쿠키파일의 보안성을 향상하기 위한 보안영역 설계

서희석† · 최요한††

요 약

쿠키는 단순한 텍스트 파일로 사용자가 웹 서비스를 제공 받은 서비스 내용이 기록되어 있다. 쿠키파일에 기록 되는 정보 중에는 사용자의 개인정보도 포함된다. 개인정보가 기록된 쿠키파일을 공격자 획득하게 되면 사용자의 개인정보 유출로 인해 금전적인 피해까지 입을 수 있다.

본 논문에서는 쿠키 파일의 보안성 향상을 위해 제시된 관련연구와 쿠키파일의 낮은 보안성으로 인해 발생할 수 있는 취약성을 살펴본다. 관련 연구와 쿠키 파일의 취약성 분석을 통해 쿠키 파일을 안전하게 보관할 수 있는 보안영역에 대한 설계와 효율적인 보안영역을 제시를 위해 보안영역을 구현할 수 있는 방법에 대해서 알아본다. 또한, 성능평가를 통해서 보안성을 확인해 보았다. 보안영역을 통해서 사용자 PC에 저장된 쿠키파일의 보안성을 향상시켜 사용자의 개인정보를 안전하게 유지할 수 있을 것이다.

주제어 : 웹, 쿠키파일, 보안영역, HTTP, 개인정보

Designing on Security zone to improve Cookie File Security level

HeeSuk Seo† · YoHan Choi††

ABSTRACT

Cookie is simple text file, which contains records of web service which provided to user. some of data included in Cookie has user's private information. When attacker has Cookie which included user's private information, will causing financial losses.

In this paper we designed security section which can improve vulnerable Cookie's security level. Through research and vulnerability analysis of Cookie file, we find out how to implement security area to offer efficient security area and design security area for cookie file. Also we checked security level to performance evaluation. Through this security level, we can keep user's private information secure using Cookie's improve security level which stored in user's personal computer.

Keywords : Web, Cookie File, Security Zone, HTTP, Personally Identifiable Information

† 정 회 원: 한국기술교육대학교 컴퓨터공학부 교수
†† 준 회 원: 한국기술교육대학교 컴퓨터공학부 학사과정 (교신저자)
논문접수: 2011년 09월 16일, 심사완료: 2011년 10월 08일, 게재확정: 2011년 10월 17일
* 본 논문은 한국기술교육대학교 교육연구진흥비지원 프로그램의 (일부) 지원에 의하여 수행되었음

1. 서 론

국내 네트워크는 해외와 비교해 볼 때 급속도로 발전하고 있다. 이러한 네트워크의 발전은 개인, 기업 그리고 국가 간 정보 교류의 빈도를 증대시켰다.

또한 네트워크를 이용한 다양한 웹 서비스가 많은 사용자에게 제공되고 있다. 하지만 서비스 제공과정에서 많은 취약성들이 발견되면서 접근제어 및 보안 시스템에 대한 필요가 증가되었다.[1]

현재 웹 환경에서 제공되는 전자상거래 서비스의 경우 HTTP프로토콜을 이용하는 HTML로 작성되어 있다. HTTP프로토콜은 단순하지만 이전 세션의 연결 상태를 유지할 수 없다는 단점이 존재한다. 이러한 단점을 해결하기 위해서 쿠키파일이 도입되어 많은 웹 서비스에 적용되어 사용되고 있다.

또한 웹 서비스를 이용한 전자상거래의 중요성이 높아지면서 사용자의 결제 정보를 쿠키파일에 저장하는 웹 서비스의 수가 증가했다.[2] 지불 서비스를 제공하기 위해 신용카드 및 개인정보가 쿠키파일에 저장하면서 쿠키파일의 노출을 방지할 수 있는 안전한 보안 시스템이 필요하게 되었다.[3]

웹 서비스에서 사용하고 있는 쿠키파일은 평문 형태의 정보가 텍스트파일로 저장된다. 이렇게 평문 형태로 저장되는 쿠키파일은 공격자의 접근이 매우 쉽다. 이러한 취약점으로 인해 쿠키파일을 악용하는 범죄가 증가하고 있으며, 그 피해가 이용자에게 전가 되는 사례가 빈번히 발생되고 있다.

