Forensic Analysis of Kik Messenger on Android Devices

Olawale Surajudeen Adebayo
Federal University of Technology Minna
16 PUBLICATIONS **122** CITATIONS

John Alhassan
Federal University of Technology Minna
59 PUBLICATIONS **141** CITATIONS

Shafi'i Muhammad Abdulhamid
Federal University of Technology Minna
105 PUBLICATIONS **1,393** CITATIONS

FORENSIC ANALYSIS OF KIK MESSENGER ON ANDROID DEVICES

Olawale Surajudeen Adebayo, Salamatu Aliyu Sulaiman, Oluwafemi Osho, Shafi'I Muhammad Abdulhamid and John K Alhassan
Department of Cyber Security, Federal University of Technology, Minna, Nigeria.
salma.aliyu824@gmail.com, waleadebayo@futminna.edu.ng, femi.osho@futminna.edu.ng, shafii.abdulhamid@futminna.edu.ng, jkalhassan@futminna.edu.ng

ABSTRACT

The impact of forensic evidence found on smartphones cannot be overemphasized when compared to that found on their digital counterparts such as personal computers (PCs). Recently, third-party instant messaging applications have gradually replaced the traditional messaging applications and as such they contain a large amount of information which are far from forensic solutions. This seminar paper focuses on the forensic analysis of Kik messenger: a multi-platform instant messaging or chat application on android devices. Forensic images of three android devices with android versions 4.4 (KitKat), and 5.0 (Lollipop) and different android manufacturers are captured and data related to Kik are identified and examined. Artefacts of forensic values are then identified and analyzed. The result of this research will help digital forensic investigators and academia alike in locating and acquiring digital evidence from Kik messenger on android platforms.

Keywords: *android forensics; Kik messenger forensics; mobile device forensics; third-party mobile app forensics;*

1 INTRODUCTION

Smartphones are usually kept in close physical proximity to their owners; the significance of digital evidence found on them cannot be matched to the ones found on other sources like personal computers (PCs), desktop computers, and tablet computers. Most mobile device users opt for third-party instant messaging (IM) applications for several reasons which may include enhanced features not available on the messaging apps embedded on their devices. These IM apps have a large amount of information which could be used for forensic investigations; however; most mobile forensic software solutions often just analyse smartphone data such as phonebook, text messages and call records. As a result, information which can serve as digital evidence is often overlooked by these software solutions. This research endeavour focuses on conducting a forensic analysis on Kik Messenger which is a multi-platform IM app on android smartphones.

Kik Messenger has been synonymous with some child exploitation crimes, although not limited to only Kik (Ovens & Morison, 2016). Kik Messenger has gained quite a number of popularity among the younger generation, as four out of ten US teens use the application. Although there has been a tremendous amount of research on forensic analysis of instant messaging applications, at the time of this study there has been no extensive research work on the forensic analysis of Kik messenger on the android platform. With android's growing market reach, and Kik's popularity, a forensic analysis of the IM app on the android platforms is essential.

The aim of the study is to carry out a forensic analysis of Kik instant messaging application on an android device. The objectives are to identify and recover forensic artefacts by using Kik messenger on android devices; also to analyse and interpret the artefacts.

It will interest law enforcement to know that Kik do not have a database of sent and received messages and is unable to recover any sent or received messages (Kik, 2015). Consequently, it is imperative that forensic analysts device methods of retrieving as much artefacts as possible from suspects' seized mobile devices to support examinations.

Results obtained from this study will add value to the forensic community's knowledge of the types of forensic artefacts that are likely to remain after the use of Kik messenger on devices running the android operating system. This research presents the first comprehensive forensic analysis of Kik on Android devices leaving Windows, Amazon and other mobile device operating system outside the scope of this study and is therefore recommended for further research.

