韓國컴퓨터情報學會 論文誌
第16卷 第4號, 2011. 04.

2011-16-4-3-6

논리적 분석 기반의 안드로이드 스마트폰 포렌식 도구 구현

김 익 수*, 안 영 건*, 이 정 현*, 양 승 제**, 김 명 호*

Implementation of an Android Smart phone Forensic Tool Based on Logical Analysis

Ik-Su Kim*, Young-Geon An*, Jeong-Hyun Yi*, Seung-Jei Yang**, Myung-Ho Kim*

요 약

과거에는 국내 모바일 포렌식에 관한 연구가 휴대폰에 한정되었지만, 스마트폰 사용량의 증가 추세에 따라 스마트폰 포렌식에 관한 연구도 활발히 진행될 것으로 예상된다. 특히, 안드로이드 스마트폰의 시장 점유율은 급속히 증가하고 있기 때문에 안드로이드 스마트폰 포렌식에 관한 연구는 매우 중요하다. 이에 본 논문에서는 논리적 분석 기반의 안드로이드 스마트폰 포렌식 도구 구현을 설명한다. 본 포렌식 도구는 Oxygen Forensic Suite 2010과 비교할 때, 검색 기능을 추가로 제공하며 추출된 미디어 정보에 대한 리소스 링크를 제공하기 때문에 포렌식에 소요되는 시간을 더욱 절약할 수 있다. 아직까지 국내에는 스마트폰 포렌식 도구가 전무하기 때문에 본 논문을 통해 구현된 포렌식 도구는 스마트폰 포렌식 기술 발전에 기여를 할 것으로 기대된다.

▸ 키워드 : 포렌식, 안드로이드, 스마트폰

Abstract

In the past, the domestic research on mobile forensics has been limited to cell phones. Increasing use of smart phones, studies on smart phone forensic will be conducted actively in the future. In particular, the study on Android forensic is very important because Android smart phone market share is increasing rapidly. In this paper, we describe an implementation of an Android smart phone forensic tool based on logical analysis. Compared with Oxygen Forensic Suite 2010, this tool saves time it takes to perform Android smart phone forensic because this tool provides search feature and resource links for extracted media information. So far, no smart phone forensic tool is introduced in Korea. Accordingly, this tool would contribute to the advancement of the technology on smart phone forensic.

▸ Keyword : Forensic, Android, Smart phone

• 제1저자 : 김익수 • 교신저자 : 김명호
• 투고일 : 2010-12-10, 심사일 : 2011-01-04, 게재확정일 : 2011-01-10
* 숭실대학교 컴퓨터학부(School of Computer Science and Engineering, Soongsil University)
** ETRI 부설 연구소(The Attached Institute of ETRI)
※ 본 연구는 2010년도 ETRI 부설 연구소의 지원에 의한 것임.

Ⅰ. 서 론

최근 스마트폰에 대한 관심이 높아지면서, 스마트폰 운영체제 시장에서도 새로운 경쟁구도가 형성되고 있다. 전 세계적으로 스마트폰 판매량은 2010년 3사분기 기준으로 노키아 스마트폰이 1위를 차지하고 있으나 해마다 시장 점유율의 감소 추세를 보이고 있는 반면, 안드로이드 스마트폰은 시장 점유율에 있어 급속한 성장을 보여 현재 스마트폰 시장에서 2위를 차지하고 있다. 아울러 현재 모바일 단말기 사용자들이 과거의 단순 기능만을 제공하던 휴대폰을 뒤로하고 막강한 기능을 제공하고 있는 스마트폰으로 이동하는 추세이다.

스마트폰은 기존 휴대폰과 달리 단순한 전화통화, 문자메시지 기능뿐만 아니라 인터넷을 통한 풀 브라우징(Full Browsing)을 지원하며, GPS(Global Positioning System)를 통한 위치 기반 서비스도 제공한다. 또한, 대용량 메모리를 탑재하고 있어 전화번호부, 일정관리는 물론, 용량이 큰 미디어 파일도 저장이 가능하여 이동 저장 장치로도 활용이 가능하다. 따라서 스마트폰은 휴대폰과 비교할 때 범죄 사건에 대한 포렌식(Forensic) 과정에서 더 많은 증거 자료를 획득할 수 있다.

지금까지 국내에서는 휴대폰과 관련한 포렌식 도구 개발과 포렌식 방법에 관한 연구[1-3]만이 진행되었을 뿐, 스마트폰과 관련한 포렌식 연구 및 도구 개발은 거의 전무하다. 반면, 국외에서는 예전부터 스마트폰의 활성화가 이루어져 휴대폰은 물론 스마트폰과 관련한 포렌식 연구 개발이 활발히 진행되고 있다[4-7]. 최근의 국내 스마트폰 사용자의 증가 추세와 안드로이드 스마트폰의 시장 점유율을 고려할 때, 바로 지금이 안드로이드 스마트폰의 포렌식 연구 및 도구 개발에 있어 매우 중요한 시기라고 볼 수 있다. 이에 본 논문에서는 논리적 분석 기반의 안드로이드 스마트폰 포렌식 도구 구현을 설명한다. 본 포렌식 도구는 안드로이드가 제공하는 컨텐트 프로바이더(Content Providers)를 이용하여 스마트폰에 저장된 정보를 추출하기 때문에 안드로이드폰의 루팅(Rooting) 과정이 요구되지 않는다. 게다가 상용 포렌식 도구인 Oxygen Forensic Suite 2010과 비교할 때, 검색 기능을 추가로 제공하며 추출된 미디어 정보에 대한 리소스(이미지, 오디오, 비디오 파일) 링크를 제공하기 때문에 포렌식에 소요되는 시간을 더욱 절약할 수 있다. 아직까지 국내에는 스마트폰 포렌식 도구가 전무하기 때문에 본 논문을 통해 구현된 포렌식 도구는 스마트폰 포렌식 기술 발전에 기여를 할 것으로 기대된다.

