情 報 保 護 學 會 誌
第17卷 第4號, 2007. 8

가상환경을 이용한 악성코드 탐지기술

서 정 택*

요 약

악성코드 탐지기술에 대한 연구는 최근에도 지속적으로 진행되고 있다. 특히, 에뮬레이터나 가상머신을 이용한 악성행위 탐지기술은 사용자 시스템에 악영향을 미치지 않는 독립적인 공간에서 코드의 실행이 가능하며, 빠른 초기화가 가능하다는 장점으로 인해 최근에 이슈가 되고 있다. 본 논문에서는 최근의 에뮬레이터나 가상머신을 이용한 악성행위 탐지기술의 연구동향을 분석하고, 관련 기술의 발전방향을 제시하고자 한다.

Ⅰ. 서 론

최근의 컴퓨터 자원에 대한 위협 요소는 스크립트를 이용한 단순 침입으로부터 다양한 기능의 악성코드를 이용한 컴퓨터의 오용에 이르기까지 공격 유형이 다양해지고 있으며, 피해의 규모도 증가하고 있다. 또한, 악성코드가 고전적 의미의 바이러스 및 해킹도구들의 특징들을 모두 가지는 복합기능형 악성코드로 발전하고 있어 기존의 백신 프로그램이나 침입탐지시스템에서 사용하는 탐지기술로는 효율적으로 대처하기 힘들다. 또한, 시스템에 이미 감염이 이루어졌거나, 악성행위가 수행된 이후의 탐지는 효과적이지 못하며, 이미 시스템이 망가진 상태일 수도 있다.

따라서 최근에는 클라이언트 시스템이 아닌 에뮬레이터(Emulator)나 가상 머신(Virtual Machine)에서 의심되는 파일들을 직접 실행해보며 악성행위가 포함되어 있는지 탐지하는 기술에 관심이 기울여지고 있다. 이들 기술들 중 가상머신을 이용하는 기술에서는 대상 파일을 가상머신에서 직접 실행하며, 악성행위를 탐지한다. 가상머신을 이용함으로써 사용자 시스템과는 독립적인 공간에서 코드의 실행이 가능하며, 사용자 시스템에는 어떠한 악영향도 미치지 않는다. 또한, 사용자 시스템에 악성코드가 설치되거나 시스템이 망가지게 되면 이를 재설치 하는데 시간이 많이 소요되게 되는데, 가상머신을 이용하면 빠르게 초기화가 가능하여 계속적으로 여러 개의 코드들을 실행해 볼 수 있다는 장점을 갖는다.

본 논문에서는 에뮬레이터와 가상머신을 이용한 악성행위 탐지기술에 대하여 소개하고, 관련 연구결과인 Columbia 대학의 연구결과, Norman의 Sandbox, Sunbelt의 CWSandbox 및 Ikarus S/W의 TTAnalyze와 실제 시스템을 이용하지만 목적이 비슷한 Litterbox에 대하여 각각의 기술을 분석하고, 관련 기술의 발전방향을 제시한다.

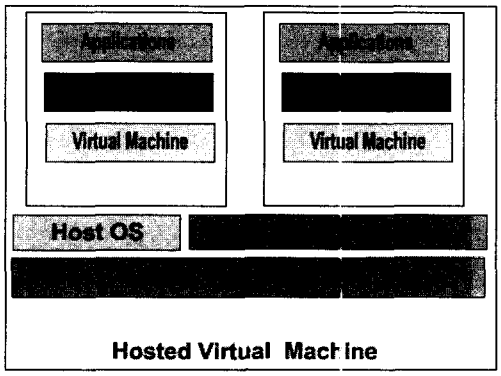
Ⅱ. 관련기술 동향분석

2.1. 에뮬레이터와 가상머신을 이용한 악성행위 탐지 기술

에뮬레이터를 이용한 악성행위 탐지 기술은 실제의 사용자 시스템에 별도의 하드웨어가 필요 없이 독립적인 컴퓨터 시스템에 대한 시뮬레이션을 수행한다는 개념을 이용한다. 의심스러운 파일은 시뮬레이션 하드에 위치시키고, 시뮬레이션 환경에서 동작시킨다. 시뮬레이션 환경 속에서 파일 감염, 파일 삭제, IRC 서버 연결, 이메일 송신 및 리스닝 포트 오픈까지 다양한 악성행위의 수행이 가능하다. 이러한 에뮬레이션 환경을 이용하여 대상 파일을 실행하며 탐지를 수행하는 기술이 에뮬레이터를 이용한 악성행위 탐지기술이다.

