	: 위수 q를 갖는 기저
G_q	: 위수 q를 갖는 기저들로 나타낼 수 있는 그룹
π	: 사용자의 패스워드
$h()$	: Mask Generation Function
g_π	: Mask Generation Function으로 변환된 패스워드 값($g_\pi = h(\pi)^2 \bmod p$)
K	: 사용자와 서버의 세션키
x	: 딜리가 생성하는 그룹 비밀키
y	: 딜리가 생성하는 그룹 공개키
x_i	: 서버에게 할당되는 비밀 분산용 "shares"
V_U	: 패스워드 검증자(Verifier)
ID_U	: 사용자의 ID
S_i	: 비밀 분산에 참여하는 i번째 서버의 ID
I	: 비밀 분산에 참여하는 서버의 집합 ($1 \le i \le n$)
D	: 각 서버에게 부분 정보를 분배하는 딜리

2. 셋업 과정

셋업 과정은 등록 과정과 키분배 과정 이전에 딜러가 그룹 공개키 및 비밀 복원시에 사용하는 "share"를 생성하는 과정으로 구성되어 있다. 딜러와 서버 $\{S_i\}_{i \in I}$간의 통신은 안전한 통로(secure channel) 상에서 전송된다고 가정한다. 이후 모듈러 p 연산이 명확한 식에 대해서는 $\bmod p$를 생략한다.

1) 딜러의 셋업 과정

딜러 D는 키분배 프로토콜에 참여하는 n개의 서버 $\{S_i\}_{i \in I}$의 그룹키(global key)쌍 (x, y)를 생성하기 위하여, $x \in_R Z_q^*$를 만족하는 그룹 비밀키 x를 선택한 후, 그룹 공개키 $y = g^x$를 계산하여 게시판(Bulletin board)에 공개한다.

또한, 딜러 D는 Pinch의 "On-Line multiple secret sharing"[7]을 이용하여, 그룹 비밀키 x에 대한 $\{S_i\}_{i \in I}$의 'share'인 x_i를 아래와 같이 계산한다.

① 딜러 D는 각 서버 $\{S_i\}_{i \in I}$의 비밀값 $x_i < q$를 랜덤하게 선택한다.

② 비밀값 x_i을 비밀리에 각 서버 $\{S_i\}_{i \in I}$에게 전송한 후, 다음과 같이 T값을 계산한다.

$$T = x - f(g^{\prod_i x_i}) \quad (1)$$

③ 딜리 D는 (p,q,g,T,y) 값을 게시판에 공개한다.

2) 서버 $\{S_i\}_{i\in I}$의 셋업 과정

딜리로부터 안전하게 비밀값 x_i를 전송 받은 서버 $\{S_i\}_{i\in I}$는 자신의 데이터베이스에 안전하게 보관한다.

3. 등록 과정

등록 과정은 사용자에 대한 인증 및 패스워드 검증자를 딜리에게 전송하는 과정이다. 따라서, 각 서버 $\{S_i\}_{i\in I}$들은 등록 과정에서는 사용자와 직접적인 접촉 없이 딜리로부터 검증자를 안전한 채널을 통하여 단순히 전송 받아 사용자의 아이디와 함께 저장한다. 이러한 특징은 지금까지 제안된 다중 서버를 이용한 패스워드 기반 키분배 프로토콜[2,3,5]들이 등록 과정에 모든 키 서버들이 참여하는 것에 비하여 훨씬 효율적이다.

사용자와 딜리는 패스워드 검증자를 교환하기 위하여 아래와 같은 과정을 수행한다..

① 사용자는 패스워드 π를 Z_p^*상의 곱연산에 대한 위수가 q인 원소로 대응시키는 함수 h를 이용하여 $g_\pi = h(\pi)^2$를 생성한다.

② 사용자는 랜덤수 $a \in_R Z_q$를 선택하여, g_π에 대한 ElGamal 암호문 E_C를 아래와 같이 생성하여 딜리에게 전송한다.

$$E_C = (y^a \cdot g_\pi, g^a) \quad (2)$$

③ 암호문 E_C를 전송 받은 딜리는 아래와 같은 복호화 과정을 거쳐 g_π를 구한다.

$$\begin{aligned} g_\pi &= (y^a \cdot g_\pi)/(g^a)^x) \\ &= (g^{x\cdot a} \cdot g_\pi)/(g^{a\cdot x}) \end{aligned} \quad (3)$$

④ g_π를 구한 딜리 D는 사용자 ID_U에 대한 검증자를 아래와 같이 생성한다.

$$V_U = g_\pi^x \quad (4)$$

딜리는 V_U와 ID_U를 함께, 각 서버 $\{S_i\}_{i\in I}$ 에게 비밀리에 전송한 후, g_π 및 V_U를 자신의 메모리에서 삭제한다.

⑤ ID와 검증자 V_U를 전송 받은 서버 $\{S_i\}_{i\in I}$들은 위 값을 안전하게 자신의 데이터베이스에 저장한다.

4. 인증 및 키분배 과정

사용자는 특정 서버 $i_{\in I}$와 자신의 패스워드만을 이용하여 인증과 세션키를 교환한다. 본 논문에서 제안하는 프로토콜은 묵시적 키 인증을 제공한다.

