논문 02-27-11C-8 한국통신학회논문지 '02-11 Vol.27 No.11C

인증서 기반의 개선된 보안 쿠키의 설계와 구현

정회원 양 종 필*, 이 경 현**

The proposal of improved secure cookies system based on public-key certificate

Jong-Phil Yang*, Kyung-Hyune Rhee** *Regular Members*

요 약

웹 프로토콜인 HTTP은 이전 상태 정보를 저장하지 못하는 stateless 특성을 해결하기 위해서 쿠키(cookie)가 제안되었다. 그러나 쿠키는 평문 형태로 전송이 되며, 사용자 컴퓨터에 일반 텍스트 형태로 저장된다. 따라서, 공격자에게 쉽게 노출되어 쿠키 파일의 복사, 수정이 가능하여 보안적인 안전성에 심각한 위험이 존재한다. 본 논문에서는 이러한 쿠키의 보안 문제를 해결하기 위해서 공개키 인증서 기반의 새로운 보안 쿠키를 설계한 후 이를 구현하였다. 제안된 보안 쿠키는 사용자와 웹 서버간의 상호 인증 및 사용자 정보의 기밀성 및 무결성을 제공한다. 또한 웹 서버의 사용자 관리에 따른 부가적인 관리비용을 최소화시키기 위해 사용자 관리 정보를 보안 쿠키에 포함시킬 수 있다. 부가적으로 제안 방안의 성능 평가를 위해 기존의 HTTP 환경에서의 보안을 위해서 널리 사용되고 있는 SSL과의 수행 시간을 비교 분석하였다.

ABSTRACT

The HTTP does not support continuity for browser-server interaction between successive visits of a user due to a stateless feature. Cookies were invented to maintain continuity and state on the Web. Because cookies are transmitted in plain and contain text-character strings encoding relevant information about the user, the attacker can easily copy and modify them for his undue profit. In this paper, we design a secure cookies scheme based on X.509 public key certificate for solving these security weakness of typical web cookies. Our secure cookies scheme provides not only mutual authentication between client and server but also confidentiality and integrity of user information. Additionally, we implement our secure cookies scheme and compare it to the performance with SSL(Secure Socket Layer) protocol that is widely used for security of HTTP environment.

Ⅰ. 서 론

컴퓨터와 통신의 비약적인 발달로 인터넷을 통한 개인간, 기업간, 국가간의 정보 교류가 더욱더 빈번해지고 복잡해지는 추세에 있다. 최근 들어 웹을 이용한 정보 교환에 많은 취약성들이 대두되면서 접근 제어 및 보안 시스템에 대한 필요가 증대되고 있다. 특히, 전자상거래와 같은 상업적인 목적의 웹 사이트인 경우 지불 서비스가 필수적이고, 그에 따른 신용카드 및 개인정보 노출을 방지 할 수 있는 안전한 보안 시스템이 필요하게 되었다.

웹 서버는 HTTP 프로토콜을 이용한다. 이 프로토콜은 단순하기 때문에 클라이언트와의 이전 연결 상태를 유지하지 못한다. 즉, 웹 서버가 클라이언트의 요구에 응답을 완료하면, 요구한 클라이언트와 관련된 모든 정보를 잃어버린다. 따라서 이와 같은 문제점을 보완하기 위하여 만들어진 것이 쿠키 기술이다. 일반적으로 쿠키는 웹서버를 방문한 사용자의 ID, 패스워드 등과 같은 사용자 정보를 저장하여 다음 접속할 때 ID와 패스워드의 입력 없이 웹

* 부경대학교 전자계산학과 ** 부경대학교 전자컴퓨터정보통신공학부

※ 본 논문은 2002년 4월 JCCI 학술대회에서 우수논문으로 선정되어 게재 추천된 논문입니다.

