모바일 포렌식을 이용한 메신저 증거 비교 분석 연구

황 태 진*, 원 동 호*, 이 영 숙**

요 약

스마트폰 사용이 보편화되면서 자연스럽게 서로간의 소통이 메신저를 통하여 이뤄지게 되었다. 하지만 서로간의 대화공간이 범죄를 공모하는 공간으로도 활용되고 있는 실정이다. 이에 따른 범죄와 관련된 증거들이 스마트폰에 저장된다. 스마트폰의 특성상 저장정보의 삭제를 손쉽게 할 수 있기 때문에 증거를 신속히 확보 하는 것이 중요하다. 따라서 본 논문에서는 국·내외에서 대표적으로 사용 중인 모바일 메신저에 대하여 데이터 파일의 아키텍처 분석을 수행하였다. 사용자가 메시지 삭제를 위해 이용할 수 있는 방법에 대한 시나리오를 설정하고 실험을 통해 메시지 복원가능 여부에 대하여 비교 분석한다.

A study on the Comparison Analysis for Messenger Evidence Using Mobile Forensics

Taejin Hwang*, Dongho Won*, Youngsook Lee**

ABSTRACT

As the use of smartphones become more common, the communication via instant messenger becomes natural. However, it is important to secure the relevant information promptly since the chat room between participants can be used as a space for a criminal conspiracy, and crime-related contents can be stored and deleted easily on smartphones. Therefore, this study aims to identify the available data and to use it as proof by comparing and analyzing the instant messengers with high usage rate.

Key words : mobile forensic, messenger analysis, instant messenger, message recovery

접수일(2018년 5월 9일), 수정일(1차: 2018년 6월 26일, 게재확정일(2018년 6월 29일)

* 성균관대학교 정보통신대학원

** 호원대학교 사이버보안학과(교신저자)

1. 서 론

최근 스마트폰의 사용이 보편화되면서 사용자들에게 중요하고 민감한 정보를 생산하고 활용하는 수단으로 이용되고 있다. 이러한 점이 악용되어 범죄행위에 이용되고 있는 실정이다. 메신저를 통해 전화, 메시지, 사진, 동영상 등 자유로운 의사소통과 정보를 공유할 수 있기 때문이다[1].

디지털 증거를 신속히 획득하는 것이 매우 중요하다. 범죄현장에서 일반 증거보다 디지털 증거가 사건의 방향성을 잡고 실마리를 풀 수 있는 단서가 되는 중요한 정보가 될 수 있기 때문이다. 디지털 증거 확보하는 방안은 법적 효력을 가질 수 있는지의 여부를 결정하기 때문에 포렌식의 절차를 준수하여 획득하는 것이 중요하다[2].

하지만 스마트폰에서 생성된 정보에 대해서 메시지 삭제, 어플리케이션 삭제, 휴대폰 초기화 등을 통해 생성된 정보를 삭제하여 손쉽게 숨기고 증거를 인멸하고 포렌식 분석을 어렵게 할 수 있다. 스마트폰은 PC와 다르게 전문적인 지식이 없어도 휴대폰 초기화, 앱 삭제, 메시지 삭제 등이 용이하며 생성된 데이터에 대한 삭제가 손쉬우며 이를 복구하기는 어렵다. 이에 따라 디지털 증거를 확보하는 방안은 법적 효력을 가질 수 있는지의 여부를 결정하기 때문에 매우 중요하다[3].

본 논문에서는 모바일에서 사용하고 있는 인스턴트 메신저의 현황과 특징을 기반으로 사용자가 메시지를 삭제할 수 있는 시나리오를 설정하여 모바일 포렌식을 수행하고 결론을 정리하고자 한다. 2장에서는 국내·외에서 사용빈도수가 높은 모바일 메신저를 알아본다. 3장에서는 메신저 앱을 대상으로 스마트폰에 저장된 데이터에 대한 아트팩트를 비교분석한다. 4장에서 메시지를 삭제에 따른 메시지 복구 가능여부에 대하여 확인한다. 마지막으로 5장에서 결론을 서술한다.

