2001년 한국정보처리학회 춘계 학술발표논문집 제8권 제1호

새로운 일회용 패스워드 방식 제안

이용호, 박희운, 이임영
순천향대학교 정보기술공학부
e-mail:abysskey@cse.sch.ac.kr

A new One-Time Password System Proposed

Yong-Ho Lee, Hee-Un Park, Im-Yeong Lee
Division of Information Technology Engineering, Soonchunhyang University

요약

컴퓨터와 인터넷의 발전을 통하여 실생활에서 이루어지는 많은 일들이 가상공간에서 실현 가능하게 되었다. 그러나 이러한 가상공간을 활용함에 있어서 고려할 사항은 시스템 접속 요구자의 정당성을 확인하는 식별 및 인증 과정이다. 이러한 고려사항을 해결하기 위해서 많은 일회용 패스워드 방식이 제안되었다. 기존에 제안된 방식들은 도청이나 재시도 공격에는 안전하나 man-in-the-middle attack이나 서버 위장 공격에 취약점을 드러내고 있다.

본 논문에서는 기존의 일회용 패스워드 기술들 중 대표적인 방식들을 알아보고, 이에 대한 문제점을 도출한다. 그리고 일회용 패스워드 기술의 일반적인 고려사항과 기존에 제안된 방식들의 문제점들을 해결할 수 있는 새로운 일회용 패스워드 방식을 제안한다.

1. 서론

컴퓨터와 네트웍의 발전은 우리 환경에 많은 변화를 가져왔고, 실생활에서 이루어지는 대부분의 일들을 가상공간상에서 해결할 수 있게 되었다. 이와 같은 네트웍의 이용시 우선적으로 고려해야 할 사항은 컴퓨터 사용자의 정당성을 확인하는 사용자 인증 기술이다.

정당한 사용자를 확인하기 위한 사용자 인증 기술 중 가장 일반적인 방식은 패스워드 인증 방식이다. 그러나, 이 방식은 네트웍 환경에서 많은 문제점을 가지고 있다. 특히, 패스워드가 네트웍을 통해 전송되므로 도청과 같은 위험요소에 매우 취약할 뿐만 아니라 한번 정해진 패스워드를 계속적으로 사용하게 되므로, 한번 도난 당하면 새로운 패스워드를 다시 생성 해야하는 문제점을 가지고 있다.

이러한 문제점을 해결하기 위해서 현재 많은 패스워드 누출방지 기술이 나오고 있다. 이 중에서 최근에 부각되고 있는 인증 방식이 일회용 패스워드 방식이다.

본 고에서는 기존에 제안된 일회용 패스워드 방식들 중 대표적인 것들을 알아보고 이에 대한 문제점을 도출한다. 그리고 기존의 문제점들을 해결할 수 있는 안전하고 효율적인 일회용 패스워드 방식을 제안한다.

2. 기존 방식 분석

본 장에서는 기존에 제안된 방식 중 대표적인 것들에 대하여 알아보고 각각의 문제점에 대해 분석해 본다.

2.1 RFC 1760 표준 방식

이 장에서는 일방향 해쉬 함수 H를 사용하여 일회용 패스워드를 생성하는 표준 방식에 대해 기술한다[1][2]. 또한, 패스워드 생성에 있어 seed 값을 생성하고 이를 기초로 하여 매번 접속시 새로운 일회용 패스워드로 인증 과정을 수행한다.

2.1.1 시스템 계수

다음은 이 방식에서 사용되는 시스템 계수를 서술한 것이다.

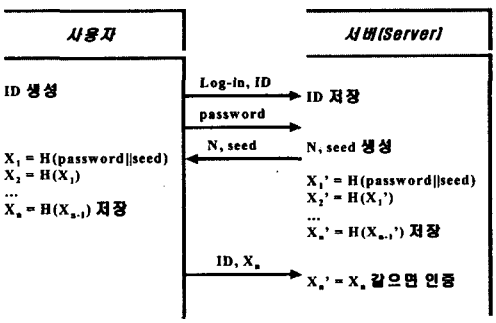
- seed : 사용자 및 서버에서 패스워드 생성시 사용하는 초기 공유 값
- H : 안전한 일방향 해쉬 함수
- X_n : seed 값을 n번 해쉬한 값
- ID_U : 사용자 U의 식별 ID

2.1.2 프로토콜

(1) 초기화 단계
- 사용자는 ID_U를 생성해 서버에 log-in 한다.
- 서버는 seed 값과 해쉬할 회수 n을 선택하여 안전하게 사용자에게 전송한다.

