사용자 식별을 통한 웹 페이지 필터링 시스템 (WFS)의 설계

박수빈, 조동섭
이화여자대학교 컴퓨터공학과
e-mail:subin.ewha@gmail.com
dscho@ewha.ac,kr

A Study on Webpage Filtering System Using Client Identification

Su-Bin Park, Dong-Sub Cho
Dept of Computer Science and Engineering, Ewha womans University

요 약

웹의 발전에 따라 정보 전달의 매체가 웹으로 이동됨에 따라 대부분의 기업이나 기관에서는 그 유용성 및 용이성 때문에 홈페이지를 통해 정보와 서비스를 제공하고 있으며 고객, 직원, 협력사 등과의 거래를 행하는 데 있어 웹 애플리케이션을 사용하는 경우가 많아졌다. 이러한 현상과 함께 웹에 대한 공격이 급증하면서 웹 공격을 필터링 하기 위한 여러 가지 방법이 제시되었다. 본 논문에서는 적은 오버헤드로 웹 서버와 브라우저의 영역에서 보다 신뢰성 있는 웹 페이지 전송을 위한 방법을 기술한다.

1. 서론

월드 와이드 웹(World Wide Web)이 사람들 사이의 정보 전달(Communication)에 있어 혁명을 일으키게 된 후, 매일 만들어지는 새로운 웹 페이지의 수는 수천만 개, 온라인 문서들은 수십억 개에 이르고 있다. 대부분의 기업이나 기관에서는 그 유용성 및 용이성 때문에 홈페이지를 통해 정보와 서비스를 제공하고 있으며 고객, 직원, 협력사 등과의 거래를 행하는 데 있어 웹 애플리케이션을 사용하는 경우가 많아졌다.

이에 웹에 대한 공격이 급증하면서 웹 어플리케이션 보안의 중요성이 증가하고 있다. 웹 어플리케이션의 해킹은 전통적인 해킹기법에 비해 상대적으로 취약한 공개 소스 기반의 웹 어플리케이션 해킹으로도 진행되고 있다. 웹 페이지 변조에서부터 명의도용, 개인 신상 정보 유출 등 다양한 방식의 위협에 노출되어 있다[1][2][3].

<표 1> 2007년 Top10 웹 애플리케이션 취약점

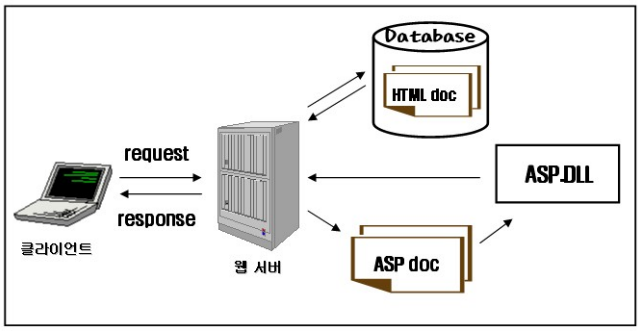
A1	크로스 사이트 스크립팅(XSS)
A2	인젝션 취약점
A3	악성 파일 실행
A4	불안전한 직접 객체 참조
A5	크로스 사이트 요청 변조(CSRF)
A6	정보 유출 및 부적절한 오류 처리
A7	취약한 인증 및 세션 관리
A8	불안전한 암호화 저장
A9	불안전한 통신
A10	URL 접속 제한 실패

웹 공격은 웹 서비스를 제공하는데 필요한 웹 어플리케이션을 공격하여 정상적인 웹 서비스를 방해하거나 권한 없는 정보를 습득하는 일련의 행위를 말한다. Gartner Group은 오늘날 사이버 공격의 75%가 애플리케이션 레벨에서 행해진다고 추정한다[4]. 웹 어플리케이션에 대한 보안 취약점과 관련하여 OWASP(Open Web Application Security Project)에서는 <표 1>에 제시된 웹 어플리케이션 취약점을 10가지로 분류하여 발표하였다[5]. 이러한 웹 공격을 방어하기 위해 패턴 매칭을 이용한 필터링을 수행하거나 코드를 수정하는 방법이 있을 수 있지만 새로운 공격에 대해서는 탐지 및 방어가 어렵다[6]. 또한 웹 방화벽과 같은 단위보안 제품을 도입할 수 있지만 많은 비용과 노력이 요구된다. 웹 공격은 완전한 보안이 어렵다. 본 논문에서는 공급자가 인가한 사용자만이 요청한 웹 페이지에 대한 정보를 받아볼 수 있는 서비스를 제공하는 웹 서버-클라이언트의 안전성을 위한 시스템을 제안한다.