2. 관련연구

2.1 모바일 인스턴트 메신저 현황

모바일 사용자가 증가함에 따라 가족, 친구 및 직장 동료들 간에 지속적이고 즉각적인 커뮤니케이션의 필요성이 커지고 있다. 또한 모바일 메신저 앱을 사용하여 무료 문자 메시지를 보내고 음성 및 화상통화, 심지어 사진 및 파일 공유에 이르기 까지 다양한 서비스와 상호작용을 제공받고 있다. 소셜 미디어 및 콘텐츠 마케팅에 대한 통계정보를 제공해주는 웹사이트 Statista는 2018년 1월 현재 전 세계에서 가장 많이 사용되는 모바일 메신저 앱에 대한 통계조사를 발표하였다. 1위는 WhatsApp으로 월간 활성 사용자수가 약 13억 명으로 가장 많이 사용되고 있는 것으로 확인되었다. 그 뒤를 잇는 2위는 Facebook Messenger로 12억명의 사용자가 매달 활발하게 사용하고 있으며, 3위는 WeChat, 4위는 QQ Mobile 순으로 나타났다. 국내에서 왕성하게 사용되고 있는 KakaoTalk은 전체 11위로 전 세계 월간 활성 사용자 수는 4천9백만 명이라고 발표했다[4]. 국내 앱 분석 업체 와이즈앱은 국내 모바일 앱 점유율 조사한 결과 국내 사용자는 약 3천만 명이 국내 95%의 점유율로 Kakaotalk을 사용하고 있는 것으로 밝혔다[5].

2.2 모바일 인스턴트 메신저 특징

스마트폰이 보급되기 시작하면서 메신저 앱은 단순히 문자를 대체할 수 있는 수단에 불가하였다. 현재는 사람들이 모일 수 있는 하나의 플랫폼으로 발전하고 있다. 또한 메신저의 기능만 가지고 있는 것이 아니라 무료 통화, 파일 공유, 상품의 판매, 콘텐츠 제공, 광고의 수단 등 다양한 기능과 상업적 수단으로 사용되고 있다. 개인 및 그룹채팅이 가능하고 위치정보, 사진, 동영상. 문서 및 주소록과 같은 항목을 채팅을 하는 사람들과 공유할 수 있고, 음성 통화까지 지원한다. 따라서 메신저에 대한 포렌식을 진행하기 위해서는 앱에서 생성되는 데이터에 대한 아트팩트 분석이 선행되어야한다.

3. 모바일 메신저 아트팩트 분석

메신저 앱 분석은 일대일 채팅, 그룹채팅, 멀티미디어(사진, 동영상, 문서) 전송 등의 사용자에 의해 생성될 수 있는 시나리오를 바탕으로 진행한다. KakaoTa

lk이나 Telegram과 같은 특정 애플리케이션은 연락처 및 채팅 대화내용 등이 DB파일 내에 데이터 값이 암호화 되어 저장된다. 구체적으로 내용을 분석하기 위해서 애플리케이션을 디컴파일 후 소스코드를 분석하여 암호화 로직을 식별하고 암호화 키의 위치를 파악한다. 하지만 키 값을 바탕으로 복호화 로직을 만들어 원문을 추출하는 과정에서 소스코드가 난독화되어 있어 암호로직 파악에 어려움이 있는 경우도 있다. 이는 암호화된 데이터를 복호화하여 원문으로 분석할 수 있는 모바일 포렌식 도구를 이용하여 해결한다.

본 논문에서는 메신저 앱의 생성된 아티펙트를 분석하기 위하여 실제 수사기관에서 사용 중이고 국내에서 가장 많이 사용되고 있는 공신력 있는 상용 모바일 포렌식 도구를 활용하여 데이터를 추출하여 분석하였다.

3.1 메신저 아트펙트 구조

3.1.1 WhatsApp

WhatsApp 어플 내에 저장되는 데이터들은 /data/com.whatsapp/databases 경로 아래 SQLite 데이터베이스에 기록된다. 메시지와 관련된 데이터는'msgstore.db'에 저장되어 있고 연락처 정보와 관련된 데이터는 'wa.db'파일로 구분되어 저장된다.

저장되어 있는 데이터는 평문으로 저장되어 있기 때문에 쉽게 메시지와 관련된 데이터를 확인이 가능하다. 무료통화기능(음성통화, 영상통화)을 지원하고 있어 이에 대한 정보도 식별이 가능하다, 채팅방 내에서 내려 받은 멀티미디어(사진, 음성, 영상, 문서) 파일에 대해서는 /media/WhatsApp/Media 경로 아래서 확인이 가능하다[6].

