韓國 컴퓨터情報學會 論文誌
第13卷 第3號, 2008. 5.

2008-13-3-3-3

트래픽 분석에 의한 웹 어플리케이션 공격 방지

장 문 수*, 오 창 석**

Web Application Attack Prevention by Traffic Analysis

Moon-Soo Chang*, Chang-Suk Oh**

요 약

웹 서비스를 이용한 개인 정보 유출은 보안시스템 구축에도 불구하고 급격하게 줄어들지 않고 있다. 방화벽 시스템 구축 후에도 HTTP 80 port나 HTTPS의 443 port는 외부로부터의 접근을 허용하여 지속적으로 서비스를 하고 있으므로 웹 어플리케이션으로 유입되는 트래픽은 취약하다. 따라서 본 논문에서는 웹 서비스 환경으로 유입되는 다양한 형태의 공격 패턴 및 취약한 트래픽을 분석하여 정상 트래픽과 비정상 트래픽을 분류하였다. 분류된 비정상 트래픽을 대상으로 OWASP(Open Web Application Security Project)에서 권고한 웹 어플리케이션 보안 취약점을 바탕으로 공격 패턴을 분석하고, 실시간 공격탐지 및 차단이 가능한 시스템을 설계하여, 웹 어플리케이션의 보안 취약성을 이용한 다양한 공격 패턴을 즉각적이며, 효과적으로 방지하여 유해한 공격 트래픽으로부터 공격 방지 효율을 높일 수 있는 방법을 제안하였다.

Abstract

Despite of information security installation, leakage of personal information in web services has not decreased. This is because traffics to web applications are still vulnerable by permitting external sources to access services in port HTTP 80 and HTTPS 443, even with firewall systems in place. This thesis analyzes various attack patterns resulted from web service environment and vulnerable traffic and categorizes the traffics into normal and abnormal traffics. Also this proposes ways to analyze web application attack patterns from those abnormal traffics based on weak points warned in OWASP(Open Web Application Security Project), design a system capable of detect and isolate attacks in real time, and increase efficiency of preventing attacks.

▸ Keyword : OWASP(Open Web Application Security Project), 공격방지(Attack prevention), 트래픽분석(Traffic Analysis), 웹 어플리케이션(Web application)

• 제1저자 : 장문수 *교신저자 : 오창석
• 접수일 : 2008. 3. 26, 심사일 : 2008. 3. 28, 심사완료일 : 2008. 5. 24.
* 충북대학교 컴퓨터공학과, ** 충북대학교 전기전자컴퓨터공학부
※ 본 논문은 2007년도 충북대학교 학술연구지원사업의 연구비 지원에 의하여 연구되었습니다.

Ⅰ. 서 론

전현대 생활의 인프라는 인터넷이라는 도구가 명실상부한 실생활의 수단으로 자리매김하고 있는 실정이며, 정보화 사회에서 유비쿼터스 사회로 진화하는 과도기에 있다. 오프라인으로 구매하던 물건들이나, 은행을 직접 방문해서 처리했던 일들은 인터넷을 통해서 쇼핑몰, 금융기관 등에서 제공되는 웹 어플리케이션을 이용하여 집 또는 사무실에서 간단하게 온라인으로 해결하는 패턴으로 바뀌고 있으며, 전자상거래(EC : Electronic Commerce)를 이용하여 각 기업, 기관들은 웹 어플리케이션을 이용하여 고객들을 맞이하는 창구를 개설하여 다양한 서비스와 편의를 제공하고 있다.

이러한 창구를 통해서 이동되는 사용자 정보에는 개인정보, 조직정보, 금융정보 등 다양하고 중요한 정보들을 포함하고 있으며, 악용될 경우 사용자에게 직접적이고 금전적인 피해를 끼칠 수 있다. 따라서 최근 해커들의 동향은 클라이언트/서버 기반의 해킹 수법이나, 특정 시스템을 공격하여 피해를 입히는 등의 방법에서 악성코드를 결합해 피싱(Phishing)을 진행하는 신종 공격인 파밍(Pharming) 형태로 진화하여 사용자 정보를 후킹(Hooking)할 수 있는 통로인 웹 어플리케이션을 공략하는 형태로 공격목표를 옮겨가고 있는 추세이며, 보안이 고려되지 않은 웹 어플리케이션의 경우 이러한 공격목표로부터 자유로울 수 없는 것이 현실이다.

따라서 본 논문에서는 트래픽 탐지 방법 및 웹 어플리케이션의 취약점에 대해서 알아보고, 최근 보안 이슈사항으로 대두되고 있는 웹 어플리케이션 보안을 유지하기 위하여 OWASP Top 10을 기반으로 웹 어플리케이션의 기술적 취약점 점검 및 이러한 취약점을 악용한 공격으로부터 실시간으로 트래픽을 분석하여, 정상적인 트래픽요청은 웹 어플리케이션에 전달하여 정상적인 서비스를 제공하고, 유해트래픽을 이용한 공격트래픽은 공격탐지 및 차단할 수 있는 시스템을 제안하여, 기존의 시스템 환경보다 제안시스템의 안정성 및 효율성을 실험 결과를 토대로 고찰하였다.

Ⅱ. 트래픽 탐지 방법 및 웹 어플리케이션 취약점

2.1 IDS

침입탐지시스템(Intrusion Detection System)은 네트워크의 트래픽을 분석하여, 컴퓨터 시스템의 비정상적인 사용, 오용, 남용 등을 규정하는 시스템으로 가능하면 실시간으로 처리하는 시스템이다[1]. 침입 탐지 모델에 따른 침입 탐지 방법은 〈표 1〉과 같다.