Ⅱ. 관련 연구

1. 스마트폰 운영체제 시장 동향

가트너(Gartner) 보고서에 따르면 2010년 3사분기를 기준으로 심비안이 표 1에서와 같이 전체 스마트폰 운영체제 시장의 36.6%를 차지하며 시장 점유율 1위를 차지하고 있으며, 그 뒤를 이어 안드로이드가 25.5%로 시장 점유율 2위를 차지하고 있다[8]. 여기서 주목할 만한 사항은 심비안은 해마다 시장 점유율에서 감소 추세를 보이고 있는 반면, 안드로이드는 급속한 시장 점유율의 증가를 보이고 있다는 점이다.

현재 국내 스마트폰 시장만을 고려할 때, 안드로이드폰은 가장 늦게 우리나라에 공급되었음에도 불구하고 아이폰과 심비안폰, 윈도우즈 모바일폰의 시장 점유율을 앞서 나가고 있다.

표 1. 스마트폰 운영체제 시장 현황
Table 1. Smartphone OS Market share

Company	2010년 3사분기 판매량(천 대)	2010년 3사분기 시장 점유율(%)	2009년 3사분기 판매량(천 대)	2009 3사분기 시장 점유율(%)
Symbian	29,480.1	36.6	18,314.8	44.6
Android	20,500.0	25.5	1,424.5	3.5
iOS	13,484.4	16.7	7,040.4	17.1
Research In Motion	11,908.3	14.8	8,522.7	20.7
Microsoft Windows Mobile	2,247.9	2.8	3,259.9	7.9
Linux	1,697.1	2.1	1,918.5	4.7
Other OS	1,214.8	1.5	612.5	1.5
Total	80,532.6	100.0	41,093.3	100.0

2. 안드로이드 포렌식 동향

일반적으로 안드로이드 스마트폰에서 정보를 추출하기 위한 포렌식은 네 가지 방법을 통해 가능하다[4].

- SD card 분석 : 거의 모든 안드로이드 디바이스는 SD card 저장 공간을 지원한다. 이 공간에는 애플리케이션에 사용되는 데이터가 저장된다. SD card는 FAT32 파일 시스템을 사용하므로 일반적인 방법으로도 분석이 가능하다.
- 논리적 분석 : 안드로이드 애플리케이션을 이용하는 방법으로, 안드로이드 포렌식 애플리케이션을 설치하여 스마트폰에 저장된 정보를 SD card에 저장한다. 이 방법을 통해서 브라우저, 통화내역, 연락처, 멀티미디어(메타데이터), MMS, SMS, 일정, 애플리케이션 정보를 얻을 수 있다.
- 물리적 분석 : 더 자세한 분석이 요구될 경우, 리눅스 dd 명령어를 이용해서 NAND 메모리의 덤프(Dump) 이미지를 분석하는 방법으로 관리자 권한이 필요하다. 이 방법은 다양한 사용자 데이터 파티션의 이미지를 획득할 수 있다. 이 파티션들은 YAFFS2 파일 시스템을 사용하며, 아직 상용 포렌식 도구는 없다.
- 칩 분리 : 플래쉬(Flash) 메모리를 물리적으로 분리하고 리더기(Reader)를 이용하여 데이터를 획득하는 방법이다. 리더기는 포렌식 목적의 장비가 아니기 때문에 법정에서 포렌식 무결성 문제가 발생할 수 있다.

아직까지 국내에서는 스마트폰 포렌식을 위한 도구가 개발되지 않았지만, 국외에서는 이미 스마트폰의 활성화로 스마트폰 포렌식을 위한 다양한 도구들이 개발되어 왔다. Cellebrite사에서 개발된 UFED(Universal Forensic Extraction Device) 도구는 그림 1과 같은 하드웨어적인 장비를 이용해서 스마트폰에 저장된 정보를 추출할 수 있으며, 메모리 덤프 이미지를 획득하여 쉽게 분석할 수 있는 소프트웨어 도구를 함께 제공한다[9].

그림 1. UFED 장비들
Fig. 1. UFED devices

이 도구를 통해서 Palm OS, Microsoft Windows Mobile, Blackberry, Symbian, iPhone, Android가 탑재된 스마트폰들의 정보를 추출할 수 있다. 추출할 수 있는 정보로는 연락처, SMS, 통화내역, 오디오, 비디오, 이미지, 벨소리, 전화기 정보 등이 있다.

Oxygen Forensic Suite 2010은 통합적인 포렌식 프로그램으로 다양한 플랫폼의 스마트폰 정보들을 종합적으로 확인할 수 있는 소프트웨어이다[10]. 플랫폼별로 획득할 수 있는 정보들에는 차이가 있으며, 안드로이드 플랫폼 기반에서는 그림 2와 같이 문자 메시지 정보를 포함하여, 연락처, 일정, 통화내역에 관한 정보 등을 획득할 수 있다.

Folder	Remote Party	Phone Number	Time Stamp	Text
SMS - Inbox	김	010	2010-07-05 오후 4:02:26	미
SMS - Sent	김	010	2010-07-05 오후 3:58:49	아
SMS - Inbox	김	010	2010-07-05 오후 3:57:03	혈
SMS - Inbox	11	114	2010-07-05 오후 12:02:24	SK
SMS - Inbox	02	022	2010-07-05 오전 8:05:01	우
SMS - Inbox	02	022	2010-07-05 오전 8:04:58	우
SMS - Sent	장	010	2010-07-04 오전 9:06:07	가
SMS - Inbox	장	010	2010-07-03 오후 5:54:24	오
SMS - Sent	장	010	2010-07-03 오후 5:54:02	아

그림 2. 추출된 문자 메시지 정보
Fig. 2. Extracted SMS message

그 외에도 Paraben 사의 Device Seizure와 Micro Systemation사의 XRY 프로그램을 통해서도 포렌식이 가능하다[11,12].

3. 안드로이드 프레임워크

안드로이드는 운영체제와 미들웨어(Middleware) 및 주요 애플리케이션을 포함하는 모바일 디바이스를 위한 소프트웨어 모음을 말한다. 안드로이드는 자바 언어를 사용하여 안드로이드 플랫폼 상에 응용프로그램 개발을 하는데 필요한 도구들과 API를 제공한다. 안드로이드 아키텍처를 구성하는 주요 컴포넌트들은 그림 3과 같다[13].

