情 報 保 護 學 會 誌
第17卷 第3號, 2007. 6

일회용 패스워드를 기반으로 한 인증 시스템에 대한 고찰

김기영*

요 약

공개된 네트워크 시스템 상에서의 개인 정보를 보호하기 위해 사용자 인증은 시스템 보안에 있어서 중요한 요소이다. 패스워드 기반의 인증 메커니즘은 비용과 효율성의 측면에서 널리 사용되고 있으며 최근 이중요소인증(Two-Factor Authentication)의 한 수단으로 일회용 패스워드(One-Time Password, OTP)를 도입하고 있다. 본고에서는 일회용 패스워드에 대한 인증 기술과 OTP 통합 인증 센터로 구성되는 OTP 통합 인증 시스템에 대해 검토하며 취약성에 대해 살펴보고 이에 대한 대응 방안으로 서비스 제공자의 식별자를 포함한 일회용 패스워드 생성 매체를 이용하는 사용자 인증 시스템을 구성한다. 또한 USB 토큰 형태의 일회용 패스워드 매체를 이용하여 다수의 서비스 제공자의 일회용 패스워드를 지원하는 인증 시스템을 제안한다.

Ⅰ. 서 론

인터넷의 광범위한 응용과 더불어 인터넷의 보안에 대한 관심이 높아지고 있다. 특히 전자금융거래의 활성화로 인해 인터넷 뱅킹 서비스 등의 이용이 증가하였으며 이에 따라 개인 정보 유출로 인한 전자금융 사고가 발생하였다.

개인 정보 유출로 인한 전자금융 사고의 발생을 줄일 수 있는 방법 중의 하나로 강력한 사용자 인증을 수행하는 것이다[1]. 인증(Authentication)이란 요구된 실체의 신원에 대한 보증 기능으로 현재 패스워드를 기반으로 한 인증 메커니즘이 가장 많이 사용되고 있다. 그러나 사용자만이 알고 있는 패스워드 이외에 사용자가 가지고 있는 매체나 사용자의 고유한 생체정보를 결합시켜 사용자 인증에 적용할 수 있으며, 스마트카드와 PIN(Personal Identification Number) 패스워드의 사용, 패스워드와 공인인증서의 사용 등이 이에 해당한다[1]. 두 개 또는 여러 개의 인증 수단을 사용하는 이중 요소 인증, 다중 요소 인증(Multi-Factor Authentication)을 도입하여 보안성을 강화하고 있으며 전자금융거래에서는 이중 요소 인증을 도입하여 사용자 인증을 강화하는 추세이다.

최근 보안카드나 공인인증서 이외의 이중요소인증 요소의 한 수단으로 일회용 패스워드를 도입하고 있다. 일회용 패스워드는 매번 다른 패스워드를 생성하여 사용자 인증을 수행하는 방식으로 이미 사용된 패스워드는 재사용하지 않는다.

본고에서는 일회용 패스워드에 대한 관련 기술에 대해 살펴보고 OTP 통합인증센터로 구성되는 일회용 패스워드 통합 인증 시스템 구성에 대해 검토한다. 또한 일회용 패스워드 통합 인증 시스템의 취약성에 대해 논의하며 이에 대한 대응 방안으로 서비스 제공자를 식별하여 일회용 패스워드를 생성하는 OTP 생성 매체의 사용을 제안하고자 한다.

Ⅱ. 일회용 패스워드

2.1. OTP의 특성

일반적인 패스워드는 정적인 인증 수단으로 네트워크 도청으로 인해 패스워드를 알아냈을 경우 불법적으로 재사용할 위험이 있다. 그러나 OTP는 이미 사용된 패스워드는 재사용하지 않으므로 네트워크 도청을 통하여 패스워드를 알아냈다 할지라도 더 이상 사용할 수 없으므로 이러한 위험을 방지할 수 있다. 따라서 OTP는 정적인 패스워드 사용에 따른 위험을 해결하고 개인

* 소프트포럼(주) SW연구개발실 (kiyoung@softforum.com)

정보 유출에 따른 사용자 인증을 강화하기 위해 도입되었다. OTP는 동적인 패스워드로 사용하기 위해서는 별도의 매체가 요구된다. 이 매체는 OTP를 생성할 수 있는 기능을 가지는 장치(Device)로 OTP 토큰(Token)이라고 한다. OTP는 OTP 생성매체에 의해 필요한 시점에 발생되고 매번 다른 번호를 생성한다.