(2) 패스워드 생성 단계
- 사용자는 seed 값과 자신의 비밀 password값을 연접해 해쉬 함수 H를 이용하여 n회 해쉬한다. 해쉬한 n개의 값은 안전하게 저장한다.
- 사용자는 자신의 비밀 password를 서버에게 전송한다. 서버는 이를 수신한 다음 사용자와 마찬가지로 해쉬 함수 H를 이용하여 X_1'에서 X_n'까지 생성한 후 그 값을 저장한다.

(3) 인증 과정
- 사용자는 서버에게 접속하기 위하여 ID_U, X_n을 전송한다.
- 서버는 수신한 X_n과 저장되어 있는 X_n'을 비교하여 일치하면 사용자를 인증한다.



(그림 1) RFC 1760 표준 방식의 흐름도

사용자가 log-out 후 다시 log-in 할 경우 사용자는 ID_U, X_{n-1}을 전송해 서버에게 인증을 받는다. 이렇게 하여 사용자 인증을 위해서 매 접속시 마다 다른 패스워드를 사용하게 된다.

2.1.3 RFC 1760 표준 방식 분석

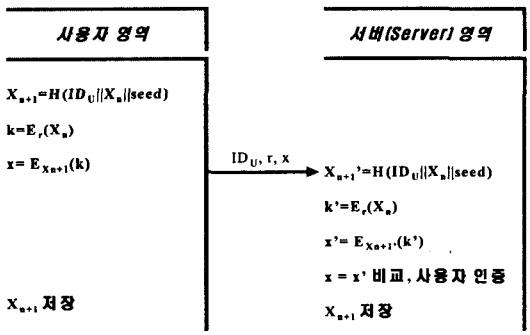
이 방식은 다음과 같은 특징을 갖고 있다[3][7].
- 구조가 단순하여 시스템에 적용하기 유용하다.
- 인증시 패스워드가 매번 새롭게 변하므로, 도청이나 재시도 공격에 안전하다.
- 그러나 이 방식은 초기에 사용 횟수를 제한하고 있어 인증에 사용되는 일회용 패스워드를 미리 생성하고 보관해야 하는 부담을 가지고 있다.
- 또한 일방향 해쉬 함수에 안전성을 두고 있다는 문제점을 가지고 있다.

2.2 S/KEY를 개선한 일회용 패스워드 방식

이 방식은 RFC 1760 표준 방식의 문제점을 개선하기 위해 제안된 방식이다.

2.2.1 프로토콜

본 방식은 초기화 설정 시 공유되는 seed를 이용하여 초기 패스워드 X_0를 생성한다. 초기화 설정 후 사용자는 (그림 2)와 같은 방법으로 인증 과정을 수행한다.



(그림 2) S/KEY를 개선한 일회용 패스워드 방식의 흐름도

2.2.2 S/KEY를 개선한 일회용 패스워드 방식 분석

이 방식은 다음과 같은 특징을 가지고 있다.
- 안전성을 암호화 알고리즘에 의존하여 표준 방식의 취약점을 해결하고 있다.
- 그러나 이 방식은 전송되는 암호화된 인증 인자의 키로서 서버에 저장되어 있는 비밀값이 그대로 사용되고, 서버 인증 과정이 없으므로 위장 공격에 취약하다.

