http://dx.doi.org/10.14400/JDPM.2013.11.12.381

안드로이드 플랫폼에서의 High-Interaction 클라이언트 허니팟 적용방안 연구

정현미*, 손승완**, 김광석**, 이강수**
한국과학기술정보연구원*, 한남대학교 컴퓨터공학과**

A High-Interaction Client Honeypot on Android Platform

Hyun-Mi Jung*, Seung-Wan Son, Kwang-Seok Kim**, Gang-Soo Lee****
Korea Institute of Science and Technology Information*
Dept. of Computer Engineering, Hannam University**

요 약 안드로이드 플랫폼에서의 새로운 변종 악성코드가 기하급수적으로 증가함에 따라 보다 빠르고 능동적인 대처방안이 필요하다. 본 연구에서는 안드로이드 플랫폼에 High-Interaction 클라이언트 허니팟을 적용하였다. 시스템 적용방안을 위하여 전체 흐름을 설계하고 각 세부모듈의 기능을 분석하여 안드로이드 플랫폼에 최적화 하였다. 제안하는 시스템은 기존 PC 환경의 High-Interaction 클라이언트 허니팟의 장점을 모두 갖추고 있으며 관리 서버와 저장 서버를 분리하여 보다 유연하고 확장된 형태로 설계되었다.

주제어 : 안드로이드 플랫폼, 모바일 악성코드 분석, 안드로이드 악성코드, 클라이언트 허니팟, 변종 악성코드

Abstract As the new variation malicious codes of android platform are drastically increasing, the preparation plan and response is needed. We proposed a high-interaction client honeypot that applied to the android platform. We designed flow for the system. Application plan and the function was analyze. Each detail module was optimized in the Android platform. The system is equipped with the advantage of the high-interaction client honeypot of PC environment. Because the management and storage server was separated it is more flexible and expanded.

Key Words : android platform, mobile malicious code analysis, android malicious code, client honeypot, variation malicious code

1. 서론

다양한 스마트폰 공격유형 중 최근 사용자에게 가장 많은 피해를 주는 것은 어플리케이션 및 콘텐츠 취약점을 이용한 모바일 악성코드의 유포이다. 이를 반영하듯 2013년도 정보보호 10대 이슈 전망에 '해커들의 공격 목표로 부상하고 있는 모바일 앱 보안 중요성 증대'와 '지능화된 피싱(Phishing)기법, 스미싱(Smishing)을 이용한

* 본 연구는 2013년 한남대학교 학술연구조성비 지원으로 이루어졌음
Received 25 October 2013, Revised 19 November 2013
Accepted 20 December 2013
Corresponding Author: Gang-Soo Lee(Computer Engineering, Hannam University)
Email: gslee@hnu.ac.kr
ISSN: 1738-1916

개인정보/금전탈취확산'이 선정되었다[1]. 모바일 악성코드의 공격방식 대부분은 개인의 정보를 유출하고 동시에 사용자에게 금전적인 피해를 입힌다.

특히, 안드로이드 플랫폼의 경우 오픈 소스 정책 및 다양한 기기의 보급으로 인하여 사용자가 급속도로 증가함과 동시에 2011년부터 악성코드로 인한 공격건수가 폭발적으로 증가하였다. 현재 안드로이드 악성코드는 빠르게 생성 및 변조되어 퍼지고 있으며, 공격의 종류도 급격하게 증가하고 있다.

현재, 안드로이드 악성코드의 증가와 함께 이를 대응하기 위한 연구가 활발히 진행되고 있으며 각 단말 제조사 및 ISP(Internet Service Provider) 사업자들은 모바일 백신 등을 유포하여 개인 소비자들을 보호하고 있는 추세이다. 그러나 대게 안드로이드 악성코드 유포의 목적이 개인정보 및 금전 탈취이기 때문에 공격자들은 악성코드의 빠른 진화를 통해서 자신들의 목적을 이루려고 한다. 따라서 이러한 악성코드에 대처하기 위해서는 보다 악성코드의 빠른 수집과 분석이 필요하다.

