SNAKE 프로토콜*

김상진 오희국

sangjin@cse.hanyang.ac.kr hkoh@cse,hanyang.ac.kr

한양대학교 전자계산학과

SNAKE(Secure Network Authenticated Key Exchange) Protocol

Hanyang University, Department of Computer Science and Engineering

요 약

본 논문은 네트워크 인증시스템 AUTHSYS에서 사용하는 SNAKE 프로토콜에 관한 논문이다. SNAKE 프로토콜은 키 분배 센터를 사용하는 중재 방식의 프로토콜이며, 티켓을 활용하는 프로토콜이다. SNAKE 프로토콜은 대칭형 암호 알고리즘과 해시 알고리즘만을 사용하는 프로토콜이며, nonce 기반 기법을 사용하여 메시지의 최근성을 보장한다. 또한 시스템간에 클럭 동기화를 요구하지 않으며, 클라이언트의 연산부담을 최소화하여 이동 컴퓨팅 환경에서 효과적으로 사용할 수 있도록 설계한 프로토콜이다.

1. 서론

컴퓨팅 기술과 통신 기술의 발달로 인하여 현재 네트워크를 이용한 정보 교환과 정보 처리의 사용 빈도는 기하급수적으로 증가하였으며, 가까운 미래에는 정보 교환의 매체로서 가장 중요하고 널리 사용될 것이 분명하다. 특히 90년대 초부터 활성화되기 시작한 인터넷은 빠르게 대중화되어 이제는 거의 모든 분야에서 활용되고 있으며, 정보를 제공하고 얻는 가장 빠르고 용이한 수단이 되었다. 또한 전자상거래의 활성화로 인하여 인터넷을 통한 상거래가 점차 활발하게 이루어질 전망이다. 고도 정보화 사회에서는 보다 편리하고 신속하게 많은 정보를 교환하는 것이 국가 경쟁력을 좌우하지만 이에 못지 않게 중요한 것은 교환되는 정보에 요구되는 보안 요구사항을 충족시키는 것이다. 일반적으로 보안 기술을 활용하지 않고 통신 채널로 정보를 전달할 경우, 이 정보는 아무에게나 노출되어 있다고 하여도 과언이 아니다. 이렇게 노출된 정보들은 악의적인 사용자들에게 남용되어 개인의 사생활을 침해받을 수 있으며, 경제적으로 큰 손실을 입을 수도 있다. 따라서 보안 기술에 대한 투자와 연구가 절실히 요구되고 있는 시점이다.

특히 현재 인터넷 서비스에서 가장 보편적으로 사용되고 있는 패스워드 기반 인증 메커니즘은 사용자의 계정명과 패스워드를 인가되지 않은 노출로부터 보호하지 않은채 그대로 전달하는 방식을 아직도 많이 사용하고 있다. 그러나 이더넷 방식을 사용하는 근거리 통신망의 경우에는 sniffering 과 같은 공격 방법을 이용하여 쉽게 다른 사용자의 패스워드를 알아낼 수 있기 때문에 보호가 요구된다. 현재 월드와이드웹 서비스의 경우에는 Netscape 사에서 개발하였고 현재 ITEF TLS(Transport Layer Security) 워킹그룹에서 인터넷 표준으로 제안되고 있는 SSL(Secure Socket Layer)를 이용하여 안전하게 사용자를 인증할 수 있다. 또한 텔넷, 파일전송, 전자우편과 같은 다른 인터넷 서비스는 Kerberos 와 같은 네트워크 인증시스템을 이용할 수 있으며 [1], 전자우편의 경우에는 S/MIME 를 이용할 수도 있다. 그러나 이들 시스템은 미국의 암호 기술 수출 억제 정책으로 우리 나라에서 현재 합법적으로 사용할 수 없거나, 보안 강도가 매우 약한 시스템만을 사용할 수 있다. 따라서 국내 사용자들이 보편적으로 사용할 수 있는 국내 기술로 개발된 인터넷 서비스 보안 메커니즘이 시급히 요구되고 있다.

