2000년 한국정보처리학회 추계 학술발표논문집 제7권 제2호

One-Time Password를 이용한 사용자 인증 시스템 설계에 관한 연구

윤석현*, 정경숙, 정태충
경희대학교 전자계산공학과
e-mail : pineapple@iislab.kyunghee.ac.kr

A Study of Design of User Authentication System Using the One-Time Password

Suk-Hyun Yun*, Tae-Chung Jung
Dept of Computer Engineering, Kyung-Hee University

요약

정보화 사회로의 진전으로 웹 환경에서의 다양한 서비스가 요구되고 있다. 그 용도가 점차적으로 증가 추세에 있는 웹 환경에서의 다양하고 중요한 정보에 대한 서비스를 위해서는 안전한 사용자 인증과 데이터의 무결성 등이 이러한 서비스에 대하여 필수적이라고 할 수 있다. 현재에는 SET이나 SSL과 같은 안전한 암호화 통신을 위한 많은 프로토콜이 연구 개발되고 있다. 하지만, 본 논문에서는 현재 그 중요성이 대두되고 있는 사용자 인증과 서버와 클라이언트간의 보다 신뢰할 수 있는 안전성을 위한 암호화 통신을 위해서 보안상 안전한 웹 시스템 환경을 설계하고자 한다.

1. 서론

WWW(World Wide Web)의 등장과 함께 수많은 일반인과 기업들은 인터넷을 통하여 막대한 정보를 교환하며 상호 통신을 하고 있다. 사용자는 브라우저 하나만 있으면 웹을 통해서 인터넷에서 제공하는 멀티미디어 정보를 손쉽게 접근할 수 있다는 장점 때문에 인터넷의 이용은 급격히 증가하는 추세에 있다. 인터넷의 급격한 이용 증가는 기업으로 하여금 웹을 통해서 홍보, 광고, 홈쇼핑, 홈뱅킹과 같은 웹서비스를 제공하면서 이른바 인터넷 비즈니스의 활성화를 가져오게 되었다.

이에 따른 중요한 정보의 안전한 통신을 제공하기 위해서 요구되는 전반적인 인터넷 보안 요소로는 로그온 ID 신분확인, 접근제어, 비밀성, 데이터 무결성, 패스워드 보호, 인터넷 연결망의 보호, 부인봉쇄 등이 주를 이루어 연구개발 되고 있다.

일반적으로 기존의 웹 시스템에서 제공하고 있는 보안 기능으로는 Basic Authentication, Network Domain Access Control, Message Digest Authentication, Channel-Based Security 등과 같은 단순한 기능만을 제공하고 있다. 이러한 보안 기법들은 침입자로 하여금 어렵지 않게 정보에 침투할 수 있다는 취약성을 띄고 있다.

이러한 웹 보안 문제를 해결하기 위해서 암호화 기법을 이용한 여러 방법들이 제안되고 있지만, 다음과 같은 어려운 점들로 인하여 활성화되지 못하고 있다.

- 기존의 HTTP를 확장
- 암호화 기법이 추가된 새로운 브라우저와 서버 개발
- 암호화 알고리즘의 다양성
- 키 관리와 관련한 기반 기술 요구

본 논문에서는 기존의 HTTP를 그대로 이용하면서, 클라이언트 브라우저에서 자바 DLL과 서버측 통신을 위한 자바 애플릿과 JSP(Java Server Pages) 기법을 이용하여 기존의 클라이언트와 서버를 새로 개발하지 않고 양쪽에 암호화 모듈을 결합한다. 통신하는 양쪽의 시스템과 암호화 모듈을 완전히 독립시킨 형태로 구성함으로써 안전성 등의 문제로 인해 발생할 수 있는 새로운 암호화 모듈의 교체가 용이해질 수 있도록 한다.

기존의 웹 시스템에서 제공하는 기본적인 사용자 인증을 포함하여 보다 강력한 사용자 인증을 제공하는 원타임 패스워드 기법을 도입하여 보다 신뢰할 수 있는 사용자 인증을 제공하고, 또한 보다 안전한 암호화 통신을 가능하도록 하기 위해서 공개키 암호화 방식인 타원곡선 암호화 알고리즘을 이용해서 모든 송수신 메시지는 암호화해서 전송하게 한다.

