웹 어플리케이션 특성 분석을 통한 공격 분류
(Attack Categorization based on Web Application Analysis)

서 정 석 † 김 한 성 † 조 상 현 † 차 성 덕 ††
(Jeong Seok Seo) (Han Sung Kim) (Sang Hyun Cho) (Sung Deok Cha)

요 약 최근 웹 서비스의 증가와 함께 웹 서비스에 대한 공격과 그 피해 규모는 증가하고 있다. 그러나 웹 서비스에 대한 공격은 다른 인터넷 공격들과 성격이 다르고 그에 대한 연구 또한 부족한 현실이다. 더욱이 기존의 침입 탐지 시스템들도 웹 서비스를 보호하는데 적합하지 않다. 이 연구에서는 먼저 웹 공격들을 공격 발생 원인과 공격 탐지 관점에서 분류하고, 마지막으로 위험성 분석을 통하여 웹 공격들을 분류하였다. 이를 통해 웹 서비스를 보호하기 적합한 웹 서비스 특화된 침입 탐지 시스템을 설계, 개발하는데 도움을 주고자 한다.

키워드 : 웹 공격, 공격 분류, 침입 탐지, 네트워크 보안, 웹 어플리케이션, 정보전, 취약점 분석

Abstract Frequency of attacks on web services and the resulting damage continue to grow as web services become popular. Techniques used in web service attacks are usually different from traditional network intrusion techniques, and techniques to protect web services are badly needed. Unfortunately, conventional intrusion detection systems (IDS), especially those based on known attack signatures, are inadequate in providing reasonable degree of security to web services. An application-level IDS, tailored to web services, is needed to overcome such limitations. The first step in developing web application IDS is to analyze known attacks on web services and characterize them so that anomaly based intrusion detection becomes possible. In this paper, we classified known attack techniques to web services by analyzing causes, locations where such attack can be easily detected, and the potential risks.

Key words : web attack, attack categorization, intrusion detection, network security, web application, information warfare, vulnerability analysis

1. 서 론

인터넷의 발전과 보급으로 인하여 On line 서비스의 중요성은 날로 높아지고 있다. 그 서비스들 중에서 가장 급속도로 발전한 서비스가 바로 WWW(World Wide Web) 서비스이다. 아래 그림 1은 WWW 서비스의 증가량을 보여주고 있다. 1993년에는 WWW 사이트의 수가 130개이었는데 반해, 1996년에는 약 30만개, 2002년에는 무려 4천만 개 가까이 웹 사이트가 증가하였다.

실제로 웹 서비스는 회사나 기업의 정보전달이나 홍보의 목적 외에도, 근래에는 전자상거래나 기업의 마케팅, 개인을 위한 정보 전달을 위해 그 사용도가 높아지고 있다. 또한 국가차원에서도 웹 서비스의 중요도는 이루 말할 필요가 없을 것이다.[1] 이러한 웹 서비스를 좀 더 안전하게 만들고자 하는 노력의 필요성도 상대적으로 커지게 되었다. 더욱이 최근에는 인터넷 쇼핑몰이나 인터넷 뱅킹 서비스와 같은 웹 서비스 제공만을 목적으로 하는 서버들이 많아짐에 따라, 웹 서비스들을 대상으

· 본 연구는 첨단정보기술 연구센터를 통하여 과학재단의 지원을 받았습니다.

† 비 회 원 : 한국과학기술원 전산학과
jsseo@salmosa.kaist.ac.kr
†† 종신회원 : kimhs@salmosa.kaist.ac.kr
shcho@salmosa.kaist.ac.kr
논문접수 : 한국과학기술원 전산학과 교수
심사완료 : cha@salmosa.kaist.ac.kr
2002년 8월 20일
2002년 10월 14일

1) 1999년 5월 11일 중국 해커들이 미국 백악관 홈페이지(www.whitehouse.gov) 시스템을 침입하여 보안 시스템과 웹 사이트가 24시간동안 마비된 적이 있다. 뿐만 아니라 중국 해커들은 미국정부의 3개 웹 사이트를 공격하고, 미국 에너지부서, 미 내무성, 국립공원서비스의 주요 웹 사이트에 미국을 비난하는 메시지를 남겼다. (CNN 보도자료)

로 하는 공격들에 더욱 빠르게 반응할 수 있고, 웹 서비스에 대한 침입 탐지 false alarm을 줄이도록 노력한 웹 서비스에 특화된 침입 탐지 시스템의 필요성이 급증하고 있다.

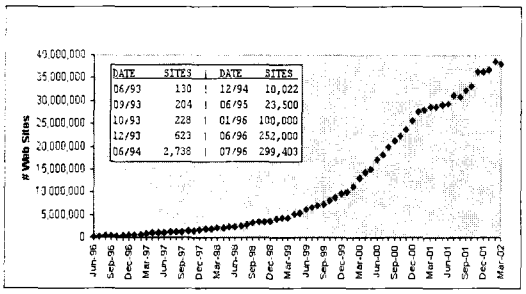


그림 1 WWW 서비스의 증가량[1]

이 논문에서는 웹 공격들을 탐지하는데 도움을 주기 위하여, 웹에 대한 공격들을 알아보고, 웹 공격들을 공격 특징에 따라 분류하였다.

1.1 웹 공격의 정의

웹 어플리케이션이란 웹을 이용하는 소프트웨어의 집합이다. 즉 웹 브라우저나 웹 서버뿐만 아니라, 웹 서버와 같이 서비스를 제공하는데 사용되는 3rd party 어플리케이션[2]과 어플리케이션에 사용되는 데이타를 모두 포괄하는 의미이다.

웹 공격은 웹 서비스를 제공하는데 필요한 웹 어플리케이션을 공격하여 정상적인 웹 서비스를 방해하거나 권한 없는 정보를 습득하는 일련의 행위를 말한다. 그러나 이 논문에서는 보호하고자 하는 관점을 클라이언트를 제외한 웹 서버의 영역만 보기로 한다.

1.2 웹 서비스의 특징과 공격 분류의 필요성

표 1은 인터넷 서비스와 웹 서비스의 특징을 비교해 보았다. 첫 번째로 여러 인터넷 서비스들을 서비스 접근 허용 관점에서 비교해보면 웹 서비스는 telnet이나 ftp 등의 다른 인터넷 서비스와는 달리 개방적인 서비스이다. 로그인을 통해 사용자의 인증을 요구하는 웹 서비스들도 로그인을 위한 페이지까지는 임의의 사용자가 접근할 수 있을 뿐만 아니라, 대부분의 웹 서비스들은 회원이 아닌 다른 외부인 들을 위한 서비스 페이지도 따로 가지고 있는 서비스들이 많다. 즉 방화벽과 같은 접근 관리 시스템은 웹 서비스를 보호하기 위한 시스템으로 부적합하다.

둘째로 웹 서비스는 다른 telnet이나 ftp 등의 서비스와는 달리 그 구조가 계층적 구조를 가지고 복잡한 시스템으로 구성되어있다는 특징이 있다. 여기서 말하는 웹 서비스의 계층적 구조에 대해서는 III장에서 자세히 알아보도록 한다.

표 1 인터넷 서비스들의 특징

인터넷 서비스 / 특징	Telnet	FTP	E mail	News	WWW
서비스의 복잡성	단순	단순	단순	중간	복잡
접근 관리의 효율성	효율성이 높다	효율성이 높다	효율성이 중간	효율성이 낮다	효율성이 낮다
대표적인 보호 시스템	방화벽 VPN IDS	방화벽 VPN IDS	Anti Virus Mail Filter	None	IDS(일부 보호 가능)

웹 공격을 분류하고 이를 통하여 웹 공격의 특성을 파악해야 하는 이유는 웹 공격이 다른 인터넷 공격들과는 상이한 성격을 지니고 있기 때문이다. 그러므로 웹 서비스를 보호하기 위해서는 웹 서비스에 특화된 침입 탐지 시스템이 필요하다.

첫째, 웹 서비스는 다른 인터넷 서비스들과 달리 기업 정보나 국가 이미지 제공을 목적으로 한다. 그리고 최근에는 전자상거래와 같은 비즈니스 용도로도 많이 사용된다. 따라서 웹 서비스의 파괴는 기업이나 국가의 이미지 손상과 곧바로 연결되는 것은 물론 경제적인 손실에 이르기까지 중대한 손해를 초래할 수 있다. 게다가 이러한 웹 서버들은 기업이나 조직으로 침투하기 위한 악의적인 공격의 발판으로 사용되기 때문에 더욱 위험하다.

둘째, 웹 서비스의 활용이 많아짐에 따라 웹 서비스의 요구 수준도 점점 더 복잡하고 기능이 다양한 서비스를 요구하고 있다. 게다가 웹 컨텐츠들은 매우 빠른 속도로 증가하고 있으며 이를 기반으로 하고 있는 웹 서버나 웹 컨텐츠들은 매우 복잡한 시스템으로 구성되어있다. 시스템의 구성이 복잡하면 복잡할수록 취약점이 존재할 수 있는 확률이 높아지고, 취약점을 발견하기가 어려워진다.

셋째, 기존의 침입 탐지 시스템들은 웹 서비스를 위한 웹 IDS로 적합하지 않다. 웹 컨텐츠의 내용에 종속적인 웹 공격의 경우 웹 공격을 효과적으로 탐지할 수 있는 signature를 만들기 어렵다. 또 대부분의 범용 IDS들이

2) 웹 서버를 제외한 웹 서비스를 제공하기 위해 사용되는 모든 응용프로그램을 가리킨다. 예를 들면, cgi 모듈이나 database 등이 있다.

웹 서버의 종류(IIS, apache 등등)에 관계없이 모든 종류의 웹 서버 공격에 대한 signature를 가지고 있기 때문에 사용자가 필요로 하는 signature만을 제공해 주지 못한다. 예를 들면, 기존의 범용 IDS를 사용하여 웹 서비스를 보호하고 있는 사이트에 해당 웹 서버와 관련이 없는 공격은 불필요한 false alarm을 낼 수 있다. 또 다른 관련 없는 인터넷 서비스들의 공격을 통해 침입 탐지 시스템에 과부하를 걸어 놓고 실제 중요한 웹 공격을 사이사이에 수행하는 방법을 사용하면 오히려 웹 서비스를 위한 침입 탐지를 못할 수도 있다. 따라서 웹 서비스를 보호하기 위해서는 보호하고자 하는 웹 사이트의 성격을 반영한 웹 서비스에 특화된 침입 탐지 시스템이 필요하다.

넷째, 다양한 종류의 웹 사이트 성격을 반영한 웹 IDS가 필요하다. 다양한 웹 서비스들은 목적에 따라 서비스 내용 자체의 차이가 있을 수도 있고, 웹 서비스의 내용을 전달하기 위한 컨텐츠 코드가 다를 수도 있기 때문에, 여러 가지 특성에 따라 웹 사이트의 성격이 달라진다. 따라서 탐지 기법의 관점에서 보면 웹 서비스의 코드를 변조하는 공격과 같이 웹 사이트의 성격을 이용한 공격들은 기존의 signature based IDS로는 탐지가 불가능하다. 이런 공격들을 탐지해내기 위해서는 각각의 웹 사이트들에 대하여 각각의 정상적인 사용들을 프로파일(profile)하고 이것을 바탕으로 비정상적인 공격들을 탐지해내야 한다.

이와 같이 일반적인 인터넷 서비스들과 상이한 특징을 가지고 있는 웹 서비스를 효과적으로 보호하기 위해서는 웹 공격의 분류를 통해 웹 공격들의 특징을 알아보고, 웹 서비스를 제공하는 웹 사이트의 성격을 반영한 웹 서비스 특화된 침입 탐지 시스템을 만들어야 한다.

1.3 논문의 구성

이 논문에서는 웹 서비스에 특화된 침입 탐지 시스템을 설계하기 위한 목적으로 웹 공격들의 특징을 파악하기 위하여 웹 공격들을 분류한다. 먼저 웹 공격을 웹 공격들을 공격의 원인과 원인의 위치에 따라 분류하여 웹 공격의 특징을 파악하고, 분류된 웹 공격들을 탐지하기 위하여 공격 탐지 기법과 탐지 위치의 관점에서 어떤 특징을 가지는지 알아본다. 2장에서는 관련 연구들을 통하여 웹 공격들이 어떻게 분류되었는지 소개하고, 3장에서는 웹 공격 분류에 웹 서비스의 특징을 반영하기 위하여 웹 어플리케이션의 특징과 구성 요소에 대해서 알아보도록 한다. 4장에는 실제 웹 공격의 원인과 원인의 위치에 따라 웹 공격들을 분류하고 이를 바탕으로 웹 공격의 탐지 기법과 탐지 위치의 관점에서 웹 공격이 어떤 특징을 가지는지 알아본다. 5장에서는 결론과 향후 연구 계획에 대해서 토의한다.

2. 관련 연구

웹 IDS 개발을 위하여 미국의 Sanctum 사와 Zurich IBM 연구소에서 웹 공격들을 몇 가지로 분류하였다. 이 장에서는 두 곳의 웹 공격 분류를 고찰하고 공개된 네트웍 IDS인 snort가 제공하는 웹 관련 rule들을 살펴보자.

