http://dx.doi.org/10.3745/KIPSTC.2012.19C.3.185

실시간 행위 분석을 이용한 악성코드 유포 웹페이지 탐지 시스템에 대한 연구

공 익 선[†] · 조 재 익[††] · 손 태 식[†††] · 문 종 섭[††††]

요 약

최근 웹페이지를 통해 악성코드를 유포하는 공격 방법이 이용되면서, 인터넷을 이용하는 사용자들이 웹페이지에 접속하는 것만으로 악성코드에 감염되는 위험에 노출되어 있다. 특히 웹페이지를 통한 악성코드 유포 방법은 사용자가 인지하지 못하는 사이 악성코드를 다운로드하고 실행하게 된다. 본 논문에서는 기존의 분석서버를 이용한 탐지 방법의 한계점을 보완하기 위해, 사용자 영역에서의 실시간 행위 분석을 방법을 사용하여 정상적인 실행 흐름을 벗어난 비정상 다운로드 파일의 실행을 탐지하고 차단하는 시스템을 제안한다.

키워드 : 악성코드, 행위기반, Drive-by Download

A Study on the Malicious Web Page Detection Systems using Real-Time Behavior Analysis

Icksun Kong[†] · Jaeik Cho[††] · Taeshik Son[†††] · Jongsub Moon[††††]

ABSTRACT

The recent trends in malwares show the most widely used for the distribution of malwares that the targeted computer is infected while the user is accessing to the website, without being aware of the fact that, in which the harmful codes are concealed. In this thesis, we propose a new malicious web page detection system based on a real time analysis of normal/abnormal behaviors in client-side. By means of this new approach, it is not only the limitation of conventional methods can be overcome, but also the risk of infection from malwares is mitigated.

Keywords : Malware, Behavior Analysis, Drive-by Download

1. 서 론

최근 악성코드를 유포하기 위해 웹페이지를 이용하는 방법이 증가하고 있다. 공격자들은 Drive-by download 기법을 이용하여, 사용자가 웹사이트에 접속하게 되면, 악성코드를 다운로드 하고 실행 되도록 한다. 사용자는 악성코드의 다운로드와 실행을 인지하지 못하고, 브라우저를 통해 실행되기 때문에 높은 권한을 갖는다. 또한 웹페이지를 통한 공격은 방화벽이나 침입탐지 시스템을 우회하기 쉽다. 기존의 탐지 방법은 분석 서버에서 사전에 웹페이지를 수집하고, 정적/동적 분석을 통해 Black List URL을 생성한 후 사용자가 웹페이지에 접속할 때 DB조회를 통해 탐지하기 때문에, 수집 주기 사이에 발생한 악성코드 유포 웹페이지를 탐지하기 어렵고, ARP Spoofing을 통한 중간단계의 웹페이지 변조를 탐지하지 못하는 한계점을 갖고 있다. 본 논문에서는 사용자 영역에서 실시간 행위 분석을 통해, 정상 행위와 비정상 행위를 분류하고, 정상 행위를 벗어난 다운로드 파일의 실행을 차단하는 방법으로, 사용자가 인지하지 못하는 형태의 공격에 대하여 악성코드 유포 웹페이지를 탐지하고, 악성코드의 실행을 차단하는 시스템을 제안한다.

† 정 회 원 : 고려대학교 정보보호학과 석사과정
†† 정 회 원 : 고려대학교 정보보호학과 박사과정
††† 정 회 원 : 아주대학교 정보컴퓨터공학부 부교수
†††† 정 회 원 : 고려대학교 정보보호대학원 교수(교신저자)

논문접수 : 2012년 3월 28일
심사완료 : 2012년 4월 24일

2. 관련 연구

본 장에서는 웹페이지를 통한 악성코드 유포에 사용되는 방법과 기존의 탐지 방법에 대해 설명 한다.

2.1 악성코드 소스 은닉

분석가나 탐지 시스템에 의한 분석을 회피하기 위해 공격

자는 다양한 방법으로 은닉을 시도한다. <LINK> <IFRAME> <OBJECT>등의 태그의 속성을 숨기거나, 크기를 0으로 설정하여 브라우저 화면상에 보이지 않게 하거나, 스크립트 코드에서 사용하는 문자열을 인코딩하고 함수 이름을 변경하여 난독화(Obfuscation)한다[1]. 스크립트를 분석하는 정적 분석을 어렵게 하고, 분석에 소요되는 리소스를 많이 소모하게 된다.

