# 악성행위 판단 기술

김 준 모[1] 조 성 제[2] 황 병 연[3]

목 차



- 요 약 -

컴퓨터 시스템 내부에 존재하는 악성행위를 탐지하는 기술은 보안전문가의 기술과 경험 등에 의해 미리 발견되고 분석된 보안취약성 및 악성행위 등을 분석대상 시스템에서 찾아낸 후, 발견된 사항을 데이터베이스화하여 분석대상 시스템에 그러한 사항들이 있는지를 비교 탐색하는 것이다. 그러나, 이러한 기술은 새로운 악성행위 및 악성기술에 대한 대응책으로는 사용할 수 없다.

본 연구는 주어진 소프트웨어 안에 악성행위가 은닉되어 있는지를 알아내는 기법에 대한 것이다. 악성행위의 종류와 구현기술에 관계없이 사용자의 데이터를 갈취하려는 대부분의 악성동작을 포착할 수 있도록 고안된 기법이다. 어떤 데이터의 액세스 및 이동이 데이터의 탈취인지 아닌지를 구분하는 게 큰 문제이지만, 본 연구에서는 그 문제에 대한 타당한 해결책을 제시하고자 한다. 현 단계에서는 기술의 타당성을 보이고 개발을 위한 검토를 한다. 궁극적으로는 아직 알려지지 않은 형태의 악성행위들을 찾아내고, 그에 포함된 알려지지 않은 악성기법들을 분석하는 과정을 자동화함으로써 소프트웨어 보안취약성 발견기술을 발전시키고자 한다.

## 1. 서 론

주어진 컴퓨터 시스템에서 알려지지 않은 악성행위를 찾아내는 문제를 해결하기 위해서는 여러 가지 접근방법을 제시할 수 있겠으나, 전산 전반의 지식을 활용하며 대상 프로그램을 분석하여 악성행위의 형태를 파악하는 전형적인 방법으로는 무한하게 많은 조합의 경우들을 분석해야 하는 난관에 부딪히게 된다.

본 연구는 악성행위를 하는 트로이목마의 활동을 탐지해 냄으로써 트로이목마의 위치를 찾아내고 분석할 수 있는 기반을 마련하는 것이다. 트로이목마는 타인의 악성의도에 의해 몰래 사용자의 컴퓨터에 설치된 프로그램으로서 자동으로 은밀히 실행되며, 사용자의 정보를 빼내거나 사용자의 컴퓨터에 침입하는 도구로 사용된다.

컴퓨터를 이용한 범죄는 종류와 기술이 다양하고 컴퓨터 시스템의 모든 요소를 악용대상으로 하므로 한가지의 일관적인 방법으로는 탐지하거나

1) 한국정보보호진흥원 기술단 선임연구원
2) 단국대학교 정보컴퓨터학부 교수
3) 가톨릭대학교 컴퓨터정보공학부 교수

분석할 수 없었다. 그리고 각각의 악성기법들에 대해 개별적으로 대응하는 기존의 대응기법은 복잡해지는 컴퓨터 소프트웨어 시스템을 보호하는 접근방법으로서는 부족하다. 하지만 컴퓨터를 이용한 악성기법들이 아무리 다양해도 그 주요목적 중의 하나는 최종적으로 사용자의 데이터를 탈취한다는 것을 전제로 하면 다소 포괄적인 해결안을 찾을 수 있다.

특정한 데이터 파일의 비정상 액세스를 포착하여, 포착시점을 중심으로 사건을 재연할 수 있는 여건을 조성하고, 추후 사건을 재연해가면서 악성행위의 기점을 파악해내는 시나리오를 적용하면 매우 포괄적인 악성행위 탐지기술을 구현할 수 있다. 본 안은 침입대응기법으로 활용될 수도 있으며 아무리 교묘한 기법을 구사하는 악성행위라도 빠져나가기 힘든 일종의 그물을 쳐두는 방식이다. 일단 발견된 새로운 악성행위는 분석되어 참고자료로 활용된다. 악성행위의 발생을 기다리다가 포착하는 본 연구의 방안은 수동적인 접근이 되겠으나 알려지지 않은 악성기법에 대해 무방비인 현재의 상황에서는 이 방안이 최선의 길이라 할 수 있다.