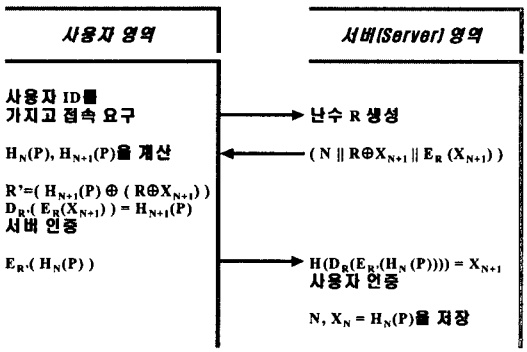
2.3 변형 일회용 패스워드 방식

이 방식은 상기 제시된 방식의 문제인 전송되는 값의 안전성과 위장 공격을 해결할 수 있는 방식으로 다음과 같은 특성을 가지고 있다.
- 일회용 패스워드 특성

· Challenge-Response 방식
· 단방향 해쉬함수 특성
· 사용자-인증서버 상호인증 기능
· 소프트웨어로 적용 가능

2.3.1 프로토콜

본 방식은 초기화 설정 시 일회용 패스워드 사용 회수 N을 결정한다. 사용자는 해쉬함수를 사용하여 비밀정보 P를 N+1회 해쉬한다. 이 해쉬된 값과 N+1을 서버에게 전달한다. 초기화가 끝난 후 사용자와 서버는 (그림 3)과 같은 방식으로 인증이 수행된다.



(그림 3) 변형 일회용 패스워드 방식의 흐름도

2.3.2 변형 일회용 패스워드 방식 분석

이 방식은 다음과 같은 특징을 갖고 있다.

· 표준 일회용 패스워드 방식과 마찬가지로 사용 횟수 N을 결정한다.
· 서버에 저장되어 있는 정보는 N+1과 X_{N+1} 이다.
· 상호인증을 통하여 사용자 및 서버의 위장공격을 방어할 수 있다.
· 그러나 이 방식은 상호간에 전송되는 값들을 이용한 인증 인자 공격에 취약성을 보이고 있다.

3. 새로운 일회용 패스워드 방식 제안

본 장에서는 새로운 방식의 일회용 패스워드 방식을 제안한다. 본 방식은 안전하고 효율적으로 사용자와 서버 사이의 상호 인증을 수행한다.

3.1 프로토콜 제안

3.1.1 시스템 계수

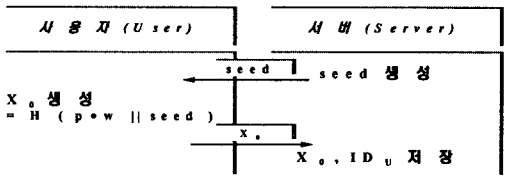
· seed : 일회용 패스워드 설정 초기 값
· P · W : 사용자만이 가지고 있는 비밀 패스워드
· ID_U : 사용자 U의 ID
· T_U : 사용자(접속자) U의 Time Stamp
· R : 랜덤 수
· X_n : n번째 일회용 패스워드
· H : 안전한 일방향 해쉬 함수
· $E_A(B)$: 메시지(B)를 키(A)로 암호화하는 안전한 암호 알고리즘

3.1.2 프로토콜

(1) 초기화 단계

이 단계는 off-line상에서 안전하게 이루어지며, 그 과정은 다음과 같다.

· 서버는 스마트 카드와 같은 안전한 채널을 이용하여 사용자에게 seed 값을 전달한다.
· 사용자는 seed값과 자신의 비밀 password를 연접해 해쉬 함수 H를 수행하여 해쉬값 X_0을 만든다. 이 값을 서버에게 안전하게 전달한다[5].



(그림 4) 제안 방식의 초기화 단계

(2) 서버 인증 단계

· 사용자는 ID_U를 가지고 서버에 log-in 한다.
· 서버는 난수를 생성한 후 seed값과 연접한다. 이를 해쉬해서 암호 키를 생성한다.

$$KE = H(r \| seed)$$

· 이 암호 키와 X_n을 이용하여 서버 인증 인자S_M을 생성한다. 이를 난수와 함께 사용자 U에게 전송한다.

$$S_M = E_{KE}(X_n)$$

· 사용자는 수신 정보를 이용해 다음을 계산한다.

$$KE' = H(r \| seed)$$
$$S_M' = E_{KE'}(X_{n-1})$$

· 사용자는 수신된 S_M과 S_M'가 동일한지 확인하고, 만약 동일하다면 정당한 서버로서 인정된다.

(3) 사용자 인증 단계

· 서버 인증 과정을 거친 후 사용자는 다음과 같이 사용자 인증 인자를 생성하여 타임스탬프와 함께 서버에게 전송한다.