표 1. 탐지 모델에 따른 침입 탐지 시스템의 분류
Table 1. Classification of the intrusion detection system according to the detection model

구분	침입탐지방법
오용 탐지	조건부 확률
	전문가 시스템
	상태 전이 분석
	키 입력 감시
	모델 기반 침입탐지
비정상 행위 탐지	통계적 접근
	특징 추출
	비정상 행위 측정 방법의 절차
	예측 가능한 패턴 생성
	신경망

오용 탐지 모델은 Signature-based 탐지 모델이라고도 하며, 이미 알려져 있는 공격에 대한 취약점 정보와 감사 데이터를 비교하여 침입을 탐지하는 시스템으로, 알려진 취약점에 대하여 false positive 오류를 줄일 수 있고 비교적 구현 방법이 수월하다는 장점이 있지만 취약점 정보에 대한 의존도가 매우 높아 새로운 유해 트래픽에 대해서는 탐지하지 못하는 단점이 있다[2].

비정상 행위 탐지 모델은 Profile-based 탐지 모델이라고도 하며, 사용자의 비정상 행위나 컴퓨터 자원의 사용을 탐지하는 것으로 정상적인 행동 프로파일 모델을 벗어나는 경우를 침입으로 간주한다. 오용 탐지 모델에 비해 false negative 오류를 줄일 수 있는 장점이 있지만 정상적인 트래픽에 유해 트래픽이 포함되므로 상당한 주기의 프로파일 갱신이 필요한 단점이 있다[2][3]. 〈그림 1〉은 IDS의 침입탐지 프레임워크를 나타낸다[1]

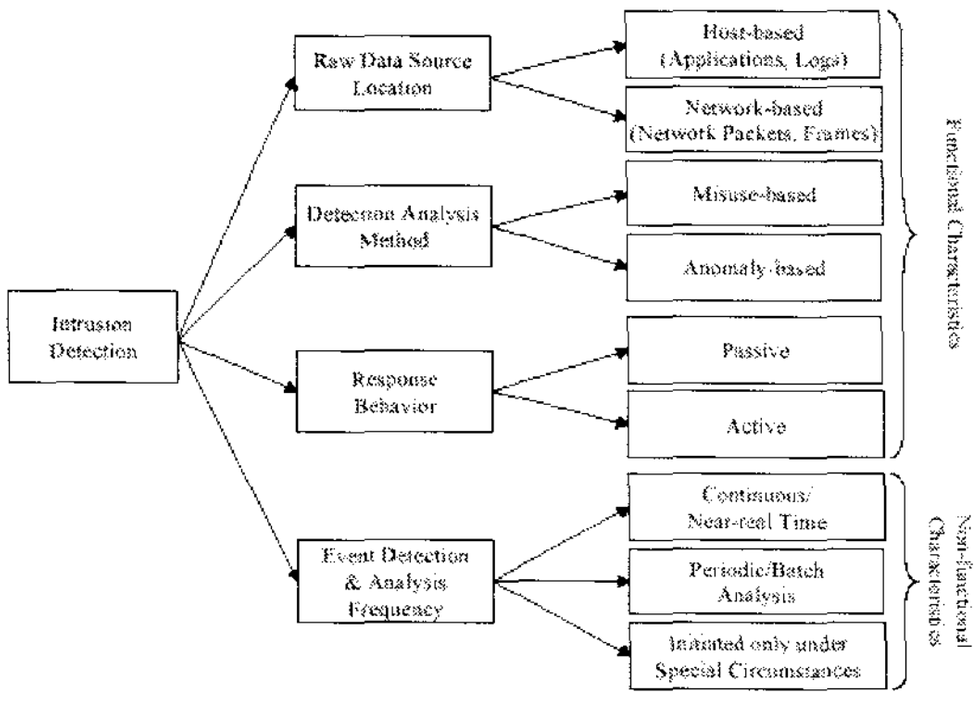


그림 1. IDS의 구조
Fig 1. IDS Framework (ISO/IEC TR 15947)

2.2 IPS

침입방지시스템(Intrusion Prevention System)은 특정 패턴을 기반으로 공격자의 유해 트래픽을 탐지하는 IDS와는 다르게 기존 네트워크의 구성을 변형시키지 않는 인라인 모드로 네트워크에 구현되어 유해 트래픽을 차단하는 것은 물론 이를 통해 네트워크의 안정성과 유해 트래픽을 경감하여 보다 빠르게 트래픽 속도를 높여 준다.

최근 몇 년간 웜을 포함한 해킹공격의 유형은 크게 다형성(Polymorphic), 신속한 전파(Propagation), 지능화(Machine Learning) 등으로 구분돼 급속하게 발전했다. 이러한 특성은 제로데이 공격(Day-Zero Attack)이라는 평범하지 않은 정보보안 체계의 새로운 이슈를 야기 시켰으며, 이를 대변하듯 보호대상 네트워크에 유해트래픽이 미치지 못하도록 네트워크 기반에서의 탐지 및 차단을 요구하는 시간적인 응답속도와 탐지 범위 또한 다각화되고 넓어지기 시작했다[4].

대부분 네트워크 중계 구간으로 이뤄진 통신사나 망사업자의 네트워크 인프라를 위협하는 공격은 개별 단위의 고밀도 해킹보다는 우선 L2 기반에서의 최대 문제점인 ARP 스푸핑, ARP 포이즈닝, ARP 플러딩과 같이 서비스 지연을 목적으로 하는 DoS, DDoS, DrDoS 계열들이며, L3~L4 기반에서는 ICMP 플러딩, Syn 플러딩, UDP 플러딩, 스캔 & 스윕 등의 웜 감염, 감염시도에서 오는 유해트래픽의 탐지그룹 및 웜 감염 이후 발생되는 공격 시도, 또는 이에 따른 대용량 유해트래픽이 된다[6].

정상과 비정상의 구분 없는 서비스 트래픽을 제어하기 위해 기존의 방화벽에서 사용되던 기술인 서비스 필터링(Service Filtering)도 사용된다[5].