## 2. 센서파일을 이용한 트로이목마 탐지

센서파일(sensor file)은 주어진 컴퓨터 시스템에서 정상적인 경우에는 사용되지 않는 dummy 파일이다. 센서파일은 경우에 따라 적절한 임의의 데이터를 포함할 수도 있으며, 그 파일이름과 내용은 프로그램에 의해 자동 생성되고 모든 디렉토리의 파일사이에 분산 배치된다. 한편, 트로이목마는 잠입한 시스템에서 어느 파일을 탈취할 것인가를 고민하게 되는데 트로이목마를 이용하는 Spy Master가 정확한 파일이름을 알고있지 않는 이상 임의의 사용자 파일을 뒤지게 된다. 그러한 과정에서 센서파일을 건드리게 되는 것이다. 센서파일기법은 트로이목마를 상대로 지뢰를 설치하는 기법이다. 트로이목마가 사용자의 파일을 탈취하는 동작을 시도하면 결국 지뢰를 밟게 되는 것이다.

센서파일이 어떠한 이유에선지 컴퓨터의 프로세스에 의해 오픈 되면, 컴퓨터 화면에 경고 창이 뜨고, 컴퓨터의 포트를 통한 모든 데이터 전송 또는 데이터 수신이 중단되게 한다. 센서파일이 컴퓨터 시스템을 빠져 나오게 되면, intranet manager는 해당 컴퓨터에 이 사실을 통보한다. 이러한 두 가지 상황은 전혀 액세스되지 않도록 정해 놓은 dummy 파일이 액세스된 경우이다. 이러한 경우는 다음과 같이 세 가지로 해석할 수 있다.

첫째, 사용자의 컴퓨터에 설치된 트로이목마가 어떤 상황이나 이벤트의 발생에 의해 또는 주기적으로 활동을 개시하여 임의의 사용자 데이터를 전송하려 하는 과정에서 센서파일을 건드리는 경우다.

사용자의 컴퓨터에 트로이목마를 설치한 후 탈취된 파일을 전송받는 악성행위자 즉, Spy Master는 일반적으로 탈취대상 사용자 파일의 이름을 알지 못하며, 트로이목마는 그다지 길지 않은 프로그램이라 판단능력이 없으므로 dummy 파일을 구분하여 가려낼 수 없다. 따라서 트로이목마는 사용자의 컴퓨터에 있는 파일을 임의로 선택하여 Spy Master에게 보내려 하는 것이다. 이런 과정에서 트로이목마는 확률적으로 센서파일을 건드리게 된다. 따라서 보호하고자 하는 파일의 중요도에 따라 센서파일의 개수가 증가할 것이다.

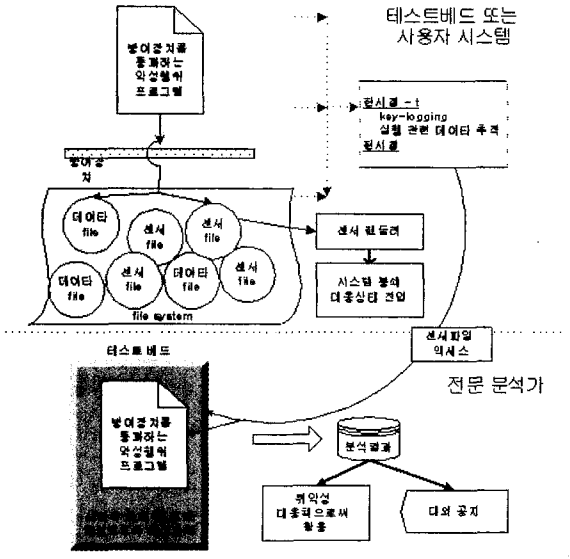
참고로, 우리의 목표는 불특정 다수의 사용자 시스템에 설치되어 있는 트로이목마를 탐지하는 것이다. 악성행위자는 자신이 원하는 파일의 목록을 트로이목마에게 미리 알려줄 수도 있지만, 이렇게 특정 사용자를 대상으로 하여 탈취할 파일들의 이름을 이미 알고 있는 경우는 사건 이전에 사용자와 악성행위자 사이에 정보의 공유가 있었다고 볼 수 있으므로 본 문제의 해결 대상에서 제외한다.