3.1.2 Facebook messenger

Facebook messenger 어플 내에서 주고받은 메시지와 관련된 데이터들은 'threads.db2' 파일에서 확인 가능하다. 통화기록은 'call_logs_db[user_id]' 파일에 기록된다. 채팅방 내에서 미디어파일(이미지, 음성, 동영상)을 공유할 수 있으며 공유된 정보는 DB파일에 기록되지만 실제 파일은 별도 경로에 저장된다.

3.1.3 Telegram

Telegram은 어플 내에서 생성되는 주요 대부분의 데이터들은 'cache4.db' 파일에 저장한다. 채팅방 내에서 미디어 파일(이미지, 동영상, 음악, 파일), 연락처, 위치정보를 공유가 능하다. 공유된 정보는 DB파일에 기록되지만 실제 파일은 별도 경로에 저장된다. 따라서 대화내용 및 공유한 파일에 대한 추적 및 분석하기 위해서는 DB파일과 실제파일이 저장된 경로를 동시에 확인해야 한다.

Telegram 내에서의 대부분의 데이터는 SQLite 데이터베이스에 기록된다. 애플리케이션의 환경설정, 권한, 속성 값이 기록되어 있는 Preference는 xml 포맷으로 /data/org.telegram.messenger 경로 아래 각각의 databases, shared_prefs 폴더 아래 저장되어 있다. 이를 제외한 멀티미디어 파일(문서, 사진, 음성, 동영상, 캐시)은 /media/Telegram/ 경로 아래서 확인이 가능하다.

3.1.4 LINE

LINE을 사용할 때 생성되는 데이터들은 /data/jp.naver.line.android/databases 경로 아래 존재한다. 그중 메시지에 대한 관련 정보는 'naver_line', 'call_history' 파일에 평문으로 저장되어 있다.

발・수신 메시지 및 통화내역에 대한 정보를 추적하기 위해서는 데이터베이스 내에 존재하는 여러 테이블과 연계분석을 하여야 한다. 또한 발・수신 메시지 상에 첨부된 파일(이미지, 음성, 문서 등)에 대해서는 /media/Android/data/jp.naver.line.android/storage 경로 아래 채팅방을 구분해주는 고유 식별 ID값으로 폴더가 생성되어 저장된다. 이 부분은 첨부파일의 파일명이 파일의 출처를 추적하는데 중요한 단서가 될 수 있다[7].

3.1.5 KakaoTalk

KakaoTalk 어플 내에서 생성되는 데이터들은 /data/com.kakao.talk/databases 및 /data/com.kakao.talk/files 경로 아래 'KakaoTalk.db'와 KakaoTalk2.db' 파일에 구분되어 저장된다. 이에 대한 백업파일이'103466717.backup'와' - 1087431639.backup' 파일로 존재한다. 필드 값(이름, 메시지, 전화번호 등)은 암호화된 상태로 저장되어 일반적으로 파일을 획득했다하더라도 DB테이블 내 필드

값이 복호화 상태가 아니면 정상적인 내용 분석이 불가능하다.

KakaoTalk 채팅은 일반채팅, 그룹채팅, 비밀채팅, 오픈채팅(그룹채팅, 개인채팅) 5개 유형을 서비스한다. 채팅방 내에서 사진, 동영상, 음성메시지, 지도, 연락처, 파일을 전송할 수 있으며 무료통화 및 영상통화 기능이 존재한다. 따라서 이러한 각 유형에 따라 데이터 저장상태가 다르게 기록되어 있다.

송・수신된 미디어파일(사진, 동영상, 음성)은 /media/Android/data/com.kakao.talk/cache/ 경로 아래 'chat_logs_chat_id'값의 폴더이름 아래 파일들을 저장한다. 또한 사진 및 동영상파일에 대한 이미지 캐시는 /media/Android/data/com.kakao.talk/cache/[chat_id] 경로 아래 존재한다. 실제 데이터는 별도의 저장 폴더에 기록된다[5].

4. 메신저 대화내용 포렌식 분석

메신저에 저장된 대화내용에 대한 포렌식을 수행하는 데 있어 4가지의 시나리오를 설정하였다.

