The Journal of The Institute of Internet, Broadcasting and Communication (IIBC)
Vol. 15, No. 2, pp.1-7, Apr. 30, 2015. pISSN 2289-0238, eISSN 2289-0246

http://dx.doi.org/10.7236/JIIBC.2015.15.2.1
JIIBC 2015-2-1

모바일 가상화기반의 악성코드 행위분석

Malware Behavior Analysis based on Mobile Virtualization

김장일*, 이희석**, 정용규***

Jang-Il Kim*, Hee-Seok Lee, Yong-Gyu Jung*****

요 약 최근 전 세계적으로 스마트폰의 사용이 급증하고 있으며, 국내의 경우 스마트폰 가입자 수는 약 2400만명으로 전체 이통사의 가입자중 47.7%가 스마트폰을 사용하고 있다. 스마트폰의 경우 보안에 대해 취약점을 가지고 있으며, 스마트폰을 이용한 보안관련 사고피해가 해가 갈수록 증가하고 있다. 그러나 기존의 방식은 사전 대책이 아닌 대부분 사후대책으로써 전문가의 경우를 제외하면 피해를 입은 뒤에 그 피해가 발생한 악성코드의 분석이 이루어지고 있다. 이에 따라 본 논문에서는 가상화 기술을 적용한 모바일 기반의 악성코드분석 시스템을 구현하고 이를 통하여 행위분석을 하도록 설계한다. 가상화는 컴퓨터 리소스의 물리적인 특징을 추상화하여 게스트에게 논리적인 리소스를 제공하는 기술이다. 이러한 가상화 기술은 클라우드 컴퓨팅 서비스와 접목시켜 서버, 네트워크, 스토리지등 컴퓨팅 자원을 탄력적으로 제공함으로 자원의 효율성을 높이고 있다. 아울러 사용자 관점에서 사전에 보안사고를 대비할 수 있는 시스템을 제시한다.

Abstract As recent smartphone is used around the world, all of the subscribers of the mobile communication is up to 47.7% about 24 million people. Smartphone has a vulnerability to security, and security-related incidents are increased in damage with the smartphone. However, precautions have been made, rather than analysis of the infection of most of the damage occurs after the damaged except for the case of the expert by way of conventional post-countermeasure. In this paper, we implement a mobile-based malware analysis systems apply a virtualization technology. It is designed to analyze the behavior through it. Virtualization is a technique that provides a logical resources to the guest by abstracting the physical characteristics of computing resources. The virtualization technology can improve the efficiency of resources by integrating with cloud computing services to servers, networks, storage, and computing resources to provide a flexible. In addition, we propose a system that can be prepared in advance to buy a security from a user perspective.

Key Words : Radiation level, Clustering, SimpleKMeans Algorithm, EM Algorithm, Isotope

I. 서 론

최근 전 세계적으로 스마트폰의 사용이 급증하고 있다. 2012년 2월 기준으로 미국의 이동통신 가입자 중 50%가 스마트폰을 사용하고 있다. 국내에서도 같은 시기를 기준으로 확인된 국내 스마트폰 가입자 수는 24,794,337명으로 전체 이동 통신가입자의 47.7%가 스마트폰을 사용하고 있으며, 2012년 3월 한 달간 신규 가입

*정회원, 을지대학교 대학원 의료IT마케팅학과
**정회원, 을지대학교 대학원 의료IT마케팅학과
***종신회원, 을지대학교 의료IT마케팅학과(교신저자)
접수일자 : 2015년 2월 27일, 수정완료 : 2015년 3월 28일
게재확정일자 : 2015년 4월 10일

Received: 27 February, 2015 / Revised: 28 March, 2015 / Accepted: 10 April, 2015
***Corresponding Author: ygjung@eulji.ac.kr
Dept. of Medical IT and Marketing, Korea