본 논문의 전체적인 구성은 다음과 같다. 2장에서는 관련 연구 내용으로 웹 서버의 구성과 웹 공격의 유형을 소개하고 3장에서는 웹 서버를 통한 웹 어플리케이션 보안 시스템 설계의 기본 아이디어를 설명하고, 마지막으로 4장에서 결론을 맺는다.

2. 관련 연구

본 장에서는 웹 서버의 구성과 동작 원리에 대해 설명하고 기존의 웹 어플리케이션의 보안 방법과 본 논문에서 중점적으로 처리할 웹 공격에 대하여 논한다.



(그림 1) 웹 서버의 동작원리

2.1 웹 서버의 구성

웹 서버는 웹 브라우저와 실제 사용자가 웹브라우저를 통해 요청한 정보를 연결시켜 주는 역할을 하는 웹 어플리케이션에서 가장 중점적인 역할의 구성 요소이다. 웹 서버는 HTTP/HTTPS 요구를 관리하고, 사용자의 세션을 관리하며, 웹 서비스의 모든 과정을 처리할 수 있도록 담당하는 역할을 하는 서비스 프로그램으로 기본적인 레벨에서 웹 브라우저에 정적인 콘텐츠를 제공한다.

웹 서버는 웹을 구동시키는 프로세스와 웹을 구성하는 파일과 폴더로 이루어져 있는데 사용자가 요청한 파일이 웹 서버의 파일 시스템 내에 존재한다면 웹 서버는 이를 읽어 웹 브라우저에 전송해주고 정적인 페이지가 아닌 ASP 문서를 요청할 시에는 ASP 프로그램을 통해 읽어온 페이지를 전송해준다(그림 1).

2.2 웹 애플리케이션 보안

현재 웹 응용 프로그램의 보안을 위해서 Scott 와 Sharp는 응용 프로그램에 유효하지 않고 외부로부터의 악의적인 입력을 필터링 하는 게이트웨이(gateway)의 사용을 제안하였다[3][4]. 이러한 방식은 AppSheild[5]와 InterDo[6]과 같이 Application Layer에서 응용 프로그램 데이터에 대한 검사를 수행하는 웹 응용 프로그램 게이트웨이 방식의 상용제품에 적용되었다. 그러나 이러한 방식은 사용자의 환경설정이 요구되며, 이에 따라 보안상의 안정성이 크게 좌우된다. 또한 게이트웨이는 웹 응용 프로그램의 전단부에서 해당 웹 응용 프로그램에 대한 모든 HTTP 요청을 검사하게 되므로 웹사이트의 성능이 저하되는 단점을 가지고 있다.

웹 애플리케이션 보안제품은 웹 해킹 및 웜으로부터 핵심적인 웹 애플리케이션을 보호하는 전용 솔루션을 의미한다. 쿠키 변조, 세션 하이재킹, 폼필드 변조, 파라미터 변조와 같은 해킹에 대한 대응력이 뛰어나고 안전한 웹 서비스를 보장하기 위해 나온 웹 방화벽은 기존의 전통적인 방화벽처럼 Positive Security Policy 방식을 채택해 알려지지 않는 웹 공격에도 방어할 수 있도록 설계되어 있다[7]. 웹 방화벽은 네트워크를 지나는 HTTP/HTTPS 트래픽을 분석하여 웹 서버를 보호할 수 있는 네트워크 기반 방화벽과 웹 서버가 제공하는 API를 기반으로 구현, IIS나 Apache 웹 서버의 플러그인 형식으로 탑재되는 웹 서버기반의 방화벽으로 구분된다. 웹 애플리케이션 보안을 실현하기 위해 '안전하게 웹 애플리케이션을 개발하기 위한 지침서'에 따라 프로그래밍을 하거나 이미 개발돼 있는 프로그램의 소스 코드를 분석하는 등 근본적인 해결책이 있지만 실제로는 적용하기가 쉽지 않고 보안 담당 관리자가 채택할 수 있는 방법도 한계가 있다.