본 논문에서는 빠르게 변조되는 안드로이드 악성코드를 단 시간 내에 수집하고 분석 대처 할 수 있는 수집시스템을 설계하기 위하여 허니팟(Honeypot) 개념을 이용하였다. 제안 시스템은 가상환경을 기반으로 한 High-Interaction 클라이언트 허니팟(Client honeypot)을 이용하여 설계되고 구현되었으며 안드로이드 플랫폼에 최적화 되었다.

본 논문의 구성은 다음과 같다. 2장에서는 허니팟을 이용한 악성코드 수집 및 분석 기술동향을 알아보고 High-Interaction 클라이언트 허니팟의 장단점을 분석 한다. 3장에서는 High-Interaction 클라이언트 허니팟의 안드로이드 적용방안을 살펴보고 시스템을 설계한다. 4장에서는 High-Interaction 클라이언트 허니팟과 제안시스템 동일 기능별 모듈을 분석한다. 5장에서는 결론을 내린다.

2. 관련 연구

2.1 허니팟(Honeypot)

허니팟 이란 외부의 다양한 공격을 유인해 공격 동향 및 유형을 파악 가능하게 구현한 가상의 시스템을 의미한다[2]. 즉 실제사용하고 있지 않은 시스템을 사용하고 있는 것처럼 보이게 속여 공격자의 침입을 허용하는 시스템이다. 크게 Low-Interaction 허니팟[3]과 High-Interaction 허니팟[3] 으로 나뉜다. Interaction 이란 허니팟이 공격자에게 허용하는 활동의 레벨의 정의를 의미한다[3]. Low-Interaction 허니팟은 제한된 Interaction, 즉 에뮬레이트 된 서비스나 운영체제 등을 이용하여 구현된 것이다. 구현이 간단하고 인프라에 대한 부담이 적은 것이 장점이지만 제한된 정보만을 수집가능하고 알려진 공격 외에 변조된 공격의 식별이 어려운 것이 단점이다. 가장 많이 알려진 Low-Interaction 허니팟 도구에는 Honeyd[4]가 있다. Honeyd는 사용하지 않는 IP space를 모니터링 하여 공격을 탐지하는 방식이다.

High-Interaction 허니팟은 실제 동작하는 시스템 및 어플리케이션 제공한다. 많은 양의 정보 수집 가능, 공격자의 행위에 대한 가정이 필요 없으므로 신종 변종 공격이 탐지 가능하다는 장점이 있다. 그러나 개발 및 유지보수가 복잡하고, 다른 시스템으로의 공격전이 위험성이 있다. 대표적인 예로는 Honeynet Project[5]가 있다.

기존 허니팟을 이용하여 악성코드를 수집하고 분석하는 것은 한계점이 존재한다[6]. 다음 표 1에서는 허니팟의 장단점 비교를 통하여 허니팟의 한계점을 보여준다.

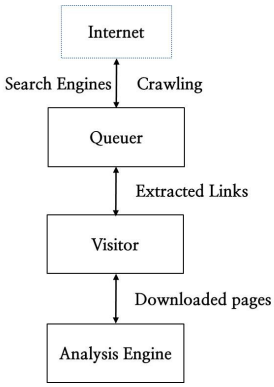
〈Table 1〉 Compare of Honeypots

Advantages	Disadvantages
- Easiness the collection of malicious data - It can be applied to a variety of systems - New attacks monitoring possible - Burdenless implementation	- Limitation of detection - Attacker can avoid it. - Management and operation needed - Risk of expansion attack

2.2 클라이언트 허니팟(Client Honeypot)

2.1장에서 보여주듯이 허니팟을 이용한 악성코드 수집 분석에는 이미 한계점이 존재한다. 또한 현재 공격기술이 발전함에 따라 기존 허니팟 기술들은 공격자의 흥미를 유발시키지 못한다. 따라서 이후 보다 능동적으로 악성코드를 수집하고 분석하기 위하여 클라이언트 허니팟[3]의 개념이 도입되었다. 이것은 웹 브라우저를 이용하여 웹페이지를 능동적으로 방문하여 해당 페이지에 악성여부를 탐지하여 악성코드를 수집하고 분석하는 방식이다. 단순히 공격을 기다려서 악성코드를 수집하는 방식