가상머신을 이용한 악성행위 탐지 기술은 사용자 시스템과 독립적인 가상머신 내에서 대상 파일을 실행함으로써, 사용자 시스템에는 악영향을 끼치지 않으며 코

* 한국전자통신연구원 부설 연구소 (seojt@etri.re.kr)



〔그림 1〕 가상머신 개념도

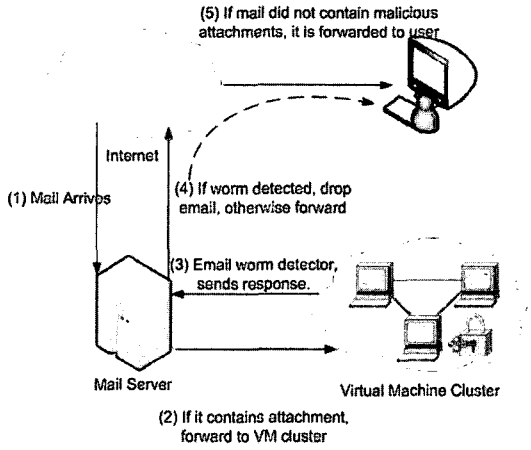
드를 실행해 볼 수 있다. 또한, 각 대상파일별로 별도의 가상머신 상에서 실행함으로써 악성코드가 실행되며 시스템에 악영향을 끼쳐도 다시 가상머신을 초기화시킴으로써 빠른 시간 내에 새로운 시스템에서 각각의 대상파일에 대한 실행 및 판단이 가능해 진다. 이는 실제 시스템을 구축하여 코드를 실행하며 악성행위 여부를 판단하는 시스템보다 시스템 복원 등에 소요되는 시간이 대폭 감소되므로 다량의 대상파일에 대한 악성행위 탐지가 용이해진다. 특히, VMware를 이용하면 하나의 시스템에 여러 개의 가상머신을 동작시킬 수 있다. 〔그림 1〕은 가상머신의 개념도이다.

일반적으로 가상머신을 이용한 악성코드 행위 분석 방법은 아래와 같다.

- 악성코드의 실행 전의 완전한 시스템의 상태 이미지와 실행 이후의 시스템 상태 이미지와의 비교를 통한 악성행위 탐지
- 디버거 등의 특별한 툴을 이용하여 악성코드가 실행되는 동안의 행위를 모니터링(시스템 콜 순서 분석을 통한 악성행위 탐지)
- 가상머신 상에서 코드를 실행하며 패턴을 이용한 오용탐지

2.2. Columbia 대학의 연구결과

Columbia 대학 Computer Science학과의 시스템보안연구실은 2005년도 싱가폴에서 열린 ISPEC 2005 학회에서 관련 논문을 발표하였는데, 최근에 이슈가 되고 있는 Zero-day 웜과 바이러스가 이메일을 통해서 들어오는 것에 대하여 대응하겠다는 취지의 연구로서, 메일서버에 도착한 메일 중에 첨부파일이 있는 메일에 한해



〔그림 2〕 Columbia 대학 연구결과물의 시스템 구성

서만 가상머신에서 실행하여 악성행위가 포함되어 있는지를 판단하는 방법론을 제안하고 있다.

〔그림 2〕는 동 연구에서 제안하는 시스템의 전체구조를 보여주고 있다. 메일이 도착하면, 첨부파일이 존재하는지 확인한 후 첨부파일이 존재하는 메일에서 첨부파일만 떼어내서 Virtual Machine Cluster로 보낸다. Virtual Machine Cluster에서는 호스트 기반의 침입탐지 엔진인 RAD(Registry Anomaly Detection)를 이용하여 대상 파일을 실행하며 악성행위 여부를 탐지한다. 악성행위로 판명된 첨부파일이 존재하는 메일은 사용자에게 전송하지 않고 메일서버에서 폐기처분시킨다.

2.2.1. 주요 접근법

대상 첨부파일을 사용자 시스템에서 직접 실행하는 것이 아니고, VMware 상에서 실행하여 탐지를 수행함으로써, 악성코드가 실행되어도 사용자 시스템에 직접적으로 악영향을 미치지 않는다. 이는 메일 서버에서 첨부파일이 존재하는 메일만을 분류하고, 대상 첨부파일을 떼어내어 Virtual Machine Cluster로 보내어 악성행위 여부를 탐지하고, 정상파일로 구분된 첨부파일에 대한 메일만 사용자에게 전송하기 때문이다.

VMware 상에서 동작하는 호스트 베이스 침입탐지 알고리즘으로는 RAD(Registry Anomaly Detection)를 사용한다. RAD는 메일 클라이언트 프로그램이 실행되며 동작하는 정상적인 행위들을 모델링하고 있다. 즉, Behavior-based 기법의 침입탐지 기법을 이용한다. 정

상적인 레지스트리 접속 정보들을 가지고 그 정상적인 레지스트리 접속 정보를 벗어나는 행위가 발견되면 악성 행위로 탐지한다.

각각의 VMware는 이미지 형태로 하나의 대상파일을 실행할 때마다 새롭게 초기화되고, 일부 어플리케이션 프로그램들도 설치 및 동작된다. 각각의 VMware가 클라이언트 시스템으로 동작하고, 하나의 대상파일을 하나의 VMware에서 실행하며, 탐지를 수행한다.