우리의 목표는 기존의 보안체계에서는 감지할 수 없던 불특정 문제점을 탐지해내는 것이므로 시스템 파일 등의 특정 파일에 대해서 보안기법을 별도로 설정해 두는 방법은 본 문제의 해결 대상에서는 제외한다. 특정 파일에 대해서 보안기법을 별도로 설정해 두는 방법은 '액세스 제한 설정' 등의 기존의 기법이 있으나, 그 용도는 특정 파일의 보호이지 알려지지 않은 악성행위의 탐지는 아니다. 특정 파일을 건드린다고 해서 악성행위로 간주할 수는 없지만, 센서파일을 건드리는 경우는 센서기법의 구현상의 문제가 아니라면 악성행위일 가능성이 매우 크다. 이는 센서파일 기법이 False Positive의 문제를 대폭 해결함을 시사한다.

둘째, Spy Master가 트로이목마를 사용해서 사용자의 컴퓨터를 모니터링하고 있다가 특정 파일을 열어보는 경우이다. 이것은 사용자의 컴퓨터가 가지고 있는 정보가 무엇인지 알아보기 위해 파일들을 열어보는 경우이다. 그렇지 않으면 Spy Master는 사용자의 많은 데이터를 모두 탈취해야하기 때문이다.

Spy Master는 센서파일을 피하기 위해 센서파일을 열어보지는 않고 센서파일의 외적인 정보를 유심히 검토할 수도 있다. 이 경우 파일의 이름, 크기, 생성일자 등이 Spy Master가 참조할 수 있는 정보다. 하지만, 그러한 제한된 정보만으로는 센서파일을 피해 나갈 수 없다. 센서파일의 개수를 늘이면 Spy Master를 위한 행운의 확률은 크게 줄어든다. 또한, 이 경우는 Spy Master가 Covert Channel 등을 확보하고 있는 경우이므로 네트워크 침입탐지와 관련한 대응책을 마련할 수도 있다.

셋째, 센서파일 기술의 구현상의 문제로 정상적인 컴퓨터의 동작에 의해 센서파일이 건드려진 경우다. 이러한 경우는 다시 발생하지 않도록 시스템을 수정하여야 하지만, 정상적인 명령어나 프로그램 등이 센서파일을 액세스하는 경우는 많지 않으므로 리스트를 만들어 관리하면 이러한 문제는 해결될 수 있다. 악성행위자가 리스트된 명령어나 프로그램을 간접경로로 활용하지 못하도록 하는 기술이 추가될 수 있다.




(그림 1) 센서파일 사용 탐지기술 개요도

센서파일을 만들어서 기존의 파일 사이에 잘 분산하는 기법은 컴퓨터마다 다르며 각 경우에 맞게 별도로 설계할 수 있다. 이 방안의 구현은 센서파일을 만들어 두고 센서파일이 오픈되거나 네트워크로 전송될 때 포착해내는 것이므로 기술적으로 어려운 부분은 없으며 전반적인 개요는 (그림 1)에 나타나 있다. 본 연구에서는 센서파일기법의 구현가능성을 검증하기 위해 리눅스상에서 지정된 파일을 액세스하는 모든 경우를 탐지하는 기능을 구현하기도 하였고[1, 2], 기법의 구체성에 근거하여 발명특허를 출원하였다[3].