This paper is organized into five sections. The remainder of this paper is structured as thus: Section 2 provides an overview of related research works conducted in the field of instant messaging applications forensics and approaches for procuring data from android devices are discussed. Section 3 describes the experiments undertaken to address the activities in a forensic investigation. The results and analysis of the experiments are reported in Section 4 while Section 5 presents conclusions drawn from the study and proposes possibilities for future research.

2 RELATED LITERATURE

There has been a lot of existing research works on IM application forensics on smartphones, however, both the IM apps and smartphones' operating systems are constantly being developed thereby requiring constant research in the area to fill the unending gap. In this chapter, a concise summary for each related work is presented.

Al Mutawa, Baggili, & Marrington (2012) conducted forensic analysis on three social network applications: Facebook, Twitter and MySpace installed on three mobile platforms: Apple iOS, Blackberry and Android. Logical forensic images of the devices were captured and manual forensic analysis was conducted on each device for all the social network applications. Logical acquisition on the android device was achieved by first rooting it to get access to protected directories that contain user data and then fully backing up with a backup application. The study revealed that the most forensic evidence was recovered on the Apple and Android devices and none on the blackberry device.

A research on 20 most popular instant messaging applications was carried out by Walnycky, Baggili, Marrington, Moore, & Breitinger (2015) on android devices. Network traffic and data stored on the device were retrieved and analysed. Being able to intercept and reconstruct data during the use of these apps was the focus of this study. The study showed that most of the applications examined employed poor security and privacy measures. Logical acquisition of the android device was performed using XRY forensic solution and validated with another backup application and later extracted and analysed using required tools. Their work is closely related to this study in that Kik messenger was one of the applications examined but the study involved a large amount of IM apps and so the analysis of Kik and also due to differing goals.

A thorough forensic analysis on the artefacts generated by WhatsApp Messenger on the android platform is presented by Anglano (2014). Importance is placed on relating the artefacts recovered so as to gather more meaningful evidence than when the artefacts are examined alone. Software android emulators are employed for each device in the research, whose internal memory are then imaged using FTK Imager and extracted and analysed with suitable tools. The study provides guidelines for forensic investigators for deciphering, analysis, and connection of WhatsApp Messenger artefacts on Android devices and also sets a benchmark against WhatsApp extraction tools measured.

Mahajan, Dahiya, & Sanghvi (2013) and Sgaras & Kechadi (2014) presented an investigation on forensic artefacts left by WhatsApp and Viber instant messaging applications; however the main focus of Sgaras & Kechadi (2014) was the Tango VoIP. They compared the forensic analysis of WhatsApp and Viber to that of Tango and discovered that artefacts left by Tango were encrypted while in (Mahajan et al., 2013) focused on determining what kind of instant messenger application related data and information could be retrieved on the device's internal memory. Both of them used Cellebrite UFED forensic solutions to acquire, extract and analyse the relevant files and both works was on android platforms.

Satrya, Daely, & Shin (2016) focus on the forensic analysis of the private chat feature of IM apps on android devices. The IM apps used as case studies are Kakao, Telegram, and Line. They conducted both offline and live logging forensic investigations on the test devices. The test devices were rooted, backed up, and relevant data were extracted and analysed using required software. This research work is similar in terms of the mobile platform and but differs in terms of the applications examined as well as the chat feature.

A research was presented in (Iqbal, Marrington, & Baggili, 2014) on the forensic analysis on Samsung's ChatOn instant messaging application on both Android and iOS devices. A bit stream image of the data directory on the android device was acquired by injecting a payload into it thereby providing temporary root access and using the DD command. Manual inspection of the files generated by the two devices revealed that the location of the files sent in the course of chats as well as their timestamp could be identified. Although android devices were among the ones analysed in this work, it differs from the one presented in this paper both in terms of methodology used and the IM application.

Forensic artefacts generated by ChatSecure, a privacy preserving instant messenger application on Android smartphones is analysed by Anglano, Canonico, & Guazzone (2016). They use 3 android emulators to mimic 3 android smartphones with different versions with their statuses being monitored by the android device monitor. Data acquisition is achieved using the pull function of the android device monitor file explorer. The results of their work show that chat history is stored in encrypted databases and present a technique for decrypting them using security controls that could be retrieved from device memory. How to correlate the data and reconstruct the order and contents of messages is also presented. This work also differs from the one presented in this paper both in terms of methodology used and the IM application.

