한국산학기술학회논문지
Vol. 11, No. 9 pp. 3548-3557, 2010

바이오 정보를 이용한 사용자 인증 시스템 설계 및 구현

이형우[1*], 박영준[2]
[1]한신대학교 컴퓨터공학부, [2]청강문화산업대학 사이버정보보안과

A Design and Implementation of User Authentication System using Biometric Information

Hyung-Woo Lee[1*] and Yeong-Joon Park[2]
[1]**School of Computer Engineering, Hanshin University, Korea**
[2]**Dept. of Cyber Information Security, Chungkang College of Cultural Industries, Korea**

요 약 인터넷뱅킹 관련 보안사고가 급증하고 있고 피싱, 파밍 등 수법도 점차 지능화되고 있기 때문에 관련 보안 기술 개발이 시급하다. 인터넷 뱅킹 과정에서 온라인 사용자 인증 및 보안 성능을 강화하기 위해 인증서 사용과 함께 일회용 패스워드(One-Time Password: OTP)를 사용토록 하고 있다. 하지만, 기존의 OTP 기술은 MITM(Man-In-The-Middle) 공격에 극히 취약하고 상호 동기화를 맞춰야 한다. 따라서 기존 OTP 기반 시스템의 보안성을 높이기 위해서는 각 개인이 기본적으로 소유하고 있는 바이오 정보(Biometric Data)를 OTP 방식과 접목하는 방법을 사용할 수 있다. 따라서 본 연구에서는 바이오 정보에 변환함수를 이용하여 사용자의 바이오 ID로부터 일회용 바이오 템플릿을 생성하고 이를 통해 사용자 인증을 수행하는 시스템을 설계 및 구현하였다.

Abstract Security enhancement technologies are required to preventing phishing and pharming attacks on Internet banking. One-time password(OTP) should be used with certificate for enhancing user authentication and security performance. However, existing OTP technique is weak on MITM(Man-In-The-Middle) attack and synchnonization should be provided on OTP system. Therefore, more advanced mechanism such as combining biometic data with OTP can be suggested to enhancing security on authentication system. In this paper, we designed and implemented a multifactor authentication system using one-time biometric template to generate unique authentication data after adapting biometric transform on each user's biometric data.

Key Words : OTP, Multi-factor Authentication, Security, Biometric Information, Authentication System

1. 서론

최근들어 인터넷뱅킹 관련 보안사고가 급증하고 있는 것으로 나타났다. 피싱, 파밍 등 수법도 점차 지능화되고 있어 소비자들의 주의가 요망되며 관련 보안 기술 개발이 시급하다. 금융감독원에서 공개한 자료에 따르면 은행, 보험사에서 발생한 인터넷뱅킹 사고와 피해금액은 매년 급증하고 있는 것으로 집계되어 사회적 문제로 대두되고 있다. 범죄유형은 피싱이나 파밍을 통해 계좌번호와 비밀번호를 빼낸 후 비교적 검증절차가 간소한 인터넷상의 전자지불시스템을 활용하는 경우가 많았다. 따라서 이와 같은 신종 공격에 대한 능동적 대응 기술이 연구/개발되어야 한다.

보안카드 방식을 통해 인터넷 뱅킹 등을 수행하는 과정에서 사용자 인증을 수행하는 방안이 제시되었으나, ID/PW 등과 함께 보안카드를 이용하는 이중 인증(Dual Factor Authentication) 방안 역시 패킷 수집, 분석 및 키로그 분석/수집 등을 통해 공격자에게 노출될 수 있다. 또

이 논문은 2008년도 정부재원(교육인적자원부 학술연구조성사업비)으로 한국학술진흥재단의 지원을 받아 연구되었음. (KRF-2008-521-D00444).
*교신저자 : 이형우(hwlee@hs.ac.kr)
접수일 10년 07월 15일 수정일 10년 08월 04일 게재확정일 10년 09월 08일

한 인터넷을 통해 공개된 오픈소스 및 공개 소프트웨어를 통해 손쉽게 사용자의 인터넷 뱅킹 관련 정보를 수집/분석할 수 있어 심각한 문제를 유발하고 있다[1,2].

따라서 은행 등에서는 보안카드 방식과 함께 OTP(One-Time Password) 기술을 적용한 사용자 이중 인증 방안을 제시하고 있다. OTP 방식은 사용자에게 별도의 토큰 형태로 OTP를 배포하여 금융거래시 일회용 패스워드를 기입하도록 하고 있다. OTP 기술을 적용하여 일회용 패스워드 형태로 작동하기 때문에 OTP 방식은 기존의 보안카드 보다 보안성을 높일 수 있다. 하지만 OTP 기술을 이용할 경우 기존의 보안카드 보다 개선된 인증 체계를 제공할 수 있으나, 완전한 사용자 인증 및 확인 기능을 제공하지는 못하고 있다.

기존 OTP 기술의 문제점을 분석해 보면, 보안성이 강화되었으나 아직도 완전한 사용자 인증/확인 기능을 제공하지 못하고 있다. OTP 서버와의 동기화 여부/동기 설정 과정에서 오류 발생 및 인증 성능 저하 문제 발생하고 있으며, 인증 서버와의 OTP 통신 역시 MITM(Man-in-the-Middle) 공격에 무방비로 노출되어 있다. 그리고, 분실 및 도난시 사용자 인증/확인 과정에 취약한 문제점 발생한다. 항상 별도로 휴대해야 한다는 문제점이 발생하고 은행별로 개별 소지해야 하는 문제 및 구입비용 등이 높다는 문제점 등이 발생한다. 또한 OTP 분실/도난시 사용자 인증 확인이 불가능하다는 문제점이 발생한다. 결국 도난된 OTP 토큰에 대한 OTP 검증 모듈/기능등이 제시되지 않아 손쉽게 인증 과정을 통과한다는 문제점이 발생하여 이에 대한 대응 기술이 개발되어야 한다[3-6].

