제21회 한국정보처리학회 춘계학술발표대회 논문집 제11권 제1호 (2004.5)

# 악성 행위에 대한 시스템 로그 분석에 관한 연구

김은영, 이철호, 오형근, 이진석
국가보안기술연구소
e-mail : {eykim, chlee, hgoh, jinslee}@etri.re.kr

# The Study on System Log Analysis of Malicious Behavior

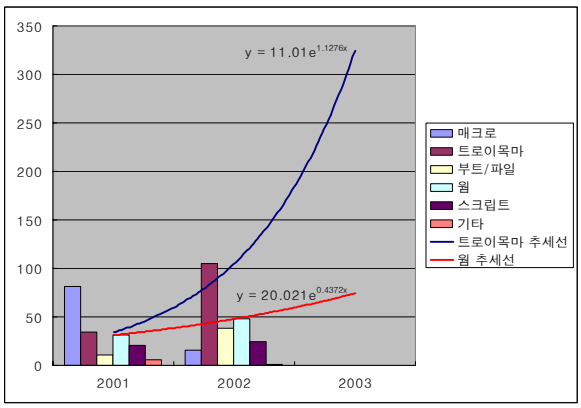
EunYoung Kim, CheolHo Lee, HyungGeun Oh, JinSeok Lee
National Security Research Institute

**요 약**

1980 년 후반. MIT 에 버너스 리 교수가 인터넷 상에 웹(WWW)을 창시하면서부터 우리의 일상생활은 엄청난 변화를 가져왔다. 시·공간을 초월할 수 있는 인터넷이라는 가상 세계에서는 개인뿐만 아니라 정치·경제·사회등 모든 분야에 걸쳐 인터넷을 통한 쉽고 간편하며 빠른 교류가 이루어짐으로써 이제 더 이상 네트워크를 이용하지 않는 분야는 찾아 볼 수 없을 것이다. 그러나 이러한 현실 속에서 인터넷은 항상 순기능만을 수행하지는 않는다. 특히 악성코드를 이용한 사이버 침해 행위 기술이 인터넷의 발전과 함께 동시에 발전함으로써 이제는 악성코드를 이용한 사이버 침해 행위를 방지하고자하는 노력을 해야할 것이다. 따라서 본 논문에서는 악성코드를 탐지하기 위해 실시간 시스템 모니터링 도구를 이용하여 악성코드가 시스템에 어떠한 침해행위를 행하고, 해당 침해 행위 모니터링 로그 분석을 통해 기존의 알려진 악성코드뿐만 아니라 알려지지 않은 악성코드를 탐지할 수 있는 악성 패턴 분석 및 추출에 초점을 두어 기술하였다.

## 1. 서론

2001 년도 이후 인터넷의 급성장과 함께 악성코드를 이용한 침해 사건이 상당히 많이 발생하였다. 또한 이러한 침해 사건 중 악성코드를 이용한 침해 사건이 날로 증가하고 있는 추세이다.




[그림 1] 연도별 악성코드 침해 사건
※한국 CERTCC-KR 에서 통계 인용

[그림 1]에서 보듯이 연도별 악성코드의 증가율이 급성장하고 있으며 특히 트로이 목마와 웜의 침해사건이 다른 악성코드에 비해 상당한 증가세를 나타내고 있다. 따라서 본 논문에서는 기존의 시그너쳐 기반의 패턴 매칭 기법의 단점 및 한계점을 인지하고 시스템 모니터링 도구를 통해 악성 프로그램의 악성행위에 발생되는 시스템 이벤트들을 모니터링한 후 이를 바탕으로 발생된 로그 분석을 통해 악성코드 행위별 패턴 분석을 시도하고자 한다.

본 논문의 구성은 2 장에서는 악성코드의 종류 및 침해 사건에 대해 기술하고 3 장에서는 기존 악성코드의 탐지 기법에 대해 기술한다. 4 장에서는 악성코드의 시스템 모니터링 실험 및 해당 로그 분석에 대한 결과를 기술하고 5 장에서 결론을 맺도록 하겠다.

## 2. 악성 코드의 종류

이 장에서는 악성코드의 종류 및 악성코드를 이용한 침해 사건에 대해 기술한다.