자 수도 923,860명에 이른다.[1] 이러한 스마트폰 플랫폼 중 하나인 안드로이드는 빠른 시장 확장을 위하여 '플랫폼 소스 공개' 및 '다양한 어플리케이션 유포 경로'의 특징을 갖는 개방형 구조를 선택하였으며 그 결과 50%가 넘는 점유율을 획득했다[2][3]. 그러나 개방형 구조는 악성코드의 작성 및 유포를 쉽게 하는 특징 또한 가져, 안드로이드에게 악성코드 취약이라는 오명 역시 부여하였다. 실제 모바일 악성코드의 대부분이 안드로이드를 공격 대상으로 삼고 있으며 그 수는 급속히 증가하고 있다.[4]

기존에는 악성코트 탐지, 침입시도 탐지 등의 특정 기능에 포커스 되어 있었다면, APT 솔루션들은 기존 개별 보안 솔루션들의 통합버전이라 할 수 있다. 대표적인 차별 점으로는 '동적 분석'과 '상관관계 기반의 위협정보 검증'을 강화했다는 점을 들 수 있는데, 최근 실시간으로 네트워크상에 오가는 파일에 대한 동적 분석 기능을 더욱 강화해 나가고 있으며, IP 및 도메인 등에 대한 위험 여부를 판단할 때 평판기반으로 축적되어 있는 데이터와의 비교를 통해 위험성을 판단해나가는 것이 특징이다.

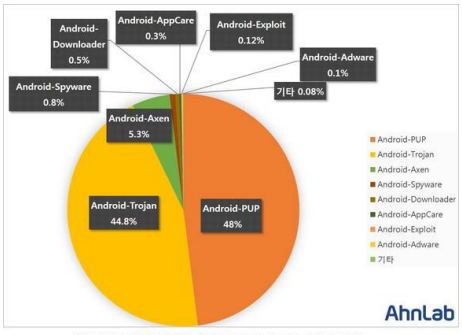

그림 1. 악성코드 접수 통계
Fig. 1. Malware Receipt Statistics

안드로이드폰 사용자를 노리는 악성코드가 폭증하고 있다. 애드웨어 수준의 악성 앱(PUP)가 20만여 건으로 전체의 48%를 차지했다. 이어서 사용자의 스마트폰에 숨어 정보유출, 과금 등 악성 행위를 하는 트로이목마 악성코드의 개수가 19만여 개로 전체의 44.8%를 차지했다. 이 두 가지 악성코드 진단 건수가 전체의 약 93%를 차지했으며, 이 뒤를 이어 사용자 몰래 정보를 수집하는 스파이웨어, 다른 악성코드를 추가로 설치하는 다운로더, 향후 악의적인 목적으로 활용될 수 있는 앱케어 등이 소량씩 발견됐다.[11][12]

한편, 사이버 공격의 주요 시작점이 되는 직원 PC, 웹서버 등 엔드포인트에 대한 모니터링 필요성이 커지면서 엔드포인트 대응 솔루션들이 부각되고 있다. 엔드포인트에서 전개되는 공격과정을 실시간 기록 및 분석해야 되기 때문에 해커가 인지하지 못하도록 Stealth 방법을 사용하면서 악성행위를 탐지해 내는 기술이 필요하다.[9] 하지만 모바일에서의 악성코드를 분석하는 방식은 가상화 기술을 활용하지 않으며 행동분석의 경우 오프라인에서 샘플데이터를 활용하여 분석을 실행하며, 방지대책으로는 위변조 탐지기법과 무결성 검증 기법을 활용한 방지를 행하고 있다.[5] 사실상 현행의 기법들은 위험성을 내포하고 있으며, 특히 무결성 검증 기법은 로직 자체에 위변조가 일어날 가능성을 내포하고 있다. 행동 분석의 경우에도 오프라인에서만 행하고 있으며 이는 사용자들에게 직접적인 피해가 발생했을 때에 그 스마트폰에서 샘플데이터를 채취해서 분석하거나, 보안기관에서 획득한 악성코드를 오프라인 PC에서 분석함을 의미한다. 이는 결과적으로는 사용자들에게 피해를 주는 악성코드들은 문자에 포함된 URL을 클릭하거나, 어플리케이션을 다운받았을 때 발생하는 악성코드의 행동에 의해 사용자들이 피해가 이미 발생한 후에 대처를 할 수 있다는 단점을 내포하고 있다. 또한 모바일 기기에서 사용 중인 대다수의 백신도 데이터베이스에 남아있는 악성코드를 제외하고 새로운 악성코드가 등장했을 때에는 감지를 하지 못한다는 최대 단점을 지니고 있기 때문에 피해가 속출하고 있는 실정이다.