2. 연구 배경 및 관련 연구

2.1 One-Time Password System

패스워드 인증을 위해 현재 가장 각광받는 S/KEY One-Time Password 시스템은 네트웍 시스템에 로그인하는데 eavesdropping/ replay attack으로부터 안전한 인증을 제공한다. 이 시스템은 기존의 one-time 혹은 multi-use 인증 시스템들에 비해 여러 가지 장점을 가지고 있다. 이 시스템은 거의 모든 UNIX 시스템에 추가적인 하드웨어 없이, 패스워드 정보를 저장하지 않고 쉽고 빠르게 설치할 수 있으며 통신 기능을 가진 PC에도 사용이 가능하다. 개념적으로 간단하며 비밀 패스워드를 기억하도록 할 수 있다. 가장 큰 특징은 어떤 비밀 정보도 호스트에 보관되지 않는다는 점이다[1].

이러한 원타임 패스워드 시스템을 구현하기 위한 방법으로는 다음과 같은 것이 있다.

- 동기화된 시간을 유지하여 Time-Stamp를 사용
- 양쪽의 임의의 패스워드 리스트 내의 위치를 동기화하여 패스워드 사용
- Sequence generator의 상태를 동기화하여 임시적인 sequence number 사용
- Challenge-Response Schemes 이용

One-Time Password는 MD4나 MD5와 같은 단반향 해쉬 함수를 여러번 적용함으로써 계속해서 생성되어진다.

2.2 공개키 암호화 방식

비밀키(대칭형) 암호화 방식이 암호화하는 키와 복호화하는 키가 같거나, 혹은 하나를 알면 쉽게 대칭되는 키를 알 수 있는 암호화 시스템을 말하는 반면에, 공개키 암호화 방식은 암호화하는 키와 복호화하는 키가 서로 다르고, 하나를 알더라도 그에 대칭되는 키를 알기 어려운 암호 시스템을 말한다. 키 생성 알고리즘을 통해 2개의 키를 생성하고 그 중 하나를 공개하고 나머지 하나를 비밀키로 자신이 보관하여 사용하는 것이다.

비밀키 암호화 방식은 공개키 암호화 방식에 비해서 상대적으로 처리 속도가 빠르고, 데이터를 암호화하는데 사용하는 키와 복호화하는데 사용하는 키가 동일하다는 점이 가장 큰 특징이다. 따라서 데이터를 송수신하기 위해서는 통신하는 양쪽이 동일한 키를 공유해야 한다. 이러한 특징으로 인해 통신하는 양쪽은 키를 분배 및 관리하는데 있어서 어려움이 따른다. 또한, 웹을 통해서 다양한 형식의 멀티미디어 정보를 공유할 뿐만 아니라, 인터넷 비즈니스의 활성화에 따라서 더욱더 중요한 정보 통신의 안전성을 위해서는 공개키 암호화 방식을 이용해서 암호화를 해서 서버와 클라이언트 간의 안전한 통신을 보장해야 한다.

공개키 암호화 방식은 데이터를 공개키를 이용해서 암호화하고 비밀키를 이용해서 복호화하는 방식이기 때문에 비밀키 암호화 방식에 비해 키 분배 및 관리 문제를 쉽게 처리 할 수 있다. 비밀키 암호화 알고리즘은 단순히 주어진 평문의 비트나열을, 적당한 규칙을 이용하여 치환하고 대치하여 만들어진 알고리즘이지만, 공개키 암호화 알고리즘에 있어서는 수학적으로 정확한 조건(즉, 알고리즘에서의 키 값)이 주어지지 않으면 풀기 어렵다고 알려진 몇가지 문제를 이용하여 구현된다. 인터넷 보안에서 공개키 암호화 방식이 중요하게 대두되는 이유는 TCP/IP 프로토콜을 사용하는 인터넷이 항상 안전성을 보장하는 채널이라고 보지 않기 때문이다.

공개키 암호화 방식을 기반으로 한 알고리즘에는 Diffie-Hellman, RSA, DSA, ELGamal, ECC(Elliptic Curve Cryptosystem)등이 있다[2].