특정하게 마크된 파일이 네트워크 등으로 전송되는 경우에 이를 추적하는 Stego Tracing 등의 유사기법은 이미 알려져 있다[4]. 구현을 얼마나 세부적이고 정확하게 하느냐 하는 것은 탐지율을

결정짓는 관건이 되며, 탐지율을 높이기 위한 연구는 지속되어야겠지만 구현자체에는 기술적인 어려움은 없으며, 다음과 같은 면에서의 구현상 구체화되어야 할 부분 및 확장적용을 검토해 볼 수 있다.

### 2.1 악성행위 판별 기술

소스가 공개되지 않은 소프트웨어에 은닉되어 있는 트로이목마를 센서파일로써 탐지할 수 있도록 하는 개념을 지원하는 보안정책을 마련한다. 즉, 센서파일을 액세스할 수 있는 정상적인 명령어나 프로그램을 미리 구분해 냄으로써, 그 외에 센서파일을 액세스하는 모든 동작은 악성행위 임을 보장할 수 있는 여건을 조성한다. 또한, Deception 기법 등을 적용하여 트로이목마의 동작을 유도하는 방안과 기술을 추가하여 탐지효과를 증가시킬 수 있다.

### 2.2 사건의 재연에 관한 기술 개발

악성행위로 판단되는 동작 발생 시 동작발생 전후에 사용되었던 자원과 파일 등을 이용하여 동작의 재연을 통한 분석을 해야 하므로, 이러한 재연에 필요한 정보를 효율적으로 기록 관리하는 기술이 필요하다. 일정한 기간동안 수행된 프로세스의 요약정보 또는 사용된 자원을 로깅하는 기법은 기존의 OS에도 충분히 구현되어 있다. 이렇게 로그된 자료들을 기반으로 해서 센서파일이 오픈 되도록 한 프로세스와 실행코드의 위치를 찾아낼 수 있다. 악성행위로 판단되는 행위가 발생하면, 경고를 하고 안전하게 대응모드로 전환시키는 기법의 연구도 필요하다.

### 2.3 Worm의 행동탐지 및 전파방지 기술 개발 (Polymorphism I)

바이러스가 변형(polymorphism)을 만들어 백신의 제작을 어렵게 하는 방식을 역이용하여 악성행위 대응 프로그램의 여러 가지 변형을 배포 설치하여 웜이 전파되지 못하게 하는 기법을 생각할 수 있으며, 여기에 센서파일 기술을 적용할 수 있다. 센서파일 대신 센서주소를 메일 주소록에 주입하는 것이다. 센서주소를 생성하는 프로그램 모듈을 각각의 사용자 컴퓨터마다 서로 다르게 구성 배포함으로써 웜이 여러 가지의 센서주소 생성 정보를 해독하기 전에는 자신을 은닉하면서 전파될 수 없게 한다. 결과적으로 백신을 신속하게 만들고 광범위하게 배포하는 노력을 절감시킬 수 있다. 악성행위자가 다양한 센서주소 생성 모듈을 모두 해독하였다 하더라도 웜이 그 모든 경우를 피해나가게 할 수는 없다. 다수의 센서주소 정보를 분석하기 위해 웜의 사이즈가 커지면 결과적으로 웜이 탐지되는 것이 쉬워지기 때문이다.