2.3 웹 어플리케이션 취약점

대부분의 웹 어플리케이션의 구조는 3tier에 입각한 구조를 갖고 있다. 일반 사용자를 대상으로 사용자 관점으로 편하게 접근할 수 있도록 표현되는 Presentation Tiers, 다양한 비즈니스 로직 처리가 구성되어 있는 Application Tiers, 관리되어지는 사용자 정보 및 비즈니스 정보들을 담고 있는 Data Tier로 구성되어 있다[7]. 이와 같이 웹 어플리케이션을 대상으로 OWASP Top 10에서 제공한 웹 어플리케이션 취약점은 다음과 같다[8].

1) A1 - 크로스사이트 스크립팅

크로스사이트 스크립팅(XSS) 취약점은 콘텐츠의 암호화나 검증하는 절차 없이 사용자가 제공하는 데이터를 어플리케이션에서 받아들이거나, 웹 브라우저로 보낼 때마다 발생한다. 크로스사이트 스크립팅은 공격자가 희생자의 브라우저 내에서 스크립트를 실행하게 허용함으로써 사용자 세션을 가로채거나, 웹 사이트를 손상하거나 웜을 심는 것 등을 가능하게 할 수 있다.

2) A2 - 인젝션 취약점

인젝션 취약점 중에서 특히 SQL 인젝션 취약점은 웹 어플리케이션에서 매우 흔하다. 인젝션은 사용자가 입력한 데이터가 명령어나 질의문의 일부분으로 인터프리터에 보내질 때 발생한다. 악의적인 공격자가 삽입한 데이터에 대해 인터프리터는 의도하지 않은 명령어를 실행하거나 데이터를 변경할 수 있다.

3) A3 - 원격 파일 실행

원격 파일 인젝션(RFI)에 취약한 코드는 공격자가 악의적인 코드와 데이터의 삽입을 허용함으로써 전체 서버 훼손과 같은 파괴적인 공격을 가할 수 있다. 악성 파일 실행 공격은 PHP, XML, 그리고 사용자로부터 파일명이나 파일을 받아들이는 프레임워크에 영향을 준다.

4) A4 - 불안전한 직접 객체 참조

직접 객체 참조는 개발자가 파일, 디렉토리, 데이터베이스 기록 혹은 키 같은 내부 구현 객체에 대한 참조를 URL 혹은 폼 매개변수로 노출시킬 때 발생한다. 공격자는 이러한 참조를 조작해서 승이ㄴ 없이 다른 객체에 접속할 수 있다.

5) A5 - 크로스 사이트 요청 변조

크로스 사이트 요청 변조(CSRF) 공격은 로그온 한 희생자의 브라우저가 사전 승인된 요청을 취약한 웹 어플리케이션에 보내도록 함으로써 희생자의 브라우저가 공격자에게 이득이 되는 악의적인 행동을 수행하도록 한다. CSRF는 자신이 공격하는 웹 어플리케이션이 강력하면 할수록

강력해진다.

6) A6 - 정보 유출 및 부적절한 오류 처리

웹 어플리케이션은 다양한 어플리케이션 문제점을 통해 의도하지 않게 자신의 구성 정보, 내부 작업에 대한 정보를 누출하거나 또는 개인정보 보호를 위반할 수 있다. 공격자는 이러한 약점을 사용해서 민감한 정보를 훔치거나 보다 심각한 공격을 감행한다.

7) A7 - 취약한 인증 및 세션 관리

자격 증명과 세션 토큰은 종종 적절히 보호되지 못한다. 공격자는 다른 사용자인 것처럼 보이게 하기 위하여 비밀번호, 키, 혹은 인증 토큰을 손상시킨다.

8) A8 - 불안전한 암호화 저장

웹 어플리케이션은 데이터와 자격 증명을 적절히 보호하기 위한 암호화 기능을 거의 사용하지 않는다. 공격자는 약하게 보호된 데이터를 이용하여 신원 도용이나 신용카드 사기와 같은 범죄를 저지른다.

9) A9 - 불안전한 통신

웹 어플리케이션은 민감한 통신을 보호할 필요가 있을 때 네트워크 트래픽을 암호화하는데 종종 실패한다.

10) A10 - URL 접속 제한 실패

웹 어플리케이션이 권한 없는 사용자에게 연결 주소나 URL이 표시도지 않도록 함으로써, 민감한 기능들을 보호한다. 공격자는 이러한 약점을 이용하여 이 URL에 직접 접속함으로써 승인되지 않은 동작을 수행한다.

〈그림 2〉는 OWASP에서 제안한 10가지 취약점 의 근원 데이터를 수집하는 MITRE의 최근 동향을 나타낸다[8]. 크로스 사이트 스크립트, 인젝션 결함, 악의적인 파일 실행, 불안전한 직접 객체 참조는 다른 항목과는 상대적으로 많은 비중을 차지하고 있다는 것을 확인 할 수 있다.

