Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.28, NO.1, Feb. 2018 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2018.28.1.95

모바일 포렌식 증거 수집방안 연구: 제조사 백업 앱 기반 데이터 획득 기법*

최 재 원,† 김 승 주‡
고려대학교 정보보호대학원

A Study on Mobile Forensic Data Acquisition Method Based on Manufacturer's Backup Mobile App*

Jaewon Choi,† Seung-joo Kim‡
Graduate School of Information Security, Koea University

요 약

스마트폰의 보급으로 사용자의 다양한 개인정보가 실시간으로 스마트폰에 기록되고 있다. 사용자 데이터의 유실을 방지하고 분실 및 도난, 휴대폰 교체 등에 대응하기 위해 제조사들은 다양한 스마트폰 백업프로그램을 사용자에게 제공한다. 과거 PC 기반의 백업프로그램 뿐만 아니라 현재는 스마트폰에서 직접 실행이 가능한 백업용 모바일 앱을 기본 제공하는 추세이다. 디지털 포렌식 데이터 획득의 관점에서 획득이 가능한지 여부와 원본 데이터의 무결성을 손상되지 않았는지는 중요한 요소이다. 특히 파편화된 안드로이드 스마트폰의 경우 원본 데이터의 무결성을 손상시키지 않으면서 획득 가능하게하기 위한 연구가 다양하게 진행되고 있다. 하지만, 최근 보안이 강화되어 출시되는 스마트폰은 기존 연구된 데이터 획득 기법을 그대로 적용하기에는 어려운 한계점들이 존재한다. 따라서 본 논문에서는 제조사에서 기본으로 제공하는 백업용 모바일 앱을 이용하여 기존 데이터 획득 방법을 그대로 적용하기 어려운 최신 스마트폰을 대상으로 무결성을 훼손시키지 않고 사용자의 데이터를 획득하는 과정에 대해 설명한다.

ABSTRACT

With the widespread use of smartphones, various personal information of users is being recorded on a smartphone in real time. For the purpose of preventing the loss of important personal information of users, manufacturer provides a smartphone backup applications. Recently, not only backup programs for PC but also backup mobile apps for smart phones have been provided. From the point of view acquiring forensic data, it is important not to compromise the acquisition possibilities and the integrity of the original data. Especially, in the case of Android smartphones, various studies are being carried out to acquire the data without damaging the integrity of the original data. However, there are limitations to apply the existing research methods. In this paper, we describe the process of acquiring data using the backup mobile app provided by the manufacturer without compromising the integrity of the latest smartphone.

Keywords: digital forensic, data acquisition, android, smartphone, backup app

Received(11. 06. 2017), Modified(12. 20. 2017), Accepted(01. 02. 2018)

* 본 논문은 2018년 대한민국 교육부와 한국연구재단의 지원을 받아 수행된 연구임(NRF-2016S1A3A2924760)

† 주저자, jwonchoi@korea.ac.kr
‡ 교신저자, skim71@korea.ac.kr(Corresponding author)

I. 서 론

스마트폰의 급격한 보급과 휴대성, 개인화된 서비스의 제공으로 인해 개인의 일상생활과 관련한 다양한 정보는 스마트폰에 기록되어 보관되고 있다. 세계이동통신사업자협회(GSMA, Groupe Spécial mobile Mou Association)에 따르면 지난해 말 세계 스마트폰 보급률이 처음으로 50%를 넘어섰고 오는 2020년에는 75%까지 상승할 것으로 전망하고 있다. 특히 한국의 스마트폰 보급률은 85%(모바일 회선 5489만9789개중 4641만8474개)로 세계 최고 수준으로 나타났다[1]. 개인이 항상 휴대하고 있는 기기의 특성과 개인의 정보가 실시간으로 기록된다는 측면에서 법적 증거물로서의 모바일 포렌식의 중요성은 더욱 증가하고 있다.

그러나 기존에 연구된 모바일 포렌식 데이터 획득 기법은 안드로이드 7.0 누가(Android 7.0 Nougat)[2] 이상의 버전을 사용하는 최신 스마트폰을 대상으로는 적용하는데 한계점이 존재한다. 하드웨어 기반의 데이터 획득 기법은 데이터 획득에 반드시 필요한 JTAG(Joint Test Access Group)[3][5] 인터페이스가 양산 과정에서 제거된 상태로 출시되거나 디스크 암호화(full disk encryption)[6], trustzone[7], knox[8], secure/trusted boot[9], hardware root of trust[10] 등 강화된 보안 기능으로 인해 메모리에 저장된 데이터에 하드웨어적으로는 접근할 수 있는 방법이 없는 한계점이 존재한다. 특히 Chip-Off 기법[4]은 메모리를 하드웨어적으로 분해하여 데이터를 이미징하는 것이 가능하다 하더라도 분해 과정에서 하드웨어 기반 키가 유실되어 암호화된 사용자 데이터에 접근할 수 있는 방법이 없어지기 때문에 현재는 사용할 수 없다. 또한 부트로더 취약점에 기반한 물리적 획득 기법[11], 커스텀 리커버리 이미지에 기반한 물리적 획득 기법[12], ADB(Android Debug Bridge) Backup[13] 등 소프트웨어 기반의 데이터 획득 기법 역시 기존의 취약점이 패치된 상태로 출시되는 경우 기존 기법을 그대로 적용할 수 없고 제한적인 데이터에 한하여 접근이 가능하는 등 포렌식 데이터 획득 측면에서 제한적으로만 사용이 가능하다.

본 논문에서는 제조사가 기본 제공하는 백업용 모바일 앱을 분석하여 루트 권한을 획득하지 않은 상태에서 일반 사용자 권한으로는 접근이 불가능한 영역에 저장된 중요 데이터(메시지, 일정, 연락처, 통화기록 등)를 루팅없이 획득하는 방법을 서술한다. 특히 현재까지 물리적 데이터 획득 기법을 적용할 수 없고 전체 메모리 데이터의 이미징이 불가능한 최신 안드로이드 스마트폰을 대상으로 논문에서 설명한 기법을 적용하여 데이터를 획득할 수 있음을 보이고 기존 물리적, 논리적 획득 기법을 사용하여 획득 가능한 데이터와 비교하여 분석한 결과를 서술한다.

본 논문의 2장에서는 안드로이드 스마트폰과 관련한 모바일 포렌식 연구를 설명하고 3장에서는 제조사의 백업용 모바일 앱을 이용하여 데이터를 획득하는 방법에 대해 설명한다. 4장에서는 3장에서 설명한 데이터 획득 방법을 실제 최신 스마트폰을 대상하고 적용하고 그 결과에 대해 설명한다.

Table 1. Distribution of Android platform (Google Developers' Dashboard, Dec.11,2017)

version	codename	API	dist.
2.3.3 ~ 2.3.7	Gingerbread	10	0.6%
4.0.3 ~ 4.0.4	ICS	15	0.6%
4.1.x	Jelly Bean	16	2.3%
4.2.x		17	3.3%
4.3		18	1.0%
4.4	KitKat	19	14.5%
5.0	Lollipop	21	6.7%
5.1		22	21.0%
6.0	Marshmallow	23	32.0%
7.0	Nougat	24	15.8%
7.1		25	2.0%
8.0	Oreo	26	0.2%

II. 관련 연구

2.1 모바일 포렌식 증거 획득 기법

모바일 포렌식 증거물 획득은 증거물의 무결성(integrity)을 손상시키지 않고 플래시 메모리에 저장된 사용자 데이터를 획득하는 것이 주요한 목표이다. 그러나 안드로이드의 경우, Table 1.에서와 같이 스마트폰 제조사별로 하드웨어와 운영체제의 파편화 정도가 iOS에 비해 심하기 때문에 다양한 하드웨어 스펙과 운영체제 버전에 따라 기존에 연구된 포렌식 데이터 획득 기법의 적용 가능 여부가 상이하게 나타난다.

모바일 포렌식 획득 기법은 Fig.1.과 같이 크게 하드웨어 기반 획득 기법과 소프트웨어 기반 획득 기

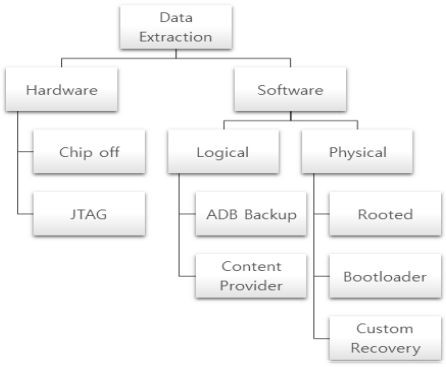

Fig. 1. Mobile Forensic Aquisition Methods

법으로 구분할 수 있다. 또한, 소프트웨어 기반 획득 기법은 세부적으로 물리적 획득 기법과 논리적 획득 기법으로 구분된다.