2.1 Sanctum Inc. 연구

미국 Sanctum에서 웹 어플리케이션 IDS를 만들기 위한 목적으로 웹 어플리케이션을 그림 2와 같이 여러 계층으로 나누어 각각의 공격들이 웹 어플리케이션의 어떤 계층에서 발생하였는지를 구분하였다[2]. 이 연구에는 분류된 12개 유형의 공격들이 실제 웹 어플리케이션의 계층적 구조의 어느 부분에 영향을 미치는 가에 대한 연구결과를 제공한다. 이 연구에서 분류한 공격 분류의 방법이 이제까지 많이 사용해 왔던 유형들로 분류되어있기 때문에 이해하기 쉽다는 장점이 있지만, 직관적으로 분류되어있기 때문에 유형구분이 완벽[3]하지 못하다는 단점이 있다.

예를 들면, 웹 서비스에서 사용자의 입력을 받아들이는 입력필드를 overflow시켜 시스템을 마비시키는 공격의 경우, 공격 원인 측면을 보면 입력 필드를 overflow하여 공격을 수행하기 때문에 이 공격은 "Buffer Over flow" 유형에 들어간다. 그러나 결과 측면을 보면 공격의 결과가 시스템을 마비시켜 정상적인 서비스를 방해하기 때문에 "Application DOS" 유형에 속한다. 이와 같이 하나의 공격이 동시에 여러 유형에 속하는 경우가

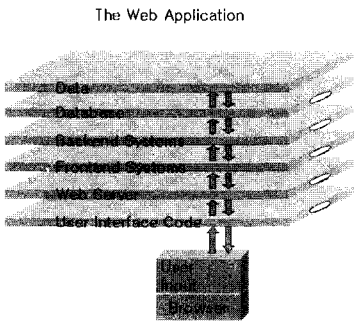


그림 2 Sanctum Inc. 웹 어플리케이션의 계층

3) 완벽한 분류(*Complete partition*) : 각 부분집합들에 공통된 요소가 없어야 하고, 어떤 한 구성요소도 하나의 부분집합에는 속해야 한다(a division of the elements in the set into disjoint subsets)[3].

발생한다. 또 공격을 분류한 관점이 어떤 유형은 공격의 원인에 따른 관점이고, 다른 유형은 공격의 결과의 관점에서 나누어져있기 때문에 공격의 분류가 체계적이지 못하다는 단점이 있다.

표 2 Sanctum Inc. - 웹 공격 분류

Threat Category	*Description*	*Consequence*
Code Scanning Server/Client	Browsing source code	Learn vulnerabilities
Cookie Poisoning	Changing cookie content	User impersonation
Hidden Manipulation	Changing hidden HTML values	eShoplifting
Forceful Site Browsing	Use URL address line	Access sensitive data
Third Party Misconfigurations	Default or improper s/w configuration	Aceess OS or data
Identified Vulnerabilities	Published vendor bugs	Access OS, crash system, access sensitive data
Buffer Overflow	Overflow field input	Access sensitive data, or crash site/application
Debug Options & Backdoors	Change code setting	Access code/application as developer or admin
Parameter Tempering Server/Client	Removal or alteration of expected parameter fields	Access OS or sensitive data
Stealth Commanding	Use meta code	Access OS or control application at OS level, site defacement
Cross Site Scripting	Use URL meta code to insert trojan code	server-side exploitation, access sensitive data
Application DOS	Invalid data input	Crash server/application

2.2 Zurich IBM 연구

Zurich IBM Research 연구소에서는 웹 어플리케이션 IDS를 만들기 위한 목적으로 웹 공격들을 표 3과 같이 분류하고, 해당 공격들을 탐지하기 위한 IDS를 설계하였다[4].

이 연구의 목적은 웹 공격들의 분류보다 웹 IDS 설계에 많은 중점을 두고 있다. 웹 IDS의 공격을 탐지하기 위한 소스를 웹 로그로부터 얻어서 각각의 parser, pattern, combination, refine, suspicious-hosts, trusted, decision, print 모듈별로 그 기능을 설계하여 웹 공격을 탐지하고자 하였다. Sanctum 연구와는 달리 웹 IDS를 웹 어플리케이션 계층의 수직적 관점에서 보지 않고, 그림 3과 같이 웹 서비스를 수평적 관점으로 보고 있다.

표 3 Zurich IBM의 공격 분류

- Penetration of the system via HTTP server vulnerabilities
  - Vulnerable CGI program requests
  - Password guessing
  - Access to sensitive information
- Denial-of-service attacks
  - Repeated accesses to non-existing resources
  - Repeated accesses to resources that cause server errors
- Legal but undesirable activity
  - Singular/outlandish use of the HTTP protocol
  - Sensitive documents accesses
- Policy violation (when used on firewall HTTP proxy)
  - External / internal policies governing access to web sites

그러나 표 3과 같이 이 연구의 공격 분류가 논리적이지 못하다는 단점을 가지고 있다. 표 3의 분류들을 보면 "Penetration of the system", "Denial of service", "Policy violation" 유형은 공격의 결과에 따른 분류인데 반해, "Legal and undesirable activity" 유형의 경우는 공격의 원인에 따른 분류이다. 또한 하나의 공격이 여러 공격 유형에 중첩되는 경우도 발생한다. 예를 들어 Microsoft의 IIS 웹 서버 4.0과 5.0에 Unicode character를 이용하여 시스템을 침입할 수 있는 공격[4] 이 존재하는 데, 이 공격의 경우 원인 측면에서 보면 Unicode character를 이용하였기 때문에 "Legal but undesirable activity" 유형에 속하지만 결과 측면에서 보면 시스템을 침입하는 공격이기 때문에 "Penetration of the system" 유형에도 속한다.

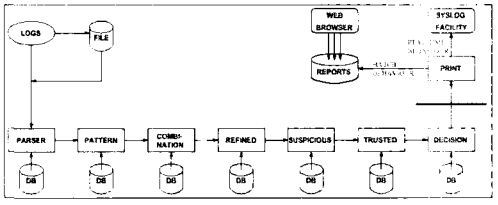


그림 3 Zurich IBM - 웹 IDS의 layout

2.3 Snort(NIDS)

Snort는 대표적인 signature based network IDS이

4) KA-2000-039(CERTCC-KR)

다[5]. Snort는 소스와 공격을 탐지하기 위한 signature가 공개되어있기 때문에 침입 탐지 시스템을 연구하는 목적으로 다양하게 사용된다. 표 4는 snort의 웹 서비스에 관련된 탐지 signature들 중 한 가지를 보여준다.

Snort는 signature를 가지고 패턴을 비교하여 공격을 탐지하는데, signature에서 "content" 필드의 내용이 실제 공격을 탐지하는데 사용되는 signature이다. 예를 들어 표 4의 signature를 보면 외부에서 80번 포트(HTTP 서비스)를 이용하는 TCP 세그먼트에 "/hsx.cgi" 와 "../../", "%00" 이라는 내용이 있으면 해당 세그먼트가 'WEB-CGI HyperSeek directory traver sal attempt' 공격임을 나타낸다. Snort는 취약점 공개 사이트 등을 통해 이미 공격 패턴이 잘 알려져 있는 공격들만 탐지할 수 있다. 따라서 웹 컨텐츠에 따라 공격 패턴이 매우 다양한 공격들이나, 공격 패턴을 정형화하여 signature로 만들 수 없는 공격들은 이러한 방법으로는 탐지할 수 없다.