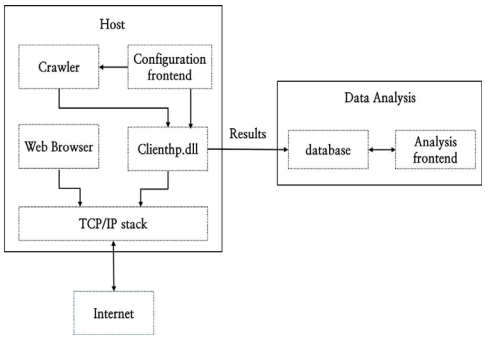
보다는 악성코드의 수집이 빠르고 제로 데이(Zero-day) 공격에 강하다는 장점이 있다. 클라이언트 허니팟 또한 다음 두 가지의 종류로 나눌 수 있는데 첫째는 Low-Interaction 클라이언트 허니팟이다. 이것은 웹상의 악성코드의 분석을 위하여 완벽한 기능의 운영체제나 웹 브라우저를 이용하지 않는다. 단지 시뮬레이트된 클라이언트를 이용한다. 이러한 방법의 대표적인 구현 예는 HoneyC[7]이다. 다음은 HoneyC의 개념도이다.



[Fig. 1] HoneyC

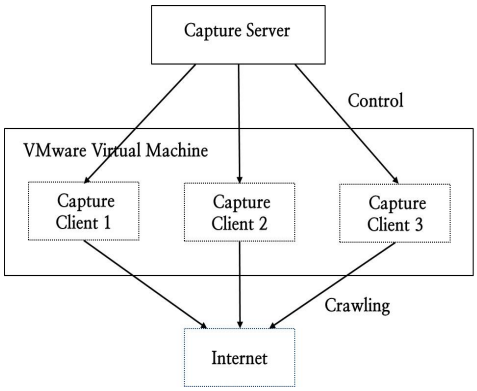
이러한 방식은 악성코드의 수집부분에 관해서만 보다 능동적 기존 Low-Interaction 허니팟의 장・단점을 모두 승계한다.

두 번째는 High-Interaction 클라이언트 허니팟이다. 이 개념도 마찬가지고 기존 High-Interaction 허니팟의 장・단점을 모두 승계한다. 다음 그림2는 일반적인 High-Interaction 클라이언트 허니팟의 일반적인 구성형태[3]이다.



[Fig. 2] High-Interaction Client Honeypot[3]

High-Interaction 클라이언트 허니팟을 이용한 악성코드 분석 도구 중 가장 많이 알려진 것이 Capture-HPC [8,9]이다. 이 도구는 허니넷 프로젝트(Honeyent Project)의 산출물로 나온 클라이언트 공격을 탐지 분석하는 도구이다[5,10]. 다음 그림3은 Capture-HPC의 개념도[3]이다.



[Fig. 3] Capture-HPC[3]

Capture-HPC는 클라이언트 서버구조를 기반으로 하고 있다. 서버는 클라이언트를 컨트롤하고 각각의 클라이언트는 가상머신 상에 구현되어 독립적으로 작동한다.

다음 표2는 Low-Interaction 클라이언트 허니팟과 High-Interaction 클라이언트 허니팟을 비교분석한 것이다.

〈Table 2〉 Compare Client Honeypot

classification	Low-Interaction	High-Interaction
Implement	- Emulated operating system and application	- Real system and application
Advantage	- Low cost - Low risk of infringement of the second - Collecting malicious codes are more fast.	- All feature implemented on system - New/variation malicious behavior can be detected. - A large amount of information can be collected
Dis-advantages	- Limited information is collected - Available only detect known attacks	- The complexity of the process of building - High cost of building - High risk of second infringement
Tools	HoneyC[7], Monkey-Spider[11] etc	Capture-HPC, HoneyMonkey[12,13] etc