2. 관련 연구

쿠키 파일을 공격자로부터 안전하게 보호하기 위해서 다양한 방법의 연구가 진행되었다. 제시된 연구 중 안전한 쿠키는 사용자의 인증을 제공하면서 쿠키의 비밀성과 무결성을 제공하여 공격자의 공격이 쿠키에 작성된 내용을 확인 할 수 없어야 하고, 공격자가 위조 하였을 때 이를 판단할 수 있어야 한다. 이러한 쿠키는 암호학적인 기법이 적용되어야 한다.[4]

2.1 인증 제공을 통한 쿠키파일 보호

웹 서버에서 사용자 인증(User Authentication)을 하는 방법은 사용자의 IP를 기반으로 한 방법 이외에도 패스워드를 기반으로 한 인증, 전사서명을 기반으로 한 인증 방법 등이 있다. 주소기반 인증은 사용자의 IP주소를 쿠키에 저장하여 인증하는 방법이다. 쿠키 파일에 저장된 정보를 사용하기 이전에 주소쿠키를 통해 상대방을 인증하고 인증이 완료되면 통신을 한다. 이 방법은 주소 스푸핑(Sniffing) 공격[5]을 통해 주소를 위조할 수 있기 때문에 확실하게 안전한 인증이 될 수 없다.

다음으로 패스워드를 기반으로 한 인증은 웹 서버의 패스워드를 사용자 PC에 해시 값으로 저장한 후, 쿠키를 사용하기 이전에 해시 값을 웹 서버에 전달하여 웹 서버와 사용자의 패스워드 해시 값을 비교하여 인증하는 방법이다. 공격자가 패스워드로 예상되는 값을 무작위로 입력하는 사전공격(dictionary attacks)을 통해서 패스워드를 알아내는 것을 막기 위해 패스워드를 해시함수를 거쳐 저장한다.

마지막으로 전자서명 기반 인증은 전자 서명방식에 의해 인증하는 방법으로 만약 공개키를 알고 있다면 DSA(Digital Signature Algorithm)와 RSA(Rivest, Shamir, Adleman) 알고리즘과 유사하게 전자서명을 할 수 있는데 이와 같은 방법에 의한 웹 서버와 사용자 간의 상호 인증 방법이다.

3. 웹 사용자 인증 및 쿠키의 취약성

3.1 사용자 인증

웹 서비스는 특성상 사용자의 정보를 웹 서버에 저장하고 있어야 한다. 이렇게 저장된 사용자의 정보에 접근하기 위한 인증은 주로 아이디와 패스워드를 사용자로부터 입력을 받거나 혹은 접속하고자 하는 웹 서버가 생성한 쿠키를 이용한다. 웹 서버는 사용자의 쿠키파일을 획득하여 서버에 저장된 사용자의 정보와 비교하여 사용자 인증을 수행한다.

사용자 인증을 위해 쿠키파일 내부 정보를 웹

서버에게 전달할 때에는 평문 형태로 전송된다. 평문형태로 쿠키파일의 정보가 전송되는 과정에서 공격자가 전송되는 쿠키파일의 정보를 획득하는 스니핑등의 네트워크 공격에 취약하게 된다.

3.2 쿠키 취약성을 이용한 공격 방법

대표적인 공격유형은 네트워크 공격, 종단시스템 공격, 쿠키획득 공격으로 분류된다.

3.2.1 네트워크 공격 취약성

네트워크 공격은 네트워크상에 전송되는 평문 형태의 쿠키가 노출되어 수정되는 것을 말한다. 이러한 공격은 서버와 브라우저에 설치된 SSL(Secure Socket Layer)프로토콜을 사용함으로 네트워크 공격을 저지할 수 있다. 하지만 이는 쿠키가 네트워크상에 전송되는 동안에만 보호될 수 있다는 단점이 있다.

3.2.2 종단 시스템 공격 취약성

종단 시스템 공격은 브라우저가 설치된 종단 시스템에 쿠키가 전송되어 평문형태로 하드디스크나 메모리에 존재하기 때문에 가능해진다. 이러한 쿠키는 사용자에 의해 쉽게 수정될 수 있고 다른 컴퓨터로 복사될 수 있다.

