클럭 카운트를 이용한 스트림 암호의 OTP 인증 프로토콜

조상일* · 이훈재** · 이상곤** · 임효택**

OTP Authentication Protocol Using Stream Cipher with Clock-Counter

Sang-Il Cho* · HoonJae Lee** · Hyo-Taek Lim** · Sang-Gon Lee**

요 약

현재 네트워크 상에서 사용자의 인증 부분이 시스템 보안상으로 아주 중요한 역할을 지내고 있다. 이러한 중요한 사용자 인증 부분에 일회용 패스워드 (OTP : One Time Password) 방식을 사용하기 위해 많은 기술적인 시도 및 개발이 이루어 지고 있다. 일회용 패스워드는 사용자가 인증 받고자 할 때 새로운 패스워드를 생성하고 사용 후 버린다는 구조를 가지고 있다. 이는 매번 같은 패스워드를 사용했을 때 발생하는 보안 문제점을 해결 할 수 있다. 그러나 OTP 인증 방법에도 여러 가지 공격 방법에 취약하다는 문제점이 노출 되어 있다. 본 논문에서는 기존의 인증 프로토콜 문제점을 개선하고 스트림 암호 알고리즘을 OTP에 사용할 수 있도록 클럭 카운트 기법을 이용한 새로운 인증 프로토콜을 제안한다.

ABSTRACT

User authentication has been one of the most important part of the network system. OTP(One-Time Password) has been developed and applied to the existing authentication system. OTP makes a different password and abrogates used password each time when user is authenticated by the server. Those systems prevent stolen-key-problems which is caused by using the same key every log-in trial. Yet, OTP still has vulnerabilities. In this paper, an advanced protocol which is using clock-count method to apply a stream cipher algorithm to OTP protocols and to solve problems of existing OTP protocols is proposed.

키워드

OTP, 인증 프로토콜, 클럭 카운터, 네트워크 보안

Key word

OTP, Authentication Protocol, Clock-Counter, Network Security

* 동서대학교 유비쿼터스 IT 학과
** 동서대학교 컴퓨터정보공학부

접수일자 : 2009. 04. 24
심사완료일자 : 2009. 05. 28

Ⅰ. 서 론

최근 인터넷과 같은 통신 기술이 급속하게 발달하여 많은 부분의 업무들이 인터넷을 통해서 이루어지고 있다. 인터넷은 개방형 네트워크이기 때문에 어떠한 사용자라도 접속 가능하게 된다. 이는 악의적인 사용자가 공격을 목적으로 접근하게 되었을 때 도청, 침입, 도난 등의 피해를 당할 수 있다. 이러한 피해를 막기 위해 최근 인터넷 보안에 대한 관심이 높아지고 있다. 특히 전자금융거래의 급속한 발전으로 인해 인터넷 뱅킹 서비스의 이용이 증가되고 있으며 이에 따른 전자금융 해킹 및 보안 사고가 발생되고 있다.

이러한 보안 사고의 발생을 줄일 수 있는 방법 중 사용자 인증 방법이 가장 보편적으로 사용된다. 인증(Authentication)이란 특정 사용자가 접속을 요구할 때 사용자의 신원에 대한 보증 기능으로 현재 Identity/Password를 기반으로 한 인증 기법이 가장 많이 사용되고 있다. 간단한 패스워드 인증 방법 이외에 사용자가 소유하고 있는 매체나 사용자 고유의 생체정보를 이용한 강력한 인증 방법도 사용되고 있다. 이러한 인증 방법들은 보안의 중요성에 따라 구분되어서 사용된다.

