특 집

전자서명과 인증

정 현 철
소프트포럼(주)

Ⅰ. 서 론

정보통신기술의 발달과 정보통신망의 확충으로 기존의 종이문서가 전자문서로 대체되고 있으며 전자상거래, 계좌이체 등의 금융행위도 증가되는 추세이다. 이로써 생산성과 편의성을 동시에 향상시킬 수 있게 되었다. 그러나 이러한 장점에도 불구하고 직접 상대방을 확인하고 거래하지 못하는 점에서 여러 가지 문제가 야기될 수 있다.

인터넷과 같은 온라인 상에서 거래 당사자간의 신원확인이 어려우며, 또한 디지털 정보의 특성상 전달 과정에서 종이 문서에 비해 위·변조가 상대적으로 용이하다는 문제점을 갖는다. 최근 이 같은 위험요소를 극복하기 위한 방법으로 공개키 암호를 이용한 전자서명과 사용자 인증이 각광을 받고 있다.

본 문서에서는 안전한 상거래와 전자문서 사용에 필수적인 전자서명과 인증에 대해 살펴보고자 한다. 2장에서 전자서명의 기본이 되는 공개키 암호, 인증서를 비롯하여 전자서명에 대한 간략한 원리를 기술한다. 3장에서는 이를 이용한 인증 방법에 대해 살펴보고, 4장에서 결론을 맺는다.

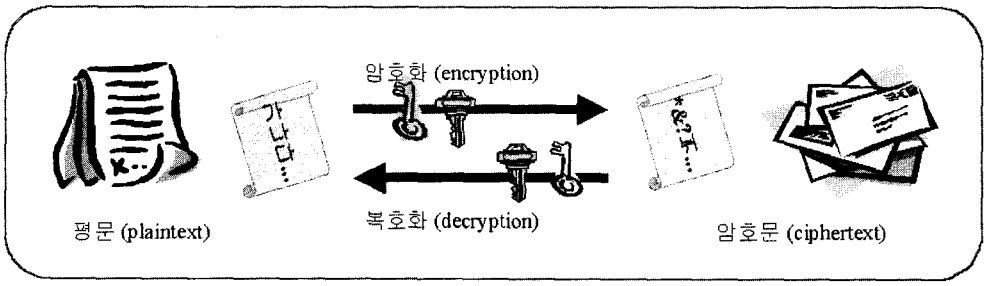
Ⅱ. 전자서명

1. 암호화 기술

암호기술은 군사적인 목적으로 이용되기 시작하였으며 그 역사는 매우 깊다(Julius Caesar는 전쟁 중에 명령을 전달하기 위해 암호를 이용했다는 기록이 있다). 고대 사회에서 암호의 목적은 멀리 떨어져 있는 군대 사이에서 명령을 안전하게 전달하기 위한 것이다. 이 경우, 전령이 적군에게 붙잡히거나 또는 통신망이 도청 당할 위험이 있으므로 어떠한 경우라도 명령의 내용이 비밀로 유지될 수 있어야 한다. 암호기술은 이러한 문제를 해결하기 위해 개발되었으며, 현대의 암호 기술은 암/복호화 연산은 해당되는 키가 있을 경우에만 수행될 수 있도록 구성된다.

우리는 암호화에 사용되는 키를 암호화키라 하고, 복호화에 사용되는 키를 복호화 키라 하며 암호화키와 복호화키는 쌍을 이룬다. 평문(plaintext)을 암호화 키를 이용하여 암호화 하였을 경우 우리가 알아볼 수 없는 암호문(ciphertext)이 생성되며 이 값을 다시 복호화 키를 이용하여 복호화 하였을 경우 원래의 평문이 나온다. 이를 요약하면 〈그림 1〉과 같다.