표 4 snort - 웹 관련 탐지 signature

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS
80 (msg:"WEB-CGI HyperSeek directory traversal
attempt"; uricontents:"/hsx.cgi"; content:"../../"; con
ten t:"%00"; flags:A+; reference:bugtraq,2314; refe
rence :cve,CAN-2001-0253; classtype:web-application-
attack; sid:803; rev:2;)
```

Snort(1.8.6 version)의 rule set을 보면, 각각의 signature들이 그룹으로 묶여 있다. 이 중에서 웹 서비스에 관련된 그룹들은 web-misc, web-iis, web front page, web-coldfusion, web-cgi, web attacks의 6개이다. 전체 signature의 개수는 1267개이고, 이 중 웹 서비스 관련된 signature는 아래의 표 5와 같다.

만약 웹 서버로 Microsoft IIS 서버만을 사용하는 시스템을 보호하고자 한다면 이 시스템에서 사용할 수 있는 웹 서버 관련 signature는 웹 서버를 위한 signature 343(208+88+47)개 중에서 IIS 서버를 위한 signature로 135(88+47)개 밖에 사용할 수 없다. 실제로 snort에 있는 웹 관련 탐지 signature들의 개수가 웹 공격 개수보다 훨씬 적고 pattern matching을 이용한 signature로는 탐지할 수 없는 공격들도 존재하기 때문에 snort를 가지고 웹 서비스를 보호하고자 하는 웹 IDS로 사용하기에는 효과적이지 못하다고 할 수 있다.

이 논문에서는 웹 공격들을 설명하거나 침입 탐지 시스템을 예로 들 때, snort를 기본적인 signature based IDS로 예를 들어서 설명하도록 하겠다.

표 5 Snort의 웹 관련 탐지 signature

웹 관련 signature 그룹	signature 수	내 용
web misc	208	IIS 서버를 제외한 다양한 종류의 웹 서버들을 위한 signature
web iis	88	IIS 서버를 위한 공격 signature
web frontpage	36	
web coldfusion	33	
web-cgi	104	cgi 모듈 공격을 탐지하기 위한 signature
web attacks	47	웹 서버의 종류에 관계없이 발생할 수 있는 공격
총 계	516	

3. 웹 어플리케이션 구성요소

이 장에서는 웹 어플리케이션의 구성요소에 대하여 알아본다. 아래의 그림 4는 웹 어플리케이션 구성요소(component)의 구조를 보여준다.

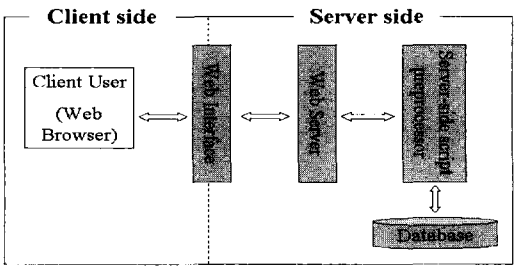
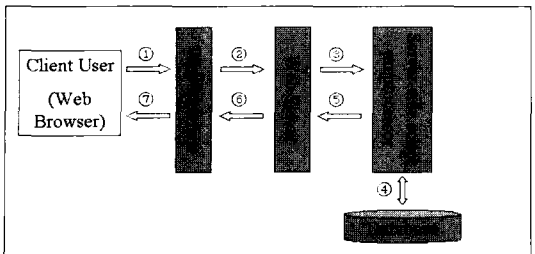


그림 4 웹 어플리케이션 구성요소(component)의 계층적 구조

아래와 같은 query를 웹 서버에 요청하면 그림 5와 같은 과정으로 웹 구성요소들이 동작한다. 웹 서버는 사용자로부터 웹 서비스 요청을 받아 웹 서버와 server side scripts preprocessor에 의해 일련의 연산을 수행하고 수행한 결과를 사용자의 웹 브라우저(web browser)에 보여준다.

http://www.myweb.com/cgi-bin/my.cgi?case=3&no=9



1 사용자의 웹 서비스 요구
2 URL query (http://www.myweb.com/cgi-bin/my.cgi?ca se=3& no=9)
3 cgi 모듈 요청 (my.cgi?case=3&no=9)
4 my.cgi 프로그램 수행 (case와 no 2개의 프로그램 argument를 가지고 필요에 의해 database와 연결하여 프로그램 실행)
5 my.cgi 프로그램의 결과를 웹 서버에게 넘겨준다.
6 cgi 모듈로부터 결과를 받아서 사용자에게 넘겨준다.
7 사용자는 웹 서비스 요구의 결과로 html, cookie 등을 받는다.

그림 5 웹 어플리케이션 구성요소의 작용

3.1 Web Browser

웹 브라우저는 널리 사용되는 웹 서비스 클라이언트 도구이다. 그 종류는 Microsoft Internet Explorer, Net scape Communicator, Hot Java, Mosaic, Opera 등 매우 다양하다. 웹 브라우저는 사용하기 매우 쉽도록 설계되어있다. 대부분의 사용자들은 보안에 관하여 초보이거나 훈련되지 않은 경우가 대부분이기 때문에 웹 공격을 막기 위한 효과적인 지식을 가지고 있지 못하고 웹 공격에 대한 위협을 모르는 사람들이 대부분이다.

실제 웹 공격을 수행하거나 정보를 얻기 위해 해커들이 많이 사용하는 클라이언트 도구로 가장 간단한 telnet을 들 수 있겠다. 이 논문에서는 웹 서비스 보호대상을 클라이언트를 제외한 웹 서버 측의 영역만을 보호대상으로 한다.

3.2 Web Interface

웹 서비스를 사용하는 사용자와 웹 서버가 상호작용하기 위해 사용되는 인터페이스 구성요소이다. 다른 구성요소 들이 프로그램으로 이루어져 있는 반면에 인터페이스 구성요소는 HTTP 프로토콜과 데이타, 코드로 구성된다. 즉, 클라이언트의 웹 브라우저를 통해 사용자에게 보여지는 HTML 코드와 client side scripts 코드, 세션 정보를 위한 쿠키(cookie)뿐만 아니라, 사용자가 웹 서버에 서비스를 요청하는 URI까지 포괄한다. 따라서 이 구성요소는 웹 서비스에서 클라이언트 영역을 제외하고는 사용자 측에 가장 가까운 구성요소이다.

웹 인터페이스의 특징은 사용자가 웹 브라우저를 통해 보고 있는 인터페이스의 클라이언트 소스 코드를 볼 수 있다는 특징이 있다. 때문에 악의적인 사용자는 클라이언트의 HTML 코드를 변경시켜 사용자가 원하도록 웹 요청(request)을 보낼 수도 있다.

그림 7을 예로 들면 웹 인터페이스에는 사용자가 웹 서버에 요청한 URL query와 웹 서버가 사용자 요청의 답으로 보내준 HTML 페이지와 쿠키가 여기에 속한다. 이 때 웹 서버의 요청으로 보내준 HTML 페이지에는 java scripts나 다른 client-side scripts 코드가 포함되어있을 수도 있다. 또 쿠키는 웹 서비스의 특징에 따라 쿠키가 서버로부터 보내어지거나 서버의 요구에 의해 클라이언트 영역에 저장된 쿠키가 서버에게로 보내질 수도 있다.

3.3 Web Server

웹 서버는 사용자의 웹 브라우저와 실제 사용자가 원하는 정보를 연결시켜 주는 역할을 하는 웹 어플리케이션에서 가장 중점적인 역할을 하는 구성 요소이다. 웹 서버는 HTTP/HTTPS 요구를 관리하고, 사용자의 세션을 관리하며, 웹 서비스의 모든 과정을 처리할 수 있도록 담당하는 역할을 한다. 웹 서버의 종류로는 Micro soft IIS, Microsoft PWS, Apache, iPlanet, NCSA, CERN, JAVA Web Server, Netscape Enterprise Server, Oracle Web Server, O' Reilly Web Site, Stronghold, Spyglass 등 다양한 종류가 있다.

3.4 Server side Scripts Preprocessor

Server side scripts preprocessor는 사용자가 웹 서버에 요청한 query가 실제 OS나 database에 적용되기 이전에 프로그램 수행의 필요 여부에 따라 적용되는 과정이다. 실제로 이 과정은 기본적으로는 설치되지 않지만, 웹 서비스에서 필요에 의해 설치되는 모듈이다. 현재 대부분의 웹 서비스에서는 웹 서비스의 기능을 확장하기 위하여 server side scripts preprocessor 모듈들을 사용하고 있다. 이러한 preprocessor의 종류로는 CGI(Common Gateway Interchange), JSP 또는 ASP 등이 있다. CGI는 주로 PHP, Perl, C/C++, Python이나 shell scripts 언어로 주로 표현되어있으며, JSP는 Java scripts 언어로, ASP는 ActiveX로 작성된 스크립트 언어이다.

이 scripts preprocessor들은 웹 서버의 기능을 확장하기 위하여 사용되기 때문에 그 기능에 따라 대부분의 소프트웨어들이 패키지 형태로 제공되는 경우가 많다. 패키지의 규모에 따라서 무료로 배포되는 패키지도 있고 경우에 따라서는 매우 비싼 패키지도 있는데, 대부분

의 패키지들이 효용성과 기능성을 중시하는 성격이 짙어서 보안에 대한 고려가 매우 부족하다. 그렇기 때문에 대부분의 패키지들이 내부적으로 매우 많은 취약점을 가지고 있다.

3.5 Database(data)

데이타베이스는 데이타들을 쉽게 입력하고, 검색하고, 관리할 수 있게 하기 위해서 만들어진 데이타들의 집합이다. 현재 대부분의 웹 서비스는 데이타베이스 없이 이루어질 수 없다고 하여도 틀린 말은 아니다.

가장 흔히 쓰이는 웹 어플리케이션용 데이타베이스로는 MySQL, Oracle, DB2, Microsoft SQL 서버 등이 있다. 데이타베이스 관리자는 해당 데이타베이스의 환경설정에 주의를 기울여야 한다. 일반 사용자들에게는 보여져서는 안 될 데이타가 관리자의 환경 설정 미숙으로 인하여 보여지는 문제점이 발생할 수도 있기 때문이다.

데이타베이스의 오류로 인하여 발생할 수 있는 웹 공격은 이 논문의 관점에서 벗어난 것이기 때문에, 이 논문에서는 웹 공격을 분류할 때 웹 어플리케이션 구성요소로써 데이타베이스를 거론하지 않기로 한다.

4. 웹 공격 분류

이 장에서는 가장 먼저 웹 공격들의 특징을 알아보기 위하여 웹 공격을 공격 발생 원인과 공격 발생 원인의 위치에 따라 분류해 보고 이를 바탕으로 각각의 분류된 유형들이 어떤 특징을 가지는지 알아보도록 한다.

다음으로 각 공격들을 탐지하기 위하여 웹 공격들을 실제 탐지 할 수 있는 탐지 위치와 탐지 기법에 따라 분류를 해보도록 한다. 공격을 탐지하기 위한 탐지의 관점에서는 앞에서 분류한 공격들이 어떻게 분류되는지 조사하여 웹 어플리케이션 IDS를 설계하는데 도움을 줄 수 있을 것이다.

마지막으로 웹 공격들에 대한 위험도를 분석하기 위하여 웹 공격을 공격 결과에 따라 분류해보도록 한다. 웹 서비스를 제공하고자 하는 목적과 내용에 따라 실제 어떤 부류의 공격들이 위험하고 어떤 공격들은 위험하지 않은지를 알 수 있다면, 해당 웹 서비스를 보호할 IDS를 설계하는데 많은 도움을 줄 수 있을 것이다.

4.1 공격 원인과 원인 위치에 따른 분류

웹 공격은 정형화 가능한가에 따라 크게 2가지 부류로 나눌 수 있다. 하나는 해당 공격의 원인이 잘 알려져 있거나 해당 공격의 원인을 이용하는 공격이 정형화된 패턴을 이루고 있어서 공격을 탐지하기 위한 signature를 만들 수 있는 공격들이다. 다른 하나는 공격의 원인이 웹 컨텐츠에 종속적이거나 비정형적이어서 특정한 signature를 만들 수 없는 공격들이다. 즉 바꾸어 말하면 전자의 공격들은 공격을 탐지하기 위한 signature가 있어 해당 signature에 의해 비정상적인 공격이라고 판단할 수 있지만, 후자의 공격들은 해당 공격들을 탐지하기 위한 signature를 만들 수 없기 때문에 signature를 가지고 이 공격들을 탐지해 낼 수 없다. Microsoft IIS 4.0과 5.0에서 Unicode character translation 취약점을 이용하여 시스템을 침입할 수 있는 공격이 존재하는데, 이 공격을 탐지할 수 있는 snort의 signature에 의하면 웹 서비스를 요구하는 URI중에 "/scripts/.. %c1%1c../" 내용이 있으면 해당 공격을 탐지하게 된다. 이 공격은 공격의 원인이 Microsoft IIS 서버의 구현 오류에 있고, 취약점을 공격하는 공격 패턴이 정형화되어있어 탐지 signature에 의해 탐지될 수 있는 공격이다. 반면에 공격자가 자신의 쿠키를 변조하여 다른 사용자인 것처럼 위조하는 공격의 경우 웹 사이트마다 다양한 쿠키 사용 방법을 가지고 있고, 그에 따라 다양한 사용자로 위조가 능하기 때문에 공격의 특정 패턴을 정형화 할 수 없다. 이 경우 공격의 원인이 공격자의 변조에 있고, 공격을 탐지하기 위한 signature를 만들 수 없기 때문에, signature에 의해 탐지될 수 있는 공격이 아니다.

표 6은 웹 공격들을 공격이 일어나는 원인과 원인이 발생한 웹 어플리케이션 구성요소의 위치에 따라 여러 유형으로 분류됨을 보여준다.

표 6 공격 원인과 원인의 위치에 따른 공격들의 분류

정형화 가능성	정형화된 패턴이 있는 공격		정형화된 패턴이 없는 공격 (웹 컨텐츠에 종속적인 공격)		
공격 원인 / 공격 원인 위치	Implemen tation 오류 (가)	HTTP specification 오류 (나)	공격자의 변조에 의한 공격 (다)	비정상적인 parameter 입력에 의한 공격 (라)	과도한 throughput 에 의한 공격 (마)
Web Interface (I)	Type 가-I	Type 나-I	Type 다-I	Type 라-I	Type 마-I
Web Server (S)	Type 가-S	Type 나-S	Type 다-S	Type 라-S	Type 마-S
Pre-processor (P)	Type 가-P	Type 나-P	Type 다-P	Type 라-P	Type 마-P

분류의 기준 중 하나는 공격이 발생한 원인이다. 우선 공격이 정형화 가능한지 여부에 따라 크게 정형화된 패턴이 있는 공격과 정형화된 패턴이 없는 공격으로 나누어진다. 이 중 정형화된 패턴이 있는 공격들은 대부분 취약점 공개 사이트에 해당 취약점들이 공개되어 있다.

여기에 속하는 공격들의 원인은 웹 서버나 server side scripts preprocessor들의 구현상의 오류가 원인이거나 HTTP specification 상의 오류가 원인이다. 반면에 정형화된 패턴이 없는 공격들은 웹 컨텐츠의 내용과 그 내용을 이루고 있는 코드들에 의해 공격의 패턴이 다양하게 나타난다. 따라서 여기에 속하는 공격을 탐지하기 위해서는 웹 사이트의 특성을 반영한 profile을 가지고 침입을 탐지해야 한다. 여기에 속하는 공격들로는 공격자가 소스코드나 데이타 변조가 원인이 되어 일어나는 공격, 비정상적인 parameter 입력이 원인이 되어 일어나는 공격, 과도한 throughput이 원인이 되어 일어나는 공격이 있다. 여기에서 말하는 공격 패턴이란 명시할 수 있는 공격 행위의 일련을 말할 수도 있고, 실제 공격이 수행되고 있는 네트웍 패킷이나 명령어 자체를 나타낼 수도 있다.

다른 한 가지 분류 기준은 공격의 원인이 존재하는 위치이다. 공격이 실제 일어나는 원인의 위치가 웹 어플리케이션 구성요소의 웹 interface에 존재하는지, 웹 서버 혹은 server side scripts preprocessor에 존재하는지에 따라서 나눈 것이다.

Type 가-I 유형에 존재하는 공격들은 공격의 원인이 구현 오류이면서 원인의 위치가 interface에 있는 공격들이다. 위 표 6에서 색깔이 짙은 부분은 실제 웹에 대한 공격들이 존재할 수 있는 부분이고, 색이 없는 부분은 논리적으로 공격이 존재할 수 없는 부분이거나 공격이 존재할 수 있는 확률이 매우 낮은 부분을 나타낸다. 즉, type 가-I 공격은 공격 원인이 웹 interface상에 있으면서 어플리케이션의 구현 오류가 원인이 되는 공격들의 집합이다. 그러나 웹 interface는 URI과 같이 프로토콜을 의미하거나 HTML 코드나 쿠키와 같은 데이타 등을 의미한다. 따라서 구현이 아니므로 해당 유형에 속하는 공격은 존재하지 않는다. Type 가-S 유형에 존재하는 공격들은 공격 원인이 구현에 있으면서 원인의 위치가 웹 서버에 있는 공격들이다. 즉, 웹 서버의 구현상의 문제점이 원인이 되어 발생하는 공격들이 이 유형에 속한다. Type 가-P 유형은 cgi, PHP, ASP들과 같은 server side scripts preprocessor의 구현상의 문제점이 원인이 되어 발생하는 공격들이 속한다.

Type 나-I, 나-S 유형은 HTTP specification 오류가 원인이 되어 발생하는 공격들이 속한다.

Type 다의 공격은 공격자가 쿠키나 HTML 코드 등을 변조함으로써 발생하는 공격이고 type 라의 공격은 사용자가 HTML 페이지나 게시판 등의 입력 form에 비정상적인 입력을 넣음으로써 발생하는 공격이고, type 마의 공격은 사용자가 웹 서버에 비정상적으로 과도하게 요구를 요구하여 서비스에 마비를 가져오는 공격이다. 따라서 type 다, type 라, type 마 공격들은 웹 interface가 원인이 되어 발생하는 공격이다. 따라서 공격 원인의 위치가 웹 서버나 preprocessor가 될 확률은 매우 낮다고 할 수 있다.

4.1.1 Type 가 구현 오류에 의한 공격

Type 가의 공격들은 웹 서버나 server side scripts preprocessor의 구현상의 오류에 의해 발생하는 것으로 프로그램 내에 존재하는 취약점에 의해 공격이 수행된다. 이 공격들은 대부분 보안 권고 사이트에 취약점과 공격 코드, 공격 패턴이 알려져 있고, signature based IDS에 의해 잘 탐지되는 특징이 있다.

Type 가-S 유형에 속하는 공격들은 웹 서버의 구현상의 오류에서 발생하는 취약점이 원인이 되어 발생하는 공격들이다. 여기에 속하는 취약점들은 대부분 취약점 공개 사이트나 웹 서버를 제공하는 회사의 웹 사이트를 통하여 공개되어있기 때문에, 관리자들의 지속적인 관리를 통하여 취약점이 해결될 수 있다. 실제로 웹 서버의 다양한 종류에 따라 그 취약점 내용과 공격 방법이 다르기 때문에, 침입 탐지의 관점에서는 웹 서버의 종류마다 탐지 signature도 다양하게 가지고 있어야 한다.

Type 가-S 공격들의 대표적인 예는 Microsoft IIS 5.0 .printer ISAPI Extension Buffer Overflow Vulnerability 공격이 있다. Windows 2000에 기본적으로 제공되는 서비스로 Internet Printing Protocol(IPP)이라는 서비스가 있다. 이 프로토콜은 산업 표준 프로토콜로써 HTTP를 통해 프린트 작업을 실행하고 통제하기 위해서 사용된다. Windows 2000에서는 ISAPI extension을 통하여 IPP 서비스를 제공한다. 이 ISAPI 서비스를 구성하고 있는 파일 중 msw3prt.dll에는 버퍼의 크기를 체크하지 않아 buffer overflow[6] 공격이 가능해지는 취약점이 존재한다. 따라서 공격자는 이 취약점을 이용해서 원격지에서 IIS 5.0 서비스가 진행중인 취약 서버를 overflow하여 공격자가 원하는 임의의 코드를 실행할 수 있다.

예로 IIS ISAPI 서비스의 unchecked buffer의 특징을 이용했던 공격 중에서 가장 심각했던 공격으로는 CodeRed worm을 들 수가 있다. 표 7은 CodeRed worm에 대한 snort의 signature이다. TCP 세그먼트 중 외부에서 80번 HTTP 서비스로 가는 URI에 "|FF8B8D64 FEFFFF0F BE1185D2 7402EBD3|" 라는 내용이 있으면 이를 CodeRed worm에 의한 공격으로 간주하고 alarm을 낸다.

표 7 Snort signature - CodeRed Worm