서버에 곧바로 접속할 수 있는 기능 등을 제공한다. 그러나 대부분의 쿠키는 사용자 정보 및 패스워드와 같은 비밀이 보장되어야 할 자료들을 평문으로 네트워크상에 전송되거나 클라이언트에 저장하기 때문에, 안전성이 보장되지 않는 단점이 있다. 안전하지 못한 쿠키에 보안 기능을 추가하면 사용자 관련 자료(예를 들어, 사용자 ID, 패스워드, 신용카드 정보)를 서버 시스템 내부의 데이터베이스에 저장하지 않고 클라이언트 시스템의 쿠키에 저장하기 때문에, 서버의 사용자 관련 정보의 유지보수 비용을 줄일 수 있는 장점이 있다.

본 논문은 기존의 안전하지 못한 쿠키를 공개키 인증서 기반의 상호 인증과 대칭키 암호 기술을 기반한 사용자 정보의 기밀성을 제공하도록 보안 기능을 강화하였다. 2장에서는 기존의 쿠키 방안 및 취약점에 대해서 기술하며, 3장에서는 공개키 인증서 기반의 보안 쿠키 구조를 제안한다. 또한, 4장에서는 기본적인 보안 쿠키의 기능 확장에 대해서 고려하며, 5장에서는 제안 방안의 구현 및 성능 분석에 대해 기술한다. 마지막으로 6장에서 결론을 맺는다.

Ⅱ. 쿠키의 보안 위협 및 기존 연구

본 장에서는 현재 인터넷 환경에서 사용되고 있는 일반적인 쿠키의 구조 및 취약점을 기술하고, 안전한 쿠키를 위해서 지금까지 제안되었던 방안에 대해서 기술한다.

1. 쿠키의 구조 및 취약점

현재 웹에서 사용되고 있는 일반적인 쿠키의 구조는 그림 1과 같으며[1], 현재 일반적인 쿠키에 대하여 알려진 보안 위협은 아래와 같다[3],[6].

	Domain	Flag	Path	Cookie_Name	Cookie_Value	Secure	Date
Cookie 1	paper.net	True	/	Name_cookie	Yang	False	12/31/2002
Cookie n	paper.net	True	/	Role_cookie	student	False	12/31/2002

그림 1. 웹에서의 일반적인 쿠키

- 네트워크 위협 : 쿠키들은 네트워크 상에서 평문으로 전송되므로, 재생공격이나 변경에 취약하다.
- 종단 시스템 위협 : 쿠키는 하드디스크나 메모리에 평문으로 저장되어 있으므로, 변경, 복사가 가능하다. 따라서, 공격자는 손쉽게 다른 사용자로 가장할 수 있다.
- 쿠키 획득 위협 : 공격자가 적법한 웹사이트로 가장하여 쿠키들을 수집하고, 그 쿠키들을 이용하여 다른 사이트의 접근이 가능하다.

2. 관련 연구

본 절에서는 이미 제안되었던 안전한 쿠키 구조에 대하여 살펴본다. V. Khu-smith와 C.J. Mitchell[8]은 쿠키에 대한 보안 방안을 server managed cookie와 user managed cookie의 두 가지 형태로 분류하였다. Server managed cookie는 사용자에게 투명성을 제공한다. 또한, 서버 시스템의 변경이 요구되지 않는 장점이 있다. 하지만, 재생 공격에 취약한 단점이 존재한다. User managed cookie는 사용자가 언제, 어떻게, 무엇에 대하여 보안 메커니즘이 적용될 것인지에 대한 제어가 가능하다. 하지만, 이를 위해서 어떤 특별한 브라우저(browser)나 부가적인 소프트웨어가 요구된다.

J.S. Park과 R. Sandhu[6]는 아래의 3가지 형태를 가지는 server managed cookie 방안을 제안하였다.

