Digital Forensics Study of a Cloud Storage Client: A Dropbox Artifact Analysis

Gandeva Bayu Satrya
School of Applied Science, Telkom University
Bandung 40257, Indonesia
Email: gbs@telkomuniversity.ac.id

***Abstract*—The rapid development of cloud storage technology paired with the prevalence of smartphone usage presents wide-ranging challenges for digital forensics practitioners. Data are more easily uploaded and shared between multiple devices and across multiple platforms. So, the process has increased the opportunities for criminality. Criminality undertaken in cloud computing can be directly seen on logs stored on the cloud storage server, which records user activity. However, because of user privacy protection, these logs cannot be easily used as evidence in court. This issue emphasizes the need for a reliable means of identifying, acquiring, and preserving evidential data from the client-side. This study identifies the data artifacts of a user accessing Dropbox via smartphone (Android Lollipop and Android Nougat). The data are from performing several common activities such as installing, signing up, uploading, downloading, sharing, and others. About 14 artifacts are identified by documenting the Dropbox client database changing contents as these activities are carried out. This study increases knowledge of the artifacts that are leftover by Dropbox client on Android smartphones. The results propose this comparing and analyzing method can be used by digital forensics investigators in carrying out investigations and cyberlaw practitioners as guidance in criminal cases.**

***Index Terms*—Cloud Storage, Digital Forensics, Mobile Forensics, Dropbox Analysis, Artifacts, Android-Based**

I. Introduction

CLOUD storage services are included in the Infrastructure as a Service (IaaS) model of cloud computing [1, 2]. Cloud storage services provide storage for users. It can be used to store files such as documents, pictures, music, and video. Those files can be accessed via a web browser and cloud storage client application on PCs, smartphones, and tablets [3, 4]. There are several cloud storage service providers. Some of them offer a certain capacity of storage for free, such as Dropbox™, Microsoft One Drive®, Box©, and Google Drive™ [5, 6].

Received: July 12, 2019; received in revised form: Aug. 21, 2019; accepted: Sep. 17, 2019; available online: Oct. 07, 2019.

Although cloud storage is not universally used, for many people, it has become invaluable. For example, if a researcher or engineer forgets their flash drive, they can still access their data anywhere and anytime as long as there is an Internet connection. It can be said that cloud storage is today's digital storage technology. However, the development of technology has indirectly facilitated the increase of cybercrime that may affect individual, corporate, and government data. Any illegal act involving computers, systems, or their applications is a cybercrime [7–9]. One method for finding traces and files left from cybercrime is digital forensics. This study focuses on mobile forensics.

Digital mobile forensics processes are conducted in accordance with the procedures mentioned by Ref. [10]. Those are to find, collect, validate, identify, analyze, and document the digital evidence from digital devices to reconstruct criminal events [7, 10]. Digital forensics science dictates that there should not be any changes made at all to the original evidence. However, in reality, this is not possible. In performing forensic analysis, some 'scrapes' and 'smudges' on the original evidence cannot be avoided. To proceed in a forensically manner, all 'scrapes' and 'smudges' on the original evidence must be explained. Mobile forensics can be performed on Windows Phone, Apple iOS, Blackberry, and Android. This study focuses on the mobile forensics of Dropbox cloud storage client on Android smartphones.

A brief illustration of the procedure conducted is as follows. For mobile forensics, there is a prerequisite that the Android smartphones used have been through a rooting process. This is the first stage performed on the smartphone in this study. Next, the installation of Android Debug Bridge (ADB) and Busybox is carried out. The acquisition process of smartphones follows it. Further details regarding the installation of both programs along with the integrity of the files that would be acquired will be described in the next section. This study only focuses on the client-side of

the Dropbox application [11]. The requirements are that the smartphones are in good condition, unlocked, and encrypted files found are ignored. The purpose of this research is to show the behavior of cybercrime by using Dropbox cloud storage client as storage and deployment media. There are two major stages in this research. The first stage is analyzing the artifacts using OPPO A37 smartphone (Android Lollipop). In the second stage, the steps performed in the Oppo A37 are repeated using a Samsung A7 smartphone (Android Nougat). By ensuring that the directories created during user activities are identical on both smartphones, the results are valid across different OS and vendors. The method of documenting the changes in Dropbox client database throughout a series of user activities assists in finding digital forensics artifacts. In the future, this method can be used by digital forensics investigators and cyberlaw practitioners in carrying out investigations.

Many studies have proposed diverse approaches for cloud storage forensics. Reference [12] surveyed forensic challenges in cloud computing and analyzed recent solutions and developments. Then, Ref. [13] contributed by studying MEGA cloud client app in Android and iOS platforms. Reference [14] presented an in-depth understanding of the types of terrestrial artifacts. Those were likely to remain after the use of cooperative storage cloud on Symform client devices. Similarly, Ref. [15] introduced the concept of cloud-native digital artifacts with Google Docs as the case study. Next, Ref. [16] provided a container-based software framework named as SCAlable Realtime Forensics (SCARF). It could be applied to achieve high-performance digital forensics.

