2002년도 대한전기학회 하계학술대회 논문집 2002.7.10-12

자바 애플릿을 이용한 시간 기반 사용자 인증 시스템

민수홍, 나인순, 조동섭
이화여자대학교 컴퓨터학과

The Time Based Authentication System using Java Applet

Su-Hong Min, In-Soon Na and Dong-Sub Cho
Dept of Computer Science and Engineering, Ewha Womans University

Abstract - 인터넷 서비스의 확대에 따라 이를 이용하는 사용자와 다양한 웹사이트들이 생겨나게 되었다. 이와 함께 사용자의 정보와 웹사이트들의 컨텐츠를 보호하기 위한 목적으로 인터넷 보안에 대한 인식도 커지게 되었다. 본 논문에서는 패스워드 기반 사용자 인증 시스템에 대해 연구하였다. 사용자는 시스템 자원을 활용하기 위해서 시스템으로부터 사용자 인증을 받아야 한다. 일반적으로 패스워드 기반 인증 방식에서는 사용자가 시스템에 등록한 ID, 패스워드를 통해 인증을 받는다. 그러나 현재 숫자나 문자로 이루어진 패스워드 입력 값은 인증 받지 않은 사용자에게 도용될 가능성이 높다. 따라서 본 논문에서는 숫자, 문자로 이루어진 패스워드에 입력 시간 값을 적용해서 패스워드를 암호화하는 방법에 대해 제시한다.

1. 서 론

최근 몇 년 동안 인터넷을 사용하는 사용자의 수는 급격히 증가하였으며, 그 이용 범위 또한 다양한 분야로 확대되어가고 있다. 사용자들은 인터넷을 통해 전자 상거래 서비스, 이메일 서비스 등의 다양한 서비스를 이용할 수 있으며, 이들을 제공하는 웹사이트들은 사용자의 인증을 거쳐 자사의 컨텐츠들을 제공하고 있다. 현재 대부분의 웹사이트들은 패스워드 기반의 인증 시스템을 채택해 사용한다. 기존의 패스워드 기반의 인증 시스템을 살펴보면, 사용자는 시스템의 자원을 활용하기 위해서 시스템으로부터 사용자 인증을 받아야 한다. 일반적으로 패스워드 기반 인증 방식에서는 사용자가 시스템에 등록한 패스워드와 아이디를 통해 시스템으로부터 인증을 받는다. 그러나 현재 사용자가 등록하는 패스워드는 키보드로부터 입력한 간단한 숫자와 문자의 조합으로 이루어져 있어 악의를 가진 사용자에 의해 불법적으로 이용될 수 있다. 따라서, 본 논문에서는 이 같은 문제점을 해결하기 위해 사용자가 키보드를 통해 패스워드를 입력할 때, 입력하는 각각의 문자 값과 그에 해당하는 입력시간 (the duration of keystrokes)을 측정해서 이를 시간기반 패스워드 암호화 알고리즘 [5]을 이용해 사용자가 입력한 패스워드를 암호화하는 방법을 제시하였다. 본 논문에서 제시한 시간기반 사용자 인증 시스템은 사용자가 키보드를 통해 입력한 패스워드의 문자 값과 그에 해당하는 키보드를 누른 시간의 상대적인 장단 (다른 문자 값을 입력했을 때의 시간과 비교해서)을 이용한 것으로, 기존의 패스워드 기반 인증 시스템에 시간 값을 이용해 성능을 향상시켰다. 또한 간단한 알고리즘으로 이루어져 있어 프로그램의 이식성이 매우 용이해 사용자 인증 시스템을 필요로 하는 다양한 분야에 적용이 가능하다.

본 논문의 구성은 다음과 같다.

2장에서는 관련 연구로서 기존의 사용자 인증 서비스 중 ID-패스워드 기반의 인증 시스템에 대해 기술하고, 3장에서는 본 연구에서 제안하고 있는 시간기반 사용자 인증 시스템의 구현된 기능과 수행 절차에 대해 기술한다. 그리고 마지막 4장에서는 결론 및 향후 방향에 대해 기술한다.

이 논문은 과학기술부의 '여자대학교 연구기반 확충사업' 에 의하여 지원되었음.

2. 관련 연구

이 장에서는 기존의 패스워드 기반 사용자 인증 시스템의 패스워드 기술에 대해서 알아보고자 한다.