- 주소 기반의 인증 : 쿠키의 쿠키값으로 사용자의 IP 주소를 가진다. 사용자의 IP 주소가 동적으로 할당되거나, 사용자가 프락시 서버를 사용할 경우에 이 방안은 바람직하지 못하다. 또한, IP spoofing의 위협이 존재한다.
- 패스워드 기반의 인증 : 쿠키의 쿠키값으로 사용자의 해쉬처리된 패스워드를 가진다. 이 방안은 동적인 IP 주소, 프락시 서버를 사용하는 환경에서 적절히 동작하며, IP spoofing에 대한 위협이 없다. 하지만, 사전공격(dictionary attack)에 대한 위협이 존재한다.
- 전자서명 기반의 인증 : [6]에서 제안되는 방안으로는 통신 개체의 공개키의 유효성을 검증하지 못한다.

V. Khu-smith와 C.J. Mitchell[8]은 대칭키 암호시스템과 공개키 암호시스템에 기반한 user managed cookie 방안를 제안하였다. 이 방안은 부가적인 소프트웨어를 통하여 인증, 무결성, 기밀성과 같은 보안 서비스를 제공한다.

제안 방안은 기본적으로 user managed cookie이다. 하지만, 4장에서 소개될 제안 방안의 확장은 server managed cookie이다. 제안 방안에서는 X.509 인증서를 사용함으로써, 적절한 공개키 암호화 및 전자서명의 수행이 가능하다. 또한, 사용자가 서버로 로그인 할 때, 패스워드 기반의 메커니즘을

도입하여 종단 시스템 위협 및 쿠키 획득 위협을 방지할 수 있다.

Ⅲ. 공개키 인증서 기반의 보안 쿠키 설계

본 장에서는 공개키 인증서 기반의 보안 쿠키 구조를 제안하고, 이에 적용되는 보안 프로토콜에 대해서 정의한다.

1. 용어 정리 및 보안 쿠키의 구조

본 논문에서 사용되는 보안 프로토콜의 기술을 위해서 아래의 용어를 사용한다.

- C, S : 클라이언트(사용자)와 웹서버를 나타내는 식별자
- $Passwd$: 사용자가 웹서버에 로그인을 하기 위해서 사용하는 패스워드를 나타내는 식별자
- PR_X, PU_X : 공개키 암호 시스템에서의 통신 개체 X의 private key 및 public key
- SK : 대칭키 암호 시스템을 위한 secret key
- T_X : 통신 개체 X의 타임 스탬프
- $H(m)$: 메시지(m)를 일방향 해쉬 처리
- $E_K(m)$: 메시지(m)를 키(K)로 암호화
- $SIG_K(m)$: 메시지(m)를 키(K)로 전자서명

본 논문에서 제안하는 보안 쿠키셋(secure cookies set)의 전체적인 구조는 그림 2와 같다.

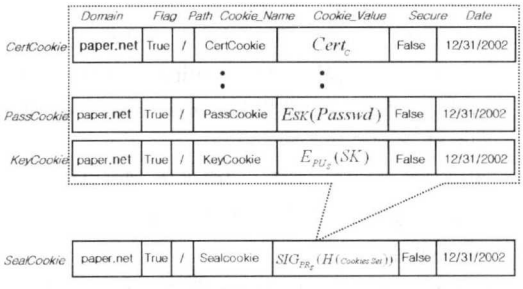


그림 2. 보안 쿠키셋의 구조

- CertCookie : 클라이언트의 사용자 인증서를 쿠키값으로 가진다.
- PassCookie : 사용자의 패스워드를 SK로서 암호화한 값을 쿠키값으로 가진다.
- KeyCookie : SK를 서버의 공개키(PU_S)로서 암호화한 값을 쿠키값으로 가진다.
- SealCookie : 위에서 생성된 쿠키들을 해쉬 처리하여, 서버의 개인키(PR_S)로서 전자서명한 값을 쿠키값으로 가진다.

Remark 제안하는 보안 쿠키에서 CertCookie, PassCookie, KeyCookie와 같은 쿠키들을 쿠키셋(cookies set)이라고 하며, 위의 쿠키셋과 SealCookie의 집합을 보안 쿠키셋(secure cookies set)이라고 정의한다. 어떤 서버의 관리자는 자신의 목적에 맞는 새로운 쿠키들을 생성한 후, 안전하게 보안 쿠키셋에 추가가 가능하다.