현재 전자 금융 거래에서 사용되는 인증 방법은 보안카드와 공인 인증서를 이용하여 사용자를 인증하고 있다. 그러나 최근에는 일회용 패스워드(OTP)[1,2] 기법을 새롭게 도입하여 사용되고 있다. 일회용 패스워드(OTP)는 사용자가 인증을 요구할 때 패스워드를 생성하여 사용하는 방법으로 매번 생성된 패스워드는 서로 다른 값을 가지고 있어 한번 사용된 패스워드는 재사용하지 않게 된다. 이는 공격자가 네트워크상에서 패스워드를 도청하거나 사용자가 패스워드를 분실 하더라도 안전을 보장할 수 있게 된다. 또한 일회용 패스워드는 익명성, 휴대성, 확장성의 특징을 가지고 있으며 사용자의 개인정보를 저장하여 이용하지 않기 때문에 개인 정보 유출을 사전에 방지 할 수 있다.

본 논문에서는 기존의 OTP 기술을 보완 할수 있는 클럭 카운트를 이용한 OTP 인증 프로토콜을 제안하고, 안전성 분석 및 성능을 분석한다.

본 논문의 2장에서는 OTP(One Time Password)에 대한 기술 개요를, 3장에서는 기존의 OTP 기술을 알아보고 4장에서는 스트림 암호 알고리즘을 이용한 OTP 인증 프로토콜을 제안·분석하며, 5장에서는 결론을 맺는다.

Ⅱ. 본 론

OTP[1,2]는 사용자가 인증을 요구할 때마다 새로운 패스워드를 생성하여 사용하는 방식이다. OTP 생성 방식에는 OTP 기기와 인증 서버간의 동기화 여부에 따라 비동기화 방식과 동기화 방식으로 나뉘게 된다[1,2].

2.1 비동기화 방식

비동기화 방식은 OTP 기기를 초기화 할 때 인증서버에 미리 약속해 놓은 동기화 정보를 사용하지 않고 사용자가 직접 인증 서버로부터 받은 질의 값을 OTP 기기에 입력하여 OTP 값을 생성하는 방법이다. 이러한 방법을 이용한 대표적인 기법이 질의-응답 (Challenge-Response) 방식이며 현재 전자금융거래에서 도입하여 사용되고 있다.

질의-응답 방식은 사용자가 인증을 요청할 때 인증 서버로부터 받은 임의의 값을 OTP 기기에 직접 입력한 후 OTP 값을 생성하는 방식으로 생성된 OTP 값을 인증 서버에 재전송하여 인증 절차를 완료하게 된다. 이는 OTP 기기와 인증 서버간의 동기화 정보가 필요 없는 장점을 갖고 있지만 사용자가 직접 OTP 기기에 질의 값을 입력해야 되고 다시 OTP 값을 입력해야 되는 불편함을 가지고 있다. 또한 인증 서버는 사용자의 질의 값을 따로 관리해야 되는 부담을 가지게 된다.

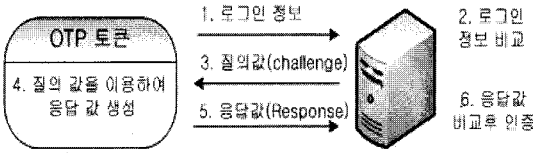

그림 1. OTP 비동기화 방식
Fig. 1 Asynchronous OTP

2.2 동기화 방식

동기화 방식은 OTP 기기를 초기화 혹은 생산될 때 인증 서버와 미리 약속된 동기화 정보를 이용하여 OTP 값을 생성하는 방식이다. OTP 기기와 인증 서버간에 반드시 동기화가 이루어져야 정확한 인증 절차를 이룰 수 있는 단점이 있지만, 질의-응답 방식에서 사용자가 직접

질의 값, 응답 값을 입력해야 되는 불편함을 개선한 방식이다.

동기화 방식에는 동기화를 위한 정보를 무엇으로 사용하느냐에 따라 시간 동기화(Time Synchronous) 방식, 이벤트(Event Synchronous) 동기화 방식, 조합(Time-Event Synchronous) 방식으로 구분된다.