2.2 ARP Spoofing에 의한 악성코드 삽입

ARP Spoofing은 로컬 네트워크에서 패킷의 흐름을 공격자에게 향하도록 변경시키는 공격이다. 공격이 성공하게 되면 공격자의 사용자에게 전송되는 패킷을 가로챌 수 있는데, 사용자에게 전송되는 웹페이지를 변조하여 스크립트 코드나 태그를 삽입하는 중간 단계 공격을 할 수 있다. 서버 기반의 분석 방법으로는 이러한 유형의 변조 공격은 탐지할 수 없다.

2.3 악성코드의 다운로드 및 실행 공격

웹페이지에서 사용되는 스크립트 코드는 IE 브라우저의 Heap 메모리 영역을 사용할 수 있는데, 브라우저나 ActiveX의 취약점을 이용하여 버퍼 오버플로(Buffer Overflow)를 발생시키고, Heap 메모리 영역에 NOP+Shell Code를 배치시켜 특정 Shell Code를 실행 시키는 Heap Spray 공격 방법을 사용하거나, 웹페이지에 포함된 이미지 파일을 위장한 실행가능 파일을 통해 악성코드를 실행 시킨다. (그림 1)은 Heap Spray 공격의 기본적인 개념을 나타낸다.

2.4 기존 시스템의 탐지 방법

구글(Google)사의 Safe Browsing서비스는 자사의 검색엔진을 통해 악성코드유포 페이지에 대한 정보를 제공하고 있다. 웹크롤링을 통해 사전에 웹페이지를 수집하고, 스크립트에 대한 정적 분석[6]과 가상환경에서 악성코드의 행위(파일, 레지스트리, 프로세스)를 통해 위험한 웹사이트 URL을 DB로 생성하고, 사용자가 웹사이트를 검색할 때 URL을 DB에 조회하여 위험정보를 확인할 수 있다. 맥아피사의 SiteAdvisor도 분석 서버에 의한 사전 정보와 자사의 안티바이러스 엔진을 이용한 파일 검사를 이용한다. Caffein-Monkey 프로젝트[4]는 오픈 소스 Javascript 인터프리터인 Spider Monkey를 이용하여 스크립트 수행과정에서 할당되는 문자열 패턴을 찾고, x86 emulator를 이용하여 Shell Code 패턴에서 의심 행위를 탐지한다[11,16]. 한국 인터넷진흥원은 MC Finder 프로그램을 이용하여 2005년부터 20만개 이상의 웹사이트를 대상으로 수집하여 태그 분석, 문자열 디코딩, 구문분석을 통해 탐지 패턴을 추출하여 Black List 기반의 탐지 방법을 사용한다.

3. 제안 방법

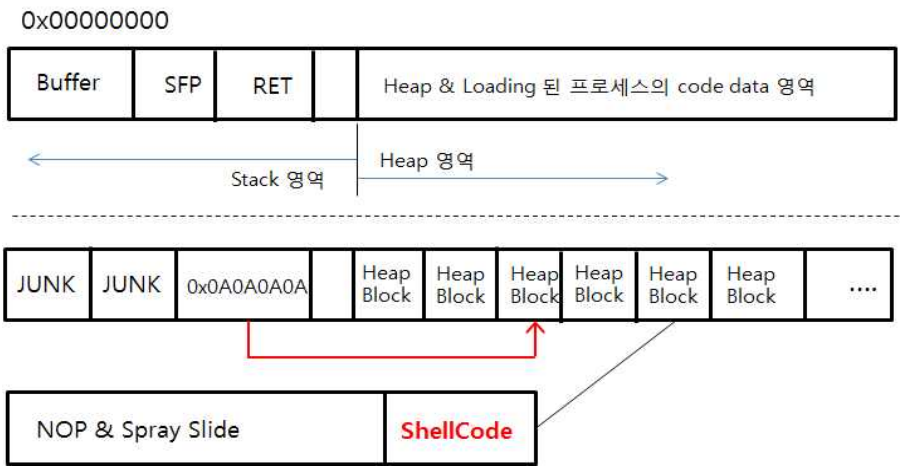
본 장에서는 논문에서 제안하는 탐지 시스템에 대해 설명하고 각각의 탐지 방법과 과정을 설명한다. 제안 시스템은 파일 다운로드와 실행흐름을 정상/비정상으로 분류하고, 다운로드 된 파일과 이상 스크립트를 검사하는 방법을 통해 악성코드 감염 시도를 탐지하고 실행을 차단하는 할 수 있도록 한다.

