시간을 이용한 효율적인 일회용 패스워드 알고리즘

박 중 길[†] · 장 태 주[††] · 박 봉 주[†††] · 류 재 철[††††]

요 약

사용자 고유번호와 패스워드 기반의 사용자 인증 메커니즘을 수행하는 네트워크 시스템 환경에서는 스니퍼 프로그램 등을 이용하여 불법 도청함으로써 쉽게 사용자 패스워드를 알아낼 수 있다. 이러한 불법적인 도청에 의한 패스워드 노출 문제를 해결하는 방법으로 일회용 패스워드, challenge-response 인증 방식이 유용하게 사용되며, 클라이언트/서버 환경에서는 별도 동기가 필요 없는 시간을 이용한 일회용 패스워드 방식이 특히 유용하게 사용될 수 있다. 그러나 시간을 이용한 일회용 패스워드 방식에서는 시간편차에 의한 인증 실패가 발생할 수 있다. 이 논문에서는 시간편차에 의한 인증이 실패할 가능이 있는 기간을 표시하는 1-비트 정보를 이용하여, 시간편차에 의한 인증 실패가 발생하지 않는 효율적인 일회용 패스워드 알고리즘을 제안한다. 제안된 알고리즘은 기존의 시간을 이용한 일회용 패스워드 시스템에 프로토콜의 변경 없이 쉽게 구현이 가능하다.

An Effective One-Time Password Algorithm Using Time

Joonggil Park[†] · Taejoo Chang[††] · Bongjoo Park[†††] · Jae-cheol Ryou[††††]

ABSTRACT

Almost all network systems provide an authentication mechanism based on user ID and password. In such system, it is easy to obtain the user password using a sniffer program with illegal eavesdropping. The one-time password and challenge-response method are useful authentication schemes that protect the user passwords against eavesdropping. In client/server environments, the one-time password scheme using time is especially useful because it solves the synchronization problem. However, it has a problem with associated time-slippage, a problem that causes the systems authentication to fail. In this paper, we propose an effective one-time password algorithm, which solves the time-slippage problem through the use of 1-bit information, which denotes duration in which authentication may be failed because of time-slippage. This algorithm is easily and quickly added to current one-time password systems that use time without requiring any change of protocols.

키워드 : 일회용 패스워드(one-time password), challenge-response, S/key, 시간편차(time-slippage)

1. 서 론

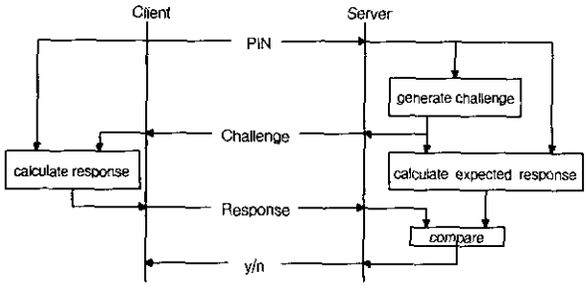
사용자 고유번호와 패스워드 기반의 사용자 인증 메커니즘을 수행하는 네트워크 시스템에서는 고정된 패스워드를 사용하기 때문에 패스워드 누출이 쉽고, 네트워크상에서 해커가 패스워드를 가로채는 경우에 방어할 방법이 없다. 실제로 국내외에서 발생한 대부분의 해킹 사건은 IP 주소 및 패스워드 도용을 통한 방법이었으며, 사용자 패스워드는 스니퍼 프로그램등을 통하여 불법 도청함으로써 쉽게 알아낼 수 있다. 이러한 패스워드 누출을 방지하는 기술은 사용자 인증 기술의 확보로 가능하며, 대표적인 사용자 인증 기술은 패스워드, 비암호학적인 방법, 암호학적인 방법으로 분류할 수 있다[1].