### 2.1 컴퓨터 바이러스

컴퓨터 시스템의 부트 영역, 메모리 영역, 파일 영역 등에 기생하면서 자기 증식 및 복제가 가능하고 프로그래머가 인위적으로 제작한 파괴성을 가진 컴퓨터 프로그램을 컴퓨터 바이러스로 정의하고 있다. 생물학적 바이러스와 마찬가지로 자신을 복제하기 위해 숙주 파일을 필요로 하며 돌연변이되고 진화하면서 백신 프로그램에 대항하기도 한다. 1985 년 파키스탄의 한 소프트웨어 개발자가 자신의 소프트웨어 불법 복

제에 대한 보복의 수단으로 처음 만든 후 1999 년부터는 해킹 기법을 응용하여 유포되는 악의적인 형태로 발전하고 있다.

### 2.2 웜

웜이란 다른 시스템에 직접적인 피해를 끼치지는 않지만 가능한 범위내에서 단순히 자기 자신을 계속 복제하는 프로그램을 의미한다. 최근 전자우편을 통해 전달되는 형태를 많이 가지며 바이러스와는 달리 정규 파일이나 부트영역에서는 거의 자리 잡지 않으며, 소프트웨어나 시스템의 보안 허점을 이용한다. 웜은 다른 운영체제에도 전파된다는 점에서 컴퓨터 바이러스와 비교되며, 전파되기 위한 사용자의 간섭이 거의 없거나 전혀 없는 것이 그 특징이라 하겠다.

### 2.3 트로이목마

일리아드의 트로이목마 신화에서 유래한 트로이목마는 정상적인 프로그램으로 위장하고 있으나 그것을 실행시키면 시스템에 악영향을 주거나 데이터를 파괴하는 특징을 가지고 있다. 트로이목마는 자기 복제 능력은 없고 고의적으로 포함되었다는 점에서 프로그래머의 실수인 버그(bug)와 다르다. 또한, 자기 자신을 다른 파일에 복사하지 않는다는 점에서 컴퓨터 바이러스와 다르다. 따라서 트로이목마 프로그램이 발견되면 해당 시스템에서 삭제만 하면 퇴치된다.

### 2.4 악성실행코드

실행코드란 사용자 데스크 탑 컴퓨터 시스템의 전부 또는 일부 자원을 통제할 수 있는 스크립트 또는 컴퓨터 프로그램을 말한다. 이러한 프로그램은 누구나 쉽게 제작할 수 있고 브라우저를 수행할 수 있는 어떤 컴퓨터에서도 수행이 가능하여 현재까지 그 유형이나 종류가 꾸준히 증가하고 있는 추세이다. 실행코드 유형에는 자바 애플릿(Java Applet), 자바 스크립트(Java Script), 엑티브 엑스 컨트롤(ActiveX Control), 비주얼 베이직 스크립트(Visual Basic Script) 등이 대표적이며 이외 다수의 실행코드들이 존재한다.

### 2.5 악성코드 해킹 사건

최근 악성코드를 이용한 국내 해킹 피해 사례는 다음과 같다.

- -K 대학의 백오리피스를 이용한 주요자료 유출 : 1999. 3.
- -국외 해커의 H 대학의 홈페이지 해킹 사건 : 1999. 6.
- -K 대학 침입 후 18 만여 사이트 취약점 수집 : 1999.11.
- -한나라당 홈페이지 해킹사건 : 2000.1.
- -취업준비생 전산망 해킹해 성적조작

또한 국외의 악성코드를 이용한 해킹 피해 사건은 다음과 같다.

- -미상원의 웹 사이트가 MOD 라는 해커 집단에 의해 침입
- -바이러스 보안 업체인 시만텍 사의 웹 사이트 해킹 : 1999.8.
- -YAHOO, CNK, E-Bay 등 사이트가 DOS 공격을 당함 : 2000.2.
- -미연방 수사국 FBI 웹 사이트 해킹 : 2000.2.
- -미국 온라인 증권사 NDB 와 마이크로 소프트 웹 사이트 해킹 : 2000.2.
- -LA 타임즈 웹 사이트 해킹 : 2000.2.