3.1 BHO를 이용한 URL 정보 수집

BHO(Browser Helper Object)는 Internet Explorer에 플러그인 되어 브라우저의 이벤트를 모니터링 할 수 있다. 사용자가 웹사이트에 접속하게 되면, 해당 URL로 접속하기 전에 Before_Navigate2 이벤트가 발생하게 되고, 접속 URL 정보를 수집할 수 있다[2].

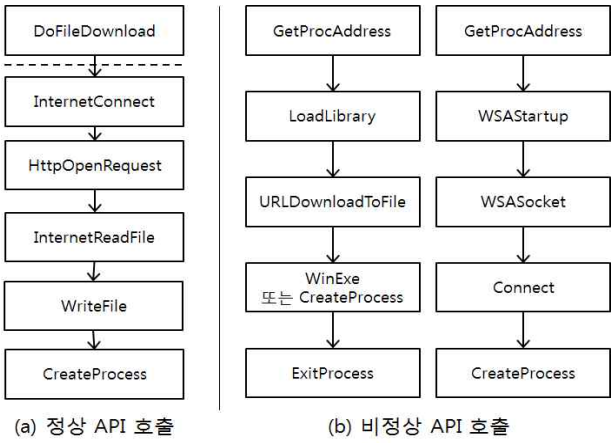
3.2 API Hooking을 이용한 다운로드 행위 탐지

API Hooking을 통해 정상적인 다운로드 과정에서 사용되는 API와 비정상적인 다운로드 과정에서 사용되는 API 목록을 분류하고, 매개변수를 통해 요청되는 파일 목록을 수집할 수 있다. 브라우저에서 웹페이지에 표시되는 이미지 파일을 요청하거나, 사용자에 의해 인지되는 정상적인 과정으로 파일을 다운로드 할 때 호출되는 API의 흐름은 (그림



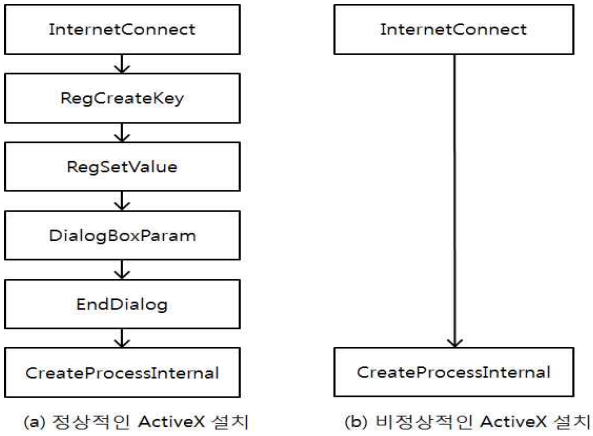
(그림 1) Heap Spray 공격기법의 개념

2(a))와 같고, Drive-by-download 기법에서 자주 사용되는 API의 흐름은 (그림 2(b))와 같다. 웹사이트 접속 시 호출되는 API 목록과 요청 파일 목록을 각각 정상행위, 비정상 행위로 분류하여 수집한다.



(그림 2) 다운로드 API 호출 행위 탐지

또한 ActiveX 설치 과정[3]에서 호출되는 정상 API 목록(그림 3(a))과 비정상 과정을 통한 API 목록(그림 3(b))을 모니터링 하여 이상탐지 여부를 판단한다.