OTP는 사용자가 가지고 있는 OTP 생성매체와 이에 의해 생성되는 패스워드로 사용자 인증을 수행하므로 이중요소 인증 수단으로 정적인 패스워드와 같은 한 가지 인증요소만으로 인증 받는 방식에 비해 높은 보안 수준을 갖는다.

2.2. OTP의 생성 방식

OTP는 OTP 토큰과 OTP 인증 서버의 동기화 여부에 따라 비동기화(Asynchronous) 방식과 동기화(Synchronous) 방식으로 분류한다.

2.2.1 비동기화 방식

비동기화 방식은 OTP 토큰과 OTP 인증 서버 사이에 동기화되는 기준 값이 없으며 사용자가 직접 임의의 난수값을 OTP 토큰에 입력함으로서 OTP 값이 생성되는 방식이다[2]. 대표적인 예로 질의-응답(Challenge- Response) 방식으로 OTP 도입 초기에 주로 사용되었다. [그림 1]과 같이 사용자는 OTP 인증 서버로부터 받은 질의 값(Challenge)를 직접 OTP 토큰에 입력하고, 이 때 생성된 OTP 값을 응답 값(Response)로 전송한다.

이 방식은 사용자가 OTP인증 서버로부터 받은 질의 값을 직접 입력하여 OTP를 생성하므로 보안사고 발생 시 책임소재를 명백히 가릴 수 있으며 서로 질의 값과 응답 값을 주고받으므로 상호 인증이 가능하다[1]. 그러나 사용자가 매번 질의 값을 입력하므로 이용하기 불편하며 질의 값이 중복되거나 동일한 값이 자주 나온다면 취약성을 가질 수 있다. 이를 방지하기 위해 OTP 인증 서버에서 질의 값을 별도로 관리한다면 인증서버의 부담이 증가된다. 또한 동기화 방식에 비해 네트워크 부하가 증가될 수 있으며 아이디와 패스워드를 기반으로 한 기존의 어플리케이션과의 호환이 용이하지 못하다.

2.2.2 동기화 방식

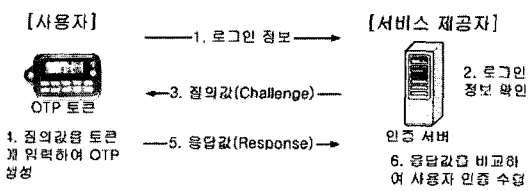
동기화 방식은 OTP 토큰과 OTP 인증 서버 사이에 동기화되는 기준 값에 따라 OTP 값이 생성되는 방식이다. 동기화의 기준 값에 따라 시간 동기화(Time-Synchronous) 방식과 이벤트 동기화(Event-Synchronous) 방식, 시간-이벤트 조합 (Time-Event-Synchronous) 방식으로 구분된다.

시간 동기화 방식은 OTP 토큰이 매 분마다 패스워드를 자동으로 생성하는 형태로 시간을 기준 값으로 하여 OTP 토큰과 OTP 인증 서버가 동기화 되어 있다. 따라서 사용자는 별도의 질의 값을 입력할 필요가 없으며 사용이 간편하다. 그러나 OTP 토큰과 OTP 인증 서버 간에 시간이 동기화되어 있어야 하므로 사용자가 일정 시간동안 OTP를 전송하지 못하는 경우에는 다시 새로운 OTP 값이 생성될 때까지 기다린 후 입력해야 한다.