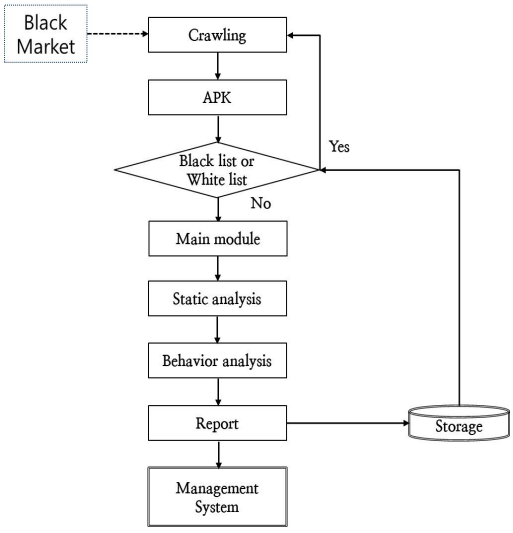
3. 제안시스템 설계

3.1 안드로이드 플랫폼에서의 High-Interaction 클라이언트 허니팟 적용방안

그림 2의 High-Interaction 클라이언트 허니팟의 일반적인 구성을 보면 크게 악성코드를 수집하는 Host 부분과 악성코드를 분석하는 Data Analysis 시스템으로 구분된다. Host의 구성 프론트엔드는 탐색엔진을 가지고 웹 페이지 탐색을 위한 키워드 또는 다양한 파라미터를 제공한다. Host 상의 Clienthp.dll은 다이얼로그박스의 조정, 시스템 무결성 체크 및 웹브라우저의 네비게이팅 기능을 하는 모듈이다. 모든 로그파일은 원격으로 데이터베이스에 저장된다. Data analysis 시스템의 분석 프론트엔드(frontend)는 수집된 악성코드를 실제 환경에서 분석하는 역할을 한다. 만일 분석 프론트엔드가 실제 네트워크상에 위치한다면 제 2차 침해 위험의 발생위험이 있다. 분석대상인 악성코드가 네트워크 전파 특성을 가지고 있다면 동일 네트워크상에 구현되어있는 실제 시스템으로의 전이 가능성이 있기 때문이다. 이러한 단점을 극복하기 위하여 분석 시스템은 실제 네트워크상에서는 분리된 형태인 예를 들어 가상환경으로 구성하는 것이 바람직하다. 또한 가상환경은 분석 프론트엔드에서 악성코드 분석 후 다음 파일 분석 전에 시스템 재설치를 용이하게 한다. 따라서 제안하는 시스템은 위의 장점을 적용되고 적은 자원으로 높은 가치의 침해공격정보 수집이 가능하도록 가상환경으로 구축한다.

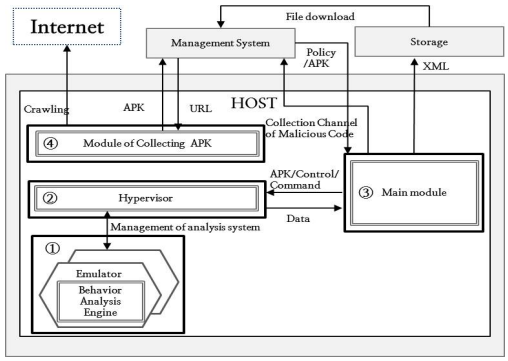
3.2 제안 시스템 구조

3.1에서 분석한 결과를 안드로이드 플랫폼에 적용하여 시스템을 설계 한다. 현재 안드로이드 악성코드는 대부분 어플리케이션에 의하여 전파 된다. 이러한 이유로 안드로이드 악성코드 수집의 효율성을 위하여 악성어플리케이션 유포의 근원이 되는 블랙마켓을 중심으로 크롤링을 수행한다. 수집된 APK(안드로이드 실행파일)가 만약 기존 분석을 통하여 블랙 또는 화이트리스트 분류가 되었다면 분석 대상에서 제외 한다. 이후 분석을 위하여 메인모듈의 기능을 이용하여 각 에뮬레이터 기반의 분석엔진으로 악성의심파일이 전달되며 메인모듈에서는 정적 분석을 수행하고 에뮬레이터기반의 행위분석엔진에서는 동적 분석을 통하여 악성 여부를 빠르게 탐지한다. 제안하는 시스템의 전체 흐름도는 다음 그림 4와 같다.