2.2.2. 가상머신

클라이언트 사용자 시스템에서 직접 대상 코드를 실행하는 것이 아니고 가상머신에서 코드를 실행한다. 즉, 클라이언트 사용자 시스템과는 분리된 독립된 환경에서 실행함으로써 대상코드가 악성행위를 수행하여 시스템에 악영향을 끼치는 것을 방지할 수 있다. 또한, 시스템에 직접 실행하며 탐지를 수행하는 도중에 시스템에 악성행위가 실시되어 시스템이 동작하지 못하는 상태가 되면, 이를 복구하기 위해 다시 운영체제부터 응용프로그램까지 설치해야 하는 번거로움이 발생할 수 있는데, 가상머신에서 실행함으로써 간단하게 새로운 이미지를 올리는 것으로 대체할 수 있다. Columbia 대학의 연구에서는 가상머신으로 VMware를 사용하였다.

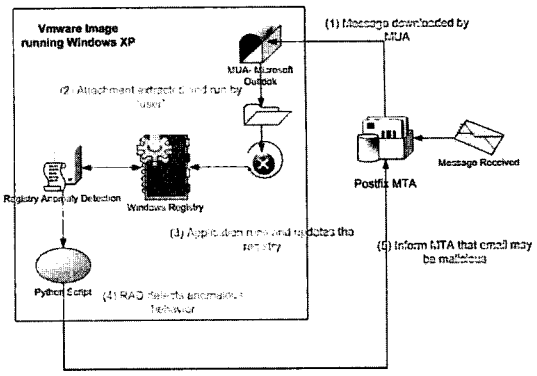
2.2.3. RAD(Registry Anomaly Detection)

RAD 탐지 알고리즘은 실시간으로 윈도우즈 레지스트리에 접속하는 정보를 확인하여 악성행위 여부를 판단하는 알고리즘이다. 이는 비정상 행위에 대한 높은 탐지율을 보이며, 계산에 필요하게 되는 오버헤드를 줄일 수 있는 강점을 갖는다. 본 연구에서 탐지에 사용한 Feature는 아래와 같다.

- 레지스트리에 접근한 프로세스 이름
- 레지스트리로 보내진 질의어의 타입
- 레지스트리에 접근한 키 값
- 레지스트리로부터 응답되는 값
- 레지스트리 접근에 대한 결과 값

2.2.4 시스템 구현 및 실험

1) MTA(Mail Transfer Agent)



〔그림 3〕 Columbia 대학 연구결과 시스템 구성도.

메일서버를 표현하는 것으로 악성 이메일을 분류 및 필터링을 수행하며, VMware 상에서 동작하는 Host-Based IDS Cluster와 통신하는 역할을 수행한다. 또한, 첨부파일이 있는 이메일을 분류하여 Virtual Machine cluster에 보내는 역할을 수행한다.

2) MUA(Mail User Agent)

부주의한 사용자가 이메일 첨부파일을 실행하는 역할을 수행한다. 또한 동 연구에서는 자세히 고려하지 않았지만 의심스러운 URL의 링크를 실행하는 역할도 수행한다. 동 연구에서는 Microsoft Outlook을 사용하고 있다. 〔그림 3〕은 Columbia 대학 연구 결과물의 전체 시스템 구성도이다.

2.2.5. 실험 결과

동 연구의 실험 결과를 살펴보면, 첨부파일을 다운로드 받아 VMware 상에서 실행하여 악성행위 여부를 탐지하고 MTA Message 큐를 업데이트하는데 평균적으로 28초의 시간이 소요되었다. 또한 하나의 VMware를 재시작하여 초기화시키는 데는 대략 평균 4초의 시간이 소요되었다. 또한, Columbia 대학의 일반 이메일을 대상으로 고려했을 때 50,000개의 이메일을 받았을 경우 그중 8%에 해당하는 이메일에 첨부파일이 존재하였고, 하나의 첨부파일에 대해 실행 및 탐지에 소요되는 시간이 대략 30초였다. 결과적으로 하나의 가상머신에서 하루 3,000개의 이메일이 처리 가능하다는 실험결과가 얻을 수 있다.

2.3. Norman의 Sandbox

Norman Sandbox는 Norman의 백신 제품인 Norman Virus Control의 내부 기능으로 사용되다가 최근에는 별도의 제품인 Sandbox Analyzer, Sandbox Analyzer Pro, Sandbox Online Analyzer 및 Sandbox Reporter로 개발하여 판매되고 있다. Norman Virus Control의 주요 기능은 실시간 바이러스 검사 및 치료 기능, 이메일 보호 기능, 수동 바이러스 검사 및 치료 기능, Norman Sandbox 기술로 알려지지 않은 악성 프로그램으로부터 시스템을 보호해주는 기능 등이다. 실시간 바이러스 검사 및 치료는 실행파일 또는 인터넷 실행 시 자동으로 검사 및 치료 기능을 수행하며, 이메일의 첨부파일을 다운받기 전에 검사하여 원천적으로 바이러스를 차단해 준다. 또한 Norman Sandbox 기술을 제공하여 알려지지 않는 바이러스, 웜, 트로이목마와 같은 악성코드를 감지하고, 악의적인 활동을 방지하여 시스템이 감염되지 않도록 한다. 또한 시스템 트레이의 아이콘을 통하여 설정 변경 및 바이러스 검사, 인터넷 업데이트를 수행할 수 있다.