(1) 개별 메시지 삭제
채팅방 내에서 주고받은 메시지들 중에서 삭제하고자 하는 메시지만 선별 삭제 하여 흔적을 재거하는 경우

(2) 채팅방 삭제
상대방과 함께 참여중인 채팅방 자체를 삭제하여 흔적을 제거 경우

(3) 어플리케이션 삭제
설치한 메신저 어플리케이션을 기기에서 삭제하여 흔적을 제거하는 경우

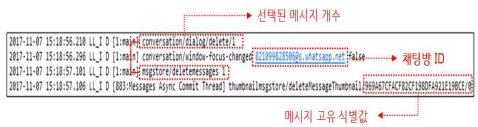
(4) 기기 공장 초기화
스마트폰을 공장 출하 상태로 만들어 매시지 흔적을 제거하는 경우

4..1 WhatsApp

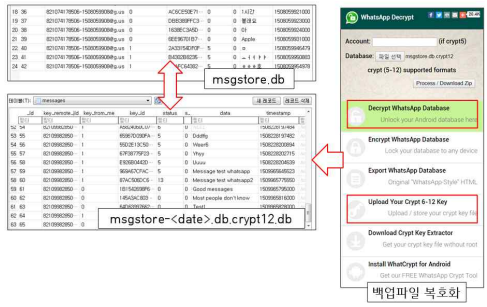
메시지에 대한 정보는 'msgstore.db'파일의'message's' 테이블에서 관련 데이터를 확인 할 수 있다. 채팅방 내에서 주고받을 수 있는 메시지는 유형 총 6개(텍스트, 이미지, 음성, video, 연락처, 위치)로 그 유형에 따라 'media_wa_type' 컬럼의 값으로 구분 가능하다. 발・수신 메시지는 'key_from_me'컬럼 값을 통해 확인할 수 있다[3].

(1) 단일메시지 삭제

채팅방에서 메시지를 개별 삭제 한 경우는 /data/com.whatsapp/files/Log 경로 아래 whatsapp.log에서 삭제된 메시지에 대한 이벤트의 정보를 [그림 1]과 같이 확인 할 수 있다. 삭제된 메시지는 'msgstore.db', 'msgstore.db-wal', msgstore-<date>.crypt 파일에서 확인되었다. 로그에서 확인된 삭제정보와 DB파일 내 chat_list, messages 테이블에서 고유 식별 값을 비교하여 삭제 메시지를 식별할 수 있다. msgstore-<date>.crypt 파일은 백업파일로 AES 192 알고리즘으로 암호화되어 있어 [그림 2]와 같이 복호화 후 분석하였다[6, 7].



[그림 1] 단일 메시지 삭제 흔적



[그림 2] 백업 파일 복호화

(2) 채팅방 삭제

채팅방을 삭제한 시나리오에서는 삭제된 채팅방 I

D, 채팅방 내 삭제된 메시지의 개수, 메시지 삭제 시간에 대한 정보만 로그파일에서 확인 가능하다. 채팅방 삭제에 따른 메시지 흔적은 [그림 3]에 나타내었다.

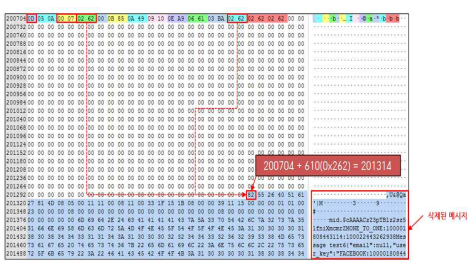
[그림 3] 채팅방 삭제 시 흔적

4..2 Facebook messenger

메시지에 대한 정보는 'threads.db2' 파일 내 'messages' 테이블에서 관련 데이터를 확인 할 수 있다. 채팅방에서 주고받은 메시지와 관련된 정보(메시지 내용 및 일시, 발・수신자, 첨부파일 정보 등)를 확인 가능하다.

(1) 단일메시지 / 채팅방 삭제

단일 메시지 및 채팅방을 삭제 한 경우 threads_db2 파일에서 삭제된 메시지의 식별이 가능하였다. 기존 실험한 메시지 앱 들과는 달리 삭제된 메시지는 삭제 전 DB 위치했던 곳에 그대로 존재하고 있었다. 삭제 메시지를 추적하기 위해서 [그림 4]와 같이 메시지 내용을 저장하고 있는 페이지의 헤더로 이동하여 삭제된 메시지에 의해 변화된 레코드 주소영역을 찾아 삭제된 레코드의 추적한다.