2.2 하드웨어 기반 데이터 획득 기법

하드웨어 기반의 데이터 획득 기법은 대표적으로 Chip-off 인터페이스를 이용한 데이터 획득 방법[4]과 JTAG(Joint Test Action Group) 인터페이스를 이용한 데이터 획득 방법으로 구분된다[3][5][14].

2.2.1 Chip-off 데이터 획득 기법

Chip-off 데이터 획득 기법[4]은 스마트폰의 PCB(Printed Circuit Board)에서 플래시 메모리를 물리적으로 분해한 후 메모리에 저장된 데이터를 비트 단위로 복제하여 데이터를 획득하는 방법이다. 그러나, 메모리칩을 분해 과정에서 메모리에 저장된 데이터와 스마트폰 자체가 손상될 우려가 존재하며 최근 출시된 안드로이드 스마트폰의 경우 디스크 암호화(full disk encryption)[6]가 기본 적용되는 추세이기 때문에 데이터를 로우 레벨로 복제하더라도 하드웨어 기반의 저장소에 보관되어 있는 복호화 키를 추출할 수 있는 방법이 현재까지 알려지지 않아 데이터 획득이 불가능한 한계가 존재한다.

특히, ARM TrustZone[7]의 TEE(Trusted Execution Environment)와 하드웨어 보안 모듈로 구성되는 안드로이드 기기의 디스크 암호화(full disk encryption)는 DEK(Device Encryption Key)로 불리는 무작위 생성된 키에 의해 관리된다.

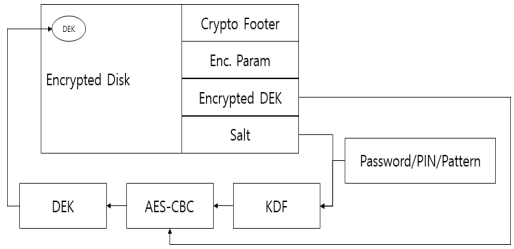

Fig. 2. Android FDE(Full Disk Encryption)

DEK는 PIN, 비밀번호, 패턴 등에서 비롯된 다른 키와 연동되어 암호화가 이루어지기 때문에 Chip Off 과정을 통해 메모리가 분해되면 이와 연동된 복호화 키를 확보할 수 없게 된다. 따라서 최근 디스크 암호화가 기본 적용되어 있는 스마트폰에는 적용이 제한적인 방법이다[15]. Fig.2.는 Android Full Disk Encryption의 동작 원리를 나타낸다.

2.2.2 JTAG 데이터 획득 기법

JTAG(Joint Test Action Group) 인터페이스를 이용한 데이터 획득 기법[3][5]은 스마트폰의 PCB에 노출된 JTAG 인터페이스에 직접 물리적으로 연결하여 활성화된 포트를 통해 플래시 메모리의 데이터를 획득하는 방법이다.

JTAG은 IEEE 1149.1[16] 표준에 따른 임베디드 시스템 개발 시에 사용하는 디버깅 장비를 의미한다. 포렌식 데이터 획득 시에는 일반적으로 Riff Box[17], Octopus Box[18]와 같은 플래셔 툴을 이용하여 스마트폰의 JTAG 인터페이스에 물리적으로 연결한 후 메모리의 정보를 추출하는 방법을 사용한다. JTAG을 기반한 데이터 획득 기법은 하드웨어 및 소프트웨어의 손상이 거의 없는 상태에서 데이터를 획득할 수 있다는 장점이 있는 반면, 데이터 획득에 비교적 시간이 많이 소요된다는 단점이 존재한다.

그러나 최근 출시된 스마트폰의 경우 JTAG 인터페이스를 숨기거나 아예 핀 자체를 제거하여 출시하는 경우 사용이 불가하고 Secure JTAG 등 추가적인 보안 기능이 적용된 경우 포트에 물리적으로 연결이 가능한 경우에도 메모리에 접근이 불가한 한계점이 존재한다[19].

2.3 소프트웨어 기반 데이터 획득 기법

소프트웨어 기반 데이터 획득 기법은 세부적으로 논리적 획득 기법과 물리적 획득 기법으로 구분할 수 있다.

2.3.1 논리적 데이터 획득 기법

논리적 획득 기법은 대표적으로 활성 데이터 획득 기법, ADB Backup 기능에 기반한 획득 기법[20][23], Android Content Provider[21]를 이용한 획득 기법, 백업 기능을 이용한 획득 기법 등이 있다[22].

활성 데이터 획득 기법은 스마트폰의 내부 저장소에 저장된 데이터를 단순히 복사하는 방식으로 획득하는 기법이다. 따라서 루팅된 스마트폰이 아닌 경우에는 획득할 수 있는 정보가 제한적인 방법이다. 비할당 영역 등 삭제한 데이터에 대한 복구도 지원하지 않는다.

ADB Backup 기반 획득 기법[20][23]은 안드로이드에서 내장된 백업 기능을 활용하는 방법이다. 안드로이드 백업 파일을 생성하기 위해서는 "adb backup -apk -shared -all -f TargetDevice.ab" 명령을 실행한다. 또한, 생성된 백업 파일은 "Android Debug Extractor" 툴을 이용하여 tar형식의 압축파일로 변환할 수 있고 압축을 해제하면 백업된 데이터의 획득이 가능하다. 백업 파일에 포함된 데이터를 확인해보면 스마트폰의 내부 저장소가 마운트된 "/storage/emulated/0/" 디렉토리와 앱 데이터의 일부가 저장되는 "/data/data" 디렉토리가 포함된 것을 확인할 수 있다. 그러나 백업 대상에서 시스템 앱 데이터는 제외되고, "/storage/emulated/0" 디렉토리의 "Android" 폴더도 제외되기 때문에 일부 데이터만 백업이 가능하다.

Android Content Provider 기반 데이터 획득 기법[21]은 안드로이드 앱 간에 데이터를 공유하는 기술을 이용하여 데이터를 획득한다. 안드로이드 환경 하에서 동작하는 앱은 저마다 샌드박스를 갖고 있어 서로의 데이터에 접근이 불가능하다. 그러나 앱 간의 데이터 교환이 필요한 경우, Content Provider를 사용하여 데이터를 공유한다. 안드로이드에 기본 설치된 앱의 경우 외부에 Native Content Provider를 공개하고 있기 때문에 이를 이용하여 연락처, 통화기록, 메시지, 브라우저 정보, 알람 정보, 미디어 스토어 등의 데이터를 획득할 수 있다. Content Provider에 접근하기 위해서는 URI(Uniform Resource Identifier)을 사용하는데 안드로이드에 기본 설치된 앱은 자신들의 데이터에 접근이 가능하도록 URI를 공개하고 있다. 따라서 디지털 포렌식 관점에서 중요한 메시지, 연락처, 통화기록, 일정 등의 데이터는 Content Provider를 활용하면 획득이 가능하다. 그러나 Content Provider 방식 또한 앱에서 제공하는 인터페이스를 이용하여 데이터를 획득하는 방식으로 직접 데이터가 저장된 파일에 접근할 수 없어 삭제된 데이터에 대한 획득은 불가능한 방법이다. Table 2.는 대표적인 시스템 앱의 외부에 공개된 URI를 나타낸다.

PC용 백업 프로그램을 이용한 데이터 획득 기법은 대표적으로 Samsung Kies 프로그램을 이용한 방법[22]이 연구되었다. Kies로 생성된 백업 파일의 내부 구조를 분석하여 백업된 파일에서 데이터를 직접 복원하는 기법으로 백업 파일 내부의 데이터를 분석하여 파일 타입별 Signature를 기반으로 데이터를 카빙한다. 해당 연구가 진행된 시점에는 백업 파일에 대한 암호화가 적용되기 전으로 Signature 기반의 카빙이 가능하였으나, 현재는 백업 파일에 대해 암호화가 기본 적용되어있고 최신 안드로이드 스마트폰의 경우에는 더 이상 Kies 기반의 백업을 지원하고 있지 않아 최신 스마트폰에는 현재 적용이 불가하다.

이와 같이, 논리적 획득 기법은 루트 권한을 전제로 하지 않으면 활성 상태의 삭제되지 않은 데이터에 대한 접근만 가능하기 때문에 공통적으로 삭제된 데이터에 대한 복구는 지원하지 않는 한계가 존재한다.