이와 같이 Norman Antivirus Solution에서는 기존의 패턴을 이용한 바이러스 검사와 에뮬레이터 형태의 Sandbox를 이용하여 알려지지 않은 악성코드에 대한 탐지를 동시에 수행하게 된다. Sandbox는 실제 어린 아이들이 놀다가 다치는 것을 방지하기 위한 방안으로 모래를 가득채운 상자의 의미가 있는데, Norman에서 개발한 Sandbox 역시 바이러스에 의해 실제 컴퓨터 시스템이 공격 받는 것을 방지하기 위한 것으로 가상공간 에뮬레이터의 역할을 한다.

Norman Sandbox는 Winsok, Kernel 및 MPR과 같은 윈도우즈 기능과 호환성 있게 동작하며, 네트워크 및 인터넷 기능인 HTTP, FTP, SMTP, DNS, IRC 및 P2P 기능도 지원한다. 이는 실제 사용자 시스템에 별도의 하드웨어가 필요 없이 독립적인 컴퓨터 시스템에 대한 시뮬레이션을 수행한다는 의미이다.

의심스러운 파일은 시뮬레이션 하드에 두어 시뮬레이션 환경에서 동작하게 한다. 시뮬레이션 환경에서 해당 파일은 다른 파일을 감염시키고, 중요 파일을 삭제하고, 네트워크 너머의 시스템에 자신을 복사할 수도 있으며, IRC 서버와 연결하고, 이메일을 보내기도 하고, 리스닝 포트를 오픈하는 것까지 모두 가능하다.

Norman Sandbox가 파일을 실행하여 의심스러운 행위를 찾아내면, 악성행위를 아래의 카테고리로 분류할 수 있다.

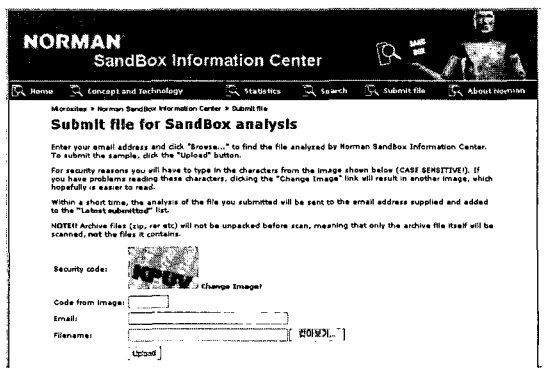
- W32/Malware
- W32/EMailWorm
- W32/NetworkWorm
- W32/BackDoor
- W32/P2PWorm
- W32/FileInfector
- W32/Dialler
- W32/Downloader
- W32/Spyware

Norman Sandbox는 아래와 같이 다양한 기능을 제공한다.

- 3,500개 이상의 API들에 대한 에뮬레이션 기능을 제공
- 의심스러운 프로그램이 여러 개의 독립적인 쓰레드를 이용하여 독립적인 기능으로 수행 가능
- 원격 프로세스에 대한 쓰레드 인젝션 기능 제공
- 이메일 주소록 정보 수집에 대한 탐지 기능 제공
- P2P, POP3, DNS, IRC, Web 등 다양한 네트워크 서비스 제공
- Instant Messaging 통신 기능 제공

Norman Sandbox는 일반에게 공개되어 있으며 〔그림 4〕와 같이 해당 사이트에 접속해서 악성코드로 의심되는 파일에 대한 분석을 간단히 의뢰할 수 있다. 분석을 의뢰할 때는 분석을 의뢰한 파일에 대한 결과를 받아 볼 이메일 주소와 의뢰할 파일을 첨부하여 업로드하면 된다.