공격자는 내용을 수정한 쿠키파일을 이용하여 정당한 사용자로 위장하여 웹 서비스를 제공 받을 수 있다.

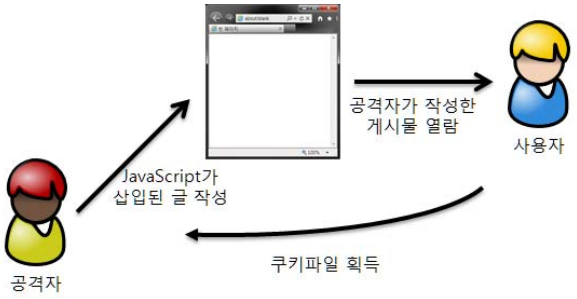
3.2.3 쿠키 획득 공격 취약성

쿠키 획득 공격은 웹 서비스에서 제공되고 있는 게시판을 악용하여 이루어진다. 현재 많은 웹 사이트에서 HTML을 이용하여 게시판에 글을 작성 할 수 있도록 제공하고 있다. HTML을 이용하여 글을 작성할 수 있게 되면서 사용자는 HTML 태그를 이용하여 다양한 형태의 글을 작성할 수 있게 되었고, 또한 JavaScript를 게시판에 사용할 수 있게 되었다.[4]

웹 서버는 게시판에 작성된 JavaScript 동작을 제한하지 않는 경우가 많다. <표 1>과 같은 JavaScript 코드를 게시판에 삽입하여 글을 읽는 사용자의 쿠키파일 정보를 공격자가 원하는 곳으로 전송 할 수 있는 위험이 존재한다.

<표 1> 쿠키 파일을 전송하기 위한 JavaScript

```
<script language=javascript>
  window.open(" 공격자의 웹 사이트 주소?cook=
                                "+document.cookie);
</script>
```



<그림 1> 쿠키 파일 노출 경로

<표 1>과 같은 JavaScript 코드가 삽입되어 있는 게시 글을 읽을 경우 사용자의 쿠키파일은 <그림 1> 같은 과정을 통해 공격자가 지정한 곳으로 전송된다.

이러한 공격들은 쿠키파일이 평문형태로 저장되어 있어 고도의 해킹기술이 요구되지 않기 때문에 더욱 빈번히 발생한다.

4. 쿠키파일 관리를 위한 보안영역 제시

쿠키 파일은 앞에서 설명한 다양한 공격에 노출 되어있다. 쿠키파일은 웹브라우저에서 지정한 기본 저장 위치에서 생성되고 관리된다. 즉, 쿠키파일이 한 장소에서 보관됨으로 인해 사용자의 쿠키 파일이 다량으로 유출될 수 있고, 유출된 쿠키파일을 통해서 개인정보가 침해 될 수 있다.

쿠키파일이 생성되고 관리되는 위치를 옮기거나 분산시켜 저장하면 한 번에 모든 쿠키파일이 유출 되는 상황을 막을 수 있을 것이다.

사용자의 웹 사이트 방문 실태를 살펴보면 사용자는 동시에 3개 이하의 사이트를 방문하는 경

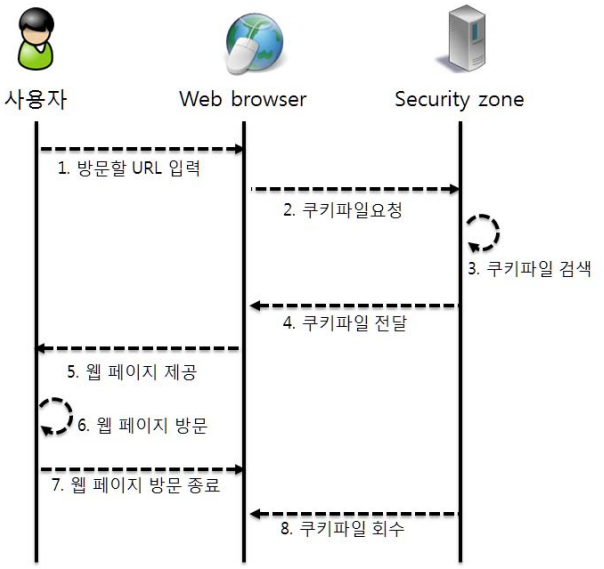
우의 빈도가 가장 높다. 즉, 클라이언트에 저장되어 있는 쿠키파일을 동시에 사용하는 개수가 3개 이상인 경우는 적다. 대부분의 웹 사이트에서 쿠키파일을 생성하지만 실제 사용자가 사용할 때에는 현재 이용 중인 웹사이트의 쿠키파일 이외의 쿠키는 웹브라우저에서 지정된 쿠키 폴더에 존재할 필요가 없다.

