International Journal of Advanced Science and Technology
Vol. 29, No. 7s, (2020), pp. 3535-3542

Forensics Analysis of Lock Application on Mobile Devices

Abdul Hakim*[1], Yohan Suryanto[2], and Ayubi Wirara[3]

[1, 2, 3] Department of Electrical Engineering, University of Indonesia, Indonesia.

E-mail: abdul.hakim81@ui.ac.id, yohan.suryanto@ui.ac.id, ayubi.wirara@ui.ac.id

Abstract

Encryption is a technique used to support confidentiality from the information. At this time, many applications that provide encryption services for information security on mobile phones like lock gallery, data, etc. These applications are also currently widely used by mobile users. This application will undoubtedly be challenging for digital examiner forensics when the cell phone is evidence using the lock application. To encounter these challenges, this is necessary to prove scientific that can recover encrypted data so that it can assist in the analysis process. In this study, the researcher will use a forensic tool, namely XRY and manual acquisition, using ADB. The results obtained to know to go beyond which forensic tools can be relied on, make a Lock Applications profile, which is possible to get the key that the application uses.

Keywords**:** *encryption, digital forensic, mobile application, forensic tools*

1. Introduction

1.1. Background

The mobile phone becomes an essential element in this current time. Nowadays, mobile phones not only a device but also a "smart" part of our life[10][12]. The development of mobile devices and services increased rapidly, and the mobile industry connects over 3.5 billion people globally to the internet[8][11]. For many of these individuals, the mobile phone becomes a part of life because of the excessively develop features such as web connectivity, increased storage capacity, computing power, upload capabilities, and attractive interface[9].

Mobile phones have a combination of functionality and storage space like business laptops or computers, such as a large amount of data stored on them, especially personal information. For example address book, e-mail, text messages, digital photograph, call history, contact numbers, notes, voice records, web history, videos, photos, calendar events, tasks, SMS, MMS, GPS navigators, browsing history, voicemail recording which can use to address sensitive queries during investigations[9][1][4].

Smartphones have become an essential and potential resource for digital evidence of criminal investigations. Forensic investigators can resolve criminal cases with data stored on a cellphone[9].

Although there are many similarities between computer and mobile phone functions, there are some differences in terms of digital forensic computer and mobile phone devices, as for these differences, explained in Table 1[13].

ISSN: 2005-4238 IJAST

Table 1. Comparison of Computer and Mobile phone Forensic

Aspect	Computer Forensic	Mobilephone Forensic
Source of evidence	– Hard disk – RAM – External memory card	– Internal memory – SIM – External memory card
Can remove the internal storage media	Yes the hard disk can remove easily	No
Operating System	A Limited number of OS	Wide range of OS
Can bypass the authentication password	Yes	Cannot avoid the authentication password during logical acquisition
Power and data cables	Standard power and data cables	Wide range of power and data cables
File system	Standard file system such as FAT	Wide range of file system

In recent years, a new trend that has the potentially alarming to digital forensic has been detected by forensic investigation experts, called anti-forensic [14]. This application is also often referred to as "vault" or "locker". This application looks like a vault or locker, or this application can resemble other simple applications, such as a calculator, media manager, or even a stock application. This application requires users to use secure passcode (like a password) to be able to access and view the data stored in it. Currently, there are hundreds of applications like this that can be downloaded on the Google Play Store [15]. From the personal privacy perspective, a vault application can be used right to secure personal data. Still, as previously stated, that 3.5 million people are connected to the internet via mobile phones, of course, criminals are included [7] [8]. They use mobile phone technology to "gain" something and use mobile phones for their illegal activities and crime [7].

In recent years there has been an increase in police investigations involving this vault application. For example, in 2015, several high school students in Colorado exchanged their nude photos (Colorado sexting scandal: High school faces felony investigation, 2016). Nearly more than half of the students involved in this case used the vault application in the form of a calculator application to hide their pictures from their parents [7].

Currently, there is still no unique and standard definition of anti-forensic; therefore, in this paper, we define anti-forensic to be any attempt to compromise the availability or usefulness of evidence in the judicial process. Forensic evidence can be tampered with availability by hindering its creation, hiding its existence, and by manipulating its authenticity, usability can be corrupted by erasing evidence or by damaging the truth of the data [14].