본 논문은 본 연구진에서 개발한 네트워크 인증시스템 AUTHSYS 에서 사용하는 SNAKE 프로토콜에 관한 논문이다. AUTHSYS 는 Kerberos 와 같은 네트워크 인증시스템으로서 텔넷, 파일전송, 전자우편과 같은 인터넷 서비스에 필요한 인증 기능과 교환되는 정보의 비밀성, 무결성을 보장하여 주는 시스템이다. SNAKE(Secure Net-

* 본 논문은 정보통신부의 출연금으로 수행한 초고속정보통신 응용기술개발사업의 연구결과입니다.

work Authenticated Key Exchane) 프로토콜은 AUTHSYS 에서 요구되는 사용자 인증 서비스와 세션키 교환 서비스를 제공하는 암호 프로토콜이다. 본 논문의 구성은 다음과 같다. 2 장에서는 네트워크 인증시스템에 대한 소개와 AUTHSYS 의 특징에 대해 기술한다. 3 장에서는 SNAKE 프로토콜을 설명하고, 4 장에서는 향후 연구방향과 결론을 제시한다.

2. 네트워크 인증시스템

네트워크 인증시스템(authentication system)이란 암호 프로토콜을 이용하여 서비스를 요청하는 클라이언트와 서비스를 제공하는 응용 서버의 신원을 상호 안전하게 인증시켜 주고, 교환되는 정보의 비밀성을 보장하여 주는 시스템을 말한다. 네트워크 인증시스템은 일반적으로 인증 서버(authentication server)라고 하는 신뢰성 있는 제 3 의 서버를 두어 인증 메커니즘의 중추 역할을 하게 한다. 이 인증서버는 일반적으로 필요한 세션키를 생성 및 분배하며, 사용자 등록 서비스, 감사기록 서비스 등을 제공한다. 이와 같은 인증시스템의 사용을 통해 얻을 수 있는 이득은 크게 다음과 같다.

- 사용자 패스워드의 보호
- 하나의 인증 메커니즘을 다양한 서비스에 활용
- 사용자는 분산시스템에서 하나의 개체로 활동이 가능함
- 중앙집중적으로 분산시스템 전체에 대한 사용자의 접속과 활동에 대한 감사기록을 보존할 수 있음

현재 가장 널리 사용되고 있는 인증시스템에는 MIT 에서 개발한 Kerberos 가 있으며, 이 밖에 유럽 커미션이 지원하여 개발된 SESAME, IBM 에서 개발한 Krypt-Night 시스템 등이 있다 [2]. Kerberos 시스템은 현재 버전 5 가 발표되어 있으며, ITEF 의 CAT(Common Authentication Technology) 워킹 그룹에서 인터넷 표준으로 제안되고 있다. Kerberos 는 티켓 기반의 SSO(single-sign on) 서비스를 제공하며, 키 분배 센터(KDC, Key Distribution Center)와 티켓 승인 서버를 사용하는 시스템이다.

본 연구진에 의해 개발된 Authsys 는 Kerberos 와는 달리 티켓 승인 서버를 사용하지 않으며, 안전성 때문에 SSO 서비스를 제공하지 않는다. SSO 방식에서 사용자는 자신의 인증 데이터, 보통 패스워드를 한번만 입력하면, 그 다음부터는 인증 데이터의 입력없이 서비스를 사용할 수 있게 된다. 이것은 사용자에게는 매우 편리한 서비스이지만 보안 측면에서는 허점을 지니고 있다. 만약 공격자가 클라이언트 시스템의 제어권을 획득하게 되면 아무런 노력 없이 해당 시스템의 사용자로 위장하여 서비스를 받을 수 있다. 이것이 가능한 이유는 사용자의 비밀키가 티켓와 함께 평문으로 저장되어 있거나 세션키가 평문 형태로 티켓와 함께 저장되어 있기 때문이다. 따라서 Authsys 에서는 안전성을 높이기 위해 비밀키는 절대 평문 형태로 시스템에 저장하지 않으며, 서비스를 사용할 때마다 인증 데이터를 요구한다. Kerberos 에서 티켓 승인 서버를 사용하는 이유는 SSO 서비스를 제공하기 위한 보조 수단이므로 Authsys 에서는 사용하지 않는다.