결국, 바이오 ID 정보를 통한 사용자 인증 및 OTP 생성 기술에 대한 연구가 필요하다. 현재까지 연구 개발된 기존 OTP 기술의 문제점을 보완하고 사용상 안전성을 확보하기 위해 바이오 정보와 연계된 다중 인증 체계 개발이 필요하며, 이를 위해서는 일회용 바이오 ID 생성 기술이 개발되어야 한다. 또한 바이오 정보 관련 프라이버시 보호 기능을 제공하기 위해 일회용 바이오 ID를 기반으로한 Biometric OTP 메커니즘이 필요하다. 따라서 본 연구에서는 바이오 정보 노출시 문제점을 해결하기 위해 일회용 바이오 템플릿 정보를 생성하고, 이를 OTP 생성 과정과 연계하여 사용자 인증/다중 인증성을 제공하는 Biometric OTP 기법을 제시하고자 한다.

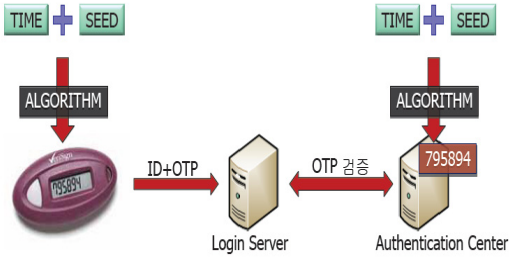
본 논문의 2장에서는 기존의 안전한 로그인을 위한 기법 관련 연구들과 그 문제점들을 제시한다. 3장에서는 기존의 문제점들을 보안하기위한 모델을 제시하며, 4장과 5장에서는 바이오 정보 기반 제안 모델 접목 방식과 구현 결과 및 성능평가를 제시하였고, 본 모델의 장점과 안전한 로그인을 위한 앞으로의 연구 방향을 제시한다.

2. 기존 OTP 기술의 문제점 분석

2.1 기존 OTP 생성 기술

OTP(One-Time Password) 기술은 일회용 패스워드 생성 기술로 매번 다른 패스워드로 사용자를 인증하기 위한 기술로, 현재의 패스워드로부터 다음에 사용할 패스워드를 유추하는 것이 수학적으로 불가능하도록 작동된다. 기존의 일반적인 Static Password는 취득후 재사용가능한 방식으로, 키로거/네트워크 스니퍼/사전공격 및 Brote Force 공격 등으로 인해 공격자에 의해 정보 획득이 가능하다는 문제점이 발생한다. 따라서 OTP 방식은 매번 다른 패스워드를 생성하기 때문에 취득한 값을 재사용할 가능성이 희박하다는 장점을 갖는다[1].

OTP 생성 메커니즘은 아래 그림 1과 같이 입력값에 대해 OTP 생성 알고리즘을 이용하여 OTP 값을 추출하고 이를 이용하여 Two-Factor 인증 과정에 활용하게 된다.



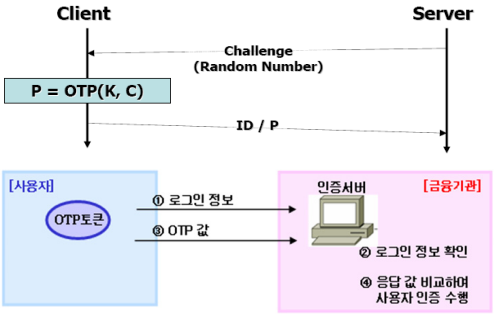
[그림 1] OTP 생성 시스템 구조

- o 입력값 : OTP 생성 알고리즘의 입력 데이터 정보[시간/이벤트 및 비밀키 등]
- o OTP 생성 알고리즘 : 입력값으로부터 OTP 값을 생성하는 알고리즘으로 일방향 해쉬함수, 대칭키 암호화 알고리즘 등을 이용
- o OTP 값 추출 함수 : Truncation function으로 OTP 생성 알고리즘을 통해 출력된 값으로부터 실제 OTP 값으로 사용할 6~8자리 숫자를 추출하는 알고리즘
- o OTP 기반 Two-Factor 인증 기법을 제공 : HW PIN[Pin 번호 기억 및 OTP 기기 소유] 정보를 통해 인증을 수행하거나, SW PIN 등을 통해 인증

2.2 기존 OTP 방식의 문제점

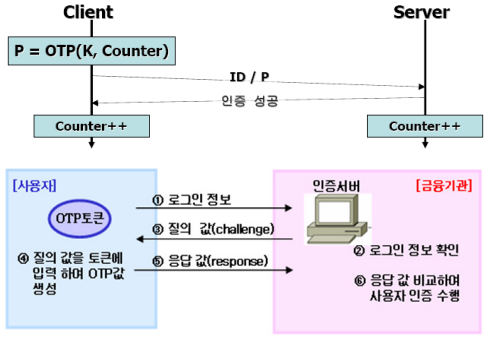
OTP 생성 기술은 그림 2, 그림 3과 같이 크게 비동기화 방식과 동기화 방식으로 나눌 수 있다. 비동기화 OTP

방식(Asynchronous OTP)은 Challenge-Response 방식으로 작동하는 것으로 질의값에 대한 응답 결과로 인증과정을 수행한다. 비동기 OTP 방식은 서버와 동기화가 필요 없으나, 사용자 입력이 불편하고 네트워크 부하가 증가하며, ID/PW 기반 프로토콜/어플리케이션과 호환이 용이하지 않다는 문제점이 발생한다.



[그림 2] 비동기화 OTP 방식

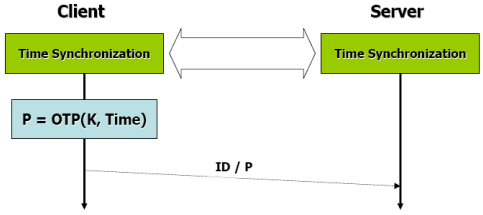
동기화 OTP 방식(Synchronous OTP)은 Time-Sync 방식, Event-Sync 방식 및 Time-Event-Sync 방식으로 구분할 수 있으며, OTP 토큰과 인증 서버 사이에 동기화되는 기준값에 따라 OTP 값을 생성하는 방식이다.



[그림 3] 동기화 OTP 방식

(1) 시간 동기화 OTP 방식의 문제점 분석

서버와 OTP 기기간 동기화된 시간 정보를 기준으로 특정 시간간격[보통 1분]마다 변하는 패스워드를 생성하는 방식으로, MITM 공격에 취약하며, 재사용 시간 제약[보통 1분]이 있다. 따라서 아래 그림 4와 같이 특정 시간동안 OTP 값이 바뀌지 않아서 일정 시간동안 MITM 공격이 가능하다. 서버와 OTP 기기간 시간정보 동기화 문제가 발생하기 때문에 일반적으로 오차 범위를 설정하여 인증을 수행하고 있으나, 인증 실패시 재시도를 위해 기다리는 불편 및 단점이 발생한다. 특정 시간동안 입력을 하지 못할 경우 중간에 패스워드가 서버에 의해 다시 생성되어 패스워드를 다시 생성해야 하는 문제점이 발생하는 등 실제 사용시 문제점이 발생한다.