## 3. 기존 악성 코드 탐지 기법

이 장에서는 기존 악성코드의 탐지 기법에 대해서 기술한다.

### 3.1 시그너쳐(Signatured) 기반의 탐지 기법

시그너쳐 기반의 스캐닝 기법은 현재 바이러스 및 트로이 목마 탐지 제품등에서 많이 사용하고 있는 기법으로 해당 악성 프로그램의 특정 스트링을 백신 시스템에 포함한다. 그리고 해당 컴퓨터에 검사하고자하는 파일을 백신에 포함되어있는 악성 프로그램의 시그너쳐와 비교하여 해당 파일이 악성 프로그램인지 판단하게 된다. 따라서 이러한 스캐닝 기법은 탐지 기법중에서 가장 접근하기 쉽고 탐지 결과 또한 확실한 방법일 수 있지만, 알려지지 않은 악성 코드나 변종 등은 해당 백신 프로그램에서 시그너쳐를 가지고 있지 않는다면 탐지는 불가능하다.

### 3.2 휴리스틱(Heuristic) 탐지 기법

휴리스틱 기법은 시그너쳐 기반의 스캐닝 기법과 유사하다. 그러나 휴리스틱 스캐닝 기법은 시그너쳐 기반의 스캐닝 기법과 같이 어떤 특정 시그너쳐를 찾는 대신에 보통의 응용 프로그램에서 발견할 수 없는 전문가들의 분석에 따른 경험적 규칙에 기반한 어떤 특별한 명령이나 시그너쳐를 찾게 된다. 따라서 이러한 휴리스틱 기법을 이용하여 구현된 탐지 엔진은 바이러스, 웜 또는 트로이 목마 등 알려지지 않은 새로운 악성행위를 탐지할 수 있다.

### 3.3 행위(Behavior) 기반 탐지 기법

행위 기반 탐지 기법은 두가지 시스템으로 분류할 수 있다. 정책 기반 제한 시스템(Policy-Based Systems)은 특정 행위를 모니터링하여 그 행위를 허용할 것인지 제한할 것인지를 결정하는 것이다. 만약 어떤 프로그램이 운영체제에 어떠한 행위를 요청하였을 경우, 정책 기반 시스템은 정책 데이터베이스에 비교하여 해당 행위를 허용할 것인지 제한할 것인지 결정하는 것이다. 이와는 대조적으로 전문가 기반 시스템(Expert-Based Systems)은 전문가가 전체 바이러스를 분석하여 그들이 시스템에 끼치는 바이러스 행위를 분석하고 의심이 가는 행위에 대해 직접 제한을 가할 수 있게 구성한다. 또한 이러한 전문가 기반 시스템은 이미 알려진 바이러스 코드의 80%가 시작 프로그램에 해당하는 시스템 파일을 접근하기 전에 레지스트리를 먼저 접근하기에 이에 준하는 정책을 설정할 수 있다. 따라서 행위 제한 기법에서는 시작 프로그램의

레지스트리 영역을 접근하여 수정을 하게되면 이 행위에 대해 제한을 하게 된다. 그러나 이러한 전문가 시스템 또한 여전히 'False Positive' 요소를 내제하고 있다.

### 3.4 CRC(Cyclic Redundancy Check) 검사법

CRC 는 시리얼 전송에서 데이터의 신뢰성을 검증하기 위한 에러 검출 방법의 일종이다. CRC 에 의한 방법은 높은 신뢰도를 확보하며 에러 검출을 위한 오버헤드가 적고, 랜덤 에러나 버스트 에러를 포함한 에러 검출에 매우 좋은 성능을 가지고 있는 것을 특징으로 한다. 이러한 에러 검출 방법을 이용하여 CRC 검사법에서는 악성코드에 감염되기 전의 CRC-32 를 생성 후 이를 가지고 있다가 검사 파일의 CRC-32 값과 비교하여 악성코드 탐지하고 있다. CRC 검사법은 오진율이 낮다는 장점을 가지고 있지만 반면에 1 byte 만 변형되어도 진단하지 못함으로써 변형된 악성코드에 대한 탐지가 불가능하다는 단점을 가지고 있다.