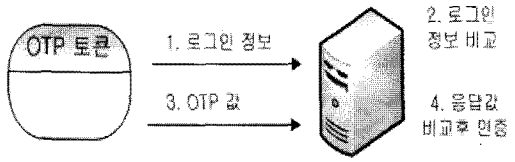

그림 2. OTP 동기화 방식
Fig. 2 Synchronous OTP

1) 시간 동기화 방식(Time Synchronous)

시간 동기화 방식은 OTP 기기에서 약속된 특정한 시간마다 패스워드를 자동으로 생성하는 형태이다. 이는 사용자가 별도의 질의 값을 입력할 필요가 없어 사용하기 편하지만 OTP 기기와 인증 서버간의 시간이 동기화 되어 있어야 되고 만일 사용자가 일정시간 안에 OTP 값을 인증 서버에게 전송하지 못하게 되면 다음 패스워드가 생성될 때 까지 대기 후 사용해야 되는 단점을 가지고 있다. 또한 공격자가 사용자와 인증 서버중간에서 MITM(Man-In-The-Middle) 공격으로 OTP 값을 획득하게 되었을 때 일정 시간 동안 사용할 수 있는 위험성을 갖고 있다.

2) 이벤트 동기화 방식(Event Synchronous)

이벤트 동기화 방식은 OTP 기기와 인증 서버간에 동일한 카운터 값을 이용하여 OTP 값을 생성하는 방식이다. 카운터 값을 공개된 시간 값과는 달리 OTP 기기와 인증 서버만이 알 수 있는 값으로 공격자가 추측 할 수 없는 장점을 가지고 있다. 이 방식은 사용자가 일회용 비밀번호를 생성할 경우 카운터 값을 입력 값으로 사용하여 비밀번호를 생성 한다. 비밀번호 생성 후에는 카운터 값을 증가시켜 저장하게 되며 증가된 카운터 값은 다음 비밀번호 생성 시 입력 값으로 사용된다. 이 방식은 여러 번의 인증 요구 시 OTP 기기와 인증 서버간의 인증 실패 혹은 통신 장애등으로 카운트 값이 달라질 수 있다. 이 같은 경우 OTP 기기와 인증 서버는 다시 카운터 값을 동기화해야 되는 불편함을 가지고 있다.

3) 조합 방식(Time-Event Synchronous)

시간-이벤트 조합 방식은 시간 동기화 방식과 이벤트 동기화 방식의 장점을 조합하여 구성한 방식으로, OTP 기기와 인증 서버 간에 동기화 된 시간 값과 동일한 카운터 값을 이용하여 OTP 값을 생성하게 된다.

Ⅲ. OTP 기술

3.1 S/Key 방식의 RFID 인증 프로토콜[3]

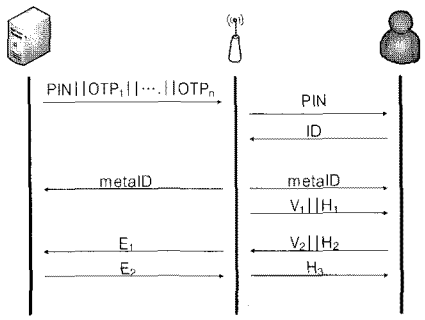

그림 3. 등록 및 인증 과정
Fig. 3 Connection and authentication process

1) 등록 과정

데이터 베이스는 PIN 과 초기값(seed)를 해쉬하여 OTP 를 n 개 생성하고 역순으로 RFID 리더에게 전송한다. 태그는 RFID 리더의 신호에 ID 값을 전송하고 ID 값을 해쉬하여 metaID 값을 생성한 RFID 리더는 데이터베이스와 태그에 재전송하여 등록을 마친다.

2) 인증 과정

Step 1 : 리더는 재전송공격을 막기 위하여 타임 스탬프를 생성하고 PIN 과의 연산 및 해쉬 값을 태그에게 전송한다.

$$V_1 = PIN \oplus TS_R$$
$$H_1 = H(PIN \| TS_R) \quad (1)$$

Step 2 : 태그는 V_1으로 TS_R'를 획득하고 해쉬 값을 검증하여 무결성을 확인한 후 TS_T를 갱신한다.

$$V_1 = PIN \oplus TS_R$$
$$TS_T = TS_R + LT \quad (2)$$

Step 3 : 태그는 $metaID$ 와 TS_T의 연산 값 및 해쉬 값을 생성하여 리더에게 전송한다.