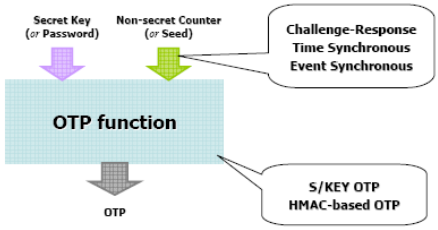
[그림 4] OTP 시간동기화 방식

(2) 이벤트 동기화 OTP 방식의 문제점 분석

서버와 OTP 기기가 동일한 카운트 값을 기준으로 패스워드를 생성하는 방식으로, OTP 입력값으로는 카운터[이벤트 회수], 서버와 OTP 기기간에 공유된 비밀키 정보를 입력하는 방식이다. 이 방식은 오프라인 공격시 시간 동기화 방식보다 취약하며, MITM 공격으로 의미있는 OTP 값을 공격자가 획득할 수 있으며, 재사용 공격에도 매우 취약하다. 또한. 서버와 OTP 간의 카운터 값을 동기/초기화 시켜야 하며, 인증서버에서 카운터의 오차범위 설정에 따라 성능/안전성에 많은 차이가 발생한다는 문제점이 발생한다.

(3) 이벤트-시간 동기화 OTP 방식의 문제점 분석

이 방식은 아래 그림 5와 같이 Time-Sync 방식과 Event-Sync 방식의 장점을 조합한 방식으로, 특정 시간간격[보통 24초~30초]마다 패스워드가 생성되며, 같은 시간간격 내에서 재시도시에는 카운트 값을 증가시켜스 패스워드가 변하도록 하는 방식이다. 입력값으로는 시간값, 카운터, 서버와 OTP 기기간에 공유된 비밀키 정보를 입력하게 된다. 하지만 이 방식 역시 MITM 공격이 가능하다. 서버와 OTP 기기간 시간정보 동기화 과정이 필요하며, 인증 실패시 인증 재시도를 위해 기다려야 하는 불편함이 있고, 특정 시간동안 입력하지 못할 경우 중간에 패스워드가 바뀌어 다시 입력해야 하는 불편함이 있다.



[그림 5] OTP 시간/이벤트 동기화 방식

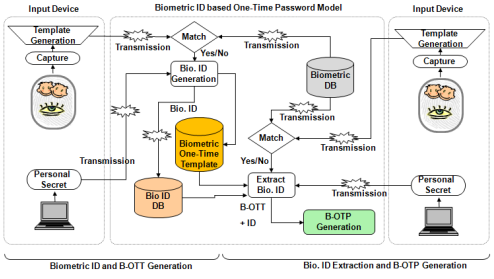
2.3 해결방안 제시

기존 OTP 방식은 OTP 토큰에 대한 실제 소유자에 대한 확인 과정이 전혀 제공되지 못하고 있다. OTP 토큰 발급시 비밀키 정보를 이용하여 패스워드를 생성하고 있으나, 실제 사용자에 대한 정보를 포함하고 있지 않아 제 3 자에 의해 사용시 이를 확인할 수 있는 방법이 제공되지 못하고 있다. 또한 OTP 정보에 대해 MITM 공격이 가능하기 때문에 이를 능동적으로 보완할 수 있는 방법이 기술적으로 제시되어야 한다.

그리고, OTP 토큰에 대한 분실시 비동기화/동기화 방식 모두 분실/도난시 대응 방안을 제시하지 못하고 있다. OTP 토큰이 분실되었을 경우 원래 소유주에 대한 확인 과정이 전혀 제공되지 않고 있다. 결국 현재까지 제시된 OTP 방식은 단순히 일회용 패스워드를 생성하는 과정에만 목적을 두고 있을 뿐, OTP 기기 및 모듈에 대한 소유자 인증/확인 과정이 제공되지 못하고 있다. 따라서 이에 대한 해결방안으로는 OTP 토큰 실제 소유자 인증을 위해 일회용 바이오 템플릿을 이용하여 소유자 확인 기능을 제공할 필요가 있다. 또한 일회용 템플릿으로부터 OTP 값을 생성하여 인증에 활용하는 방식을 제시하고자 한다. 이와 같은 방식을 사용하게 되면 바이오 정보 사용시 발생하는 프라이버시 문제도 해결할 수 있으며, OTP 정보 기반 다중 인증에도 활용 가능하다는 장점이 있다.

3. 제안 모델

바이오 정보는 변치 않으면서 개인별 고유한 특성을 지니고 있어 사용자 인증 용도로 사용 가능하다[7-9]. 실제로 지문/홍채 등의 바이오 정보를 대상으로 인터넷 기반 서비스와 연계하는 방법이 일부 구축되어 있다. 따라서 본 연구에서 제시하는 방식은 다음 그림 6과 같은 구조를 갖는다.



[그림 6] 제안 시스템 구조

o 바이오정보 기반 ID 및 일회용 템플릿 생성 : 각 사용자로부터 바이오 정보를 입력받아 이를 디지털 값으로 변환한다. 변환된 디지털 값을 이용하여 일회용 바이오 템플릿을 생성하고 바이오 정보 관련 프라이버시 문제를 해결할 수 있다.

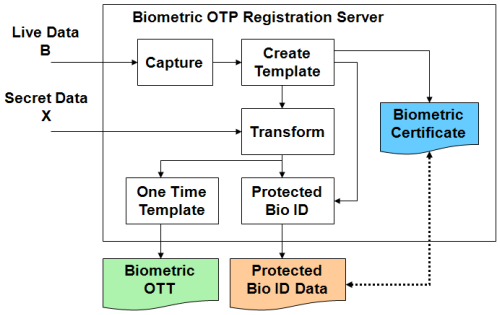
o 일회용 템플릿 기반 OTP 생성 : 일회용 템플릿으로부터 OTP 값을 생성하여 다중 인증 과정에 활용한다.