Mobile devices become an essential part of the field of Digital Forensic due in no small amount of information stored in mobile devices [3]. Here are 4 of the most common anti-forensic methods for digital forensic, namely: destroying evidence, hiding evidence, eliminating sources of evidence, and falsifying evidence [15].

With the emergence of various Anti-forensic applications or vault applications, it becomes a challenge in digital forensic, as mentioned above. Following the purpose of this paper, which is to obtain a profile of the vault/locker application from the results of a

ISSN: 2005-4238 IJAST

report, find a place to lock the lock app then examine further findings from digital forensic.

1.2. Mobile Forensic Research.

Data acquisition can be obtained either through physically and logically. Logical acquisition extract user data that is recognized by the filesystem, whereby deleted files are excluded. Physical asset gets a copy of flash memory and all the data physically stored. One approach of the logical acquisition methods is utilizing ADB-command line utilization on Android. With this, user data can be transferred from an Android device to the forensic workstation. But establishing an ADB connection required the target device to have Universal Serial Bus (USB) debugging mode enabled. Or data can be extracted with user privileges [7].

The ADB method of extraction can copy parts of file systems. If the device is not rooted, ADB will only extract selectively successful, and only unencrypted data, user data, and system information can be acquired. If the device is rooted, the ADB method will be able to extract nearly every file and folders found in the Android file system [15].

ADB's research into substantial extracts was carried out by Lessard and Kessler in 2010, but the results of the study could no longer be used on a new generation of Android devices. In situations where root privileges, locked screens, or ADB cannot be used, researchers obtain physical data by performing recovery on the Android device partition. This is done by turning off the device and entering recovery mode.

Physical acquisition is an acquisition process where bit by bit copy of the invention is made for analysis. This type of acquisition method can have the potential to restore deleted data, which is not possible with a logical acquisition.

While most acquisition typically happens using the software, but the asset can also be achieved at the hardware level. Joint Test Action Group (JTAG) and chip-off are one of the most popular methods. JTAG conducts testing through ports on mobile devices and can be connected to Printed Circuit Board (PCB), which is available on Android devices. And Chip-off detaches the flash memory chip from the PCB and acquires data using a specialized method [7].

In some instances, forensic devices can extract data from specific versions of a data security application. Still, after updating the app, the forensic tool cannot be used to retrieve data that has been locked [6].

According to Ryan Harris, anti-forensic are all experiments carried out to damage the availability or usefulness of evidence in a judicial process. There are four commonly used anti-forensic methods: destroying evidence, hiding evidence, eliminating sources of evidence, and counterfeiting evidence. Destroying evidence means that the evidence is eliminated and cannot be retrieved.

Hiding evidence is moving evidence from the ideal place so that investigative experts find it challenging to find it. This method is not always successful. Still, it is quite useful and practical, uniquely, if the evidence data is placed in a location that is unlikely to be examined by an investigative expert, for example, a calculator application or by naming files to distract. Eliminating evidence source is a method that does not destroy evidence, but this method prevents the formation of evidence.

Counterfeiting evidence is an anti-forensic method by creating a collection of fake evidence that looks like the original. This method serves to trick forensic experts into examining incorrect data [15].

ISSN: 2005-4238 IJAST

2. Methodology and Materials

To experiment with the lock application on an Android device and retrieve media stored in use, the development and testing of this research is divided into three phases: scenario development, program preparation, and program testing.

2.1. Scenario Development

This stage aims to simulate certain user data on an Android device that has been rooted and has not been rooted. Then make the selection of three lock applications found on the Google Play Store. These three applications are installed on an Android device for simulation.

2.2. Program Preparation;

This stage aims to conduct experimentation and research, and a workstation was first configured. The following is a list of equipment used to conduct experiments:

2.2.1. Smartphones

Samsung Galaxy III Mini as a Non-rooted device and Samsung Galaxy 8 as Rooted devices are used in this research.