2. 보안 쿠키셋 발행

그림 3은 제안 방안에서의 보안 쿠키 발행 과정을 간략하게 보이고 있다. 제안 방안의 적용을 위해서, 웹서버와 사용자는 사전에 공인된 인증기관(CA : Certificate Authority)으로부터 인증서를 발급 받아서 소유하고 있음을 가정한다.

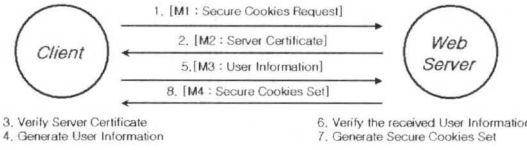


그림 3. 보안 쿠키셋의 발행

클라이언트가 웹서버로 쿠키 발행을 요청하면, 웹서버는 자신의 공개키 인증서로 구성된 $M2$를 클라이언트에게 전송한다.

$$M2 : Cert_S$$

클라이언트는 웹서버 인증서의 유효성 검증을 통하여 웹서버를 인증하고, 웹서버의 공개키를 획득하게 된다. 그리고, 생성될 쿠키값들의 암호화를 위한 SK를 생성하고, 웹서버에서 로그인시에 사용될 패스워드($Passwd$)를 결정한 후에 $M3$를 웹서버에게 전송한다. 또한, 클라이언트는 앞으로 SK를 사용하지 않으므로 $M3$의 전송후에 SK를 삭제한다.

$$M3 : Cert_C,\ E_{PU_S}(Passwd \| SK \| SIG_{PR_C}(Passwd \| SK))$$

웹서버는 $M3$내의 클라이언트 인증서를 검증함으로써 클라이언트를 인증하고, 클라이언트의 공개키 PU_C를 구한다. 웹서버는 PU_S로 암호화된 부분을 자신의 PR_S로 복호화한다. 클라이언트의 인증서에서 구한 PU_C를 사용하여 $M3$내의 서명값의 유효성을 검증한다. $M3$의 모든 검증 절차가 끝나면, 웹서버는

$$M4 : CertCookie \| \cdots \| PassCookie \| KeyCookie \| SealCookie$$

를 클라이언트에게 전송한다. 여기서, *M4*에서 SealCookie의 쿠키값은 아래와 같이 계산한다.

$$SIG_{PR_S}(H(CertCookie\|\cdots\|KeyCookie))$$

3. 보안 쿠키셋을 통한 로그인

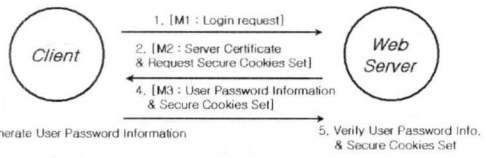


그림 4. 보안 쿠키셋을 통한 로그인

그림 4는 제안하는 보안 쿠키셋을 사용하여, 안전한 로그인 과정을 보이고 있다. 클라이언트가 로그인 요청을 하면, 웹서버는 *M2*를 생성해서 사용자에게 전송한다.

$$M2 : Cert_S \| T_S$$

*M2*를 수신한 클라이언트는 웹서버 인증서의 유효성을 검증한 후, 웹서버를 인증하고, PU_S를 얻게 된다. 클라이언트는 웹서버에 로그인을 위하여, 사용자에게 *Passwd*를 입력 받고, 이 값을 수신한 T_S 값과 함께 웹서버의 공개키로 암호화한다. 그리고, *M3*를 구성하여 웹서버에게 전송한다.

$$M3 : E_{PU_S}(Passwd, T_S) \| CertCookie \| \cdots \| KeyCookie \| SealCookie$$