3. SNAKE 프로토콜

SNAKE 프로토콜의 설계 목적은 다음과 같다. 첫째, 티켓(ticket) 기반의 프로토콜을 사용하여 처음 접속후부터는 적은 수의 메시지를 사용하여 사용자를 인증할 수 있도록 한다. 티켓의 사용은 적은 수의 메시지로 계속 인증할 수 있다는 장점뿐만 아니라 KDC 의 참여 없이 인증이 가능하다는 장점을 지니고 있다. 둘째, 공개키 암호 방식을 사용하지 않는다. 이것은 공개키를 사용하기 위해서는 공개키와 키의 소유자를 바인딩하여 주는 인증서(certificate)가 수반되어야 할뿐만 아니라, 공개키를 사용하기 위한 인증서버, 디렉토리서버와 같은 기반 구조(infrastructure)가 필요하기 때문에 비용 측면에서 효율적이지 못하다.

셋째, 시스템간에 클럭 동기화를 요구하지 않도록 한다. 일반적으로 티켓 기반 프로토콜에서는 티켓의 유효기간을 확인하기 위해 시스템간에 클럭 동기화가 요구된다. 그러나 시스템간에 클럭 동기화가 요구되면 클럭 값을 이용한 여러 가지 공격이 가능해진다 [3]. 따라서 시스템간에 클럭 동기화를 요구하지 않도록 KSL 프로토콜에서 사용하는 기법을 활용하였다 [4].

넷째, 클라이언트의 연산부담을 최소화하여 차세대 컴퓨터 환경인 이동 컴퓨팅 환경에 적합하도록 한다. 클라이언트의 연산 부담을 줄이고 서비스의 투명성을 증진하기 위해 클라이언트는 직접 KDC 에 접속하지 않는다. Kerberos 의 경우 클라이언트는 KDC 에 접속을 한 후에, 티켓 승인 서버에 접속하고, 그 다음에 응용 서버에 접속하는 방식을 사용하므로 클라이언트의 연산 부담이 높다. 따라서 본 시스템에서는 KSL 프로토콜처럼 클라이언트는 응용서버에게 서비스 요청만 하며, KDC 와 필요한 접촉은 응용 서버가 대신하는 방식을 사용하였다.

SNAKE 프로토콜에서 사용하는 가정은 다음과 같다. 첫째 각 사용자는 KDC 가 생성하는 키에 대해서는 신뢰한다. 둘째, 각 사용자는 등록과정을 통해 KDC 와 비밀키를 공유하게 된다. SNAKE 초기 프로토콜은 그림 1 에 기술되어 있다. SNAKE 초기 프로토콜은 클라이언트 A 가 응용서버 B 와 처음 접속할 때 사용하는 프로토콜이며, 처음 접속을 통해 티켓을 얻으면 그림 2 에

기술되어 있는 SNAKE 연속 프로토콜을 이용하여 서비스를 요청하게 된다. SNAKE 프로토콜은 KDC S 를 사용하는 중재 방식의 프로토콜이며, 초기와 연속 프로토콜은 모두 nonce 기반 기법을 이용하여 메시지의 최근성을 보장한다 [5]. 또한 해시함수를 사용하여 암호문에 불필요한 여분(redundancy) 정보를 최소화하였다. 그러나 본 프로토콜은 사전 공격에 대한 방어력은 지니고 있지 않다. 클라이언트와 KDC 간에 공유키 K_{as}, 응용서버와 KDC 간에 공유키 K_{bs}는 랜덤하게 생성한 키가 아니고, 사용자와 관리자가 선택한 패스워드로부터 변형한 키이다. 본 시스템에서는 Kerberos 에서 사용하는 키 변형 알고리즘과 유사한 알고리즘을 이용하여 패스워드를 키로 변형한다.

Message 1. A → B: A, N_a, L

Message 2. B → A: A, B, N_a, N_b

Message 3. S → B: $\{K_{ab}\}.K_{as}, h(N_a,B,K_{as},K_{ab})$
$\{K_{ab}\}.K_{bs}, h(N_b,A,K_{bs},K_{ab})$

Message 4. B → A: $\{K_{ab}\}.K_{as}, h(N_a,B,K_{as},K_{ab})$
$\{A,K_{ab},T_b,T'_b\}.K_{bb}, \{h(tck_{ab}),N'_b\}.K_{ab}$