[Fig. 4] System Flowchart

전체 시스템은 하나의 물리적 시스템을 기반으로 가상환경으로 구현 하였다. 즉 하나의 물리 시스템 안에 분석을 시행하는 에뮬레이터의 환경을 다양하게 구성할 수 있으며 적은 자원으로 많은 양의 데이터를 처리 할 수 있다. 다음 그림 5는 전체 시스템의 구성을 보여준다.



[Fig. 5] Design of system

3.3 제안 시스템 모듈별 기능

제안 시스템은 하나의 서버시스템에 행위분석엔진을 탑재한 에뮬레이터, 가상환경에서의 호스트 자원과 게스트 운영체제 및 두 모듈간의 송·수신 제어 메커니즘을 가지고 있는 하이퍼바이저, 각종 분석 결과 및 정보를 수

집하고 가상머신의 재 복구 등을 수행하는 메인모듈 그리고 크롤링을 이용하여 안드로이드 악성코드 어플리케이션을 수집하는 어플리케이션 수집모듈로 나뉜다. 또한 관리를 위한 시스템과 저장을 위한 시스템을 별도의 서버에 구현하여 관리 및 저장에 확장성을 높였다.

다음 표 3 에서는 제안 시스템의 구현을 위하여 각 모듈별 기능을 분류하였다.

〈Table 3〉 Module of System

No	Classification	Main Function
①	Emulator	- Malicious behavior of an application module for collecting information with the Android virtual machine
②	Hypervisor	- Host resources control
		- Guest OS, control
		- Transmitting and receiving transmission control
③	Main module	- Collecting of process name, package name, main activity
		- Collecting information generated in XML format
		- Analysis results of XML forward to storage.
		- Analysis results of XML insert to database
		- Revert to type of emulator
④	Module of Collecting APK	- Collecting of URL
		- Link URL filtering
		- Link URL filtering insert to database

4. 제안시스템평가

4.1 분석

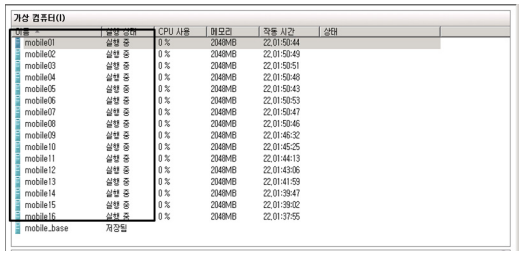
일반적인 High-Interaction 클라이언트 허니팟의 각 모듈과 제안된 안드로이드 플랫폼에 적용 시스템의 각 모듈을 비교 하였다. 다음 표4는 공통 기능에 대한 모듈 비교이다. 비교결과를 분석해보면 기존 환경에서의 구성 프론트엔드와 데이터베이스를 각각 별도의 서버 시스템으로 설정하여 정보 및 정책관리를 유연하게 구현할 수 있다. 또한 분석 프론트엔드, Clienthp.dll 및 크라울러에 해당하는 모듈을 하나의 물리머신 내에 구현하고 특히 분석 프론트엔드를 가상머신으로 설정함으로써 보다 적은 자원으로 다양한 분석결과를 도출가능하게 설계되었다.

〈Table 4〉 Comparison of function

High-Interaction Client Honeypot	System	Function
- Configuration front-end	- Management System	· Management of information and policy for the collection of malicious code,
- Database	- Storage	· Save analysis data and results
- Analysis front-end	- Emulator	· Analysis of collected malware files
- Clienthp.dll	- Hypervisor - Main module	· Physical resources control · Analysis control · Insert a database of collect information · System integrity check, etc.
- Crawler	Module of Collecting APK	· Collecting of Malicious suspected file

4.2 적용결과

제안된 시스템을 적용하여 구현 하였을 경우 분석 프론트엔드에 해당하는 안드로이드 에뮬레이터의 환경 설정을 자유로이 변경가능하다. 다음 그림 6은 제안된 시스템을 이용하여 구현한 안드로이드 에뮬레이터의 현황 이다.