A similar study by Ref. [17] explained the process of getting the remnant data/log from Dropbox applications on a PC, with Windows 7 operating system, and on the iPhone 3G. The researchers checked the MD5 hash value during the acquisition process to ensure the integrity of the files for digital mobile forensics. The difference with this study is that the mobile forensics analysis is conducted on two Android devices with different vendors and different types of OS. Along with the latest developments, this research uses SHA-256 to check the integrity of the files during the acquisition process. Therefore, it can be accepted as valid evidence [18, 19]. To collect a logical image of the Android device, the researcher applies ADB tool.

For example, user A (a man suspected of cybercrime) has distributed images of user B (the victim) to extort money from him. In the oral investigation process, user A denies distributing the images via Dropbox. Based on that story, the investigators should look for digital artifacts that can be used as valid evidence showing user A as the perpetrator. Investigators can search for many possible areas. However, finding the required artifacts to use for evidence is both challenging and time-consuming. In this case, the methods used in this research can be applied.

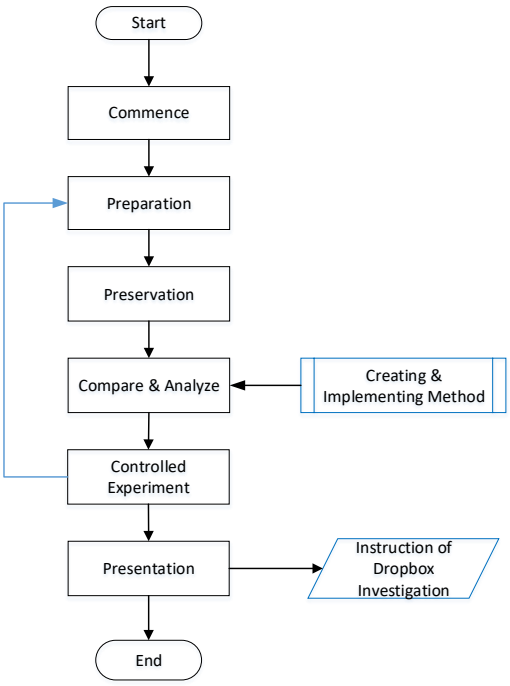

Fig. 1. The flow of this study on the digital forensics.

II. Research Method

In conducting this study, sound mobile forensics procedures are used. It can be seen in Fig. 1. It was developed by Refs. [10, 20]. The stages are as follows. In the commencement stage, the focus of this research is to determine the artifacts of normal user activities performed by a user on a cloud storage client. The cloud storage client used in this study is Dropbox application installed on an Android-based smartphone [11, 17]. The artifacts can include username, password, files that have been modified for each activity, time and date of the activity, and others. It can also be artifacts that have been left behind when an application is uninstalled.

For the preparation stage, the hardware and software tools needed for the investigation are identified. The required software components are Dropbox client software version 150.2.4, VRoot version 1.7.3, Android Debug Bridge (ADB) v1.0.31, SQLite version 3.8.11, SQLiteBrowser version 3.7.0, Busybox Pro v27, and ES File Explorer File Manager v3.2.3.5. Meanwhile,

the required hardware components are OPPO A37 smartphone (Android Lollipop), Samsung A7 smartphone (Android Nougat), and PC (Intel Core i7-3770 3.40 GHz, 4GB RAM, and 1TB HDD).

The next stage is to preserve the data that would be analyzed [21, 22]. The rooting is done. The smartphone is connected to the PC, and a rooting procedure is performed. Rooting is a process that allows users to obtain the highest control privilege [9]. Rooting is important because there are folders and specific data that can only be accessed when the smartphone has been rooted. It may seem impossible to ensure that the original data are not changed. However, by using scientific methods that are acceptable in court and international law, it can be done [10, 20, 23].

Next, the software tools described previously are installed. It is important to clarify that the ADB and Busybox installation process on a smartphone does not affect the later acquisition of the artifacts. These open-source forensics tools have been adopted by Refs. [18, 19]. In digital forensics, the final step of the preservation phase is to create the duplicates of the original data. As previously stated, the law requires that analysis cannot be performed on the original data. In order to simulate a real crime scene, a sequence of activities is performed on Dropbox.

The activities performed are Dropbox installation, signing up to Dropbox, logging out of Dropbox, logging in to Dropbox, uploading files to Dropbox, downloading files from Dropbox, opening the file, creating a new folder in Dropbox, creating a new file, moving the file, renaming the file, sharing files, deleting files, logging out of Dropbox, and uninstalling Dropbox. It is necessary to make a backup of the entire phone's contents after each activity has been performed. It is also necessary to take a hash value (SHA-256) for the relevant database after each activity is done. For example, after the download activity, the backup data is obtained by using Algorithm 1. The hash values of the download databases are also taken for later analysis.

Then, for comparing and analyzing stage, the artifacts created for each user activity are identified by using a comparison method. The Algorithm 2 is created to determine the changes in the hash values of database files (Fig. 2). The hash values from the user activity are compared with the hash values after the next activity has been carried out. If hash values have changed, it means that new artifact files have been created inside of those databases. The advantage of using this algorithm is that it saves time in the investigation process. It means there is no need to go through each database one by one to look for artifacts. Having identified the locations of the artifacts, the directories are opened, and the file names are recorded.