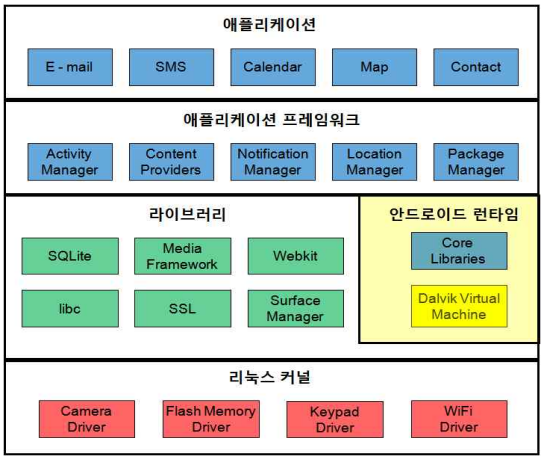


그림 3. 안드로이드 아키텍처
Fig. 3. Android Architecture

- 애플리케이션(Applications) : 안드로이드가 기본적으로 제공하는 이메일 클라이언트, SMS 프로그램, 달력, 지도, 브라우저, 연락처 및 기타 다양한 애플리케이션이 존재한다.
- 애플리케이션 프레임워크(Application Framework) : 애플리케이션에 제공되는 API 집합을 포함한다. 개발자는 안드로이드에 이미 탑재된 코어 애플리케이션에 사용된 것과 동일한 프레임워크 API를 완전하게 접근할 수 있다.
- 라이브러리(Libraries) : 안드로이드 시스템의 다양한 컴포넌트들에 의해 사용되는 C/C++ 라이브러리들을 포함하고 있다.
- 안드로이드 런타임(Android Runtime) : 모든 안드로이드 애플리케이션은 자신의 프로세스에서 Dalvik 가상머신의 자체 인스턴스를 가지고 구동된다. Dalvik은 디바이스가 다중 가상 머신을 효과적으로 구동할 수 있도록 만들어졌다. Dalvik 가상머신은 최소의 메모리를 사용하도록 최적화된 달빅 실행(.dex) 포맷으로 파일을 실행한다.
- 리눅스 커널(Linux Kernel) : 안드로이드는 보안, 메모리 관리, 프로세스 관리, 네트워크 스택 및 드라이버 모델과 같은 코어 시스템 서비스 용 리눅스 버전에 의존한다. 커널은 하드웨어와 나머지 소프트웨어 스택 사이의 추상 계층으로서 동작한다.

안드로이드 애플리케이션은 다음과 같은 네 가지 유형의 컴포넌트로 구성된다.

- 액티비티(Activity) : 액티비티는 사용자와 상호작용할 수 있는 하나의 사용자 인터페이스를 나타내며, 하나의 애플리케이션은 한 개 이상의 액티비티로 구성된다.
- 서비스(Service) : 서비스는 사용자 인터페이스를 갖지 않고 백그라운드에서 실행된다. 서비스는 미디어 플레이어와 같이 액티비티가 사라진 후에도 동작해야 하는 경우에 사용된다.
- 브로드캐스트 리시버(Broadcast Receivers) : 브로드캐스트 리시버는 브로드캐스트 공지(announcement)를 수신하고 응답한다. 주로 핸드폰의 상태가 바뀌었거나 특정 이벤트가 발생했음을 알려준다. 사용자 인터페이스를 갖지 않으며, 특정 브로드캐스트가 발생했을 때 특정 액티비티를 시작하거나 NotificationManager를 통해 사용자에게 알려주는 역할을 한다.
- 컨텐트 프로바이더(Content Providers) : 컨텐트 프로바이더는 다른 애플리케이션에게 사용 가능한 특정 애플리케이션의 데이터 집합을 만든다. 데이터는 파일시스템 터는 QLite DB를 비롯해서 여러 가지 의미 있는 방식으로 저장될 수 있다. 주로 애플리케이션 간의 데이터를 공유하기 위해 사용된다.

4. 루팅된 안드로이드폰 정보 추출

안드로이드 애플리케이션은 각자 자신만의 공간에 데이터베이스를 생성하며, 애플리케이션이 데이터베이스를 생성하면 그 데이터베이스는 /data/data/[package_name]/databases/ 디렉터리에 존재하게 된다.

일반적으로 애플리케이션 간에는 다른 애플리케이션의 디렉터리에 대한 접근 권한이 없기 때문에 사용자가 데이터베이스에 직접 접근하는 것이 불가능하지만 루팅된 폰에서는 직접 접근이 가능하다. 안드로이드폰에서의 루팅이란 안드로이드 운영체제의 관리자 권한을 얻는 행위를 의미한다. 안드로이드 폰의 관리자 권한을 얻게 되면, 일반 사용자 권한으로는 삭제할 수 없는 내장 애플리케이션을 임의로 삭제할 수도 있으며, 안드로이드 폰 시스템 설정을 임의로 변경할 수 있기 때문에 시스템 성능을 향상시킬 수 있다는 장점이 있다. 하지만 루팅된 폰은 A/S 지원이 중단되며, 스마트폰 보안상 위험성에 노출될 가능성이 더욱 높아진다.

데이터베이스 정보는 안드로이드 버전에 따라 상이하며, 본 연구에서는 안드로이드 버전 1.6 에뮬레이터 기준으로 데이터베이스 정보를 추출하였다. 통화 내역과 연락처 정보는 /data/data/com.android.providers.contacts/databases 디렉터리에 contacts.db 파일로 존재하며, Android Debug

Bridge(adb)의 shell에서 sqlite3 명령어를 통해 저장된 데이터를 확인할 수 있다. adb는 안드로이드 툴킷(Toolkit)에 포함되어 있는 도구로서 개발자가 장비의 상태를 관리할 때 유용하게 사용할 수 있다.

그림 4는 .table 명령어를 통해 데이터베이스 파일에 존재하는 모든 테이블을 확인한 결과이며, 그림 5는 select 문을 통해 people 테이블에 저장된 정보를 출력한 결과이다.