```
alert tcp any any -> any 80 (msg: "CodeRed Worm
Defacement Sent" ; flags:PA+; content: "|FF8B8D64
FEFFFF0F BE1185D2 7402EBD3|" ; depth:16;)
```

Type 가 P 유형에 속하는 공격들은 server side scripts preprocessor의 구현상 오류에서 발생하는 취약점이 공격의 원인이 되는 공격들이다. PHP, ASP, JSP 등의 preprocessor들은 웹 서버의 기능을 확장하여 보다 유용하고 효과적인 서비스를 제공하기 위해 사용된다. 그러므로 server side scripts 대부분이 서비스 특징에 맞게 package화되어 개발되는 경우가 많다. 문제는 그런 프로그램들이 대부분 효율성과 편리성에 바탕을 두고 개발되었고, 보안에 대하여 신경을 거의 쓰지 않았기 때문에 상당히 많은 취약점이 존재하고 해당 취약점에 의한 결과도 매우 위험한 공격들이 많다. 또 package도 다양하기 때문에 관리자가 하나하나 package 배포 사이트를 찾아다니며 그 취약점을 찾아내고 패치 하기 어렵다는 문제가 있다. Snort의 web cgi rule에는 type 가 P 유형 공격들을 탐지할 수 있는 signature들이 있다.

4.1.2 Type 나 - HTTP specification 오류를 이용한 공격

Type 나의 공격들은 HTTP specification에 정의를 따르지 않고, 비정상적인 query를 웹 서버에 요청함으로써 웹 서버가 비정상적인 행동을 나타내는 공격들이다. 현재까지 취약점 공개 사이트나 웹 서버 제품을 제공하는 사이트에 HTTP specification의 오류를 이용한 취약점은 발표되지 않았다[5]. 그러나 예를 들어 HTTP 프로토콜 헤더에 있는 DATE 필드를 미래의 시간으로 설정하여 웹에 서비스 요청을 하였을 때 만약 웹 서비스를 마비시킬 수 있으면, 해당 공격은 이 영역에 속한다.

HTTP specification의 오류를 이용한 공격이 발생하면, 해당 공격의 원인의 위치에 따라 원인이 웹 interface에 있다면 type 나-I 유형에 속하고, 공격의 원인이 web server에 있다면 type 나-S 유형에 속하게 된다. 그러나 server side scripts processor는 HTTP 프로토콜을 사용하지 않으므로 type 나-P 유형에 속하는 공격은 논리적으로 존재하지 않을 확률이 크다고 할 수 있다. 현재까지는 이런 공격이 발표되지 않았지만, 이런 종류의 공격이 없다고 단정할 수는 없다. 왜냐하면 HTTP 프로토콜에 대해서는 아직 specification 오류를 이용한 공격이 발견되지 않았지만 다른 프로토콜 TCP, UDP, IP, ICMP 등에 대해서는 specification 오류를 이용한 공격들이 많이 보고되고있기 때문이다. 이와 같이 아직 HTTP 프로토콜에 대해서는 specification을 비정상적으로 사용함에 따라 발생하는 공격이 아직 발견되지는 않았지만 언제라도 이런 종류의 공격이 발생할 수 있다.

4.1.3 Type 다 공격자의 변조에 의한 공격

Type 다의 공격들은 공격자가 소스코드나 데이타 등을 변조하여 웹 서버에 요청을 함으로써 웹 서버는 비정상적인 행동을 보이거나 공격자에게 허가되어서는 안되는 정보를 유출하는 공격이다. 실제로 클라이언트 영역에 있는 공격자가 웹 서버 영역에 있는 서버나 cgi 모듈, 데이타베이스의 내용을 직접적으로 변경하기는 어렵기 때문에, 대부분의 공격이 클라이언트 영역에서 내용을 볼 수 있는 HTML 코드나 client side script, 쿠키 등을 변조하여 공격하기 때문에 공격들의 원인이 웹 interface에 있다. 따라서 type 다 유형의 공격들이 대부분 type 다 I 공격들이고 type 다-S, type 다-P 공격들은 존재할 확률이 매우 낮다.

Type 다-I 유형에 속하는 공격들은 공격자가 웹 interface 구성요소 영역에 속한 코드나 데이타들을 임의로 변조하여서 웹 서비스를 비정상적으로 이용하는 공격들이다. 예를 들면 HTML 코드나 client side script code, 쿠키들은 모두 웹 어플리케이션의 구성요소 중에서 interface 영역에 속한다. 이 공격들은 공격 방법이 웹 컨텐츠의 내용과 서비스의 특징에 따라 매우 다양하기 때문에 공격을 탐지하기가 매우 어렵다. Type 다-I 공격들의 대표적인 예로 쿠키 변조 공격(Cookie poisoning)과 소스 코드(client code source) 변조에 의한 공격을 들 수 있다.

쿠키는 상태(state) 정보를 가지지 못하는 HTTP session에서 상태 정보를 나타내기 위하여 사용하는 방법이다. 쿠키는 사용자 클라이언트의 로컬에 plain text 형태로 저장되며, 웹 사이트마다 쿠키의 내용이 다르지만 주로 사용자의 정보를 담고 있다. 문제는 이러한 쿠키가 암호화되어있지 않기 때문에, 쿠키로부터 해당 사용자의 정보를 훔쳐올 수 있을 뿐만 아니라, 해커는 쿠키의 내용을 다른 사람의 정보로 변조함으로써 마치 다른 사용자가 session을 맺고 있는 것처럼 웹 서버를 속일 수도 있다.

웹 서비스의 가장 큰 특징 중의 하나는 웹 브라우저에서 사용되는 클라이언트 측의 코드들을 사용자가 쉽게 볼 수 있고, 변경할 수 있다는 특징이 있다. 따라서

5) 2002년 6월

공격자는 서버로부터 온 클라이언트 코드를 분석함으로써 서버의 정보를 유추하거나, 클라이언트 코드를 변경하여 웹 서버에 query를 보냄으로써 웹 서버에 권한이 없는 연산을 수행하거나 비정상적인 행동을 하도록 만들 수가 있다. 그림 6은 쇼핑몰 'A'사의 HTML 코드 일부 중에서 java script 코드를 보여주고 있는데, boardDel() 함수는 쇼핑몰 고객게시판에서 게시판의 내용을 삭제하는 연산을 수행한다. 함수의 파라미터 id와 cust_mst_id는 각각 글 번호와 글을 작성한 사용자의 ID를 의미한다.

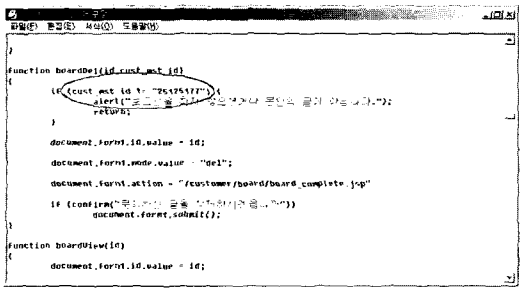


그림 6 쇼핑몰 'A'사의 HTML 코드 일부

이 코드에서 ID를 검사하는 if문을 삭제하거나 주석처리하여 사용자의 권한을 체크하는 부분을 무력화시킬 수 있다. 위와 같이 클라이언트 코드의 일종인 HTML 코드나 java script 코드는 사용자 측에서 코드의 내용을 볼 수 있고, 코드의 내용을 수정할 수도 있다. 이러한 취약점들은 대부분의 쇼핑몰 사이트들의 코드가 패키지화하여 개발되고, 개발단계에서 보안의 요소들을 고려하지 않아서 발생한다. 이 공격들이 일어나지 않게 막으려면 사용자의 상태 검사나 연산 수행의 허가를 검사하는 코드를 클라이언트에서 수행하는 것을 막고 모두 서버에서 수행하도록 코드 개발단계에서 고려하는 방법밖에는 없다. 그리고 위와 같은 공격들을 탐지하고자 한다면, 기존의 signature based IDS로는 불가능하다. 반면에 웹 IDS는 웹 페이지를 요구하는 사용자들의 session에서 중요한 정보들을 기억하고 있다가, 사용자의 요청이 있을 때 정보들이 변경되었는지를 검사하여 위와 같은 공격을 탐지할 수 있다.

4.1.4 Type 라 - 비정상적인 parameter 입력에 의한 공격

Type 라 유형의 공격들은 웹 서비스의 입력으로 악의적인 입력을 넣어서 웹 서비스를 비정상적으로 이용하는 공격이다. URI(Uniform Resource Identifier)를 비정상적으로 입력하거나, 게시판, text form 등에 악의적인 HTML 태그를 입력하여 웹 서버에 이상을 일으키거나 웹 서비스의 정상적인 행위를 방해하는 공격을 예로 들 수 있다. 웹 서비스 입력의 일종인 URI는 웹 interface 구성요소 중 하나이고 게시판이나 text form 등에 입력을 넣어서 웹 서버에 요청을 하는 경우, 게시판이나 text form은 클라이언트의 HTML 소스 코드에 존재하고, 실제로 게시판이나 text form 등에 입력을 넣어서 웹 서버에 요청을 하면 이 요청도 URI를 통하여 웹 서버에 전해지기 때문에 공격의 원인이 웹 interface 구성요소에 있다. 즉, type 라 공격들은 클라이언트 영역의 HTML 코드에 존재하는 게시판이나 text form 입력에 악의적인 입력을 가하거나 악의적인 URI를 이용하여 공격을 하는 공격들이기 때문에 대부분의 공격 원인이 웹 interface에 존재한다. 따라서 type 라의 공격들은 대부분이 type 라-I 유형에 속하고 type 라-S, type 라-P 공격들은 존재할 확률이 매우 낮다.

Type 가-S의 예로 들었던 Unicode character를 이용한 Microsoft IIS 4.0/5.0 웹 서버 내부 명령어 실행 공격의 예를 보면 특별한 Unicode를 이용한 URI를 이용하여 웹 서버를 공격한다. 그러나 이 경우 공격의 원인이 정확하게 Microsoft IIS 서버에 있다고 알려져 있고, 공격 패턴 또한 정형화되어있어 탐지 signature를 만들 수 있기 때문에 type 라-I 영역에 속하는 것이 아니라 type 가-S의 영역에 속한다. 반면에 type 라 I 공격들은 공격 패턴이 정형화되어있지 않고 웹 컨텐츠 내용에 따라 공격 내용이 다양하게 나타난다.

Type 라 I 공격의 예로 비정상적인 URI를 이용한 공격과 게시판을 이용한 오용 공격을 보자. 비정상적인 URI를 이용하는 공격은 웹 서비스 요청을 하는 URI에 정상적인 요청과는 달리 악의적인 데이타나 파라미터 등을 넣어서 웹 서비스를 불법적으로 이용하거나 서비스를 방해하는 공격을 의미한다. 공격자가 요청하는 URI는 정상적인 사용자들이 웹 요청을 하는 URI와 다르기 때문에 정상적인 사용을 profile하고 이 profile을 기반으로 오용 탐지 기법을 사용하여서 해당 공격을 탐지할 수 있다.

예로 쇼핑몰의 상품 가격 정보를 클라이언트 코드인 HTML 코드에 가지고 있거나 상품 가격을 URL의 파라미터로 넘겨주는 웹 사이트의 경우 공격자는 코드를 수정하여 가격을 변조한 후에 상품을 정해진 가격보다 싸게 구입할 수 있다. 이 때 이 공격은 공격자에 의해 코드가 변조되었기 때문에 type 라-I 영역에 들어간다. 여기서는 상품 가격을 URL의 파라미터로 넘겨주는 웹

사이트에서 파라미터 입력을 악의적으로 넣어서 가격을 변경시키는 공격의 예를 들도록 한다. 그림 7은 아래 URL의 실행 결과이다.

http://www.myweb.com/shoppingbag.php3?cate=128&stockno=3256&mode=add&price=10

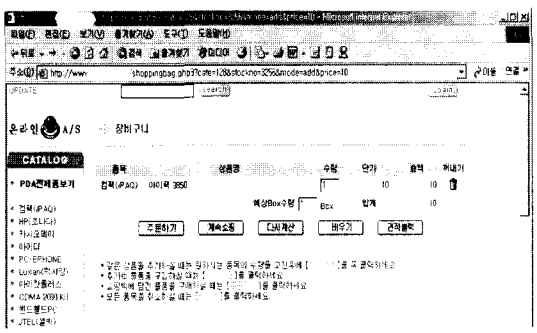


그림 7 쇼핑몰 'B'사의 shopping cart

위 취약점은 대부분의 쇼핑몰 코드들이 개발단계에서 가격이나 상품 정보들을 URL의 파라미터로 넘겨주도록 설계되어있기 때문이다. 이것은 개발 단계에서 보안 요소들을 고려하지 못하고 디버깅이나 모듈 개발의 편의를 위해 대부분의 쇼핑몰들이 이런 방법들로 설계되어 있기 때문이다.

게시판 오용 공격은 게시판이나 input을 위한 form 등에 악의적인 태그나 데이타를 넣어서 웹 서비스를 방해하거나 비정상적으로 만드는 공격이다. 공격의 원인이 웹 서버에 보내는 URI에 있는 클라이언트의 HTML 코드를 통한 악의적인 태그나 데이타이기 때문에 type 라-I 유형에 들어간다. 악의적인 태그로 사용할 수 있는 태그들의 종류를 조사하여 signature화하면, 해당 공격을 탐지할 수 있다. 그러나 실제로 명확히 악의적인 태그라고 규정할 수 있는 태그들의 종류는 몇 가지 되지 않고 이 유형의 공격들도 웹 컨텐츠의 내용이나 웹 서비스의 특징에 따라 공격 내용이 다양하기 때문에 signature를 가지고 공격을 탐지하기에는 적합하지 않다.