Last, in the controlled experiment stage, it is to ensure that these results apply to different vendors (using Android OS), and the artifacts created on the Oppo A37 smartphone will be the same as Samsung A7 smartphone. The Dropbox user activities are carried out, and it is checked that the same directories and databases are created.

Algorithm 1: Data acquisition in Dropbox

```
fileA ← find all files in the phone contain name 'dropbox'
store fileA in database with table name acqfileA
counter ← count number of row in acqfileA;
for i ← 1 to counter do
    acqdb ← data in database where field name 'id' in
      acqfileA table equal with 'i'
    pull data acqdb from smartphone;
end
```

III. Results and Discussion

In this study, the testing and analysis of the Dropbox application are conducted on Oppo A37 and Samsung A7 Android smartphones. It involves 14 user activities. The following analysis shows the artifacts that are created during each user activity. These are identified by using a comparison method (see Algorithm 2).

A. Analysis of Installation Data

When Dropbox is installed, it creates some new files. As listed in Table I, there are five files formed after the Dropbox application is installed. One of the files that should be considered is `data/app/com.Dropbox.Android-1.apk`. It is `.apk` file from Dropbox that is automatically saved on the device during installation. If these five files exist on a smartphone, it means that the user has already installed the Dropbox application.

B. Analysis of Signup Data

When the signup activity is performed, several new files are created by the Dropbox application. However, from all files that are created, there are only two main files that can be used as the reference in this step. First, `data/data/com.dropbox.Android/databases/prefs.db` contains user signup information, as shown in Fig. 3. Another file that needs to be considered is `data/data/com.dropbox.Android/databases/ID-db.db`. It contains a list of files stored in the cloud. If an investigator can find both of those files and analyze them, the next stage of the investigation can be carried out easily.

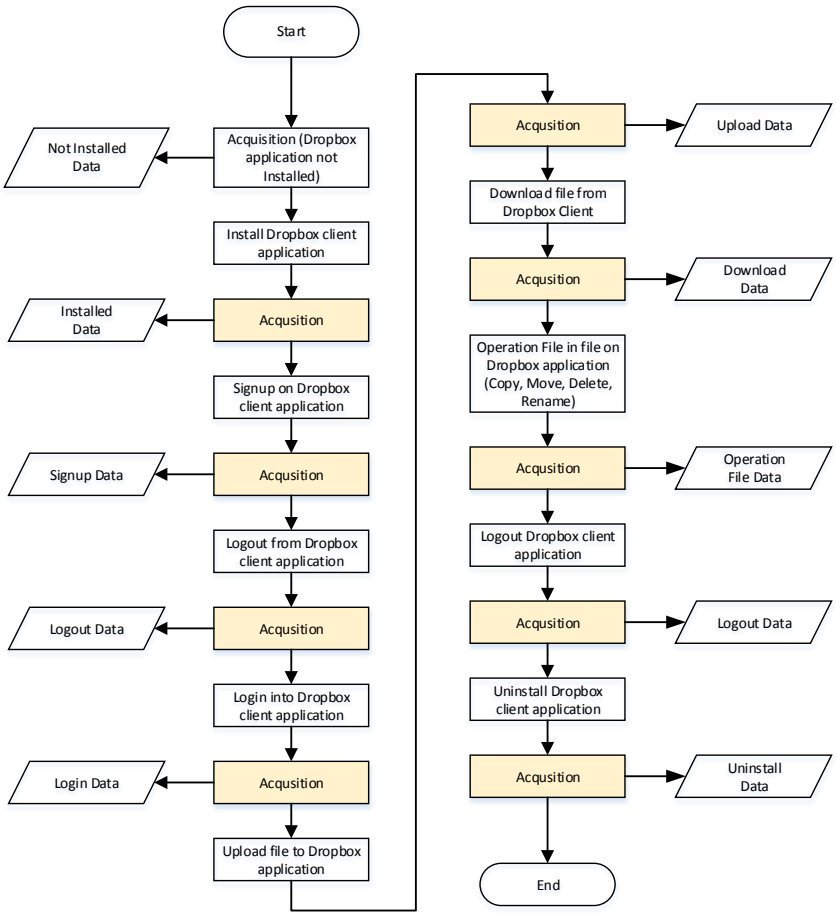

Fig. 2. Acquisition process in Dropbox.

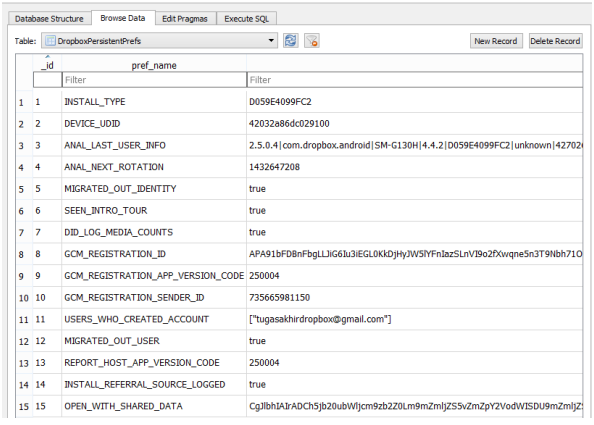

Fig. 3. Signup activity in Dropbox from browser’s database.