$$V_2 = metaID \oplus TS_T$$
$$H_1 = h(metaID \| TS_T) \quad (3)$$

Step 4 : 리더는 저장하고 있는 OTP 중 $n-1$ 번째 패스워드인 $n-1$ OTP 과 V_2, H_2, TS_T 를 PIN 으로 암호화하여 E_1 을 생성하여 데이터베이스로 전송한다.

$$E_1 = E_{PIN}[OTP_{n-1}, V_2, H_2, TS_T] \quad (4)$$

Step 5 : 데이터베이스는 E_1 을 복호화하여 OTP_{n-1} 과 V_2, H_2, TS_T 를 획득하고 $metaID$ 와 OTP_n 을 검증하여 인증하고 OTP_n 과 TS_{DB} 를 생성한다.

$$D_{PIN}[E_1] = OTP_{n-1} \| V_2 \| H_2 \| TS_T$$
$$V_2 \oplus TS_T = metaID$$
$$OTP_n \leftarrow OTP_{n-1} \quad (5)$$
$$TS_{DB} = TS_T + LT$$

Step 6 : 데이터베이스는 ID 와 TS_{DB} 를 PIN 으로 암호화하여 E_2 를 생성하여 리더로 전송한다.

$$E_2 = E_{PIN}[ID \| TS_{DB}] \quad (6)$$

Step 7 : 리더는 E_2 를 복호화하고 TS_{DB} 를 검증하고 ID 와 TS_T 를 해쉬하여 태그에게 전송한다.

$$D_{PIN}[E_2] = ID \| TS_{DB}$$
$$H_3 = H(ID \| TS_T) \quad (7)$$

Step 8 : 태그는 H_3 을 검증하여 인증한다.

$$H_3 = H(ID \| TS_T) \quad (8)$$

3) 분석

S/Key 기반의 인증 프로토콜은 대칭키 암호화 알고리즘을 사용하여 데이터를 암호화/복호화 하고 있다. S/Key 알고리즘은 일방향 해쉬 함수를 반복해서 수행하여 OTP 값을 생성하게 된다. 이러한 방법은 n 번의 숫자가 커질 경우 많은 수학적 연산이 요구된다. 또한 대칭키 알고리즘은 RFID 와 같은 소형 프로세서에 적용하기에는 연산과정이 너무 많아 사용하기 힘들다.

최근 개인 정보가 중요시 되면서 익명성이 강조되고 있다. OTP 는 매번 다른 값을 생성하기 때문에 익명성을 보장해 준다. 그러나 S/Key 기반의 RFID 인증 프로토콜은 고정된 $metaID$ 값을 이용하여 데이터를 송수신하기 때문에 익명성을 보장 받지 못하는 문제점이 있다.

3.2 스트림암호 알고리즘의 OTP구현 Ver.1[6]

스트림 암호 알고리즘은 기본적으로 비트 또는 워드 단위의 키 수열로 이루어져 있다. 스트림 암호 알고리즘의 출력된 키 수열은 그림, 문자, 통신 등에 이용하기 위해 XOR 연산의 통해 암호화를 하고 있다.

스트림 암호 체계는 기본적으로 항상 같은 주기의 반복적인 수열을 출력하고 있으며, 동기화된 인증 시스템에 많이 사용되어지고 있다. 이러한 동기화된 인증 시스템은 재전송 공격에 취약하다는 단점을 가지고 있다.

이러한 재전송 공격에 취약하다는 단점을 보완 하기 위해 스트림 암호 알고리즘에 클럭 카운트(clock-counter) 를 이용하여 일회용 알고리즘을 구현하였다.

1) 클럭 카운터(Clock-Counter)

스트림 암호 알고리즘은 1 클럭의 주기로 1 비트 또는 워드 단위로 키수열을 생성 해 낸다. 이러한 연속적인 주기성은 스트림 암호 알고리즘의 특성을 파악 하고 공격자의 중요한 공격수단이 되고 있다.