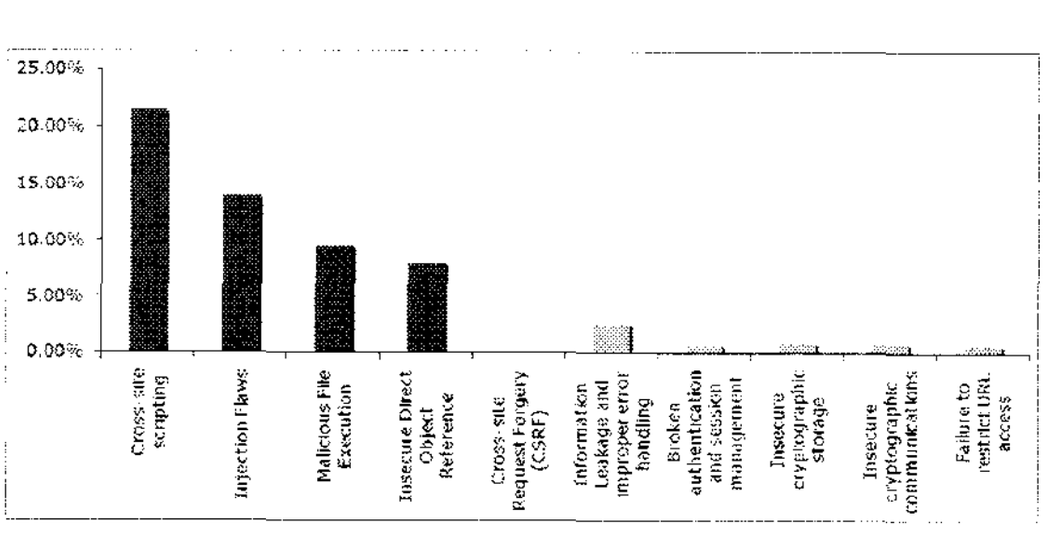


그림 2. MITRE 취약점 경향 (2006)
Fig 2. MITRE Vulnerability Trends for 2006

XSS와 같은 취약점을 악용할 수 있는 피싱(Phishing) 공격과 약하거나 존재하지 않는 인증 또는 승인 확인의 경우에는 A1, A4, A7, A10에 해당하는 내용의 취약점을 가지고 있다. 불충분한 확인, 비즈니스 규칙, 그리고 취약한 권한 확인으로 인한 개인정보 보호 위반의 경우에는 A2, A4, A6, A7, A10에 해당하는 내용의 취약점을 가지고 있으며, 존재하지 않는 암호화 통제는 A8, A9, 원격 파일 삽입은 A3, 인증, 비즈니스 규칙, 권한 확인을 통한 신원 도용은 A4, A7, A10의 취약점을 가지고 있다. 또한 인젝션과 원격파일 삽입을 통한 시스템 훼손, 데이터 변조, 혹은 데이터 파괴 공격은 A2, A3의 취약점을 가지며, 권한 없는 처리와 CSRF 공격을 통한 금융 손실은 A4, A5, A7, A10의 취약점을 가지고 있다[8].

OWASP Top10 2007에서는 기존의 가장 심각한 어플리케이션 취약점에 대해 새롭게 재작성 되었으며, 취약점에 대한 방어 방법에 대한 논의와 보다 많은 취약점을 해결할 수 있는 정보들을 제공하고 있다. 위의 10가지 리스트만 보더라도 보안은 일회성의 이벤트 형태로 처리해서는 안 된다. 보안상 안전한 개발은 웹 어플리케이션을 기반으로 모든 개발의 라이프 사이클 단계에서 다루어져야 한다. 보다 안전한 프로그램은 설계, 개발, 그리고 전체 과정에 걸쳐 기본적으로 보안을 고려해야 한다[9].

III. 트래픽 분석 및 차단 기법 제안

본 논문에서 제안한 트래픽 분석 및 차단 기법은 웹 어플리케이션 서버 앞단에 위치하여 웹 어플리케이션으로 들어오는 모든 트래픽을 대상으로 한다. 대부분의 웹 어플리케이션을 운용하는 시스템과 네트워크 인프라 내에는 기본적으로 IDS와 같은 방화벽 시스템이나, IPS 시스템이 위치하게 된다. 기존 방화벽 시스템들은 포트 단위로 개방해야 하기 때문에 HTTP, HTTPS를 위해 80 포트, 443 포트를 열어놓아야 한다. 이 포트로 유입되는 트래픽을 대상으로 공격트래픽을 판단하여 제거하며, 웹 어플리케이션을 보다 안전하게 방어하기 위해, 취약점으로부터 보호한다. 〈그림3〉은 본 논문에서 제안한 트래픽 분석 시스템의 구조도이다.

ISP나 학내망의 경우 기존 Legacy 보안 시스템이 구축되어 운용되고 있는 실정이며, 이 Legacy 보안 시스템에서는 전체적인 네트워크 보안 관점에서 접근하기 때문에 특정 웹 어플리케이션 시스템으로 유입되는 공격 트래픽을 완벽하게 탐지 및 차단하지 못하는 것을 확인 할 수 있었다. 기존의 Legacy 보안 시스템을 통과한 트래픽을 대상으로 본 논문에서 제안한 트래픽 분석 시스템으로 유입되는 트래픽을 대상으

로 정책설정을 통해 구축된 트래픽 분석 데이터를 기반으로 하여 SQL Injection공격, XSS 스크립트 공격, 불량 태그 등을 검출하여 공격으로 판단된 트래픽의 경우 Drop을 한다[10]. 정상으로 판단된 트래픽의 경우에는 웹 어플리케이션 시스템으로 다시 트래픽을 전달하게 된다.