C. Analysis of Login Data

When the login activity is undertaken on the Dropbox application, it forms some new files listing the user’s files in the cloud. When the login process is complete, `<stringname=accounts>[]</string>` is filled with codes associated with user authentication. For example, `>[{\&quot;userToken\&quot;:\&quot;7t3t8tidn8zci9f1|7rqnawvf4uymat8\&quot;,\&quot;isLinked\&quot;:true,\&quot;userId\&quot;:\&quot;427026848\&quot;}]</` in `data/data/com.dropbox.android/shared_prefs/dropbox-credentials.xml` is added as shown in Fig. 4. The user ID, in this case user with ID 427026848 in table of `DropboxPersistentPrefs` on database of `data/data/com.dropbox.android/databases/prefs.db` is also added.

Algorithm 2: Comparison and analysis processes in Dropbox

```
pathA ← acquisition data directory
pathB ← acquisition data directory
fileX ← listing all files from pathA
fileY ← listing all files from pathB
store fileX in database with table name cmpfileX
store fileY in database with table name cmpfileY
Full outer join table cmpfileX with cmpfileY and store it to table cmpfileXY
counter ← count row from joining database
for i ← 1 to counter do
    cmpX ← data in field name fileX where field name 'id' on cmpfileXY table equal with 'i'
    cmpY ← data in field name fileY where field name 'id' on cmpfileXY table equal with 'i'
    sha256sum_X ← file on cmpX path SHA-2 checksum
    sha256sum_Y ← file on cmpY path SHA-2 checksum
    if cmpX is equal with 'NULL' and cmpY is not equal with 'NULL' then
        set table field name newfile to 'YES' where field name 'id' equal with 'i'
    end
    if cmpY is equal with 'NULL' and cmpX is not equal with 'NULL' then
        not equal table field name deletefile to 'YES' where field name 'id' equal with 'i'
    end
    if sha256sum_X is equal with sha256sum_Y then
        not equal table field name modifiedpart to 'Nothing Changes' where field name 'id' equal with 'i'
    end
    if sha256sum_X is not equal with sha256sum_Y then
        compare sha256sum_X and sha256sum_Y
    end
end
```

TABLE I
THE DETAILS OF DROPBOX INSTALLATION ACTIVITY.

No	Path
1	data/dalvik-cache/data@app@com.dropbox.android-1.apk@classes.dex
2	data/system/dropbox/event_data@1432611487514.txt
3	data/system/dropbox/SYSTEM_RESTART@1432611461699.txt
4	data/app/com.dropbox.android-1.apk
5	data/app-lib/com.dropbox.android-1/libDummyArchIndicator.so

TABLE II
ANALYSIS OF DROPBOX LOGOUT DATA.

No	File Name
1	data/data/com.dropbox.android/databases/ID-db.db-journal
2	data/data/com.dropbox.android/databases/ID-db.db
3	data/data/com.dropbox.android/databases/ID-prefs.db-journal
4	data/data/com.dropbox.android/databases/ID-prefs.db

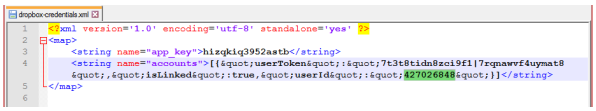

Fig. 4. Login activity in Dropbox from browser's database.

Both of these files can be used to identify which user is active and what files are contained in the application, even if there are some encrypted words. The most important part of this step is the user ID. It can be used as a reference in the next steps. It will always be the same if there is only one user logging in.

D. Analysis of Logout Data

When the logout activity is done in the Dropbox application, it also creates some files. One of the files is `data/data/com.dropbox.Android/databases/global.db-journal`. In addition, some files are deleted when the user logs out. Generally, the deleted files are files related to the list of files owned by the user in the cloud. Those are listed in Table II.

When the logout process is done, `>{\&quot;userToken\&quot;:\&quot;snwor0960gdbzlmd|d3mciuoaxxha1p3\&quot;,\&quot;isLinked\&quot;:true,\&quot;userId\&quot;:\&quot;427026848\&quot;}]</` in `data/data/com.dropbox.android/shared_prefs/dropbox-credentials.xml` is removed. User with ID 427026848 in database of `data/data/com.dropbox.android/databases/prefs.db` is deleted when the user logs out. This is the same for the login process. Both files, as explained before, can be used as references that the user has been logged out if their contents are empty or deleted. Furthermore, login and logout activities are saved with time stamps in the UNIX

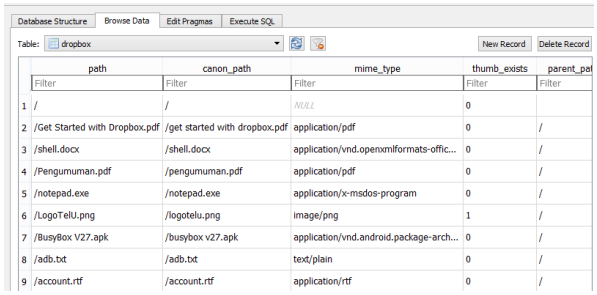

Fig. 5. Upload activity in Dropbox from browser's database.