암호기술은 크게 비밀키 암호기술과 공개키 암호 기술로 구분할 수 있다. 공개키 암호 기술은 1976년 Diffie와 Hellman이 처음으로 제안되었으며, 암호화 키와 복호화 키가 서로 다름은 물론, 암호화키(공개키)가 공개되더라도 복호화키(개인키)는 알 수 없다는 특성을 갖는다. 따라서 따라서 우리는 암호화 키를 통신망에 공개하는 방법을 이용할 수 있고, 누구든지 그 키를 이용하여 문서를 암호화 할 수 있다는 장점을 갖는다. 이를 반대로 이용하면 자신의 개인키로 암호화(또는 서명) 한 문서는 해당 공개키만이 복호할 수 있으므로, 송신자에 대한 신원확인에 사용할 수 있음을 알 수 있고, 이것이 2.2장에서 다룰



〈그림 1〉 암/복호화 과정

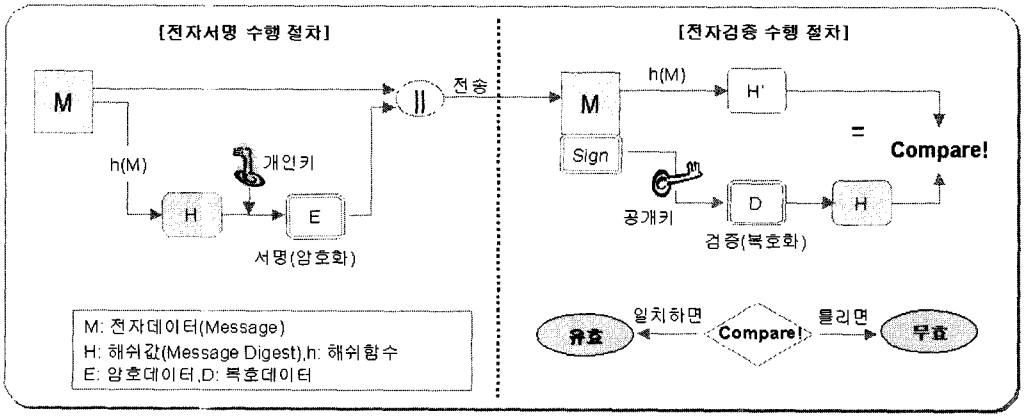
전자서명의 원리이다.

공개키 암호 기술을 사용하기 위해서는 키의 인증 문제에 대한 해결 방안이 요구된다. 아무런 조치 없이 통신망에 공개키를 공개하면, 공개키가 자신의 것임을 입증할 방법이 없다. 일례로 자신의 공개키를 다른 사람의 공개키로 위장하면, 암호화된 문서를 가로챌 위험이 있다. 공개키 기반구조(Public key Infrastructure, PKI)는 공개키 암호기술에 기반하여 암호키(공개키)의 소유자를 확인시켜주는 방법이다. PKI에서는 각 개인이 인증기관이라 불려지는 믿을 수 있는 기관으로부터 신분증(인증서)을 발급 받는 방식을 이용한다. 인증서에는 각 개인의 신원 및 암호화키가 명시되어 있으며, 이 값은 인증기관의 전자서명으로 보호된다.

2. 전자서명

암호화와 더불어 가장 널리 이용되는 암호 기술로는 전자서명이 있다. 우리가 실생활에서 거래문서를 작성할 때, 부인방지를 위하여 서명을 한다. 전자서명이란 그 이름에서 볼 수 있듯이 이러한 성질을 전자적으로 구성한 것이다. 전자서명은 서명값이라 불려지는 데이터를 생성할 수 있는 자를 특정 키를 알고있는 자만이 생성할 수 있도록 구성된다.

최근 공개키 암호기술에 기반한 전자서명이 많이 이용된다. 2.1에서 언급한 바와 같이 자신의 개인키로 문서를 암호화 하면, 공개키를 이용하여 문서의 생성자가 누구인지 알 수 있다. 마치 종이문서에 자신만의 고유한 서명을 기록한 것과 같은 효과를 갖는다. 따라서 전자서명에서는 암호화와는 반대로 개인키를 이용하여 서명을 생성하도록 하고 공개키를 이용하여 서명을 검증하는 방법을 사용한다. 전자서명은 일반적으로 해쉬함수와 함께 사용되며, 해쉬값에 대해 서명을 하는 방식을 이용한다. 그리고 전자서명의 검증은 공



〈그림 2〉 전자서명 수행/검증 과정

개키를 이용하여 서명값을 푼 값과 문서에 대한 해쉬 값이 같은지를 비교하는 방법을 사용한다. 이를 요약하면 〈그림 2〉와 같다.