4.1 보안영역을 통한 쿠키 파일 관리 방법

보안영역은 웹 서버가 사용자의 PC에 생성한 쿠키파일의 보안성 향상을 위해 웹브라우저와 함께 동작해야 한다.

보안영역을 관리하는 커널은 쿠키파일을 안전하게 보관하기 위해서 웹브라우저가 지정한 폴더에 존재하는 쿠키파일을 보안영역으로 이동 시킨다. 이러한 사전 작업을 통해서 보안영역 이외에 장소에 쿠키파일이 존재하지 않도록 한다.

사용자는 방문하기 원하는 웹 서비스의 주소를 웹브라우저 주소 표시줄에 입력을 하게 된다. 웹브라우저는 사용자가 입력한 웹 서비스 주소를 보안영역을 관리하는 커널에게 전달한다. 커널은



<그림 2> 보안영역을 통한 쿠키 파일 관리 절차

보안영역에서 존재하는 쿠키파일 중 방문하려는 웹 서버가 작성한 쿠키파일이 있는지 존재여부를 확인한다.

보안영역에 사용자가 방문하려는 웹사이트가 작성한 쿠키파일이 존재하는 경우 웹브라우저에서 지정한 쿠키파일 폴더로 해당 쿠키파일을 이동시킨다.

웹브라우저는 이동된 쿠키파일을 웹 서버에게 전달하며 웹 서버에서 쿠키파일을 활용하여 정보를 제공해 준다.

사용자가 웹 서비스의 방문을 마치면 웹브라우저는 보안영역을 관리하는 커널에게 사용자의 방문이 끝났다는 정보를 전달한다.

커널은 사용이 끝난 쿠키파일을 다시 보안영역으로 이동시켜 악성코드로 인한 쿠키파일 유출을 사전에 차단한다.

보안영역이 쿠키파일을 관리하는 방법은 <그림 2>와 같다.

4.2 보안영역 구현 방법

보안영역을 구현하는 방법은 다양하다. 구현 방법에 대해서 알아보고 적절한 구현방법을 선정해야 한다.

4.2.1 데이터베이스를 이용한 보안영역 구현

보안영역을 구성하기 위해 데이터베이스를 이용할 수 있다. 쿠키파일이 일반적인 텍스트 파일로 구성되어 있어 공격자의 공격에 취약하다. 쿠키파일을 보호하기 위해 텍스트파일형태가 아닌 데이터베이스에 쿠키파일을 내용을 저장하고, 사용자가 웹 페이지를 방문할 때, 해당 웹 페이지가 작성한 쿠키파일의 내용을 데이터베이스로부터 추출해 쿠키파일로 만들어 웹 서버에게 제공한다.

데이터베이스를 이용하여 보안영역을 구현할 경우 쿠키파일의 이동을 위한 작업 외에도 데이터베이스를 관리하기 위한 추가적인 요소들이 필요하다. 또한 사용자가 웹 페이지에 방문할 때 마다 데이터베이스에 접근해 저장된 데이터를 기반으로 쿠키파일을 만들어야 한다. 이러한 과정으로 인해서 사용자는 웹 페이지를 제공받기 위해 많은 더 많은 시간을 기다려야 할 것이다.

데이터베이스를 이용할 경우 추가적인 시간이 필요하지만 데이터베이스에 쿠키파일의 내용을 저장하기 때문에 쿠키파일을 남겨두지 않아 공격자로부터 쿠키파일을 안전하게 보관할 수 있다.

4.2.2 로컬시스템을 이용한 보안영역 구현

보안영역을 로컬시스템 내에 구현할 수 있다. 웹 브라우저가 지정한 쿠키파일 저장 위치 이외의 장소에 보안영역을 구현함으로써 공격자로부터 쿠키파일을 안전하게 보호할 수 있을 것이다. 하지만 로컬시스템에 보안영역을 구현하여 쿠키파일을 안전하게 보관할 수 있다.