패스워드를 이용한 사용자 인증 기술은 대부분의 실제 인증 시스템에서 채택하고 있는 메커니즘이지만, 외부 노출, 추측, 도청, 재연등에 특히 취약하다. 비암호학적인 방법으로는 일회용 패스워드, challenge-response, 개인특성, 주소기반 등의 방식들이 있으며, 넓은 의미에서 challenge-response도 일회용 패스워드의 범주에 포함시킬 수도 있다. 그러나 challenge-response 방식이 수행하는 인증 프로토콜 패스가 2 패스로 일회용 패스워드 1 패스와 다르고, 기존의 실제 시스템에 구현 시 고려 사항들도 많이 상이함으로 일반적으로 다른 방법으로 분류된다[1]. 암호학적인 방법으로는 Kerberos 등의 관용키 방식과 X.509 등의 공개키 방식 등이 있으나 실제 시스템 구현에서는 키 관리 등에 많은 노력과 비용이 소요된다[2, 3].

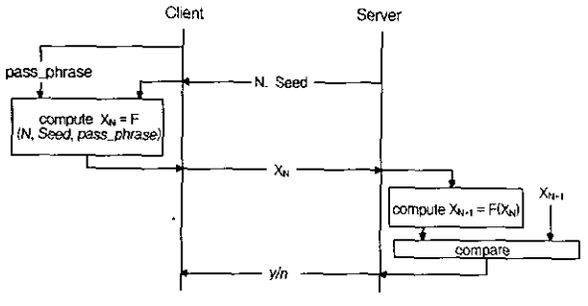
실제로 구현된 많은 사용자 인증 시스템은 패스워드 방식 또는 비암호학적인 방법 중에서 일회용 패스워드, challenge-

† 준 회 원 : 국가보안기술연구소 선임연구원
†† 정 회 원 : 국가보안기술연구소 책임연구원
††† 정 회 원 : (주)테크노밸리 책임연구원
†††† 종신회원 : 충남대학교 컴퓨터과학과 교수
논문접수 : 2001년 5월 18일, 심사완료 : 2001년 6월 25일

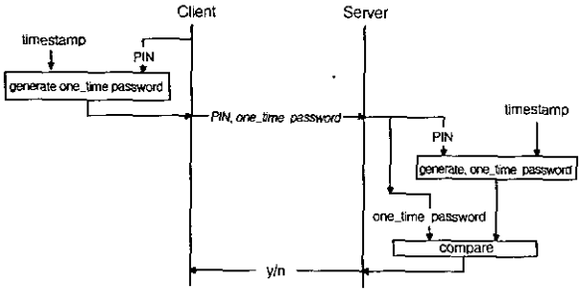
response 방식들이다. 이 방식들은 국내외적으로 많은 연구가 진행되어 왔으며[4-8], 현재 수십 종의 인증제품이 생산되어 다양한 응용 분야에 적용되어 사용 중에 있다[9, 10]. (그림 1~3)은 각각 challenge-response 인증 시스템, S/key 인증 시스템 및 시간을 이용한 일회용 패스워드 인증 시스템을 나타낸다.



(그림 1) Challenge-response 인증 시스템



(그림 2) S/key 인증 시스템



(그림 3) 시간을 이용한 일회용 패스워드 인증 시스템

비암호학적인 방법 중에서, 클라이언트가 생성한 인증 값을 서버에 전달하여 바로 인증을 수행하는 1 패스 인증 프로토콜인 일회용 패스워드 시스템은 기존의 사용자 고유번호와 패스워드 기반 인증 프로토콜에 아무런 변경 없이 적용이 가능하다. 즉 프로토콜에는 변화 없이 클라이언트 프로그램에 인증 값 생성 함수와 서버 프로그램에 검증 값 계산 함수의 추가로 쉽게 구현된다. 그러나 challenge-response 방식은 2 패스로 인증 프로토콜을 수행하며, 클라이언트 프로그램에 challenge를 받아서 response를 계산하는 함수와 서버 프로그램에 challenge 생성, 전송 및 response를 계산하는 함수가 구현되어야 한다. 따라서 기존의 사용자 고유번호 및 패스워드 기반의 인증 시스템에 비암호학적인 방법들을 적용하여 구현하는 경우에, 일회용 패스워드 방식이 훨씬 빠르고, 쉽게 구현될 수 있다.

