사용자의 쿠키를 이용한 웹 페이지의 암호화

한소희, 조동섭
이화여자대학교

Web Page Customizing Using User' s Cookie and Serverside String Encryption

So-Hee Han, Dong-sub Cho
Ewha Woman's University

Abstract - 대부분의 인터넷 사용자들은 본인도 모르는 사이에 쿠키의 사용을 허용하고 있으며 이에 따른 각종 해킹과 트랙킹의 위험도 아울러 묵과하고 있는 실정이다. 하지만 역시 대부분의 사용자들은 쿠키의 위험성에도 불구하고 쿠키의 사용을 멈추지 않을 것이다. 쿠키로 인해 향상되는 인터넷 속도의 이점을 포기하지 못하기 때문이다. 쿠키를 비롯해 해킹과 트랙킹의 위험을 내포하고 있는 또 하나의 인터넷 컨텐츠는 HTML 문서이다. 현재 HTML 문서는 소스보기를 통해 원본의 모든 출처를 노출하고 있다. 따라서 쿠키만큼 해킹과 트랙킹의 기회를 제공하고 있는 셈이다. 본 논문은 이렇게 불가피하게 사용되는 쿠키 필연적인 성격을 이용하여 웹 페이지의 암호화를 제안한다. 특히, 웹 서버와 클라이언트 환경으로 한정하여 웹 서버가 클라이언트에게 웹 페이지를 전송할 시 쿠키를 키로 한 알고리즘을 사용하여 암호화를 한 후 전송함으로써 해킹이나 트랙킹의 위험을 최대한 낮추고자 한다.

1. 서 론

급속한 인터넷의 확산은 컴퓨터나 인터넷에 대한 전문적인 지식이 없어도 누구나 넷상의 모든 정보를 관람하고 또한 사용자의 동의 없이도 사용가능하게 만들었다. 따라서 인터넷 화면에 한번 소개된 자료는 모든 사용자가 동시에 소유주가 되며 이 자료를 악용한 사례들은 법적 다툼으로, 심지어는 국가외교 문제로까지 번지고 있다. 현재도 인터넷 선을 타고 도는 수백, 수천만 개의 웹 페이지들은 그들의 상태를 그대로 노출하며 각종 해킹과 트랙킹의 좋은 기회를 제공하고 있다. 쿠키 역시 강도 높은 해킹과 트랙킹의 위험을 안고 있다. 웹 페이지에 나타나는 일반적인 컨텐츠 말고도 아이디나 비밀번호와 같은 중요한 정보도 때로 노출되므로 쿠키 보호의 중요성은 간과될 수 없다. 그렇다고 이러한 위험성을 전면에 세워 쿠키의 사용을 포기하는 것도 불가능하다. 최신, 최고속도를 외치는 현재의 인터넷 환경에서 쿠키를 사용하지 않는 것은 컴퓨터 앞에서의 작업 시간을 보장할 수 없는 것과 다름없기 때문이다. 또한 쿠키로 인해 가능한 인터넷 쇼핑몰의 장바구니나 각종 사이트들의 자동 로그인 기능은 쿠키의 부재로 인한 불편함을 참을 수 없게 만들 것이다.

최근에 인터넷에서 전 세계적인 열풍을 주도한 것 중의 하나는 블로그이다. 블로그는 개인만의 자료들을 전시하고 싶어 하는 사람들의 심리를 적극 활용한 대표 사례이다. 그러나 최근에는 지적재산 운운하며 지적재산권 침해 소송으로 불거지게 만든 장본인이기도 하다. 사람들은 개인의 컨텐츠 또는 본인과 관련된 넷상의 자료들에 크게 의미를 부여하며 함부로 사용, 오용되는 일이 발생하기 않도록 요구한다.

위와 같은 상황을 배경으로 본 논문에서는 클라이언트마다 고유하게 전송되는 쿠키에 해쉬함수를 적용, 키로써 활용하여 웹 페이지를 암호화하는 것을 제안한다.