전자서명은 서명자가 누구인지 알 수 있는 것 외에 여러 가지 효력을 갖는데, 전자서명의 효력을 갖추려면 다음의 조건들을 만족해야 한다.

- 위조불가 : 다른 사람의 서명을 위조하여 사용할 수 없어야 함.
- 서명자 인증 : 서명을 확인함으로써 이 서명은 분명히 서명한 그 사람이 생성한 것임을 확인할 수 있어야 함.
- 부인불가 : 서명을 하고도 서명 사실을 부인할 수 없어야 함.
- 변경불가 : 서명 후 문서의 변경이 불가능하여야 함.
- 재사용 불가 : 다른 문서에도 동일한 서명 값을 이용할 수 없어야 함.

현재 널리 이용되는 전자서명 함수로는 RSA, DSA 등이 있다. 그리고 이들은 모두 위의 조건들을 만족함이 알려져 있다.

Ⅲ. 인 증

1. 인증이란?

인터넷과 같은 통신망에서는 서로 비대면, 비접촉으로 통신이 이루어진다. 따라서 통신하고자 하는 상대방의 신원을 확인할 수 있는 방법이 요구되며 우리는 이를 인증이라 한다. 사용자 인증 방법은 인증에 이용할 요소에 따라 다음과 같이 크게 3가지로 구분할 수 있다.

- 패스워드 : 사용자가 알고 있는가?
  (패스워드, PIN)
- 토큰검사 : 사용자가 가지고 있는가?
  (스마트카드, OTP 토큰)
- 생체인식 : 사용자가 누구인가?
  (지문인식, 홍체인식, 음성인식)

위의 각 방법은 서로 다른 장단점을 갖는다. 패스워드를 이용하는 방법은 부가 장비가 필요 없고 가장 간편하다는 장점을 가지고 있으나 인간의 기억력에 의존하는 방법이기 때문에 낮은 엔트로피(entropy)를 가지고 있다. 또한 네트워크 상에서 전송되는 패스워드나 해쉬된 값을 가로채서 공격하는 사전공격(dictionary attack) 등이 가능하게 된다는 문제점을 가지고 있다. 토큰 검사 방법은 이러한 문제를 해결할 수 있다는 장점을 갖으나 부가 장비로 인한 비용 소모와 항상 휴대하고 다녀야 하는 불편함, 그리고 분실 및 도난의 위험도 가지고 있다. 마지막으로 생체인식(Biometrics) 방법은 위조, 위장 등의 위험성은 적으나 별도의 부가 장비로 인한 비용이 소요된다. 현대의 인증 시스템에서는 원하는 보안 정도 및 편의성에 따라 위의 방법들 중 하나, 또는 두 가지 이상의 방법을 복합적으로 사용하는 경우도 있다.

우리가 다룰 전자서명에 기반한 인증 방식은 통신하는 상대방이 그의 공개키 인증서에 포함된 공개키에 대응되는 개인키를 알고 있는지 여부를 검사하는 방법을 사용한다. 그리고 이러한 개인키를 스마트카드와 같은 별도의 장비에 저장하는 방법을 적용하여 보다 보안성을 높이는 방법 역시 이용되고 있다.

2. 인증 프로토콜

인증 프로토콜이란 통신하고 이는 상대방이 누구인지를 확인할 수 있는 프로토콜을 말한다. 이 프로토콜은 접근 제어를 요구하는 대부분의 시스템들이 채택하고 있는 방식이며, 대표적인 것으로는 UNIX system의 ID-password를 이용한 사용자 인증 방법이 있다. 그리고 전자서명에 기반한 인증 프로토콜은 전자서명을 이용하여 개인키 값을 알고 있는지 여부를 확인하는 방법을 사용한다. 일예로 철수가 영이의 시스템에 접속한다고 가정한다. 우리는 문자열 msg에 대한 철수의 전자서명은 Sign-철수(msg)라고 표기하고, 영이는 철수의 공개키 인증서를 알고 있다고 가정한다.