A broad forensic analysis of Line IM app on android platform is described in (Iqbal, Alobaidli, Almarzooqi, & Jones, 2015). Data acquisition on the android device is achieved by temporary rooting the device and getting a bit by bit image of the data partition using the Unix DD command. Their research revealed that most evidence could be retrieved but the hidden message features of Line IM app could only be partially recovered.

A comprehensive forensic analysis of Kik messenger on Apple iOS devices is reported by (Ovens & Morison, 2016). Artefacts created or modified by the instant messenger application were thoroughly described and used to answer the typical questions in the course of investigations. Their research revealed that deleted evidence in the form of attachments could be retrieved both from the device, device backups and the Kik servers.

Of the related research work reviewed, only the works Ovens & Morison (2016) and Walnycky et al. (2015) is related to this research endeavor in terms of instant messaging application analysed. However, our work is different to that in (Ovens & Morison, 2016) on account of mobile platform being analysed while that of Walnycky et al. (2015) is not exhaustive. Also the experiments were on older Kik versions which have been improved. This research endeavor attempts to fill this gap.

3 METHODOLOGY

3.1 EXPERIMENT SETTING AND APPARATUS

Before carrying out the investigations, a forensic workstation is configured and all the required hardware was set up and software was installed. Similar to the presentation in (Al Mutawa et al., 2012). The configurations of the test devices were not altered as doing that would possibly affect the size of digital evidence to be discovered on them. The list of the hardware and software used to for the research is given in the table below.

TABLE I: HARDWARE/SOFTWARE USED

Hardware/Software	Usage	Specifications
Laptop	Forensic investigation and analysis	HP, windows 8.1, intel CORE i3
HTC One M8	Device to be analyzed	Android V5.0.1
HTC One mini	Device to be analyzed	Android V4.4.2
Samsung Galaxy S3	Device to be analyzed	Android V4.4.2
Titanium Backup	To backup android devices	Version 7.6.1
WinRar	To unzip and extract backup file	Version 5.30.0
DB browser for SQLite	To view .db files	Version 3.9.1
KingRoot	To the android devices	Version 4.5.0
Kik	IM app to be analyzed	Version 11.4
Micro SD card & card reader	To save device backup and read SD card on the laptop	

3.2 INVESTIGATION PROCEDURE

The investigation process consisted of three main stages:

3.2.1 PRELIMINARY DEVICE SETUP

The first stage involved preparing the android devices, carrying out user activities on the test application. The Kik messenger application was downloaded and installed from Google PlayStore.

Fictional user accounts were created on the application and predefined set of activities were performed on each of the test device. These activities includes uploading profile pictures, text chatting, video chatting, creating group chats, group membership, sharing sketches, surfing the web within the application, as well as blocking contacts, deleting contacts and deleting messages. These activities would interest forensic examiners especially exchanging images, sketches, deleting messages and contacts and blocking contacts. In order to examine whether the investigation was accurate, the number of contacts in the phone must be stated and recorded before deletion, the remaining contacts before and after the investigation are compared to ensure the accuracy.

This stage was simple and same for all the different smartphones.

3.2.2 ACQUISITION

The second stage involved manually acquiring an image of the Android devices using a method similar to those in (Al Mutawa et al., 2012; Lone, Badroo, & Chudhary, 2015; Walnycky et al., 2015). The acquisition method required rooting the android devices so as to get access to the devices file system. Rooting the devices gives us access to otherwise inaccessible parts of the file system. After rooting the devices, a backup of the device was done to acquire a logical image of the devices memory. Both rooting and backup was achieved using KingRoot and Titanium back up respectively as specified in Table I.