3.1 바이오 정보 기반 일회용 템플릿 생성

사용자로부터 입력된 바이오 정보를 입력받아 이를 디지털 값으로 변환한다. 본 연구에서 사용한 바이오 정보는 지문 정보를 대상으로 하였다. 지문 정보에 대해 일반적인 지문인식 장비를 통해 입력받는다.

다음 그림과 같이 사용자로부터 자신의 바이오 정보인 'B' 값을 입력하고 동시에 비밀정보 'X'를 입력한다. 이때 입력된 바이오 정보 'B'는 지문 정보와 같은 각 개인별 고유한 바이오 정보를 입력하는 과정이다.

예를들어 입력된 지문 정보에 대해서는 CBEFF 형태에 해당하는 포맷으로 변환되어 일회용 바이오 템플릿을 생성하고, 입력된 비밀 정보 'X'를 이용하여 이를 변환하고 서버는 이를 저장하게 된다. 생성된 일회용 바이오 템플릿 정보는 X.1089에 정의된 바이오인증서 형태로 생성되며, 바이오 ID 값은 토큰 형태로 USB 등과 같은 별도의 저장장치에 안전하게 저장된다. 전체적인 구조는 다음 그림 7과 같다.



[그림 7] 제안하는 바이오 템플릿 생성 구조

바이오 정보로부터 일회용 템플릿을 생성하고 이를 통해 다시 사용자에게 일회용 토큰을 할당해 주어 이를 통해 OTP 생성 과정에 사용할 수 있도록 적용하는 방법이다.

사용자로부터 입력된 바이오 정보로부터 다음과 같은 단계를 수행한다.

1단계 : 바이오 특징점 추출

2단계 : 특징점 등에 대한 변환함수 적용

3단계 : 개인 식별과정 수행 및 OTT 생성 과정 수행

구체적으로 일회용 바이오 템플릿(OTT)을 생성하기 위해 OTT를 생성하기 위해서는 다음과 같은 과정을 수행한다.

1단계 : 각 개인별 바이오 정보로 구성된 템플릿 데이터를 입력받음

2단계 : 템플릿 정보에 대해 OTT를 생성하는 과정을 수행함

3단계 : 템플릿 생성 관련 Secret 정보에 해당하는 변환정보(Transform Information)을 저장

바이오 정보는 벡터 정보로 구성되어 있으며 Euclidean Distance 함수를 이용하여 매칭 과정을 수행하고, 다음과 같은 함수를 적용하게 된다. $\mathbf{x}$ 는 각 개인별 바이오 정보를 의미하고, 임의의 난수 형태에 해당하는 Othogonal matrix $\mathbf{A}$ 와 난수 벡터 $\mathbf{b}$ 를 이용하여 $\mathbf{g}$ 값을 생성하게 된다.

$$\mathbf{g} := \mathbf{Ax} + \mathbf{b}$$

이제 위와 같은 과정을 수행하고 원본 바이오 템플릿에 해당하는 $\mathbf{x}$ 를 저장하지 않으면서도 인증 시스템에서는 $\mathbf{g}$ 값만을 저장하여 안전한 인증 과정을 수행하게 된다. 물론 $\mathbf{A}$ 행렬과 $\mathbf{b}$ 값을 사용자가 설정하는 비밀값에 해당한다. 이는 사용자 개인적으로 안전하게 저장하게 되며 일종의 토큰 형태로 저장되는 정보이다.

변환함수에 의해 생성된 결과에 대해 검증하는 과정은 다음과 같다. 사용자가 자신의 바이오 정보를 검증 과정에서 다시 입력하였을 경우 이를 $\mathbf{y}$ 라고 하면, 사용자는 자신만이 알고 있는 토큰값 $\mathbf{A}$ 와 $\mathbf{b}$ 값을 이용하여 다음과 같은 $\mathbf{p}$ 값을 생성하게 된다.

$$\mathbf{p} := \mathbf{Ay} + \mathbf{b}$$

사용자는 바이오 정보인 $\mathbf{y}$ 를 전송하는 대신에 위 식에서 생성된 $\mathbf{p}$ 값을 전송하게 된다. 그러면 해당 시스템에서는 다음과 같은 확인 과정을 수행하게 된다.

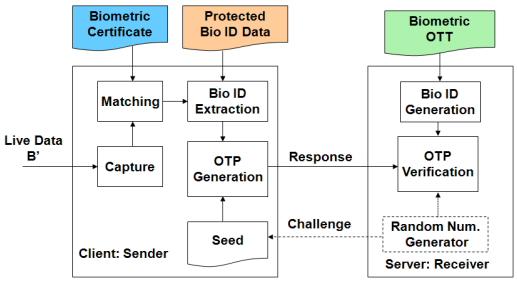
$$\begin{aligned}\|\mathbf{g}-\mathbf{p}\|^2 &= (\mathbf{g}-\mathbf{p})^T(\mathbf{g}-\mathbf{p}) \\ &= (\mathbf{Ax}+\mathbf{b}-\mathbf{Ay}-\mathbf{b})^T(\mathbf{Ax}+\mathbf{b}-\mathbf{Ay}-\mathbf{b}) \\ &= (\mathbf{Ax}-\mathbf{Ay})^T(\mathbf{Ax}-\mathbf{Ay}) \\ &= (\mathbf{x}-\mathbf{y})^T\mathbf{A}^T\mathbf{A}(\mathbf{x}-\mathbf{y}) \\ &= \|\mathbf{x}-\mathbf{y}\|^2\end{aligned}$$

만일 사용자가 잘못된 $\mathbf{A}$ 와 $\mathbf{b}$ 값을 입력하였다면 $\mathbf{g}$ 와 $\mathbf{p}$ 의 Euclidean Distance 값은 $\mathbf{x}$ 와 $\mathbf{y}$ 의 Distance 값보다 커지게 되므로 인증 과정을 통과하지 못하게 된다. 따라서 원본 바이오 정보 대신에 $\mathbf{A}$ 와 $\mathbf{b}$ 값 기반 변환함수를 이용하여 일회용 토큰 정보 $(\mathbf{A}, \mathbf{b})$ 값을 생성하게 되고 이를 이용하여 OTP 기반 인증 과정에 적용할 수 있다.