공개키기반 인증의 인증 요소는 상대의 올바른 개인키의 유무이다. 즉, 클라이언트가 올바른 개인키를 가지고 있으면 올바른 서명을 생성할 수 있고, 서버는 서명을 보고 상대방이 올바른 사용자라고 인증하게 되는 것이다. 전자서명에 기반한 인증을 수행하기 위해서는 철수가 영이에게 자신의 서명값(Sign_철수(msg))을 전달하고, 영이는 그 서명 값이 맞는지 여부를 검사하는 방식을 사용한다. 이 경우 전자서명의 안전성으로부터 서명 값은 철수가 생성한 것임을 확인할 수 있다. 그러나 이러한 방식으로 프로토콜을 구성하였을 경우 재시도 공격에 대해 취약하다. 재시도 공격이란 공격자가 도청을 통하여 인증에 필요한 철수의 서명값을 저장해 두었다가, 나중에 이 값을 영이에게 접속하고 싶을 경우 이 값을 재전송하는 방법을 통하여 철수로 가장하는 공격방법을 말한다. (서명 값인 Sign_철수(msg)은 철수가 생성한 서명이므로 항상 인증을 위한 영이의 서명 검증을 통과한다.)

이러한 문제점을 해결하기 위해서 Challenge-response에 기반한 방법과 Time-stamp에 기반한 방법이 널리 이용된다. 이 방식들의 원리는 "Sign_철수(msg)" 값이 사용될 수 있는 시점에 대해 제약을 가하는 것이다.

- Challenge-response : 영이는 먼저 자신이 생성한 난수 값을 철수에게 보내고, 철수는 인증을 위하여 이 값을 포함한 것에 대한 서명을 생성한다.(msg에 영이가 생성한 난수값을 포함시킨다.)
- Time-stamp : msg에 철수의 현재 시간(T_철수)을 포함시킨다.

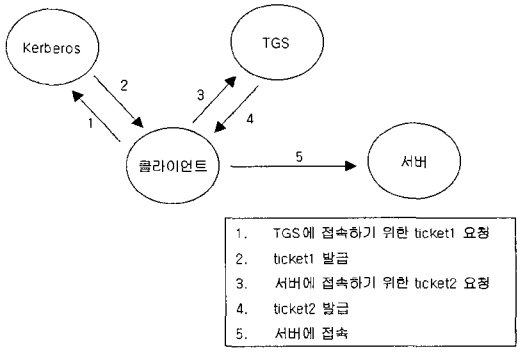
Challenge-response 방식에서 영이는 msg가 자신이 생성한 난수값을 포함하고 있는지 여부를 확인한다. 이럴 경우 난수의 특성으로부터 철수의 서명 값은 영이가 난수를 생성한 시점 이후임을 확인할 수 있다. 또한 Time-stamp 방식에서는 영이는 자신의 시간과 msg에 포함된 "T_철수" 값의 차이를 확인하고 차이가 적을 경우에만 서명 값을 받아드린다.

3. SSO(Single Sign On)

SSO 방식은 사용자가 한번의 로그인 과정을 통해 복수의 서버에 접속하기 위해 사용된다. SSO는 제휴 사이트들의 급속한 증가로 사용자 로그온시 많은 불편함 및 시간 소비 현상 발생을 줄일 수 있다는 장점을 갖으며 많은 ID/Password 관리로 보안 취약성 문제 역시 해결할 수 있다는 장점을 갖는다.

현재 SSO를 위해 가장 많이 이용되고 있는 표준 프로토콜은 Kerberos이다. Kerberos는 신뢰할 수 있는 제3자가 중재하는 TCP/IP 원격지 인증 프로토콜이다. DES(Data Encryption Standard) 알고리즘을 사용하여 보안을 유지하며, 각 사용자의 개별적인 키를 가지고 있다.