스트림 암호 알고리즘의 OTP 방식에 사용하기 위해서는 클럭 카운트(clock-counter)가 필요하다. 클럭 카운트는 스트림 암호 알고리즘이 키 수열을 생성하는 클럭의 수를 카운트 하여 스트림 암호 알고리즘의 키 수열의 발생 시점을 바꿀 수 있다. 발생 시점을 바꿈으로써 공격자의 암호 알고리즘의 특성 파악을 어렵게 함으로써 스트림 암호의 효율성을 높일 수가 있는 것이다.

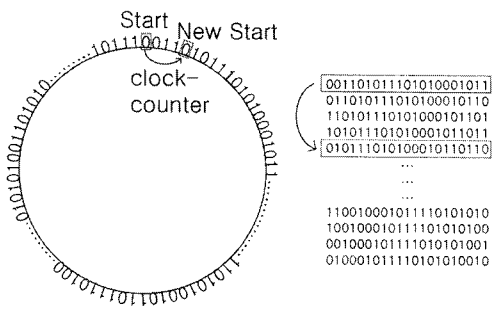

그림 4. 스트림 암호 알고리즘의 클럭 주기 및 클럭 카운터
Fig. 4 Clock period and counter of stream cipher algorithm

2) 클럭 카운트를 이용한 스트림 암호의 OTP 인증 프로토콜

스트림 암호 알고리즘의 OTP 상호 인증 프로토콜은 클럭 카운트를 이용하여 매번 사용자의 LFSR(Linear Feedback Shift Register)[7] 수열을 교체 함으로써 스트림 암호 알고리즘의 OTP 생성이 가능하게 하였다.

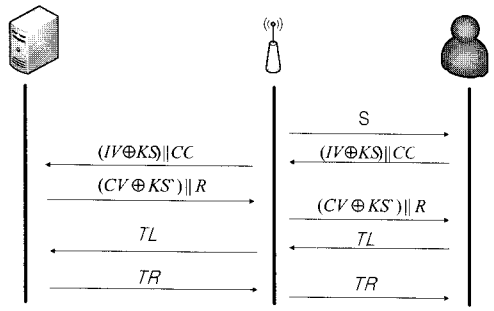

그림 5. 스트림 암호 알고리즘의 OTP 상호 인증 프로토콜 Ver. 1
Fig. 5 OTP mutual authentication protocol ver. 1 of stream cipher algorithm

Step 1 : 리더기에서 랜덤 값 S 를 생성 후 태그로 전송

Step 2 : 태그에서는 랜덤한 값과 클럭 카운트 값을 누적 시킨 후, 누적된 클럭 카운트 값 만큼 클럭을 이동시킨 후, 키 수열 (KS)을 발생 시킨다. 발생한 키 수열 값과 초기 벡터(IV) 값을 XOR 연산을 통해 암호화 시킨다.

$$cc = cc + s$$
$$(IV \oplus KS) \| cc \tag{9}$$

Step 3 : Step 2 에서 생성된 $(IV \oplus KS) \| cc$ 값을 리더기를 통해 서버로 전달

Step 4 : 리더기를 통해 $(IV \oplus KS) \| cc$ 값을 전달 받게 되면 서버는 cc 값 만큼 클럭 이동 후 KS 를 발생 시켜 KS 를 XOR 시켜 초기 IV 값을 찾아 내어서 ID 를 식별하게 된다.

$$(IV \oplus KS) \oplus KS \tag{10}$$

Step 5 : 서버는 IV 을 확인 후 재 검증을 위해 랜덤 R 값을 생성 R 값 만큼 클럭 이동 후 바뀐 초기화 벡터 값 (CV) 와 KS' 생성 시킨다. 변경된 초기화 벡터 CV 와 KS' 를 XOR 연산을 통해 암호화 하여, R 값이 같이 태그에서 재전송 하게 된다.

$$(CV \oplus KS') \| R \tag{11}$$

Step 6 : 식 11 와 R 값을 받은 태그는 R 값 만큼 클럭 이동 후 CV 값과 KS' 값을 생성, 서버로부터 받은 값을 검증 받게 된다. 서버로부터 받은 값이 참이면 태그는 다시 128 비트 만큼의 키 수열 T 를 생성하게 된다.