### 3.5 가상 실행(Virtual Executable) 탐지 기법

백신 프로그램 안에 가상의 컴퓨터 환경을 만든 후 파일이 동작하는 기능 하나하나를 분석해서 악성코드로서의 특정한 동작을 하느냐 하지 않느냐로 구분하여 악성코드 존재 여부를 찾아내는 검사법이다. 일종의 시뮬레이션(simulation, 모의 실험) 기법이라고도 한다. 즉, 가상실행엔진을 이용하여 악성코드를 실행해 가면서 추적하는 방식이며, 기존의 패턴 매치를 이용한 방법에서 해결하지 못했던 복잡한 암호화 및 다형성 방법에까지 폭넓게 적용 가능하다는 장점이 있다.

### 3.6 면역 시스템(Immunity System)

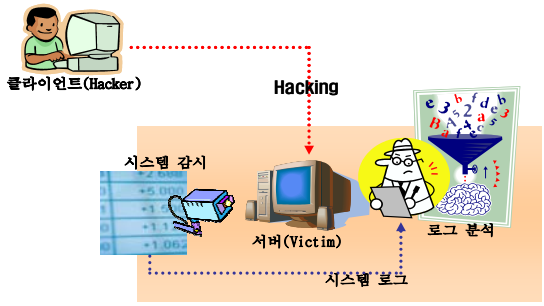
자연적인 면역 계통들의 역할을 컴퓨터 보안 시스템에 적용함으로써 기존 보안 시스템의 한계에서 벗어나 능동적이고 항구적인 보안 체계를 갖추고자 하는 의도에서 자연 면역 시스템의 컴퓨터 보안 시스템에 대한 적용 방법이 연구되었다. 컴퓨터 보안 시스템에 적용되는 면역 시스템은 자연 면역체계와 마찬가지로 컴퓨터 시스템을 보호하는 문제를 "자기 자신이 아닌 것" 과 "자신인 것" 을 구별하는 문제로 해결할 수 있을 것으로 판단하고 있다. 이러한 면역 시스템은 시만텍 사, UNM(University of New Maxico) 및 IBM 등에서 연구되고 있으나 정상 프로그램과 악성 프로그램을 명확히 구분하지 못하는 긍정적 오류(False Positive)의 발생률이 높아 아직 상용화 되지는 못하고 있는 실정이다.

## 4. 시스템 모니터링을 통한 악성 행위 실험

이 장에서는 시스템 모니터링 도구를 이용하여 악성코드의 행위별 로그 분석 실험에 대해 기술한다.

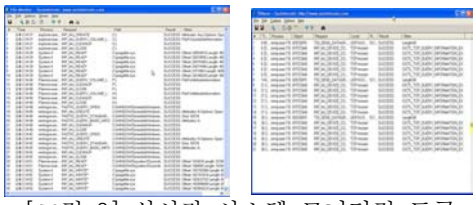
### 4.1 실험 환경

본 절에서는 악성 코드가 악성 행위시 시스템에 어떠한 영향을 미치는가에 대한 시스템 로그 분석 실험을 행할 것이다. 해당 실험을 위해 실험 환경 및 실험에 사용될 악성코드와 시스템 모니터링 도구에 대해 기술하겠다. 먼저 실험 환경은 [그림 2]에서 보는바와 같다.



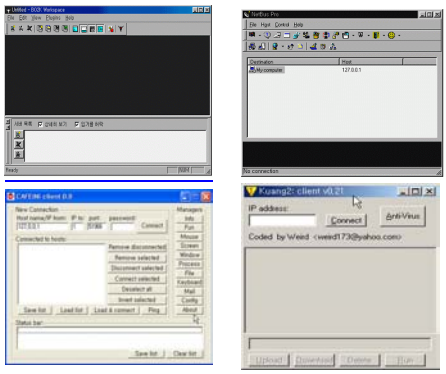

[그림 2] 실험 환경

클라이언트는 서버에 설치되어 있는 트로이 목마를 조정하는 프로그램이 설치가 되고, 서버에서는 클라이언트의 조정을 받는 트로이 목마 서버 프로그램이 설치가 된다. 서버에는 악성 행위에 따른 패턴을 추출하기 위해 시스템 모니터링 모듈이 설치한다. 실험 시스템 운영체제는 윈도우 2000 Professional 이며, 시스템 모니터링 모듈로는 시스인터널스(SYSInternals)에서 만든 FileMon 과 TDIMon 을 이용한다.