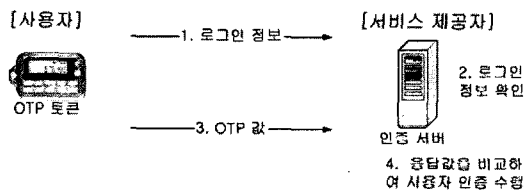
이벤트 동기화 방식은 OTP 토큰과 OTP 인증 서버의 동기화된 인증 횟수(Counter)를 기준으로 사용자가 인증을 요청할 때마다 OTP 값이 생성된다. 이는 OTP 토큰가 인증 서버 간에 시간이 맞지 않을 경우 수동으로 시간을 맞춰야 하는 시간 동기화 방식의 단점을 보안하기 위해 도입되었다[1].

시간 이벤트 동기화 방식은 시간 동기화 방식과 이벤트 동기화 방식을 결합한 형태로 OTP 토큰과 OTP 인증서버 간에 동기화된 사간 값과 동일한 인증 횟수를 기준으로 OTP 값이 자동으로 생성된다.

[그림 2]와 같은 동기화 방식은 OTP 인증 서버가 질의 값을 별도로 관리할 필요가 없으며 사용자는 질의 값을 OTP 토큰에 직접 입력하여 응답 값을 받을 필요가 없다. 또한 비동기화 방식에 비해 네트워크 부하가 상대적으로 감소되며 아이디와 패스워드를 기반으로 한 기존의 어플리케이션과의 호환성이 높다. 그러나 OTP



[그림 1] OTP의 질의-응답 방식



[그림 2] OTP의 동기화 방식

토큰과 인증 서버 간에 기준 값이 동기화되어 있어야 한다.

2.3. OTP의 생성 매체

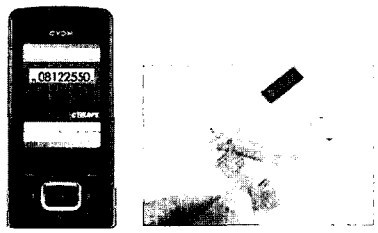
OTP 생성매체는 전용 하드웨어 OTP 토큰과 OTP 생성 기능을 소프트웨어로 탑재한 모바일 OTP, 카드형 OTP 등이 있다[2].

전용 하드웨어 OTP 토큰은 OTP 자체를 생성할 수 있는 연산 기능, 암호 알고리즘 등이 내장되어 별도의 하드웨어 매체로 [그림 3]과 같이 포켓용 계산기 모양, 호출기 모양, USB 등이 있다. OTP 생성 기능만을 지닌 전용 토큰이므로 추가 장비 필요 없이 사용이 가능하며 시스템 적용에 용이하여 많이 사용되고 있다. 그러나 사용자가 별도로 토큰을 구입해야 하므로 구입비용에 대한 부담과 휴대에 대한 불편함이 있다.

모바일 OTP는 [그림 4]와 같이 OTP 생성 알고리즘이 소프트웨어 모듈로 휴대폰에 탑재되어 있는 형태이다. 따라서 별도의 OTP 토큰을 휴대할 필요가 없으며 전용 토큰 구입비용을 절감할 수 있다. 그러나 이를 사용할 수 있는 서비스가 한정되어 있으며 OTP 생성 기능이 지원되는 전용 단말기를 구입해야 한다.

디스플레이형 OTP 생성 카드는 카드 리더기가 필요 없는 IC 카드형 OTP 토큰으로 IC 카드 내에는 OTP 생성 모듈이 내장되어 있다. [그림 4]와 같이 IC 카드 앞면에는 디스플레이 창이 있고 카드 뒷면에는 OTP 생성 버튼이 부착되어 있다. 이는 전자금융거래 등의 다양한 서비스에서 이용이 가능하며 별도의 OTP 토큰을 구입할 필요 없이 IC카드를 활용할 수 있다. 또한 리더기가

[그림 3] 전용 OTP 토큰[2]



[그림 4] 모바일 OTP와 디스플레이형 OTP 생성 카드[2]

필요 없는 OTP 토큰이라는 점에서 관심을 끌 것으로 보이나 초기 구입비용이 높다. 그 외에는 IC 카드에 OTP 생성 모듈, 배터리, 버튼, 스피커를 내장시켜 사용자가 버튼을 누르면 특정 소리가 생성되고 그 생성된 소리가 OTP로 사용되는 오디오형 OTP 생성 카드도 개발되었다.