Table. 2 URI info. of android pre-load apps

items	content provider URI
Call Log	CallLog.Calls.CONTENT_URI
Contact	ContactsContract.Data.CONENT_URI
Calendar	content://com.android.calendar/events
SMS	content://sms/inbox
MMS	content://mms/inbox

2.3.2 물리적 데이터 획득 기법

물리적 획득 기법으로는 루팅에 기반한 획득 기법, bootloader의 취약점을 이용한 기법[11], custom recovery image[12]를 이용한 기법이

연구되었다.

루팅에 기반한 획득 기법은 스마트폰에 USB 케이블을 연결하여 플래시 메모리의 처음 주소지부터 마지막 주소지까지 전체를 읽어 데이터를 획득하는 방법이다. 스마트폰이 루팅된 경우 USB 디버깅 모드를 활성화하여 "ADB Shell" 명령으로 접속하여 root권한을 확보한 상태에서 "dd" 명령 등을 이용하여 전체 메모리를 덤프하는 방식으로 복제 이미지를 생성한다. 그러나 루팅이 반드시 선행되어야 하기 때문에 최근 출시된 스마트폰은 루팅이 가능하게 되기 전까지는 적용이 불가하고 루팅이 가능한 경우에도 루팅 과정에서 메모리에 저장된 증거물의 무결성(integrity)이 손상될 위험성이 존재한다.

Cellebrite UFED와 같은 상용 모바일 포렌식 제품에서 제공하는 bootloader 취약점에 기반한 데이터 획득 기법[11]은 전체 메모리에 대한 물리적 이미징이 가능한 기법이다. 그러나 신규로 스마트폰이 출시될 경우 최신 안드로이드와 최신 펌웨어 버전을 적용하여 기존에 존재하던 취약점들은 제거되는 경우가 많아 동일한 방법으로는 데이터 획득이 불가하게 된다. 향후 동일한 방법을 사용하여 데이터를 획득할 수 있게 되기까지 신규 취약점에 대한 연구가 선행되어야하여 신규 스마트폰에 즉시 적용이 불가한 사례가 존재한다.

Custom recovery image를 이용한 데이터 획득 기법[12]은 사용자 데이터에 대한 무결성(integrity)이 보장되는 획득 방법이지만, 최근 출시된 스마트폰의 경우 "Secure Boot", "Knox Warrnaty" 등의 강화된 보안기능이 적용되어 현재 적용이 불가한 상태이다.

이처럼 기존에 연구된 데이터 획득 기법은 최신 스마트폰을 대상으로 즉시 적용이 어려운 제약사항들이 존재한다. 이후의 장에서는 제조사의 백업용 모바일 앱을 기반으로 스마트폰에 저장된 사용자 데이터를 획득하는 방법에 대해 상세히 설명한다.

III. 백업용 모바일 앱 기반 획득 기법

3.1 백업 앱 기반 데이터 획득의 필요성

포렌식 데이터 획득 기법에서 취약점을 이용하여 루트 권한을 획득하는 목적은 사용자 데이터가 저장된 폰의 플래시 메모리 전체를 이미징하여 분석하기 위함이다. 이는 현재 폰에 활성 상태로 저장된 데이터의 획득 뿐만 아니라 사용자 또는 악성코드에 의해 임의로 삭제된 비할당 영역에 남아있을 가능성이 존재하는 데이터를 복구하여 사고조사 및 사용자 행위와 관련한 포렌식 관점에서 유의미한 결정적 데이터를 획득하는 것을 의미한다. 그러나 Table 3.에서와 같이 2007년부터 현재까지 국내에 출시된 21개 제조사의 1,850대의 폰을 대상으로 기존 데이터 획득 기법의 적용 가능 여부를 조사한 결과[31], 하드웨어 기반의 Chip Off, JTAG 방식의 경우 2010년 이후 출시된 폰에 본격적으로 적용 가능하였으나, 2013년 이후 지속적으로 감소하여 2017년 현재 출시된 폰 중 적용 가능한 폰은 없는 상태이다. 또한, 소프트웨어 기반의 Custom Image, Bootloader 기반의 데이터 획득 기법 역시 2011년 적용 가능한 폰이 최초 등장한 이후 일부 모델들에 한하여 적용 가능하였으나 제한적인 폰에 한하여 적용이 가능하였고 2016년 이후 감소하여 2017년 현재 출시된 폰 중 적용 가능한 모델은 없는 상태이다. 이와 같이, 기존의 하드웨어, 소프트웨어 기반의 데이터 획득 기법은 2017년 출시된 폰을 대상으로는 적용 가능하지 않음을 확인하였다.

Fig. 3은 연간 시장에 출시된 전체 폰 모델 중 기존 데이터 획득 기법을 적용할 수 있는 폰의 비율을 나타낸다. 최초 데이터 획득 기법이 소개된 이후 시간이 경과할수록 적용 가능한 폰이 감소하는 것을 확인할 수 있다. 적용 가능한 폰이 감소하는 이유는 기존의 데이터 획득 기법이 폰의 취약점에 기반한 기법이기 때문이다. 취약점은 공개된 이후 패치와 업데이트를 통해 해당 취약점은 제거되는 과정을 거치게

Table 3. Applicable phone model by existing data acquisition techniques by product release date

year	# of phone	Chip Off	JTAG	Custom Image	Boot loader
2007	28	1	1	0	0
2008	46	2	2	0	0
2009	40	2	2	0	0
2010	89	37	37	0	0
2011	154	52	47	3	12
2012	168	64	27	2	47
2013	240	73	6	8	57
2014	341	84	0	35	58
2015	420	63	0	56	36
2016	249	12	0	28	13
2017	75	0	0	0	0
total	1,850	390	122	132	223

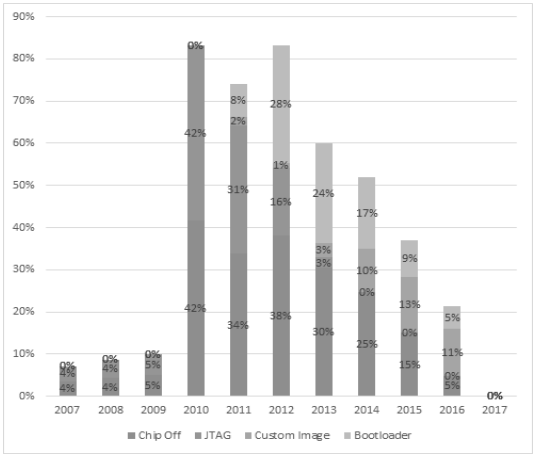

Fig. 3. Percentage of phones that can be applied to existing data acquisition techniques compared to released phones

되고 이후 신규로 출시되는 폰은 취약점 자체가 제거된 상태로 출시된다. 또한 지속적으로 강화된 보안 기법들이 적용되기 때문에 동일한 패턴의 취약점을 지속적으로 확보하는 것도 더욱 어려운 상황에 놓이게 된다.

포렌식 데이터 획득 관점에서 지속적이고 즉시 사용이 가능한 획득 기법에 대한 연구는 반드시 필요하다. 또한, 패치되면 제거될 취약점에 기반한 방식이 아닌 폰 제조사가 사용자에게 필수로 제공해야하는 기본 기능에 기반한 획득 방법이면 획득의 지속성과 즉시성을 확보하는 측면에서 효과적인 방법일 것이다. 여기에 더해 현재 출시된 폰의 물리적 데이터 획득이 불가능하다는 것을 전제로 단순 파일을 복사하는 방식의 논리적 획득 방식으로는 추출하기 어려운 삭제된 사용자 데이터의 획득까지 가능하다면 기존 논리적 데이터 획득 방식의 효과적인 대안으로 충분히 활용될 수 있을 것이다.

3.2 백업 앱 기반 데이터 획득 절차

본 장에서는 제조사에서 제공하는 사용자 데이터 백업용 모바일 앱을 이용하여 스마트폰의 플래시 메모리에 저장된 데이터를 획득하기 위한 기법에 대해 설명한다. 스마트폰 제조사들은 자사 스마트폰을 이용하는 사용자의 데이터를 안전하게 보관하고 필요시 원상태에 가깝도록 복원하기 위한 목적으로 백업용 모바일 앱을 기본 제공하고 있다. 모바일 포렌식 관점에서 루트 권한을 획득하지 않은 상태에서 데이터의 무결성(integrity)을 손상시키지 않고 데이터를 획득하기 위한 기법으로 사용이 가능하다.

3.2.1 획득 가능 데이터 분석

스마트폰 제조사들은 다양한 백업용 모바일 앱을 사용자에게 기본으로 제공하고 있다. Table 4.은 스마트폰 제조사들이 제공하는 백업 앱을 나타낸다.