*M3*를 수신한 웹서버는 PR_S를 통해서, 자신의 PU_S로 암호화된 것을 복호화하여, *Passwd*와 자신이 전송한 T_S를 획득한다. *M3*에서 획득된 T_S와 *M2*에서 전송한 T_S를 비교하여, 두 값이 일치하면, 수신한 *M3*가 재전송이 아님을 확인한다. 그리고, 수신한 쿠키셋을 해쉬처리 및 전자서명을 하여, SealCookie의 쿠키값과 일치하는 가를 검증한다. 만약, 일치할 경우에는 수신된 쿠키는 사용자의 컴퓨터에서 공격자로부터 변경되지 않았으며, 또한 자신(웹서버)이 발급했던 보안 쿠키셋임을 확인하게 된다. 웹서버는 자신의 PR_S를 사용하여, KeyCookie의 쿠키값을 복호화하여 사용자의 *SK*를 획득한 후, PassCookie의 쿠키값을 복호화하여, *M3*내의 *Passwd*와 PassCookie내의 *Passwd*가 일치하는지를 비교한다. 만약 두 값이 동일할 경우, 정당한 사용자가 전송한 보안 쿠키셋임을 확인한다. 그리고, CertCookie의 쿠키값에서 클라이언트 인증서를 획득한 후, 유효성을 검증하여 최종적으로 클라이언트를 인증하게 된다.

4. 제안된 보안 쿠키의 시큐리티

본 논문에서 제안된 보안 쿠키는 아래와 같은 암호학적인 특징을 가진다.

- 상호 인증 : 보안 쿠키 발행 및 로그인시에 상호간의 인증서 교환 및 검증절차를 통해서 상호 인증을 제공하며, 기 제안된 방안[6]에서의 공개키 분배 문제를 해결한다.
- 기밀성 : 사용자가 보안 쿠키 발행시에 생성한 *SK*를 통해서 CertCookie를 제외한 각 쿠키의 쿠키값들에 대한 기밀성을 제공한다.
- 무결성 : SealCookie의 쿠키값 생성을 위해 일방향 해쉬함수를 적용함으로써, 쿠키셋들의 무결성을 제공한다.
- 출처인증 : SealCookie의 쿠키값 생성을 위해 전자서명을 도입함으로써, 보안 쿠키셋의 서명자를 유일하게 식별할 수 있다.

제안하는 보안 쿠키는 2.1절에서 소개된 알려진 쿠키의 보안 위협을 다음과 같이 해결하였다.

- 네트워크 위협 : 기밀성, 무결성, 출처인증으로써 기본적인 방어가 가능하며, 보안쿠키를 통해서 로그인을 시도할 때, 보안 쿠키셋을 통한 로그인에서 사용된 타임스탬프를 통해서 재생공격을 막을 수 있다.
- 종단 시스템 위협 : 공격자는 쿠키 값의 변경은 불가능하며, 복사는 가능하다. 하지만, 사용자의 패스워드를 알아야만 획득한 보안 쿠키셋의 사용이 가능하므로, 사용자 가장 공격을 막을 수 있다.
- 쿠키 획득 위협 : 공격자는 웹서버의 개인키(private key)를 알아야만 사용자의 암호화된 쿠키값을 알 수가 있다. 또한, 보안 쿠키값 자체를 획득하더라도 사용자의 패스워드를 알아야만, 악의적인 목적으로 사용이 가능하다.

Ⅳ. 보안 쿠키셋의 확장

본 장에서는 4장에서 제안되었던 보안 쿠키셋을 이용하여, 인증된 세션 트래킹을 위한 기법과 단일 인터넷 도메인내에 존재하는 웹서버간의 로그인을 위한 방안을 제안하고자 한다.

1. 단일 서버에서의 인증된 세션 트래킹

사용자가 단일 웹서버내의 다른 페이지로 이동을 하거나 동일 세션에서의 새로운 connection을 생성

할 시(예, Internet Explorer의 경우 ''ctrl+n''키를 누른 경우)에 세션 트래킹을 수행하기 위해서, 보안 쿠키셋을 통한 로그인 절차가 적절이 수행되고 난 후, 웹서버는 STCookie를 클라이언트에게 발행한다.