아래의 예는 HTML 코드에 'META' 태그를 사용하였는데, 웹 브라우저는 이 HTML 태그를 만나면 해당 URL로 점프한다.

<META HTTP-EQUIV="Refresh" content "0; URL= http://www.my.shopping.com">

최근의 인터넷 홈페이지들에는 게시판이 없는 서비스를 찾아보기가 어렵다. 쇼핑몰 사이트나, 언론, 국가 기관의 홈페이지에 QandA이나 사용자 불편 신고 등을 받기 위해 하나 이상의 게시판을 사용하고 있다. 그런데 이러한 게시판들에 부적절한 HTML 태그를 넣게 되면 해당 게시판의 글을 읽게 되는 임의의 사용자는 해당 HTML 코드에 의해 부당한 서비스를 강제로 당하게 될 수 있다.

이와 같이 악의적인 HTML 코드를 이용하여 임의의 사이트로 강제 점프하게 만드는 공격(forceful brow sing) 외에도 게시판을 여는 순간 클라이언트의 창을 100개 또는 그 이상으로 충분히 컴퓨터가 다운될 수 있을 정도로 여러 개의 창을 동시에 띄우는 공격이 가능하다. 비단 이런 공격이 게시판에서만 나타나는 것은 아니다. Text form으로 이루어진 모든 입력(사용자 가입을 위한 정보 입력 등)에서 나타날 수 있으며, 만약 사용자의 입력으로부터 연산(execute)을 수행하여 결과를 보여주는 사이트 같은 경우, 예를 들어 주소를 입력받아 최단거리를 나타내주는 driving 서비스 경우에 공격자가 실행하고 싶은 실행 가능한 코드를 넣어서 다른 사용자의 세션을 가로채거나, 다른 사용자의 개인 키(private key)를 훔쳐올 수도 있다. 물론 이런 공격들은 웹 서비스의 내용과 구현 방법에 따라 달라진다.

위와 같은 공격을 차단하기 위해서는 게시판이나 text 입력 등의 사용자가 HTML 코드를 입력할 수 있는 부분에 악의적인 HTML 코드가 들어가는 것을 탐지할 수 있어야 하고, 이런 악의적인 코드가 들어가는 것을 막아야 한다.

4.1.5 Type 마 - 과도한 throughput에 의한 공격

웹 서버의 CPU 사용률이나 네트웍 throughput을 과도하게 높임으로써 서비스를 비정상적으로 만드는 DOS (Denial of Service)공격들은 여러 원인에 의해 나타난다. DOS 공격 중에서도 공격의 원인이 정상적인 대규모 동영상(멀티미디어) 파일 요청을 과도하게 요구하여 웹 서비스를 마비시키거나 여러 분산된 호스트에 의해서 정상적인 웹 요청을 과도하게 수행하여 웹 서버를 마비시키는 공격(Distributed DOS)들이 type 마 유형에 들어간다. 다시 말하면 웹 요청 하나하나를 분석하여 보면 정상적인 웹 요청들이지만 이 요청들로 인하여 웹 서비스가 마비되는 공격들이 type 마 유형에 속하는 공격들이다.

Type 마 공격의 경우 웹 요청을 원하는 URI를 여러 호스트에서 과도하게 요구하여 공격을 하기 때문에 비정상적인 입력이 원인이라고 생각할 수도 있다. 하지만 이런 공격의 요청 URI 하나하나는 정상적이고 요청 하나를 놓고 보았을 때는 비정상적이라고 판별할 수 있는 근거가 없다. 이런 정상적인 요청이 여러 호스트에 의해

과도하게 요구되었다는 것이 원인이기 때문에 이 공격은 type 라가 아닌 type 마 유형에 들어간다. 이런 공격들은 정상적인 웹 요청과 달리 동일한 웹 요청이 갑자기 증가하거나, 평상시에는 없던 웹 요청이 갑자기 여러 차례 나타나는 특징이 있기 때문에, 정상적인 웹 사용들을 profile하고 이 profile을 바탕으로 비정상적인 공격들을 탐지할 수 있다. 대다수 type 마의 공격들은 정상적인 웹 요청을 과도하게 여러 번 요청함으로써 발생하는 공격들이므로 공격의 원인이 웹 interface에 있고 type 마-I 유형에 속한다. Type 마-I 공격의 대표적인 예로 분산 서비스 거부 공격을 보기로 한다.

그림 8은 apache 웹 서버의 access 로그 일부분을 나타낸다. 이 그림을 보면 지속적으로 6.asf와 3.asf 멀티미디어 파일을 다운로드 하고자 요청을 하고 있음을 알 수 있다. 여러 대의 호스트가 시간차를 두고 지속적으로 멀티미디어 파일을 요청하고 있다. 서비스를 요청한 각각의 호스트별로 분석을 해 보면 한 호스트가 멀티미디어 파일의 요청을 4~5번 정도로 한 것으로 매우 정상적인 모양을 하고 있다. 그러나 장기적인 시간의 관점에서 보면 갑자기 여러 대의 호스트가 평상시에 없던 멀티미디어 파일 요청을 집중적으로 하고 있음을 알 수 있다. 공격자는 웹 서버에 asf나 mpg와 같은 크기가 큰 멀티미디어 파일이 있음을 알아내고, IDS에 탐지되지 않기 위하여 여러 대의 호스트를 이용하여 각기 다른 시차를 두어 웹 서버를 공격하는 매우 지능적인 공격 방법을 사용하였고 공격의 특성상 자동화된 툴에 의하여 이루어진 것처럼 보인다. 이러한 공격을 막기 위해서는 일정한 시간동안 각 사용자들의 서비스 요청을 분석해야만 이러한 공격이 있는지 판별해 낼 수 있고 의심스러운 서비스 요청과 호스트에 대한 정보를 생성해 낼 수 있을 것이다.

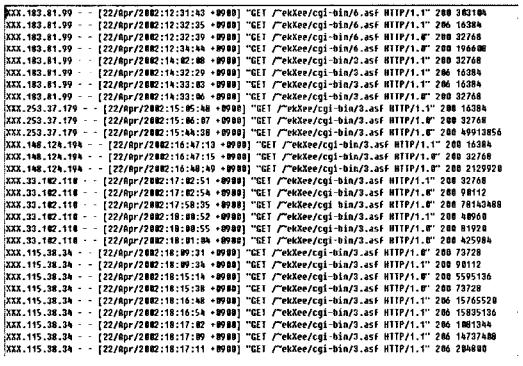


그림 8 Apache 웹 서버의 access 로그

4.1.6 공격 원인과 원인 위치에 따른 분류 정리

4.1장에서는 웹 공격의 발생 원인과 발생 원인의 위치에 따라 웹 공격들이 어떻게 분류되는지 알아보았다. 웹 공격은 공격 발생 원인에 따라 5가지 type의 공격이 나타난다. Type 가, 나 유형의 공격들은 명확한 공격 패턴이 나타나기 때문에 signature에 의해 비정상적인 공격인지 아닌지를 판별해 낼 수 있는 공격들이다. 만약 공격 signature가 항상 최신의 취약점에 대하여 갱신되어있다면, 이러한 공격들은 IDS에 의해 탐지될 수 있다.

다른 인터넷 서비스들과 상이한 특성을 가지고 있는 웹 서비스는 서비스를 제공하기 위한 웹 컨텐츠와 웹 서비스 특징에 따라 다양한 공격 방법이 나타난다. 이런 공격 방법 중에서 웹 서비스에 제공되는 코드나 데이타를 공격자가 임의로 변경하여 웹 서비스를 오용하는 공격을 type 다 유형으로 분류하고, 웹 서비스를 이용하기 위해 사용자가 입력하는 입력 형태를 비정상적으로 입력하여 웹 서비스를 오용하는 공격을 type 라 유형으로 분류하였다. 마지막으로 웹 공격의 특이한 경우로 웹 요청이 정상적인 요청이지만, 이 요청이 웹 서버의 서비스를 마비시킬 정도로 웹 서버에 부하를 가져오게 되는 서비스 거부 공격을 type 마 유형으로 분류하였다.

Type 다, 라, 마의 공격들은 웹 서비스의 내용과 구현 방법에 따라 그 공격방법이 달라지기 때문에 탐지를 위한 signature를 만들기가 어렵다. 그러나 자세히 살펴보면 이러한 공격들은 모두 공격자가 해당 사이트의 약점을 알아내고, 웹 입력에 비정상적인 입력을 넣거나 웹 서비스를 비정상적으로 이용해서 나타나는 공격이다. 따라서 이러한 공격들을 탐지해내기 위해서는 먼저 웹 서비스의 특징을 잘 파악하고, 정상적인 사용이나 비정상적인 사용에 대하여 profile이나 model을 생성해 내고, 이것을 바탕으로 탐지해 낼 수 있다.

공격을 탐지하기 위해서는 해당 공격의 특징을 알아야 하는데, 일반적으로 공격을 탐지하기 위해서 공격의 원인을 아는 것이 중요하다. 다시 말하면 공격의 탐지는 공격의 원인과 밀접한 관계를 가지는데, 다음 4.3장과 4.4장에서는 앞에서 분류한 공격의 원인에 따라 분류한 공격들이 공격을 탐지하기 위한 관점에서 어떤 특징들을 가지는지 알아보기로 한다.

4.2 웹 공격 분류의 활용(CASE Study)

취약점 공개 사이트 중에 하나인 CERT[7]에서는 매년 20~40개 정도의 취약점이 공개된다. 여기에는 OS, 인터넷 서비스, rpc 서비스, 프로토콜, 어플리케이션을 포함한 모든 시스템들의 취약점들이 공개되고 있으며 대부분 심각한 취약점들이 공개되고 있다. 이 장에서는

2000년부터 2002년 6월 현재까지 CERT에서 발표한 취약점들 중에서 웹 서비스에 관련된 공격들을 실제로 공격 원인과 원인의 위치의 관점으로 분류하여 보았다. 2000년부터 2002년 6월까지 CERT에는 75개의 취약점(advisory)이 공개되었는데, 이 중에서 웹 관련된 취약점은 모두 12개이다.

- CA-2002-09:Multiple Vulnerabilities in Microsoft IIS
- CA-2002-08:Multiple Vulnerabilities in Oracle Servers
- CA-2002-05:Multiple Vulnerabilities in PHP fileuploadl
- CA-2001-26:Nimda Worm
- CA-2001-23:Continued Threat of the "Code Red" Worm
- CA-2001-19:"Code Red" Worm Exploiting Buffer Overflow in IIS Indexing Service DLL
- CA-2001-18:Multiple Vulnerabilities in Several Implementations of the Lightweight Directory Access Protocol (LDAP)
- CA-2001-13:Buffer Overflow In IIS Indexing Service DLL
- CA-2001-12:Superfluous Decoding Vulnerability in IIS
- CA-2001-11:sadmind/IIS Worm
- CA-2001-10:Buffer Overflow Vulnerability in Microsoft IIS 5.0
- CA-2000-02:Malicious HTML Tags Embedded in Client Requests

표 8 CERT의 취약점(2000~2002.6)의 원인과 위치에 따른 분류

공격 원인 / 공격 원인 위치	Type 가	Type 나	Type 다	Type 라	Type 마
Type I				CA-2000-02	
Type S	CA-2002-09 CA-2001-26 CA-2001-23 CA-2001-19 CA-2001-18 CA-2001-13 CA-2001-12 CA-2001-11 CA-2001-10				
Type P	CA-2002-08 CA-2002-05				

표 8을 보면 CERT에 공개된 대부분의 취약점들이 type 가에 집중되어있음을 알 수 있다. 즉, 취약점 공개 사이트들에 알려진 취약점들은 주로 구현상의 오류에 집중되어있고, 이를 통해 telnet이나 ftp 등의 단순하고 서비스 내용에 의해 공격방법이 다양하게 나타나지 않는 공격들을 탐지하기에는 유용하다. 그러나 웹 컨텐츠와 웹 서비스의 종류에 따라 공격 방법이 다양하게 나타나는 웹 서비스를 보호하기에는 매우 부적합하다는 것을 알 수 있다.

4.3 공격 탐지 기법에 따른 분류

공격들을 탐지하기 위한 탐지 기법에 따라 각 공격들이 어떻게 나누어지는지 알아본다. IDS를 탐지 기법에 따라 나누어 보면 잘 알려진 공격들에 대하여 signature를 바탕으로 탐지하는 방법과 정상적인 사용을 profile하여 anomalous한 공격을 탐지하는 방법 2가지로 크게 분류할 수 있다. 자세히 살펴보면 전자는 잘 알려진 취약성을 통해 공격에 대한 정보를 가지고 명백한 침입을 탐지하는 방법이고, 후자는 정상적인 시스템 사용에 대하여 프로파일을 만들고 현재의 행위들을 정상적인 프로파일과 비교하여 비정상적인 공격들을 탐지해 내는 방법이다. 이 때 비교 과정에서 기존의 프로파일을 수정하거나 새로운 프로파일을 추가하기도 한다.

표 9는 각각의 공격 type들이 탐지 기법에 따라 어떻게 탐지될 수 있는지를 나타낸다.

표 9 탐지 기법에 따른 공격 분류의 특징

정형화 가능성	정형화 가능한 패턴이 존재	정형화 가능한 패턴이 존재하지 않음	
탐지 기법 / 공격 type	Known signature	Web content independent profile	Web content dependent profile