[그림 4] 삭제된 메시지 추적

4..3 Telegram

Telegram에 생성된 메신저 정보는 'messages' 테이블에서 확인 가능하다. 메시지 내용은 data 컬럼에 기록되고 데이터 타입은 TEXT 타입이 아닌 BLOB (입력된 값 그대로 바이너리로 저장) 타입으로 저장된다. 상대방이 메시지를 확인했는지 유무는 read_state의 값으로 알 수 있다. 채팅방 참여자가 모두 확인을 하였다면 인원수만큼의 숫자가 카운트되기 때문에 채팅방의 참여 인원을 파악하는 정보로도 활용할 수 있다.

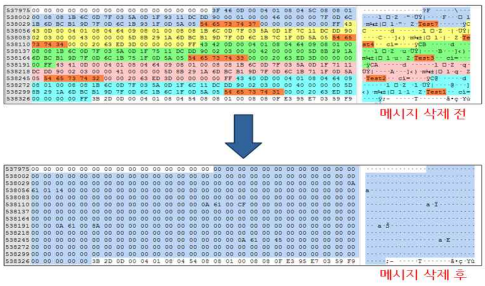
(1) 단일메시지 삭제

채팅방에서 발・수신한 메시지를 개별적으로 삭제를 하였을 경우 [그림 5]에서 보이는 것처럼 메시지의 정보를 담고 있는 셀의 영역이 '0x00'으로 채워진다. 따라서 채팅방 내에서 일부 메시지를 삭제한 흔적은 확인되지만 어떤 메시지를 삭제하였는지 확인이 불가능하다. 메시지의 내용 확인은 어렵지만 cache4.db-wal 파일에서 삭제된 메시지가 존재하는 것을 확인할 수 있다.

[그림 5] 메시지 삭제 전/후 상태

(2) 채팅방 삭제

채팅방 내에 존재하는 메시지에 대하여 채팅방 자체를 삭제 하였을 경우 역시 채팅방 내 존재하는 메시지에 대한 정보를 담고 있는 모든 셀의 영역이 [그림 6]과 같이 '0x00'으로 채워진다. 채팅방에 대한 정보, 메시지 내용 등 정보일체의 확인이 불가능하였다. 하지만 cache4.db-wal 파일에서 삭제된 채팅방의 메시지가 존재하는 것을 확인 가능하였다.



[그림 6] 채팅방 퇴장 후 상태

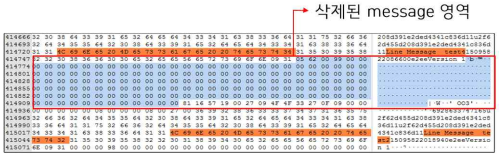
4..4 LINE

메시지에 대한 정보는 'chat_history' 테이블에 채팅방, 참여자, 메시지 내용, 메시지 생성 일시, 첨부 메시지의 유형이 구분되어 저장된다.

메시지 유형이 텍스트일 경우'content' 컬럼에 내용이 저장되지만 사진, 동영상, 음성, 연락처, 문서 등 첨부파일 형태의 메시지일 경우'content' 컬럼에는 데이터가 비어있다[8].

(1) 단일메시지 삭제

채팅방에서 발・수신한 메시지를 개별적으로 삭제하였을 경우 [그림 7]에서 보이듯이 메시지에 대한 해당영역이 '0x00'으로 채워진다. 'chat_history'테이블에 기존 메시지에 대한 데이터는 삭제되어 삭제된 흔적만 'id'의 번호로 유추할 수 있을 뿐 기존 메시지는 확인이 불가능하다.



[그림 7] 메시지 삭제 후 메시지 상태

(2) 채팅방 삭제

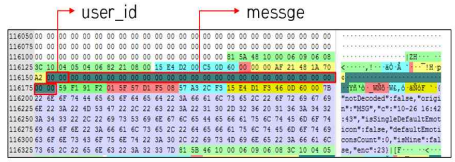
참여하고 있는 채팅방을 퇴장하는 방법으로 메시지를 삭제하는 경우는 메시지가 저장되어 있던 영역이 모두 '0x00'으로 변경된다. 따라서 기존 메시지에 대한 내용 확인은 불가능하다.