① 사용자는 랜덤수 $r_u \in_R Z_q$를 선택하고, 아래와 같은 정보를 생성하여 i번째 서버에게 전송한다.

$$m = g^{r_u} \cdot g_\pi \quad (5)$$

② m을 전송 받은 서버 i는 그룹 내에 속해있는 다른 서버 $j_{\neq i}$들과 비밀 분산 복원과정인 $REC_x(p,q,g,T,y)$을 통하여 복원한 그룹 비밀키 x와, 서버 i에 저장되어 있는 패스워드 검증자 V_U를 이용하여, 사용자 인증 및 키분배를 위한 토큰을 생성하기 위한 아래의 연산을 수행한다.

$$\mu = \frac{m^x}{V_U} = \frac{g^{r_u\cdot x} \cdot g_\pi^x}{g_\pi^x} = g^{r_u\cdot x} \quad (6)$$

③ 서버 i는 키분배를 위한 랜덤수 $r_S \in_R Z_q$를 선택하고, 아래와 같은 세션키 생성을 위한 정보를 생성하여 사용자에게 전송한다.

$$\sigma = y^{r_S} \quad (7)$$

④ σ를 전송 받은 사용자는 아래와 같이 세션키를 계산한다.

$$K = \sigma^{r_U} = g^{x\cdot r_U\cdot r_S} \quad (8)$$

⑤ 랜덤수 r_S를 선택한 서버 i는 아래와 같이 세션키를 생성한다

$$K = \mu^{r_S} = g^{x\cdot r_U\cdot r_S} \quad (9)$$

5. 서버의 그룹 비밀키 복원 과정

사용자로부터 인증 정보를 받은 서버 S_i는 그룹 비밀키 x를 복원하기 위하여 Pinch[7]의 On-Linc 비밀 분산을 이용하는 $REC_x(p,q,g,T,y)$를 수행한다. 비밀 복원 과정의 자세한 수행 과정은 다음과 같다.

① 서버 $S_{i\in I}$는 $r \in_R Z_q^*$를 선택하여 $g^{r\cdot x_i}$를 계산하고, 다음 서버 $S_{j,\forall j\in I\setminus\{i\}}$에게 위 값을 전송한다.

② 위 값을 전송 받은 $S_{j,\forall j\in I\setminus\{i\}}$는 $(g^{r\cdot x_i})^{x_j}$를 계산하고, 이 값을 서버 S_{j+1}에게 전송하며,

서버 S_i를 제외한 모든 서버가 위 과정을 수행한다.

③ ②번 과정을 수행하는 마지막 j번째는 서버 $S_{j,\forall j \in \Lambda\{i\}}$는 $g^{r \cdot x_i \cdot x_j}$를 계산하여 서버 S_i에게 위 결과 값을 전송한다.

④ 서버 S_i 는 다음과 같이 그룹 비밀키를 복원하기 위한 중간값 V_x를 계산한다.

$$V_x = (g^{r \cdot x_i \cdot x_j})^{r^{-1}} = g^{x_1 \cdot x_2 \cdots x_n} \quad (10)$$

⑤ 서버 S_i 는 다음과 같이 그룹 비밀키 x를 복원한다.

$$x = T + f(V_x) \quad (11)$$

⑥ 복원한 비밀이 정확한지 검증하기 위해 g^x 과 y를 검사한다.

III. 결론 및 향후 연구 계획

본 논문에서는 믿을 수 있는 딜러를 가정하여 사용자와 인증 및 패스워드 검증자를 교환하고, 사용자가 실제 네트워크상에 존재하는 서버와 키 분배를 수행할 때는 전체 서버를 대상으로 프로토콜을 수행하지 않고 그룹 상에 존재하는 서버 중, 특정한 서버와 단독으로 키분배를 수행할 수 있는 새로운 방식의 멀티 서버를 이용한 패스워드 기반 키분배 방식을 제안하였다.

각 서버가 사용자와 키분배를 수행하기 위해서는 그룹 내에 있는 모든 서버와 비밀 복원 과정을 수행해야 하므로, 각 서버에 존재하는 패스워드 검증자는 서버 단독으로 패스워드 검증자를 복원할 수 없고, 패스워드 검증자가 공격자에게 노출되더라도 그룹 내에 존재하는 모든 서버가 공격당하지 않는 한 패스워드 추측 공격이 불가능하다.

그러나 서버가 한번 비밀 복원 과정을 하면, 그룹 비밀키가 서버에 노출된다는 단점이 존재한다. 이러한 문제는 재 사용 가능한 비밀 분산 방식 등, 다른 비밀 분산 방식을 사용함으로써 해결할 수 있을 것이다.

참고문헌

[1] S.M. Bellovin, M. Merritt, "Eencrypted Key Exchange: Password-Based Protocols Secure Against Dictionary Attacks", Proceedings of the IEEE Symposium on Research in security and Privacy, 1992.

[2] W. Ford & B. Kaliski, "Server-Assisted Generation of a Strong Secret from a Password", IEEE 9th International Workshops on Enabling Technologies: NIST, Gaithersburg MD, June 14-16, 2000.

[3] D. Jablon, "Password Authentication Using Multiple Servers", LNCS 2020: Topics in Cryptology-RSA 2001, April 8-12, 2001 Proceedings, pp. 344-360, 2001

[4] D. Jablon, "Extended Key Exchange Protocols Immune to Dictionary Attacks", Proceedings of the Sixth Workshops on Enabling Technologies, IEEE Computer Society, June 18-20, 1997, Cambridge, MA, pp. 248-255.

[5] D. Jablon, "Strong Password-Only Authenti- cation Key Exchange", Computer Communi- cation Review, Vol.26, No.5, pp.5-26, October, 1996.

[6] P. MacKenzie, T. Shrimpton, & M. Jakobsson, "Threshold Password-Authenticated Key Exchange", CRYPTO 2002, Wed. Aug. 21, 11:55-12:20.

[7] R. G. E. Pinch, "On-Line Multiple Secret Sharing", Electronics Letters, 1996.

[8] "키분배 프로토콜 설계에 관한 연구", 성균관 대학교, 1998.