본 논문에서는 기존 백신의 가지고 있는 데이터를 기반으로 감지하는 기법이 아닌 악성코드의 행동을 사전에 대처를 할 수 있도록 알 수 없는 URL이나 어플리케이션을 받았을 시 PC의 가상화와 같이 모바일 가상화 기술을 적용하여 모바일 기기 내에서 가상화 서버에서 우선 구동을 시킨 후 해당 URL이나 어플리케이션의 행동분석을 실행하는 시스템을 제시한다.

II. 관련 연구

1. Offline 모드 악성코드 탐지 기법

일반적으로 Offline에서 적용 가능한 악성코드 탐지기법은 크게 정적 분석(Static Analysis)과 동적 분석(Dynamic Analysis) 방법으로 구분된다.

정적 분석 방법은 프로그램에 대한 수행 없이 애플리케이션을 분석하는 방식이다. 안드로이드 애플리케이션에 대한 정적 분석은 Dalvik Bytecode 자체나 dex2jar, DED, soot, JD-GUI와 같은 도구를 사용하여 추출된 애플리케이션 소스 코드를 분석 대상으로 삼는다.[2] 정적 분석의 목표는 애플리케이션의 보안 취약점이나 악의적 행동 여부를 탐지하는 것으로 이를 위하여 Root 권한 획득을 위해 사용되는 Exploit 탐지, Permission 오남용 분석, 데이터 정보 누출 가능성탐지, IPC(Inter-Process Communication) 통신 취약점 분석 등 다양한 보안 관점을 분석 대상으로 고려하고 있다. 또한, 정적 분석 방법은 악성코드 및 일반 애플리케이션의 분석 결과로부터 습득한 경험적 정보(Heuristic)들을 이상 행동 및 취약점 탐지에 활용하며, 이를 위하여 애플리케이션의 Permission, Semantic, Control Flow 및 Data Flow 등을 분석한다. 일례로, 데이터 누출 가능성을 탐지하기 위하여 일반적인 애플리케이션이 Hard Coding된 특정 목적지로 SMS를 보내지 않는다" 는 Heuristic을 이용하며, 이를 탐지하기 위하여 애플리케이션 소스 코드의 Semantic을 분석한다.

정적 분석 방법은 안드로이드 플랫폼 수정이나 애플리케이션 수행을 필요로 하지 않기 때문에 타 탐지 기법에 비해 분석 시간 및 비용이 적게 드는 장점을 가진다. 따라서 급속하게 증가하고 있는 모바일 애플리케이션 수를 고려하였을 때, 추가적인 세부 분석 대상을 찾아내기 위한 필터링 수단으로서 활용될 수 있다. 그러나 정적 분석 방법은 분석 결과가 애플리케이션으로부터 추출된 소스 코드의 정확도에 의존하는 문제점을 가진다. 즉, 코드 난독화 기술이 적용된 애플리케이션에 대한 분석이 어렵다는 단점을 가진다. 최근 이러한 난독화 기술을 통한 정적 분석 탐지 우회 문제를 일부 해결한 시스템(RiskRanger)이 소개 되었다. 그러나 RiskRanger는 난독화 된 코드를 직접적으로 분석하기보다는, 정적 분석 시 사용된 기술을 우회할 수 있는 난독화 코드의 존재 여부를 탐지함으로써 난독화에 따른 False Negative를 줄이는 방식으로 동작한다.