2.2.2. Android Data Security Application Used in this research.

LOCKit version 2.2.68_ww, AppLock version 3.0.3, and app lock version 1.105 are used in these experiments. Effectively, it is not conceivable to review all Android data security applications here. Simply the three above mentioned applications are evaluated in this research.

2.2.3. Tools Related

ADB for rooted device and XRY version 8.0.0 for the unrooted device were used for this research.

2.3. Program Testing

This stage aims to simulate by installing the three lock apps on rooted and non-rooted Android devices. The results of testing are analyzed to find out the characteristics of each application.

3. Result.

The data extraction method performed on non-rooted Android devices using XRY tools is Full Logical. On the other hand, data extraction for the rooted Android system is created by using the ADB manually.

3.1. Applock version 3.0.3

Results for AppLock on an Android system that already rooted obtained all application data, which was saved on /data/data/com.domobile.applock. In order to see if there are data changes, pull data process can be done before and after running the application setting. The following is the collation results:

Hidden image files using the vault feature are stored in a folder that differs from the previous folder. The information was obtained in the database at /data/data/com.domobile.applock/files/Media. The following image is the information record in the medias table in the database.

ISSN: 2005-4238 IJAST

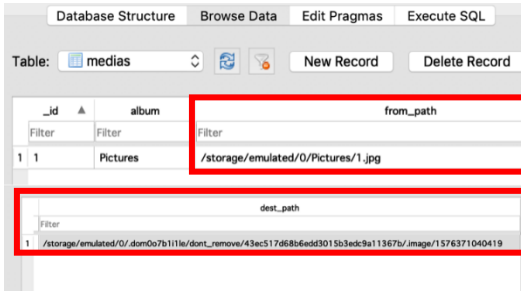


Figure 1. Secured Image Information on Database

Figure 2. AppLock Preference Information

According to the Figure 1 above, the image hidden in the vault has moved into different location from

/storage/emulated/0/Pictures to storage/emulated/0//.dom0o7b1i1le/dont_remove/ 43ec517d68b6edd3015b3ed c9a11367b/.image/1576371040419.

All of the AppLock setting information is saved in com.domobile.applock/shared_prefs with an XML format. There is a change of data inside the XML file, i.e., the file com.domobile.applock_preferences.xml. The file contains credential information of the application. However, some of the contents of the file are not in plain text, as shown in Figure 2 below.

Based on Figure 2 above, we can make a further explanation about what type of algorithms can be used so that it is possible to obtain a password/pin/pattern.

There is not much data that can be obtained on a non-rooted Android system. Based on Android rooted information, an examination is done by using several keywords such as domobile, the Applock, and dom0o7b1i1le. No related results were found while using domobile and Applock keywords.

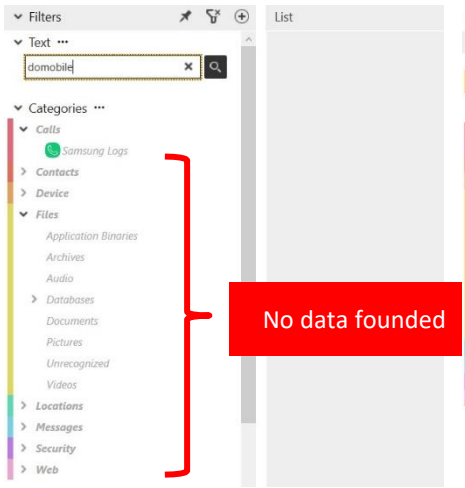


Figure 3. Data List for Keyword domobile

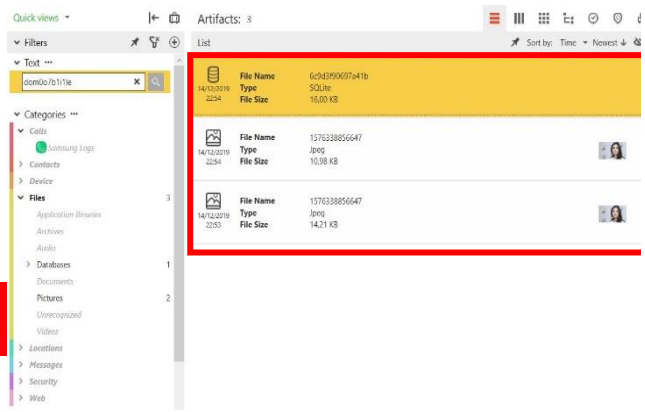


Figure 4. Artifacts List for Keyword "dom0o7b1i1le"

One database file obtained when using dom0o7b1i1l1 keywords, and there is a picture stored in 2 folders. This picture is the same picture previously saved in the vault feature.