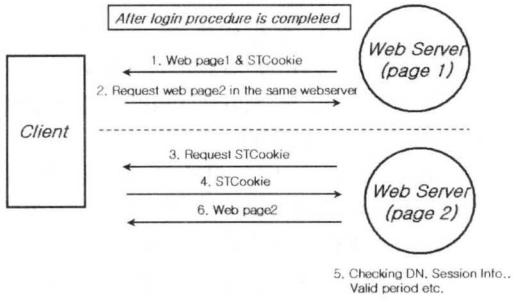


그림 5. STCookie를 통한 인증된 세션 트래킹

그림 5는 단일 웹서버내에서 클라이언트가 여러 웹페이지로 이동을 할 경우에 필요한 세션 트래킹 절차를 보여주고 있다. STCookie는 클라이언트의 세션트래킹을 목적으로 하며, 사용 목적상 단일 세션 동안 비교적 단기간 동안 사용하는 쿠키이므로 기존의 보안 쿠키셋에 포함되지 않는다. STCookie는 아래와 같은 쿠키값을 가진다.

$$SI \| C_{DN} \| VP \| SIG_{PR_S}(H(SI \| C_{DN} \| VP))$$

- SI : HTTP 세션 관련 정보
- C_{DN} : 클라이언트 인증서내의 클라이언트 DN (Distributed Name)
- VP : STCookie의 유효기간으로, 적용되는 시스템에 따라서 적절한 시간을 분배할 수 있다. 만약 유효 기간이 짧은 경우는 재생공격의 위험이 낮지만, 사용자가 로그인 절차를 너무 자주 실행해야 하는 번거러움이 생긴다. 또한, 유효 기간이 긴 경우는 사용자 입장에서는 편리하지만, 재생공격의 위험이 높아진다.

2. 다중 서버간의 인증된 로그인

어떤 동일한 인터넷 도메인에 존재하는 웹서버들은 암호학적인 쿠키를 사용함으로써, 특별한 부가적 절차가 없이 클라이언트가 접속하길 원하는 웹서버로 안전하게 접근이 가능하다.

Assumption 클라이언트가 보안 쿠키셋을 통하여 안전하게 어떤 웹서버에 로그인 된 상태이다. 각 웹서버(S_i, $1 \le i \le n$)들은 동일한 인터넷 도메인내에 존재한다. 또한, 각 웹서버(S_i)들은 상호신뢰함을 가정한다. 따라서, 각자 서로의 공개키 (PU_{S_i})를 신뢰하고 있다.

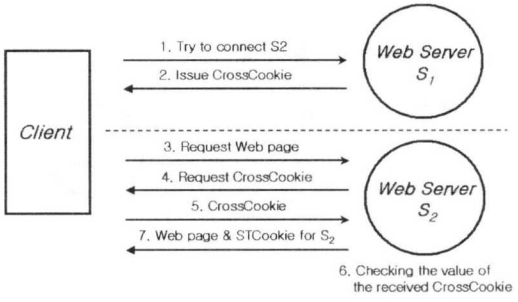


그림 6. CrossCookie를 통한 다중 서버간의 인증된 로그인

클라이언트가 현재 접속중인 웹서버(S_1)에서 다른 웹서버(S_2)로의 이동을 하고자 할때(즉, 사용자가 하이퍼 링크를 클리했을 때), 클라이언트가 S_2로 간단히 로그인이 성공하도록, S_1은 클라이언트에게 CrossCookie를 발행한다. CrossCookie는 아래의 쿠키값을 가진다.

$$S_{i\,DN} \| S_{j\,DN} \| C_{DN} \| VP \| \cdots \| SIG_{PR_S}(H(S_{i\,DN} \| S_{j\,DN} \| C_{DN} \| VP))$$

- $S_{i\,DN}$: 클라이언트가 현재 접속하고 있는 웹서버의 인증서내에 기술된 웹서버(S_i) DN, $1 \le i \le n$.
- $S_{j\,DN}$: 클라이언트가 접속하고자 하는 웹서버의 인증서내에 기술된 웹서버(S_j) DN, $1 \le j \le n, j \ne i$.
- C_{DN} : 클라이언트 인증서내에 기술된 클라이언트 DN.
- VP : CrossCookie의 유효기간. 사용자가 다른 웹서버로 이동하는 시간은 짧기 때문에, STCookie의 유효기간과는 달리 짧은 유효기간을 갖는다.