[그림 3] 실시간 시스템 모니터링 도구

악성코드 시스템 로그 분석에 사용되는 백오리피스, 넷버스, Cafeini, Kuang 을 이용하였다.



[그림 4] 실험에 사용된 악성코드

각각의 악성코드들은 원격 관리가 가능하며 설치가 매우 용이하고 사용하기 쉬운 인터페이스를 가지고 있다. File manager, Registry managerem, Application Redirect 등과 같은 원격 관리 기능과 더불어 화면 캡쳐, 키보드 후킹 등과 같은 스파이 기능을 제공한다.

### 4.2 악성코드의 시스템 침해 로그 분석

클라이언트가 서버에 악성 행위를 하기 위한 초기

작업으로 연결 요청시 보여지는 행위에 대한 시스템 로그 및 그에 대한 로그 분석 결과이다. 다음 로그는 백오리피스가 서버(Victim)에 연결설정시 발생되는 TDIMon 의 로그이다.

| # | Time | Process | Request | Local | Result |
|---|---|---|---|---|---|
| 1 | 11.5641 1176 | Backdoor .Win32. | IRP_MJ_CREATE | TCP:Connection obj | S |
| 2 | 11.5641 6540 | Backdoor .Win32. | TDI_ASSOCIATE _ADDRESS | TCP:Connection obj | S |
| 3 | 11.5642 3664 | Backdoor .Win32. | IRP_MJ_DEVICE _CONTROL | TCP:<none> | S |
| 4 | 11.5644 3108 | Backdoor .Win32. | TDI_SEND | TCP:129.254.1 63.105:20000 | S |
| 5 | 11.5967 4970 | Backdoor .Win32.: | TDI_SEND | TCP:129.254.1 63.105:20000 | S |
| 6 | 11.5968 2513 | Backdoor .Win32 | TDI_SEND | TCP:129.254.1 63.105:20000 | S |
| 7 | 11.5969 0056 | Backdoor .Win32 | TDI_SEND | TCP:129.254.1 63.105:20000 | S |
| 8 | 11.5970 1286 | Backdoor .Win32 | TDI_SEND | TCP:129.254.1 63.105:20000 | S |
| 9 | 11.5970 7907 | Backdoor .Win32 | TDI_SEND | TCP:129.254.1 63.105:20000 | S |
| 10 | 11.5971 8467 | Backdoor .Win32. | TDI_SEND | TCP:129.254.1 63.105:20000 | S |
| ..... | | | | | |
| 43 | 11.5975 9618 | Backdoor .Win32 | TDI_SEND | TCP:129.254.16 3.105:20000 | S |

**Result : S(Success)*

백오리피스, 넷버스, Cafeini 그리고 Kuang 를 시스템 모니터링 도구를 통해 네트워크 연결 설정시 생성되는 파일 로그를 분석한 결과는 다음과 같다. 백오리피스, 넷버스, Cafeini, Kuang 을 가지고 연결 요청에 대한 로그 분석 결과 File Mon 에서는 어떠한 시스템 이벤트에서는 공통점을 발견할 수 없었고, TDIMon 에서는 넷버스, Cafeini, Kung 는 'IRP_MJ_CREATE → TDI_ASSOCIATE_ADDRESS' 이벤트가 호출되었고, 백오리피스에서는 'IRP_MJ_CREATE → TDI_ASSOCIATE_ADDRESS → IRP_MJ_DEVICE_CONTROL'이벤트가 1 번의 행위에 따라 주기적으로 발견되었다.