3.2.3 ANALYSIS

The third stage entailed analysing and examining the backup files manually using specific tools to extract and view the files and search for data related to the test application. Titanium Backup produces a folder which contains zipped files and is saved on the memory card. The memory card in then inserted in to a card reader and connected to a PC for further analysis. The zipped files are extracted using WinRar and software required for each file type (such as DB browser for viewing SQLites .db files) is used to view and examine them.

4 RESULTS AND DISCUSSIONS

KIK MESSENGER FORENSIC INVESTIGATIONS

Kik offers its customers with numerous forms of communications and entertainment over a cellular internet connection or a local WIFI. These communications can be in the form of user-to-user chats, bot chats, and group chats where contents such as text messages, pictures, videos, sketches etc. are exchanged. Entertainments are usually in the form of games with Bots or other users, or just surfing the web from the application. Users just need to register a username and a birthdate not below 13 years to sign up on Kik.

The rest of this section describes the artefacts retrieved during the analysis of files extracted using the methods outlined in the previous section. Similar to the steps presented by Ovens & Morison (2016) and Anglano (2014) the forensic examinations are presented in the following subsections.

4.1 KIK ARTIFACTS LOCATION

The backup files generated from the three test devices are copied to the PC for further analysis. The analysis revealed that Kik stores several artefacts in different files and databases that are located into the kik.android folder located in the /data/data directory of the Android file system. The data of forensic interest generated by Kik is reported in Table II

TABLE II: KIK ARTIFACTS LOCATION

Kik Artifacts	Directory	Sub-Directory
Messages and Contacts details	data/data/kik.android/	/databases/
Exchanged Images and their metadata	##/cache/	/chatPicsBig/ and/contentpics/
Profile Pictures	/cache/	profPics
User data and preferences	/shared prefs/	KikPreferences

4.2 INVESTIGATION OF CONTACTS EVIDENCE

The value of the contact information in a forensic examination of an IM application such as Kik cannot be overemphasized as an investigator would like to know who the user (suspect or victim) has been in contact with. The information regarding contacts is contained in the KikContacts Table of the kikDatabase.db in the databases folder identified in Table II above. The twelve tables that make up the kikDatabase database is presented in Table III together with their descriptions. In this section, a detailed analysis of the contacts table, KIKContactsTable is given while the other tables are left for later sections.

4.2.1 RETRIEVING CONTACT INFORMATION

Every Kik account is connected to a set of contacts each of which is associated with its display name (i.e., the name used by the Kik user to denote the buddy), its username (that is unique to each Kik user for identification.), and a profile picture (an optional picture selected by the user, and displayed with the username and display name). The information relating to user contacts is found in two distinct tables, namely contacts and friendattribution, which were described in Table III. A record is saved for every contact in the KikContacts table the structure of which is shown in Table IV while the friend_attribution table stores a record of how each contact was added to the friends list.

TABLE III: STRUCTURE OF THE KIKDATABASE DATABASE

Table	Description
KIKContentRetainCountTable	Messages content_id and retain counts mostly in 0,1or 2
KIKContentTable	Meta data about sent or received media
KIKContentURITable	Meta data about sent or received media
KIKConversationStatusTable	Empty but might change with extensive application usage
KIKSuggestedResponseTable	Contains automatic messages suggested to the user when chatting with other users or bots and other metadata related to the chats
KIKContactsTable	Contains list of contacts and metadata related to each contacts
KikFriendAttributionTableName	List of users friends, timestamps of when when they became friends and how they became friends
Android_metadata	House keeping
chatMetaInfTable	List of friends with the order of how they have chatted with user
memberTable	List of groups user is a member of as well as the group's metadata
messagesTable	Messages exchanged, the timestamps and other related metadata

As can be observed when looking through the KIKContacts table, details of all the contacts that the user has communicated with directly and contacts without any direct communication can be found on the kikContacts table. These contacts include the Kik user's fiends on the friends list, the ones that they share same groups with, Kik users suggested by the Kik friend search feature , as well as Bots both Kik bot or other commercial bots. However the contacts that haven't been in direct contact with the user can be differentiated by the in_roster field.