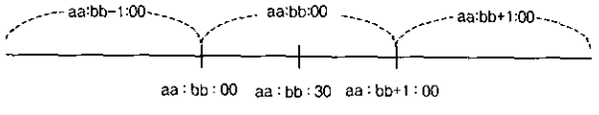
이러한 일회용 패스워드 인증 시스템을 구현하기 위한 대표적인 방법과 유지 관리해야 할 필요사항으로는 다음과 같다[1].

- Random 패스워드 목록 이용(패스워드 목록 내의 위치 동기화 필요)
- Pseudo-random sequence number 이용(sequence generator의 상태 동기화 필요)
- 시간 이용(시간의 동기화 필요)

많은 수의 클라이언트/서버 환경에서는, 위의 random 패스워드 목록과 sequence number 방법은 유지 관리에 많은 문제점을 유발하지만, 시간을 이용한 방법은 별 다른 유지 관리 문제 없이 사용할 수 있다[1].

2. 기존의 시간을 이용한 일회용 패스워드 알고리즘 및 문제점

시간을 이용한 일회용 패스워드 인증 시스템은 시간의 단위기간 내에서 대표값을 취하여 단위기간 내에서는 이 대표값을 인증 시스템에 입력 값으로 한다. 일반적으로 단위기간 내에서 대표값은 그 구간의 시작점으로 선택한다. 예를 들어, 1분마다 패스워드가 변경되는 인증 시스템은 (그림 4)와 같이 시각이 aa : bb : cc일 때는 그 대표값으로 aa : bb : 00을 취한다.



(그림 4) 분 단위 패스워드 인증에서 대표값

특히, 모든 단위기간이 일정할 때, 이 단위기간을 유효기간이라 부르며, 이 유효기간 동안은 동일한 대표값이 지속된다. 새로운 대표값이 시작되는 점을 변경점이라 하면 위 시스템의 유효기간은 1분이고 변경점은 매 0초가 되는 시점이다. 위의 시스템을 일반화하기 위하여 이 후 모든 시간은 초로 환산한 시간을 사용하며, 시간을 이용한 일회용 패스워드의 일반적인 알고리즘은 다음과 같다.

T_C : 클라이언트의 시간
T_S : 서버의 시간
$R(x)$: 시간 x를 초로 환산한 후 해당되는 대표값을 계산하는 함수
ID_C : 클라이언트 고유번호, Password : 클라이언트 패스워드
H : 일반적인 해쉬함수

<시간을 이용한 일반적인 알고리즘>

(클라이언트)
단계 C1 : T_C 읽음
단계 C2 : $R(T_C)$ 계산.
단계 C3 : $H(ID_C, Password, R(T_C))$ 계산.
단계 C4 : ID_C, $H(ID_C, Password, R(T_C))$ 전송

(서버)
단계 S1 : ID_C, $H(ID_C, Password, R(T_C))$ 수신
단계 S2 : T_S 읽음
단계 S3 : $R(T_S)$ 계산.
단계 S4 : ID_C의 패스워드 Password 읽음
단계 S5 : $H(ID_C, Password, R(T_S))$ 계산.
단계 S6 : $H(ID_C, Password, R(T_C))$, $H(ID_C, Password, R(T_S))$ 비교

위의 시간을 이용한 알고리즘에서 다음과 같은 3가지의 문제점이 존재한다[1, 10].