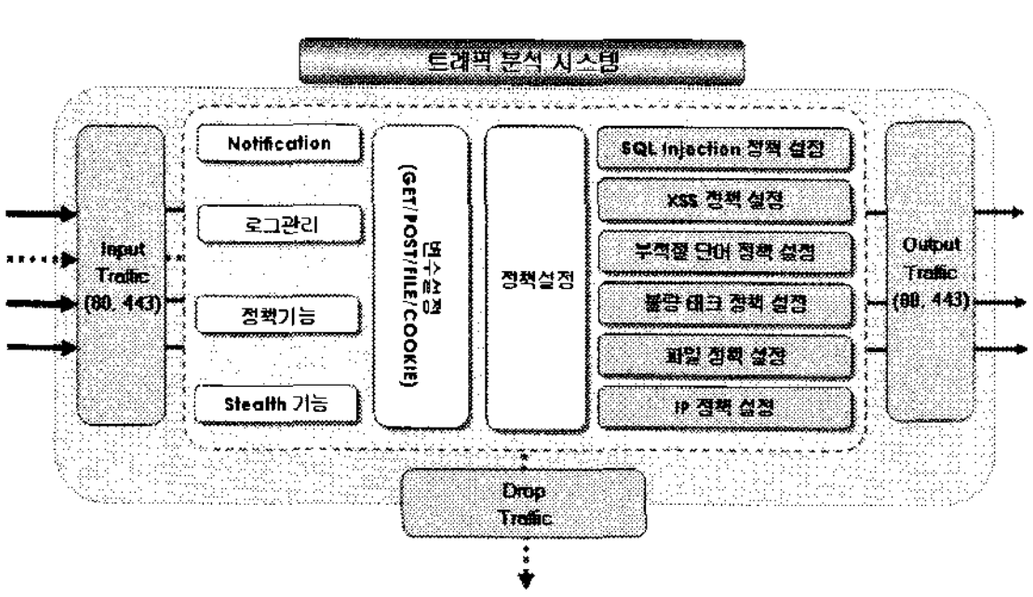


그림 3. 트래픽 분석 시스템
Fig 3. Traffic Analysis System

SQL Injection 공격, XSS 스크립트 공격, 불량 태그의 검출은 다음과 같은 입력문자 ', ", /, \, :, ;, Space, 〈, 〉와 같은 제한된 특수문자 목록을 토대로 검출하였으며, SQL 쿼리로 입력할 수 있는 UNION, SELECT, DELETE, INSERT, UPDATE, DROP 등과 같이 DBMS 쿼리를 직적적으로 포함하는 형태의 입력태그들 또한 직접적인 입력을 차단하기 위해 제한된 특수 문자 목록에 포함하여 차단하였다.

본 논문에서 제안한 트래픽 분석 시스템은 Input traffic을 대상으로 로그관리, Notification, 정책기능, Stealth 기능으로 구성된다. 유입되는 트래픽에 대해서는 정책설정을 통해 구성된 데이터와 비교되어 처리되며, OWASP에서 제안한 크로스사이트 스크립팅(XSS) 취약점(A1, A5), SQL Injection 취약점(A2), 원격 파일 인젝션(A3), 취약한 인증 및 세션관리(A7), URL 접속 제한 실패(A10) 등등에 관련된 사항을 중점적으로 분석하고 판단한다. 〈그림 4〉는 본 논문에서 제안한 트래픽 분석 시스템의 흐름도를 나타낸다.

정상적인 트래픽의 경우에는 웹 어플리케이션으로 전달하여 정상적인 결과를 사용자에게 전달하도록 구성되어 있으며, 비정상적인 트래픽의 경우에는 사용자에게 웹 페이지가 없다는 정상적인 notification page를 보여주게 처리하고, 내부적으로는 웹 어플리케이션으로 트래픽을 전달하지 않고, 로그관리를 통해 로그파일로 저장되고, 바로 Drop 처리 하여 원천적으로 웹 어플리케이션 시스템으로의 접근을 봉쇄하여 보다 안정적인 서비스를 운용할 수 있게 하였다.

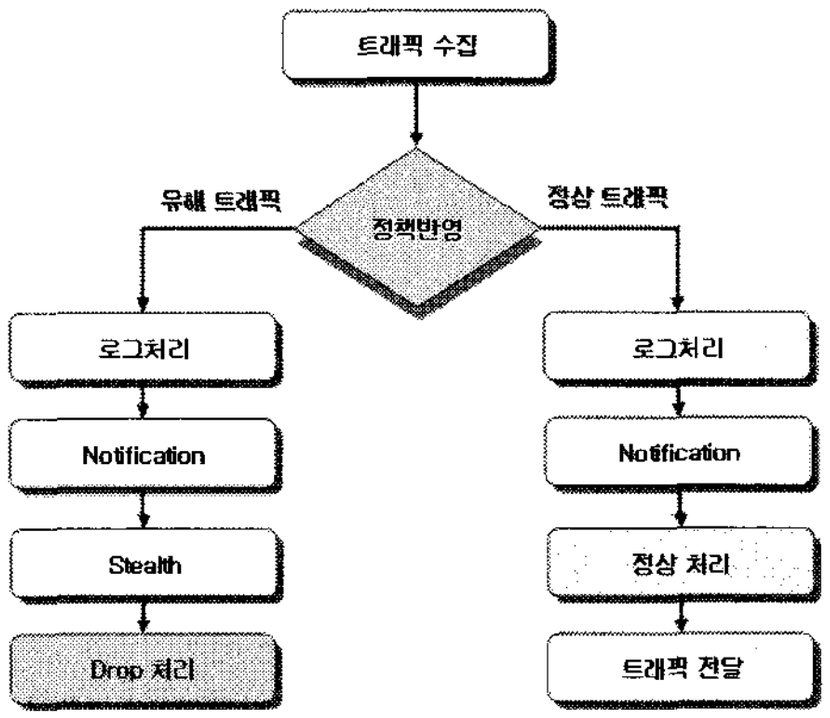


그림 4. 트래픽 분석 시스템 흐름도
Fig 4. Traffic Analysis System Flow Chart

정책 데이터는 공격 트래픽의 경우 지속적으로 관리되어져 새로운 패턴의 공격형태도 즉각적으로 대응하여 반영할 수 있는 형태로 구성되어 있다.

Ⅳ. 실험결과

본 논문에서 제안한 웹 어플리케이션으로 유입되는 트래픽 분석을 위한 실험환경은 〈그림 5〉와 같다. 웹 서버는 MS Internet Information Server 2003, .NET Framework 3.0, 기반의 ASP.NET 환경으로 구성되어 있다. 외부로부터 웹 어플리케이션을 이용하는 사용자를 대상으로 정상적인 트래픽, 비정상적인 트래픽을 발생하였으며, 기존의 레거시 보안 시스템으로 유입되었을 때 트래픽 분석과, 제안시스템인 트래픽분석 시스템으로 유입되었을 때 트래픽 분석을 통하여 비정상 트래픽을 추출하였다.