Profile pictures have forensic value as well as they can be used to link an account to the actual identity of the person using it (the profile picture displayed could be a self-portrait of the user, or any image that is usually uniquely associated to the user). The profile picture of a contact with jabber id <x>_<abc>@talk.kik.com can be viewed as a JPEG file from the URL in the field photo_url. The URL is entered into a web browser and /orig.jpg or /thumb.jpg is appended to it. The timestamps found in the field photo_timestamps indicate when the contacts set the current profile picture. Thumbnails of contact's profile picture can be found in the directories listed in Table I, rows no. 4 and 5.

During a forensic examination, the need may arise to check when a contact was added to the contact list. This information is not stored in the KIKContacts table but can be deduced in the FriendAttributtion table (timestamp field).The FriendAttributtion table also provides information about how a user became friends or chat buddies either through explicit username search or through inline suggestions or from public groups.

TABLE IV: STRUCTURE OF THE KIKCONTACTS TABLE

Field Name	**Description**
_id	Primary key;Unique auto incremental identifier
Jid	Jabber ID of the contact (a string in the form of '<x>_<abc>.@talk.kik.com.', where 'x' is the username of the contact and 'abc' is a combination of the English alphabets)
display_name	Current name of user displayed
user_name	Contact's chosen unique username
in_roster	contains '1' if the contact is on the user's chat buddy list, '0' otherwise
photo_url	Contains web url of contact's profile picture if set or empty if not set
photo_timestamp	Contains epoch time of when profile picture was set
is_group	Contains '1' if the contact is a group, '0' otherwise
is_blocked	Contains '1' if the contact has been blocked, '0' otherwise
is_ignored	Contains '1' if the contact has been ignored, '0' otherwise
pending_in_roster	Contains '1' if the contact has not added the user to his/her buddies by replying or initiating a chat
is_user_admin	Contains '1' if the contact is an admin of a group, '0' otherwise
user_permission_level	Set to 'super admin'or'basic' denoting the permission level of a contact in a group
group_hashtag	Contains the displayed hashtag of a group
is_user_removed	Contains '1' if the group has been left by the user, '0' otherwise

4.2.2 DEALING WITH BLOCKED CONTACTS

In the course of a forensic investigation, it might become necessary to know if any contact has been blocked and when. The ability to block a contact is provided by Kik Messenger. When a contact has been blocked, subsequent messages will not be received by the user. However, investigations revealed that

- blocked contacts were found on the KikContacts table but were indicated to be blocked in the is_blocked field (set to '1')
- messages from blocked contacts were still delivered to the device and could be found in the messageTable table in the kikDatabase database but was not displayed to the user
- Attachments from blocked users were still present in the locations shown in Table 2, row no. 3.
- no information whatsoever could be found regarding when a contact was blocked or unblocked

4.2.3 RECOVERING DELETED CONTACTS

Deleted data either in the form of contacts or messages are artefacts of high evidentiary value. Kik messenger allows its users to delete contacts from their chat lists. When a contact is deleted, information relating to it is removed and any previous chat with that contact is removed from the chat list. The user doesn't see anything related to the deleted contact anymore. Notably, our investigations revealed that

- deleted contacts were still on the kikcontacts table with no indications of their deletions
- deleted group contacts as well as the group members were still on the kikcontacts table, however, field is_user_removed (if set to '1') indicated their deletions
- messages from deleted contacts and groups were still available on the messages tables

How long this data remains within the database is dependent upon the volume of messages the user subsequently sends and receives, as the data will eventually be overwritten with new data. This issue is discussed further in Section Dealing with Deletions.