TABLE III
THE CHANGE CAUSED BY DATA UPLOAD.

No	File Name
1	data/media/0/Android/data/com.dropbox.android/cache/u427026848/thumbs/LogoTelU.png/large.png
2	mnt/shell/emulated/0/Android/data/com.dropbox.android/cache/u427026848/thumbs/LogoTelU.png/large.png

format.

E. Analysis of Uploading Data

Some files are modified when the upload activity is performed. One of the modified files is `data/data/com.dropbox.android/databases/ID-db.db`. Database of `data/data/com.dropbox.android/databases/ID-db.db` adds data related to the files uploaded to the `dropbox` table as shown in Fig. 5. The time stamp uses the UNIX time format. Therefore, from the table, it can determine when the file is uploaded by looking at `modified_milis` in `dropbox` table. When the uploading is done in Dropbox, some new files are created. One example of uploading is an image named "`LogoTelU.png`" in Dropbox. The changes in the Dropbox client database are listed in Table III.

These acquisitions and analyses can assist a digital forensics investigation in determining what files linked to a cybercrime case that have been uploaded.

F. Analysis of Downloading Data

When the download activity from Dropbox is carried out, it forms some new files in `data/media/0/Android/data/com.dropbox.android/files/UID/scratch/` and `mnt/shell/emulated/0/Android/data/com.dropbox.android/files/uID/scratch/`. The new files are downloaded from Dropbox. This scenario can help the investigation

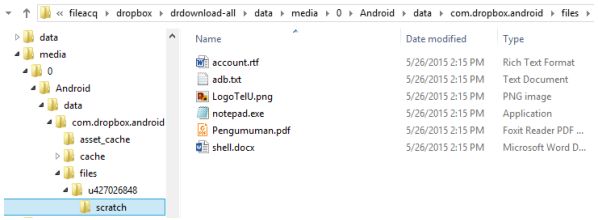

Fig. 6. Download activity in Dropbox from browser's database.

process. For example, User A uploads a file to Dropbox cloud storage while User B downloads the file from a link previously created by User A. Using the methods mentioned, the file can be found in the scratch folder as shown in Fig. 6.

G. Analysis of File Operation Data (Open)

When a file is opened in the Dropbox application, some new files are formed in directory of `data/media/0/Android/data/com.dropbox.android/cache/UID/docpreviews/`. This file is a cache file so that files that have been uploaded to Dropbox can be opened and viewed directly. Text files such as `.txt`, `.rtf`, `.doc` are converted to `.pdf` before they can be displayed. The modified file when a file is opened is `data/data/com.dropbox.android/databases/ID-db.db`. The table of `preview_cache` is filled with information about files that have been opened using the Dropbox application. Field access time uses the UNIX timestamp format. To obtain the actual time, it must be converted into a time format that is commonly used.

H. Analysis of File Operation Data (New Folder)

When a new folder is created, database of `data/data/com.dropbox.android/databases/ID-db.db` is modified. Folder data are added to `dropbox` table for that database as shown in Fig. 7. Using this method, the search for the new folders created by the user can be accelerated. The positions of new files in the Dropbox database table are usually at the bottom of `data/data/com.dropbox.android/databases/ID-db.db`

I. Analysis of File Operation Data (New File)

When a new file is created, the database of `data/data/com.dropbox.android/databases/ID-db.db` is modified, and data associated with the file created earlier are added. An example of an activity carried out at this stage is the creation of a new file in Dropbox called

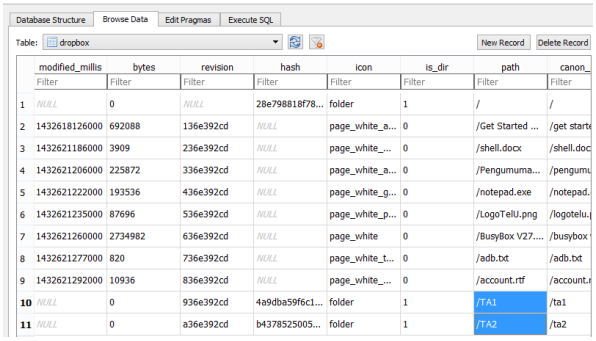

Fig. 7. New folder activity in Dropbox from browser's database.

TABLE IV
THE EXAMPLE OF NEW FILE CREATION ACTIVITY.

No	File Name
1	data/media/0/Android/data/com.dropbox.android/files/u427026848/scratch/Foresty.txt
2	mnt/shell/emulated/0/Android/data/com.dropbox.android/files/u427026848/scratch/Foresty.txt

TABLE V
THE EXAMPLE OF NEW FILE CREATION ACTIVITY.