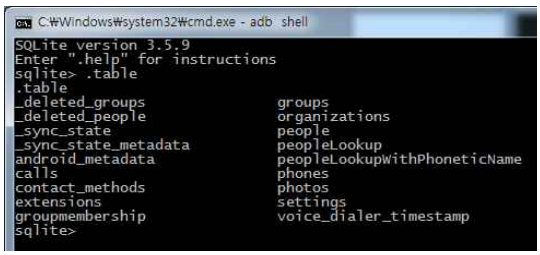


그림 4. .table 명령어 실행 결과
Fig. 4. Result of .table command execution

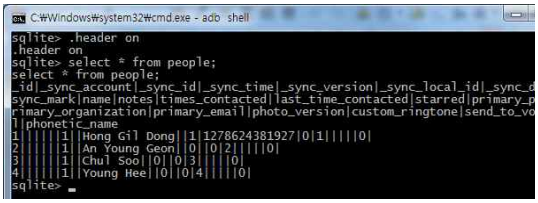


그림 5. people 테이블에 저장되어 있는 레코드들
Fig. 5. Records stored in people table

결과를 통해 안드로이드폰에 저장된 연락처 정보를 확인할 수 있으며, select 문을 통해 calls 테이블을 검색하면 통화 내역을 확인할 수 있다. 그 외에도 mmssms.db에는 문자메시지와 관련된 테이블들이 존재하여 발신 및 수신 메시지에 대한 정보를 확인할 수 있으며, browser.db 파일을 통해서는 북마크, 웹 검색 이력을 확인할 수 있다. 하지만 /data 파티션은 관리자 권한을 가져야 접근이 가능하기 때문에 데이터베이스 파일에 접근하기 위해서는 안드로이드폰을 루팅해야만 한다.

III. 논리적 분석 기반 포렌식 도구

스마트폰의 제조사와 탑재된 운영체제에 따라 안드로이드폰의 루팅 기법은 달라진다. 따라서 새로운 루팅 기법이 공개되지 않은 안드로이드폰에서도 신속하게 포렌식 정보를 획득하는 것은 매우 중요하다. 본 장에서는 루팅되지 않은 안드로이드폰으로부터 정보를 추출하는 포렌식 도구에 관해 기술한다.

1. 포렌식 도구 설계

루팅되지 않은 안드로이드폰 상에서 포렌식 정보를 추출하고, 추출된 정보를 포렌식 과정에서 효과적으로 이용할 수 있도록 하기 위한 포렌식 도구 구현에 있어서 다음과 같은 사항들이 요구된다.

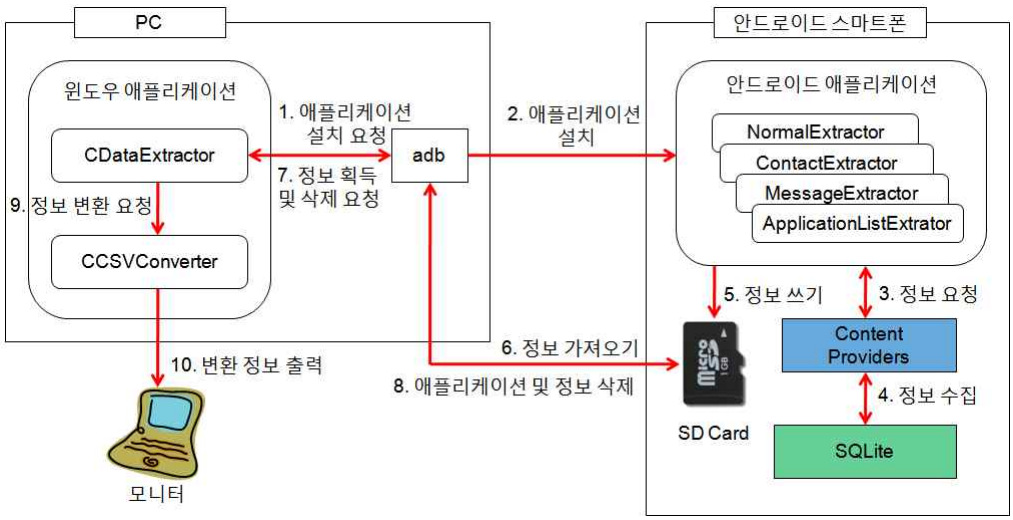


그림 6. 시스템 구성
Fig. 6. System Architecture

표 2. 애플리케이션 구성요소
Table 2. Application components

애플리케이션	클래스	설명
안드로이드 애플리케이션	NormalExtractor	통화내역, 이미지, 비디오, 오디오, 브라우저, 일정 정보 추출
	ContactExtractor	연락처 정보 추출
	MessageExtractor	SMS, MMS 정보 추출
	ApplicationListExtractor	설치된 애플리케이션 정보 추출
윈도우즈 애플리케이션	CDataExtractor	안드로이드 애플리케이션 설치 및 삭제, CSV 파일 수신 및 삭제
	CCSVConverter	CSV 파일을 문자열로 변환

1) 루팅되지 않은 안드로이드폰으로부터 포렌식 관련 정보를 추출하기 위한 방법 필요하다.
2) 추출된 정보가 포렌식 분석에 효과적으로 사용되기 위한 가공 절차가 필요하다.
3) 추출된 정보는 스마트폰의 협소한 액정 화면에서의 분석이 매우 어렵기 때문에 PC 상에서의 분석이 가능해야 한다.
4) 추출된 정보 집합으로부터 신속하게 증거를 획득하기 위해 검색 기능이 제공되어야 한다.
5) 포렌식 도구의 설치 및 실행과 정보 추출 과정이 용이해야 한다.
6) 추출된 정보(메타데이터)로부터 연관된 리소스(미디어 파일)를 바로 확인할 수 있는 기능을 제공해야 한다.

그림 6은 구현하고자 하는 포렌식 도구의 시스템 구성이며, PC에 설치되는 윈도우즈 애플리케이션과 스마트폰에 설치되는 안드로이드 애플리케이션으로 구성된다. 윈도우즈 애플리케이션은 안드로이드폰에 안드로이드 애플리케이션을 설치하고 실행하는 역할을 하며, 안드로이드폰에서 실행된 안드로이드 애플리케이션은 정보를 수집하여 SD 카드에 저장하는 역할을 한다. 이후에 윈도우즈 애플리케이션은 SD 카드에 생성된 파일을 PC로 가져온 뒤, 파일들을 읽어 형식화 된 형태로 모니터에 디스플레이 한다.