그림 6은 웹서버(S_1)에서 하이퍼 링크된 다른 웹서버(S_2)로 이동시에 CrossCookie의 사용에 대해서 보여주고 있다. CrossCookie를 통하여, 클라이언트는 웹서버들 사이에 안전하게 로그인이 가능하다.

V. 보안 쿠키의 구현

본 장에서는 제안된 보안 쿠키에 대한 구현 및 성능 평가 결과를 소개한다.

1. 구현 환경

제안된 보안 쿠키는 표 1과 같은 환경하에서 구현되었다.

표 1. 구현을 위한 시스템 환경과 암호 알고리즘

Language	Java(JDK 1.3) JSP CryptixJCE Java Web Start V1.01
System Hardware	Pentium III 866 MHz 256MB RAM
Web Server	Apache 1.3.19(for Win32) Jakarta Tomcat 3.2.1
Network	LAN
Cryptographic Algorithm	Public key alg. : RSA 1024bit One way hash alg. : MD5 Symmetric key alg. : DES

웹서버는 일반적인 HTML과 JSP로서 구현되었으며, 사용자와 웹서버간의 보안 프로토콜을 수행할 사용자 에이전트 프로그램를 위해서 Java Web Start기술을 적용시켰다. 구현의 테스트를 위해서 사용자와 웹서버의 인증서는 자기 서명된 인증서(self-sigend certificate)를 사용하였다[2],[4],[5],[7].

2. 구현 결과 및 성능 비교

사용자는 웹서버의 홈페이지에 접속하여, 쿠키 발행을 위한 웹페이지로 이동하면, 자동적으로 "쿠키 발행 에이전트" 프로그램이 실행된다. 사용자는 웹서버에서 사용자 식별을 위해서 사용될 패스워드와 인증서 사용 권한 확인을 위한 패스워드를 입력한 후에 접속을 시도한다. 보안 쿠키셋 발행 절차가 성공적으로 수행되면, 클라이언트는 보안 쿠키를 발급받게 된다.

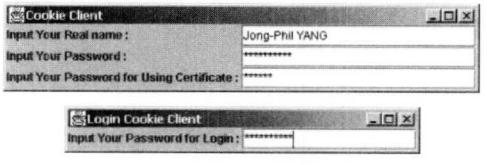


그림 7. 보안 쿠키셋 발행 및 로그인을 위한 사용자 프로그램

또한, 사용자는 보안 쿠키를 사용하여 로그인하는 웹페이지로 이동하면, 자동적으로 "보안 쿠키 로그인 에이전트" 프로그램이 실행된다. 이 때, 사용자는 서버에서의 사용자 단순 식별을 위한 패스워드만 입력한 후에 접속을 시도하게 된다. 보안 쿠키셋을 통한 로그인 절차가 적절히 수행되면, 사용자는 안전하게 로그인을 성공한다. 그림 7은 쿠키 발행 및 로그인을 위한 에이전트 프로그램이며, 그림 8은 클라이언트의 시스템에 저장된 보안 쿠키셋을 보이고 있다.

그림 8. 클라이언트의 로컬 하드 드라이브에 저장되는 보안 쿠키셋

본 논문에서 구현된 보안 쿠키 시스템의 성능 평가를 위해 SSL의 세션 설정에 필요한 시간을 측정하였다. 성능 분석을 위해서 사용된 SSL은 JSSE 1.0.2의 라이브러리를 사용했으며, 클라이언트와 서버간의 상호 인증 및 fullhandshake가 수행하는 환경에서 테스트를 하였다[7]. 표 2은 제안 방안과 SSL의 성능 테스트 결과를 보여주고 있다.