- 시간의 동기화(time synchronization) 문제
- 유효기간에서 동일 패스워드 재사용 문제
- 시간편차 문제

시간을 이용한 방식에는 인증 데이터를 일치시키기 위해서는 시간에 대한 동기가 보장되어야 하는데, 이 시간의 동기화 문제는 클라이언트와 서버가 지역적으로 멀리 떨어져 있는 경우에 발생하며, 이는 그리니치(Greenwich) 표준시간 프로그램을 클라이언트/서버 양쪽에 설치함으로써 별 무리 없이 해결이 될 수 있다[10].

유효기간(현재의 일회용 패스워드가 계속적으로 사용되는 기간)에서 동일 패스워드 재사용 문제는, 시간을 이용한 방식에서 유효기간 내에 클라이언트가 생성하는 일회용 패스워드가 항상 같게 되므로, 해커가 네트워크 상의 패스워드를 도청하여 유효기간 내에 재접속을 시도한다면, 해커가 사용자 인증에 성공하게 되어 인증 시스템에 문제를 야기한다. 이 문제는 사용자 별로 가장 최근의 인증 성공 시간을 유지함으로써 해결이 될 수 있다. 즉 한 번 인증 후에, 유효기간 내에 또 다시 접속 시도가 오면 서버는 가장 최근의 인증 성공 시간을 보고 유효기간이 끝나지 않았을 경우에 인증을 거부하도록 하면 된다.

그러나 시간을 이용한 방식에는 더욱 심각한 시간편차의 문제가 항상 존재한다[1, 10]. 클라이언트들과 서버들 간에는 교정 주기에 의한 시간편차와 클라이언트의 인증 데이터 계산시간, 그리고 전송시간 등에 의해 시간편차가 발생한다. 이 시간편차는 시간 동기를 보장하지 못하여, 인증이 제대로 수행되지 않는 기간을 발생시킨다. 인증 실패확률은 식 (1)과 같으며, 시간이 경과함에 따라 시간편차가 증가함으로, 점점 더 실패확률은 커져서 큰 문제가 된다(실패확률이 1보다 큰 경우에는 항상 인증 실패를 의미함).

$$\text{인증 실패확률} = (\text{시간편차} + \text{클라이언트 계산시간} + \text{전송시간}) / \text{유효기간} \quad (1)$$

일반적인 알고리즘에서 사용하는 해쉬함수는 계산 속도가 빠른 알고리즘을 사용하고 있으며, 전송시간이 시간편차에 비하며 적은 크기 일 때, 위의 인증 알고리즘에서 인증 실패확률을 줄이기 위해서는 시간편차가 적은 시계를 사용하고 유효기간을 확장하면 된다. 그러나 시간편차가 적은 시계를 사용하기 위해서는 많은 비용이 소요되며, 유효기간을 확장하는 것은 유효기간 내에 대표값을 반복 사용하는 것으로 인한 안전성에 위협을 받을 수 있다

일반적으로, 시간편차에 의한 인증 실패 시에 해결하는 방법으로는 3가지가 있다. 첫째는 별도의 조치를 취하지 않고, 시간편차에 의한 인증 실패 시에 다시 인증을 받도록 하는 것인데, 이는 인증 시스템의 성능 저하를 유발할 것이다. 둘째는 인증 실패 시에 서버에서 현재 인증 값 전후의 여러 값과 비교하여 인증을 수행하는 것이다. 이 방법은 사용자의 패스워드가 서버에서 여러 패스워드로 간주되어 비교 사용되므로, 그 만큼 비도가 떨어진다. 셋째는 인증 실패 시에 서버가 시간 값을 클라이언트에 재전송하여 다시 인증을 수행하는 것이다. 이 방법 또한 3 패스 인증이 되며 여전히 시간편차에 의한 인증 실패확률이 존재한다.