4.2.4 UNDERSTANDING GROUP MEMBERSHIP

Kik messenger provides the facility for users to chat with each other as part of a group. The groups can have up to fifty members and are either private, where entry is by invitation, or public, open to anyone. Analysis of the kik.sqlite database found a table concerned exclusively with Kik groups; the memberTable table is populated with the groups the user is a member of along with all the other group members. Group administrators could be identified from the field: permission_level and banned could be identified in the is_banned field.

Group members were added to the KIKContacts table however, the in_roster field was set to 1 except if there was a one to one chat with the group member and the Kik user.Group administrative messages (field: sys_msg in the messagesTable table) provides group housekeeping information such as when a user joined, got blocked, left and even when the profile picture gets changed. If available at the time of investigation, this information can provide investigators with answers to 'when a user joined or left the group' from the field: timestamp of the messagesTable table.

TABLE V: STRUCTURE OF THE MEMBERTable TABLE

Field Name	Description
_id	Primary key; auto-incremental identifier
group_id	Contains group ID in the format <x>_g@groups.kik.com where x is a 13 digit combination of numbers
member_jid	Contains group member's Jabber ID
is_admin	Unknown, contains 'Null' all through
is_banned	Contains '1' is the group member has been banned and '0' otherwise
permission_level	Contains member levels in a group; 'SUPER_ADMIN', REGULAR_ADMIN, and 'BASIC'

4.3 INVESTIGATION ON CHAT HISTORY

All the exchanged messages by users in Kik Messenger are stored in the messagesTable table of the KikDatabase.db database (located in the directory listed in Table II, row 2). Examining this table allows us to recreate the sequence of events of exchanged communications, particularly: determining when a communication was instantiated, the contents of the communication, the communicating parties as well as if and when the exchanged message was received by the parties involved.

Subsequently, we give a description of the structure of the messagesTable table (Sec. The structure of the exchanged messages table), and then we give details on how to (a) reconstruct the chronology of chat messages (Sec. Reconstruction of the chat history), (b) find and retrieve the contents of a communication (Sec. Retrieving Communication Attachments), (c) determine the status of a message (Sec. Determining the state of the message), (d) determining the parties involved in the communication (Sec. Determining the communicating parties) and, in conclusion, (e) deal with deleted communications (Sec. Deleted Communications).

4.3.1 RECONSTRUCTION OF THE CHAT HISTORY

Of high forensic value is the ability to reconstruct the time in which a communication took place, the content of the communication, and the communicating parties.Anytime there is a form of communication, a record is stored in the KikDatabase database that contains the information relating to the communication such as the real body of the communication and other metadata of the communication.

TABLE VI: STRUCTURE OF THE MESSAGETABLE TABLE

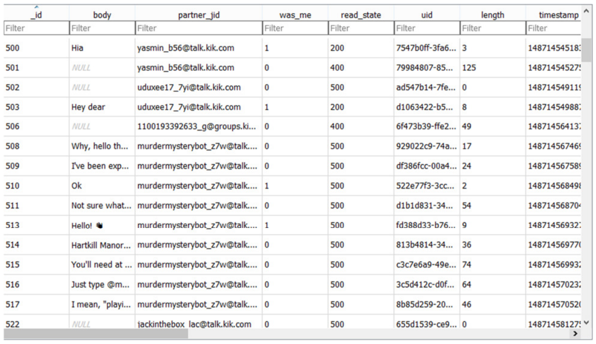
Field Name	Description
_id	Primary key, unique identifier
Body	Message content
Partner_jid	Communicating partner jabber ID
Was_me	Populated with message direction, 1 = sent or 2= received
Read_state	Delivery status, 400= unread 500= read 200= unknown 300= sent but not delivered
Uid	Unique message identifier
Length	Message length
Timestamp	Contains epoch time of when message was exchanged
Bin_id	Same as partner jabber ID
Sys_msg	Kik generated message upon joining a group or starting a group
Stat_msg	User status displayed
Content_id	Contains content ID with which the non-textual message content can be found
App_id	Contains URL or Application of content type

Friend_attr_id	Contains friend attribution ID of user communicated with
Mention_reply	Contains user_jid of contacts that replied a mention
Mentioned_contact_id	Contains user jid that was mentioned in a conversation