3.2 일회용 템플릿 기반 OTP 생성 단계

바이오 인증서와 일회용 바이오 템플릿을 생성한 후에, OTP 값을 생성하는 과정은 다음과 같다. 우선 다중 인증이 필요할 경우 사용자는 자신의 바이오 정보 'B''을 입력한다. 그러면 앞서 생성된 바이오 인증서를 통해 바이오 정보에 대한 매칭 과정을 수행하고 만일 동일 사용자라는 것이 확인되었다면 바이오 인증서로부터 바이오 ID 정보를 추출한다.

추출된 바이오 ID 정보를 이용하여 서버로부터 전송된 Challenge 값을 이용하여 OTP 값을 생성하고 이를 서버로 전송한다. 서버에서는 일회용 바이오 템플릿으로부터 추출된 바이오 ID 값을 이용하여 아래 그림 8과 같이 클라이언트로부터 수신된 OTP 값에 대한 검증 과정을 수행한다.



[그림 8] 바이오 템플릿 기반 OTP 구조

$\mathbf{g}_0$ 은 초기 등록 과정에서 저장된 템플릿을 의미하며 $\mathbf{p}_0$ 를 등록 과정 수행후 사용자가 인증을 요청한 처음 형태의 템플릿이라고 할 경우, 다음과 같이 각각의 값은 계산된다.

$$\mathbf{g}_0 := \mathbf{A}_0\mathbf{x} + \mathbf{b}_0,$$

만일 사용자가 n 번째 OTP 기반 인증 요청 과정을 수행할 경우, 인증서버에는 $\mathbf{g}_n$ 및 사용자 변환함수 토큰 정보 $\mathbf{A}_n$, $\mathbf{b}_n$ 를 저장하게 된다. 또한 사용자는 자신이 소유한 OTP 토큰에 저장된 $\mathbf{A}_n$ 및 서버로부터 전송된 도전

값(challenge) $\mathbf{b}_n$ 값을 이용하여 n 번째 인증 요청에 해당하는 $\mathbf{p}_n$을 다음과 같이 생성하고 이를 인증 서버에 전송하게 된다.

$$\mathbf{p}_n := \mathbf{A}_n\mathbf{y} + \mathbf{b}_n$$

그러면 인증서버는 $\mathbf{p}_n$과 앞서 자신의 서버에 저장된 $\mathbf{g}_n$ 값에 대해 바이오 검증 과정을 수행하게 된다. 따라서 앞에서 제시한 바와 같이 두 템플릿에 대한 인증 확인 과정은 다음과 같이 수행하게 된다.

$$\|\mathbf{g}_0 - \mathbf{p}_0\|^2 = \|\mathbf{x} - \mathbf{y}\|^2$$

이제 n번의 인증 과정을 수행하게 되면, $\mathbf{p}_n$과 $\mathbf{g}_n$은 다음과 같이 계산되며 인증 확인 과정에 대해서도 다음과 같이 검증 가능하다.

$$\mathbf{g}_n := \mathbf{A}'_{n-1}\mathbf{g}_{n-1} + \mathbf{b}'_{n-1},\ \mathbf{p}_n := \mathbf{A}_n\mathbf{y} + \mathbf{b}_n,$$

$$\|\mathbf{g}_n - \mathbf{p}_n\|^2 = \|\mathbf{x} - \mathbf{y}\|^2$$

결국 바이오 정보를 이용하여 사용자에 대한 OTP 값을 생성하고 이를 이용하여 인증 과정에 적용하게 된다.

4. 시스템 구현

4.1 환경설정

4.1.1 시스템 구조

안전한 로그인을 위한 보안카드 기반 인증시스템은 웹 서버 및 인증 서버로 구성이 되며, 보안토큰 발급 및 보안토큰기반 로그인 모듈로 구성된다. 기본적으로 작동하는 순서는 다음과 같다. 일반적으로 홈페이지에서 제공되는 로그인 관련 홈페이지를 Login.html 파일이라고 가정하였을 경우 사용자가 입력한 ID/Password 정보는 확인 과정에 해당하는 Log_ID_Process 모듈로 전송되어 ID/Password에 대한 정보와 사전에 기록된 정보를 확인하고, 이를 통해 소프트 형태의 보안카드 모듈로 전달된다. 보안카드 모듈에서는 사용자에 대한 확인 과정을 수행하여 최종적인 다중 인증 과정을 수행하게 된다.

가입 및 일회용 토큰 발급 단계는 다음과 같다. 웹 시스템을 기반으로 작동하는 로그인 시스템일 경우 사용자에 대한 회원가입 과정을 수행하며, 핸드폰과 연계하여 일회용 토큰 발급 요청 과정을 통해 일차적으로 사용자에 대한 인증 과정을 수행하고 핸드폰 내에 일회용 토큰 정보를 생성하고, 키 값을 발급받게 된다.

일반적으로 기존의 회원 가입의 절차와 유사하며 필요조건으로서 핸드폰 번호와 생년월일 정보를 입력하여야 한다. 처음 로그인시 ID와 생년월일을 입력하며 입력된 ID와 생년월일이 일치할 경우 보안카드와 비밀번호를 입력하게 된다. 보안카드의 경우 핸드폰을 통해 보안 카드가 발급되기 때문에 핸드폰 번호는 필수 입력 사항이 된다. 또한 부가적으로 장치를 통해 각 개인별 바이오 정보에 대해서 입력/전송하는 과정을 수행하게 된다.

다음 의사코드는 회원 가입을 위한 코드이다. Member_Information 이라는 회원 정보를 위한 구조체 변수가 선언이 되며 필수정보 입력을 판단하기 위한 Result 변수가 선언된다.

```
/* 입력 : 사용자 정보
출력 : 회원가입 신청결과 */
Bool Join(){
    Struct Member_Information Join_Member;
    Bool Result;
    Input( Join_Member.ID,
           Join_MemberPassword,
           Join_Member.Birthday,
           Join_Member.PhoneNumber
           etc...);
    Result=Check_Blank( Join_Member.ID,
                        Join_MemberPassword,
                        Join_Member.Birthday,
                        Join_Member.PhoneNumber);
    return Result;
}
```

Input()은 사용자로부터 정보를 입력받는 함수이며 파라미터는 회원 가입시 입력되는 정보들이다. Check_Blank()는 필수정보의 확인을 위한 함수이다. Check_Blank()의 파라미터에는 필수정보인 ID, Password, Birthday, PhoneNumber를 전달한다.