3. 시간을 이용한 효율적인 일회용 패스워드 알고리즘

이 논문에서는 시간을 이용한 일회용 패스워드 시스템 설계에서, 가장 큰 문제점으로 제시되고 있는 시간편차 문제를 해결할 수 있는 알고리즘을 제시한다. 각 클라이언트와 서버에 내장된 시계는 수정 진동자의 오차가 존재하며, 시계 교정 주기에 의해 오차 값은 변경 될 수 있다. 즉, 각 클라이언트와 서버 시계는 시계의 수정 진동자와 교정주기에 의한 오차를 갖게 된다(일반적으로 클라이언트는 일회용 패스워드 생성기라 불리는 토큰(token)을 이용한다). 이러한 오차로 인해, 내장 시계의 시간을 이용한 일회용 패스워드 인증 시스템을 사용하는 임의의 두 클라이언트/서버 간의 인증을 수행할 때, 다른 인증 값이 발생되며 인증 실패의 한 요인이 된다.

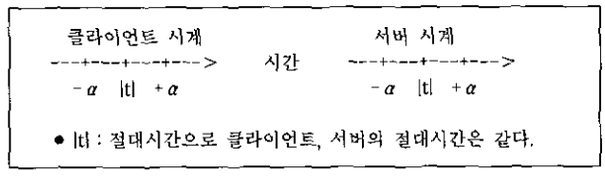
α를 각 내장된 시계의 최대오차라고 하면, α는 시계가 가진 수정 진동자의 최대오차와 컴퓨터의 시계 교정주기의 최대값에 의해 결정된다. (그림 5)에서 보여지는 것처럼 절대시각 $|t|$에 대해 클라이언트와 서버 시계의 최대오차가 $\pm\alpha$이므로, 클라이언트와 서버간의 시계 최대편차 T는 다음과 같이 구해진다.

$$T = 2\alpha \quad (2)$$

(예) 시계에 내장된 수정 진동자의 주파수 안정도가 e ppm(10^{-6})이고, 시계의 최대 교정주기가 일주일(7 일)인 경우에 최대오차(α)와 최대편차(T)는 다음과 같이 구한다.

최대오차(α) = 7 × 24(시간) × e ppm
= 7 × 24 × 3600(초) × e × 10^{-6}
= 0.6048 × e(초)
최대편차(T) = 2 × 최대오차
= 2 × 0.6048 × e(초)
= 1.2096 × e(초)

즉, 수정 진동자의 안정도가 10ppm 이내라고 하면, 최대 편차는 13초 이하이다.



(그림 5) 절대 시간에서 서버, 클라이언트 시간 최대편차

이제 기존의 시간을 이용한 인증 시스템에서 다음의 이론적인 배경을 도입하여 시간을 이용한 효율적인 인증 시스템을 구현하고자 한다.

$V \geq 2T$인 어떤 자연수 V, T에 대해서, 집합 $Z, N \rightarrow N$ 특성함수(characteristic function) $flag_{VT}(x)$, $N \rightarrow N$ 단계 함수(step function) $R(x)$를 다음과 같이 정의한다.

$$Z = \{x \in N \mid nV - T \leq x < nV + T \text{ for some } n \in N\}, \quad (3)$$

$$flag_{VT}(x) = \begin{cases} 1 & \text{if } x \in Z \\ 0 & \text{otherwise,} \end{cases} \quad (4)$$

$$R(x) = \lfloor x/V \rfloor \times V, \quad (5)$$

여기서 $\lfloor y \rfloor$는 y보다 크지 않는 최대정수를 의미한다.

보조정리 3.1 자연수 x, y, V, T에 대해서, $|x-y| < T$, $V \geq 2T$ 및 $flag_{VT}(x) = 0$을 만족하면, $R(x-T) = R(y)$가 성립한다.