This is found in the messagesTable table of the KikDatabase database. Let us consider figure I, Critical examination of it shows that

- communication took place with several contacts displayed in the field : partner_jid (one-to-one and in a group),
- the user started some of the chats while others were started by other contacts(as indicated by field was_me)
- the textual content, if any of the communication is displayed in field: body (as in row 1 for one-to-one chats, row 5 for groups and row 7 for bot chats)
- the time of each communication is found in field: timestamp; looking through rows 6- 10, the user receives two messages from a chatbot at approximately Wed, 15 Feb 2017 08:01:14.692 GMT and Wed, 15 Feb 2017 08:01:15.892 GMT (rows 6 and 7). The user then sends a reply at approximately Wed, 15 Feb 2017 08:01:24.989 GMT (row 8)

We furthermore observed that each message has its distinct identifier as indicated in the uid field.



_id	body	partner_jid	was_me	read_state	uid	length	timestamp
Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
500	Hia	yasmin_b56@talk.kik.com	1	200	7547b0ff-3fa6...	3	148714545183
501	NULL	yasmin_b56@talk.kik.com	0	400	79984807-85...	125	148714545275
502	NULL	uduxee17_7yi@talk.kik.com	0	500	ad547b14-7fe...	0	148714549119
503	Hey dear	uduxee17_7yi@talk.kik.com	1	200	d1063422-b5...	8	148714549887
506	NULL	1100193392633_g@groups.ki...	0	400	6f473b39-ffe2...	49	148714564137
508	Why, hello th...	murdermysterybot_z7w@talk....	0	500	929022c9-74a...	17	148714567469
509	I've been exp...	murdermysterybot_z7w@talk....	0	500	df386fcc-00a4...	24	148714567589
510	Ok	murdermysterybot_z7w@talk....	1	500	522e77f3-3cc...	2	148714568498
511	Not sure what...	murdermysterybot_z7w@talk....	0	500	d1b1d831-34...	54	148714568704
513	Hello! 👋	murdermysterybot_z7w@talk....	1	500	fd388d33-b76...	9	148714569327
514	Hartkill Manor...	murdermysterybot_z7w@talk....	0	500	813b4814-34...	36	148714569770
515	You'll need at ...	murdermysterybot_z7w@talk....	0	500	c3c7e6a9-49e...	74	148714569932
516	Just type @m...	murdermysterybot_z7w@talk....	0	500	3c5d412c-d0f...	64	148714570232
517	I mean, "playi...	murdermysterybot_z7w@talk....	0	500	8b85d259-20...	46	148714570520
522	NULL	jackinthebox_lac@talk.kik.com	0	500	655d1539-ce9...	0	148714581275

Figure I: screenshot of the messagesTable table

Interestingly, the ChatMetaInfTable table (Table III, row 9) can also be used to recreate the chat history as it gives the order in which a user chatted with other contacts. However, deleted contacts could not be found in this table.

4.3.2 RETRIEVING COMMUNICATION ATTACHMENTS

Aside from messages that have textual content, Kik messenger allows it users to exchange messages with other type of content such as images, sketches, video files and web contents. Information relating to non-textual message content is spread across tables messages Table, KIKcontentTable and KIKcontentURITable as well as the directories listed in table 4.1 row 5.

The messages that have non-textual contents are identified from a unique ID in the content_id field in the messagesTable table of the KIKDatabase. This ID is then used to locate and view the content either from the folder specified above or from the content_Table table. However, the contents whose content_id were located in this folder were only found in small sizes but could not be viewed clearly while the ones located on the contentTable and content_uriTable pointed to URLs on the Kik servers. Deleted contents could still be located and viewed without any indication that it has been deleted by the user.