4.1.2 일회용 토큰 발급

연결된 핸드폰은 프로그램뿐만 아니라 보안카드 번호를 생성하기 위해 개인 비밀(Secret)에 기초하여 마스터 키(Master Key)값을 생성하고 이를 전송하게 된다.

```
/* 입력 : ID, PhoneNumber
출력 : Message, Key */

void Send_Message(ID,PhoneNumber){
    Master Key = Key_Generate(ID, Secret);
    Send(PhoneNumber,Master Key);
}
```

보안카드를 발급하는 모듈에서는 ID와 핸드폰 번호를 입력받으며 Key 생성을 위한 변수를 선언한다. ID에 따라 고유의 키가 Key_Generate()를 통해 생성이 되며 생성된 Key는 핸드폰번호와 함께 Send모듈로 전송된다. Send 모듈은 핸드폰으로 보안카드 프로그램과 생성된 고유의 키를 전송한다.

4.1.3 로그인을 위한 일회용 바이오 토큰 생성

로그인을 하기 위해서는 인증을 위해 일회용 패스워드가 필요하기 때문에 보안카드 기반 인증 과정을 수행해야 한다. 이때 사용자로부터 입력된 바이오 데이터로부터 보안카드 번호를 생성하게 된다. 각 사용자로부터 입력된 바이오 정보로부터 해쉬 함수를 적용하여 일정한 크기의 출력값을 얻게 된다. 보안카드 번호는 4자리 숫자로 이루어지며 30개의 번호가 생성된다. 이 번호들은 여러 방식을 이용하여 마스터 키(Master Key)를 생성하게 되며 보안카드내 생성된 번호들은 겹치지 않도록 생성한다.

본 시스템에서는 핸드폰에 프로그램 설치시 받은 고유의 Key값을 가지고 120자리의 Key를 생성하며 생성된 Key값을 4자리씩 토큰화 시켜 30개의 카드번호를 생성할 수 있으며, 결국 각 개인별 바이오 정보를 이용하여 마스터 키 값을 생성할 수 있다.

```
/* 입력 : Bio_Data
출력 : CardNumber[] */

void Create_CardNumber(Bio_Data){
    String CreateKey;
    int CardNumber[30];
    CreateKey=Hash(Bio_Data);
    CardNumber[]=Token(CreateKey);
}
```

위 코드는 핸드폰에 들어가는 플랫폼 기반의 프로그램 중 일부분이다. 회원가입시 전송받은 고유의 Key 값이 파라미터로 전달이 되며 고유의 Key 값을 통해 생성될 값이 CreateKey로 선언된다. CardNumber[]는 생성된 카드번호를 저장할 배열변수이다. 입력된 고유의 Key 값은 Hash()함수를 통해 120자리의 번호가 생성이 되어 CreateKey로 반환된다. 반환된 CreateKey는 Token()함수를 통해 4자리씩 나누어 CardNumber[]로 저장되게 된다.

```
Result_key = hash(Bio_Data)
token[4, 30] = Result_key
card[30]=int(token[30])
```

이렇게 발급된 숫자의 경우 총 94개의 경우를 가지기 때문에 단순 입력이나 추측으로는 예측하기 어렵다.

이와 같은 방식을 적용하여 기존 로그인 방식의 보안 취약성을 보완하기 위해 본 연구에서 제시하는 기법의 전체적인 구조는 다음 그림과 같으며, 각 단계별 세부 과정을 통해 로그인 과정에서의 다중 인증을 통한 보안성을 향상시키고자 하였다.

일회용 토큰을 핸드폰 내에 포함시켜서 작동시키는 것은 인증서버를 중심으로 인증 과정을 거치는 방법보다 개인 보안성을 높일 수 있으며, 핸드폰 자체의 사용자 확인 기능에 부가적으로 보안카드 체계를 접목할 수 있기 때문에 보다 안전한 다중 인증 방식을 구축할 수 있다는 장점이 있다.

4.2 사용자 다중 인증 시스템

4.2.1 바이오 ID를 통한 로그인

각 개인별 ID 정보와 함께 바이오 ID를 입력하도록 한다. 본 연구에서는 각 사용자 고유의 바이오 정보 중에서 얼굴 정보를 대상으로 실험하였다. 사용자 얼굴 정보에 변환함수를 적용하여 변형된 템플릿을 생성하고 이로부터 각 사용자의 바이오 ID 값을 추출하였다. Password를 먼저 입력하지 않는 이유는 Password를 스니핑으로 부터 보호하기 위해서 이다. 앞에서 제시한 변환 구조를 이용하여 바이오 정보로부터 바이오 ID 정보를 변환 생성하고 이를 이용하여 인증 과정에 적용하게 된다. 기타 로그인 부분은 기존의 사이트에서 그대로 사용할 수 있게 하기 위하여 큰 변화를 주지 않았다. 기존의 회원제 서비스를 제공하는 사이트의 로그인 모듈을 약간의 수정으로 사용할 수 있다.

```
/* 입력 : Boi_ID
출력 : 1차로그인결과 */

Bool Login_Bio_ID(){
    String Boi_ID, Template;
    Bool Result;
    Input( Bio_ID);
    Result = Compare( Bio_ID, Template);
    return Result;
}
```

위 코드는 바이오 ID 정보를 기반으로 사용자 인증 과정을 수행하게 된다. Input()함수를 통해 입력받은 정보와 데이터베이스에 저장된 값을 비교하고 그 결과는 Result에 저장되어 반환된다.