2.1 일회용 패스워드 (One-Time Password)

S/Key에 기반을 둔 OTP (One-Time Password)는 래슬리 램포트 (Laslie Lamport)가 설계하고 벨코어 (Bellcore)사가 개발하였다. OTP는 사용자가 인증을 받고자할 때 매번 새로운 패스워드를 사용해야 하는 보안 시스템으로 전 세계적으로 가장 많이 쓰이는 프로토콜이다. OTP는 몇 개의 초기 값(사용자가 제공하거나 임의로 생성)을 가지고 시작하여 키의 순서를 얻기 위해 선택된 MD4 또는 MD5 해싱 알고리즘을 반복적으로 적용하면서 동작한다. OTP를 이용해 패스워드를 설정한 사용자는 매번 로그인을 시도할 때마다 새로운 패스워드를 이용한다. 이 방법은 침입자들이 이용하고 있는 스니핑 (Sniffing: 패킷 가로채기) 공격을 근본적으로 막아준다 [1,2,4].

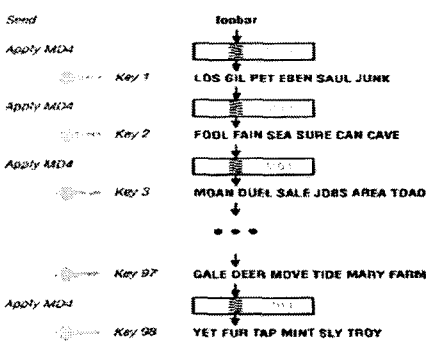

그림1 OTP 작동 방법 [2]

2.2 Challenge-Response 방식

사용자가 인증 요구와 함께 사용자 식별 번호(PIN)를 인증 서버에게 전달하면, 인증 서버는 난수를 생성하여 challenge로 사용자에게 전달한다. 이와 동시에 인증 서버는 이용자의 사용자 식별 번호에 해당하는 패스워드를 키 데이터 베이스에서 꺼내 이것을 이용하여 난수의 암호화를 시작한다. Challenge를 받은 사용자는 그것을 자신의 패스워드로 암호화하여 response로 인증 서버에게 반환한다. 사용자로부터 response를 받은 인증 서버는 서버 자신이 계산한 값과 수신된 response 값을 비교하여 일치하는 경우에 사용자를 정당한 사용자로 인증한다. Challenge-Response 방식은 여러 번의 절차로 인해 다소 느리다는 단점이 있기는 하지만, 복잡성이 낮은 반면 안전성이 높기 때문에 최근 이 방법을 적용한 인증 시스템이 국외에서 많이 개발되고 있는 실정이다 [3].

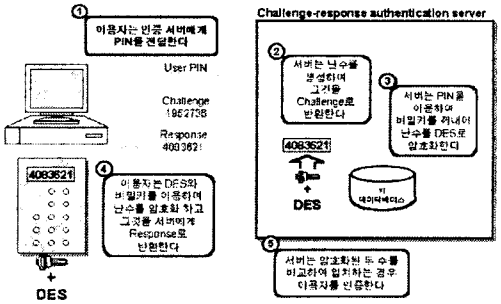

그림 2 Challenge-Response 동작 방법 [3]

3 시간 기반 사용자 인증 시스템

3.1 시스템 개요

본 논문에서 제시한 자바 애플릿을 이용한 사용자 인증 시스템은 사용자가 패스워드를 등록할 때 키보드를 통해 입력한 패스워드의 문자 값과 그에 해당하는 입력시간 (the duration of keystrokes)을 이용해서 새로운 패스워드로 암호화시키는 방식이다. 다음은 사용자 인증 시스템의 전체 구조이다.