이들 백업용 모바일 앱은 스마트폰의 플래시 메모리에 저장된 사용자 데이터를 백업하고 복원하는 기능을 기본으로 제공한다. Table 5.는 백업용 모바일 앱을 이용하여 백업이 가능한 주요 데이터 항목을 나타낸다.

제조사의 백업용 모바일 앱은 일반사용자 권한으로는 접근이 불가능한 데이터를 포함하여 백업하는 것을 확인할 수 있다. 기존 데이터 획득 기법에서는 하드웨어적으로 플래시 메모리에 접근하여 전체 데이터를 덤프하거나 취약점 등을 이용하여 root 권한을 획득한 후 물리적 데이터 획득 방법을 사용하여야만 해당 데이터에 대한 추출이 가능하다.

Table 4. Backup mobile apps

manufacturer	name	version
Samsung	Smart Switch moblie	3.4.08.4
Apple	Move to iOS	2,10,0
Huawei	Phone Clone	8.0.0.305
LG	Mobile Switch	3.3.3
Sony	Transfer Mobile	2.2.A.4.44

Table 5. Backup data and access rights

Items	data type	access right
Contact	DB(phone no.+name)	system
Message	DB(SMS)+File(MMS)	system
Calendar	DB(Events)	system
Call Log	DB(call send/receive log)	system
Setting	DB(homescreen, WiFi)	system
Photo	File(jpg, gif)	user
Movie	File(mp4, avi)	user
Voice	File(m4a, 3ga)	user
Document	File(doc, pdf)	user

3.2.2 백업용 앱을 이용한 데이터 획득 과정

백업용 모바일 앱을 이용하여 획득 대상 데이터를

선정한 후 스마트폰 간에 데이터가 전송되는 과정에서 자동으로 생성, 삭제되는 파일을 추출하여 데이터를 획득한다. Fig. 4.은 백업 앱의 데이터 전송과정을 나타내며 분석 결과 사용자의 데이터가 백업되고 복원되는 과정은 다음과 같다.

① 백업 앱 실행 및 백업 대상 데이터 선택

데이터 획득 대상의 스마트폰의 백업용 모바일 앱을 실행한 후 전송 대상 데이터를 선택한다.

② 백업파일 생성

백업용 모바일 앱은 시스템 앱의 앱 데이터에 저장된 정보를 추출하여 압축한 후 암호화하여 백업용 모바일 앱의 앱 데이터 영역에 저장한다(system권한, /data/user/0/com.sec.android.easyMover/files/). 암호화 시에는 "AES/CBC/PCKS5Padding" 암호 알고리즘을 사용하여 백업 파일을 암호화하고 세션 키는 xml파일 형태로 공유한다.

③ 수신용 폰으로 생성파일 전송

암호화된 압축파일은 무선(Wifi) 또는 USB 연결을 통해 수신용 스마트폰의 Userdata 파티션의 내부 메모리 영역에 전송된다(/sdcard/Smart Switch/tmp/)

④ 백업 파일 전송 및 복호화

전송된 암호화된 압축파일은 공유된 세션 키를 이용하여 복호화 된다.

⑤ 압축 해제 후 데이터 복원

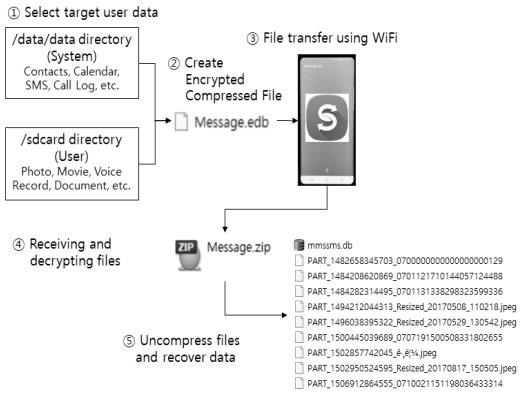

Fig. 4. Backup and recover process

복호화된 압축파일은 압축이 해제되고 복원 대상이 되는 시스템 앱의 앱 데이터 영역의 해당 디렉토리에 각각 복사되어 데이터가 복원된다. 복원과정에서 생성된 암호화가 해제된 임시 파일은 작업 종료 후 모두 삭제된다.

데이터 획득 대상 스마트폰에서 수신용 스마트폰으로 데이터가 전송되고 복원되는 과정에서 임시 생성된 파일을 중간에 가로채는 방식으로 사용자 데이터의 획득이 가능하다. 또한, 임시 생성된 파일은 루트 권한이 불필요한 일반 사용자 권한으로 접근이 가능한 스마트폰의 내부 메모리 영역에 생성되기 때문에 파일 생성 정보를 모니터링 하여 생성된 파일에 대해 직접 추출하는 방식을 사용한다. 또한, 백업 파일은 암호화가 적용된 상태로 전송 시 공유되는 세션 키를 가로채는 방식으로도 복호화가 가능하나, 복원 과정에서 스마트폰에 복호화된 파일을 임시 생성하기 때문에 별도의 복호화 과정은 필요하지 않다.

수신용 폰에 복호화된 임시 파일이 생성되는 경로는 Table 6.와 같다. 연락처, 메시지, 일정 등 DB 형태로 저장되는 파일은 암호화가 적용된 상태로 전송되며, 이미지, 동영상, 음성녹음, 문서 등 멀티미디어 파일은 수신용 스마트폰의 기존 획득 대상 스마트폰과 동일한 경로에 파일이 복사되는 것을 확인할 수 있다.

Table 7.은 수신용 스마트폰에서 복원 과정 중 임시 생성되었다가 삭제되는 파일을 나타낸다. 해당 파일이 복원 과정에서 추출할 직접적인 대상이 되는 파일이다. 암호화가 적용된 백업 파일에 대한 복호화는 공유된 세션 키를 기반으로 복호화가 가능하나,

Table 6. Backup data locations

items	backup locations
Contact	/data/user/0/com.sec.android.easyMover/files/SmartSwitch/tmp/vcard
Message	/storage/emulated/0/SmartSwitch/tmp/MESSAGE
Calendar	/storage/emulated/0/SmartSwitch/tmp/Calendar
Call Log	/storage/emulated/0/SmartSwitch/tmp/CALLLOG
Photo	/storage/emulated/0/DCIM/
Movie	/storage/emulated/0/DCIM/
Voice	/storage/emulated/0/Voice Recorder/
Document	/storage/emulated/0/Document/

Table 7. Temporary files before recovery

items	temp file name & path
Message	/storage/emulated/0/SmartSwitch/tmp/MESSAGE/Message.zip
Calendar	/storage/emulated/0/SmartSwitch/tmp/Calendar/CALENDER.zip
Call Log	/storage/emulated/0/SmartSwitch/tmp/CALLLOG/CALLLOG.zip

백업용 앱 기반 데이터 획득 과정에서는 불필요한 작업으로 본 논문의 구현 범위에 포함하지 않으며, 향후 연구에서 추가할 계획이다.

3.2.3 기존 데이터 획득 기법과의 비교

기존 물리적, 논리적 획득 기법과 백업용 모바일 앱을 이용하여 획득 및 분석 가능한 데이터의 범위와 결과를 비교한다. Table 8.와 같이 데이터 획득 범위 측면에서는 물리적 획득 기법이 플래시 메모리의 비할당 영역까지 포함하여 획득이 가능하기 때문에 가장 효과적인 기법이라고 할 수 있다. 그러나 플래시 메모리는 데이터 삭제 시점에 컨트롤러 단에서 TRIM[27], Discard[28] 등의 명령이 내부적으로 수행되기 때문에 플래시 메모리를 기본 저장매체로 사용하는 스마트폰은 삭제된 파일이 플래시 메모리의 비할당 영역에서 복구될 수 있는 가능성이 매우 낮다.

또한, 삭제 데이터 복구 관점에서 기존의 논리적 획득 기법은 삭제되지 않은 활성 데이터에 대한 복구만 가능한 것과 비교하여 백업 앱 방식을 사용하면 데이터가 저장되어 있는 원본 DB파일을 직접 획득할 수 있어 DB 내부의 비할당 영역을 분석하면 삭제한 데이터에 대한 복구가 가능하였다. 이는 데이터의 획득 측면에서 획득 가능 데이터의 범위는 물리적 데이터 획득 기법의 범위와 동일한 수준이며, 논리적 데이터 획득 기법으로 획득 가능한 데이터의 범위보다 더 효과적이라고 하겠다. 또한 신규 출시되는 스마트폰의 경우 물리적 데이터 획득 기법의 적용이 가능하기까지 추가적으로 수행되는 취약점의 분석과 확보에 소요되는 시간과 비교했을 때 동일한 제조사에서 출시되는 스마트폰에 기본적으로 탑재되는 백업 앱 방식을 사용하면 즉시 적용이 가능하기 때문에 보다 효과적인 방법임을 확인할 수 있다.