No	File Name
1	data/media/0/Android/data/com.dropbox.android/cache/u427026848/thumbs/TA2/LogoTelU.png/large.png
2	mnt/shell/emulated/0/Android/data/com.dropbox.android/cache/u427026848/thumbs/TA2/LogoTelU.png/large.png

"foresty.txt". When the new file is created, some additional files are created. Those are listed in Table IV.

J. Analysis of File Operation Data (Move)

When files are moved, file in data/data/com.dropbox.android/databases/ID-db.db is modified. In the Dropbox table on database of data/data/com.dropbox.android/databases/ID-db.db, the parent path field changes according to the location where the files are moved. An example of movement activity is done by moving a file to a different folder. File in /TA1/LogoTelU.png is moved to folder /TA2. When this activity is done, some new files are created, as listed in Table V.

These newly created files are thumbnails of LogoTelU.png. Those are in the new directory position in folder of TA2.

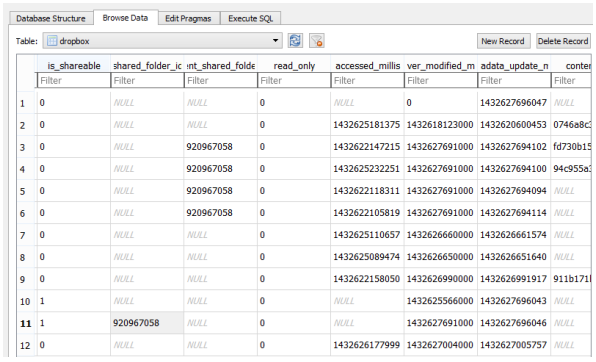

Fig. 8. Sharing activity in Dropbox from browser's database.

K. Analysis of File Operation Data (Rename)

When a file is renamed, several fields in the Dropbox table in data/data/com.dropbox.android/databases/ID-db.db are changed accordingly. An example of a renaming activity is done by changing the name of LogoTelU.png to TelU.png. Similarly, some fields in the Dropbox table in database of data/data/com.dropbox.android/databases/427026848-db.db are also changed.

L. Analysis of File Operation Data (Share)

When a sharing activity is done, fields of shared_folder_id and ent_shared_folder in dropbox table in data/data/com.dropbox.android/databases/ID-db.db are updated. It is shown in Fig. 8. An example of a sharing activity is performed by sharing links of files. Then, the links can be accessed by another user. Some changes occur in the database in dropbox table for the user that gives access. It is expected that in an investigation process, the search for users with whom a file has been shared can be found quickly.

M. Analysis of File Operation Data (Delete)

When a file is deleted, data/data/com.dropbox.android/databases/ID-db.db is also updated. Data related to the deleted files is also deleted. The example is done by deleting a file from a certain directory. Information changes in the Dropbox database table become NULL.

If ID-db.db database is opened using SQLite, the files that have been deleted are not be visible. In here, the researcher uses an additional tool which is hex editor in order to look deeper. By using Hex Editor, files that have been deleted can be found. Files that have been deleted before still exist inside the cloud server where user_id is located. Evidence

TABLE VI
LOG OF DELETING ACTIVITY.

Logcat timestamp
05-2815:35:12.0527341247VWindowManager: ChangingfocusfromWindow{42c23530u0com. dropbox.android/com.dropbox.android.activity. DbxMainActivityEXITING}toWindow{42346840u0com. dropbox.android/com.dropbox.android.activity. DbxMainActivity}Callers=com.android.server.wm. WindowManagerService.removeWindowLocked:2857 com.android.server.wm.WindowManagerService. removeWindow:2796com.android.server.wm. Session.remove:182android.view.IWindowSession\ $Stub.onTransact:197

of the deleted files is located on dropbox web-based application, although the web-based version is a little different from the android-based application. In https: //dropbox.com (after signing in) on the top toolbar, there is "show deleted files" button. As long as the user does not delete the files permanently, investigators can still restore those deleted files.

The next analysis uses an online forensics approach called live forensics. By using ADB tools, it shows how the delete process takes place. During testing, two threads appear (`com. android.server.wm.Session.remove:182` and `\_cache\_remove\_p:`). It causes the database to show NULL. With ADB tools, the date and time of deleting are obtained. Log of live forensics can also be used as digital evidence in cyberlaw cases. Results from the log of live forensics are in Table VI.

N. Analysis of Uninstallation Data

When the Dropbox application is uninstalled, the remaining data relating to Dropbox are log data. It states when the use of the Dropbox application starts and ends. This data can be accessed on `data/system/ Dropbox`. This data help to prove that the user (in this case, the suspect) has installed or uninstalled the Dropbox. Table VII shows the remaining files after the uninstallation. The exact data cannot be determined, as they will be different for every account. However, the results will not be far from these thirteen files.

This analysis is strengthened if it is combined with Android live forensics. An absolute requirement in doing Android live forensics is that the victim's smartphone (in this case) should always be on (have not been restarted or shut down). With `adblogcat` commands, all activities in the smartphone will be retrieved. However, the focus on this step is the uninstallation. There are three pieces of remnant data along with the time of occurrence. It can be used as references for that particular smartphone after the user has uninstalled Dropbox as shown in Table VIII. These

TABLE VII
EXAMPLE OF DROPBOX UNINSTALLATION ACTIVITY.