동적 분석 방법은 안드로이드 디버거나 에뮬레이터를 이용하여 프로그램을 수행시켜 봄으로써 애플리케이션의 세부 행동을 분석하는 방식이다. 그러나 모바일 단말 상에서 디버거를 이용한 직접 적인 분석은 분석 도중 모바일 단말에 발생 가능한 피해 외에도 디버거 탐지 및 우회 로직을 포함한 악성코드들을 분석하지 못하는 단점을 갖는다[2]. 이러한 이유로, PC 기반 악성코드의 동적 분석에서와 같이 안드로이드 기반 악성코드의 동적 분석 역시 가상 머신이 활용되고 있으며, Honeynet Project와 DroidScope과 같은 가상 머신 상에서 안드로이드 애플리케이션의 동적 분석을 돕기 위한 도구들이 개발되고 있다. 동적 분석 방법은 악성코드의 특징 및 시그니처를 추출하기 위해 많이 이용되고 있으며 코드 난독화가 수행된 악성코드 역시 분석 가능하다는 장점을 갖는다. 그러나 모든 가능한 프로그램 실행 패스를 다 분석하지 못하는 문제점을 가지고 있다. 실제로 초기 동적 분석 방법은 단지 하나의 프로그램 실행 패스만을 분석할 수 있었으며, 현재 동적 분석 범위를 높이기 위한 목적으로 다중 실행 패스 분석 기법들이 연구되고 있다.[7]

2. 데스크탑 가상화 환경

데스크탑 가상화는 서버 기반 컴퓨팅을 기반으로 하는 기술로서, 컴퓨터 본체의 기능을 가상화 기술을 활용하여 수십 대의 컴퓨터를 1대의 중앙 서버에 구축하고, 사용자는 단말기와 주변장치만을 이용해 개인 PC를 이용하는 것처럼 업무처리를 지원하는 시스템이다. 사용자의 요청에 따라 개인화된 사용자 환경 설정, 운영체제, 응용프로그램 등을 조합하여 가상 데스크탑 환경을 만들고 이 이미지를 사용자에게 제공한다.

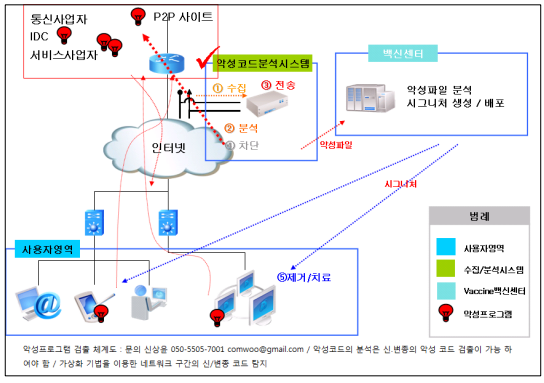

그림 2. 가상화 기법을 이용한 네트워크 구간의 악성코드 탐지
Fig. 2. Malware detection on the network segment using virtualization techniques

데스크탑 가상화 기반의 환경을 흔히 VDI(Virtual Desktop Infrastructure)라 부르는데, 캡슐화 된 파일 이미지를 통해 Desktop을 제공함으로서 관리하기가 용이

하고, 하드웨어 유형에 상관없이 사용가능하다. 관리적인 측면에서 보면 모든 작업이 서버 내에서 이뤄지므로 데이터 유출이 방지되고, 프로그램 업데이트나 패치가 매우 용이하다. 또한 이미지 파일로 보관되어서 데이터 백업과 복원이 쉽다. 사용자들에게는 개인 고유의 사용자 환경을 보장해줌으로써 지속적인 데이터의 보존이 가능하고, 다른 사용자들로 인한 바이러스 감염 및 시스템 이상으로부터 벗어날 수 있다.