2.4. OTP의 표준화 기술 현황

2004년 2월 미국 VeriSign 사의 제안으로 OATH (Open AuTHentication)이라는 조직을 설립하였다. 기존의 오픈된 표준을 기반으로 하여 견고한 인증 기술 체제를 확립하며 상호 운용성, 그 기술에 대한 일반 보급을 목적으로 현재 OTP와 관련하여 표준화를 진행하고 있다[2]. HOTP(HMAC based One-Time Password)는 OATH에서 표준안으로 제안한 OTP 알고리즘으로 이벤트 동기화 인증 방식을 사용한다. 또한 HMAC-SHA1 알고리즘을 이용하며 OATH가 지원이 가능한 OTP 기기에서 사용한다[2]. 이는 2005년 12월 IETF RFC 4226 문서로 작성되었으며 2006년 3월에 IETF 회의에서 발표되었다.

Ⅲ. 일회용 패스워드 통합 인증 시스템

3.1. OTP 통합 인증 시스템의 구성

OTP 통합 인증 시스템은 서비스 이용자가 하나의 OTP 토큰을 이용하여 다수의 서비스 제공자 간에 공동으로 OTP를 인증하는 시스템으로 [그림 5]와 같이 구성된다. OTP의 질의-응답 생성 방식은 사용자의 입력 작업으로 인한 불편함, 네트워크 부하, 호환성 문제 등을 포함하고 있으므로 시간 동기화 방식과 이벤트 동기화 방식, 시간-이벤트 동기화 방식을 고려한다.

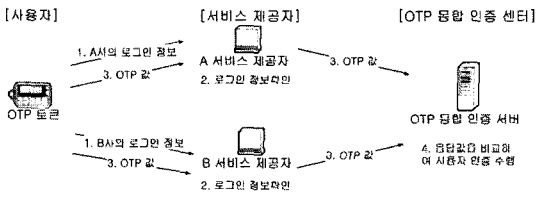
[그림 5]와 같이 사용자는 서비스 제공자(예. 은행 등)에 접속하여 OTP를 이용하여 사용자 인증을 수행하려고 한다. 사용자는 자신이 소유하고 있는 OTP 토큰을 이용하여 OTP를 생성하고 생성된 OTP 값을 OTP 통합 인증 서버에 전송한다. 시간 동기화 방식인 경우 OTP 통합 인증 서버는 시간 값을 기준으로 수용 가능한 범위의 OTP를 생성한 후 사용자가 전송한 OTP와 비교하고 인증을 완료한다. 이벤트 동기화 방식인 경우 OTP 통합 인증 서버는 전송 횟수를 기준으로 OTP를

생성한 후 사용자가 전송한 OTP와 비교하여 인증을 수행한다.

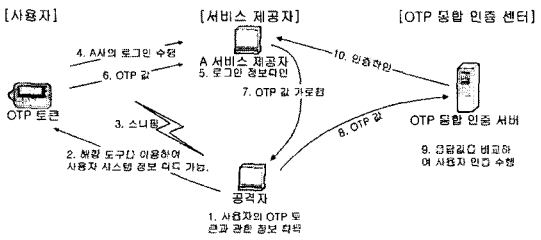
3.2. OTP 통합 인증 시스템의 취약성

〔그림 5〕와 같이 하나의 OTP 토큰을 이용하여 다수의 서비스 제공자 간에 공동으로 사용자 인증을 수행하는 OTP 통합 인증 시스템은 중간자 공격(Man-in-the-middle attack)에 관한 취약성을 가질 수 있다. 〔그림 6〕에서 볼 수 있듯이 해킹 도구를 이용하여 사용자의 시스템에서 서비스 제공자로 전송되는 데이터를 PC단 또는 네트워크단에서 스니핑(Sniffing) 할 수 있다고 가정한다.