분석할 대상 파일을 보내면 얼마 뒤 이메일로 분석 결

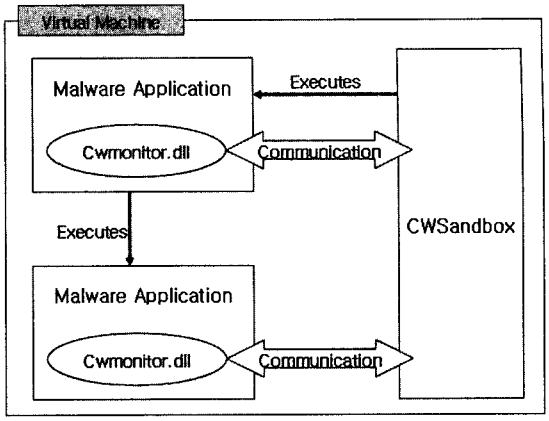


〔그림 4〕 Norman Sandbox에 분석을 의뢰하는 화면

```
D:VIRUSMYTEST.EX_ : W32/Backdoor
===> Sandbox output:
[General information]
***IMPORTANT: PLEASE SEND THE SCANNED FILE
TO: ANALYSIS@NORMAN.NO -
REMEMBER TO ENCRYPT IT (E.G. ZIP WITH
PASSWORD)**.
* Display message box (sample) : sample, te amo!.
* Display message box (KERN32) : KERN32, te amo!.
* File length: 58368 bytes.
* MD5 hash: 60a8d2e41147f48364e1eb3729ac53fb.

[Changes to filesystem] -> 파일의 변화
* Deletes file
C:WINDOWSSYSTEM32kern32.exe.
* Creates file
C:WINDOWSSYSTEM32kern32.exe.

[Changes to registry] -> 레지스트리 변화
* Creates key "HKLMSoftwareMicrosoftWindows
 CurrentVersionRunOnce".
* Sets value "kernel32"="C:WINDOWS
 SYSTEM32kern32.exe -sys" in key
"HKLMSoftwareMicrosoftWindowsCurrentVersionRunOnc
e".

[Changes to system settings] -> 시스템 설정 변화
* Creates WindowsHook monitoring keyboard activity.

[Network services] -> 네트워크 접속 정보
* Connects to "xxx.xxx.xxx.xxx" on port 6667 (TCP).
* Connects to IRC server.
* IRC: Uses nickname CurrentUser[FRK][19].

[Process/window information] -> 기타 정보
* Creates a mutex ZZM9H9YY.
* Creates a mutex SrVFrK.
```

〔그림 5〕 Norman Sandbox 분석 결과

과물을 받아볼 수 있다. [그림 5]는 Norman Sandbox Analyzer에 의해서 분석된 결과의 예이다. 그 내용을 보면 자동화된 분석 결과로 생성되는 파일, 레지스트리의 변화, 시스템 설정 변화, 네트워크 접속 정보 및 기타 이상 증상들을 리포팅하고 있음을 알 수 있다. 또한 이 파일이 알려진 바이러스인지, 알려지지 않은 바이러스인지의 여부도 체크하여 그 결과를 알려준다. 물론 이는 일반 사용자들에게 공개된 버전이라 분석 정보는 미약하지만 실제 내부에서는 개발자 버전이 존재하며 공개된 버전보다 더 상세한 분석 정보를 제공하고 있다고 한다.



〔그림 6〕 CWSandbox 구성

2.4. Sunbelt의 CWSandbox

CWSandbox는 독일 Mennheim 대학의 랩에서 연구 및 개발을 수행하였고 2006년 말부터 세계적인 소프트웨어 업체인 Sunbelt에서 판매하고 있는 제품이다. CWSandbox는 윈도우즈 시스템 리소스로 파일 시스템, 레지스트리 및 응용프로그램들 간의 API Call 정보를 로그로 남기는 역할을 수행한다. CWSandbox도 다른 시스템들과 마찬가지로 VMware 상에서 실행하므로 사용자 시스템에는 직접적인 악영향을 미치지 않도록 동작한다.

〔그림 6〕은 CWSandbox의 구성을 나타낸다. CWSandbox는 실행파일이 실행되면 해당 파일의 쓰레드로 모니터링 모듈인 CWMonitor.dll 파일을 인젝션시킨다. 인젝션 이 후 해당 실행파일이 실행되는 모든 절차를 추적하여 해당 정보를 모두 CWSandbox로 전달한다.

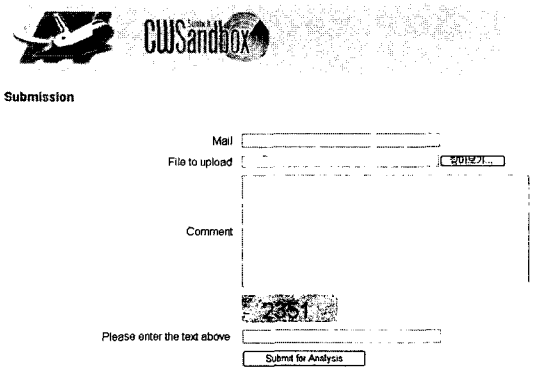
또한, 실행파일이 실행되는 과정에서 발생하는 윈도우즈 API 호출 정보를 API 후킹 기법을 이용하여 가로챈 후 CWSandbox로 전달한다. 전달된 윈도우즈 API 호출은 CWSandbox를 통해 윈도우즈 커널로 전달하는 정상적인 절차를 수행하게 되고 이때 호출되는 윈도우즈 API 역시 호출한 해당 실행파일로 바로 전달하지 않고 CWSandbox를 거쳐서 전달한다.

여기서 호출되는 윈도우즈 API가 CWSandbox를 통해 실행 파일로 전달된 후 다시 그 결과 값이 윈도우즈 운영체제로 바로 전달되지 않고 CWSandbox를 통해 전혀 다른 결과 값을 윈도우즈 운영체제에 전달한다. 따라서 실행파일이 악성코드로 악성행위를 수행하고자 할 때 CWSandbox에 의해 걸러지게 되므로 실제적인 사

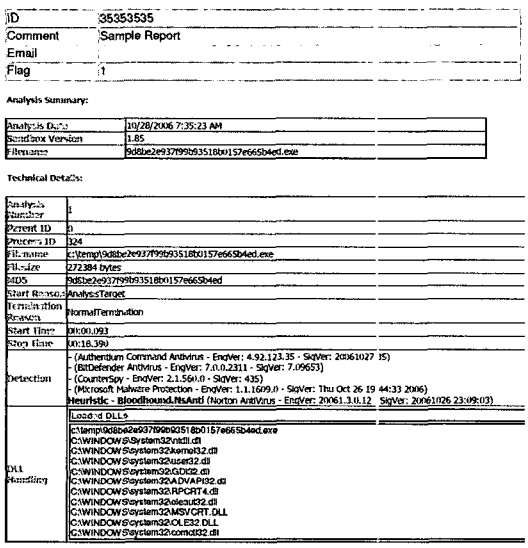
용자 시스템으로의 감염은 발생하지 않게 된다.