본 논문의 구성은, 2장에서는 관련연구를 언급하고, 3장에서는 본 논문의 핵심인 웹 페이지의 암호화 방식을 제안, 4장에서는 결론과 향후과제를 제시함으로써 마친다.

2. 본 론

2.1 쿠키의 사용

쿠키는 그 활용도의 중요성에도 불구하고 인터넷 사용자들의 무관심의 범위를 벗어나지 못하고 있다. 모든 웹 서버들은 클라이언트가 접속해 오면 예외 없이 클라이언트 컴퓨터 내에 쿠키를 저장한다. 웹 브라우저는 임시 인터넷 파일을 통해서 쿠키 자동 생성기능을 제공하기도 한다. 이러한 쿠키의 내용은 클라이언트의 접속 기록 뿐 아니라 클라이언트가 접속한 페이지의 컨텐츠 내용을 포함하기 때문에 일종의 캐쉬역할을 하게 된다. 그리고 사용자는 이러한 쿠키의 매력을 포기할 수 없을 것이다. 또한 위 내용뿐 아니라 아이디나 패스워드 같은 극비의 개인정보도 담고 있기 때문에 신상정보 유출에 대한 심각한 위험을 안고 있다. 쿠키를 암호화해서 전송하는 것이 일반적이지만 암호화하지 않고 그대로 전송하는 사이트들의 경우 사용자는 자신도 모르게, 접속하는 그 순간에 바로 자신의 정보를 노출하게 된다.

2.2 웹 페이지의 암호화 요구

웹 페이지를 구성하는 컨텐츠들은 HTML 태그와 각종 스크립트 언어로 작성된다. 따라서 일반 평문으로 구성된 웹 페이지는 그 내용뿐 아니라 자료의 출처도 쉽게 읽을 수 있도록 되어있다. 아무런 권한 없이 누구에게나 읽기가 가능하기 때문에 쉽게 정보의 악용이 가능하다. 이에 인터넷 사용자들은 웹 페이지의 전체 혹은 일부가 보이지 않기를 원하게 되었다. 그 하나의 해결책으로 현재 몇몇 인코더 툴들이 소개되어 있다. 그러나 이러한 툴들은 암호화라고 하기에는 비교적 단순하게, 문자를 다른 문자로 매치하는 수준이기 때문에 디코딩 또한 간단하여 더 수준 높은 암호화 방식이 요구된다.

3. 웹 페이지의 암호화

3.1 웹 서버 클라이언트 환경

최근 발생하고 있는 디지털 저작권 문제는 비단 전문적인 해커뿐만 아니라 일반적인 인터넷 사용자들에게도 책임소재를 묻고 있다. 해커나 트랙거들은 웹 서버에 접속하는 클라이언트의 접속 기록, 관람한 컨텐츠들을 추적하여 중요한 신상정보들을 빼내간다. 또한 일반 사용자들도 웹 페이지의 원본 출처를 쉽게 읽을 수 있기 때문에 원본 자료를 그대로 복사, 유포하여 사회적 문제를 일으키기도 한다. 이러한 문제들은 근본적으로 정보를 제공하고 받아보는 웹 서버 클라이언트 환경에서 주고받는 웹 페이지의 노출에서 기인한다. 따라서 본 논문에서는 웹 서버 클라이언트 환경에 중점을 두고 웹 페이지의 암호화를 설명하고자 한다. 다음은 이 연구에서 사용하고자 하는 웹 서버와 클라이언트 프로그램, 알고리즘, HTML 문서내의 컨텐츠를 보여주고 있다.