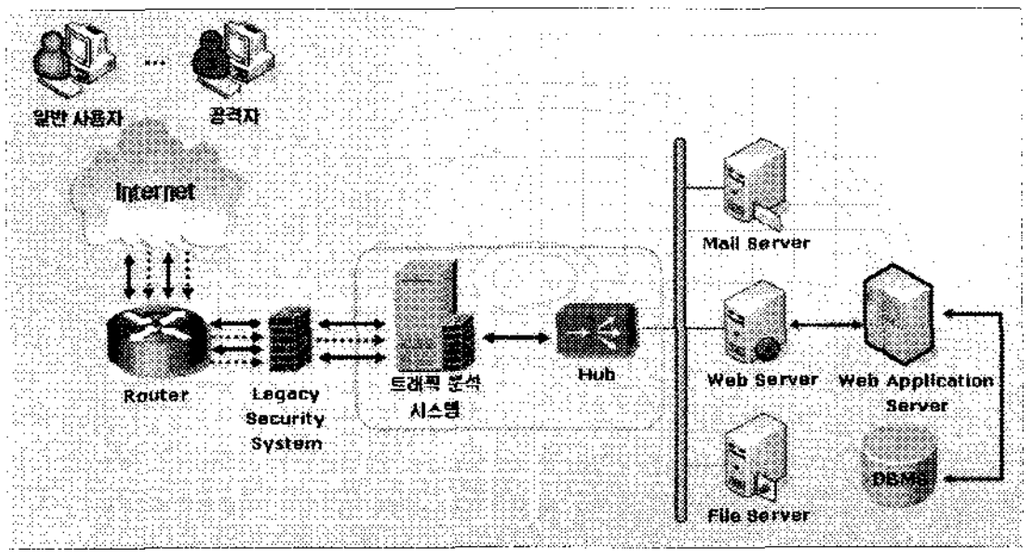


그림 5. 트래픽 분석 시스템 실험환경
Fig 5. Test Environment of Traffic Analysis System

실험을 위한 비정상 트래픽 데이터는 OWASP에서 제안한 10가지 취약점 중에서 가장 빈번하게 발생되고, 웹 어플리케이션으로 직접적으로 전달 가능성이 높은 A1, A2, A3, A4, A5, A7, A10 항목을 대상으로 구성하였다. 실험 데이터는 각 항목이 포함된 100건의 서로 다른 데이터를 3회 실시하여 얻은 평균값으로 산출하였다.

본 연구의 트래픽 분석 시스템을 구현하여 TCP 80포트로 유입되어 웹 어플리케이션으로 전달되는 트래픽을 대상으로 효율적이며, 실시간적으로 분석하였다. 일반 사용자와 공격자는 제안시스템의 외부에 위치하며 웹 어플리케이션에서 제공하는 서비스를 정상적으로 사용하고자 하는 HTTP 요청 트래픽과, 실험을 위하여 데이터를 산출하고, 웹 취약점을 악용한 비정상적인 HTTP 요청 트래픽이 발생하도록 구성한 공격자로로 구분하였다. 정상적인 트래픽의 경우에는 일반적으로 정상적인 요청과 응답처리 결과를 확인하였다.

실험데이터를 기초로 하여 기존의 레거시 보안 시스템에서의 트래픽 분석 결과는 〈그림 6〉과 같다.

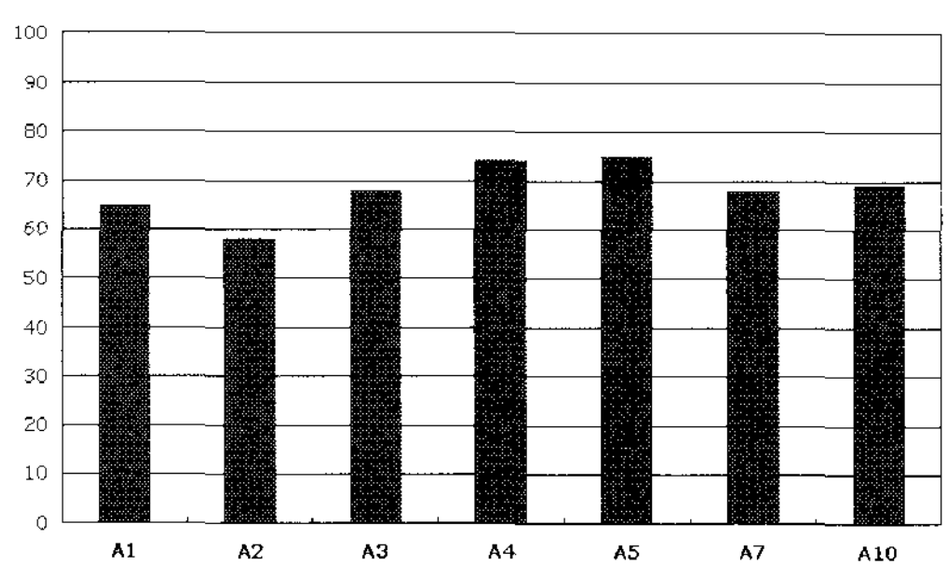


그림 6. 기존 레거시 보안시스템의 트래픽 분석
Fig 6. Traffic Analysis of Legacy Security System

제안시스템의 경우 비정상 트래픽 검출 결과를 〈그림 7〉와 같이 실험 데이터를 토대로 확인하였다.