CWSandbox도 Norman Sandbox와 마찬가지로 일반 사용자들이 악성으로 의심되는 바이너리파일을 웹서버에 자신의 이메일 주소와 함께 업로드하면, 자동으로 대상 파일에 대한 분석이 수행되고 수행결과가 사용자의 이메일 주소로 전송된다. [그림 7]은 CWSandbox에 사용자가 의심스러운 바이너리 파일을 업로드하는 화면이다.

[그림 8]은 CWSandbox를 이용하여 분석한 결과 레포팅 형태이다. 분석결과는 HTML, XML 및 TXT 형태로 제공되는데, 아래 그림들은 HTML 형태의 분석결과 리포팅 예이다. 분석 결과서에는 파일사이즈, 백신 프로그램에서의 탐지 결과 및 사용한 API Call 정보와 WinSock 및 패킷들의 정보가 표시된다. 예를 들면,



[그림 7] CWSandbox에 분석을 의뢰하는 화면



[그림 8] CWSandbox Reporting 형태

[표 1] CWSandbox로부터 얻을 수 있는 주요행위

대분류	상세분류	설 명
DLL 로드	DLL 로드	어떤 DLL을 로드하는가?
파일 접근	새로운 파일 생성	어떤 파일을 생성하는가?
	파일 내용 읽기	어떤 파일의 내용을 읽는가?
	파일접근 순서	파일에 대한 접근을 시간순서대로 나열
	파일 복사	어떤 파일을 어디로 복사하는가?
레지스트리 접근	읽기	어떤 Key 값을 읽는가?
	쓰기	어떤 위치에 어떤 값을 쓰는가?
프로세스 접근	프로세스 생성	어떤 프로세스를 생성하는가?
	프로세스 읽기	어떤 프로세스 영역의 정보를 읽는가?
	프로세스 쓰기	어떤 프로세스 영역 내에 어떤 정보를 쓰는가?
	쓰레드 실행	어떤 프로세스 영역 내에 어떤 쓰레드를 실행하는가?
서비스 접근	서비스 등록	어떤 서비스를 등록하는가?
	서비스 실행	어떤 서비스를 실행하는가?
	서비스 중지	어떤 서비스를 중지하는가?
	서비스 제거	어떤 서비스를 제거하는가?
시스템 객체 접근	시스템 객체 생성	어떤 시스템 객체를 생성하는가?
	시스템 객체로부터 읽기	어떤 시스템 객체로부터 정보를 읽는가?
	시스템 객체로 쓰기	어떤 시스템 객체로 정보를 쓰는가?
네트워크 접근	연결 유형	외부 시스템으로 접근하는지? 외부로부터 접근을 기다리는지?
	외부 시스템 주소	연결된 외부시스템의 IP 주소 및 도메인은?
	프로토콜	어떠한 프로토콜 사용하는지? (TCP, UDP)
	연결포트	사용 포트번호는 몇 번인가?

DLL-Handling, Filesystem, Registry, Process Management, Service Management, System Info, Threads, Virtual Memory 등과 관련한 API Call 정보들이 로그로 남는다. [표 1]은 CWSandbox로부터 얻을 수 있는 주요행위를 정리한 표이다.

2.5. Ikarus Software의 TTAnalyze

다른 접근방법으로는 Sandbox와 유사한 TTAnalyze가 있다. TTAnalyze는 2006년 European Inst. for Computer Antivirus Research (EICAR)에 "TTanalyze: A Tool for Analyzing Malware" 제목의 논문으로 발표되었다. TTAnalyze는 PC 에뮬레이터 QEMU2를 사용한다. QEMU는 GNU GPL을 따르는 오픈소스 프로세서 에뮬레이터 프로젝트이다. 가상머신이 그 위에서 동작하는 프로그램의 코드를 인터프리터 방식처럼 실시간으로 하나씩 번역해서 실행하며, 실제의 CPU에서 실행되는 것처럼 돌려주는 방식인데 반해서, QEMU는 "a fast and portable dynamic translator"라는 프로젝트 제목처럼 변환기(Translator)가 대상 CPU 명령어를 실제 호스트의 CPU 명령어로 동적으로 바꾸어 실행하게 된다. TTAnalyze는 악성코드를 분석하여 일반 정보, 파일 정보, 레지스트리 정보, 서비스 정보, 프로세스 정보 및 네트워크 정보를 제공한다.