로컬시스템에 보안영역을 구현할 경우 쿠키 파일이 저장된 위치가 공격자에게 노출될 수 있다. 쿠키 파일의 위치가 공격자에게 노출되는 것을 막기 위해서 로컬시스템에 저장되는 위치를 난수화하여 저장함으로서 공격자가 쿠키 파일이 저장된 장소를 알지 못하도록 해 쿠키 파일의 안전성을 향상시킨다.

4.2.3 효과적인 구현방법 제시

보안영역을 구현하기 위해 데이터베이스를 이용하는 방법과 로컬시스템을 이용하는 방법을 살펴보았다. 보안영역을 구현하기 위해서 데이터베이스를 이용할 경우 데이터베이스를 구축하기 위한 추가적인 시스템에 대한 설계가 필요하다. 또한 , 효율성 측면에서 살펴보면 데이터베이스를 이용한 보안영역은 사용자가 방문하기 원하는 웹페이지에서 작성한 쿠키 파일이 데이터베이스 내에 있는지 확인하기 위한 검색과 쿠키파일을 생성하기 위한 작업이 추가적으로 필요하게 되어 빠르게 웹 서버에 쿠키파일을 전달 할 수 없다.

그에 비해서 로컬시스템을 이용한 보안영역 구현은 데이터베이스를 이용한 보안영역 구현보다 쉽게 구현할 수 있다. 또한 데이터베이스를 이용하지 않기 때문에 데이터베이스 검색과정이 필요하지 않다. 쿠키파일을 웹 서버에게 제공하기 위한 쿠키파일 생성과정 또한 필요하지 않아 데이터베이스를 이용한 구현보다 웹 서버에게 쿠키파일을 전달하기 위해 필요한 사전준비 과정이 줄어들게 된다.

쿠키파일의 안전성 향상과 사용자 측면에서 보안영역 구현을 위해서 로컬시스템을 이용한 보안영역 구현방법을 제시한다.

4.3 보안영역의 사전 전제 사항

웹 브라우저의 종류에 따라서 서로 다른 장소에 저장되는 쿠키파일을 안전하게 보관하기 위해 보안영역은 다음과 같은 기본적인 기능이 있어야 한다.

보안영역은 쿠키파일을 안전하게 보관 및 관리하기 위한 목적으로 설계되었기 때문에 보안영역에 대한 접근 설정은 엄격히 다루어져야 한다. 또한 보안영역은 공격자가 사용자PC에 설치한 악성코드나 백도어를 이용하는 등의 접근이 불가능해야 한다. 커널의 기능을 불법적으로 취득하는 API Hooking도 불가능해야 한다.

웹브라우저의 상태를 확인하고 이를 통해 현재 사용 중이지 않은 쿠키파일이 웹브라우저에서 지정한 쿠키파일 폴더에 존재하지 않도록 관리한다.

5. 성능 시험

보안영역을 어떠한 방식으로 구현하는가에 따라 사용자가 웹페이지 URL을 입력 후 웹 페이지를 사용자에게 보여주기까지 시간이 달라질 수 있다.

사용자가 원하는 웹 페이지를 제공하기 위해 소요되는 시간은 보안영역 구현에 따라 달라질 수 있다. 보안영역 구현을 통해 쿠키파일의 안전성을 유지하면서 최단시간에 사용자가 원하는 웹 페이지를 제공할 수 있는 방법에 대해서 확인해 보았다.

성능시험을 위해서 1,000개의 쿠키파일을 수집하였다. 웹 페이지에 따라서 쿠키파일의 크기는 다양하지만 수집된 쿠키파일의 크기는 대부분 80~500바이트의 크기를 가졌다. 가장 큰 쿠키파일은 3.64KB이다.

성능 시험에 사용된 디스크는 7,200rpm을 32MB 버퍼를 가지는 디스크와 USB 메모리 디스크를 이용하였다.

성능시험의 요소로는 파일 탐색 시간, 파일 전송 시간을 선정하였다.