3. 모바일 가상화를 이용한 기술(클라우딩)

컴퓨팅자원의 효율성을 위하여 1960년 6대 가상메모리 활용에서부터 시작된 가상화 기술은 메인프레임을 시작으로 C/S환경을 거쳐 지속적인 발전을 통해서 Hypervisor에 이르기까지 신기술이 단계적으로 선보이고 있다. 또한, 인터넷에 기반 한 분산컴퓨팅환경에서 그리드 컴퓨팅 및 유틸리티 컴퓨팅에 이르기까지 IT 산업 전반에 걸쳐 기술적용이 광범위하게 진행 되고 있다. 컴퓨팅 기술에서의 가상화는 물리적인 자원을 논리적으로 분할하거나, 다수의 물리적 자원을 통합하여 공용화 하는 기술로 정의할 수 있는데, 서비스차원에서의 시각으로 확대하여 대규모 사용자에게 컴퓨팅 자원의 효율적인 분배와 회수를 위해서 가상화를 활용한 플랫폼과 구조 그리고 소프트웨어 및 컨텐츠를 제공하는 것을 통해서 사용자가 내용과 장소 및 장비에 무관하게 누릴 수 있게 하는 것, 우리는 이것을 클라우드 컴퓨팅이라 일컫는다. 이것을 가능하게 하는 가상화 기술을 내용의 '인도(Delivery)'측면에서 크게 다음 세 가지로 나누고자한다. 스트리밍 방식의 어플리케이션 가상화의 경우는 모바일 적용에 제약사항이 존재하므로 이를 제외한 Presentation Virtualization(이후PV), Desktop Virtualization(이후 DV), Client Virtualization(이후 CV)으)로 구분될 수 있으며, 조금 더 그룹핑을 해보면 리소스 사용방식에 따라 PV와 DV는 서버 리소스를 활용하고 있고, CV는 클라이언트 리소스를 활용하여 가상화를 구현한다.[8]

III. 시스템 설계

일반적으로 PC에서의 가상화 시스템은 개인의 PC 클라이언트와 가상화 서버가 독립적으로 구성되어 있어서 가상화에서 발생하는 어떠한 현상이 직접적으로 개인 PC에 영향을 미치지 않아서, 최근 Cuckoo Sandbox와 같은 PC에서의 가상화를 기반으로 한 악성코드분석법이 연구되어 지고 있다. 가상화 머신을 구동시킨 후 그 안에서 Cuckoo Sandbox와 같은 분석도구를 활용하여 악성코드로 의심되는 파일이나 URL을 분석한 뒤 사용자에게 레포트 형식(XML)으로 출력하는 방식이 활용되고 있다. 이러한 방식은 직접적으로 PC에 영향을 주지 않으며, 행동분석 또한 가상화로 이루어진 OS에서 이루어지기 때문에 정확히 그 악성코드의 행동분석이 가능하며, 시간과 비용적인 면에서 효율적인 방식이 될 수 있다.

하지만 이러한 방식은 악성코드에 관한 전문지식이 있어야 이해가 가능하며, 사실상 스마트폰 사용자들이 정확히 의심되는 APK파일이나 URL을 구분해 낸다는 것은 쉽지 않다. 오히려 아는 지인이나 인증된 기관에서 보냈고, 배포되었다는 이유로 의심 없이 열어보는 경우에 발생하는 피해가 대다수 였다.