ISSN: 2005-4238 IJAST

3.2. App Lock version 1.105

The rooted Android system utilizes the same process, that is pull data performed before and after the application is activated. The result is all of the application data will be stored in /data/data/com.lock.appslocker. However, there is not much information that can be attained. Similar to AppLock, all setting data was stored in path shared_prefs on the XML format.

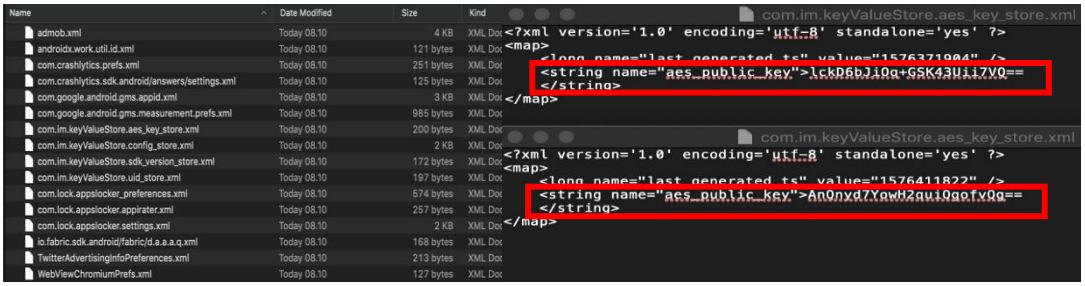
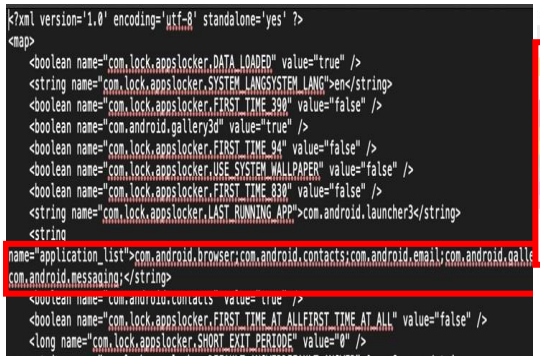


Figure 5. List File on shared_prefs app lock

Figure 6. Content com.im.keyValueStore.aes_key_store.xml

There is one XML file that needs to be highlighted, which is com.im.keyValieStore.aes_key_store.xml file. Data stored in this XML file known to have aes_public_key value and have a different value when using a different *PIN, but it is yet to be discovered what algorithm used to generate this value.*

The information shows an application locked in com.lock.appslocker.settings.xml files. This data matches any application with the previous button settings.



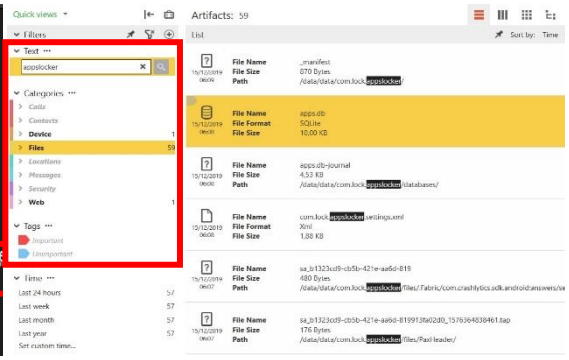


Figure 7. Content com.lock.appslocker.settings.xml

Figure 8. Artifacts List for Keyword "appslocker"

While for non-rooted Android, collation performed by using appslocker keywords. As a result, one artifact for device category, 59 category files artifacts, and one web category artifact. All those artifacts also matched with outcomes gained on rooted Android.