구현에 사용한 알고리즘은, 타원곡선을 이용한 공개키 암호시스템 즉, 유한체(finite fields) 위에서 정의된 타원곡선 군에서의 이산대수 문제에 기초한 타원곡선 암호시스템(ECC, Elliptic Curve Cryptosystem)으로써 1985년 N.Koblitz와 V.Miller에 의해 처음 제안된 이후 활발히 연구되고 있다[3]. 더욱이, 타원곡선방법(ECM, Elliptic Curve Method)은 최근 RSA 암호시스템의 근간이 되는 인수분해 문제와 소수성 테스트를 위한 효율적 알고리즘을 제공하기도 하였다. 가장 대표적인 공개키 암호화 알고리즘이라 할 수 있는 RSA를 사용하지 않은 이유는, RSA는 512bit, 1,024bit 키를 사용하는데 반해 타원곡선 암호화는 56bit, 84bit, 96bit의 작은 키로 강력한 보안이 가능하기 때문이다. KEY 길이가 1,024bit인 RSA 암호와 같은 수준의 안전성을 실현하는데 타원곡선 암호는 키 길이가 160bit면 되기 때문에 컴퓨터에 걸리는 부하가 적다.

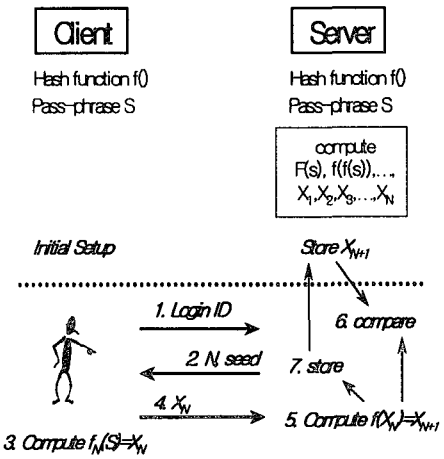
또한, RSA에서 사용하는 주요 연산은 곱하기인데, 곱하기가 연속으로 사용되는 만큼 수행 시간이 길어진다. 그러나 타원곡선 알고리즘에서는 주요 연산이 더하기이기 때문에 시간에서도 많은 절약을 할 수 있다. 수행 속도의 차이를 보면 타원곡선 알고리즘이 RSA에 비해 약 10배 정도 빠르다. 공개키 암호화 방식이 비밀키 암호화 방식에 비해 처리속도가 느리다는 단점이 있지만, 타원곡선 암호시스템은 비트당 안전도가 타 공개키 시스템보다 효율적이라는 것이 알려졌고, 최근 높은 속도의 구현이 가능하게 되었다[4].

본 논문에서는 공개키 암호화 알고리즘 중에서 ElGamal ECC 방법을 사용하여 웹 환경에 적합한 빠른 처리 속도와 안전성을 유지할 수 있게 하였다. 키의 분배와 관리를 효과적으로 하기 위해서 사용자 인증시 생성되는 OTP를 공개키로 이용함으로써 키 분배 문제를 해결하였고, 또한 사용자가 매번 로그인할 때 마다 새로 생성되는 OTP를 키로 사용하게 함으로써 양쪽은 공유키를 안전하게 저장해야 하는 어려움을 해결했다. 본 논문에서 제안한 방식의 장점은 추측하기 어려운 OTP를 공개키 암호화 알고리즘의 키로 이용함으로써 안전성이 보장된다는 것이다.

3. 시스템 설계 및 구성도

3.1 사용자 인증을 위한 기본 구성

본 논문에서 사용자 인증을 위해서 사용한 원타임 패스워드 시스템은 Bellcore의 S/KEY 시스템을 기반으로 구현한다[5]. 이 시스템은 RFC 2289에서 기술하고 있는 원타임 패스워드 시스템을 구현한 것이다. 여기서 사용되는 일반적인 S/KEY 원타임 패스워드를 간략하게 설명하면 다음과 같다[그림1].



먼저 사용자가 인증을 요구하면(로그인 시도), 서버쪽에서 One-time password의 일련번호와 시스템 특유의 "seed"를 포함하는 Challenge 메시지를 생성해서 사용자에게 전송한다. 사용자는 일련번호 N번째의 One-time password를 계산하여 그것을 서버로 전송하게 된다. 다시 서버는 사용자에게 전송했던 일련번호 N과 "seed"값을 이용하여 로그인을 요구한 사용자의 One-time password를 계산하고 비교하여 사용자에게 Response 메시지를 전송하여 사용자 인증을 해주는 방식이다[6].