(그림 3) ActiveX 설치과정의 API 호출 흐름

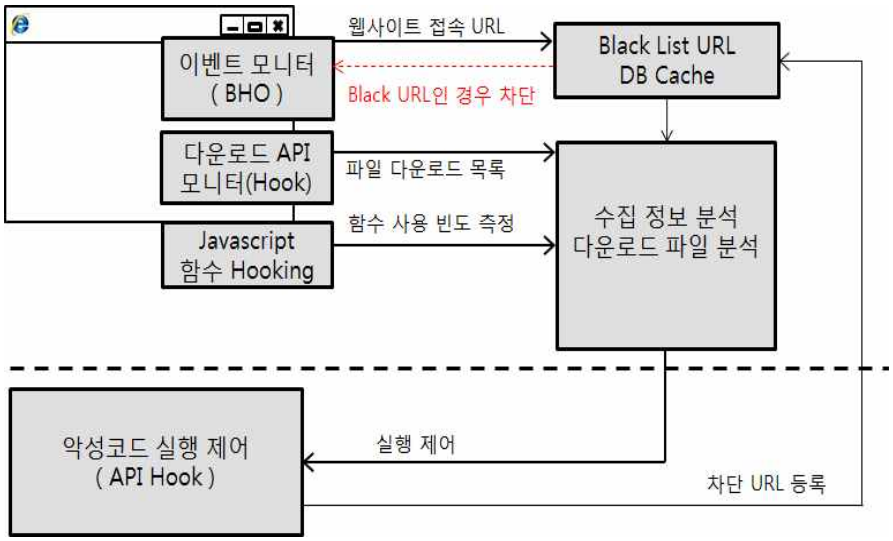
다음 <표 1>은 정상/비정상 API호출 분류 기준이다.

3.3 Javascript 내부함수 호출 탐지

Javascript에서 제공하는 내부 함수에 대한 API Hooking을 통해 이상탐지 여부를 판단한다. eval함수와 document.write함수는 수식 형태의 문자열 매개변수나 문자열이 아닌 매개변수를 자동 전환하여 동적 형태의 함수 호출 할 수 있어, 인코딩 된 문자열이나 함수명 분할 등의 방법으로 난독

〈표 1〉 정상/비정상 호출 API 탐지 목록

	API	모듈 이름	결과
#1	DoFileDownload	shdocvw.dll	정상
	InternetConnect	wininet.dll	
	HttpOpenRequest	wininet.dll	
	InternetReadFile	wininet.dll	
	WriteFile	kernel32.dll	
	CreateProcess CreateProcessInternal	kernel32.dll	
#2	UrlDownloadToFile	urlmon.dll	비정상
	CreateProcess CreateProcessInternal	kernel32.dll	
#3	UrlDownloadToCache,	urlmon.dll	비정상
	CreateProcess, CreateProcessInternal	kernel32.dll	
#4	WSAStartUp	Ws2_32.dll	비정상
	WSASocket	Ws2_32.dll	
	Connect	Ws2_32.dll	
	CreateProcess, CreateProcessInternal	kernel32.dll	
#5	InternetConnect	wininet.dll	정상
	RegCreateKey	Advapi32.dll	
	RegSetValue	Advapi32.dll	
	DialogBoxParam	User32.dll	
	EndDialog	User32.dll	
	CreateProcess, CreateProcessInternal	kernel32.dll	
#6	InternetConnect	wininet.dll	비정상
	CreateProcess, CreateProcessInternal	kernel32.dll	



(그림 4) 제안 시스템 구성도

화된 스크립트를 사용하기 위한 코드이다. Hooking을 통해 [9] 두 함수의 호출과 전달되는 매개변수의 비난독화(Deobfuscation)된 문자열 코드를 검사 할 수 있다.

3.4 다운로드 파일 포맷 검사

웹페이지에 포함된 멀티미디어 파일로 위장하여 악성코드를 실행하는 행위를 탐지하기 위한 방법으로, 파일의 헤더 포맷을 확인하고, PE 파일 포맷이 포함되어 있는지 검사한다. GIF 이미지 파일의 경우 첫 6 byte 부분의 서명 'GIF87a' 또는 'GIF89a'인지 확인한다. JPG 이미지는 6-9 byte '4A 46 49 46'(JFIF)값을 확인한다. 실행 가능 파일을 탐지하기 위해 파일 내의 'MZ'문자열 검사를 통해 확인한다.