4.2.2 보안카드를 통한 로그인

보안카드를 통한 로그인페이지는 1단계 신원 확인 과정이 성공할 경우 수행된다. 바이오 정보로부터 생성된 보안카드의 번호를 입력하는 화면은 하나의 보안카드 의 Index번호와 비밀번호 입력창을 보여준다. 해당 Index번

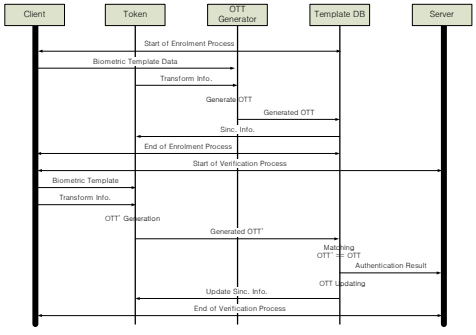
호와 연결이 되는 카드의 번호를 입력하면 입력된 번호와 비밀번호를 암호화 하여 전송하게 된다. Index번호에 따른 카드번호는 회원가입시 핸드폰에 저장된 프로그램을 통해 알 수 있게 된다.

```
/* 입력 : ID, CardNumber, Password
출력 : 2차 로그인 결과 */

Bool Login_CardNumber_Password(ID){
    String CardNumber, Password;
    Bool Result;
    String EncodeData;
    print( Random_Card_IndexNumber );
    Input( CardNumber, Password);
    EncodeData=Encoding(CardNumber,Password);
    Result = Compare( ID, EncodeData);
    return Result;
}
```

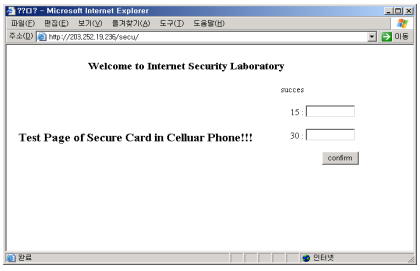
카드번호와 비밀번호를 통한 로그인 모듈에서는 카드번호와 비밀번호를 입력받기 위한 변수가 선언이 되며 결과를 출력하기 위한 Result가 선언이 된다. 우선 Print() 함수를 통해 랜덤하게 생성한 카드의 Index번호인 Random_Card_IndexNumber를 출력한다. 출력된 Random_Card_IndexNumber에 맞추어 사용자가 CardNumber와 Password를 입력하게 되며 입력된 CardNumber와 Passwrod는 Encoding()함수를 통해 암호화과정을 거친다. 암호화과정을 통해 생성된 data는 EncodeData에 저장되며 암호화된 EncodeData와 ID를 Compare()함수에 전달하여 사용자정보가 저장된 서버와 비교를 하게 된다. 비교 결과는 Result함수에 전달되어 반환하게 된다.

본 실험은 리눅스 시스템에서 이루어 졌으며 웹서버는 Apache, 데이타베이스 서버는 Mysql, 시스템 언어로는 PHP를 사용하였다. 현재 대부분의 웹사이트는 ID, Password만을 비교하기 때문에 한번 누출된 ID와 Password만을 가지고 다른 사람인 것처럼 인증을 받을 수 있다. 본 연구에서 제시한 보안카드 기반 인증 시스템의 전체적인 작동방식을 살펴보면 다음 그림 9와 같다.



[그림 9] 본 연구에서 제시한 모듈 실행 단계

위 그림은 휴대폰 기반 일회용 토큰 방식을 적용한 다중 인증 구조에 대한 것으로 세션키를 생성하는 과정에서 일반적인 비밀값 기반의 암호학적 대칭키를 사용할 수도 있고, 바이오 정보와 연계된 개인키/공개키를 사용할 수 있다. 사용자는 최근 이슈가 되고 있는 안드로이드 OS 기반으로 부가적인 SW를 설치/운용할 수 있는 핸드폰을 가지고 있으며 바이오 인증서 등과 연계될 수 있는 인증 서버에 접속하여 다중 인증 과정을 수행하게 된다.



[그림 10] 다중 인증 정보 입력 화면

현재 대부분의 웹사이트는 ID, Password만을 비교하기 때문에 한번 누출된 ID와 Password만을 가지고 다른 사람인 것처럼 인증을 받을 수 있다. 하지만 그림 10과 같이 본 연구에서 제시한 기법을 이용할 경우 사용자에 대한 바이오 정보와 연계하여 키를 생성하게 되며, 생성된 키 값 정보는 핸드폰 기반 일회용 토큰 생성 과정에 적용되어 보다 안전한 인증 시스템에 적용 가능한 구조이다.

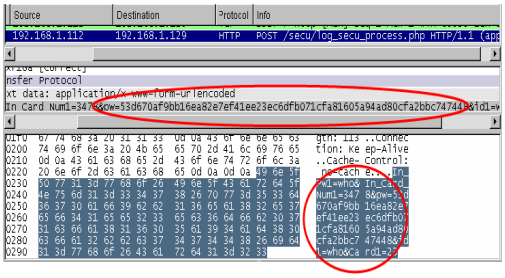
본 논문에서 제시한 모델에서 송수신되는 패킷 정보를 스니핑하여 분석한 결과는 다음과 같다. 다음 그림에서와 같이 캡춰된 패킷 내용을 분석해 보면 전송되는 데이터가 해쉬 방식으로 변환되었으며 바이오 키 값을 이용하여 전자서명되어 전송된다. 따라서 전송되는 메시지 자체에 대한 인증 및 송신자에 대한 부인봉쇄 기능도 제공할 수 있다. 결국 인증 시스템에서의 안전성을 높일 수 있었다.

4.3 시스템 비교분석 및 평가

본 연구에서 제시한 기법은 기존의 휴대폰 기반 인증에서 가장 간단하게 nonce 값을 이용하여 로그인 사용자의 신원을 확인하는 방법보다 안전성을 높일 수 있다. nonce 값을 전송하기 위해서는 별도의 안전한 채널을 통해 nonce 값만을 전송하고 이를 통해 일회용 세션 키 또는 일회용 패스워드(OTP) 값을 생성할 수 있으나. 그림 11과 같이 이른바 대포폰 및 비 인가된 핸드폰 등에 의해 불법 사용될 수 있다는 문제점이 있다. 또한 기존 기법

[10,11]인 경우 바이오 정보를 이용하여 인증 과정에 활용하고 있지만, 기존 연구와 비교하였을 경우 본 연구에서 제시한 기법인 경우 단순히 nonce 값을 생성하는 것 대신에 핸드폰 내 일회용 토큰에 기반한 1차적 인증 과정을 수행하고, 추가로 바이오 정보 등을 이용한 다중 인증 모듈로도 적용 가능하다는 장점이 있다.

물론 본 연구에서 제안한 핸드폰 기반 일회용 토큰 기법인 경우 바이오 정보와 연계하지 않을 경우 기존의 nonce 기반 사용자 인증 방법으로도 활용 가능하다.