Type 가	탐지 가능	탐지 불가	탐지 불가
Type 나	알려진 공격 탐지 가능	탐지 가능	탐지 불가
Type 다	탐지 불가	탐지 불가	탐지 가능
Type 라	극히 일부 탐지 가능	탐지 불가	탐지 가능
Type 마	탐지 불가	탐지 불가	탐지 가능

웹 공격을 탐지하기 위한 탐지 기법으로는 크게 3가지로 나누어 볼 수 있다. 하나는 공격 signature에 의해 탐지가 가능한 공격들이다. Type 가의 공격들은 취약점 공개 사이트에 잘 알려져 있고, 공격 패턴도 정형화 가능하기 때문에 signature에 의해 쉽게 탐지 될 수 있다. SecurityFocus[8], NTBugtraq[9], CVE[10] 등의 취약

점 공개 사이트나 웹 서버의 개발사인 Microsoft, Apache 등에 공개되어있고, 또 대부분의 IDS들은 IDS를 판매한 회사로부터 공격 탐지를 위한 최신의 signature를 업데이트 할 수 있도록 되어있다. 따라서 항상 최신 탐지 signature를 유지할 수 있으면 이러한 공격들을 대부분 탐지할 수 있다. Type 나 유형은 HTTP specification을 벗어나 비정상적으로 HTTP 서비스를 요구하여 서비스에 악영향을 미치는 공격들이다. Type 나 공격들은 다른 인터넷 서비스 프로토콜 오류와 같이 정형화 가능한 signature를 통하여 공격을 탐지할 수 있다. Type 라의 공격들 중에 악의적인 HTML 태그를 입력하여 공격하는 경우, 악의적으로 사용될 수 있는 태그들을 조사하여 signature를 만든다면 type 라의 공격 중 아주 일부분의 공격은 탐지할 수 있다.

두 번째로는 웹 컨텐츠에 독립적인 profile을 사용하여 탐지할 수 있는 공격들이다. 즉 오용 탐지를 하기 위한 데이타로 웹 서비스와 상관없이 한번 만들어진 profile을 가지고 어떤 웹 서비스나 상관없이 공격을 탐지해 낼 수 있다는 것을 의미한다. Type 나 유형은 HTTP specification을 벗어나 비정상적으로 HTTP 서비스를 요구하여 서비스에 악영향을 미치는 공격들이다. HTTP의 정상적인 사용을 profile하고 이 profile들과 비교하여 비정상적인 사용을 탐지해 낼 수 있다. 이 profile은 HTTP 프로토콜을 사용하는 모든 서비스에 적용할 수 있기 때문에 한 번 만들어진 profile이면 다른 장소에서 profile할 필요 없이 어디에나 바로 적용가능 하다는 특징이 있다.

세 번째로는 웹 컨텐츠에 종속적인 profile을 사용하여 탐지할 수 있는 공격들이다. 웹 컨텐츠 종속적인 profile은 웹 서비스의 내용과 종류에 따라 탐지를 위한 profile이 달라지는 것을 의미한다. Type 다, type 라, type 마의 공격들은 웹 컨텐츠와 웹 서비스를 제공하는 코드에 따라 그 공격 방법이 다르다는 특징이 있다. 따라서 type 다, type 라, type 마의 공격들은 웹 서비스의 특징을 잘 반영하여 정상적인 서비스 이용들을 profile하고 profile된 정상적인 행위들과 비교해서 비정상적인 공격들을 탐지해 내야 한다. 그러므로 웹 서비스마다 그 profile이 달라질 수밖에 없고, IDS를 운용하기 위해서는 해당 사이트에 종속적인 profile을 만들기 위한 시간이 필요하다.

Type 다의 공격들은 공격자의 변조에 의한 공격들이기 때문에, 만약 침입 탐지 시스템이 웹 서비스 session 중에서 변경되어서는 안될 중요한 정보들을 알고 있다면 웹 요청 결과(result)와 사용자의 웹 요청(query)에서 정보들이 변경되었는지 확인하여서 해당 공격을 탐지할 수 있다. 또 호스트별로 또는 사용자 별로 웹 서비스 사용을 profile하고, 평상시의 웹 서비스 사용 profile과 다른 사용을 탐지하여 공격자에 의한 변조 공격 여부를 판단하는데 도움을 줄 수 있다. Type 라의 공격들은 비정상적인 파라미터 입력에 의한 공격이다. 이 경우는 웹 서비스의 정상적인 입력들을 profile하고 일반적으로 웹 서비스의 입력으로 들어올 수 있는 profile을 가지고 있다면, 이를 현재 들어온 웹 서비스의 입력과 비교하여 비정상적인 입력인 type 라의 공격들을 탐지할 수 있다. 예를 들어 게시판에 악의적인 HTML 태그를 넣어 공격하는 경우, 대부분의 게시판 사용자들은 HTML 태그들을 넣지 않고 사용할 것이고 일부 정상적으로 HTML 태그를 넣어서 사용하는 사람들도 악의적인 태그를 넣고 사용하지는 않을 것이다. 따라서 게시판의 정상적인 입력에 대한 profile을 가지고 있다면, 이를 현재 들어온 게시판의 입력과 비교하여 비정상적인 사용인지 아닌지를 판별할 수 있을 것이다. Type 라의 공격들은 시간별로 또는 호스트별로 웹 서비스의 사용패턴을 profile하고 이를 시간별로 또는 호스트별로 비교하면서 평상시의 웹 서비스 요청과 다른 패턴이 나타나면 이상한 웹 요청이 있음을 탐지할 수 있다.

4.4 공격 탐지 위치에 따른 분류

4.1장에서 알아본 것과 같이 각 공격들의 발생 원인이 다르기 때문에, 해당 공격들을 탐지할 수 있는 위치도 다를 수 있다. 예를 들면 어떤 공격들은 공격의 발생 원인이 되는 웹 어플리케이션 구성요소의 위치와 공격을 탐지할 수 있는 위치가 일치하는 경우도 있겠지만, 어떤 공격들은 공격 발생 원인의 위치와 실제 해당 공격을 탐지할 수 있는 위치가 다른 공격도 있을 수 있다. Unicode translation 공격의 경우에는 공격의 원인이 웹 서버에 있고, 실제 탐지도 웹 서버에서 탐지되어야 매우 효과적이다. 만약 이 공격이 웹 서버의 위치에서 공격을 탐지하지 않고, 다른 위치에서 탐지를 한다면 IDS는 웹 서버의 종류를 모르기 때문에 모든 종류의 웹 서버에 해당하는 탐지 signature를 가지고 있어야 한다. 뿐만 아니라 실제로 웹 서버에 대한 공격이 성공하여 웹 서버에게 영향을 주었는지, 공격이 실패하였는지를 판단하기 위해서는 IDS가 웹 서버와 상호 작용을 하여야 할 필요가 있다. 따라서 웹 서버에 대한 공격의 경우는 웹 서버의 위치에서 탐지되어야 효과적일 수 있다. 또 4.1.3 장에서 알아본 소스코드 변조 공격 같은 경우에는 HTML 코드를 변조하였으므로 공격의 원인이 웹 interface에 있지만, 실제 이 공격을 탐지하려면 실제 해

당 코드가 적용되는 웹 서버나 script preprocessor에서 공격을 탐지하여야 한다. 이 장에서는 실제 공격을 탐지하기 위해 IDS를 설치하고자 할 때, 어떤 위치에 설치되어야 해당 공격들을 탐지해 낼 수 있는지 알아보기 위해 각 공격들을 탐지 위치의 관점에서 분류하여 보자.

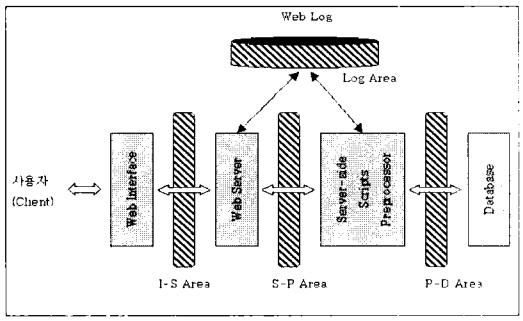


그림 9 공격을 탐지할 수 있는 IDS의 위치

그림 9는 각 웹 어플리케이션 구성요소의 위치에 따라서 공격을 탐지할 수 있는 위치들을 나타내고 있다. I-S 영역 영역은 웹 서버의 앞단에서 웹 서버에 들어오는 URI들을 감시하고, S-P 영역 영역에서는 웹 서버에서 server side scripts preprocessor로 들어가는 입력들을 감시한다. P-D 영역 영역에서는 database로 들어가는 입력을 감시하고 log 영역은 웹 서버가 모든 요청을 처리하고 난 뒤에 남기는 log들을 감시한다.

탐지 위치의 특성상 I-S 영역, S P 영역, P D 영역에서는 공격이 일어나기 전에 공격을 탐지하고, log 영역에서는 공격이 일어난 뒤에 웹 서버가 남기는 log를 가지고 탐지하기 때문에 공격이 이미 일어난 뒤에 공격을 탐지할 수 있다.

표 10은 공격 탐지의 위치와 공격 탐지 기법에 따라 각각의 공격들이 어떻게 나타나는 지를 보여준다. 이 표의 내용은 4.1장에서 알아본 공격들이 공격을 탐지하기 위한 영역의 어느 부위에서 어떤 탐지 기법을 사용하여 탐지될 수 있다는 것을 나타내고 있다.

예를 들면, Type 가-S 공격들은 공격의 원인이 웹 서버의 구현 오류이므로 I S의 영역에서 탐지 될 수 있을 것이다. 그리고 앞의 4.3장에서 본 것과 같이 signature에 의해 공격을 탐지할 수 있다. 따라서 요약하면 type 가-S의 공격들은 잘 알려진 탐지 signature에 의해 I S 영역에서 탐지될 수 있다는 것을 의미한다. 또 type 가 S의 가장 대표적인 예인 Unicode translation 오류 공격을 보면 웹 log에도 공격의 흔적이 남기 때문에 웹 log를 분석해서도 비록 공격이 일어난 후에 탐지하게 되지만, 공격을 탐지할 수 있다. 마찬가지로 type 가-P 공격들도 S P 영역에서 signature에 의해 탐지할 수 있다.

표 10 탐지 위치와 탐지 기법에 따른 공격 분류의 특징

탐지 위치 / 공격 type	공격이 일어나기 전 탐지			공격이 일어난 후 탐지
	I S	S P	P D	Log
Type 가 S	Signature			Signature
Type 가 P		Signature		Signature
Type 나 I	Signature Profile			Signature Profile
Type 나 S	Signature Profile			Signature Profile
Type 나 I	Profile	Profile		
Type 라 I	Signature (일부) Profile	Signature (일부) Profile		Signature (일부) Profile
Type 마 I	Profile		Profile	Profile

Type나 공격들은 공격의 원인이 HTTP specification 오류이므로 HTTP 프로토콜을 사용하는 웹 서버의 앞 영역인 I S 영역에서 탐지 될 수 있을 것이다. 그리고 4.3장의 탐지 기법에 따른 웹 공격의 특성에서 알아본 바와 같이 아직 공격이 발견되지는 않았지만, 발견되었다면 정형화된 공격 패턴인 signature에 의해서도 일부 탐지될 수 있다. 또 웹 컨텐츠에 독립적인 HTTP 프로토콜 사용에 관한 정상적인 profile을 가지고 비정상적인 HTTP 프로토콜 사용을 탐지할 수 있다.

Type 다 공격들은 원인이 공격자에 의한 코드나 데이타의 변조에 있기 때문에 해당 코드나 데이타를 사용하는 위치에서 탐지할 수 있을 것이다. 또 클라이언트 영역에서 사용하는 client side script 코드를 변조하는 경우에는 변조된 내용이 웹 서버에게 전해지고 실제로 변조된 내용에 의해 연산이 수행되는 위치에서 해당 공격을 탐지할 수 있을 것이다. 대부분의 공격들은 I S 영역에서 profile에 의해 공격자에 의해 변조된 공격 여부를 탐지할 수 있다.

Type 라의 공격들은 공격의 원인이 비정상적인 파라미터 입력이기 때문에 마찬가지로 해당 입력을 사용하는 위치에서 탐지할 수 있을 것이다. 웹 서버에서 사용하는 URI의 파라미터 입력을 변경하는 경우는 웹 서버 영역인 I S 영역에서 탐지할 수 있고 게시판이나 input

text form의 경우는 대부분 server side script preprocessor가 처리하기 때문에 S-P 영역에서 탐지할 수 있다. 이 공격들은 4.3장의 탐지 기법에 따른 웹 공격의 특성에서 알아본 것과 같이 극히 일부분은 signature에 의해 탐지될 수 있지만, 대부분의 공격들은 정상적인 웹 서비스 입력 profile과 비교하여 비정상적인 웹 서비스 입력을 탐지해 낼 수 있다.

Type 마의 공격들은 웹 서비스를 마비시킬 정도로 웹 서비스 요청을 무리하게 요구하는 공격들이다. 이 공격들은 웹 서비스 요청 사용 관점에서 보면 평상시 웹 서비스 요청과는 다른 사용 패턴을 나타낸다. 특정 시간 동안 웹 서비스 요청이 급증한다던가 특정 서비스 요구에 집중되는 경향이 나타난다. 따라서 정상적인 웹 서비스 요청에 대한 profile을 만들어 놓고, 이를 웹 서비스 요청과 비교하면서 갑작스럽게 웹 서비스 요청이 증가하거나 특정 서비스 요구에 집중되는 현상을 통해 type 마 공격을 탐지할 수 있다. 다음에는 각각의 공격을 탐지할 수 있는 위치들에 대해서 알아보도록 한다.