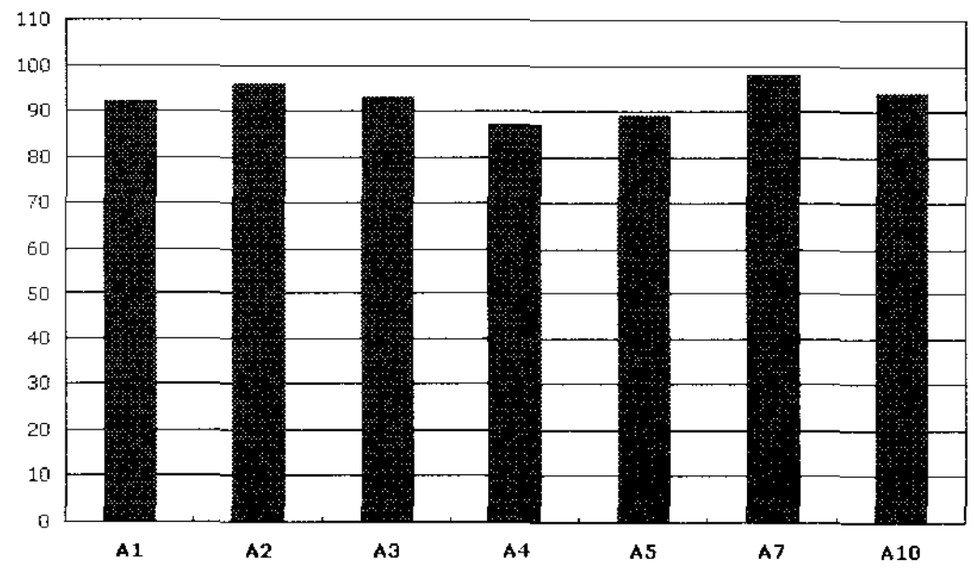


그림 7. 제안시스템의 트래픽 분석
Fig 7. Traffic Analysis of Proposed System

실험환경을 토대로 웹 어플리케이션으로 유입되기 전에 발생된 비정상 트래픽을 검출한 결과를 나타낸다.

두 가지 실험데이터에서 확인한 결과 제안시스템은 기존의 레거시 보안시스템의 트래픽분석 결과보다 비정상 트래픽 검출 효율이 높아진 것을 볼 수 있다.

크로스 사이트 스크립팅(A1), SQL 인젝션 취약점(A2), 원격 파일 인젝션(A3), 불안전한 직접 객체 참조(A4), 크로스 사이트 요청 변조(A5), 취약한 인증 및 세션관리(A7), URL 접속 제한 실패(A10) 항목의 취약점 분석 결과는 웹 어플리케이션의 취약점을 악용할 수 있는 피싱(Phishing), 또는 보다 진보된 신종 공격인 파밍(Pharming), 개인정보 보호, 시스템 훼손, 데이터 변조, 취약점을 악용한 손실 등의 피해를 을 막을 수 있었으며, 기존 레거시 보안 시스템과 제안시스템을 비교하였을 때 그 효율성은 상대적으로 높은 것으로 나타났다. 100건의 실험 데이터를 기초로 하여 기존 레거시 보안 시스템과 제안 시스템과의 비정상 트래픽 분석 결과는 〈표 2〉와 같다.

표 2. 트래픽분석 결과
Table 2. Result of Traffic Analysis

구분	기존시스템 평균값	제안시스템 평균값	제안시스템 효율
A1	65	92	27 %
A2	58	96	38 %
A3	68	93	25 %
A4	74	87	13 %
A5	75	89	14 %
A7	68	98	30 %
A10	69	94	25 %

기존의 레거시 보안 시스템의 경우 이미 설정되어 반영된 정책으로 인하여 평균 68.14%의 비정상 트래픽 검출을 나타냈으며, 제안시스템의 경우에는 새로운 형태의 비정상 트래픽도 좀 더 다양한 정책 자료를 기반으로 실시간 반영되어 평균 92.71%의 비정상 트래픽 검출을 나타내어, 기존 레거시 보안시스템보다 제안시스템이 보다 안정적이고, 높은 수준의 비정상 트래픽 검출을 확인 하였다.

제안시스템의 트래픽 분석 효율은 〈표 2〉와 같은 정도를 나타낸다. 기존 레거시 보안 시스템과 비교하여, 크로스 사이트 스크립팅(A1), SQL 인젝션(A2), 취약한 인증 및 세션관

리(A7) 항목은 기존 레거시 보안 시스템에 비하여 비정상 트래픽 검출 빈도가 높게 나와 상대적으로 안정된 트래픽 흐름을 유지할 수 있었으며, 불완전한 직접 객체 참조(A4), 크로스 사이트 요청 변조(A5) 항목은 기존 레거시 보안 시스템보다 어느 정도 성능 향상은 있었지만, 비정상 트래픽을 직접적이고 확실하게 검출하는 것은 제안시스템을 좀 더 보완하는 부분이 있어야 하겠으며, 기존 레거시 보안 시스템보다 제안시스템은 신뢰할 수 있는 트래픽 흐름을 유지할 수 있으며, 웹 어플리케이션으로 정상적인 트래픽을 보내어 안정적이고, 효율적인 사용자 요청(Request) 처리를 확인 하였다.

〈그림 8〉은 실험데이터를 토대로 기존의 레거시 보안 시스템과 제안시스템의 트래픽 흐름을 나타낸다.

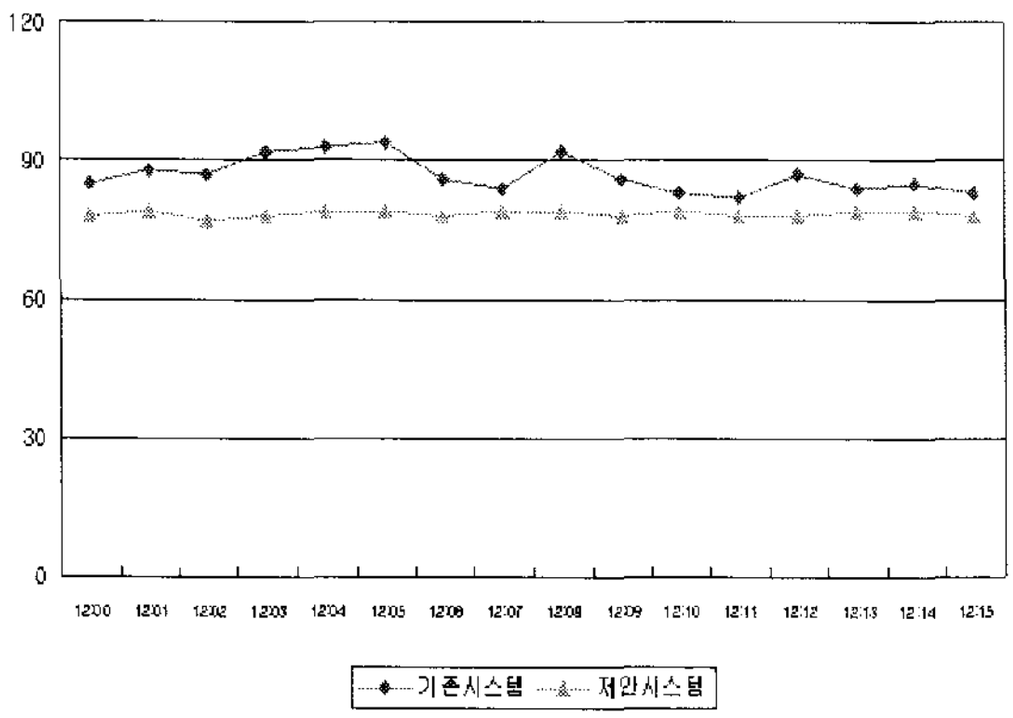


그림 8. 기존시스템과 제안시스템의 트래픽 흐름 비교
Fig 8. Traffic Flow Comparison of Legacy Security System and Proposed System