안드로이드 애플리케이션을 안드로이드폰에 설치하고 실행한 후, 추출된 정보를 PC로 가져오는 기능은 adb 도구를 이용하여 구현할 수 있다. 안드로이드는 기본적으로 각각의 애플리케이션들을 샌드박싱(Sandboxing)하기 때문에 다른 애플리케이션의 데이터에 접근하는 것이 불가능하다. 만일, 한 애플리케이션이 다른 애플리케이션에 데이터를 제공하고자 한다면 컨텐트 프로바이더를 이용해야 한다. 컨텐트 프로바이더는 애플리케이션 사이에서 데이터를 공유할 수 있는 방법으로, 안드로이드는 폰에서 사용하는 공통의 데이터(연락처, 미디어 파일, 문자 메시지)에 대한 컨텐트 프로바이더를 제공한다. 따라서 구현하고자 하는 포렌식 도구는 그림 6에서와 같이 안드로이드가 자체적으로 제공하는 컨텐트 프로바이더를 이용하여 데이터베이스로부터 정보를 가져온다.

컨텐트 프로바이더를 이용하여 추출되는 정보에는 레코드들에 대한 컬럼 정보가 없다. 따라서 추출된 정보는 Extractor 계열의 클래스를 구현한 후, 이를 통해 재가공 절차를 거쳐 SD 카드에 저장된다.

표 2는 애플리케이션 별 세부 구성요소를 나타낸다. 안드로이드 애플리케이션에는 데이터베이스에 저장된 정보들을 추출하기 위한 Extractor 계열의 클래스들이 존재하며, 이 클래스들은 정보를 추출한 후에 CSV 파일 형태로 SD 카드에 저장한다. 윈도우즈 애플리케이션의 CDataExtractor 클래스는 안드로이드 애플리케이션을 설치 및 실행하며, 안드로이드폰에서 생성된 CSV 파일을 가져오는 역할을 한다. 그리고 CSV 파일은 CCSVConverter 클래스를 통해 문자열로 변환되어 출력된다.

표 3. 주요 정보들의 URI
Table 3. URI for main information

URI	설명
Contacts.CONTENT_URI Contacts.Data.CONTENT_URI	연락처
Call.CONTENT_URI	통화내역
content://sms	SMS
content://mms	MMS
Images.Media.EXTERNAL_CONTENT_URI	이미지
Video.Media.EXTERNAL_CONTENT_URI	비디오
Audio.Media.EXTERNAL_CONTENT_URI	오디오
Browser.BOOKMARKS_URI	브라우저
content://com.android.calendar/events	일정

2. 안드로이드 애플리케이션 구현

안드로이드의 컨텐트 프로바이더는 데이터베이스에 저장된 데이터에 접근할 수 있는 URI를 제공한다. 따라서 안드로이드 애플리케이션은 URI를 통해 데이터를 요청한 후, 요청에 의해 리턴된 Cursor 객체를 이용하여 데이터에 순차적으로 접근할 수 있다. 안드로이드폰에서 획득 가능한 주요 정보들의 URI는 표 3과 같다.

안드로이드 애플리케이션을 구성하는 Extractor 계열의 클래스들은 데이터베이스에 저장된 정보를 추출하는 클래스들이다.

```
/*** NormalExtractor, ContactExtractor, MessageExtractor
                클래스에서의 정보 추출 ***/
cursor = activity.managedQuery(정보 추출을 위한 URI, 추
출할 컬럼 목록, null, null, 정렬 유형);
retVal = cursor.moveToFirst(); // 커서의 처음으로 이동
if(retVal)
  csvWriter = new CSVWriter(파일명); // CSV 파일 저장

/*** ApplicationListExtractor 클래스에서의 정보 추출 ***/
pm = activity.getPackageManager(); // 패키지 매니저 객
체 얻기
Intent intent = new Intent(Intent.ACTION_MAIN, null); //
액티비티 조건 설정
intent.addCategory(Intent.CATEGORY_LAUNCHER); // 액
티비티 조건 설정
list = pm.queryIntentActivities(intent, PackageManager.PE
RMISSION_GRANTED); // 액티비티 목록 추출
csvWriter = new CSVWriter(파일명); // CSV 파일 저장
```

그림 7. 정보추출을 위한 클래스 코드
Fig. 7. Class code for extracting information

NormalExtractor, ContactExtractor, MessageExtactor 클래스에서는 그림 7과 같이 managedQuery 메소드를 통해서 정보를 추출하며, 정보 추출을 위한 URI는 표 3에 명시되어 있는 URI를 사용한다. 그리고 ApplicationListExtractor 클래스의 경우에는 패키지 매니저 객체를 구한 후, 인텐트(Intent)를 이용해서 가져올 액티비티의 조건을 설정하여 액티비티 목록을 추출한다.

표 4. 연락처 클래스와 기호상수
Table 4. Contacts classes and symbolic constants

클래스	기호상수	설명
CommonDataKinds.Phone(전화번호)	NUMBER	전화번호
	TYPE	타입(집, 직장 등)
CommonDataKinds.Email(이메일)	DATA	이메일 주소
	TYPE	타입(집, 직장 등)
CommonDataKinds.Im(메신저)	DATA	메신저 주소
	TYPE	타입(집, 직장 등)
CommonDataKinds.StructurePostal(주소)	FORMATTED_ADDRESS	형식화 된 주소
	TYPE	타입(집, 직장 등)
	POSTCODE	우편번호
	COUNTRY	국가 정보
	CITY	도시 정보
	STREET	거리 정보
CommonDataKinds.Website(웹 사이트)	URL	웹 사이트 URL
	TYPE	타입(홈페이지, 블로그 등)
CommonDataKinds.GroupMembership(그룹) CommonDataKinds.Groups(그룹)	GROUP_ROW_ID	그룹 ID
	TITLE	그룹 이름
	NOTES	그룹에 대한 메모
	SUMMARY_COUNT	그룹에 포함된 연락처 개수
CommonDataKinds.Organization(조직)	COMPANY	회사 정보
	TITLE	직위
	DEPARTMENT	부서
	OFFICE_LOCATION	사무실 위치
CommonDataKinds.Event(이벤트)	START_DATE	이벤트 날짜
	TYPE	타입(기념일, 생일 등)
CommenDataKinds.Note(메모)	NOTE	메모 정보
CommonDataKinds.Photo(사진)	PHOTO	사진 객체