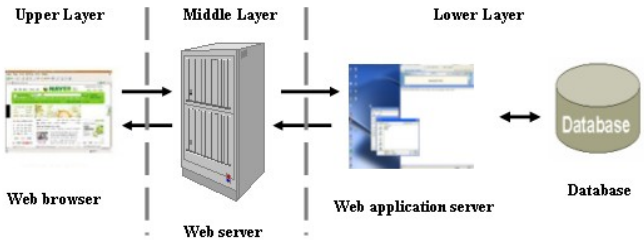
3. 설계

본 장에서는 웹 페이지 필터링 시스템인 WFS (Webpage Filtering System)의 동작원리를 설명하고 기능 및 장점을 알아본다.

공급자는 자신이 인가한 사용자만이 요청한 웹 페이지에 대한 정보를 받아볼 수 있는 안전성을 제공받고자한다. 이에 본 논문에서는 일반적인 웹 환경에서 사용자의 부가정보를 웹 문서에 삽입시키는 기능을 서버에 추가하여 공급자가 인가한 사용자가 아닌 인가되지 않은 제 3자가 페이지를 획득했을 시 브라우저에서 렌더링을 해주지 않는 WFS를 제안한다. 본 장에서는 WFS의 구성과 프로세스의 흐름에 대해 설명하고 WFS에서 생성된 웹 페이지의 처리과정을 설명한다.

3.1 WFS 서버 구조

웹은 (그림 2)와 같이 3단계로 구성되어 있다. WFS는 middle later 부분에서 사용자의 브라우저로 전송되는 웹 페이지에 부가 정보를 삽입하여 보낸다.

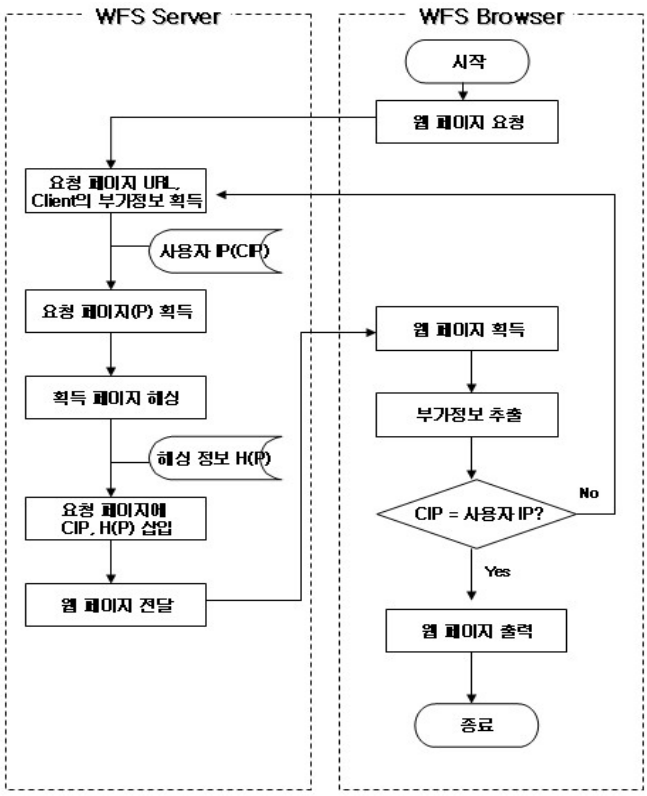


(그림 2) 웹의 구성

WFS 서버의 동작(그림3)은 크게 2가지로 이루어진다. 사용자가 요청한 웹 페이지를 생성하여 해싱하는 부분, 요청 시 획득한 앞에서 해싱한 웹 페이지 정보와 사용자의 IP주소, 시간정보를 웹 페이지에 삽입해주는 부분으로 나누어진다.

3.1.1 웹페이지 해싱

사용자가 요청한 웹 페이지를 생성하여 사용자에게 전송하기 전 단계에 웹 서버에서 웹 페이지의 내용 X를 기반으로 해싱하여 메시지요약 H(X)를 얻는다.



(그림 3) WFS 동작 프로세스

3.1.2 부가 정보 삽입

사용자가 페이지를 요청 시 사용자의 IP와 요청시간 정보 등 부가정보들을 획득하여 사용자에게 전송하기 전 단계에 요청한 웹 페이지에 삽입하여 전송한다. 삽입 방법은 다음과 같다.