시간 동기화 방식을 이용하여 OTP 인증을 수행한다면 〔그림 6〕과 같이 사용자가 OTP를 이용하여 OTP 통합 인증 서버로부터 인증을 수행 받고자 하나 공격자는 이 전송되는 사용자의 OTP를 획득한 후 사용자 시스템의 전송 시간을 지연시키면서 OTP 통합 인증 서버에게 인증을 요청한다. OTP 통합 인증 서버는 시간 값을 기준으로 수용 가능한 범위의 OTP를 생성하고 공격자가 전송한 OTP를 비교하여 정상적인 인증을 수행한다. 그리고 사용자도 그 지연시간을 짧게 가져갈 경우 OTP 통합 인증 서버와의 시간 동기화가 맞으므로 사용자 인증에 성공하게 된다. 시간 경계치를 넘어 실패를 한 경우에도 재시도를 하면 성공하기 때문에 사용자가 공격을 감지하기 어렵다.



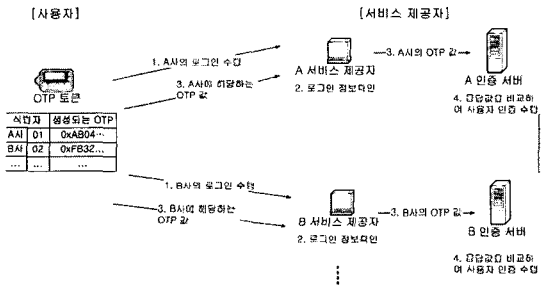
〔그림 5〕 OTP 통합 인증 시스템의 구성



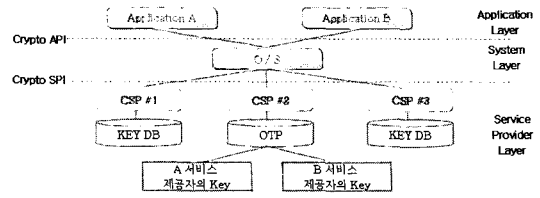
〔그림 6〕 OTP 통합 인증 시스템의 취약성

이벤트 동기화 방식을 이용하여 OTP 인증을 수행한다면 〔그림 6〕과 같은 시간 동기화 방식과 동일한 환경에서의 중간자 공격을 고려할 수 있다. 즉 공격자는 사용자 시스템과 OTP 통합 인증 서버 간에 발생한 인증 횟수와 동일한 환경을 구성하고 사용자 시스템에서 전송되는 OTP를 가로챈다. 공격자는 사용자 시스템의 OTP 인증 수행을 지연시키면서 가로챈 사용자 OTP를 OTP 통합 인증 서버에게 전송하여 올바른 사용자 인증을 수행한다. 그러나 이때 사용자는 다음 이벤트에 해당되는 서버의 OTP값과 비교가 되므로 인증에 실패할 수 있다. 물론 재시도를 할 경우 성공할 수도 있기 때문에 사용자가 공격을 감지하기 어려울 수 있다.

이와 같은 방식으로 해킹을 하게 될 경우 서비스 제공자마다 OTP 인증 시스템을 구성하고 서비스를 제공하는 경우 복수 로그인 방지만으로도 방어가 가능하나 복수 기관에서 동일 OTP를 사용하는 OTP 통합 인증 시스템으로 구성한다면 방어하기가 어렵게 된다. OTP 통합 인증 서버에서 이를 감지하기 위해서는 OTP제출값의 간격을 점검할 수 있으나 이 경우 공격자가 지연시간을 조절할 경우 이 또한 어렵다. OTP 통합 인증 서버에서 기관과 제출 시간 등을 조합한 좀 더 복잡한 방어 로직을 추가 할 수 있겠으나 이 또한 사고가 난 뒤에 감지하는 것이어서 큰 의미를 부여하기를 힘들다. 방어



〔그림 7〕 서비스 제공자의 식별자를 포함한 OTP 토큰을 사용한 사용자 인증 시스템



〔그림 8〕 USB 토큰 형태의 OTP를 이용하는 경우의 사용자 시스템 구성

를 위해서는 질의 응답 방식을 고려할 수 있으나 질의 응답 방식은 OTP통합 인증센터에서는 지원을 하지 않고 있기 때문에 현재로서는 방어가 불가하다고 판단이 된다.