3.5 제안 시스템의 구성

제안 시스템은 (그림 4)의 형태로 구성되며, 정상 다운로드 과정을 벗어난 다운로드 파일 목록에 포함되어 있거나, 이상 스크립트가 탐지 되었을 때, 그리고 파일 포맷이 비정상으로 의심되는 경우 시스템은 파일의 실행 또는 시작 프로그램의 등록을 차단한다.

4. 실험 및 결과

실험 환경 및 테스트에 사용된 URL 목록은 <표 4>와 같다. 일반적인 Windows XP SP2와 IE 7.0 버전을 대상으로 하였으며, 악성코드 유포 웹페이지 URL은 실행되는 바이너리를 기준으로 안티바이러스 프로그램에 의해 검증하는 방법(virustotal.com)을 사용하였다.

기존 탐지 방법을 이용한 시스템과 탐지율에 대한 비교 실험 결과는 다음과 같다.

〈표 2〉 제안 시스템의 실험 환경

구분	내용
운영체제	Microsoft Windows XP Professional SP2
브라우저	Microsoft Internet Explorer 7
브라우저 보안설정	· 서명 안 된 ActiveX 다운로드 (사용) · ActiveX 컨트롤 및 플러그인 실행 : 사용 · 파일 다운로드 : 사용
대상 웹페이지	· 유효한 악성코드 유포 웹페이지 URL
	· 정상 웹페이지 URL
	· 유효하지 않은 악성코드 유포 웹페이지
	· 기타 URL (정상 URL을 ARP Spoofing을 통해 변조한 웹페이지)

〈표 3〉 기존 탐지 시스템과 제안 시스템의 탐지율 비교

대상	오탐 1)	오탐 2)	정탐	미탐
구글 Safe Browsing	4/100	11/50	24/50	26/50
맥아피 Site Advisor	3/100	5/50	15/50	35/50
MC-Finder	4/100	6/50	22/50	28/50
제안 시스템	0/100	0/50	48/50	2/50

* 오탐 1)은 정상 웹페이지를 위험사이트로 분류한 경우
* 오탐 2)는 이전에 위험 사이트로 분류되었으나, 현재 정상인 웹페이지를 위험 사이트로 계속 분류한 경우

<표 3>의 실험 결과를 보면 기존의 탐지 시스템에 비해 탐지율이 높게 도출되었다. 제안 시스템의 탐지율이 높은 것은 악성코드 유포 웹페이지의 생성과 소멸 주기가 빨라짐에 따라, 기존의 서버 영역에서 주기적 분석 방법보다 제안 시스템이 신속적으로 대응할 수 있기 때문으로 판단된다.

〈표 4〉 기존 탐지 시스템과 제안 시스템의 기능 비교

항목	Safe Browsing	Site Advisor	MC-Finder	제안 시스템
유포지 변경 탐지	△1)	△1)	X	O
Arp spoofing 삽입 공격	X	X	X	O
신종 악성코드	X	X	X	O
난독화 스크립트	O	O	X	△2)
위장 미디어 파일	X	-	X	O

△1)은 구글의 Safe Browsing과 맥아피의 Site Advisor는 주기적인 수집/분석을 실행 하므로, 악성코드 유포지의 변경을 탐지하는데 한계가 있으며 △2)의 경우 제안 시스템에서 성능적인 이유로 스크립트 코드의 정적 분석을 하지 않아 일부 스크립트의 행위를 탐지하는데 한계점이 있다.

5. 결 론

본 연구는 사용자 영역에서 실시간으로 행위 모니터링을 통해 정상 행위와 비정상 행위를 분류하고, 비정상 행위에 의한 파일의 실행을 차단함으로써 효과적인 탐지가 가능하다는 것을 실험을 통해 확인하였다. 향후 연구 과제로는 정상행위와 비정상 행위에 대한 상세한 프로파일링과 NON-PE 파일의 악성코드를 보다 정밀하게 분석하여 탐지 효율을 높이는 방법에 대한 개선이 필요하다.

참 고 문 헌

[1] 한국인터넷진흥원. “홈페이지 은닉형 악성코드 유포 패턴 분석 방법 연구”, 2010.

[2] 신화수. 문종섭. “악성코드 은닉사이트의 분산적, 동적 탐지를 통한 감염피해 최소화 방안 연구”, 한국정보보호학회, 제 21권 제 3호, 2010.