No	File Name
1	data/system/dropbox/event_data@ 1432630110161.txt
2	data/system/dropbox/event_data@ 1432628284995.txt
3	data/system/dropbox/event_data@ 1432626457348.txt
4	data/system/dropbox/event_data@ 1432624656019.txt
5	data/system/dropbox/event_data@ 1432622855222.txt
6	data/system/dropbox/event_data@ 1432620944931.txt
7	data/system/dropbox/event_data@ 1432619009741.txt
8	data/system/dropbox/event_data@ 1432617094399.txt
9	data/system/dropbox/exp_det_cert_pin\ _failure@1432615212489.txt
10	data/system/dropbox/event_data@ 1432615124845.txt
11	data/system/dropbox/event_data@ 1432613288347.txt
12	data/system/dropbox/event_data@ 1432611487514.txt
13	data/system/dropbox/SYSTEM_RESTART@ 1432611461699.txt

TABLE VIII
LOG OF UNINSTALLATION ACTIVITY.

Logcat timestamp
05-2514:05:26.4437531198IActivityManager: STARTu0{act=android.intent.action. DELETEdat=package:com.dropbox. androidcmp=com.android.packageinstaller/ .UninstallerActivity}frompid2952 05-2514:05:30.040753773IActivityManager: Forcestoppingcom.dropbox.androidappid= 10150user=-1:uninstallpkg 05-2514:05:30.058753773IActivityManager: Killing14283:com.dropbox.android/ u0a150(adj15):stopcom.dropbox.android

are `act=android.intent.action.DELETE`, `Forcestoppingcom.dropbox.android`, and `Killing14283:com.dropbox.android/ u0a150`. The results of the analysis are shown in Table IX.

To test whether the artifacts obtained are correct or not, the researcher checks the actual activities performed by the user. This is tested by applying Reversal Burden of Proof principles to the obtained results. By applying the terms of the law (Reversal Burden of Proof) [24], it should be ensured that the first test object (Oppo A37 smartphone) is not imposed with the burden of proof. The reversal burden of proof imposed on the control object (Samsung A7 smartphone) is achieved by running the same steps. Based on the analysis above, a testing table is established. It contains a list of questions for the reversal burden of proof. This means that a relationship can be proven between

TABLE IX
THE RESULTS OF DROPBOX ANALYSIS.

No	Activity	Path	Information
1	Install data	`data/app/com.Dropbox.android-1.apk`	-
2	Signup Data	`data/data/com.Dropbox.android/databases/prefs.db`	Username used for login.
3	Logout Data	`data/data/com.Dropbox.android/databases/prefs.db`	Date and time information using ADB logcat.
4	Login Data	`data/data/com.Dropbox.android/databases/prefs.db`	Username used for login.
5	Uploading Data	`data/data/com.Dropbox.android/databases/ID-db.db`	List of files uploaded by the user with information regarding the uploading date.
6	Downloading Data	`data/media/0/Android/data/com.Dropbox.android/files/uID/scratch/`	Files that have been downloaded by the user. This file can be accessed directly.
7	Operation File Data (Open)	`data/media/0/Android/data/com.Dropbox.android/cache/uID/docpreviews/`	Files that have been previewed by the user. This file can be accessed directly.
8	Operation File Data (New Folder)	`data/data/com.Dropbox.android/databases/ID-db.db`	List of files uploaded by the user with information of date about the file modification.
9	Operation File Data (New File)	`data/data/com.Dropbox.android/databases/ID-db.db`	List of files uploaded by the user with information of date about the file modification.
10	Operation File Data (Move)	`data/data/com.Dropbox.android/databases/ID-db.db`	List of files uploaded by the user with information of date about the file modification.
11	Operation File Data (Rename)	`data/data/com.Dropbox.android/databases/ID-db.db`	List of files uploaded by the user with information of date about the file modification.
12	Operation File Data (Share)	`data/data/com.Dropbox.android/databases/ID-db.db`	List of files uploaded by the user with information of date about the file modification.
13	Operation File Data (Delete)	`data/data/com.Dropbox.android/databases/ID-db.db`	Using Hex Editor and ADB logcat to find the information about deleted files.
14	Uninstall Data	`data/system/Dropbox`	Date and time information using ADB logcat.

the related files/paths, the analysis results obtained previously, and user activities performed. The same analysis results are collated and included in a list of questions on the test table. From the test results of the two experimentation objects, instructions can be made for investigators to find artifacts with Dropbox as the case study.

IV. CONCLUSION

Based on the method proposed and 14 scenarios described, the researcher can conclude several things. First, the artifacts from user activities in Dropbox on Android smartphones (with Lollipop and Nougat OS) can easily be found by comparing directories and databases created from those activities. Second, digital forensics practitioners, researchers, and investigators can use the results of this study to identify artifacts in criminal cases and as guidance in carrying out investigations.