표 5. SMS, MMS, 일정 컬럼
Table 5. SMS, MMS, and calendar columns

종류	컬럼명	설명
SMS	_id	메시지의 primary key
	address	전화번호
	date	송수신 시각
	read	읽었는지 여부
	subject	제목
	body	문자 내용
MMS	_id	메시지의 primary key
	date	송수신 시각
	m_size	메시지 크기
	m_type	메시지 종류
	read	읽었는지 여부
	sub	메시지 제목
	text	메시지 문자열
	address	전화번호
일정	title	일정 이름
	eventLocation	일정 장소
	description	일정 설명
	hasAlarm	알림 설정 여부

추출된 정보는 cursor를 통해 레코드 형태로 접근이 가능한데, 각각의 레코드로부터 각 필드로의 접근을 수행하기 위해서는 getstring() 메소드를 이용할 수 있다. 즉, getstring() 메소드의 입력 값으로 정수 값을 전달하면 해당 열 정보를 구할 수 있다. 표 4는 URI를 통해 추출 가능한 연락처 정보들의 요약분이며, 클래스에 미리 정의되어 있는 기호상수 값을 그림 7의 추출할 컬럼 목록에 명시하면 해당 정보를 추출할 수 있다. 그룹 정보의 경우에는 GROUP_ROW_ID를 이용해서 Groups 클래스에 정의된 CONTENT_URI를 통해 정보를 요청하면 해당 그룹의 상세 정보를 추출할 수 있다.

표 5는 SMS와 MMS, 일정정보들의 요약분을 나타낸다. 이 정보들은 안드로이드의 공식 API에 따로 정의되어 있지 않기 때문에 스마트폰 제조사 마다 다를 수 있다. MMS의 경우 3개의 테이블을 통해 정보를 추출할 수 있는데, 우선 content://mms로 접근하여 메시지에 대한 정보들과 메시지의 ID를 획득한다. 획득된 ID를 이용하여 content://mms/[ID]/part에 접근한 후, MMS에 저장된 메시지 문자열을 추출할 수 있으며, content://mms/[ID]/address를 통해 전화번호(address)를 추출할 수 있다.

마지막으로 표 6은 통화내역, 이미지, 비디오, 오디오, 브라우저 정보들의 요약분을 나타낸다.

표 6. 통화내역, 이미지, 비디오, 오디오, 브라우저 클래스와 기호상수
Table 6. Calls, image, video, audio, browser classes and symbolic constants

클래스	기호상수	설명
CallLog.Calls(통화내역)	CACHED_NAME	저장된 번호일 경우 저장된 이름
	CACHED_NUMBER_TYPE	전화번호 종류
	DATE	통화한 시각
	DURATION	통화한 시각
	NUMBER	통화한 전화번호
	TYPE	타입(수신, 발신, 부재중)
MediaStore.Images.Media(이미지) MediaStore.Video.Media(비디오)	DATE_TAKEN	이미지(동영상)가 생성된 시각
	DESCRIPTION	이미지(동영상) 설명
	LATITUDE	이미지(동영상)가 생성된 위도
	LONGITUDE	이미지(동영상)가 생성된 경도
	DATE_MODIFIED	수정된 시각
	TITLE	컨텐츠 제목
MediaStore.Audio.Media(오디오)	ALBUM	앨범 정보
	ARTIST	아티스트 정보
	COMPOSER	작곡가 정보
	YEAR	녹음된 연도
	DATE_MODIFIED	수정된 시각
	TITLE	컨텐츠 제목
Browser.BookmarkColumn(브라우저)	BOOKMARK	북마크/방문기록 여부
	CREATED	생성된 시각
	DATE	마지막으로 방문한 시각
	TITLE	이름
	URL	주소
	VISITS	방문횟수

```
executeADBCommand(L"/c adb install ₩"Logical Forensics.apk₩" 2 > result");  // 폰에 애플리케이션 설치
executeADBCommand(L"/c adb shell am start -a android.intent.action.MAIN -n
kr.ac.ssu.logicalforensics/kr.ac.ssu.logicalforensics.activities.LogicalForensicsActivity");  // 애플리케이션 실행

/* 1초에 한 번씩 주기적으로 작업이 완료되었는지 검사 */
do{
    sleep(1000);
    executeADBCommand(L"/c adb pull /sdcard/extract_complete .");
}while(!CFile::GetStatus(L"extract complete", status));

/* CSV 파일 수신 */
executeADBCommand(L"/c adb pull /sdcard/sms.csv .");
```

그림 8. CDataExtractor 클래스 코드
Fig. 8. CDataExtractor Class code

3. 윈도우즈 애플리케이션 구현

윈도우즈 애플리케이션은 스마트폰에 안드로이드 애플리케이션을 설치 및 실행하며, 스마트폰에 생성된 획득 정보를 PC로 가져오기 위해 adb를 이용한다.

그림 8에서와 같이 윈도우즈 애플리케이션 내의 CDataExtractor 클래스는 adb install 명령어를 통해 스마트폰에 안드로이드 애플리케이션을 설치하며, adb shell 명령어로 애플리케이션에 있는 메인 액티비티를 실행한다. 그리고 안드로이드 애플리케이션이 정보를 추출 작업을 완료하였는지 조사하기 위해서 1초 간격으로 작업 완료를 나타내는 파일(extract_complete)이 생성되었는지를 검사하며, 이 파일이 생성되었을 경우 SD 카드로부터 추출된 정보가 저장된 CSV 파일을 PC로 가져온다. 이후 CCSVConverter 클래스는 CSV 파일로부터 정보를 문자열로 형식화하여 출력한다.