Ⅳ. 대응 방안 및 고려 사항

본 절에서는 Ⅲ장 2절에서 고려한 OTP 통합 인증 시스템의 취약성에 대한 대응 방안으로 서비스 제공자를 식별하여 OTP를 생성하는 OTP 토큰의 사용을 제안한다. 이는 OTP 통합 인증 시스템의 본래 취지 중의 하나인 하나의 OTP토큰으로 여러 서비스 제공자 환경에서 사용할 수 있도록 해주며 OTP의 보안성은 그대로 유지할 수 있다. 물론 이런 경우에도 복수 로그인 방지를 할 경우 보안 강도를 좀 더 높일 수 있다. [그림 7]과 같이 OTP 통합 인증 서버를 이용하는 대신 각 서비스 제공자별로 인증 서버를 구성하며 사용자는 하나의 OTP 토큰을 사용한다. 이 OTP 토큰은 각 서비스 제공자의 식별자를 포함하여 사용자는 OTP 토큰에 포함된 서비스 제공자의 식별자를 선택하고, OTP 토큰은 해당 서비스 제공자의 인증 서버에 사용자 인증을 수행할 수 있는 OTP를 생성한다. 서비스 제공자의 인증 서버는 자신의 식별자와 사용자가 전송한 OTP를 이용하여 사용자 인증 여부를 결정한다.

또한 USB 토큰 형태나 스마트카드 형태의 OTP를 이용하는 경우 HSM과 동일하게 키를 안전하게 보관하면서 키를 외부로 유출하지 않는 범위 내에서 내부에서 OTP를 생성하여 외부로 전달할 수 있다. 이 경우 PC 등과의 인터페이스가 가능할 경우 외부 프로그램을 사용한다면 [그림 8]과 같이 구성된다.

현재 USB 토큰 형태의 OTP 제품들은 위와 같은 기능을 지원하고 있으며 RSA사에 의하여 마이크로소프트사의 CSP와 PKCS#11의 형태로 구현되어 있다. 따라서 서비스 제공자에 따른 키를 지정하여 서로 다른 키를 입력하고 사용할 수 있으며 이는 서비스 제공자의 식별자 등을 통해 키가 저장되어 있는 저장소에 접근할 수 있다. 안전한 키 관리를 지원할 수 있으며 사용자의 입력을 줄여 편리하게 사용할 수 있다. 또한 입력시간 오차 및 조작 오류에 의한 오차를 줄여 OTP의 보안성을 강화할 수 있다. 사용자가 직접 OTP 값을 생성하는 경우에도 사용자가 짧은 식별자코드 입력만으로 OTP 값을 생성할 수 있다. 예를 들면 한 자리 식별값을 이용하는 경우 10개의 기관에서 사용이 가능하고, 두 자리의 식별값을 이용한다면 100개의 기관에서 사용이 가능하다.

그러나 마이크로소프트사의 OTP 토큰을 지원하는 CryptoAPI[3]와 PKCS#11 메커니즘[4]은 서로 다른 형태의 API를 사용한다. 마이크로소프트사의 CryptoAPI는 해쉬 함수를 이용하여 OTP 값을 생성하고 있으나 PKCS#11 메커니즘은 서명 함수를 사용한다. 현재 이 방식은 RSA사에 의하여 작성 및 적용되고 있는데 이는 개념적으로 혼란을 가져올 수 있다. OTP는 기본적으로 서버와 클라이언트에서 동일한 키를 가지고 있는 것을 고려할 때 PKCS#11에서 사용하는 서명함수를 통하여 구현하는 것 보다는 마이크로소프트사의 CryptoAPI를 통해 구현한 사례처럼 키를 사용한 해쉬 개념으로 이해하는 것이 바람직스럽다고 볼 수 있다. 특히 OATH에서 표준안으로 제안된 HOTP(HMAC-based OTP)를 고려할 경우 더욱 타당하다고 볼 수 있다.