IV. 실 험

디지털 포렌식 과정을 통해 스마트폰의 데이터를 획득하는 것은 수집 대상 데이터의 획득이 가능한지 여부와 확보한 데이터가 증거물로서 무결성(integrity)이 훼손되지 않는지 여부를 모두 고려하여야 한다. 이러한 2가지 요소를 중심으로 제조사의 백업용 모바일 앱 기반의 획득 기법을 실제 적용하여 최근 출시되어 기존의 물리적 획득 기법의 적용이 불가한 스마트폰을 대상으로 데이터 획득이 가능함을 확인한다. 또한, 획득 대상 데이터는 일반 사용자 권한으로는 접근이 불가한 시스템 앱 데이터 중에서 메시지(SMS) 데이터를 획득 대상으로 선정하였다. 또한 획득한 증거 데이터에 대한 분석을 수행하여 활성 데이터뿐만 아니라 DB의 비할당 영역에 잔여하는 삭제된 데이터에 대한 복구를 시도하고 복구 가능 여부를 확인한다. 삭제한 메시지의 복구에 사용

Table 8. Comparison with existing data acquisition techniques(if data acquisition is possible)

type	items	access rights	data acquisition			delete data recovery (● : live+deleted, ○ : live only)		
			physical	logical	ours	physical	logical	ours
DB	Contacts	system	○	○	○	●	○	●
	Message	system	○	○	○	●	○	●
	Calendar	system	○	○	○	●	○	●
	Call Log	system	○	○	○	●	○	●
File	Photo	user	○	○	○	○	○	○
	Movie	user	○	○	○	○	○	○
	Voice	user	○	○	○	○	○	○
	Document	user	○	○	○	○	○	○
Unlocated		system	○	×	×	×	×	×

된 기법은 SQLite 기반의 비할당 영역에 잔존하는 삭제된 레코드 복구 기법을 사용하였다[24][25]

4.1 실험 환경

Table 9.은 실험에 사용된 대상 장비와 장비별 버전 정보를 나타낸다. 갤럭시 S8 및 S8+ 장비는 MD-NEXT[29], Final Mobile Forensic[30] 등 상용 모바일 포렌식 툴에서 전체 플래시 메모리에 대한 덤프를 지원하지 않는 모델이다. 또한, 사용자 데이터 영역(Userdata Partition)은 디스크 암호화(Full Disk Encryption)가 적용되어 있어 Chip-off 및 JTAG 방식 등 하드웨어 기반의 데이터 획득 기법을 이용하여 획득이 가능하더라도 암호화된 데이터의 복구가 불가하다. 소프트웨어 기반 데이터 획득 기법인 bootloader 기반 데이터 획득 기법과 Custom Recovery Image 기반의 데이터 획득 기법의 적용도 현재 공개된 적용 방법이 존재하지 않는다. 대상 장비는 2016년12월16일 이후로 실사용한 메시지가 저장되어 있으며 총 233개(SMS 103개, MMS 130개)의 메시지가 저장된 상태이다.

Table 9. test device and versions

device Info.		version
Target Device	Galaxy S8+ (SM-G955N)	G955NKSU1AQI1 (Android 7.0)
	Smart Switch	3.4.08.4
	Message App	4.2.03.0
	LG V30 (LGM-V300S)	N2G47H (Android 7.1.2)
	Mobile Switch	3.3.3
	Contact App	6.10.9.2
Receive Device	Galaxy S8 (SM-G950N)	G950NKSU1BQIE (Android 7.0)
	Smart Switch	3.4.08.4
	Message App	4.2.02.14
integrity check (Rooted)	Galaxy S7 Edge (SM-G935S)	G935SKSU1DQG1 (Android 7.0)
	Smart Switch	3.4.08.4
	Message App	4.1.16

4.2 메시지(SMS/MMS) 데이터 획득 및 복구

4.2.1 실험 #1 - 메시지(SMS/MMS) 데이터 획득

획득용 스마트폰과 수신용 스마트폰은 모두 "비행기 모드"를 설정하여 네트워크 접속을 차단하여 네트워크를 통한 데이터 유입을 차단한다. Fig. 5.와 같이 획득 대상인 갤럭시 S8+에서 "설정" 앱을 실행한 후 "클라우드 및 계정", "Smart Switch(스마트 스위치)"를 차례로 실행한다. 전송방법으로는 "무선" 또는 "USB"를 선택하고 "보내기" 기능을 선택하여 수신용 스마트폰인 갤럭시 S8과 연결 전 상태로 대기한다. 단, "무선" 방식의 경우 WiFi Direct 기능을 이용하여 두 기기간의 연결만 가능하고 타 기기와의 WiFi연결 또는 데이터 네트워크로의 접속 등은 불가한 상태로 동작하기 때문에 추가 데이터 유입은 없으나 데이터 유입으로 인한 무결성 손상을 원천적으로 차단하기 위해 본 실험에서는 "USB" 방식의 연결을 사용하도록 하였다.

데이터 수신용으로 설정한 갤럭시 S8은 획득 대상 스마트폰과 동일한 과정을 거쳐 수신 대기상태로 설정한다. "개발자 모드"를 활성화한 후 "USB 디버깅" 옵션을 활성화하고 PC와 USB케이블을 이용하여 연결한 후 PC의 커맨드 창에서 "ADB Devices" 명령으로 연결 가능여부를 확인한다. 이후 획득 대상인 폰과 마찬가지 과정을 수행하고 Smart Switch Mobile 앱을 구동한 후 "설정" 앱을 실행하고 "클라우드 및 계정", "Smart Switch(스마트 스위치)"를 차례로 실행한다. 전송방법으로는 "무선" 또는 "USB"를 선택하고 "받기" 기능을 선택한 후 송신 기기를 "안드로이드"로 선택한 후 연결 전 상태로 대기한다.

송신용과 수신용 스마트폰이 연결 전 상태로 준비된 상태에서 두 기기를 연결한 후 획득 대상인 폰에서 전송항목 중 메시지 항목을 선택하여 수신용 폰으

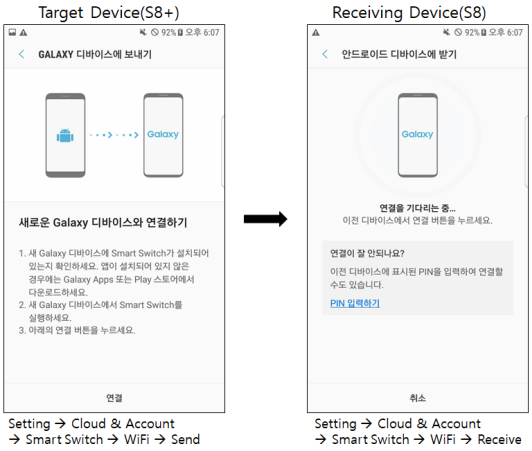

Fig. 5. setting screen between transmitting and receiving phones

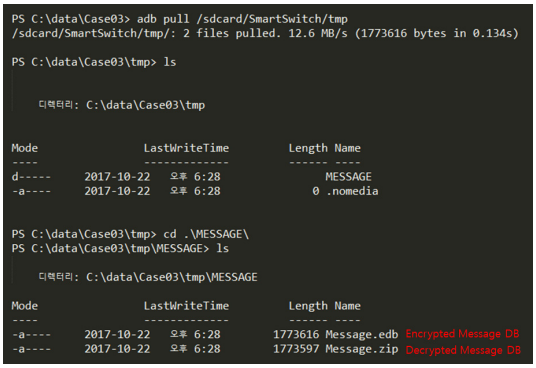

Fig. 6. Acquisition of decrypted DB using adb pull command

로 전송한다. 이와 동시에 Fig. 6.와 같이 "ADB pull" 명령을 실행하여 미리 파악한 경로에 임시 생성된 데이터를 획득한다. 그 결과 "Message.edb"파일과 "Message.zip"파일 총 2개의 파일을 획득할 수 있다.

4.2.2 실험 #2-메시지(SMS/MMS) 데이터 복구

획득한 파일을 압축을 해제하면 Fig. 7과 같이 백업된 메시지의 DB(mmssms.db)와 MMS에 첨부되어 송수신된 사진을 확인할 수 있다. SQLite의 비할당 영역에 잔존하는 삭제된 레코드 복구 기법을 이용하여 확보한 DB의 "Leaf 페이지"에 남아있는 삭제한 메시지의 존재 여부를 분석한다.