태그는 다시 키 수열 T 를 TL, TR 로 각각 64 비트씩 분할 생성한다. 검증을 위해 태그는 다시 TL 을 서버에게 전송하게 된다.

Step 7 : 서버는 Step 4 에서 $(CV \oplus KS') \| R$ 값을 전달 후 다시 T 를 생성 태그로부터 받은 TL 값을 재검증 하게 된다. TL 값이 정확하면 TR 값을 전달 함으로써 인증이 완료되었다는 것을 확인 한다.

Step 8 : 서버로부터 받은 TR 값을 태그가 생성한 TR 값과 비교 맞으면 서버와의 인증이 완료 되어 CV 값을 초기화 벡터(IV)로 변환함으로써 인증을 완료하게 된다.

3) 분석

클럭카운트를 이용한 스트림 암호의 OTP 인증 프로토콜은 OTP 기술 중 이벤트 동기화 방식의 단점을 보완하였다. 통신 과정 중 에러로 인한 동기 이탈이 발생하면, 이후 인증 서버에 접속하여 동기 이탈 여부를 확인 후 재동기화 할 수 있고, 스트림암호를 이용함으로써 빠른 암/복호화 및 적은 메모리 용량으로도 구현이 가능하다는 장점을 가지고 있다. 하지만 Step 3에서 서버로 전

송된 IV 값은 항상 변할 수 있다는 장점을 가지고 있지만 항상 변화는 값에 의해 다른 태그의 값과 동일하게 될 수 있다는 단점을 가지고 있다.

Ⅳ. 제안 방식

4.1 스트림암호 알고리즘의 OTP구현 Ver. 2

스트림암호 알고리즘의 OTP 구현 Ver. 2 에서는 Ver. 1에서 IV값의 동일화 문제를 해결하고자 태그의 고유 번호(Identification number)를 이용한 OTP 인증 알고리즘을 구현하고자 한다. 그리고 빠른 처리를 위해 DBSN(DataBase Section Number)를 이용해서 서버에 부하를 줄일 수 있도록 하였다. Ver. 1에서 과정은 동일하며 Setp 2~6 까지의 과정을 바꾸어서 구현하였다.

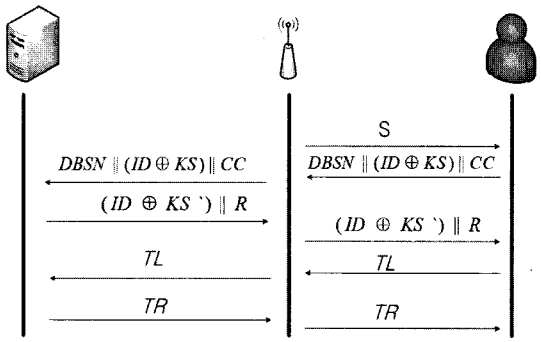

그림 6. 스트림 암호 알고리즘의 OTP 상호 인증 프로토콜 Ver. 2
Fig. 6 OTP mutual authentication protocol ver. 2 of stream cipher algorithm

1) 수정된 인증 단계

Step 2 : 태그에서는 랜덤한 S 값과 클럭 카운트 값을 누적 시킨 후, 누적된 클럭 카운트 CC 값 만큼 클럭 이동 시킨 후, 키 수열 KS를 발생 시킨다. 발생한 키 수열 값과 태그 고유 번호(ID)를 XOR 연산을 통해 암호화 시킨다. 암호화 시킨 값 $(ID \oplus KS)$ 과 DBSN, cc 값을 리더에서 서버로 전송 전송시킨다.