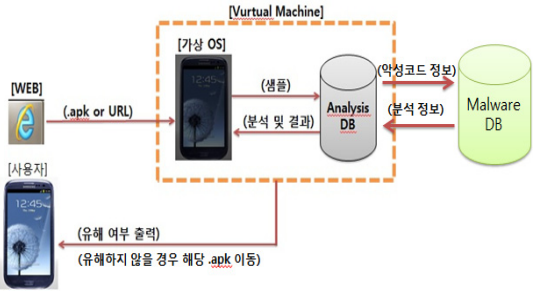

그림 3. 본 논문에서 제시하는 시스템 구성도
Fig. 3. Suggested System Diagram in this paper

위 그림은 본 논문에서 제시하는 가상화 기술을 이용한 악성코드 분석의 시스템 구성도 이다. 기존의 사용자들은 외부에서 URL이나 APK파일을 받게 되면 URL의 경우 실행 즉시 피해가 발생이 되며, APK파일 또한 받거나 실행하는 즉시 악성코드의 행동이 발생한다.

그러나 본 논문에서 제시하는 시스템의 구성도는 외부(WEB)에서 APK파일이나 URL이 첨부된 메시지를 받게 될 경우 이를 사용자 기기에 내려 받게 되지 않고, 가상화로 구성된 가상 OS서버의 저장소에 저장이 된다.

VM 내부에서 가상화 기반의 악성코드 분석도구인 Cuckoo Sandbox를 활용하여 분석이 실행되며, 행동분석이 끝난 뒤에 해당 악성코드의 정보는 외부 악성코드 분

석 DB에 저장이 되며 이 DB는 웹서버와 연동이 되어 실시간으로 악성코드 정보가 누적이 된다. 마지막으로 내부에서 분석 되어진 결과는 외부 DB로 전송된 자세한 정보와는 다르게 사용자가 이해하기 쉽게 유해한 자료인지 아닌지를 판단할 수 있게 간략한 정보를 제공한다. 이는 기존의 백신과 같은 데이터를 기반으로 한 감지 대책과는 다르게 사용자가 열어보고 싶은 URL이나 어플리케이션을 실제적으로 실행을 시키고 결과를 본 뒤에 해당 메시지나 파일을 삭제하거나 활용할 수 있으므로 기존의 시스템의 피해 발생 이후의 대처가 가능하다는 최대 단점을 사용자가 해당 분야의 전문적인 지식이 없어도 직접적으로 분석을 실행하는 경우 이므로 기존의 단점이 개선되었으며, 새롭게 등장하는 악성코드에 대한 대처가 더욱 신속하기 때문에 기존의 시스템보다 더욱 효율적임을 알 수 있다.

Ⅳ. 시스템구축

본 논문에서 제시하는 시스템에서 사용하는 악성코드 분석도구는 가상화기반의 Cuckoo Sandbox이다. Cuckoo Sandbox는 Python 언어 기반으로 이루어진 소스이며 시스템 구축에서는 이 가상화 기반의 Cuckoo Sandbox 구현장면을 다룬다.

4.1 가상화 환경 구축 및 Cuckoo Sandbox 실행

```
$cd /opt/cuckoo # cuckoo설치 폴더로 이동
$./cuckoo.py  #cuckoo 실행
```

Cuckoo Sandbox를 우분투(리눅스)환경에서 구축하며, 사전에 관련 패키지와 Virtualbox(가상화 환경 툴)을 설치한다. 이후 다음과 같은 파이썬 명령어를 통해 Cuckoo Sandbox에 진입한다. 실행하게 되면 Virtualbox 환경을 이용해 실행 했다는 메시지와 함께 관련정보가 출력이 된다.

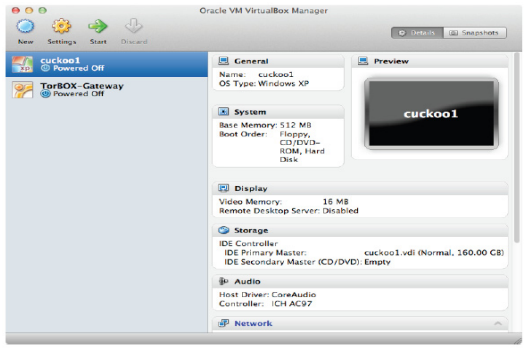

그림 4. 설치된 Virtualbox(가상화 환경)
Fig. 4. Installed Virtualbox (virtual environment)