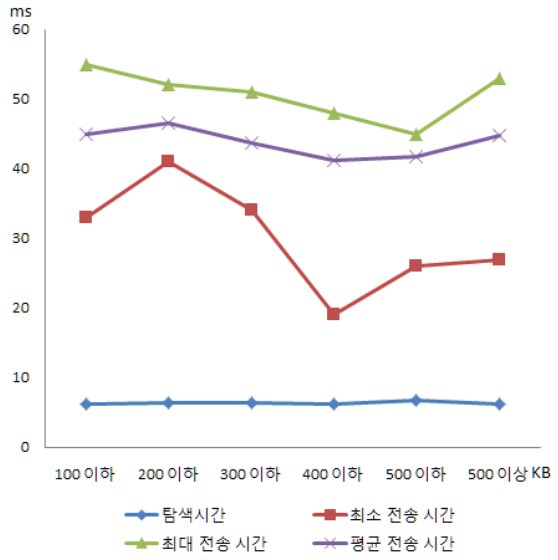
5.1 하드디스크를 이용한 보안영역 구현

웹브라우저가 설치된 디스크에 보안영역을 구현하여 성능시험을 수행하였다.

성능시험 결과 하드디스크라는 단점으로 쿠키 파일 탐색시간과 전송 시간이 다소 지연되는 것을 확인할 수 있다.

하드디스크에 저장된 파일에 접근하기 위해서는 디스크 헤더가 저장되어 있는 섹터 위에 위치해 있어야 한다. 이러한 하드디스크의 특징으로 파일 탐색, 전송하는 시간이 지연된다.

전송시간은 최소 18.5MB/S, 최대 55.8MB/S 평균 45.2 MB/S를 가지고 탐색시간은 8.3ms를 가진다.



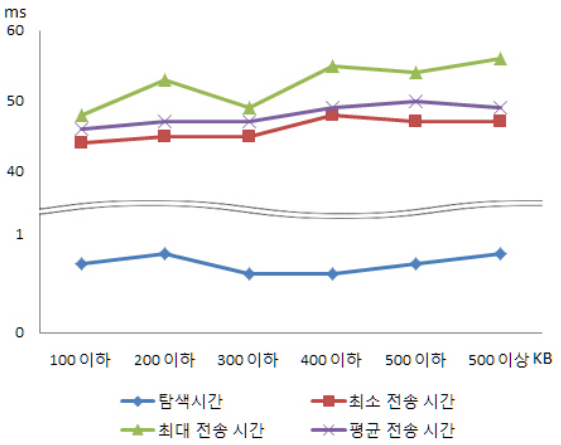
<그림 3> 하드 디스크를 이용한 보안영역 성능 평가

5.2 플래시 메모리를 이용한 보안영역 구현

USB 형태의 플래시 메모리를 이용하여 보안영역을 구현하고 성능 시험을 수행하였다.

USB를 이용하여 파일을 전송하기 때문에 파일 전송 시간은 하드디스크를 이용한 경우와 비슷한 시간을 보였다.

하지만 플래시 메모리를 이용하기 때문에 파일 탐색시간은 평균 0.85ms의 시간을 얻을 수 있었다. 이는 하드디스크를 이용한 구현하는 방법 보다 탐색시간이 9배 정도 빨라졌다



<그림 4> 플래시 메모리를 이용한 보안영역 성능 평가

5.3 성능평가를 통한 효율적인 보안영역 제시

효율적인 보안영역을 제시하기 위해서 하드디스크와 USB형태의 플래시 메모리를 이용하여 성능시험을 수행해 보았다.

하드디스크와 플래시 메모리의 전송속도를 측정한 결과 두 방법 모두 평균 45MB/S의 속도를 가졌다.

하지만 탐색시간의 경우 하드 디스크는 평균 8ms를 보였다. 하드디스크에 비해 플래시 메모리는 평균 0.85ms의 탐색시간을 보였다. 하드디스크와 플래시 메모리의 전송시간은 비슷하지만, 탐색시간은 플래시 메모리가 약 9배 정도 빠른 결과를 보였다. 또한 플래시 메모리는 사용자가 쉽게 휴대가 가능하다는 이점이 존재한다. 사용자가 다른 컴퓨터를 이용하여 웹 서비스를 제공 받을 때에도 플래시 메모리에 저장된 쿠키파일을 이용하면 이전에 제공받던 웹 서비스를 계속해서 제공받을 수 있다.