증명 : $V \geq 2T$, $flag_{VT}(x) = 0$이므로, $nV + T \leq x < (n+1)V - T$를 만족하는 자연수 n이 존재한다. 따라서 $nV \leq x - T < (n+1)V - 2T$로 표기할 수 있으며 식 (5)의 정의에 의해 $R(x-T) = nV$이다. $|x-y| < T$이므로, $x - T < y < x + T$이고, $nV < y < (n+1)V$이다. 즉 $R(y) = n$. 그러므로 $R(x-T) = R(y)$이다. □

정리 3.1 자연수 x, y, V, T에 대해서, $|x-y| < T$, $V \geq 4T$ 및 $flag_{VT}(x) = 1$을 만족하면, $R(x-T) = R(y-2T)$이 성립한다.

증명 : $flag_{VT}(x) = 1$이므로, $x = nV + r$을 만족하는 적당한 $n \in N$과 $r (-T \leq r < T)$가 존재한다. 그러면, $R(x-T) = (n-1)V$이고 $|x-y| < T$이므로, $x - T < y < x + T$. $nV + r - T < y < nV + r + T$. $nV + r - 3T < y - 2T < nV + r - T$. 그러므로 $nV - V < y - 2T < nV$. $R(y-2T) = (n-1)V$. □

이 논문에서는 시간편차에 의한 인증이 실패할 가능이 있는 기간을 표시하는 별도의 1-비트를 추가 전송하여, 시간을 이용한 기존의 인증 시스템에서 발생하는 시간편차에 의한 인증 실패확률을 제거하는 알고리즘을 제안한다.

T : 클라이언트와 서버 시계의 시간 최대편차
V : 유효기간
Ttflag : 시간편차에 의한 인증이 실패할 가능이 있는 기간을 표시하는 1-비트 플래그

<시간편차 보정 알고리즘>

(클라이언트)
단계 C1 : T_C 읽음
단계 C2 : R(T_C) 계산.
단계 C3 : Ttflag = 0
단계 C4 : IF($T_C - R(T_C) < T$) then Ttflag = 1
IF($V - (T_C - R(T_C)) <= T$) then Ttflag = 1
단계 C5 : IF(Ttflag = 1) then $T_C = T_C - T$
단계 C6 : H(ID_C, Password, $R(T_C)$) 계산.
단계 C7 : ID_C, H(ID_C, Password, $R(T_C)$), Ttflag 전송

(서버)
단계 S1 : ID_C, H(ID_C, Password, $R(T_C)$), Ttflag 수신
단계 S2 : T_S 읽음
단계 S3 : IF(Ttflag = 1) then $T_S = T_S - 2T$
단계 S4 : $R(T_S)$ 계산.
단계 S5 : ID_C의 패스워드 Password 읽음
단계 S6 : H(ID_C, Password, $R(T_S)$) 계산.
단계 S7 : H(ID_C, Password, $R(T_C)$), H(ID_C, Password, $R(T_S)$) 비교

위 알고리즘의 인증 성공확률은 유효기간과 시계의 시간편차에 영향을 받는다. 보조정리 3.1과 정리 3.1에 의하여 $V \geq 4T$ 이면, 위의 시간편차 보정 알고리즘은 정상적인 사용자와는 항상 인증 가능하다.

그러나 위의 시간편차 보정 알고리즘은 전송시간을 고려하지 않아 언제나 인증을 성공하는 것은 아니다. 클라이언트 계산, 전송, 지연 그리고 토큰 입력 시간 등을 간략히 지연시간이라 하면 지연시간은 위 알고리즘의 안정성에 많은 영향을 준다. 이 지연시간이 일정하다면 효과적인 인증 알고리즘을 도출할 수 있지만 일정하지 않다면 인증 알고리즘 설계에 문제가 있다. 이 논문에서는 다음과 같은 변수 β, γ를 고려한 인증 알고리즘을 제시한다.