위 그림은 가상화 환경으로 이용할 Virtualbox 설치가 완료된 후 실행된 화면이다. 이 Virtualbox를 통해 가상화 환경(Windows XP)을 실행시키면 아래 그림과 같이 PC에서의 윈도우즈 바탕화면을 볼 수 있다. 초기상태는 악성코드의 영향을 받지 않고 정상적인 시스템을 가지고 있는 운영체제이므로 스냅샷을 찍는다. 스냅샷을 찍을 때에는 터미널에서 다음과 같은 명령어를 입력한다.

(“ $ virsh snapshot-create "<Name of VM> ”)

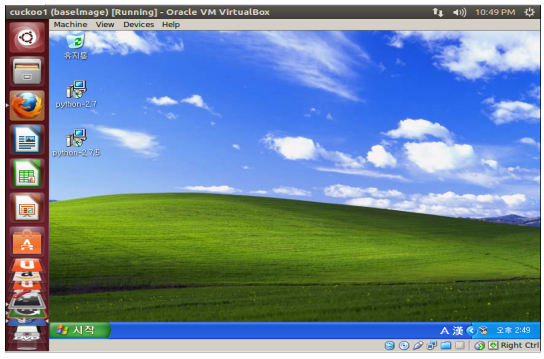

그림 5. 가상화 환경에서의 Windows XP
Fig. 5. Window XP in the virtualization environment

여기서 스냅샷은 악성코드의 행동분석 이전의 정상적인 상태를 저장하는 것으로 행동분석 이후 정상 시스템으로 복구하기 위한 작업이다. (PC에서의 예시 이지만 모바일도 동일하다). 현재 가상화 환경에 쓰는 가상화 머신을 모바일에서 사용가능한 툴로 개발 중인 상태이므로 구축에 관련된 연구는 PC에서 실행하기로 한다. 이후 가상화 환경과 분석도구가 준비되었으므로 악성코드 분석을 실

행할 단계이다.

분석방법에는 Cuckoo 내부에서 분석이 이루어지는 내부 분석과 Web 유틸을 이용한 Web 분석이 존재하는데, 이는 Cuckoo의 악성코드 분석 모듈을 이용하는 것은 동일하므로 Web 분석을 이용하기로 한다. Web 분석으로 진입할 때에는 다음과 같은 명령어를 이용한다.

($ python utils/web.py)

이후에 Web에서 http://localhost:8080으로 진입하여 세팅을 끝내면 파일을 업로드 시킬 수 있는 화면에서 해당 샘플 악성코드 파일을 게시하면 다음 그림과 같은 분석결과 화면에 출력 된다.

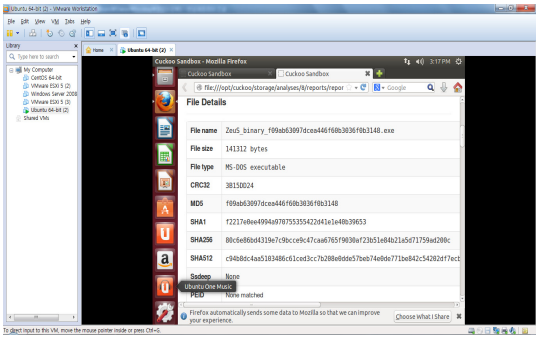

그림 6. 악성코드 분석 결과
Fig. 6. Analyzing results of malware

분석이 된 첫 번째 화면이다. 만일 내부분석을 이용할 경우 Web 분석과는 달리 파일을 직접적으로 선택하여 web.py 유틸을 이용하는 것이 아닌 submiy.py라는 분석 유틸을 이용하게 된다.(모바일에서는 submit.py라는 내부 유틸을 활용할 예정이다) 위 그림에서의 분석결과는 해당 파일에 대한 정보가 상세 기술 되어 있다. 파일명, 파일의 크기, 파일의 MD5 및 sha1~512의 암호화된 해시코드가 기술되어 있으며 이 암호화 된 해시코드를 알게 되면 변조가 되었는지의 여부를 알 수 있다.