Message 5. A → B: $\{N'_b\}.K_{ab}$

<그림 1> SNAKE 초기 프로토콜

Message 1. A → B: $A, N_a, \{A,K_{ab},T_b,T'_b\}.K_{bb}$

Message 2. B → A: $N_b, \{N_a\}.K_{ab}$

Message 3. A → B: $\{N_b\}.K_{ab}$

<그림 2> SNAKE 연속 프로토콜

SNAKE 초기 프로토콜을 요약하여 설명하면 다음과 같다. A 는 B 에게 서비스 요청을 하기 위해 자신의 계정명, nonce N_a, 그리고 티켓의 수명 L 을 B 에게 전달한다. B 는 A 를 인증하기 위해 서버명과 자신의 nonce N_b, A 의 계정명, 그리고 N_a를 S 에게 전달한다. S 는 응용서버로부터 요청을 받으면 세션키를 생성하여 A 에게 전달하기 위한 형태와 응용서버에게 전달하기 위한 형태, 두 가지를 만들어 B 에게 전달한다. 여기서 S 는 세션키만을 암호화하고, 세션키의 용도와 최근성을 확인할 수 있도록 해시 함수를 이용하여 만든 확인정보를 함께 전달한다. 이 확인정보는 공유키를 키로 사용하는 MAC(Message Authentication Code)이다. 이와 같은 방식은 IBM 의 KryptNight 와 Mao 와 Boyd 프로토콜에서 사용한 방식이며 [6], 불필요한 여분 정보를 최소화하여 사전 공격에 대한 방어력을 높일 수 있으며, 암호 연산 적용을 최적화하는 효과가 있다. 응용서버는 자신의 부분을 복호화하고 MAC 을 구하여 키를 확인한 후에 클라이언트 부분을 클라이언트에게 전달한다. 이 때 티켓 $\{A,K_{ab},T_b,T'_b\}.K_{bb}$도 함께 전달된다. 티켓은 세션키가 아닌 응용 서버 전용 비밀키로 암호화되며. 따라서 오직 응용서버만이 그 내부 내용을 알 수 있다. 이 기능 때문에 티켓 방식의 프로토콜을 사용하지만 시스템간에 클럭 동기화가 필요없다. 클라이언트는 티켓의 무결성을 확인하고, 세션키를 확인한 다음, 세션키로 응용서버의 nonce 를 암호화하여 응용서버에게 전달한다.

4. 결론

본 논문에서는 네트워크 인증시스템 AUTHSYS 에서 사용하는 SNAKE 프로토콜에 대해 설명하였다. SNAKE 프로토콜은 대칭형 암호 알고리즘과 해시 알고리즘만을 사용하는 프로토콜이며, 티켓 기반 프로토콜이다. 또한 키 분배 센터를 사용하는 중재 방식의 암호 프로토콜이다. SNAKE 은 시스템간에 클럭 동기화를 요구하지 않으며, nonce 를 이용하여 메시지의 최근성을 보장한다. 뿐만 아니라 클라이언트의 연산부담을 최소화하고 암호연산의 적용을 최적화하여 차기 이동 컴퓨팅 환경에서도 효과적으로 사용할 수 있는 프로토콜이다.

본 연구진은 현재 AUTHSYS 에서 필요하는 기본적인 시스템 구성요소인 KDC, 응용 클라이언트 이식 모듈, 그리고 응용 서버 이식 모듈을 모두 구현하였다. 앞으로 실제 텔넷, 파일전송, 전자우편과 인터넷 서비스에 활용할 수 있도록 이식 작업을 할 예정이다.

참고문헌

[1] J. Kohl and C. Neuman, "The Kerberos Network Authentication System," *ITEF CAT WG RFC No. 1510*, Sept., 1993.

[2] R. Molva, G. Tsudik, E. van Herreweghen, and S. Zatti, "KryptNight Authentication and Key Distribution System," *Proc. of ESORICS'92*, 1992.

[3] Li Gong, "A Security Risk of Depending on Synchronized Clocks," *Operating System Review*, Vol. 26, No. 1, pp. 49-54, Jan. 1992.

[4] A. Kehne, J. Schonwander, and H. Langendorfer, "A Nonce-based Protocol for Multiple Authentication," *Operating System Review*, Vol. 26, No. 4, Oct. 1992.

[5] Li Gong, "Variations on the Themes of Message Freshness," *Proc. of the IEEE CSFW VI*, pp. 131-136, Jun., 1993.

[6] Colin Boyd and Wenbo Mao, "Designing Secure Key Exchange Protocols," *Proc. of ESORICS'94*, pp. 93-105, 1994.