β : 클라이언트 계산, 전송 및 지연 토큰 입력 최소시간
γ : 클라이언트 계산, 전송 및 지연 토큰 입력 최대시간

<전송시간을 고려한 시간편차 보정 알고리즘>

(클라이언트)
단계 C1 : T_C 읽음
단계 C2 : $T_C = T_C + \beta$
단계 C3 : $R(T_C)$ 계산.
단계 C4 : Ttflag = 0
단계 C5 : IF($T_C - R(T_C) < T$) then Ttflag = 1
IF($V - (T_C - R(T_C)) <= T$) then Ttflag = 1
단계 C6 : IF(Ttflag = 1) then $T_C = T_C - T$
단계 C7 : $H(ID_C, Password, R(T_C))$ 계산.
단계 C8 : ID_C, $H(ID_C, Password, R(T_C))$, Ttflag 전송

(서버)
단계 S1 : ID_C, $H(ID_C, Password, R(T_C))$, Ttflag수신
단계 S2 : T_S 읽음
단계 S3 : IF(Ttflag = 1) then $T_S = T_S - 2T$
단계 S4 : $R(T_S)$ 계산.
단계 S5 : ID_C의 패스워드 Password 읽음
단계 S6 : $H(ID_C, Password, R(T_S))$ 계산.
단계 S7 : $H(ID_C, Password, R(T_C))$, $H(ID_C, Password, R(T_S))$ 비교

위 알고리즘은 최소 지연시간을 클라이언트에서 미리 더하여 계산하므로, 클라이언트 시간 입력값과 서버의 인증시간 차이를 줄일 수 있다. 그러나 지연시간의 변동폭 ($\gamma - \beta$)는 보정할 수 없다. 결론적으로, 클라이언트와 서버의 시계의 시간편차와 지연시간 변동폭의 합을 T라하고 $V \geq 4T$일 때, 위의 알고리즘은 시간을 이용한 일회용 인증 알고리즘으로써 매우 효과적이다.

4. 기존의 인증 알고리즘과 비교분석

제안한 알고리즘과 기존의 알고리즘과의 성능 분석의 결과는 <표 1>에 정리하여 표시하였다. 이 표에서 나타낸 것처럼, 제안한 알고리즘은 시계의 시간편차에 따른 인증 실패확률이 없고, 또한 지연시간에 따른 실패확률도 예측된 최대 지연시간 이내에서는 인증 실패가 존재하지 않는 효율적인 인증 알고리즘이다.

〈표 1〉 제안한 인증 알고리즘의 성능 분석

성능 평가 요소 / 인증 알고리즘	시계의 시간편차에 따른 인증 실패확률	기존 1 패스 시스템에 적용 가능성	인증시간	지연시간에 따른 인증 실패확률
기존의 시간을 이용한 알고리즘	편차/유효기간	가능함	빠름 (1 패스)	있음
Challenge - response	없음	불가능	느림 (2 패스)	없음
제안한 알고리즘	없음	가능함	조금 느림 (1 패스 + Δ)	예상 최대지연시간 내에서는 없음

Δ : Ttflag 값 설정시간

클라이언트마다 토큰 계산시간 및 전송시간 등 즉, 지연시간에 차이가 있는 경우에 기존의 인증 알고리즘들은 매우 불안정적이다. 그러나 사용자마다 정확한 최대 지연시간을 사전에 입력하여 이 논문에서 제안한 알고리즘을 사용한다면 이러한 문제점은 해결할 수 있으며, 사전에 시계의 시간편차의 정확한 한계치를 계산할 수 있다면, 매우 효율적인 인증 알고리즘을 구현할 수 있다.

5. 결 론

클라이언트/서버 환경에서 시간을 이용한 일회용 패스워드 인증 시스템을 구현 시에 시간편차에 의한 인증을 실패할 경우가 존재한다. 이 논문에서는 이러한 기존의 인증 시스템에서 인증 실패 기간에서 조차 인증 실패가 발생하지 않는 시간을 이용한 효율적인 인증 알고리즘을 제안하였다. 제안한 알고리즘은 일회용 패스워드 인증 시스템의 프로토콜 변화 없이, 단지 1-비트를 사용자 인증 정보에 추가함으로써 구현 가능하다. 그리고 이 알고리즘의 인증 실패확률은 클라이언트/서버의 최대 지연시간 범위 내에서는 없으며, 알고리즘이 간단하여 클라이언트/서버 양쪽에 빠른 수행시간을 제공한다. 그러므로 이 논문에서 제안한 알고리즘을 이용하여 일회용 패스워드 인증 시스템을 완전하고, 효율적으로 구현이 가능하다.