기존 레거시 보안 시스템의 트래픽 흐름을 보면 실험데이터의 흐름에 따라서 민감한 형태의 트래픽 흐름을 볼 수 있다. 이것은 비정상 트래픽을 제대로 검출하지 못하는데서 오는 흐름으로 판단된다.

비정상 트래픽의 흐름에 따라 제안시스템의 트래픽 흐름을 보면 비정상 트래픽이 차단되어 기존 시스템보다 안정적인 트래픽 흐름을 보이고 있는 것을 확인할 수 있다.

V. 결론

본 논문에서는 웹 어플리케이션으로 유입되는 트래픽을 분석하여 웹 어플리케이션을 목적으로 공격하는 트래픽을 차단하여 비정상트래픽으로부터 공격을 방지하는 시스템을 제안하였다. 제안된 시스템은 분류된 비정상 트래픽을 대상으로 OWASP(Open Web Application Security Project)에서 권고한 웹 어플리케이션 보안 취약점을 바탕으로 공격 패턴을 분석하고, 실시간 공격탐지 및 차단을 하여, 웹 어플리케이션의 보안 취약성을 이용한 다양한 공격 패턴을 찾아내고, 악의적인 외부의 서비스 요청을 지능적으로 막아내는 역할을 하며, 기존의 레거시 보안 시스템과 비교하여 전체적으로 약 25.57%의 비정상 트래픽 검출 효율을 높였다. 또한 비정상 트래픽을 즉각적이며, 효과적으로 차단하여 유해한 공격 트래픽으로부터 공격 방지 효율을 높여, 보다 안전하고, 높은 수준의 웹 어플리케이션 서비스를 유지할 수 있는 것을 실험 결과를 토대로 확인하였으며, 트래픽 흐름도 안정된 형태의 그래프를 통하여 확인하였다.

본 논문의 향후 연구과제는 제안한 트래픽분석 알고리즘을 적용하여 점차적으로 해킹 수단을 전파하는 도구로 변모하고 있는 웹 어플리케이션을 보호하고, 피싱이나 파밍에 악용될 수 있으며 네트워크 트래픽을 가로채 원래의 트래픽에 감염을 위한 데이터를 삽입, 해당 서브네트워크에 존재하는 모든 취약 시스템을 감염시키는 ARP 스푸핑 공격을 방어할 수 있는 공격방지 알고리즘을 구축하고, 트래픽 변조를 봉쇄하여 한 차원 높은 수준의 웹 어플리케이션 보안 서비스를 구축하는 것이다.

참고문헌

[1] 허영준, "IDS 기술동향", 한국전자통신연구원 보안게이트웨이 연구팀, 2000

[2] 장문수, 구향옥, 오창석 "유해 트래픽 분석을 이용한 침입 방지", 한국컴퓨터정보학회논문지, 제10권 4호 (pp.173-179), 2005.9

[3] Crothers, Tim, "Implementing Intrusion Detection Systems : A Hands-On Guide for Securing the Network", John Wiley & Sons Inc., 2002

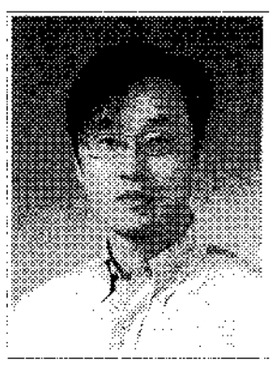
[4] 손동식, "보호대상별 IPS 방법론의 진화", 데이터넷, 2007. 03

[5] Michael Rash, and Angela D., "Intrusion Prevention And Active Response : Deploying Network and Host IPS", Oreilly & Associates Inc., 2005

[6] Omar Santos, Jazib Frahim, "Cisco ASA : All-in-One Firewall, IPS, and VPN Adaptive Security Appliance", Cisco Press., 2006

[7] Shklar, Leon/Rosen, Richard, "Web Application Architecture : Principles, Protocols and Practices", John Wiley & Sons Inc., 2003
[8] OWASP, "http://www.owasp.org/", 2007.
[9] 신동윤, "웹 서비스시대의 보안 솔루션", 온더넷, 2007. 1.
[10] Cross, Michael/Kapinos, "Web Application Vulnerabilities : Detect, Exploit, Prevent", Elsevier Science Ltd., 2007

저 자 소 개



장 문 수

1997년 2월 충주대학교 전자계산학과 (공학사)
2005년 8월 충북대학교 컴퓨터공학과 (공학석사)
2007년 3월~현재 충북대학교 컴퓨터 공학과 박사과정
〈관심분야〉 정보보안, 네트워크 보안, 시스템 보안, 망 관리

오 창 석

1978년 2월 연세대학교 전자공학과 (공학사)
1980년 2월 연세대학교 전자공학과 (공학석사)
1988년 8월 연세대학교 전자공학과 (공학박사)
1982년~1984년 한국전자 통신연구원 연구원
1985년~현재 충북대학교 전기전자컴퓨터 공학부교수
1990년~1991년 Stanford 대학교 객원교수
2007년~현재 한국엔터테인먼트산업학회 회장
〈관심분야〉 컴퓨터네트워크, 뉴로컴퓨터, 정보보호