$$cc = cc + s$$
$$DBSN \| (ID \oplus KS) \| cc \quad (9)$$

Step 3 : Step 2 에서 생성된 DBSN, $(IV \oplus KS)$, cc 값을 리더기를 통해 서버로 전달

Step 4 : 서버는 리더기를 통해 DBSN, $(IV \oplus KS)$, cc 값을 전달 받게 되면 서버는 DBSN을 이용하여 데이터베이스의 특정 섹션 부분을 cc 값 만큼 클럭 이동 후 KS 를 발생 시켜 KS 를 XOR 시켜 ID 값을 찾아 내어서 식별하게 된다.

$$ID = (ID \oplus KS) \oplus KS \quad (10)$$

Step 5 : 서버는 ID 을 확인 후 재 검증을 위해 랜덤 R 값을 생성 R 값 만큼 클럭 이동 후, KS' 생성 시킨다. ID와 새로 생성된 KS' 를 XOR 연산을 통해 암호화 하여, R 값이 같이 태그에서 재전송 하게 된다.

$$(ID \oplus KS') \| R \quad (11)$$

Step 6 : 식 4 와 R 값을 받은 태그는 R 값 만큼 클럭 이동 후 KS' 값을 생성, XOR 연산을 통해 서버로부터 받은 값에서 ID를 재 확인 하는 과정을 거치게 된다. ID값이 참이면 태그는 다시 128 비트 만큼의 키 수열 T 를 생성하게 하게 된다.

태그는 다시 키 수열 T 를 TL, TR 로 각각 64비트씩 분할 생성한다. 검증을 위해 태그는 다시 TL 을 서버에게 전송하게 된다.

4.2 프로토콜 분석

표 1 은 기존의 인증 프로토콜과 본 논문에서 제안한 인증 프로토콜의 안전성 부분을 표시한 내용이고, 표 2 는 인증프로토콜 에서 연산되는 수식의 개수를 표현한 것이다. 표 1 에서와 같이 기존의 인증 프로토콜들은 ID, MetaID 등의 정보를 단순히 전송하는 과정으로 인해 익명성이 보장 되지 못하며, 공격자에 의한 전송과정중의 차단 문제를 해결하기 위해 재 동기화 과정이 다시 이루어져야 한다. 제안한 인증 프로토콜은 ID와 키 수열을 XOR 연산을 통해 익명성을 보장하며, 클럭 카운트를 이용하기 때문에 중간에 전송 차단으로 인한 재동기화 과정이 필요 없이 동기화가 자동으로 이루어지기 때문에 익명성과 연산의 효율성을 보다 높일 수 있다. 그리고, 기존의 인증 프로토콜들은 대칭키 알고리즘 또는 많은 해쉬 연산 방식의 인증 방식을 사용하기 때문에 많은 연

산과정이 필요하다. 이러한 방식은 RFID, 센서 네트워크, 스마트 카드와 같은 소형 프로세서에 적용하기 힘들다는 단점을 가지고 있다. 본 논문에서 제안한 스트림 암호 알고리즘을 적용한 기술은 적은 양의 연산 과정을 통해 소형 프로세서에서도 높은 비도를 가질 수 있게 적용이 가능하다.

표 1. 안전성 분석
Table. 1. Security analysis

	MITM	도청	재전송 공격	익명성
강수영 프로토콜[3]	O	O	O	X
Das 프로토콜[4]	X	O	O	X
Chien 프로토콜[5]	O	O	△	△
제안 프로토콜 (Ver. 1[6], 2)	O	O	O	O

	동기화	상호인증	데이터 무결성
강수영 프로토콜	X	O	O
Das 프로토콜	X	X	X
Chien 프로토콜	X	O	△
제안 프로토콜 (Ver. 1[6], 2)	O	O	O

표 2. 성능분석
Table. 2 Performance analysis

	해쉬함수	암호화	지수연산	키스트림
강수영 프로토콜[3]	(n+5)H	4E	0	0
Das 프로토콜[4]	8H	0	0	0
Chien 프로토콜[5]	3H	4E	4Exp	0
제안 프로토콜 (Ver. 1[6], 2)	0	2E		3KS