참 고 문 헌

[1] Warwick Ford, *Computer Communication Security*, Prentice Hall, pp.109-148, 1994.
[2] Bruce Schneier, *Applied Cryptography, Second Edition*, John Wiley & Sons, 1996.
[3] William Stallings, *Network and Internetwork Security*, Prentice-Hall, 1995.
[4] Neil M. Haller, "The S/Key (TM) one-time password system," Proc. Internet Society Symposium on Network and Distributed System Security, pp.151-158, 1994.
[5] A. Simizu, T. Horioka, and H.Inagaki, "A password authentication method for contents communication on the internet," IEICE Trans. Commun., Vol.E81-B, No.8, pp.1666-1673, Aug. 1998.
[6] 박중길, 김영진, 김영길, 백규태, 백기영, 류재철, "S/KEY를 개선한 일회용 패스워드 메커니즘 개발", 한국통신정보보호학회논문지, 제9권 제2호, pp.25-35, 1999.
[7] 박중길, 장병화, 장태주, "시간을 이용한 완전한 일회용 패스워드 알고리즘", 제12회 정보보호와 암호에 관한 학술대회논문집, pp.503-511, 2000.
[8] Manjula Sandirigama, Akihiro Shimizu, Matu-Tarow Noda, "Simple and Secure Password Authentication Protocol," IEICE Trans. Commun., Vol.E83-B, No.6, pp.1363-1365, June 2000.
[9] "RSA SecuID Authenticator," http ://www.securitydynamics.com/solutions/products/secuid/datasheets/dsauthenticators.htm.
[10] "일회용 패스워드 기술", http ://www.kisa.or.kr/technology/sub4/password.html.

박 중 길

e-mail : jgpark@etri.re.kr
1986년 동국대학교 전자계산학과 졸업
1988년 서강대학교 전자계산학과(석사)
1988년~2000년 국방과학연구소 선임연구원
2000년~현재 국가보안기술연구소 선임연구원
1998년~현재 충남대학교 컴퓨터과학과 박사과정 중
관심분야 : 컴퓨터통신보안, 접근통제, 암호이론

장 태 주

e-mail : tchang@etri.re.kr
1982년 울산대학교 전기공학과 졸업(공학사)
1990년 한국과학기술원 전기및전자공학과 졸업(공학석사)
1998년 한국과학기술원 전기및전자공학과 졸업(공학박사)
1982년~2000년 국방과학연구소 선임연구원
2000년~현재 국가보안기술연구소 책임연구원
관심분야 : 정보보호, 컴퓨터통신, 통계학적신호처리

박 봉 주

e-mail : bjpark@secugene.com
1986년 서강대학교수학과 졸업(이학사)
1988년 서강대학교 대학원 수학과 졸업(이학석사)
2000년 서강대학교 대학원 수학과 졸업(이학박사)
1988년~2000년 국방과학연구소 선임연구원
2000년~2000년 국가보안기술연구소 선임연구원
2000년~현재 (주)테크노밸리 책임연구원
관심분야 : 정보보호, 컴퓨터통신, S/W 및 H/W 고속 프로토콜

류 재 철

e-mail : jcryou@home.cnu.ac.kr
1985년 한양대학교 산업공학과(학사)
1988년 Iowa State University 전산학과(석사)
1990년 Northwestern University 전산학과(박사)
1991년~현재 충남대학교 컴퓨터과학과 부교수
관심분야 : 컴퓨터 및 통신망 보안, 전자상거래, 분산