<표 1> 개발환경

웹 서버 프로그램	MICROSOFT FOUNDATION CLASS LIBRARY : httpsvr
클라이언트 프로그램	MICROSOFT FOUNDATION CLASS LIBRARY : Spider
해쉬 알고리즘	SHA-1
암호화 알고리즘	DES

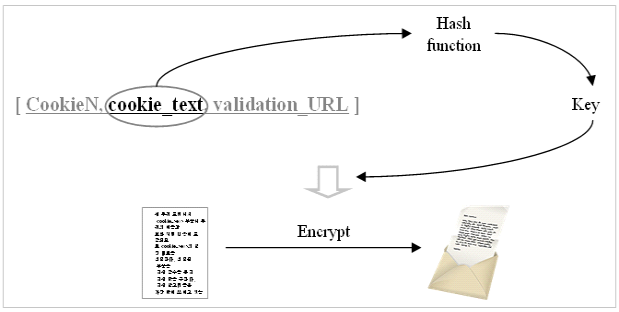
```
<!DOCTYPE html><html><head><meta http-equiv="Content-Type" content="text/html;
charset=UTF-8"><title>Google</title><style type="text/css"><!–
#svc-tab .bgp-fr{background:transparent url(/ig/f/KoR1V7i-xBw/intl/ALL_kr/tab_sprite_all.gif)
0 no-repeat} #svc-toolbar .bgp-fr{background: transparent url
(/ig/f/JB08UxZqEXw/intl/ALL_kr/svc_sprite_all.gif) 0 0 no-repeat} .ttv{background: transpare
url(/ig/f/JB08UxZqEXw/intl/ALL_kr/svc_sprite_all.gif) 0 0 no-repeat}
–></style>
<script src="/ig/f/4GmrZsj9vDQ/ig.js"></script><link rel="stylesheet" href="/ig/f/-
n3eOqyLOCA/intl/ALL_kr/homepage.css" type="text/css"></head><body onload="_KO.init
()"><div id="wrapper"><div id="guser"><a href="/url?
sa=p&pref=ig&pval=3&q=/ig">iGoogle</a><span class="separator">|</span><a
href="https://www.google.com/accounts/Login?continue=http://www.google.co.kr/&hl=ko">로
인</a></div><img id="logo" alt="
src="http://img0.gmodules.com/ig/f/oKstlUEg20s/intl/ALL_kr/logo.gif"><form
action="http://www.google.co.kr/search" name="f" ><script><!–
function qs(el) {if (window.RegExp && window.encodeURIComponent) {var ue=el.href;var
qe=encodeURIComponent(document.f.q.value);if(ue.indexOf("q=")!=-1){el.href=ue.replace(ne
RegExp("q=[^&]*"),"q="+qe);}else{el.href=ue+"&q="+qe;}}return 1;}
// –>
```

<그림 1> HTML 문서내의 컨텐츠

3.2 웹 페이지의 암호화 과정

웹 페이지 암호화를 위한 알고리즘으로 가장 널리 사용 중인 DES

대칭키 알고리즘을 적용한다. 클라이언트가 웹 서버에 접속을 요청하면 웹 서버는 클라이언트에게 쿠키를 제공한다. 웹 서버와 클라이언트는 각기 쿠키에 해쉬 함수를 적용해 키를 얻는다. 웹 서버는 이 대칭키로 웹 페이지를 DES 알고리즘으로 암호화 하여 클라이언트에게 전송한다. 클라이언트 역시 같은 대칭키로 전송받은 웹 페이지를 디코딩 한다. 다음은 위와 같은 암호화 과정을 나타낸 것이다.



〈그림 2〉 암호화 과정

3.3 대칭키 생성

DES 알고리즘에서 사용되는 대칭키는 쿠키를 통해 생성된다. 쿠키는 클라이언트마다 다른 내용을 포함하고 키로써의 크기도 4byte정도로 길지 않으므로 적당하다. 그러한 위에서 언급한 바처럼 쿠키의 노출은 트랙킹의 위험을 안고 있기 때문에 해쉬함수를 이용해 쿠키의 내용을 감 출 수 있도록 한다. 본 논문에서는 쿠키의 일정 비트를 추출하여 해쉬함수를 통해 해쉬값으로 전환하여 DES 알고리즘의 대칭키로 사용한다. 해쉬 알고리즘으로는 가장 널리 사용되는 SHA-1 알고리즘을 사용한다. 다음은 쿠키의 기본 포맷이다.