6. 결 론

사용자가 웹 서비스로부터 제공받은 서비스의 정보 및 로그인 정보 등이 쿠키 파일에 저장된다. 하지만 쿠키파일은 평문형태의 텍스트 파일로 저장되어 공격자로 부터 취약하다.[9]

이러한 쿠키의 취약성을 보완하기 위한 연구는 여러 가지 방면으로 진행되어 왔다. 하지만 쿠키

파일을 저장하고 있는 폴더에 대한 보안성을 향상시킬 수 있는 방법에 대해서는 상대적으로 연구가 많이 부족하다.

본 논문에서는 악성코드 등으로부터 쿠키 파일이 저장되어 있는 폴더가 공격자에게 노출되는 것을 보호하기 위한 보안영역을 제시 하였다.

또한 제시한 보안영역의 효율성을 측정하기 위해 하드디스크와 플래시 메모리에 보안영역을 구현하여 성능비교를 해 보았다.

고도화된 네트워크 인프라에 따라 사용자들이 웹 서비스를 더욱 빨리 사용하기 원하는 만큼 보안영역을 적용하여 쿠키파일의 보안성을 높이는 동시에 보안영역을 적용하기 이전과 비슷한 속도가 유지되어야 할 것이다.

이러한 사용자의 요구를 만족하면서 보안영역을 적용하기 위해 플래시 메모리를 이용하여 구현하는 것이 효율적이라는 결과를 얻었다.

플래시 메모리에 보안영역을 적용할 경우, 쿠키파일 보관하고 있는 폴더의 보안성 향상을 기대할 수 있으며, 0.5~0.7초의 추가 시간이 소요된다.

쿠키파일을 보관하고 있는 폴더의 보안성을 향상시키기 위해서 1초 이하의 시간이 추가적으로 더 소요되는 것은 사용자가 납득할 만한 시간이라고 판단된다.

본 연구를 통해서 쿠키파일을 보관하고 있는 폴더에 대한 보안성을 향상 시킬 수 있는 방법에 대해서 지속적으로 연구되어야 할 것이다.

참 고 문 헌

[1] 최향창, 최은복, 노봉남(2002), 쿠키 보호 시스템 설계, *정보보호학회 봄 학술발표논문집 Vol. 29. No. 1*

[2] 이창우, 김창희, 이준호(2008), 웹 2.0 환경에서의 보안 문제와 효율적 웹 검사 방안, *정보보호학회지 제18권 제3호*, 1-131

[3] 심기명, 최신 웹 해킹 대응 및 개인정보보호 보안기술, [IITA] 정보통신연구진흥원 학술정보 - 주간기술동향 1312호

[4] 홍봉화, 정윤돈, 김은원(2005), DHTML 편집기를 이용한 블로그 사이트에서 쿠키보안에 관한 연구, *전자공학회 논문지 제 42 권* TE 제 2 호

[5] http://seobangnim.com/zbxe/4030/

[6] 안철수연구소(2005) 쿠키의 취약성 및 요구되는 보안성, 안철수연구소 전문가 칼럽

[7] 임대호(2000), 악성 코드에 의한 HTTP Cookie 유출 문제점 및 대책 CERTCC-KR 권고문

[8] 양종필, 이경현(2002), 인증서 기반의 개선된 보안 쿠키의 설계와 구현, 한국통신학회논문지 '02-11 Vol.27 No.11C

[9] 서진원, 서희석, 곽진(2010) 웹서비스 공격정보 분류 방법 연구, *한국시뮬레이션학회 논문지 제19권 제3호*

서 희 석

2000 성균관대학교 산업공학과 (공학사)
2002 성균관대학교대학원 전기전자 및 컴퓨터공학과 (공학석사)
2005 성균관대학교대학원 전기전자 및 컴퓨터공학과 (공학박사)
2005~현재 한국기술교육대학교 컴퓨터공학부 교수
관심분야: 모델링&시뮬레이션, 네트워크보안, 보안 시뮬레이션, USN
E-Mail: histone@kut.ac.kr

최 요 한

2008~현재 한국기술교육대학교 컴퓨터공학부 학사과정
관심분야: 악성코드, 네트워크, 모바일 보안
E-Mail: ahluiyoo@kut.ac.kr