[Fig. 6] Implementation of System

5. 결론

기존 PC 환경에서 효율적인 악성코드 분석방법으로 알려진 High-Interaction 클라이언트 허니팟을 안드로이드환경에 적용하여 설계하였다. 제안된 시스템은 일반적인 High-Interaction 클라이언트 허니팟의 각 모듈별 기능을 살려 안드로이드 플랫폼에서도 적용가능하게 설계되었다. 이는 기존 장점을 모두 High-Interaction 클라이

언트 허니팟 가지게 된다. 또한 악성코드 분석 시스템을 가상환경으로 구축하여 적은 물리적 자원을 가지고 고 성능을 기대할 수 있는 시스템 구현이 가능하며 관리 서버와 저장 서버를 별도로 구축하여 보다 유연하고 확장가능하게 설계되었다. 앞으로 빠르게 진화하는 안드로이드 악성코드의 발생가능성을 예상하고 즉각적으로 대응 가능한 체계를 구축할 수 있게 도와주는 시스템이 될 것이다.

ACKNOWLEDGMENTS

This paper was supported by the Hannam University Research Fund in 2013.

REFERENCES

[1] http://www.kisa.or.kr/
[2] Lance Spitznet (2003), Honeypots : Definitions and Value of honeypots.
[3] Thorsten Holz, Niels Provos (2008), Virtual Honeypots
[4] http://www.honeyd.org/
[5] http://www.honeynet.org/
[6] J Fritz (2011), Hybrid Intrusion detection network monitoring with honeypots.
[7] Christian Seifert, Ian Welch, Peter Komisarczuk (2006), HoneyC - The Low-Interaction Client Honeypot.
[8] https://projects.honeynet.org/capture-hpc/
[9] Radek Hes ,Ramon Steenson ,Christian Seifert (2010), The Capture-HPC client architecture
[10] Christian Seifert, Ramon Steenson, Ian Welch, Peter Komisarczuk, Thorsten Holz, Bing Yuan, Michael A. Davis (2007),Know your Enemy: Malicious Web Servers.
[11] Ali Ikinci, Thorsten Holz, Felix Freiling(2008), Monkey-Spider: Detecting Malicious Websites with Low-Interaction Honeyclients
[12] Yi-Min Wang, Doug Beck, Xuxian Jiang, Roussi Roussev, Chad Verbowski, Shuo Chen, and Sam King (2006), Automated Web Patrol with Strider HoneyMonkeys: Finding Web Sites That Exploit Browser Vulnerabilities
[13] http://research.microsoft.com/HoneyMonkey/

정 현 미(Jung, Hyun Mi)

- 2010년 8월 : 한남대학교 컴퓨터공학과 (공학 석사)
- 2010년 9월 ~ 현재 : 한남대학교 컴퓨터공학과 박사수료
- 2012년 10월 ~ 현재 : 한국과학기술정보연구원, 과학기술사이버안전센터 선임연구원
- 관심분야 : 소프트웨어공학, 보안공학, IT 보안시스템 개발, 안드로이드 악성 코드 분석
- E-Mail: hmjung@kisti.re.kr

손 승 완(Son, Seung Wan)

- 2013년 2월 : 한남대학교 컴퓨터공학과(공학사)
- 2013년 3월 ~ 현재 : 한남대학교 컴퓨터공학과 석사과정 재학 중
- 관심분야 : 컴퓨터보안, 소프트웨어공학
- E-Mail : son2898@hnu.ac.kr

김 광 석(Kim, Kwang Seok)

- 2013년 2월 : 한남대학교 컴퓨터공학과(공학사)
- 2013년 3월 ~ 현재 : 한남대학교 컴퓨터공학과 석사과정 재학 중
- 관심분야 : 컴퓨터보안, 소프트웨어공학
- E-Mail : kkslove4721@hnu.ac.kr

이 강 수(Lee, Gang Soo)

- 1983년 2월 : 서울대학교 전산학(이학석사)
- 1985년 2월 : 서울대학교 전산학(이학박사)
- 1987년 3월 ~ 현재 : 한남대학교 컴퓨터 공학과 교수
- 관심분야 : 보안공학, 소프트웨어공학, 웹공학
- E-Mail : gslee@hnu.ac.kr