4.4.1 I-S 영역에서 탐지할 수 있는 공격

I-S 영역에서 공격을 감시하게 될 경우, 공격 탐지 소스 데이타가 어떤 것이냐에 따라서, 실제 웹 서버가 있는 호스트와는 다른 호스트에서 network을 감시하게 되어서 웹 서버로 들어가는 network packet을 감시(network based)하는 방법이 있고, 웹 서버 호스트에 설치되어서 웹 서버에 들어오는 입력을 직접 감시(host based)하는 방법이 있다. 전자의 network packet으로 데이타를 감시하게 될 경우 웹 서버의 CPU resource를 사용하지 않기 때문에 웹 서버 호스트의 연산에 영향을 전혀 주지 않으며, 웹 서버가 여러 대 설치되어있는 경우 그 앞 단에 설치되어 한 대가 여러 대의 웹 서버를 감시할 수 있다는 장점이 있다. 그러나 단점으로는 복잡한 session 단위의 공격은 탐지하기가 어렵고, 복잡한 분석을 위해서는 엄청난 양의 데이타 교환이 필요로 할 수도 있다. 반면에 후자의 웹 서버에 직접 설치되어 웹 서버로 들어오는 입력을 직접 감시하게 되는 경우에는 실제 패킷이 fragmentation이 되었는지 등과는 관계없이 실제 웹 서버로 들어가는 입력을 직접 감시할 수 있기 때문에 공격자가 IDS를 우회하여 공격하기가 힘들어진다는 장점이 있지만 웹 서버의 호스트에 직접 설치하기 때문에 웹 서버의 성능이 저하되고, 웹 서버가 설치된 호스트의 개수만큼 설치되어야 한다는 단점이 있다. 최근에는 이러한 장단을 보완해서 매우 용량이 작은 sensor들만 호스트에 직접 설치되고, 침입 분석과 알람 및 복구 기능은 네트웍을 이용하여 다른 manager 모듈에서 수행하는 복합적인 방법이 많이 사용되고있다.

I-S 영역에서는 탐지 signature를 가지고 이미 잘 알려진 type 가-S 공격들과 일부 type 나 공격들을 탐지할 수 있다. 그리고 I-S 영역에서는 정상적인 HTTP 사용들을 가지고 만들어진 profile을 가지고 비정상적인 HTTP 프로토콜 사용인 type 나 공격들 대부분을 탐지할 수 있다. 또 type 다, type 라의 공격들 중에서 웹 서버에 바로 영향을 미치는 공격들과 type 마와 같이 웹 서버에 비정상적으로 과도한 throughput을 요구하는 공격들은 웹 컨텐츠나 코드에 종속적인 profile을 만들어서 정상적인 사용에 대한 profile과 비교하면서 비정상적인 공격들을 탐지해 낼 수 있다.

예를 들면 그림 7에서 볼 수 있는 type 라의 비정상적인 URI를 이용한 공격의 경우와 같이 평상시에는 발생하지 않는 URI(http://www.myweb.com/shop.php?no=1&price=10)가 갑자기 발생하는 경우나 그림 8에서 볼 수 있는 type 마와 같이 과도하게 웹 서버의 through put을 증가시키는 공격들은 평소의 정상적인 URI의 profile과는 매우 다르기 때문에 IDS는 이 profile과 비교하여 비정상적인 공격이라는 것을 탐지해 낼 수 있다.

4.4.2 S-P 영역에서 탐지할 수 있는 공격

S-P 영역은 웹 서버와 server side scripts preprocessor 사이에 설치되어서 웹 서버로부터 preprocessor로 들어가는 입력을 감시하여 공격을 탐지한다.

이 영역에서는 탐지 signature를 가지고 이미 잘 알려진 공격인 type 가-P 공격들을 탐지할 수 있다. 또 type 다, type 라의 공격들은 server side scripts pre processor의 입력들로 정상적인 사용들의 profile을 만들고 이 profile과 비교하여 비정상적인 공격들을 탐지할 수 있다.

4.4.3 P-D 영역에서 탐지할 수 있는 공격

P-D 영역은 server side script preprocessor와 데이타베이스나 데이타를 접근하는 시스템 사이에 설치되어서 데이타의 사용량 등을 감시하여 공격을 탐지한다.

Type 마의 공격의 일부인 정상적이면서 동시에 여러 클라이언트가 동시에 큰 멀티미디어 파일을 요구함으로써 발생할 수 있는 서비스 마비 공격들은 시간별 데이타 사용량을 분석함으로써 탐지할 수 있다. 특정 시간동안 특정 데이타의 요청이 갑자기 증가하거나 일부 몇 가지 데이타에 서비스 요청이 집중되면서 웹 서비스가 마비되면 type 마의 공격이 발생한 것을 탐지할 수 있다.

그 외에도 P-D 영역에서는 일반적으로 공격을 탐지하기 보다는 다른 곳에서 공격을 탐지하는데 도움을 줄 수 있는 정보를 제공할 수 있다. 가장 자주 사용되는 데

이타에 대한 정보나 특정한 시간대에 따라 달라지는 데이타 사용량 등의 정보들을 제공함으로써 다른 영역에서 웹에 대한 공격을 탐지하는데 도움을 줄 수 있는 정보를 제공할 수 있다.

4.4.4 Log 영역에서 탐지할 수 있는 공격

Log 영역은 웹 서버가 사용자의 요청에 따라 웹 서비스를 제공하고 난 후에 웹 서비스 제공 결과를 기록하는 영역이다. Log를 통해 사용자의 어떤 웹 서비스 요청이 실패하였는지, 성공하였는지 여부와 해당 서비스 요청이 일어난 시각 등을 알 수 있다. Log를 통해 웹 공격을 탐지하는 것은 웹 공격 사후의 탐지이기 때문에 이미 시스템은 피해를 입었을 수도 있다. 따라서 log 영역에서는 공격 탐지의 목적보다는 웹 서버의 특성을 파악하고 앞 단계에서 공격을 탐지하는데 도움을 줄 수 있는 유용한 정보를 제공하는데 사용될 수 있다. 예를 들면 특정한 요일별로 웹 서비스의 사용패턴을 수집하거나, 가장 요구량이 많은 웹 서비스의 컨텐츠를 분석하는 등의 도움을 줄 수 있다.

4.5 웹 공격 결과에 따른 분류와 위험성 분석

웹 IDS를 설계하기 위해서는 해당 웹 사이트의 특성을 반영하여야 한다는 것을 앞에서 알아보았다. 실제 각각의 웹 사이트들의 서비스 목적이 다르기 때문에 동일한 공격에 대해서도 공격의 위험도는 달라질 수 있다. 예를 들어 소스 코드를 변조하여 게시판을 삭제하거나 수정하는 공격의 경우, 아마존과 같은 쇼핑몰 사이트보다는 정보 제공을 목적으로 하는 CNN이나 Time의 경우에는 제공하고자 하는 정보를 삭제하거나 심각하게는 정보의 변조를 통하여 잘못된 정보를 제공하게 할 수 있으므로 위험도가 높을 것이다. 그러나 쿠키 변조와 같이 사용자의 정보를 위조하는 공격의 경우에는 후자보다는 당연히 아마존과 같은 쇼핑몰 사이트의 경우 위험도가 높은 공격이 될 것이다. 따라서 웹 IDS를 설계할 때 해당 웹 사이트의 공격 위험도를 분석하여 반영하면 공격에 대한 우선순위를 정할 수 있기 때문에, false alarm을 낮출 수 있고 IDS의 효과는 더욱 높아질 것이다.

표 11은 공격에 의해 나타나는 결과를 나타내고 있다. 공격 결과들은 공격의 원인과 밀접한 관계가 없는 경우가 대부분이므로, 앞에서 알아보았던 공격 원인의 분류를 이용하여 공격 결과들을 알아볼 수가 없다. 여기서는 공격 결과들을 우선 공격이 시스템의 내부 상태를 변화시킬 수 있는지의 여부와 시스템의 침입(penetration)[6]

표 11 공격 결과 유형

시스템 침입 여부 / 시스템 상태 변화의 여부	시스템을 침입하는 공격 (Penetration)	시스템을 침입하지 않는 공격 (No penetration)
공격에 의하여 시스템 내부의 상태를 변화시킬 수 있는 공격 (Change)	<P C 유형>	<NP C 유형>
공격에 의하여 시스템 내부 상태를 변화시킬 수 없는 공격 (No change)	<P-NC 유형>	<NP NC 유형>

여부에 따라 나누어 보았다. 시스템의 내부 상태를 변화시킨다는 것은 공격을 통하여 내부의 데이타를 변경한다거나, 시스템의 security policy를 변화시키는 등의 내부적인 시스템 상태를 변경하는 것을 말한다. 그리고 시스템 침입은 시스템에 임의의 명령어를 수행할 수 있는 권한을 얻는 것을 의미한다. 대부분 시스템을 침입하여 장악(compromise)하는 공격들은 비록 공격 그 자체가 시스템을 변경시키지 않는다고 하여도 공격자는 시스템에 불법적인 권한을 획득하여 시스템에 대하여 임의의 명령어를 수행할 수 있기 때문에 시스템의 상태를 변화시킬 수 있는 공격의 범주에 들어간다. 그러나 서비스 거부 공격 같은 경우에는 단지 과도한 throughput이나 어플리케이션의 취약점을 이용하여 시스템이 잠시 서비스를 할 수 없도록 마비시키는 공격이므로 시스템 내부의 상태를 변화시키지는 못한다.

P-NC 유형은 공격자에 의해 시스템이 장악되었지만, 공격자에 의해 시스템의 내부 상태가 변하지 않는 공격들을 의미한다. 그러나 실제 공격자가 해당 호스트를 장악하고 시스템의 상태를 변화시키지 않는 경우는 없기 때문에 해당 유형에 속하는 공격들은 존재하지 않는다.

P-C 유형은 공격자에 의해 시스템이 장악되고, 공격자가 시스템의 상태를 변화시킬 수 있는 공격들이다. 이런 공격들은 그 공격효과가 영구적이고, 관리자가 공격을 탐지하여 공격의 흔적들을 복구하기 전까지는 공격 전의 상태로 돌아가기가 힘들다. 대부분의 공격들이 type 가의 어플리케이션 구현 오류에 의한 공격들에 속하며 대표적인 예를 들어보면 Unicode character translation 오류를 이용한 공격이나, 버퍼의 크기를 체크하지 않아서 발생하는 buffer overflow 공격들이 가장 대표적인 예이다.

NP-C 유형은 공격자에 의해 시스템이 장악되지는 않

6) Penetration : 공격자가 시스템 내부에 침입하여 임의의 명령어를 수행할 수 있는 단계를 의미한다.

지만, 공격자가 시스템 내부의 상태를 변경시키는 공격들이다. 이런 공격들은 그 공격효과가 공격당시에 순간적으로 나타나거나, 또는 특별한 경우에만 그 공격 효과가 나타나는 공격들이다. 대부분의 공격들이 type 다, type 라에 속하는 공격들이며 대표적인 공격들의 예를 들어보면, 쿠키 변조 공격, 소스 코드 변조 공격, 비정상적인 URI를 이용한 공격, 게시판이나 text form을 이용한 공격 등을 예로 들 수 있다.

NP-NC 유형은 공격자에 의해 시스템이 장악되지 않으면서, 공격자에 의해서도 시스템 내부의 상태가 변경될 수 없는 공격이다. 이런 공격들은 일시적으로 시스템을 마비시키는 공격이나, 정상적인 서비스를 불가하도록 만드는 공격들을 의미한다. 또한 공격이 발생하는 동안 일시적으로 마비되거나 공격이 끝난 후에는 다시 정상적인 공격전 상태로 돌아가는 공격들이 대부분이다. 그러나 몇몇 일부의 공격들은 시스템이 완전히 다운(frozen)되어 버려서 재 부팅이 필요한 공격들도 있다. 대부분이 type 나, type 마에 속하는 공격들이며, 서비스 마비 공격들이다. 예를 들면 어플리케이션 구현 오류로 인한 서비스 거부 공격이나 HTTP의 비정상적인 사용에 의한 서비스 거부 공격, 분산 서비스 거부 공격(DDOS), 어플리케이션 오류로 인한 access 권한이 없는 directory나 file을 볼 수 있는 공격 등이 이 공격 유형에 속한다.

표 12는 4.1장에서 분류한 공격 원인에 따른 분류의 공격들이 공격 결과에 따라 분류한 공격 유형들의 어떤 영역에 속하는 지를 보여준다.

Type 가 유형의 공격들은 시스템 구현상의 취약점을 공략하는 공격이다. 취약점 공개 사이트들에 발표된 취약점들을 보면 대부분의 취약점들이 시스템을 침입할 수 있는 위험한 취약점들이 대부분이다. Type 나의 공격들을 다른 인터넷 서비스와 비교해 볼 경우 대부분의 공격들이 서비스 거부 공격이고, specification 상에 명시되지 않은 오류들을 시스템에 넣게 되면 시스템들은 이를 어떻게 처리하여야 할 지 모르고 시스템이 마비되는 경우가 대부분이기 때문에 NP-NC 유형에 속한다고 볼 수 있다. Type 다, 라 유형의 공격들은 대부분 시스템을 침입하지 못하는 공격들이다. Type 마 유형의 공격들은 DOS나 DDOS 공격들이므로 NP-NC 유형에 속한다.