다음은 파일 보기에 대한 시스템 이벤트 로그 분석 결과이다. 실험한 악성코드의 시스템 침해 로그 분석 결과는 다음과 같다. 파일 삭제인 경우 FileMon 의 결과는 두가지로 나눌 수 있다. 먼저 'IRP_MJ_CREATE → IRP_MJ_SET_INFORMATION → IRP_MJ_SET_INFORMATION → IRP_MJ_CLEANUP → IRP_MJ_CLEANUP → IRP_MJ_CLOSE'와 같은 행위 패턴이 보이는 것은 Kuang, 넷버스 이고, 'FSCTL_IS_VOLUME_MOUNTED → IRP_MJ_CREATE → IRP_MJ_SET_INFORMATION → IRP_MJ_SET_INFORMATION → IRP_MJ_CLEANUP → IRP_MJ_CLEANUP → IRP_MJ_CLOSE'와 같은 행위 패턴 형태를 보이는 것은 백오리피스, Cafeini 이다. TDIMon 은 백오리피스, Cafeini, Kuang 가 'TDI_SEND' 메시지 하나만 보인다. 파일 리스트 보기 행위에 대한 로그 분석 결과는 다음과 같다.

시스템 파일 보기에 해당하는 악성 행위의 시스템 로그 분석 결과는 다음과 같다. 먼저 FileMon 인 경우 두가지로 나눌 수 있다. 백오리피스, 넷버스, Kuang 은 'FSCTL_IS_VOLUME_MOUNTED → IRP_MJ_CREATE → IRP_MJ_DIRECTORY_CONTROL → IRP_MJ_CLEANUP → IRP_MJ_CLOSE'의 형태를 보이고 있으며, Cafeini 는 'FSCTL_IS_VOLUME_MOUNTED' 의 이벤트 만이 보이고 있다. TDIMon 의 결과는 세가지로 나눌 수 있다. 먼저 넷버스 는 'TDI_EVENT_RECEIVE' 이벤트 패턴을 보이고 있고, Kuang 는 'TDI_EVENT_RECEIVE → TDI_SEND' 이벤트 패턴을 보이고 있다. Cafeini, 백오리피스는 'TDI_SEND' 이벤트 패턴만이 반복적으로 보이고 있다.

## 5. 결론

본 논문에서는 시스템 모니터링 툴 두가지를 가지고 악성 행위별 패턴을 추출하고자 실험을 하였다. 만약 악성 행위별 패턴 추출이 가능하다면 기존의 시그너쳐 기반의 스캐닝 기법의 단점인 알려지지 않은 악성코드를 탐지할 수 있을뿐 아니라 새로운 악성 코드 탐지 기법으로 제안되고 있는 행위 제한 기법등의 'False-Positive'를 줄일 수 있는 기초 정보로 사용할 수 있을 것이다. 그러나 백오리피스를 가지고 악성 행위에 대한 실험을 하였을때 어떤 악성 행위에 대해 시스템 패턴이 도출되는듯 하였으나, 넷버스 등의 다른 악성코드를 가지고 같은 악성 행위에 대한 시스템 이벤트 모니터링 결과 새로운 트로이 목마를 추가할 때마다 서로 상이한 시스템 이벤트의 패턴 양상을 보이고 있다. 또한 어떤 악성 행위에 대한 패턴이 있더라도 악성 행위별 대표될 수 있는 패턴을 추출할 수 없었다. 예를 들어 TDIMon 의 패턴에서보면 파일 삭제, 파일 보기 모두가 'TDI_SEND' 이벤트 메시지를 각자 자신의 악성 행위에 대한 결과로 나타내주고 있기 때문이다. 따라서 본 논문의 실험을 토대로 추후 악성 행위별 패턴을 추출하기 위해서는 해당 논문에서 실시하였던 시스템 모니터링 방법 이외에 다른 모니터링 방법 모색과 더불어 추후 해당 로그 분석시 로그 분석 방법을 데이터 마이닝 및 AI 기법을 이용하여 좀더 체계적인 패턴 분석 방법을 도입해야할 것이다.

## 참고문헌

[1]H.Han, X.L. Lu, J.Lu, C.Bo, R.L.Yong, "Data mining aided signature discovery in network-based intrusion detection system ", ACM SIGOPS Operating Systems Review, Volume 36 Issue 4, October 2002.

[2] W.Lee, W.Fan , "Special section on data mining for intrusion detection and threat analysis: Mining system audit data: opportunities and challenges", ACM SIGMOD Record, Volume 30 Issue 4, December 2001.

[3]M. Schmall. "Heuristic Techniques in AV Solutions: An Overview ", Feb 2002.

[4]C. Nachenberg. "Behavior Blocking : The Next Step in Anti-Virus Protection", March 2002.