Further forensics analysis still needs to be done with other Android OS updates such as Android Oreo, Android Pie, and others. Different versions of Dropbox application and different cloud storage applications can be tested. Future researchers can also investigate file recovery mechanisms after a user has deleted all data.

REFERENCES

[1] W. F. Hsien, C. C. Yang, and M. S. Hwang, "A survey of public auditing for secure data storage in cloud computing," *IJ Network Security*, vol. 18, no. 1, pp. 133–142, 2016.

[2] M. A. Khan, "A survey of security issues for cloud computing," *Journal of Network and Computer Applications*, vol. 71, pp. 11–29, 2016.

[3] H. Chung, J. Park, S. Lee, and C. Kang, "Digital forensic investigation of cloud storage services," *Digital Investigation*, vol. 9, no. 2, pp. 81–95, 2012.

[4] D. C. Chou, "Cloud computing risk and audit issues," *Computer Standards & Interfaces*, vol. 42, pp. 137–142, 2015.

[5] M. Muchmore. The best cloud storage and file-sharing services for 2019. [Online]. Available: https://bit.ly/2XUlAfq

[6] S. Mitroff. OneDrive, Dropbox, Google Drive and Box: Which cloud storage service is right for you? [Online]. Available: https://cnet.co/2emAPbD

[7] L. Caviglione, S. Wendzel, and W. Mazurczyk, "The future of digital forensics: Challenges and the road ahead," *IEEE Security Privacy*, vol. 15, no. 6, pp. 12–17, 2017.

[8] E. A. Vincze, "Challenges in digital forensics," *Police Practice and Research*, vol. 17, no. 2, pp. 183–194, 2016.

[9] G. B. Satrya, A. A. Nasrullah, and S. Y. Shin, "Identifying artefact on Microsoft OneDrive

client to support Android forensics," *International Journal of Electronic Security and Digital Forensics*, vol. 9, no. 3, pp. 269–291, 2017.

[10] R. McKemmish, "When is digital evidence forensically sound?" in *IFIP International Conference on Digital Forensics*, Kyoto, Japan, Jan. 28–30, 2008, pp. 3–15.

[11] Dropbox. Company info. [Online]. Available: www.dropbox.com

[12] A. Pichan, M. Lazarescu, and S. T. Soh, "Cloud forensics: Technical challenges, solutions and comparative analysis," *Digital Investigation*, vol. 13, pp. 38–57, 2015.

[13] F. Daryabar, A. Dehghantanha, and K. K. R. Choo, "Cloud storage forensics: MEGA as a case study," *Australian Journal of Forensic Sciences*, vol. 49, no. 3, pp. 344–357, 2017.

[14] Y. Y. Teing, A. Dehghantanha, K. K. R. Choo, T. Dargahi, and M. Conti, "Forensic investigation of cooperative storage cloud service: Symform as a case study," *Journal of Forensic Sciences*, vol. 62, no. 3, pp. 641–654, 2017.

[15] V. Roussev and S. McCulley, "Forensic analysis of cloud-native artifacts," *Digital Investigation*, vol. 16, pp. S104–S113, 2016.

[16] C. Stelly and V. Roussev, "SCARF: A container-based approach to cloud-scale digital forensic processing," *Digital Investigation*, vol. 22, pp. S39–S47, 2017.

[17] D. Quick and K. K. R. Choo, "Dropbox analysis: Data remnants on user machines," *Digital Investigation*, vol. 10, no. 1, pp. 3–18, 2013.

[18] D. Quick and M. Alzaabi, "Forensic analysis of the Android file system YAFFS2," in 9^{th} *Australian Digital Forensics Conference*, Perth Western Australia, Dec. 5–7, 2011, pp. 100–109.

[19] P. Albano, A. Castiglione, G. Cattaneo, and A. De Santis, "A novel anti-forensics technique for the Android OS," in *2011 International Conference on Broadband and Wireless Computing, Communication and Applications*, Barcelona, Spain, Oct. 26–28, 2011, pp. 380–385.

[20] G. B. Satrya and S. Y. Shin, "Proposed method for mobile forensics investigation analysis of remnant data on Google Drive client," *Journal of Internet Technology*, vol. 19, no. 6, pp. 1741–1751, 2018.

[21] F. M. Granja and G. D. R. Rafael, "The preservation of digital evidence and its admissibility in the court," *International Journal of Electronic Security and Digital Forensics*, vol. 9, no. 1, pp. 1–18, 2017.

[22] R. Montasari, "Review and assessment of the existing digital forensic investigation process models," *International Journal of Computer Applications*, vol. 147, no. 7, pp. 41–49, 2016.

[23] Kominfo. (2008) Undang-Undang Republik Indonesia Nomor 11 Tahun 2008 Tentang Informasi dan Transaksi Elektronik. [Online]. Available: https://bit.ly/2T2Dvho

[24] Kemenkumham. (1981) Undang Undang No. 8 Tahun 1981 Tentang Kitab Undang Undang Hukum Acara Pidana. [Online]. Available: https://bit.ly/30tCM91