3.2 웹 서버의 구성

서버의 전체적인 구성은 NT 웹 서버, MS-SQL 데이터베이스 서버, One-Time Password 인증서버, 암호화 모듈(ECC) 등 크게 네 부분으로 나뉜다.

웹서버와 인증서버, 암호화 모듈간의 인터페이스는 Java 애플릿과 JSP(Java Server Pages)를 이용해서 구현한다. 또한 One-Time Password 인증서버에서 저장해야 하는 사용자 정보는 MS-SQL 데이터베이스 서버를 이용하여 데이터베이스로 구축한다.

초기 인증시에 생성된 One-Time Password는 암호화 통신을 할 경우에 통신상의 공개키로서 동작하게 된다.

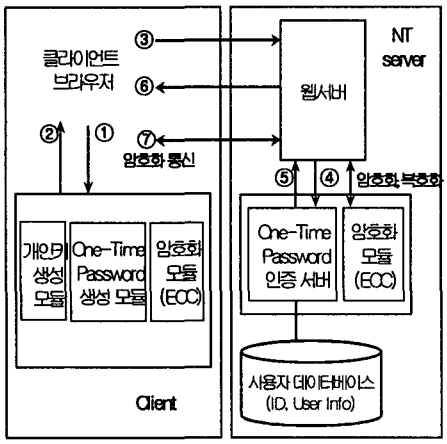
3.3 클라이언트 브라우저의 구성

클라이언트의 경우 One-Time Password 생성 모듈, 암호화 모듈을 브라우저에 삽입함으로써 사용자 인증과 암호화 통신을 구현한다.

One-Time Password 생성 모듈은 사용자가 입력한 패스워드와 이전 로그인에서 사용했던 일련번호를 통해서 일회용 패스워드를 계산하고, 암호화 모듈은 클라이언트와 서버 사이의 데이터 암호화/복호화를 수행하여 실질적인 데이터 교환과 제어를 담당한다.

3.4 동작 시나리오

클라이언트와 서버 사이의 전체적인 통신 과정은 다음과 같다.([그림 2] 참조)

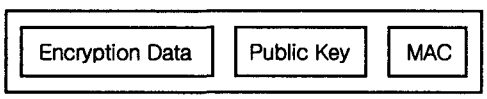


[그림 2] 시스템 동작 시나리오

① 사용자 인증을 위한 One-Time Password 생성 요구
② One-Time Password와 암호화 통신에 사용할 개인키를 생성하여 전송
③ 사용자 ID, Password, One-Time Password를 암호화하여 전송
④ 사용자 정보를 기반으로 One-Time Password 생성 요구
⑤ One-Time Password 계산 후 전송
⑥ 인증 메시지 전송
⑦ 다음 단계의 One-Time Password를 키로 이용해서 암호화 통신

3.5 인증과 통신을 위한 전송 데이터

초기 인증을 위해서 사용자가 서버로 전송하는 데이터에는 사용자 ID, Password, OTP 등이 있다. 시스템에서는 이 데이터를 다음과 같은 메시지 형태로 Java를 이용해서 서버로 전송하게 된다.



[그림 3] 초기 전송 데이터

Encryption Data는 사용자 인증을 위해서 필요한 정보인 사용자 ID, Password, OTP를 암호화 해서

전송한다. Public Key는 암호화 통신을 위한 공개키로서 이 시스템에서는 OTP 값을 전송한다. MAC (Message Authentication Code)는 사용자가 불법 사용자가 아님을 증명하기 위해서 비교되는데 사용하기 위해서 MD5 해쉬 알고리즘을 사용해서 Message Digest 한 값을 전송하게 된다. 이것은 인증서와 같은 기능을 한다.

3.6 일련번호의 동기화

OTP를 생성하기 위해서 초기에 사용되는 일련번호 N은 서버와 클라이언트(사용자)가 같은 값을 가지고 있어야 한다. 그래서, 시스템 자체에서 이 일련번호 값의 동기화를 위해서 접속 종료와 동시에 값을 서버로 전송하여 동기화를 한다. 새로운 접속에 대하여 일련번호 동기화가 되어있지 않은 경우에는 접속 전에 불법 사용자가 접속여부 등 원인을 분석하고 서버와 사용자 간의 일련번호 값을 리셋하게 된다.