여기서 웹 공격의 위험성을 분석하기에는 웹 컨텐츠 정보의 값어치(중요도)를 고려할 수 없으므로 분석 대상에서 제외하고 간단하게 웹 서버를 관리하는 관리의 입장에서 웹 공격들의 위험도가 어떻게 될지 알아보도록 한다. 표 11의 공격 결과 유형에서 알아본 것처럼 웹 서버를 관리하는 입장에서는 시스템을 침입하여 부당한 권한을 획득하는 공격이 가장 위험한 공격이 될 것이다. 왜냐하면 시스템의 내부에 접근할 수 있는 권한을 획득하였다면 일반적으로 웹 서비스를 훼손시키는 것 외에 전체 시스템의 훼손도 가능하기 때문에 가장 위험하다고 할 수 있다. 또 내부의 시스템 상태를 변화시키는 공격의 경우에는 관리자가 시스템 상태 변화를 체크하고 공격을 당한 후에는 공격전의 상태로 복원시켜 주어야 하므로 시스템 상태를 변화시키지 않는 공격보다 위험하다고 할 수 있다. 마지막으로 시스템을 침입하지 않고, 시스템 내부의 상태도 변화시킬 수 없는 공격들은 대부분 서비스 거부 공격으로 공격이 끝나고 난 뒤에 원래 상태로 돌아가는 공격들이 대부분이다. 따라서 공격 결과들을 위험한 순서대로 나열해 보면, 가장 위험한 공격들이 P-C 유형이고 다음으로 NP-C 유형, NP-NC 유형의 공격들이 위험하다.

이와 같은 정보를 바탕으로 표 12에서 조사한 것과 같이 공격 결과에 따라 각 type들의 공격 위험도를 분석하여 보면, type 가와 같이 시스템 침입과 내부 상태의 변경이 모두 가능한 type 가 유형의 공격들이 가장

표 12 공격 결과에 따른 공격 분류의 특징

공격 결과 유형 / 공격 type	P-C 유형	NP-C 유형	NP-NC 유형
Type 가	**Type 가 공격들의 대부분이 P-C 유형에 속한다.**	Type 가 공격들의 일부가 NP-C 유형에 속한다.	Type 가 공격들의 일부가 NP-NC 유형에 속한다.
Type 나	존재하지 않을 확률이 높다.	존재하지 않을 확률이 높다.	**Type 나 공격들의 대부분이 NP-NC 유형에 속한다.**
Type 다	존재하지 않을 확률이 높다.	**Type 다 공격들의 대부분이 NP-C 유형에 속한다.**	Type 다 공격들의 일부가 NP-NC 유형에 속한다.
Type 라	존재하지 않을 확률이 높다.	**Type 라 공격들의 대부분이 NP-C 유형에 속한다.**	Type 라 공격들의 일부가 NP-NC 유형에 속한다.
Type 마	존재하지 않는다.	존재하지 않는다.	**Type 마 공격들의 대부분이 NP-NC 유형에 속한다.**

위험하다. 그리고 NP-C와 NP-NC 유형의 공격들이 있는 type 다와 type 라의 공격이 다음으로 위험하다. Type 나와 type 마의 공격들은 대부분 NP-NC 유형에 들어가게 되지만, type 나의 공격들은 아직 발견된 공격들이 없고 P-C나 NP-C의 유형에 들어가게 될 공격들이 존재하게 될지도 모르기 때문에 type 마 공격보다 위험하다.

표 13 공격의 위험도 분석

공격의 위험도가 높은 순서	비 고
Type 가	시스템 침입이 가능한 공격들이 대부분이다.
Type 다, Type 라	시스템 내부의 상태를 변경시키는 공격들이 대부분이다.
Type 나	서비스 거부 공격이 대부분이다. 공격이 발생한 적은 없지만 다른 영역의 공격이 가능할지도 모른다.
Type 마	서비스 거부 공격이 대부분이다.

웹 서비스를 보호하고자 하는 웹 사이트의 CEO나 관리자는 각각의 웹 사이트들에 대한 공격의 위험도를 표 12에서 보여준 공격 유형에 따라, 공격의 심각성을 상, 중, 하 또는 필요에 따라 세부적으로 나누고 점수를 부여할 수 있을 것이다. 그 뒤에 각각의 해당하는 공격 type들에 우선 순위를 부여하고 이를 통하여 웹 IDS가 공격을 효과적으로 탐지할 수 있도록 웹 IDS의 탐지 규칙에 각 공격의 위험도를 반영할 수 있을 것이다.

이 부분에 대한 연구는 실제로 기업이나 국가 기관 등에서 웹 공격에 대하여 피해를 입은 사례를 조사하고, 각각의 기업이나 기관의 웹 사이트 특징에 따라 웹 공격들의 위험도가 어떻게 달라지는지를 연구하여 위험도를 조사할 수 있는 체크리스트나 지표를 만들어야 할 것이다. 따라서 이 부분에 대한 연구는 향후 연구 계획으로 남겨두고 향후에 웹 사이트들의 특징에 따라 공격의 위험도가 어떻게 변하는지를 조사하고 분석하는 과정이 선행되어야 할 것이다.

5. 결론 및 향후 연구 계획

웹 서비스의 특징은 기본적으로 모든 사용자에게 개방되어있는 서비스이기 때문에, 다른 인터넷 서비스들에 비해 접근 관리를 통해 서비스를 보호하기 어렵고, 공격자가 공격을 하기 위한 취약점을 발견하기 용이하다는 특징이 있다. 게다가 관리자의 입장에서 보면 웹 서비스에 접근하는 사용자들이 임의적이고 다양하기 때문에, 비정상적인 사용자들을 감시하기가 매우 어렵다.

반면에 기존의 범용 침입 탐지 시스템들은 웹 공격을 탐지하기 위한 IDS로 적합하지 못하다. 왜냐하면 우선 웹 공격들은 효과적으로 탐지할 수 있는 signature를 만들기 어렵기 때문에 웹 공격에 대한 signature들을 많이 가지고 있지 못하다. 둘째로 실시간으로 침입을 탐지해야 하는 범용 침입 탐지 시스템들은 인터넷 서비스의 종류나 웹 서버의 종류에 관계없이 모든 인터넷 서비스들을 위한 탐지 signature를 가지고 있기 때문에, 웹 서비스만을 제공하는 서버에 false alarm을 낼 수 있고 오히려 다른 인터넷 서비스들을 위한 탐지 signature들에 의해 웹 공격을 탐지하지 못 할 수도 있다. 셋째로는 공격자가 HTML 코드 또는 쿠키를 수정하거나, 웹 서비스에 악의적인 입력을 넣어서 웹 서비스를 비정상적으로 이용하는 등의 웹 컨텐츠의 특징을 이용한 공격의 경우는 특정한 공격 패턴이 존재하지 않고, 공격의 특징도 다양하기 때문에 탐지 signature를 만들기 쉽지 않기 때문이다. 따라서 웹 서비스를 보호하기 위해서는 웹 서비스의 특성을 반영한 웹 서비스 특화된 침입 탐지 시스템이 필요하다.

우리가 4장을 통해 알아본 것과 같이, 공격을 탐지하기 위한 특성이 공격의 원인과 밀접한 관계가 있고 웹 서비스의 공격들은 공격 발생 원인과 공격 발생 원인의 위치에 따라 그 특성이 잘 나타나기 때문에 웹 공격들을 이해하는데 매우 용이하게 사용될 수 있다. 가장 먼저 4.1장에서는 공격 원인과 공격 원인 위치에 따른 웹 공격 분류를 통해 각각의 공격 type 유형에 해당하는 공격들을 알아보았다. Type 가, type 나 유형에 있는 공격들은 취약점 공개 사이트들에 해당 취약점들이 잘 알려져 있고, 공격을 탐지하기 위한 탐지 signature를 만들 수 있기 때문에 이를 통하여 해당 공격들을 탐지할 수 있다. 그러나 type 다, type 라, type 마의 공격들은 웹 컨텐츠와 웹 서비스의 특징에 따라 존재할 수 있는 공격 방법이 다양하고 공격을 탐지할 수 있는 signature를 만들기 어렵기 때문에 profile을 위한 오용 탐지 기법을 이용하여 탐지할 수 있다.

두 번째로 웹 공격을 탐지하기 위한 침입 탐지 시스템 설계하기 위하여 앞에서 분류한 웹 공격들의 특징을 다른 관점에서 분석해 보았다. 즉 4.3장과 4.4장에서 웹 공격을 탐지하기 위한 탐지 기법과 공격을 탐지 할 수 있는 웹 어플리케이션 구성요소의 위치에 따라 각 공격들이 어떤 특성을 가지는지 알아보았다. 이것을 바탕으로 실제 4.1장에서 알아보았던 각각 type의 공격들이 웹

어플리케이션 구성요소의 어떤 위치에서 어떤 탐지 기법으로 탐지될 수 있는지를 명확하게 알 수 있게 되었고, 이를 통해 웹 서비스를 위한 침입 탐지 시스템을 설계할 수 있을 것이다.

마지막으로 4.5장에서 공격 결과에 따라 웹 공격들이 어떻게 분류되는지 알아보았다. 이 장을 통해서는 웹 IDS의 false alarm을 낮추고 웹 사이트의 공격을 더욱 효과적으로 탐지하기 위하여 공격의 위험도를 분석하고 이를 웹 IDS에 반영할 수 있을 것이다.

이 논문을 통하여 우리는 웹 서비스를 보호하기 위한 웹 어플리케이션 특화 된 침입 탐지 시스템을 설계하기 위하여 웹 공격들의 특징을 알아 볼 수 있었다. 웹 서비스는 다른 인터넷 서비스들과 상이한 특징을 가지고 있기 때문에, 기존의 범용 침입 탐지 시스템으로는 웹 서비스를 보호할 수 없다. 웹 공격 원인에 따른 분류를 통하여 공격 탐지 기법과 공격 탐지 위치의 특성을 알아 볼 수 있었고, 이를 통하여 웹 서비스 특화 된 침입 탐지 시스템을 설계하는 연구를 위한 교두보를 마련할 수 있었다.

향후 연구 과제로 4.3장과 4.4장에서 알아본 탐지 관점에서의 분류를 가지고 실제 웹 서비스를 위한 침입 탐지 시스템을 만들기 위하여 각각의 I-S, S-P, P-D, Log 영역에 들어갈 침입 탐지 시스템의 모듈을 signature와 웹 컨텐츠 independent, dependent profile을 가지고 탐지할 수 있도록 설계하는 연구를 진행해야 할 것이다.

또 모든 인터넷 서비스들에 대한 침입 탐지 시스템의 false alarm을 줄이려면 각각의 인터넷 서비스들에 대해 특화 된 침입 탐지 시스템을 만들어야 한다. 그렇기 위해서는 이 연구와 같은 방법으로 다른 인터넷 서비스들에 대해서도 같은 방법으로 각 서비스들의 공격 특징을 찾기 위해 해당 공격들을 분류해보고, 각각의 서비스에 특화 된 침입 탐지 시스템을 설계하기 위한 노력을 해야 할 것이다.

참 고 문 헌

[1] Hobbes' Internet Timeline, http://www.zakon.org

[2] S. Pettit, "Anatomy of a Web Application : Security Considerations," Sanctum Inc. July, 2001.

[3] C. L. Liu, Elements of Discrete Mathematics 2nd Edition, pp.113, McGRAW- HILL International Editions.

[4] M. Almgren, H. Debar, and M. Dacier, "A Lightweight Tool for Detecting Web Server Attacks," *Proceedings of NDSS 2000*, pp.157-170, Feb. 2000.

[5] Snort-The Open Source Network IDS, http://www. snort.org

[6] Aleph One, "Smashing The Stack For Fun And Profit," *BugTraq report*, Nov. 1996.

[7] CERT/CC-Computer Emergency Response Team Coordination Center(Reporting Center for Internet Security Problem) http://www.cert.org

[8] SecurityFocus, http://www.securityfocus.com

[9] NTBugtraq, http://www.ntbugtraq.com

[10] Common Vulnerabilities and Exposures, http://cve.mitre.org

서 정 석

2001년 2월 인하대학교 전자계산공학과 학사. 2002년 8월 한국과학기술원 전산학과 석사. 2002년 9월~현재 한국과학기술원 전산학과 박사과정

김 한 성

1990년 3월 육군사관학교 전산학과 졸업
1995년 9월 웨스턴 온타리오대학 전산학 석사. 2001년 3월~현재 한국과학기술원 전산학과 박사과정

조 상 현

1997년 2월 고려대학교 컴퓨터학과 졸업
1999년 2월 한국과학기술원 전산학과 석사. 1999년 3월~현재 한국과학기술원 전산학과 박사과정

차 성 덕

1983년 UC Irvine 전산학과 학사 졸업
1986년 UC Irvine 전산학과 석사 졸업
1991년 UC Irvine 전산학과 박사 졸업
1994년~2001년 한국과학기술원 조교수
2001년~현재 한국과학기술원 부교수