[3] Rami Kawach. “NEPTUNE : Detecting Web-Based Malware via Browser and OS Instrumentation”, Black Hat USA. 2010

[4] Ben Feinstein, Daniel Peck . “Caffeine Monkey : Automated Collection, Detection and Analysis of Malicious JavaScript”, 2007.

[5] M.Egele, E.Kirda, C.Krugel “Mitigating drive-by download attacks: challenges and open problems, open research problems”, Ifip International Federation For Information Processing. pp.460-465, 2009.

[6] 이성욱, 배병우, 이형준, 조은선, 홍만표. “정적 분석을 이용한 알려지지 않은 악성 스크립트 감지”, 한국정보처리학회, 정보처리학회논문지C, 제 9-C권 제 5호, pp.765-774, 2002. 10.

[7] 심원태, “악성코드 은닉사이트 탐지시스템 개발과 운영(MC Finder)”, 2010.

[8] Zhi-Yong Li, Ran Tao, Zhen-He Cai, Hao Zhang. “A Web Page Malicious Code Detect Approach Based on Script Execution”, ICNC.09. Fifth International Conference on pp.308-312, 2009.

[9] http://securitylabs.websense.com/content/Blogs/3198.aspx

[10] JooBeom Yun, Youngjoo Shin, “MiGuard : Detecting and Guarding against Malicius Iframe through API Hooking”, IEICE Electronics Express. pp.460-465, 2011.

[11] x86 shellcode detection and emulation. http://lbemu.mwcollect.org

[12] 이성욱, 방효찬, 홍만표. “코드 삽입 기법을 이용한 알려지지 않은 악성 스크립트 탐지”, 한국정보과학회 제 29권, pp.593-754, 2002.

[13] http://www.siteadvisor.com/studies/Mapping_Mal_Web_jun2009.pdf, 2009

[14] http://code.google.com/intl/ko-KR/apis/safebrowsing/developers_ guide.html

[15] 최재영, 김성기, 이혁준, “원격코드검증을 통한 웹컨텐츠의 악성스크립트 탐지” 한국정보처리학회, 정보처리학회논문지C, 제 19-C권 제 1호, pp.47-54, 2012. 02.

[16] 하정우, 임종인.“WhiteList 기반의 악성코드 행위 분석을 통한 악성코드 은닉 웹사이트 탐지 방안 연구”, 2011. 08.

[17] 강태우, 조재익, 정만현, 문종섭. “API Call 단계별 복합 분석을 통한 악성코드 탐지” 정보보호학회논문지 제 17권 제 6호, 2007. 12.

공 익 선

e-mail : peterii@korea.ac.kr

2001년 한국외국어대학교 제어계측공학과 (학사)

2012년 고려대학교 정보보호학과 석사과정

2002년~2005년 하우리 연구소 연구원

2005년~현 재 AhnLab Inc. 선임연구원

관심분야 : 네트워크 침입 탐지, 악성코드 분석

조 재 익

e-mail : chojaeik@korea.ac.kr

2005년 동국대학교 컴퓨터학과(학사)

2008년 고려대학교 정보보호학과(석사)

2012년 고려대학교 정보보호학과 박사과정

2009년~2011년 Ilinois Institute of technology 선임연구원

현 재 삼성전자 System LSI 선임연구원

관심분야 : 네트워크 모델링, 패턴인식

손 태 식

e-mail : tsshon@ajou.ac.kr

2000년 아주대학교 정보 및 컴퓨터공학부 (학사)

2002년 2월 아주대학교 컴퓨터공학(석사)

2005년 8월 고려대학교 정보보호대학원 (박사)

2007년~2011년 삼성전자 DMC 연구소 책임연구원

2011년~현 재 아주대학교 정보컴퓨터공학부 부교수

관심분야 : 무선/모바일 네트워크 보안, 무선 센서 네트워크 이상탐지

문 종 섭

e-mail : jsmoon@korea.ac.kr

1983년 서울대학교(석사)

1991년 llinois Institute of technology (전산학 박사)

1981년~1985년 금성 통신 연구소 연구원

1993년~현 재 고려대학교 정보보호대학원 교수

관심분야 : 생체인식, 침입탐지, 운영체제