4. 시스템 분석

원타임 패스워드 기법과 공개키 암호화 방식을 기반으로 한 웹 브라우저와 웹 서버는 S/KEY 원타임 패스워드 모듈과 ECC 암호화 모듈을 결합한 좀 더 안전한 형태의 웹 보안 시스템이 될 수 있다.

본 논문은 기발표된 논문[7]에서의 몇가지 문제점을 개선했다고 볼 수 있다. 첫째로, [그림 2]에서 개선된 동작 구조를 제시한 것과 같이 복잡한 형식의 절차를 축소하여 더 간단하게 사용자 인증 절차를 수행하고 암호화 통신이 가능하도록 하여 전체적으로 속도 향상을 기대할 수 있다. 둘째로, 사용자 인증 단계에서 클라이언트 브라우저만 가지고 있다면 ID 전송만으로 기본적인 사용자 인증 절차를 통과하는 형식을 사용자 ID, Password 그리고 One-time password를 동시에 암호화하여 전송함으로써 보다 더 강력한 사용자 인증 단계를 거치게 된다. 그리고, OTP를 생성하기 위해서 서버에서 클라이언트로 전송되는 Challenge 값이 외부로부터의 도청과 같은 공격으로부터 보호될 수 없다는 문제점을 해결하였다. 마지막으로는, TCP/IP 프로토콜을 사용하는 인터넷이 항상 안전성을 보장하는 채널이 아니므로, 더욱더 중요한 정보 통신의 안전성을 보장하기 위해서는 공개키 암호화 방식을 이용해서 암호화를 해서 서버와 클라이언트 간의 안전한 통신을 보장했다는 점이다. 다만, 비밀키 암호화 알고리즘(IDEA)과 공개키 암호화 알고리즘(ECC)를 사용했을 경우에 대하여 속도에 대한 문제에 대해서는 이 시스템에서 수행되는 모든 암호화되어 전송되는 데이터는 전달 인자와 같은 단순한 값들이기 때문에 속도에 대한 그 차이가 없었다.

본 논문을 통해서 얻어낼 수 있는 가장 큰 효과는 현재 널리 사용되고 있는 공개키 암호화 방식 기반의 통신 시스템에서 필수적으로 요구되는 인증서에 입각한 통신 방식에 비하여, 본 논문에서의 시스템을 이용한다면 인증기관으로부터 발급되는 인증서를 이용하지 않고 공개키 암호화 방식을 이용하여 안전한 사용자 인증과 암호화 통신이 가능하도록 한다.

5. 결론 및 향후 연구 과제

S/KEY One-Time Password 기법과 타원곡선 공개키 암호화 방식을 효과적으로 결합하여 Basic Authentication의 취약점을 해결하여 불법적인 사용자의 침입을 막는 더욱더 강력한 사용자 인증과 또한 공개키 암호화 방식과 효과적으로 결합한 형태로 One-time password를 암호화 통신의 공개키로 이용함으로써 네트웍을 통해 중요한 정보를 주고 받을 때 안전성을 쉽게 확보할 수 있을 것이다.

향후 연구 방향으로서는, 이 시스템 자체에서는 사용자가 클라이언트 브라우저에 종속적인 한계를 가지고 있으므로, 사용자가 언제 어디서나 이 시스템을 이용할 수 있도록 동적인 구성을 하면서 모든 보안 요소를 포함하는 시스템이 연구되어야 한다.

6. 참고문헌

[1] 홍승필, 고제욱, "정보보안 기술과 구현", 파워북, 1998

[2] C. Kaufman, R. Perlman, M. Speciner, "Network Security", Prentice Hall PTR, 1995

[3] "타원곡선 암호화 시스템 현황", http:// www.kisa.or.kr, 1999

[4] Alfred J. Menezes, Aleksandar Jruisic, "Elliptic Curves and Cryptography", IEEE P1363 Part6, 1995

[5] "A One-Time Password System RFC", http://www.ietf.cnri.reston.va.us/html.charters/otp-charter.html

[6] "일회용 패스워드 기술", http://www. kisa.or.kr/technology/sub4/password.htm

[7] 송상헌, "웹 보안을 위한 사용자 인증과 암호화 통신 구현", 1999

[8] 김창규, "암호학과 정보보호", http://www.koraa.or.kr/junggi/junggi_m.htm, 전파 2000년 1~2월 통권92호, 2000

[9] Bruce Schneier, "Applied Cryptography 2nd Edition", Wiley, 1996