V. 결 론

본 논문에서는 기존에 제안된 인증 프로토콜의 문제점을 분석하고 보완하여 새로운 OTP 인증 프로토콜을 제안 하였다. 매번 다른 값을 가지는 OTP 의 특성은 개인 프라이버시 정보를 보호할 수 있어 익명성을 보장해 주며, 기존의 인증 프로토콜과 다르게 스트림 암호 알고리즘을 사용하기 때문에 보안의 강도에 따라 키 스트림 값을 유동적으로 조절 가능하다. 즉 어떠한 환경에서도 사용할 수 있는 확장성이 보장된다. 또한 이벤트 동기화 방식의 단점인 동기화 이탈시 문제점을 해결하여 동기화 값을 복구 할 수 있게 되었다. 향후 제안된 인증 프로토콜을 스마트 카드, RFID, 유비쿼터스 헬스 케어와 같은 소형 프로세스에 적용할 예정이며, 소형 프로세서를 이용한 하드웨어에 적합한 설계 연구가 필요 할 것으로 보인다.

참고문헌

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

[2] T. Tsuji, T. Kamioka, and A. Shirnizu, "Simple and secure password authentication protocol" ver.2(SAS-2), IEICE Technical Report, OIS 2003-30, vol. 102, no.314, September 2002.

[3] 강수연, 이임영, "향상된 S/Key 방식을 이용한 RFID 인증 방안에 관한 연구", 한국정보처리학회 춘계학술발표대회 제 14 권, pp. 1066-1067, 2007. 5

[4] M.L. Das, A. Saxena, V.P. Gulati, "A dynamic ID-based remote user authentication scheme", IEEE Transactions on Consumer Electronics, vol. 50, no.2, 2004, pp. 629-631.

[5] H.Y. Chien, C.H. Chen, "A Remote Authentication Scheme Preserving User Anonymity" , IEEE AINA' 05, Vol. 2, pp.245-248, 2005

[6] 조상일, 이훈재, 임효택, 이상곤, "클럭 카운트를 이용한 스트림 암호의 OTP 인증 프로토콜", 한국통신학회 하계발표 대회, Vol. 37, pp. 225, 2008

[7] S. Golomb, Shift Register Sequences, Aegean Park Press, Laguna Hills (CA), revised edition, 1982.

저자소개

조상일(Sang-Il Cho)

2003년 경운대학교 컴퓨터공학과 졸업(학사)
2005년 동서대학교 컴퓨터 네트워크학과 (공학석사)
2006년~현재 동서대학교 유비쿼터스IT학과 박사과정
※관심분야 : 암호이론, 네트워크보안, OTP보안

이훈재(HoonJae Lee)

1985년 경북대학교 전자공학과 졸업 (학사)
1987년 경북대학교 전자공학과 졸업 (석사)
1998년 경북대학교 전자공학과 졸업(박사)
1997년~1998년 국방과학연구소 선임연구원
1998년~2002년 경운대학교 조교수
2002년~현재 동서대학교 컴퓨터정보공학부 부교수
※관심분야 : 암호이론, 네트워크보안, 부채널공격

이상곤(Sang-Gon Lee)

1986년 경북대학교 전자공학과 졸업 (학사)
1988년 경북대학교 전자공학과 졸업 (석사)
1993년 경북대학교 전자공학과 졸업(박사)
1991년~1997년 창신대학 전자통신과 조교수
2003년~2004년 호주 QUT ISRC (암호학연구소) Visiting Fellow
1997년~현재 동서대학교 컴퓨터정보공학부 부교수
※관심분야 : 암호이론, 네트워크보안 프로토콜, 컴퓨터네트워킹

임효택(Hyotaek Lim)

1988년 홍익대학교 전자계산학과 졸업 (이학사)
1992년 포항공과대학원 전자계산학과 졸업(공학석사)
1997년 연세대학교 컴퓨터과학과 졸업(공학박사)
1988년~1994년 한국전자통신연구소 연구원
2000년~2002년 Univ. of Minnesota(미) 컴퓨터공학과 연구교수
1994년~현재 동서대학교 컴퓨터공학과 교수
※관심분야 : Computer Network, Protocol Engineering, Storage Networking, IPv6, Mobile