IV. 구현결과 및 활용

1. 실행환경 구성

구현된 포렌식 도구가 실행되는 PC에는 포렌식 도구 실행에 필요한 파일들과 안드로이드폰으로부터 수집된 파일을 보관하는 디렉터리가 그림 9와 같이 존재한다.

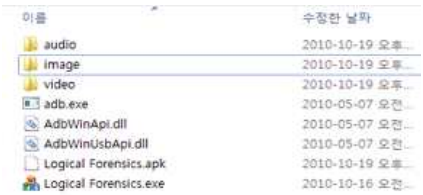


그림 9. PC 환경 구성
Fig. 9. PC environment configuration

- adb 인터페이스를 실행하기 위한 파일

애플리케이션의 설치, 삭제, 실행을 위한 adb.exe, AdbWinApi.dll, AdbWinUsbApi.dll 파일

- 안드로이드폰에 설치될 애플리케이션 패키지

안드로이드폰에 설치되어 정보를 수집하기 위한 Logical Ferensics.apk

- 미디어 파일을 저장할 디렉터리

포렌식 도구 실행 후 미디어 파일을 가져와 보관하기 위한 audio, image, video 디렉터리

- 윈도우즈용 프로그램

안드로이드 애플리케이션 설치, 삭제 및 포렌식 결과를 출력하기 위한 Logical Forensics.exe 파일

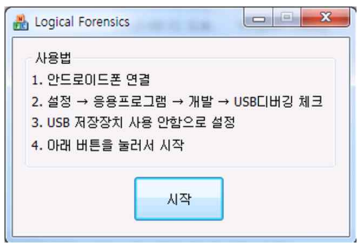


그림 10. 포렌식 도구의 실행
Fig. 10. Execution of forensic tool

그림 10은 실행 환경이 구성된 후 Logical Forensics.exe 파일을 실행했을 때 나타나는 포렌식 도구의 초기화면이다. 구현된 포렌식 도구는 초기화면의 시작버튼 원클릭으로 안드로이드 애플리케이션의 설치 및 실행과 추출 정보를 PC로 가져오는 모든 작업을 수행한다.

2. 시나리오를 통한 포렌식 도구 활용

구현된 포렌식 도구는 스마트폰을 이용한 범죄 사건의 증거를 신속히 획득하는데 사용될 수 있다. 다음 시나리오는 범

죄 사건의 일례를 통해 사건 수사에 필요한 증거를 획득하는 방법을 보여준다. 시나리오에서의 스마트폰 획득과 분석은 포렌식 무결성 유지를 위한 절차[2]에 따라 수행되었다고 가정한다.

- 시나리오

조직폭력배들이 어떤 사람에게 계획적인 상해를 입히는 사건이 발생하였다. 수사를 하던 경찰은 조직원 중 한명을 검거하는데 성공하였고, 검거된 용의자가 사용하고 있는 안드로이드폰을 입수했다. 이에 경찰은 포렌식 도구를 이용하여 사건에 대한 증거를 획득하려 한다.

그림 11. 수집된 연락처 정보
Fig. 11. Collected contact information

경찰은 폭력사건에 가담한 공범들을 찾기 위해 그림 11과 같이 용의자의 연락처 정보를 조회하였다. 경찰은 연락처에서 그룹이 지정된 것을 확인하였고, 그림 12와 같이 조직의 이름으로 추정되는 "아무개" 그룹만을 따로 조회하였다.

그림 12. 수집된 특정 그룹 정보
Fig. 12. Collected specific group information

이후 경찰은 그림 13과 같이 그룹에 속한 사람들의 통화 및 문자 내역에 대해서 조회를 시작하였다.

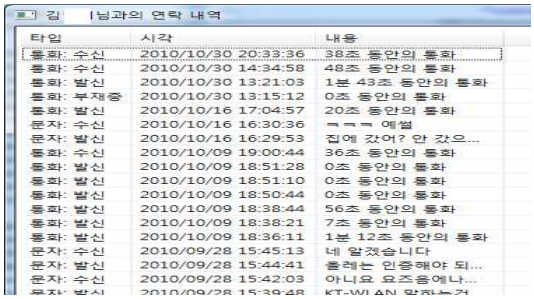


그림 13. 수집된 통화 및 문자내역
Fig. 13. Collected calls and messages information

경찰은 통화한 시각이나 문자메시지 내용을 보면서 공범으로 추정되는 사람들을 하나씩 용의자 명단에 추가하였으며, 연락처 정보에 포함된 사진들을 통해 신원을 파악하는데 성공하였다. 공범자들이 범행을 저지르기 위해서는 통화나 문자메시지 사용이 잦아진다는 것에 착안하여 경찰은 그림 14와 같이 통화 및 문자메시지 횟수에 대한 통계 자료를 이용해 확실한 용의자를 추려낸다.

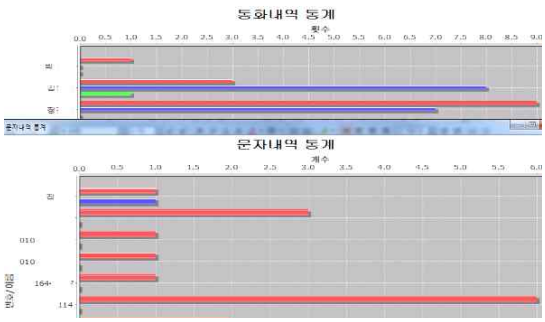


그림 14. 통화 및 문자메시지에 대한 통계자료
Fig. 14. statistics for calls and messages

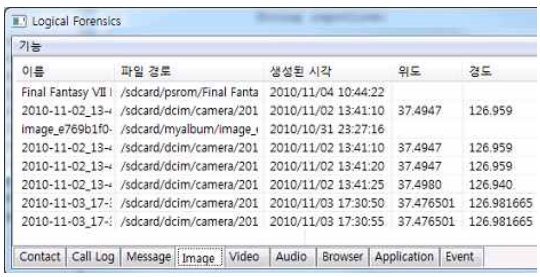


그림 15. 수집된 사진 정보
Fig. 15. Collected photo files information

경찰은 그림 15와 같이 추가적으로 스마트폰을 통해 촬영한 사진들도 확인하였다. 스마트폰을 통해 촬영한 사진에는 생성된 위치정보(위도, 경도)가 저장된다.