Kerberos 인증 시스템의 구성요소를 살펴보면, Kerberos 서버, TGS(Ticket-Granting Service), 접속을 시도하는 클라이언트, 클라이언트가 접속하고자 하는 서버로 구성된다. 우선 클라이언트는 Kerberos 서버에게 TGS에 접속하기 위한 ticket1을 요청한다. 발급되는 ticket1은 클라이언트의 비밀키로 암호화되어 전송된다. 그 후 클라이언트는 TGS에 접속하여 접속하고자 하는 특정 서버의 ticket2을 요청한다. ticket2는 Kerberos로부터 받은 ticket1 내의 키로 암호화 되어 전송된다. TGS로부터 받은 ticket2와 time-stamp가 포함된 특정 정보를 서버에게 보내면 time-stamp가 유효한 시간인지 등을 검사하여 인증이 완료된다. 이를 요약하면 〈그림 3〉과 같다.



〈그림 3〉 Kerberos 흐름도

따라서 Kerberos를 이용하는 사용자는 Kerberos 서버에만 로그인을 하면 된다. 그 이후에 서버에 접속은 Kerberos로부터 받은 ticket을 서버에 전송하는 방식을 이용한다. 마지막으로 ticket은 time-stamp를 포함하고 있으므로 공격자가 도청을 통하여 ticket을 얻었을 경우에도 안전성을 보장 받을 수 있다.

최초의 Kerberos 시스템은 DES와 같은 비밀키 암호 기술을 이용하였다. 그리고 최근에는 Kerberos 프로토콜의 확장으로 ticket 발급 과정에서 공개키 암호 기술을 이용하는 PKINIT (Public Key Initialization) 방식이 제안되어 사용되고 있다.

Kerberos 이외에도 매우 다양한 종류의 SSO 방식이 제안되어 있다. 그러나 이들 역시 Kerberos과 비슷하게 사용자는 중앙 서버에 로그인을 하고, 중앙 서버로부터 받은 ticket을 이용하여 서버에 접속하는 방식을 이용한다.

Ⅳ. 결 론

정보통신의 발달로 얻을 수 있는 디지털 문서, 전자상거래 등이 많은 편리함을 주지만 취약점을 이용하여 악용한다면 커다란 손실을 안겨줄 수 있다. 본 문서에서는 이러한 위험요소를 안전하게 지켜주는 전자서명과 사용자 인증 기술에 대해 간략히 살펴보았다.

암호 기술은 온라인 세상에 안전함을 줄 수는 있지만, 이를 이용하는 각 개인의 보안의식이 고취되고 관리/보완이 철저하다는 전제가 필요하다. 우리가 언급한 방식 역시 사용자가 자신의 암호키를 안전하게 관리한다는 조건 하에서 의미를 갖을 수 있는 것이다. 따라서 보안의 중요성에 대한 개개인의 인식과 관련 법규가 잘 뒷받침 되어야 할 것이다.

참 고 문 헌

[1] 정보보호인증관리센터 홈페이지 (http://www.rootca.or.kr/)

[2] 이만섭, 현대 암호학, 교우사, 2000

[3] B. Schneider, Applied Cryptography (2nd edition), John Wiley & Sons, 1996

[4] A. J. Menezex and P. C. Oorschot and S. A. Vanstone, Handbook of Applied Cryptography, CRC Press, 1997

[5] Charles P. Pfleeger, Security in Computing, Prentics-Hall International Inc., 1997

[6] W. Stallings, Cryptography and Network Security : Principles and Practice (2nd edition), Prentice Hall International, Inc., 1999

[7] Ross Anderson, "Security Engineering: A Guide to building dependable distributed systems," John Willy & Sons Inc., 2001

저 자 소 개

정 현 철

1989년 2월 계명대학교 전자계산학과 졸업, 1991년 2월 경북대학교 컴퓨터공학과 졸업(공학석사), 2003년 8월 경북대학교 컴퓨터공학과 졸업예정(공학박사) 1991년 2월~1998년 10월 : 한국전자통신연구원 선임연구원, 1998년 10월~현재 : 소프트포럼(주) 부사장, 2002년 12월~현재 : 알파로직스(주) 대표이사, 〈주관심 분야 : 공개키기반구조, 암호〉