$X_{n+1} = H(X_n \| T_U \| ID_U)$

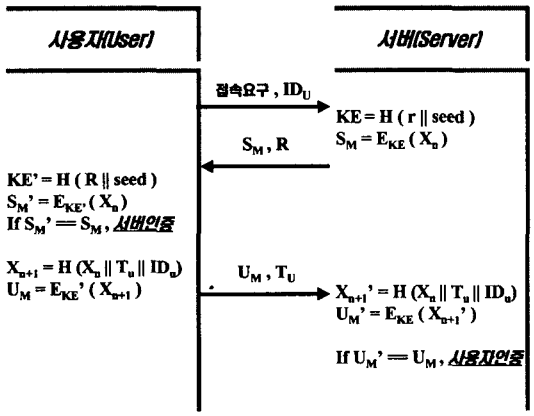
$U_M = E_{KE}(X_{n+1})$

· 서버는 수신된 정보를 이용해 다음을 생성한다.

$X_{n+1}' = H(X_n \| T_U \| ID_U)$

$U_M' = E_{KE}(X_{n+1}')$

· 서버는 수신된 U_M과 U_M'가 동일한지 확인하고, 만약 동일하다면 정당한 사용자로서 인증한다.



(그림 5) 제안 방식의 흐름도

3.1.3 제안 방식 분석

제안 방식은 일회용 패스워드 방식을 이용하여 안전하고 효율적인 상호 인증을 가능하게 하는 방식이다. 안전성이 암호 알고리즘에 의존하고, 안전한 인증 인자 생성을 통하여 기존의 문제점을 완전하게 개선하고 있다. 또한 효율적이고 간편한 구조로 이루어져 있다. 다음 <표 1>은 기존에 나와있는 방식과 제안 방식을 비교 분석한 결과이다.

<표 1> 각방식별 비교 분석

항목 / 방식	표준 문서 RFC1760	기존 방식 I	기존 방식 II	제안 방식
사용 횟수	n회	제한 없음	n회	제한 없음
패스워드 노출 방어	0	0	0	0
패스워드 재전송 방어	0	0	0	0
사전공격 방어	X	0	0	0
인증 인자 공격 방어	X	0	X	0
위장 공격 방어	X	X	0	0
안전성	해쉬함수에 근거	해쉬함수에 근거	암호알고리에 근거	암호알고리즘에 근거
상호 인증	X	X	0	0

4. 결론

네트웍의 이용에 있어서 허가된 사용자만이 자원에 접근할 수 있는 접근 통제 방식은 사용자와 서비스를 제공하는 서버 모두에게 매우 중요한 요소가 되었다. 본 논문에서는 기존에 제안되어 있는 몇몇 일회용 패스워드 방식들을 살펴보았다.

RFC 1760 표준 방식은 사전공격, 인증 인자 공격, 위장 공격 등에 취약하며, S/KEY를 개선한 일회용 패스워드 방식은 사전 공격, 인증 인자 공격 등에 취약하다. 또한 변형 일회용 패스워드 방식은 인증 인자 공격에 매우 취약함을 알 수 있었다.

본 논문에서 제안된 방식은 일회용 패스워드 방식의 기본적인 요구사항을 만족하고 있으며, 상기 방식들의 문제점들을 완벽하게 해결하고 있다.

향후 고도의 정보화 사회로의 발전을 고려할 경우 앞으로 더욱 안전하고 효율적인 방식을 위한 광범위한 연구가 진행되어야 할 것이다.

참고문헌

[1] N. Haller, "The S/Key One-Time Password System", RFC 1760, 1995.

[2] R. Rivest, "The MD4 Message- Digest Algorithm", RFC 1320, April 1992.

[3] R. Rivest, "The MD5 Message-Digest Algorithm", RFC 1321, April 1992.

[4] "전산망에서의 패스워드 누출 방지 기술 개발 보고서", 한국정보보호센터, 1997. 12.

[5] Mudge, "Vuleravilities in the S/Key one time password system",
http://10pht. cp/~modge/skey_white_paper.html

[6] B. Schneier, Applied Cryptography, Second Edition, John Wiley & Sons, 1996.