표 2. SSL과 제안 방안의 성능 비교

보안 쿠키셋 발행 시간	**6.125 sec**
보안 쿠키셋을 통한 로그인 시간	**5.796 sec**
SSL의 fullhandshake 시간	**7.721 sec**

VI. 결 론

본 논문은 일반적인 쿠키가 가지고 있는 보안 위협요소를 모두 해결하며, 기밀성, 상호인증, 무결성 등을 제공하는 새로운 쿠키 모델을 설계하고 구현하였다. 제안 쿠키 모델은 안전한 웹서핑을 수행하기 위하여, 요구되는 여러 가지 보안성을 충족시킨다. 또한, CertCookie, PassCookie, KeyCookie와 같은 기본적인 보안 쿠키들 이외의 접근제어와 기타 사용자 관련 정보를 위한 새로운 부가적인 쿠키들도 *SK*를 사용한 암호화를 통해서 기밀성을 보장한

상태로 쿠키셋에 추가시킴으로써, 웹서버에서 요구되는 새로운 기능을 추가할 수 있는 확장성을 제공하고 있다. 제안된 쿠키 모델을 통한 안전한 세션 트래킹과 인터넷 도메인내의 웹서버들 사이에 CrossCookie를 사용함으로써, 사용자는 접속하길 원하는 다른 웹서버로 안전하게 로그인을 할 수 있다.

본 논문에서 제안된 쿠키 모델은 웹서버에서 관리해야할 사용자 관련 자료를 안전하게 사용자들의 컴퓨터에 쿠키의 형태로 보존함으로써, 웹서버 데이터베이스의 유지보수 비용을 현저하게 감소시킬 것으로 기대된다.

참 고 문 헌

[1] *http://www.certcc.or.kr/advisory/ka2000/ka2000-041.html*

[2] *http://www.cryptix.org/products/jce/index.html*

[3] *http://www.netscape.com/newsref/std/cookie_spec.html*

[4] *http://java.sun.com/products/javawebstart/developers.html*

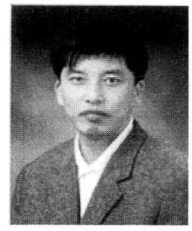
[5] *http://java.sun.com/products/jsp/download.html*

[6] Joon S. Park and Ravi Sandhu, "Secure Cookies on the Web" *IEEE Internet Computing, Volume : 4 Issue : 4*, July-Aug. 2000.

[7] Scott Oaks, *"Java Security, 2nd Edition"*, O'Reilly. 2001.

[8] V. Khu-smith and C.J. Mitchell, "Enhancing the security of cookies", in: K. Kim (ed.), *Information Security and Cryptology - ICISC 2001 - Proceedings of the 4th International Conference*, Seoul, Korea, December 2001, Springer-Verlag (LNCS 2288), Berlin (2002), pp.132-145.

이 경 현(Kyung-hyune Rhee) 정회원



1982년 2월 : 경북대학교 수학교육과 졸업

1985년 2월 : 한국과학기술원 응용수학과 석사

1992년 8월 : 한국과학기술원 수학과 박사

1985년 2월~1993년 2월 : 한국전자통신연구소 연구원, 선임연구원

1993년 3월~현재 : 부경대학교 전자컴퓨터정보통신공학부 전임, 조교수, 부교수

<주관심 분야> 암호이론, 암호프로토콜, 네트워크보안, 이동네트워크, 그룹키 관리

양 종 필(Jong-Phil Yang)

1999년 2월 : 부경대학교 전자계산학과 졸업

2001년 8월 : 부경대학교 전자계산학과 석사

2002년 3월~현재 : 부경대학교 전자계산학과 박사과정

<주관심 분야> 네트워크 및 시스템 보안 기술, 공개키 기반 구조, 비밀 분산