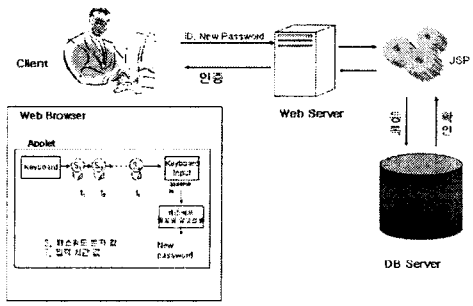

그림 3 시간 기반 사용자 인증 시스템

본 시스템을 살펴보면, 클라이언트 상에서 동작하는 프로그램은 자바 애플릿으로 자바 가상 기계 (JVM)가 웹 브라우저 안에 탑재되어 있어 별도의 작업 없이 브라우저에서 실행이 가능하다. 클라이언트가 서버로부터 데이터를 요청할 경우, JSP (Java Server Pages)를 통해 웹서버와 데이터베이스 서버에 접근한다. 클라이언트는 서버의 자원을 이용하기 위해서 인증을 거쳐야 된다. 클라이언트가 웹 브라우저를 통해 자신의 정보 (ID, 패스워드, 이름, 주민번호 등)를 서버에 제공할 때, 자바 애플릿이 실행된다. 자바 애플릿은 패스워드를 클라이언트 상에서 암호화하기 위해 필요하다. 클라이언트가 패스워드를 입력하면 입력하는 문자와 그에 해당하는 입력시간 (the duration of keystrokes)이 동시에 배열에 저장된다. 단, 클라이언트는 패스워드 입력 시 입력 시간의 장단을 고려해야 한다. 패스워드의 입력이 끝나면 저장된 패스워드는 각각의 문자와 입력 시간을 이용해 시간기반 패스워드 암호화 알고리즘을 통해 새로운 패스워드로 생성된다. 이때 만들어진 패스워드는 클라이언트는 알 수 없다. 클라이언트는 단지 패스워드로 입력한 문자와 각각의 입력시간의 상대적인 장단만 알고 있다. 새롭게 만들어진 패스워드는 JSP로 작성된 프로그램을 통해 웹 서버를 거쳐 JDBC (Java Database Connectivity)를 통해 데이터베이스 서버에 저장된다. 서버에 클라이언트가 등록이 되면 이제 본격적으로 서버를 이용할 수 있게 되는데, 서버를 이용할 때는 반드시 인증을 거쳐야 한다. 이때 인증 페이지 또한 자바 애플릿이 실행되는 웹브라우저로 클라이언트가 자신의 ID와 패스워드를 입력하면, 입력된 패스워드는 시간 기반 패스워드 암호화 알고리즘을 통해 새로운 패스워드로 만들어 서버에 저장된 것과 일치하는지 확인한다.

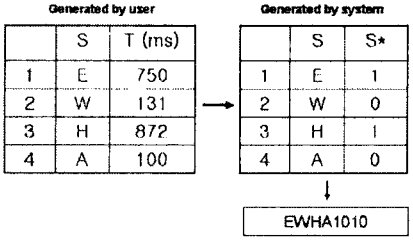

그림 4 패스워드 생성

3.2 시스템 환경

- Web Server: Apache Web Server
- DB Server: MS SQL Server 2000
- OS: MS Windows 2000 Advance Server
- Development tool: JAVA SDK-1.4, JBuilder 5.0
- Language: JAVA, JSP

3.3 구현 방법 및 수행 절차

1단계: 사용자 등록

1-1: 클라이언트 상의 웹 브라우저는 키보드를 통해 패스워드로 입력하는 각각의 문자 값과 입력시간을 입력받아 메시지 큐에 저장한다.

1-1-1: 키보드로 키를 입력받을 때 키의 핸들링이 요구되는데, 일반적으로 키보드에서 특정키를 오래 누르고 있을 경우, 문자가 연속해서 찍히게 되는 점을 고려해서 키의 입력 시간 동안 문자가 하나만 찍히도록 제어한다.

1-1-2: 입력시간은 키를 눌렀을 때 (pressed of the key)의 시간과 키 눌림이 해제(released of the key)되었을 때의 시간을 각각 구한 다음 두 시간차를 구한 값이다.

1-2: 패스워드 입력을 마치면, 메시지 큐에 저장된 각각의 입력시간을 이용해 시간의 장단을 구한다. 시간의 장단을 구하는 방법은 다음과 같다.

1-2-1: 임계치 (threshold)를 이용할 경우:

사용자는 입력시간의 평균 (식 1)을 구한 다음, 각각의 입력시간과 평균을 비교해서 평균보다 큰 값은 상위그룹에, 평균보다 작은 값은 하위 그룹에 저장을 한다. 저장된 상위 그룹의 최하 값과 하위 그룹의 최상 값의 평균을 이용해 임계치 (식 2)를 정한다. 임계치가 정해지면, 임계치보다 큰 입력시간은 '1'로, 임계치 보다 작은 입력시간은 '0'으로 설정한다.

$$T_a = \sum_{i=1}^{n} Ti / N \qquad (1)$$

$$T_t = T_{h\_min} + T_{l\_max} / 2 \qquad (2)$$

1-2-2: 시간의 장단을 구분하지 않을 경우:

사용자가 입력시간의 장단을 고려하지 않고, 일정하게 패스워드를 입력할 경우를 말한다. 본 논문에서는 배열에 저장된 입력 시간의 최대값과 최소값의 차가 100ms 이하 범위를 만족하면 이 경우에 속한다고 가정한다. 이때 입력시간의 장단은 모두 '0'으로 설정한다.

1-2-3: 예외 처리를 할 경우:

사용자가 입력 시간의 장단을 분명하게 하지 않을 경우를 말한다. '장'으로 설정된 입력 시간과 '단'으로 설정된 입력 시간의 차이가 거의 없어 사용자 자신이 장단을 인지 할 수 없을 경우로 본 논문에서는 상위 그룹의 최하 값과 하위 그룹의 최상 값의 차가 100ms 이하의 범위를 만족할 때 이 경우에 속한다고 가정한다. 이때 사용자에게 패스워드를 재설정 하도록 요구한다.