- 일반 정보 : TTAnalyze 수행에 대한 정보를 포함하여, 악성코드에 대한 일반 정보(분석 수행시간, 파일크기 등)
- 파일 정보 : 악성코드의 파일 관련 행위에 대한 정보(파일 생성, 수정 등)
- 레지스트리 정보 : 악성코드에 의해서 읽기, 생성 및 수정된 모든 레지스트리 정보와 값
- 서비스 정보 : 악성코드와 Windows Service Manager간의 모든 활동에 대한 정보(서비스 시작 및 종료 등)
- 프로세스 정보 : 프로세스 생성, 종료 및 통신에 관련된 정보
- 네트워크 정보 : 보내고 받은 모든 네트워크 트래픽에 대한 로그

Norman Sandbox와 CWSandbox가 악성코드를 자동적으로 분석 및 탐지하는데 비해, TTAnalyze는 악성코드의 행위에 대한 자세한 정보를 제공하는 것에서 그친다. 즉, TTAnalyze는 자동화된 악성코드 탐지가 목적이 아니라 사람에 의해서 수동적으로 분석하는데 많은 시간과 노력이 드는 것을 자동화하여 악성코드가 어떠한 행위를 하는지를 분석하는데 그 목적을 두고 있다.

2.6. Litterbox

Litterbox는 에뮬레이터나 가상머신을 사용하지 않고, 실제 윈도우즈 시스템에서 악성코드를 실행하여 분석한다. 해당 악성코드는 윈도우즈에서 60초간 실행 후, 시스템을 리눅스 이미지로 강제로 리부팅한다. 리눅스에서 원래의 윈도우즈 시스템을 마운트하여 윈도우즈 레지스트리와 파일리스트를 추출한다. 즉, 악성코드의 실행으로 인해서 변화된 시스템 정보를 추출하는 것이다. Litterbox는 악성코드의 네트워크 분석에 초점을 맞추고 있다. 이를 위해서 악성코드가 실행되는 동안에 모든 IRC 요청에 대해서 응답할 수 있도록 IRC 서버를 사용한 가상 인터넷 환경을 구성한다. Litterbox에서는 악성코드가 실행되는 동안에 발생하는 모든 네트워크

[표 2] 관련 기술별 특징 비교

	Columbia 대학 발표논문	Norman Sandbox	CWSandbox	TTAnalyze	Litterbox
사용하는 방식	가상머신	에뮬레이터	가상머신	에뮬레이터	실제 시스템
악성행위 탐지 가능	O	O	O	×	×
악성행위 탐지 기법	RAD 알고리즘을 이용한Anomaly 탐지	시그너처 기반의 악성코드 탐지 및 Sandbox를 이용한 행위 분석 리포팅	시그너처 기반의 악성코드 탐지 및 Sandbox를 이용한 행위 분석 리포팅	×	×
웹을 통하여 사용자가 의심파일을 업로드 가능	×	O	O	×	×
메일을 이용한 레포팅 파일의 제공 여부	×	O	O	×	×

패킷을 캡쳐하여 분석한다. 이러한 접근은 악성코드가 특정한 외부 서버에 연결되는 경우에도 항상 성공적인 IRC 연결을 지원하기 때문에 악성코드의 네트워크 활동을 분석할 수 있는 장점을 가진다. 그러나 Litterbox는 감염된 시스템에서 악성코드의 프로세스 생성과 같은 동적인 활동에 대해서 모니터링을 할 수 없으며 단지 한 순간의 스냅샷(snapshot) 정보만을 얻는다는 한계를 가진다.

Ⅲ. 관련기술 비교분석 및 전망

〔표 2〕는 Ⅱ장에서 분석한 각 기술들에 대하여 기능 및 특징을 비교한 표이다.

본 논문에서 소개하고 분석한 에뮬레이터와 가상머신을 이용한 악성행위 탐지기술은 현재 가장 필요하고 중요한 기술임은 분명하다. 하지만 현재 개발된 시스템들의 내부를 들여다보면 아직도 기술적으로 미흡한 부분이 많으며, 극복해야 할 문제점들이 있다.

• Columbia 대학의 연구결과
- 이메일 첨부파일을 대상으로 실험을 수행한 수준이며, 현재 완전 실용화 단계까지 진행되지는 못하고 있다.
- RAD(Registry Anomaly Detection) 탐지 알고리즘을 이용하여 비정상행위에 대한 탐지를 수행하고 있으나, 이 기법이 호스트에서 발생하는 모든 악성행위에 대하여 탐지가 가능하고 오탐율이 아주 낮다고 보기에는 어려움이 있다.
- 사용자의 행위 및 외부 네트워크로부터의 통신에 대한 행위가 포함되는 악성코드 분석에는 한계가 있다.

• Norman Sandbox
- 실제 악성행위에 대한 탐지 기법으로는 패턴을 이용한 탐지기법에 의존도가 높으며, 코드 수행에 대한 분석 결과를 뽑아내는 것에 그치고 있다. 따라서 분석 결과를 사용자가 참조하여 결정을 해야 한다.
- 안티 디버깅 기법이 포함되어 있는 악성코드는 분석이 불가하다.
- 비주얼베이직 및 델파이로 작성된 코드는 분석이 불가하다.
- Rightweight 한 비정상행위 탐지알고리즘을 적용할 경우 효과적일 것으로 판단된다.
- 사용자의 행위 및 외부 네트워크로부터의 통신에 대한 행위가 포함되는 악성코드 분석에는 한계가 있다.