[그림 11] 로그인 정보 스니핑

본 연구에서 제시한 기법을 기존의 인증 시스템과 비교 분석한 결과는 다음 표 1과 같다. 본 연구에서 제시한 일회용 토큰 기반 로그인 기법은 기존의 로그인 기법에 사용되는 기본정보를 사용하며 부가적 정보를 이용하여 다중 인증(Multi-factor Authentication)을 제공한다. 물론 본 연구에서 제시한 기법은 바이오 정보 등과 같은 부가적 정보 입력을 필요로 하고 있으며, 1회 생성된 일회용 토큰 정보는 기존의 보안카드와 유사하게 30개의 숫자 조합으로 생성된다는 제한을 가지긴 하지만 사용자가 요청하거나 로그인할 때마다 매번 다른 번호를 생성/입력하게 되기 때문에 일회용 패스워드(One-Time Password) 기능을 제공한다.

[표 1] 기존 시스템과의 비교분석 및 평가

종류 / 기능	클라이언트 다운로드	공개키 기반	인증서 기반	제안 기법
ID 및 Password 입력	O	O	X	O
부가정보입력	X	X	O	O
Multi-factor Authentication	X	X	X	O
One-Time Password	X	X	X	O

5. 결론

대부분의 연구는 현재 얼마나 안전하게 통신을 하느냐에 관심을 가지고 새로운 기술의 개발에만 집중할 뿐 현재 구현된 기술의 보안성 향상 및 안전성 부분을 중요시하고 있지 않는 경향이 있다. 따라서 현재 대부분의 기업 및 사용자에게 제공되는 로그인 시스템인 경우 대부분 동일하거나 유사한 ID와 Password로 로그인을 하고 있는 경우가 많다. 결국 개인 프라이버시 정보 유출의 문제점이 발생하고 있다.

또한 최근 널리 공개된 패킷 스니핑 툴 등을 이용하여 네트워크 상에 전송되는 패킷을 누구나 손쉽게 캡춰할 수 있으며, 기존 TCP/IP 기반 프로토콜의 특성상 보안 모듈 등이 패킷 헤더 구조 등에 적용되어 있지 않기 때문에 웹 로그인 과정 등에 포함된 ID 및 Password 정보 등을 공격자는 손쉽게 획득할 수 있다. 이러한 문제점을 보완하기 위해 주기적으로 ID/Password 정보를 변경해야 하는데 이 또한 상당히 번거로운 일이 되고 있다.

따라서 본 연구에서는 일회용 토큰 방식을 통해 기존 웹 시스템 및 인터넷 관련 로그인 과정에서의 안전성을 향상시키기 위한 방법을 제시하였다. 본 연구에서 제시한 일회용 토큰 시스템은 현재 로그인 시스템의 큰 수정 없이 적용이 가능하며 개인별 고유한 바이오 정보 및 개인이 소유하고 있는 핸드폰 시스템 등과 연계하여 사용자와 전체 웹 기반 시스템의 안전함을 향상 시킬 수 있다. 앞으로 다양한 형태의 바이오 정보와 연계하여 다중 인증 서비스를 제공할 수 있는 방안에 대한 연구가 필요할 것으로 생각된다.

참고문헌

[1] 최동현, 김승주, 원동호, “일회용 패스워드(OTP: One-Time Password) 기술 분석 및 표준화 동향”, 한국정보보호학회지, Vol.17, No.3, pp.12-17, 2007.

[2] 김기영, “일회용 패스워드를 기반으로 한 인증 시스템에 대한 고찰”, 한국정보보호학회지, Vol.17, No.3, pp.26-31, 2007.

[3] 강수영, 이임영, “OTP를 활용한 UICC(Universal IC Card) 기반의 인증 메커니즘에 관한 연구”, 한국정보보호학회논문지, Vol.18, No.2, pp.21-31, 2008.

[4] 추연수, 이재식, 김정재, 조창현, 전문석, “OTP를 이용한 스팸 메일 차단 모듈 설계”, 한국정보과학회 2005 가을 학술발표 논문집, Vol.32, No.2, pp.235-237, 2005.

[5] 김대진, 최홍섭, “OTP를 이용한 IPTV 콘텐츠 보호 및 인증 시스템 설계”, 한국콘텐츠학회논문지, Vol.9, No.9, pp.123-137, 2009.
[6] 추성호, 제갈명, 박홍성, “일회용 암호를 이용한 국산 암호 인증 시스템”, 멀티미디어학술회의 논문집, Vol.5, No.1, pp.127-131, 2002.
[7] A. K. Jain, A. Ross, and S. Prabhakar, "Fingerprint matching using minutiae and texture features," to appear in the International Conference on Image Processing(ICIP), Greece, 2001.
[8] O. Peter, "Biometric generation of digital keys," Mini Symposium, DMIS-BUTE, 2001
[9] P. Janbandhu and M. Siyal, "Novel biometric digital signatures for Internet-based applications," Information Management & Computer Security, Vol.9, No.5, pp.205-212, 2001.
[10] P. Tuyls and J. Goseling, "Capacity and examples of template-protecting biometric authentication systems," Proceedings of BioAW 2004, Lecture Notes in Computer Science 3087, Springer-Verlag, pp.158-170, 2004.
[11] X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, and A. Smith, "Secure remote authentication using biometrics," Advances in Cryptology - EUROCRYPT 2005, Lecture Notes in Computer Science 3494, Springer-Verlag, pp.147-163, 2005.

이 형 우(Hyung-Woo Lee) **[정회원]**

- 1994년 2월 : 고려대학교 전산과학과 (전산학 학사)
- 1996년 2월 : 고려대학교 일반대학원 전산과학과 (전산학석사)
- 1999년 2월 : 고려대학교 일반대학원 전산과학과 (전산학박사)
- 1999년 3월 ~ 2003년 2월 : 백석대학교 정보통신학부 조교수
- 2003년 3월 ~ 현재 : 한신대학교 컴퓨터공학부 교수

<관심분야>
네트워크 보안, 정보보호, 무선네트워크

박 영 준(Yeong-Joon Park) **[정회원]**

- 1979년 2월 : 고려대학교 산업공학과(공학사)
- 1985년 2월 : 고려대학교 경영대학원 전자정보처리(경영학석사)
- 1997년 3월 ~ 현재 : 청강문화산업대학 사이버정보보안과 부교수

<관심분야>
컴퓨터네트워크, 유통정보, 사이버보안