### 2.4 센서파일 관리자 자체의 방어 (Polymorphism II)

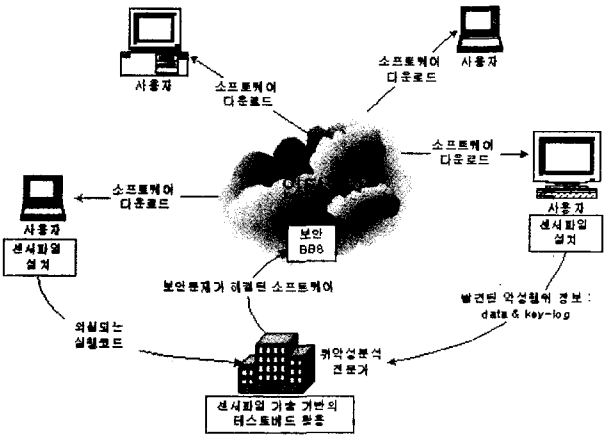
악성행위자는 센서파일을 관리하는 프로그램이 어떻게 센서파일을 만드는지를 알아내려 할 것이다. 이에 대응하기 위한 기법은 여러 가지를 제안할 수 있다. 이를테면, 사용자들이 저마다 다르게 센서파일을 생성할 수 있도록 하던지, 경우에 따라서는 사용자가 직접 센서파일의 이름을 생성해 줄 수도 있다. 주요한 데이터 파일에 대해서는 파일이름을 센서파일과 구분되지 않게 만들 수도 있고 그러한 경우 사용자는 별도의 메모를 하던지 센서파일 관리자 전용의 패스워드를 설정하여 필요시 데이터 파일과 센서파일을 구분해 낼 수도 있다. 즉, 센서파일의 크기, 이름, 생성날짜, 위치 등을 생성하는 프로그램 모듈을 악성행위자가 분석하더라도 사용자가 개별적으로 선택한 패스워드 등에 의해 파일이름 등이 예측 불가능하게 생성되도록 할 수 있다.

## 3. 센서파일 기술 기반의 분석 환경

알려지지 않은 악성행위를 발견하는 기존기술은 구현이 매우 어렵고 아직 실용화되지 않았다고 볼 수 있다. 본 연구는 특정 악성행위에만 적용되는 게 아닌 일반화된 기법이므로 실용성이 높다. 임의의 소프트웨어에 삽입되어 있는 악성행위들을 그 동작의 개시 즉시 탐지하며 악성행위와 정상행위 구분상의 오판율이 매우 적은 센서파일 기술을 활용함으로써 알려지지 않은 악성행위를 효과적으로 발견 및 분석을 할 수 있다. 이러한 발견을 통해 새로운 보안취약성 또는 악성기술을 파악할 수 있고, 약간의 응용을 통해 웜의 전파를 방지하는 데에 적용할 수도 있으며, 보안문제를 내포한 소프트웨어를 조기에 발견하고 개선하여 재배포하는 작업을 효과적으로 진행시킬 수도 있다.

이러한 센서파일 기술을 기반으로 악성행위를 판단하는 환경을 구축하면, 보안취약성을 찾아 악용하려는 악성코드 제작의 노력이 악성행위를 방어하는 노력에 비해 상대적으로 매우 크게되어, 컴퓨터시스템을 통한 정보탈취 의도를 무산시키는 여건을 조성할 수 있다. 악성코드 입장에서는 은밀히 설치되어있는 센서파일을 구분해 내기 어렵기 때문이며, 센서파일 기술은 알려지지 않은 악성행위를 개별적으로 대응하지 않고 은닉된 센서에 의해 일률적으로 탐지하기 때문이다. 결국 지속적으로 경우별로 대응하던 기술의 개발에 투자하던 비용과 노력을 절감하게 된다. 센서파일 기술은 Polymorphism만 잘 구현하면 더 이상 보완되거나 수정될 필요가 없다.

센서파일 기술 기반의 분석환경은 다음과 같이 구성할 수 있다. 전문가용 테스트베드 및 센서파일을 설치한 일반사용자의 시스템에 의해 악성행위가 탐지된다. 사용자의 시스템에서 악성행위가 탐지되면 관련정보가 보안 전문가에게 전달되어 센서파일 기술 기반의 테스트 베드에서 재연되면서 분석된다. 분석된 소프트웨어가 공개 소프트웨어인 경우에는 보안관련 BBS에 알려진다. 상용 소프트웨어인 경우에는 생산자와 협력하여 문제점을 해결한다 (그림 2).