4.4 DEALING WITH DELETIONS

Lastly, an attempt to recover artefacts deleted by the Kik user was made. Deletions are usually of the highest forensic value since the suspect or even the victim might attempt to evade detection by deleting some artefacts. In Kik messenger, when a message (textual or attachment) is deleted from the application, the user does not see it again. However from our investigations, all the deleted chats and exchanged files could still be retrieved from each of the corresponding tables of the KikDatabase. This is however subject to the amount of time elapsed since the deletions as stated in the Kik website.

It can therefore be concluded that the recovery of deleted messages and message attachments is subject to the amount of time elapsed from the time of deletion and the time of investigation.

5 CONCLUSION

A forensic analysis of the artefacts left by Kik, a popular IM application on Android smartphones is presented in this paper. The usage of Kik messenger on android phones leaves behind artefacts of forensic value. How these artefacts can be examined and interpreted to get information such as contacts list, chat history and exchanged media has been described. Whom the user has chatted with?, when the chat occurred as well as what was said and exchanged are information with high value during a forensic investigation.

The results of this study holds double value in that on one hand, it provides forensic investigators with the steps in examining and interpreting Kik messenger artefacts on Android devices while on the other hand it provides a benchmark against which future Kik forensic tools can measured.

The results of this study apply only to Android devices; in fact results from the study in (Ovens& Morison, 2016) shows that Kik messenger forensics on iOS differ in terms of how the artefacts are stored. The forensic analysis of Kik messenger on other platforms as well as android versions is left for future research.

REFERENCES

Al Mutawa, N., Baggili, I., & Marrington, A. (2012). Forensic analysis of social networking applications on mobile devices. Digital Investigation, 9(SUPPL.), S24–S33. https://doi.org/10.1016/j.diin.2012.05.007

Anglano, C. (2014). Forensic analysis of WhatsApp Messenger on Android smartphones. Digital Investigation, 11(3), 1–13. https://doi.org/10.1016/j.diin.2014.04.003

Anglano, C., Canonico, M., & Guazzone, M. (2016). Forensic analysis of the ChatSecure instant messaging application on android smartphones. Digital Investigation, 19, 44–59. https://doi.org/10.1016/j.diin.2016.10.001

Iqbal, A., Alobaidli, H., Almarzooqi, A., & Jones, A. (2015). LINE IM app Forensic Analysis. 12th International Conference on High-Capacity Optical Networks and Enabling/Emerging Technologies (HONET-ICT 2015) Poster, (IM). https://doi.org/10.13140/RG.2.1.2237.7042

Iqbal, A., Marrington, A., & Baggili, I. (2014). Forensic artefacts of the ChatON Instant Messaging application. Int. Workshop Syst. Approaches Digit. Forensics Eng., SADFE. https://doi.org/10.1109/SADFE.2013.6911538

Lone, A. H., Badroo, F. A., & Chudhary, K. R. (2015). Implementation of Forensic Analysis Procedures for WhatsApp and Viber Android Applications, 128(12), 26–33.

Mahajan, A., Dahiya, M., & Sanghvi, H. (2013). Forensic Analysis of Instant Messenger Applications on Android Devices. International Journal of Computer Applications, 68(8), 38–44. https://doi.org/10.5120/11602-6965

Ovens, K. M., & Morison, G. (2016). Forensic analysis of Kik messenger on iOS devices. Digital Investigation, 17, 40–52. https://doi.org/10.1016/j.diin.2016.04.001

Satrya, G. B., Daely, P. T., & Shin, S. Y. (2016). Android forensics analysis: Private chat on social messenger. International Conference on Ubiquitous and Future Networks, ICUFN, 2016–Augus, 430–435. https://doi.org/10.1109/ICUFN.2016.7537064

Sgaras, C., & Kechadi, M. (2014). Forensic Acquisition and Analysis of Tango VoIP.

Walnycky, D., Baggili, I., Marrington, A., Moore, J., & Breitinger, F. (2015). Network and device forensic analysis of Android social-messaging applications. Digital Investigation, 14(S1), S77–S84. https://doi.org/10.1016/j.diin.2015.05.009