Sqlite DB의 삭제된 데이터에 접근하기 위해서는 먼저 Sqlite의 헤더 페이지를 분석하여 Leaf 페이지를 탐색한다. 첫 바이트가 "0x0D"의 값을 사용하고 있는 Leaf 페이지를 식별하고 Leaf 페이지는

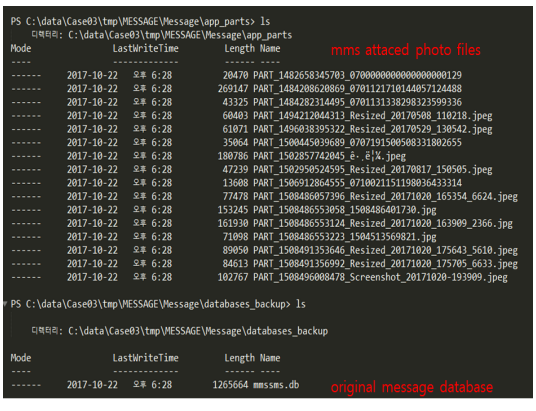

Fig. 7. The extracting result of zip file

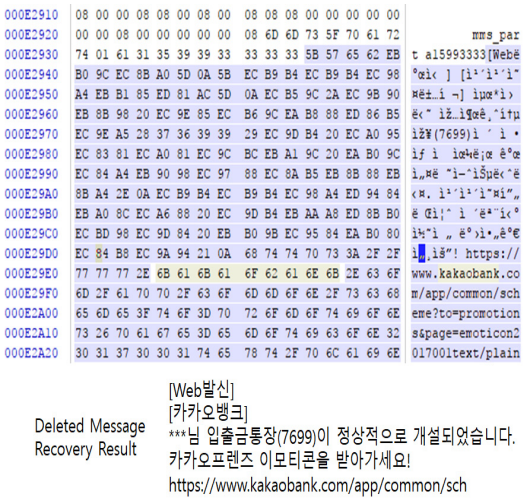

Fig. 8. Result of restoring deleted DB records

바이트 오프셋 2~3에 2바이트의 크기로 첫 번째 비할당 블록의 오프셋 값을 저장하고 있다. 각각의 블록은 바로 다음의 비할당 블록의 오프셋을 상위 2바이트에 저장하는 방식으로 비할당 블록의 체인을 구성하고 있음을 확인한다.

이렇듯 SQLite의 비할당 블록 체인을 이용하여 페이지 내 모든 비할당 블록을 탐색할 수 있다. 비할당 블록 탐색결과, Fig. 8.과 같이 Sqlite DB의 "Leaf 페이지" 내에서 삭제된 SMS의 내용을 확인할 수 있다.

전체 메시지 DB에 대한 복구 결과는 Table 10.와 같다. 삭제되지 않은 활성 메시지외에 비할당 영역에 남아있는 삭제된 메시지 총 88개를 복구할 수 있었다.

Table 10. Message DB recovery result(total)

item	type	live	deleted	total
Message	SMS	103	14	177
	MMS	130	74	204
total		233	88	321

4.2.3 실험 #3 - 메시지(SMS/MMS) 무결성 검증

디지털 포렌식 데이터가 증거 능력을 인증받기 위해서는 무결성(integrity)이 입증되어야 한다. 증거 데이터의 무결성(integrity)을 확인하기 위해 주로

사용되는 방법은 해시(Hash)값 검증이다. 무결성(integrity)은 수집 단계부터 법정에 증거물로 제출되기까지 데이터가 훼손되지 않고 보호되었음을 입증하기도 하지만, 원본 데이터가 실제 법정에 제출되는 데이터와 동일한지 동일성을 입증하기 위한 용도로도 사용된다.

획득한 데이터의 무결성(integrity)을 입증하기 위한 해시(Hash)값을 추출하는 용도로 "MD5sum 1.2" 툴을 사용하였다[26]. 원본 데이터와 백업 앱을 이용하여 획득한 데이터의 해시값 비교하여 무결성(integrity)을 검증하기 위해 현재 루팅이 가능한 갤럭시 S7 Edge 기종을 사용하였다. 루트(root) 권한을 확보한 후 "ADB pull" 명령으로 직접 MMS/SMS 데이터가 저장된 시스템 앱 데이터 영역에 접근하여 원본 데이터를 확보하였다. 이후 실험 #1, 실험 #2의 과정과 동일한 방법으로 백업 모바일 앱을 실행하여 메시지를 수신용 스마트폰으로 전송한 후 백업된 메시지 데이터를 획득한다. 이후 원본 데이터와 획득한 증거 데이터의 해시 값을 Fig.9와 같이 계산하여 각각 비교한다. 원본 데이터와 획득 데이터의 해시 값 비교 결과는 Table 11.과 같다. 원본 데이터와 획득 데이터의 해시 값이 동일한 것을 확인할 수 있고 이를 통해 백업 앱을 기반으로 한 데이터 획득 방법으로 획득한 데이터는 데이터가 획득 과정에서 무결성(integrity)이 훼손되지 않음을 확인하였다.

```
PS C:\Data\Case03\Hash> .\md5sums.exe .

MD5sums 1.2 freeware for Win9x/ME/NT/2000/XP+
Copyright (C) 2001-2005 Jem Berkes - http://www.pc-tools.net/
Type C:\Data\Case03\Hash\md5sums.exe -h for help

[Path] / filename                                MD5 sum
-------------------------------------------------------------------------------
[C:\Data\Case03\Hash\]
md5sums.exe                                      da1e100dc9e7bebb810985e37875de38
mmssms.db                                        8c76fb3f555c3f747ba7b874863448c1
PART_1424504264953_07011217101440571244888      98c55d0725bd874b4dcda84e608282e8
```

Fig. 9. Hash Value Calculation

Table 11. Comparison table of hash values of original data and acquired data(SHA-1)

data	hash	original data	acquired data
mmssms.db	MD5	8c76fb3f555c3f747ba7b874863448c1	8c76fb3f555c3f747ba7b874863448c1
	SHA1	4A5A8519E8C7B3EF74F76A939E97BE0B7EEEFC23	4A5A8519E8C7B3EF74F76A939E97BE0B7EEEFC23

4.3 주소록(Contact) 데이터 획득 및 복구

4.3.1 실험 #4 - 주소록(Contact) 데이터 획득

LG V30폰을 대상으로 모바일 스위치 백업용 모바일 앱을 이용하여 주소록에 저장된 연락처 정보를 획득하는 방법에 대해 서술한다. 획득 대상 스마트폰을 "비행기 모드"로 설정하여 네트워크 접속을 차단하여 네트워크를 통한 데이터 유입을 차단한다. Fig.10과 같이 획득 대상인 LG V30 폰의 "모바일 스위치" 앱을 실행한다. 저장위치는 "SD카드"를 선택하고 실행 기능으로 "백업"을 차례로 선택한다. 백업 대상 항목을 지정한 후 백업을 실행하면 "LGBackup_171209.lbf" 하나의 단일 파일로 백업 파일이 생성된다.

Fig. 10. Procedure of Mobile Switch(V30)

4.3.2 실험 #5 - 주소록(Contact) 데이터 복구

획득한 백업 파일을 Hex에디터로 분석한 결과, 단일 lbf 파일을 파일 Signature를 기반으로 카빙하는 기법으로 백업 원본 파일의 추출이 가능하다. "contacts2.db" 텍스트 스트링 검색으로 해당 데이

터가 ZIP파일의 시그니처 형태로 저장된 것을 확인할 수 있다. Fig.11과 같이 zip 파일 시그니처 Header(0x50 0x4B 0x03 0x04)의 시작 오프셋으로부터 Footer(0x50 0x4B 0x05 0x06)의 오프셋까지의 데이터를 카빙 기법으로 추출하면 하나의 단일 압축파일을 획득할 수 있다.

획득한 압축파일을 풀면 주소록이 저장된 DB파일의 획득이 가능하다. 획득한 "\data\user\0\com.android.providers.contacts\databases\contacts2.db" 파일을 Sqlite DB 분석 기법을 이용하여 분석한 결과, Fig.12와 같이 삭제한 연락처 데이터를 포함한 전체 연락처 정보의 획득이 가능하다.