V. 결론

모바일에서의 악성코드 분석기법으로 활용하고 있는 기존의 방법은 전문가의 경우를 제외하고는 대부분 피해가 발생한 이후에 해당 .apk파일이나 URL이 악성코드를 내포하고 있으므로 분석을 시행하는 기법이며 데이터에 기반 한 감시 방법이었다. 하지만 본 논문에서 제시하는 모바일 가상화를 이용한 행위분석기법은 사용자가 사용 직전에 바로 분석이 가능할 수 있으므로, 현 시점에서의 대응 대책(사후) 가 아닌 (사전)에 분석이 가능하므로 피해를 줄일 수 있고, 활용 가능한 정보나 어플리케이션을 제한 없이 사용할 수 있도록 해주는 이점을 가지고 있다. 현재는 시스템의 구성을 마친 상태이며, 악성코드 분석 도구에 대한 연구가 완료된 상태이다. 추후에는 모바일 가상화 환경이 마련이 되는 즉시 관련 어플리케이션 개발로 진행될 예정이다.

Reference

[1] S. H. Yeom, Statistics for Subscribers of Wire • Wireless Communication, Korea Communications Commission; 2012

[2] Sophos, “Security Threat Report 2013,” 2013

[3] L. K. Yan and H. Yin, “DroidScope: Seamlessly Reconstructing the OS and Dalvik Semantic Views for Dynamic Android Malware Analysis,” Proc 21st USENIX conf. Security Symp., Security, 2012.

[4] McAfee, “McAfee Threats Report: First Quarter 2012,” 2012.

[5] M. H. Shin, “2013 Technology Trend for Android-based Malware Detector”, Electronics and Telecommunications Research Institute, 2013

[6] H. Y. Kim, “A Study of Efficient Dynamic Analysis for Malware Detection in a Mobile Platform”, Korea Information Processing Society, 2013

[7] Y.M. Bae, “Technology Trend of Desktop Virtualization for Information Security”, Research Journal for Security Engineering, 2011

[8] B. J. Choi, A study on the concept and implementation method for Korea Virtualization-based Mobile Cloud Computing, Journal of Information Processing Society, 2011

[9] T. H. Kim, APT attack type and countermeasures,

Security News, 2015
[10] J. Y. Choi, Malware Analysis of hardware virtualization, Korea Network Information Society in 2010 Annual General Meeting and Fall Conference, 2010
[11] M. H. Lee, Android malware, year-on-year doubling, Digital Daily, 2015
[12] Y. G. Jung, Hospital Security System using Biometric Technology, J.IWIT, Vol.11, No.2 pp219-224, 2011

저자 소개

김 장 일(정회원)

- 1995년 : 순천대학교 (이학사)
- 2015년 : 을지대학교 석사과정수료
- 2013년 ~ 현재 : ㈜디플랫폼 이사
- 2011년 ~ KISA 피싱센터 자문 컨설턴트

<주 관심분야: 정보보호 및 모바일 보안, 임상데이터마이닝, 의료정보시스템>

이 희 석(정회원)

- 2015년 : 을지대학교 (이학사)
- 2015년 : 을지대학교 석사과정
- 2015년 ~ 현재 : ㈜디플랫폼 연구원

<주 관심분야: 정보보호 및 네트웍 보안, 의료정보시스템>

정 용 규(종신회원)

- 1981년 : 서울대학교 (이학사)
- 1994년 : 연세대학교 (공학석사)
- 2003년 : 경기대학교 (이학박사)
- 1999년 ~ 현재 : 을지대학교 교수

<주 관심분야: 임상데이터마이닝, 의료정보시스템, 전자거래표준>