• Sunbelt CWSandbox
- 실제 악성행위에 대한 탐지 기법으로는 패턴을 이용한 탐지기법에 의존도가 높으며, 코드 수행에 대한 분석 결과를 뽑아내는 것에 그치고 있다. 따라서, 분석 결과를 사용자가 참조하여 결정을 해야 한다.
- Rightweight 한 비정상행위 탐지알고리즘을 적용할 경우 효과적일 것으로 판단된다.
- 사용자의 행위 및 외부 네트워크로부터의 통신에 대한 행위가 포함되는 악성코드 분석에는 한계가 있다.

• TTAanlyze
- 악성행위에 대한 탐지는 수행하지 않으며, 단순히 코드 수행에 대한 분석 결과만 뽑아내는 기능을 수행한다.
- 사용자의 행위 및 외부 네트워크로부터의 통신에 대한 행위가 포함되는 악성코드 분석에는 한계가 있다.

• Litterbox
- 에뮬레이터나 가상머신을 이용하여 사용자 시스템으로부터 독립적으로 동작시키지 않고, 실제 시스템에 코드를 실행한다.
- 시간에 따른 악성코드의 동적 행위 및 실행 후 60초 이후에 동작하는 행위에 대한 결과를 얻지 못한다.
- 사용자의 행위 및 IRC를 제외한 외부 네트워크로부터의 통신에 대한 행위가 포함되는 악성코드 분석에는 한계가 있다.

Ⅳ. 결 론

본 논문에서는 최근 이슈가 되고 있는 에뮬레이터 및 가상머신을 이용한 악성행위 탐지기술에 대한 연구동향

을 분석하였다. 이들 기술들은 사용자 시스템과 독립적인 공간에서 코드를 실행하여 사용자 시스템에 악영향을 미치지 않으며, 빠르게 초기화가 가능하다는 장점을 갖는다.

이들 기술들은 가상 시스템에서 악성코드를 실행하여 실제 시스템에는 어떠한 악영향도 미치지 않겠다는 동일한 목적을 일부 달성하고 있지만 그 내부적인 기술과 실용성 측면을 고려해 보았을 때, 아직도 연구되어야 할 부분이 많이 존재한다. 특히 이들 기술들을 내부적으로 보면 패턴을 이용한 탐지를 수행하고 있으며, 해당 코드가 수행되는 과정에서 의심되는 행위들에 대한 분석 결과를 뽑아내는 것에 그치고 있으므로 알려지지 않은 비정상행위에 대한 탐지에는 한계가 있다. 또한 사용자의 행위나 외부 네트워크로부터의 통신에 의하여 동작하는 악성코드에 대해서는 실행 및 분석에 한계가 있다. 따라서 향후에는 이러한 한계를 극복하고 해결해 나갈 수 있는 기술에 대한 연구개발이 필요하다. 또한 이러한 기술 개발은 계속해서 증가하고 고도화되고 있는 악성코드에 대처하는데 효과적일 수 있다.

참고문헌

[1] Rebecca Bacel and Peter Mell, "Intrusion Detection Systems", NIST, 2003.

[2] Carl Endorf, Eugene Schultz, Jim Mellander, "Intrusion Detection & Prevention", McGraw-Hill, 2004.

[3] H. Debar, M. Dacie, and A. Wepsi, "A Revised Taxonomy for Intrusion- Detection Systems", IBM Report, 1999.

[4] F.Apap, A. Honnig, S.Hershkop, E.Eskin, and S.Stolfo. Detecting malicious software by monitoring anomalous windows registry accesses. Proceedings of the Fifth International Symposium on Recent Advances in Intrusion Detection (RAID 2002), 2002.

[5] Stelios Sidiroglou, John Ioannidis, Angelos D. Keromytis, and Salvatore J. Stolfo. An Email Worm Vaccine Architecture. Proceeding of the First Information Security Practice and Experience(ISPEC 2005), 2005.

[6] Apap, F., Honkg, A., Hershkop, S., Eskin, E., Stolfo, S.J : Detecting Malicious Software by Monitoring Anomalous Windows Registry Accesses. In: Proceedings of the 5th International Symposium on Recent Advances in Intrusion Detection(RAID). 2002.

[7] Carsten Willems, Thorsten Holz, and Felix Freiling, : Toward Automated Dynamic Malware Analysis Using CWSandbox. IEEE Security & Privacy. 2007.

[8] Norman Sandbox, http://www.norman.com/Product/Sandbox-products/en

[9] CWSandbox, http://www.cwsandbox.org

[10] Sunbelt CWSandbox, http://www.sunbelt-software. com/Developer/Sunbelt-CWSandbox/

[11] TTAnalyze, http://www.ikarus.at/

[12] TTAnalyze, http://fabrice.bellard.free.fr/qemu/

[13] TTAnalyze, http://www.tuwien.ac.at/

〈著者紹介〉

서 정 택 (Jungtaek Seo)
종신회원

1999년 2월 : 국립충주대학교 컴퓨터공학과 졸업

2001년 2월 : 아주대학교 컴퓨터공학과 석사

2006년 2월 : 고려대학교 정보보호대학원 공학박사

2000년 11월~현재 : 한국전자통신연구원 부설 연구소 선임연구원