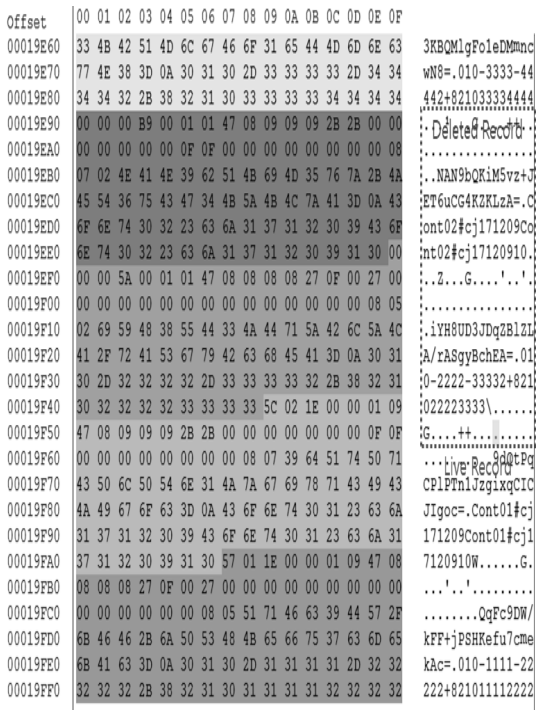

Fig. 11. active and deleted contact info of acquired file(contacts2.db

4.3.3 실험 #6 - 주소록(Contact) 데이터 무결성 검증

또한, Table 12과 같이 메시지 데이터 획득시와 동일한 방법으로 해시값을 검증하여 원본 데이터와 동일함을 확인하였다.

Fig. 12. Signature based file carving(zip)

Table 12. Comparison table of hash values of original data and acquired data

filename	Hash	original data	acquired data
contacts2.db	MD5	0E6705596D5F0969C7FD5F7F1149E7D9	0E6705596D5F0969C7FD5F7F1149E7D9
	SHA1	3FB67312DC2B858AD06839330F2B576EC93EB4EA	3FB67312DC2B858AD06839330F2B576EC93EB4EA

4.4 사용자 흔적 데이터

실험을 통해 백업 앱 기반 데이터 획득 기법으로 최근 출시된 스마트폰을 대상으로 무결성을 손상시키지 않으면서 사용자의 데이터를 획득할 수 있음을 확인하였다. 포렌식 데이터의 획득 관점에서는 궁극적으로 물리적 데이터 획득을 통한 전체 메모리의 이미징을 목표로 한다. 그러나 현재 출시된 스마트폰의 경우 앞서 설명한 추가 보안기능들로 인해 물리적 획득이 쉽지 않은 상황에서 삭제된 데이터 복원을 지원하는 백업 앱 기반 데이터 획득 기법은 활성 데이터만 수집하는 논리적 데이터 획득 기법의 다른 대안이 될 수 있다. 따라서 향후 사고조사 및 범죄수사시 증거 데이터의 획득 관점에서 활용될 수 있음을 알 수 있다. 백업 앱 기반 데이터 획득 기법을 이용하여 얻을 수 있는 정보가 무엇인지에 대한 상세 설명은 다음과 같다.

4.4.1 DB 형태 저장 데이터

DB 형태로 스마트폰의 메모리에 저장되는 데이터는 연락처, 메시지, 통화기록, 캘린더(일정) 등 이다. 이러한 데이터는 백업 앱 기반 데이터 획득 기법을 통해 수집이 가능하였으며 특히 메시지의 경우 원본 데이터가 저장된 DB 파일을 획득할 수 있어 삭제한 데이터에 대한 복원도 가능함을 확인하였다.

연락처 데이터는 사용자가 연락처 정보에 기록한 연락처명, 전화번호, 이메일, 주소 등 정보이며 이는 "/data/com.android.providers.contacts/databases/contacts2.db"에 기록되어 있어 획득이 가능하였다.

메시지 데이터는 삭제한 메시지 데이터를 포함한 "mmssms.db"에 기록되어 있으며 DB 테이블 내 메시지 본문, 제목, 생성일자, MMS첨부파일, 송신자 전화번호, 수신자 전화번호 데이터를 확인할 수 있다. 추가적으로 "SmartSwitch" 사용시 전송 및 복구 과정에서 별도의 로그 파일("SmartSwitch Log_전송일자_모델명.txt")을 생성하는데 해당 로그 파일 내부에도 JSON 포맷의 메시지 데이터가 추가적으로 기록되어 확보가 가능하다.

통화기록은 음성통화 및 메시지 수/발신 이력이 "/data/com.android.providers.contacts/databases/calllog.db"에 저장되어 있으며 수신 전화번호, 통화시간, 통화기간 데이터를 확인할 수 있다.

일정은 일정제목, 시작일, 종료일, 장소 데이터가 "/data/com.android.providers.calendar/databases/calendar.db" 경로에 저장되어 있어 확보가 가능하였다.

4.4.2 파일 형태 저장 데이터

파일 형태 데이터는 이미지, 동영상, 음성녹음 등 멀티미디어 파일과 pdf, doc, xls 등 문서 파일이 대표적이다. 물리적, 논리적 데이터 획득과 동일하게 비할당 영역에 존재하는 삭제된 파일 데이터에 대한 복구는 지원하지 않는 한계가 존재한다. 따라서 현재 획득 대상이 되는 스마트폰에 삭제되지 않은 상태로 저장된 활성 데이터에 대한 획득만 제한적으로 가능하다. 또한 멀티미디어 파일의 흔적 정보를 보관하고 있는 "미디어 로그" 관련 저장소 데이터는 백업 대상 정보가 아니기 때문에 접근이 불가하여 획득이 불가한 한계점이 존재한다.

V. 활용 방안

백업 앱 기반 데이터 획득 기법은 원본 데이터의 무결성을 훼손하지 않은 상태에서 스마트폰에 DB 형태로 저장되는 연락처, 메시지, 일정, 통화기록 데이터와 파일 형태로 저장되는 이미지, 동영상, 음성녹음, 문서 데이터의 획득이 가능한 기법이다. 특히, 기존 데이터 획득 기법의 적용을 어렵게 하는 추가적인 보안기능을 탑재한 최신의 스마트폰을 대상으로도 획득 방법에 대한 추가 연구가 필요 없이 즉시 적용이 가능하기 때문에 신속한 증거 데이터의 수집이 중요한 디지털 포렌식 측면에서도 효과적인 기법이라 할 수 있다. 또한, 기존 논리적 데이터 획득 기법에 비해 삭제한 데이터의 복구도 가능한 기법이기 때문에 사용자의 삭제된 데이터의 복원을 통해 범죄수사 및 사고조사 분야에서 결정적 증거를 확보하는 방법으로도 즉시 활용이 가능하다.

5.1 증거 데이터 수집의 즉시성 확보

변조 가능성이 큰 디지털 데이터의 특징으로 인해 신속한 증거 데이터 확보는 디지털 포렌식에서 중요한 요소이다. 그러나 앞서 언급한 안드로이드 스마트폰의 단편화, 다양성, 빠른 업그레이드 주기는 기존 포렌식 기법을 신규 출시되는 스마트폰에 즉시 적용하기 어렵게 만들고 있고 이러한 증거 데이터의 획득 자체가 불가한 문제는 포렌식 과정에서 반드시 해결해야할 문제이다. 백업 앱 기반 데이터 획득 기법은 제조사에서 출시하는 모든 스마트폰이 필수 기능으로 지원하는 백업 앱을 기반으로 동작하기 때문에 신규 스마트폰이 출시되면 즉시 적용이 가능하여 데이터 수집의 즉시성을 확보할 수 있다.

5.2 동일 제조사 스마트폰에 범용적 적용 가능

동일 제조사에서 생산되어 출시되는 모든 스마트폰은 기본적으로 동일한 백업 앱을 기반으로 사용자 데이터를 백업하고 복원할 수 있도록 지원하고 있다. 이는 반대로 디지털 포렌식 데이터 획득 측면에서 동일 제조사의 모든 스마트폰에 범용적으로 적용이 가능함을 의미한다. 타 제조사 또는 안드로이드 스마트폰 전체를 대상으로 범용적으로 적용이 가능한 백업 앱 기반 데이터 획득 기법에 대한 연구는 향후 연구 과제로 진행할 예정이다.

5.3 삭제한 데이터 복구 지원

궁극적으로 물리적 데이터 획득 기법을 통한 전체 플래시 메모리의 이미징을 목표로 데이터 획득 기법이 연구되고 있으나, 현실적으로 물리적 획득이 어려워지는 추세이기 때문에 일반적으로 논리적 데이터 획득 기법을 기반으로 증거 데이터 수집 작업을 진행한다. 그러나 백업 앱 기반 데이터 획득 기법은 실험을 통해 논리적 데이터 획득 기법으로 수집할 수 있는 데이터에 추가하여 데이터의 무결성을 확보한 상태에서 DB 형태로 저장된 데이터의 삭제된 데이터에 대한 복구도 지원하기 때문에 사용자 데이터를 분석함에 있어 추가적인 데이터를 획득을 가능하게 한다.