3.2 WFS 브라우저

WFS 브라우저(그림 3)는 WFS 서버로부터 전송받은 웹 페이지에서 부가 정보를 추출한다. 이 추출된 부가정보 IP를 사용자의 IP정보와 비교한다. 전송받은 웹 페이지에 삽입된 IP정보와 사용자의 IP정보가 일치할 경우 브라우저는 웹 페이지의 내용을 렌더링한다. 그러나 이 두 정보가 일치하지 않을 경우 웹 페이지는 중간에 제 3자에 의해 갈취당한 것으로 판단하고 웹 페이지의 내용을 화면에 출력하지 않고 다시 서버에 사용자가 요청한 페이지의 URL을 요청한다.

3.3 기능 및 장점

본 논문에서 제안한 WFS는 전용 웹 서버(WFS Server)와 전용 웹 브라우저(WFS Browser)를 통해 공급자가 자신이 인가한 사용자만이 요청한 웹 페이지에 대한 정보를 받아볼 수 있는 안전성을 제공받고자할 때 실시간으로 최초 요청자와 공격자의 정보비교를 하여 웹 페이지의 렌더링 여부를 판단한다. 그렇기 때문에 공급자가 개인화 된 웹 페이지를 제공하고자 할 때 인가받지 않은 제 3자의 접근을 막을 수 있다. 사용자의 개인 정보가 아닌 IP 정보로 판단하기 때문에 모든 처리과정은 일회성이다. 또한 획득 페이지를 해싱하여 얻은 해싱정보를 삽입함으로써 WFS Browser는 전달받은 웹 페이지를 다시 해싱하여 삽입된 해싱정보와 비교해보아 웹 페이지의 무결성을 확인할 수 있다.

4. 결론 및 향후 과제

웹에 대한 공격이 급증하면서 웹 공격을 필터링 하기 위한 여러 가지 방법이 제시되었다. 그러나 웹 공격은 완전한 보안이 어렵다. 기존의 웹 보안 솔루션은 기업의 정보를 보호하고 있으나 웹 페이지의 보안은 완벽하지 않았다 본 논문에서는 적은 오버헤드로 웹 서버와 브라우저의 영역에서 보다 신뢰성 있는 웹 페이지 전송을 위한 방법인 WFS를 설계하였다. 이는 공급자가 인가한 사용자만이 요청한 웹 페이지에 대한 정보를 받아볼 수 있는 서비스를 제공하는 웹 서버-클라이언트의 안전성을 위한 시스템이다. 이는 사용자간 차별화된 서비스를 제공해야 하는 웹 서버에 적용될 수 있을 것이다.

WFS는 웹서버-클라이언트 연결에서 IP 정보를 사용자의 식별을 통한 필터링이기 때문에 일회성이다. 그래서 서버-클라이언트간 연결이 잦을 때에는 매번 비교 처리를 해야 하기 때문에 속도가 떨어진다. 앞으로 처리시간 단축을 위한 사용자 정보의 삽입에 대한 연구가 계속되어야 할 것이다.

참고문헌

[1] Christopher Kruegel, Giobanni Vigna, William Robertson, "A multi-model approach to the detection of web-based attacks", Computer Networks:Vol48, No.5, pp.717-738, Aug, 2005.

[2] Mark Curphey, Joel Scambray, Erik Olson, "Improving Web Application Security : Threats and Countermeasures", Microsoft Corporation, 2003.

[3] Robert Auger, Ryan Barnett, "Web Application Security Consortium : Threat Classification Version 1.0", Web Application Security Consortium(www.webappsec.org), 2004.

[4] Norhazimah Abdul Malek, "SECURING APPLICATONS FROM HACKERS", Computimes, 28 November 2005.

[5] OWASP(Open Web Application Security Project) http://www.owasp.org

[6] Scott, D., Sharp, R. "Abstracting Application-Level Web Security.", Proc. 11th Int'l Conf. World Wide Web (WWW2002), pages 396-407, May 17-22, 2002

[7] 윤여웅, "게이트웨이형 웹 애플리케이션 방화벽 보호 프로파일에 관한 연구" 정보과학회지, 2007