1-3: 패스워드로 설정한 문자 값과 임계치를 이용해 설정한 시간의 장단을 이용해 새로운 패스워드를 생성한다.

```
/* 입력시간의 장단 설정

  (i≠h) Keystroke = (Si, Ti)
  Keystroke Sequence Set =
  ((S1,T1),(S2,T2),(S3,T3) · · · (Sn-1,Tn-1),(Sn,Tn))
   Si ∈ { 입력키 값 (Keyboard Characters) }
          Ti-Ei ≦ Ti ≦ Ti+Ei
          Th-Eh ≦ Th ≦ Th+Eh

while (Ti ≠ NULL)
  for each i ∈ T
      determine length of the time for Ti

    if (Tmax + Tmin/2 <=100) then

    // 입력시간의 장단을 설정하지 않았을 경우,

        Tlength ← 0

    else if (Th_min + Tl_max <=100) then

    // 예외처리

    else if (TThreshold < Ti) then

    // 임계치 보다 입력 시간이 클 경우,

        Tlength ← 1          // 시간을 '장'으로 인식

    else if (TThreshold >= Ti)

    // 임계치 보다 입력 시간이 작을 경우,
        Tlength ← 0          // 시간을 '단'으로 인식
    end if
  end for
end while
```

그림 5 시간 기반 패스워드 암호화 시스템

1-4: 기존에 입력한 아이디와 새롭게 생성된 패스워드를 데이터베이스 서버에 등록한다.

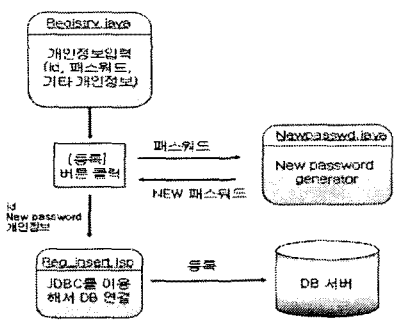

그림 6 사용자 등록 시스템

2단계: 사용자 인증

2-1: 사용자는 자바 애플릿이 실행되는 로그인 페이지를 통해 시스템에 등록한 아이디와 패스워드를 입력해 사용자 인증을 받는다. 패스워드를 입력할 때 시간의 상대적인 장단을 고려한다.

2-2: 패스워드의 입력을 마치면, 사용자 등록 시스템에서와 같은 방식으로 새로운 패스워드를 생성한 다음, 현재 데이터베이스에 저장된 패스워드와 일치하는지 확인한다.

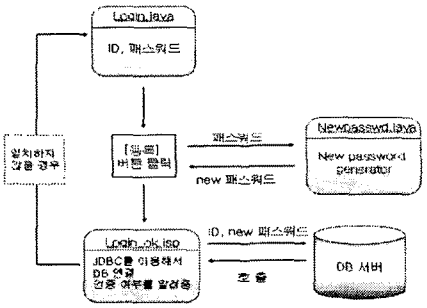

그림 7 사용자 인증 시스템

4. 결 론

본 논문에서는 기존의 패스워드 기반 인증 시스템에 시간의 개념을 도입해 패스워드를 암호화하는 방법을 제시하였다. 우리가 제안한 방법은 기존의 패스워드 기반의 사용자 인증 시스템 방식에 사용자가 패스워드 입력 시 입력 시간의 상대적인 장단만 고려해서 기존의 로그인 시스템을 개선 시켰다. 실제 사용자 인증 시스템을 구축해서 살펴 본 결과 웹기반 사용자 인증 시스템뿐만 아니라 사용자의 패스워드를 필요로 하는 다양한 분야에 적용이 가능하다는 걸 예측할 수 있었다.

앞으로 본 논문에서 제시한 시간기반 패스워드 암호화 방법을 다양한 시스템에 적용해서 테스트 할 예정이며, 알고리즘을 확장해서 현재 장·단으로 설정하는 입력 시간의 길이를 상·중·하로 나누어 설정하는 방법에 대해 연구할 예정이다.

[참 고 문 헌]

[1] D. McDonald and R. Atkinson, "One-Time Passwords In Everything (OPIE):Experience with Building and Using Stronger Authentication," Proceeding of the Fith USENIX UNIX Security Symposium, 1995

[2] D. Brent Chapman, Elizabeth D. Zwicky, "Building Internet Firewalls", O'REILLY, 1995

[3] http://www.kisa.or.kr/technology/sub4/password.htm

[4] IETF RFC 2289, "A One-Time Password System", 1998.02

[5] 민수홍, 김명은, 조동섭 "시간기반 패스워드 암호화 시스템", 정보처리학회, p.667, 2002.