VI. 결 론

스마트폰에 저장된 사용자의 데이터를 해킹으로부터 안전하게 보호하기 위해 강화된 보호 기법들이 적용되고 있다. 그러나 이러한 보호 기능들은 오히려 기존 연구된 포렌식 데이터 획득 기법들의 적용을 어렵게 만들고 있다. 이에 본 논문은 제조사가 기본 제공하는 백업용 모바일 앱을 분석하여 이를 활용하여 포렌식 데이터 획득 관점에서 증거 데이터를 확보하기 위한 방법을 설명하였다. 또한 본 논문은 주관적인 판단이 아닌 기존에 연구된 포렌식 데이터 획득 기법을 분석하여 비교하였으며 실험을 통해 획득한 데이터가 원본 데이터와 동일한지 여부를 입증함으로서 증거 데이터로서의 객관성을 확보할 수 있음을 확인하였다.

제조사별 백업 모바일 앱의 동작 방식, 암호화 적용 여부, 임시 데이터의 생성 위치 등 세부적인 사항은 제조사에 따라 다르게 구현될 수 있다. 따라서 본 논문에서 제시한 분석 과정을 다른 제조사의 백업용 모바일 앱 분석에 활용할 경우 마찬가지로 각 제조사별로 범용적으로 사용 가능한 포렌식 데이터 획득 기법으로 이용이 가능할 것으로 판단된다. 하지만 본 논문은 백업 용 모바일 앱을 분석하는 과정에서 특정 제조사를 한정하여 해당 제조사에서 생산한 스마트폰에만 범용적으로 적용 가능한 데이터 획득 기법을 연구하였기 때문에 추후 안드로이드를 사용하는 모든 스마트폰에 범용적으로 적용이 가능한 백업용 모바일 앱 기반 데이터 획득 기법에 대한 연구가 향후 과제로 남아있다.

References

[1] GSMA, "Global Mobile Trends," [Internet],https://www.gsma.com/globalmobiletrends

[2] Google, "Android 7.0 Nougat," [Internet], https://developer.android.com/about/versions/nougat/index.html

[3] K. Kim, D. Hong and J. Ryu, "Forensic Data Acquisition from Cell Phone using JTAG Interface," Proceedings of the 2008 International Conference on Security & Management, pp. 410-414, Jul. 2008.

[4] S.Y. Willassen, "Forensics and the GSM mobile telephone system," International Journal of Digital Evidence, vol. 2, no. 1, Jan. 2003

[5] L. Pierce and S. Tragoudas, "Multi-level secure JTAG architecture," On-Line Testing Symposium (IOLTS) IEEE 17th International, pp. 208-209, Jul. 2011.

[6] E. Casey and G.J. Stellatos. "The impact of full disk encryption on digital forensics," ACM SIGOPS Operating Systems Review, vol. 42, no. 3, pp. 93-98, Apr. 2008

[7] ARM, "TrustZone," [Internet], https://developer.arm.com/technologies/trustzone

[8] Samsung, "Knox," [Internet], https://www.samsungknox.com/en/knox-platform/knox-security

[9] Google, "Verified Boot," [Internet], https://source.android.com/security/verifiedboot/

[10] J.S. Dwoskin and R.B. Lee. "Hardware-rooted trust for secure key management and transient trust," CCS '17 Proceedings of the 14th ACM conference on Computer and communications security, pp. 389-400, Oct. 2007.

[11] T. Vidas, C. Zhang and N. Christin. "Toward a general collection methodology for Android devices," Digital Investigation, vol. 8, pp. S14-S24, Aug. 2011
[12] N. Son, Y. Lee, D. Kim, J. James, S. Lee and K. Lee, "A study of user data integrity during acquisition of Android devices," Digital Investigation, vol. 10, pp. S3-S11, Aug. 2013
[13] Google, "Android Debug Bridge," [Internet], https://developer.android .com/studio/command-line/adb.html?hl=ko
[14] Z. Jovanovic and D. Redd, Android forensics techniques, International Academy of Design and Technology, Bulleproof, Jan. 2012
[15] Google, "Android Full Disk Encryption," [Internet], https://sour ce.android.com/security/encryption/full-disk
[16] R.E. Tulloss, "IEEE Standard Test Access Port and Boundary-Scan Architecture," IEEE 1149.1-1990, Feb. 1990
[17] Riff Box, "Flasher," [Internet], http://www.riffbox.org/
[18] Octupus, "Octopus Box," [Internet], https://octoplusbox.com/en/features/jtag/
[19] S. Yang, J. Choi, K. Kim and T. Chang, "New acquisition method based on firmware update protocols for Android smartphones," Digital Investigation, vol. 14, no. 1, pp. S68-S76, Aug. 2015
[20] E. Nikolay, "Android Backup Extractor," [Internet], https://github.com/nelenkov/android-backup-extractor
[21] A. Hoog, Android forensics: investigation, analysis and mobile security for Google Android, 1st Ed., Syngress, Jun. 2011
[22] G. Lee, H. Hwang, K. Kim and T. Chang, "Analysis Scheme on Backup Files of Samsung Smartphone available in Forensic," KIPS Transactions on Computer and Communication Systems, 2(8), pp. 349-356, Aug. 2013
[23] J. Rongen and Z. Geradts, "Extraction and Forensic Analysis of Artifacts on Wearables," International Journal of Forensic Science and Pathology, vol. 5, no. 1, pp. 312-318, Jan. 2017
[24] J. Park, H. Chung, Y. Son and S. Lee, "Design and Implementation of Analysis Techniques for Fragmented Pages in the Flash Memory Image of Smartphones," Journal of the Korea Institute of Information Security and Cryptology, 22(4), pp. 827-839, Jan. 2012
[25] S. Jeon, K. Byun, J. Bang, G. Lee and S. Lee, "The Method of Recovery for Deleted Record in the Unallocated Space of SQLite Database," Journal of the Korea Institute of Information Security and Cryptology, 21(3), pp. 143-154, Jun. 2011
[26] MD5sum, "Generate MD5 hashes of files," [Internet], http://www.pc-t ools.net/win32/md5sums/
[27] B.R. Joshi and R. Hubbard, "Forensics Analysis of Solid State Drive (SSD)," 2016 Universal Technology Management Conference (UTMC), pp. 1-12, May. 2016
[28] M. Saxena and M.M. Swift, "FlashVM: Virtual Memory Management on Flash," USENIX Annual Technical Conference, Jun. 2010
[29] MD-NEXT, [Internet], http://www.hancomgmd.com/product/mobile-forensic-solution/mobile-forensic-software/#md_next
[30] Final Mobile Forensic, [Internet], http://www.finaldata.co.kr/mobile/

[31] UFED Supported Devices, [Internet], http://lang.cellebrite.com/mobile-forensics/support/ufed-supported-devices

〈저자소개〉

최 재 원 (Jaewon Choi) 정회원
1996년~2003년: 한양대학교 전자전기공학부(학사)
2016년~현재: 고려대학교 정보보호대학원 석사과정
〈관심분야〉 보안공학, 위협 리스크 모델링, 디지털포렌식

김 승 주 (Seung-joo Kim) 종신회원
1994년~1999년: 성규관대학교 정보공학과(학사, 석사, 박사)
1998년~2004년: 한국인터넷진흥원(KISA) 팀장
2004년~2011년: 성균관대학교 정보통신공학부 부교수
2011년~현재: 고려대학교 사이버국방학과/정보보호대학원 정교수
2017년~현재: 고려대학교 사이버무기시험평가연구센터(CW-TEC) 부센터장
2004년~현재: 한국정보보호학회 이사
2007년: 국가정보원장 국가사이버안전업무 유공자 표창
2010년: 방송통신위원회 정보통신망 침해사고 민관합동조사단 위원
2011년~현재: (사)화이트해커연합 HARU 및 SECUINSIDE 설립자 및 이사
2012년: 선관위 디도스 특별검사팀 자문위원
2014년~2015년: 육군사관학교 초빙교수
2014년~2016년: 다음카카오 프라이버시 정책 자문위원회 위원
2015년~현재: 방위사업청 방산기술보호 자문관
2016년~2018년: 개인정보분쟁조정위원회 위원
2016년~현재: 산업통상자원부 전략물자기술 자문위원
2016년~현재: 한국카카오뱅크 정보보호부문 자문교수
2017년~현재: 국방보안연구소 정보보호분야 자문위원
2017년~현재: 여신금융협회 신용카드 단말기 시험 인증위원회 위원
〈관심분야〉 보안공학 및 SDL, 위협 리스크 모델링, 보안성 평가/인증, 암호학, Usable Security