4..5 KakaoTalk

메시지에 대한 정보는 'chat_logs'테이블에 저장되어 있다. 메시지 생성 시간, 삭제시간, 메시지 내용, 첨부파일에 대한 정보를 확인가능하다. 메시지 유형에 대한 것은 'type'컬럼 값을 통해 시스템 메시지, 텍스트 메시지, 음성, 동영상, 사진 등 전송된 발・수신된 메시지의 유형을 식별할 수 있다[9].

(1) 단일메시지 삭제

채팅방에서 발・수신한 메시지를 개별적으로 삭제를 하였을 경우를 [그림 8]에 나타내었다. 'user_id'와 'message'컬럼의 값을 확인한 결과 기존 메시지는 삭제되고 26byte의 고정된 길이가 '0x00'으로 채워졌다. 'deleted_at'컬럼에는 삭제된 시간이 기록된다. 채팅방 내에서 언제 특정 메시지를 삭제했는지에 대한 흔적은 확인되지만 기존 메시지의 내용 확인은 불가능하다.



[그림 8] 메시지 삭제 후 상태

(2) 채팅방 삭제

채팅방을 삭제한 경우 단일 메시지를 삭제하는 경우와는 다르게 기존 채팅방에서 대화했던 메시지의 일부 확인이 가능하다. 채팅방을 퇴장 하면 채팅방 메시지는 모두 삭제되어 'KakaoTalk.db'파일 내에서는 삭제된 메시지의 내용확인이 불가능하다. 하지만 'KakaoTalk.db-journal' 파일에서 일부 확인이 되었다.

4..6 애플리케이션 삭제

메신저 앱을 삭제 시 data 폴더에 생성된 메신저 앱 폴더 전체가 제거가 되면서 채팅 관련 DB 파일을 포함한 모든 파일이 삭제된다. 삭제된 파일은 비할당영역 또는 슬랙영역에 존재한다. 메시지 DB 파일에 대한 전체 또는 조각파일을 복원하기 위해 키워드 검색을 실시하여 존재여부를 확인하였으나 발견되지 않았다.

한편 /data/com.android.vending/databases 경로 아래 localappstate.db 파일은 안드로이드 기기에 설치된 어플리케이션에 대한 정보를 저장하고 있다. 그곳에는 사용자에 의해 설치된 메신저앱에 대한 정보도 포함되어 있다. 사용자가 어플리케이션을 삭제하여도 기존에 기록된 정보는 삭제되지 않기 때문에 삭제된 앱의 정보(앱이름, 설치일자) 정도 까지만 확인할 수 있다.

4..7 디바이스 초기화

안드로이드 기기를 초기화 하는 방법으로 사용자가 메시저를 통해서 생성한 메시지를 삭제할 수 있다. 이런 경우, 어떤 메신저 앱이 설치되어 있었는지 조차 식별이 불가능 하였으며 메시지 복구 역시 불가능하였다. 사용자가 디바이스를 초기화하면 안드로이드 기기는 초기화 상태로 만들기 위해 recovery를 실시하게 되며 이와 관련된 시스템 로그가 /log 폴더 아래 recovery.log 파일로 생성된다. 이 파일에 대한 시간정보를 이용하여 초기화를 실시한 시점을 파악할 수 있다.

5. 결 론

스마트폰 사용이 보편화되면서 자연스럽게 서로 간의 소통이 메신저를 통하여 이뤄지게 되었다. 하지만 서로간의 대화공간이 범죄를 공모하는 공간으로도 활용되고 있는 실정이다. 이에 따른 범죄와 관련된 증거들이 스마트폰에 저장되어 있을 가능성이 크며 스마트폰의 특성상 저장정보를 손쉽게 삭제 할 수 있기 때문에 증거를 신속히 확보 하는 것이 중요하다.

실험을 통해 확인된 포렌식 분석 결과를 통해 사용자가 안티(삭제) 행위를 할 경우 모바일 메신저 종류에 따라 분석가능 여부를 사전에 식별하고 삭제된 메시지에 대하여 복원하여 확인 할 수 있었다.