[CookieN(cookie_length, URL_length), cookie_text, validation_URL]

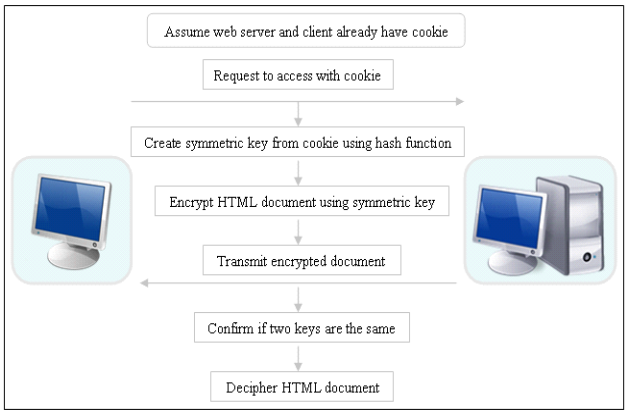
```
array<Byte>^data=gcnew array<Byte>( DATA_SIZE );
array<Byte>^ result;

SHA1^ sha = gcnew SHA1CryptoServiceProvider;
// This is one implementation of the abstract class SHA1.
result = sha->ComputeHash( data );
```

〈그림 3〉 SAH-1 알고리즘

3.4 웹 페이지 암호화 시나리오

웹 페이지를 암호화하기 위한 전체 과정을 다음의 시나리오로 설명한다. 먼저 클라이언트가 이미 웹 서버를 방문한 적이 있어 쿠키를 가지고 있다고 가정한다. 클라이언트가 쿠키를 가지고 웹 서버에 접속을 요청하면 웹 서버는 해당 쿠키를 가지고 해쉬함수를 적용해 DES 알고리즘을 위한 대칭키를 생성한다. 그리고 요청한 웹 페이지를 대칭키와 DES 알고리즘으로 암호화 한 후 클라이언트에게 전송한다. 클라이언트는 역시 해쉬함수를 적용해 대칭키를 만든 후 웹 서버로부터 받은 대칭키와 비교 후, 전송받은 웹 페이지를 복호화 한다. 이 과정이 다음 [그림 4]로 표현되었다.



〈그림 4〉 웹 페이지 암호호 시나리오

4. 결 론

본 논문에서는 쿠키의 사용, 웹 페이지의 노출 등을 배경으로 이로 인한 해킹, 크랙킹의 위험성을 문제로 삼고 웹 페이지의 암호화를 제안하였다. 제안의 핵심은 쿠키를 사용하여 대칭키를 생성, 알고리즘을 적용해 웹 페이지를 암호화 하는 것이다. 위에 언급하였듯이 쿠키의 필연성은 키로써의 적절한 사용을 허용하고, 이에 해쉬 알고리즘을 적용하여 키로써의 역할을 확고히 해준다. 또한 웹 페이지의 암호화의 요구는 현재와 같은 인터넷 자료의 난무함 속에 시의적절한 문제라고 여겨진다.

향후에는 위 논문에서 제안한 암호화 방식을 구체적으로 구현하고 발전시키는 것으로 연구를 진행할 것이다.

[참 고 문 헌]

[1] 최향창, 최은복, 노봉남, "쿠키 보호 시스템 설계", 정보과학회, 2002년.

[2] 김기성, 김광, 허신, "이기종 시스템에서 안저한 데이터 전송을 보장하는 웹 보안 모듈의 설계 및 구현", 정보과학회 논문지 제32권 제 12호, 2005년.

[3] David M. Kristol, " HTTP Cookies: Standards, privacy and politics", ACM Transactions on Interner Technology(TOIT), Volulme1, Issue2, 2001년.

[4] Daniel Lin, Michael C. Loui, "Taking the byte out of cookies: privacy, consent, and the Web", ACM SIGCAS Computers and Society, Volume 28 Issue 2, 1998.

[5] MSDN Library http://msdn2.microsoft.com/ko-kr/library/system.security.cryptography.sha1.aspx