Ⅴ. 결 론

개인 정보 유출에 따른 인터넷 사고에 대응하고자 강력한 사용자 인증을 권고하고 있다. 최근 OTP를 인증 메커니즘을 이용하여 OTP 통합 인증 시스템 구성이 고려되고 있다. 이 시스템은 하나의 OTP 토큰을 이용하여 다수의 서비스 제공자 간에 공동으로 OTP를 인증하는 시스템으로 동기화 생성 방식을 기반으로 한다. 그러나 Ⅲ장 2절에서 고려한 바와 같이 서비스 제공자마다 OTP 인증 시스템을 구현하는 경우보다 더욱 중간자 공격에 대한 취약성을 가질 수 있다. 이에 대한 방안으로 서비스 제공자의 식별자를 포함한 OTP 토큰의 사용한 사용자 인증 시스템 구성한다. 즉 OTP 통합 인증 서버 대신 하나의 OTP 토큰을 이용하여 다수의 서비스 제공자에 대한 OTP 기능을 지원할 수 있다.

USB 토큰 형태의 OTP 등을 이용하는 경우 HSM과 동일하게 서비스 제공자에 따른 키를 지정하여 서로 다른 키를 사용할 수 있으며 이는 서비스 제공자의 식별자 등을 통해 키가 저장되어 있는 저장소에 접근할 수 있게 된다. 따라서 안전한 키 관리를 지원할 수 있으며 사용자의 입력을 줄여 편리하게 사용할 수 있다. 또한 입력시간 오차 및 조작 오류에 의한 오차를 줄여 OTP의 보안성을 강화할 수 있다. 그러나 OTP 토큰을 지원하는 PC 인터페이스는 서로 다른 형태의 OTP를 생성하여 호환성에 대한 문제점이 발생할 수 있다.

OTP는 인터넷 서비스 이용에 있어 강력한 사용자 인증 수단인 만큼 전자금융거래 등의 서비스의 활성화에 기여할 수 있도록 하기 위하여 OTP 도입에 따른 사용자의 편리성을 보장하면서 기존의 사용자 인증 시스템보다 강한 보안성을 보장하는 방향을 추진되어야 할 것이다.

참고문헌

[1] 백미연, "전자금융거래의 보안 강화 방안 및 OTP (One Time Password) 이용현황", 지급결제와 정보기술, pp. 71-100, April 2006.

[2] 금융보안연구원, "금융보안 주간정보", 2006.

[3] RSA Security, "A CryptoAPI Profile for One-Time Password Tokens, V1.0", April 2006.

[4] RSA Security, "PKCS #11 Mechanisms for One-Time Password Tokens", December 2005.

[5] N. Haller et. al., "The S/KEY One-Time Password System", IETF RFC 1760, February 1995.

[6] N. Haller et. al., "A One-Time Password System", IETF RFC 2289, February 1998.

[7] 장청룡, 이용권, 양형규, 이완석, 홍기융, "일회용 패스워드 시스템의 표준화 연구", 한국통신 정보호호학회 종합학술발표회 논문집, Vol 8. No. 1, pp. 277-287, 1998.

[8] D. M'Raihi et. al., "HOTP: An HMAC-Based One-Time Password Algorithm", IETF RFC 4226, December 2005.

〈著者紹介〉

김 기 영 (Ki Young Kim)

1997년 2월 : 한양대학교 전자공학과 졸업

1997년 3월 : 포스코개발 입사

1998년 1월 : 한국후지쯔 연구개발부 입사

2000년 10월 ~ : 소프트포럼 연구소장

관심분야 : 정보보호, 유비쿼터스 보안