이러한 결과를 가지고 두 가지 측면에서 활용해 볼 수 있다. 첫 번째는 삭제된 데이터가 존재할 수 있는 파일들을 식별하여 획득한 후 삭제 데이터 찾아 복원을 하는 것이다. 두 번째는 사용자의 안티(삭제) 행위를 추적할 수 있다. 디지털 포렌식은 결과뿐만 아니라 행위에 대한 과정을 밝혀내는 것도 중요하다. 사용자의 안티행위로 인해 데이터를 복원을 하지 못하였지만 그 행위를 밝혀냄으로써 또 다른 증거(증거인멸)로 제시할 수 있기 때문이다. 이러한 측면에서 KakaoTalk(개별 삭제)과 LINE의 경우 메시지 삭제 시 복원은 할 수 없었지만 삭제 시 DB 파일 내 메시지가 존재했던 영역이 0x00 값으로 채워지는 뚜렷한 변화가 나타나는 것을 확인 할 수 있었다.

이러한 연구 결과를 바탕으로 메신저 포렌식 분석에 활용하고 또한 삭제된 메시지에 대한 복원이 불가능하더라도 데이터의 삭제 흔적 식별하여 어떻게 메시지 삭제를 하였는지에 대한 행위를 추적하는 등 안티 포렌식 행위도 찾아낼 수 있을 것이라 판단된다.

참고문헌

[1] 최우용, 은성경 “스마트폰 포렌식 기술 동향”, 전자통신동향분석, 제28권 제3호, 2013.6

[2] 윤종철, 박용석, “KakaoTalk의 채팅 메시지 포렌식 분석 연구 및 WhatsApp의 Artifacts와의 비교 분석”, 한국정보통신학회논문지, Vol.20, No.4, pp.777-786 2016.4

[3] 어수웅, 조우연, 이석준, 손태식 “모바일 포렌식 증거능력 확보 방안 연구”, 정보보호학회논문지, Vol.26, No.1, pp. 135-152 2016.2]

[4] astatista, “Most popular messaging apps 2018” https://www.statista.com/statistics/258749/most-popular-global-mobile-messenger-apps.html. (Accessed Aprill 3, 2018)

[5] 지디넷코리아, “카톡, 국내 모바일 메신저 점유율 95%”,http://www.zdnet.co.kr/news/news_view.asp?artice_id=20171121083511.(2018-3-21 방문)

[6] Cosimo Anglano, “ Forensic Analysis of Whats App Messenger on Android Smartphones”, Digital Investigation: The International Journal of Digital Forensics & Incident Response, vol11, pp. 6-15, 2014. 9

[7] “WhatCrypt Tools”, http://whatcrypt.com

[8] Asif Iqbal, Hanan Alobaidli, Ahmed Almarzooqi, Andy Jones, “ LINE IM app Forensic Analysis”, 12th International Conference on High-cap

acity Optical Networks and Enabling/Emerging Technologies, 2015.12

[9] 윤종철, 박용석, “안드로이드 환경에서의 KakaoTalk 메신저의 포렌식 분석 방법론 제안 및 분석”, 한국정보통신학회논문지, Vol.20, No.1, pp. 72-80, 2016.6

[저 자 소 개]

황 태 진 (Taejin Hwang)
2013년 3월~현재 성균관대학교 정보통신대학원 정보보호학과 재학
2014년 5월~현재 국방부조사본부 사이버범죄수사대 사이버수사관

email : p0897@paran.com

원 동 호 (Dongho Won)
1976년 2월 성균관대학교 전자공학과 공학사
1978년 2월 성균관대학교 전자공학과 공학석사
1988년 2월 성균관대학교 전자공학과 공학박사
1978년~1980년 한국전자통신연구원 전임연구원
1985년~1986년 일본 동경공업대학교 객원연구원
2002년~2003년 한국정보보호학회장
1982년~2015년 성균관대학교 컴퓨터공학과 교수
2015년 3월~현재 성균관대학교 컴퓨터공학과 행단석좌 교수

email : dhwon@security.re.kr

이 영 숙 (Youngsook Lee)
2009년 3월~현재 호원대학교 사이버보안학과 부교수
2008년 8월 성균관대학교 컴퓨터공학 박사
2005년 2월 성균관대학교 석사
1987년 2월 성균관대학교 정보공학사

email : ysooklee@howon.ac.kr