(그림 2) 센서파일 기반의 분석 환경

## 4. 결 론

주어진 코드를 대상으로 알려지지 않은 악성행위를 찾아내는 기존의 기법으로는 정적 또는 동적 분석기법과 인공지능 기법 등이 있으나, 본질적으로는 예상범위를 벗어나는 문제점을 찾아내지는 못한다. 정적분석 기법이 소프트웨어를 분석하는데 있어 보다 효과적이라고 알려져 있지만, 실행코드를 대상으로 하는 경우에는 활용할 수 있는 정보가 제한되므로 적용이 어렵다. 실행코드를 분석이 가능한 상위 레벨로 표현한 후 분석하는 기법도 있으나, 적어도 메가급의 크기를 가지는 일반적인 소프트웨어를 분석하는 데에는 한계가 있다. 본 연구는 기존의 동적 분석이나 인공지능 기법처럼 특정한 악성기술을 대상으로 하여 각각에 대해 해결안을 제시하는 것이 아니며, 사용자의 데이터를 탈취하는 대부분의 동작을 대상으로 하므로 적용범위가 넓다. 악성코드가 예상 밖의 기술을 사

용하더라도 결국에는 사용자의 파일을 액세스하려하기 때문에 본 기술의 탐지범위 안에 들어오게 되는 것이다. 본 기술은 악성행위 제작기술의 발달에 별로 영향받지 않으며, 오판율이 매우 적게 악성행위를 탐지하는 기술로 발전시킬 수 있다.

## 참고문헌


[1] 조성제, 김준모, 김홍근, "A Trojan Horse Detection System Using Sensor Data", The 2002 International Conference on Optical Communications and Multimedia, November, 2002. (http://icocm.chosun.ac.kr)

[2] 장철연, 김근래, 조성제, 김준모, "센서 개념을 적용한 침입 탐지 시스템", 한국정보과학회 제29회 추계학술논문집, 2002년 10월. (http://kiss.or.kr/SubFrame.asp)

[3] 발명자: 김준모, 김홍근, 출원인: 한국정보보호진흥원, "센서 파일을 이용한 컴퓨터 감시 시스템 및 방법 (System and Method for monitoring a computer using sensor files)", 대한민국 특허청, 출원번호 10-2002-0056067, 2002년 9월.

[4] Computer, IEEE magazine p.16 : "Packet sniffers can also identify spyware by monitoring traffic and detecting when a PC starts sending data unrelated to user activities over the Internet. This can indicate that a spymaster is using invasive software to steal information."


## 저자약력


**김 준 모**

1989년 서울대학교 컴퓨터공학과(공학사)

1990년~1993년 공군 중앙전산소

1993년~1994년 DACOM

1996년~1997년 Mathematics, University of Toronto (학부과정)

2001년 Computer Science, University of Minnesota (공학박사)

2002년~현재 한국정보보호진흥원 선임연구원

관심분야: Approximations for NP-hard problems, Deterministic vs Non-Deterministic Problems, Church's Thesis, 실행프로그램 분석 기술 등

e-mail: jmkim@kisa.or.kr

**조 성 제**

1989년 서울대학교 컴퓨터공학과 (공학사)
1991년 서울대학교 컴퓨터공학과 (공학석사)
1996년 서울대학교 컴퓨터공학과 (공학박사)
1996년 9월~1997년 8월 서울대학교 컴퓨터신기술연구소 연구원
2001년 3월~2002년 2월 University of California, Irvine 객원연구원
1997년- 현재 단국대학교 정보컴퓨터학부 컴퓨터과학전공 조교수
관심분야 : 시스템소프트웨어, 실시간 시스템, 컴퓨터 보안, 분산시스템 등
e-mail: sjcho@dankook.ac.kr

**황 병 연**

1986년 서울대학교 컴퓨터공학과(공학사)
1989년 한국과학기술원 전산학과(공학석사)
1994년 한국과학기술원 전산학과(공학박사)
1999년 2월~2000년 2월 University of Minnesota Visiting Scholar
2002년- 현재 가톨릭대학교 정보통신원 원장
1994년- 현재 가톨릭대학교 컴퓨터정보공학부 교수
관심분야: 데이터베이스 시스템, XML 데이터베이스, 전자상거래, 지리정보시스템 등
e-mail: byhwang@catholic.ac.kr