그림 16. 구글맵을 통한 위치 알림
Fig. 16. Location notification using Google Maps

그림 16은 위치정보가 저장되어 있는 사진 목록을 더블 클릭했을 때 나타나는 구글맵(Google Maps)을 나타낸다. 따라서 경찰은 용의자가 특정 시각에 어느 위치에 있었는지, 그리고 범행이 이루어진 장소가 어디인지를 확인할 수 있다.

V. 결 론

최근 스마트폰 이용률 증가에 따라 국내에서도 스마트폰 관련 연구가 활발히 진행되고 있다. 그 중에서도 주목할 만한 것은 최근 출시된 안드로이드폰의 급속한 시장 점유율 증가이다. 현재 스마트폰을 위한 보안 연구 개발이 보안업체 및 연구기관에서 수행되고 있지만, 아직까지 안드로이드 스마트폰 포렌식을 위한 도구는 전무한 상태이다. 이에 본 논문에서는 논리적 분석 기반의 안드로이드 스마트폰 포렌식 도구를 개발하였다. 시나리오에서 살펴본 바와 같이 구현된 포렌식 도구는 안드로이드폰의 루팅 절차가 필요하지 않으며, 포렌식 수사의 용이성을 위한 직관적인 인터페이스를 제공하며, 미디어와 관련된 정보를 클릭할 경우 바로 관련 파일을 확인할 수 있다. 아울러 신속한 분석에 필요한 검색 기능을 제공하며, 효과적인 분석을 위한 통계 결과도 함께 제공하여 포렌식에 소요되는 시간을 더욱 단축시킬 수 있다.

참고문헌

[1] Jin-Won Sung, Eun-Ju Back, Chang-Uk Park, Yeog Kim, and Sang-Jin Lee, “The Design and Implementation of the tool to analyze Mobile Data : Mobile Data Analyzer”, Journal of Digital Forensics, Vol. 1, No. 1, pp. 63-77, Nov. 2007.

[2] Gyu-an Lee, Dae-woo Park, and Young-tae Shin, “A Study on Forensic Integrity Proof Standard a Cellular Phone Confiscation Criminal Investigation”, Journal of Korea Information and Communications Society, Vol. 33, No. 6, pp. 512-519, Jun. 2008.

[3] DongGuk Kim, SeongYong Jang, WonYoung Lee, YongHo Kim, and Changhyun Park, “An Effective Control Method for Improving Integrity of Mobile Phone Forensics”, Journal of The Korea Institute of Information Security & Cryptology, Vol. 19, No. 5, pp. 151-166, Oct. 2009.

[4] An Introduction to Android Forensics, http://www.dfinews.com/article/introduction-android-forensics?pid=974

[5] Android, Incident Response and Forensics, *http://www1.webng.com/dhruv/material/android_report.pdf*

[6] Me G, Rossi M, "Internal forensic acquisition for mobile equipments", Proceedings of the international parallel and distributed processing symposium, pp. 1-7, 2008.

[7] Savoldi, A., Gubian, P., "Symbian Forensics: An Overview", Proceedings of IIHMSP, pp. 529-533, 2008.

[8] *http://www.gartner.com/it/page.jsp?id=1466313*

[9] UFED, *http://www.cellebrite.com*

[10] Oxygen Forensic Suite 2010, *http://www.oxygen-forensic.com*

[11] Device Seizure, *http://www.paraben.com*

[12] XRY, *http://www.msab.com*

[13] What is Android, *http://developer.android.com/guide/basics/what-is-android.html*

저 자 소 개

김 익 수

2000: 숭실대학교 컴퓨터학부 공학사
2002: 숭실대학교 컴퓨터학과 공학석사
2008: 숭실대학교 컴퓨터학과 공학박사
2009-현 재: 숭실대학교 컴퓨터학부 조교수
관심분야: 시스템 보안, 네트워크 보안, 모바일 보안
Email : iksuplorer@ssu.ac.kr

안 영 건

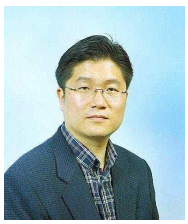
2009: 숭실대학교 컴퓨터학부 공학사
2011: 숭실대학교 컴퓨터공학과 공학석사
현 재: (주)케이사인 주임연구원
관심분야: 모바일 보안
Email : bhythmmaker@gmail.com

이 정 현

1993: 숭실대학교 전자계산학과 학사
1995: 숭실대학교 전자계산학과 석사
2005: University of California, Irvine 컴퓨터학과 박사
1995-2001: 한국전자통신연구원 연구원
2000-2001: 미국 표준기술연구원 (NIST) 객원연구원
2005-2008: 삼성종합기술원 수석연구원
2008-현 재: 숭실대학교 컴퓨터학부 조교수
관심분야: 모바일 보안, 네트워크 보안
Email : jhyi@ssu.ac.kr

양 승 제

1997: 한양대학교 컴퓨터공학과 공학사
1999: 한양대학교 컴퓨터공학과 공학석사
2003: 한양대학교 컴퓨터공학과 공학박사
2004: 뉴욕주립대 박사후과정
2005-2008: LG전자 MC연구소 책임연구원
2008-현재: ETRI 부설연구소 연구원
관심분야: 정보보호, 디지털 포렌식
Email : sjyang@ensec.re.kr



김 명 호

1989 : 숭실대학교 컴퓨터학부 학사
1991 : 포항공과대학교 전자계산학과 공학석사
1995 : 포항공과대학교 전자계산학과 공학박사
1995 : 한국전자통신연구소 선임연구원
1998, 2006 : 미국 테네시주립대 교환교수
1995-현재 : 숭실대학교 컴퓨터학부 교수
관심분야 : 분산/병렬 컴퓨팅, 그리드, 웹서비스, BI, 보안
Email : kmh@ssu.ac.kr