3.3. LOCKit version 2.2.68_ww

Similar to two previous applications on a rooted Android, all LOCKit application data is stored in /data/data/com.ushareit.lockit. All data set is saved on path shared_prefs in XML format just the same with previous applications.

On LOCKit application, PIN setting stored on setting.xml file. But the key is not provided on plain and already been processed with a specific algorithm.

ISSN: 2005-4238 IJAST

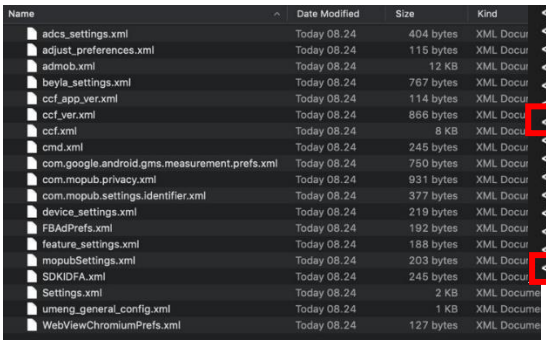


Figure 9. List File on shared_prefs LOCKit

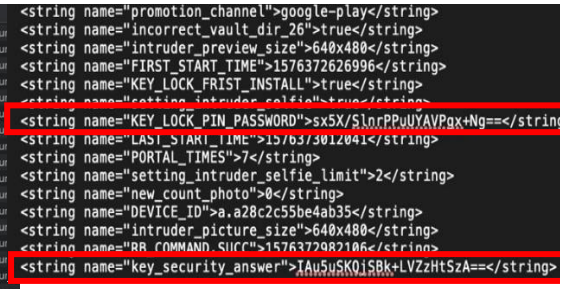


Figure 10. Content Setting.xml

While examination for non-rooted Android using keyword “lockit”. The results are one artifact for the device category, 57 artifacts for file categories, and one artifact for the web category. All those artifacts also matched with outcomes gained on rooted Android.

One of the artifacts gained by using that keyword is picture files, which previously stored into the vault feature. Different from AppLock, LocKit files are saved on encrypted condition with a certain algorithm. Those files have the extension *.kcol dan stored in path /storage/sdcard0/LOCKitVault (do not remove) / .dont_remove/pictures/.

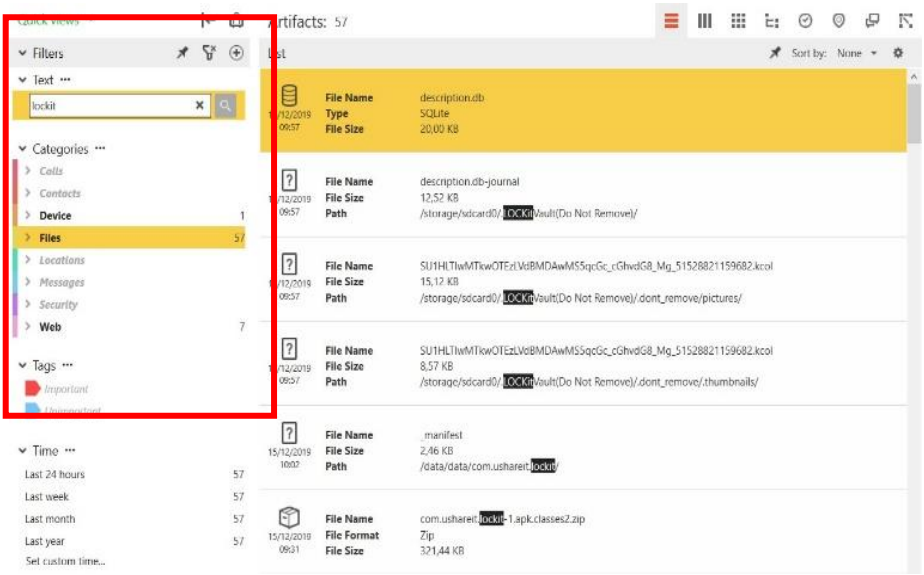


Figure 11. Artifacts List for Keyword “lockit”

4. Conclusion

On a rooted Android device, all data settings that contain credential information on all three applications are stored in the shared_prefs path in the XML format. The contents of the XML file can be read, but the credential information stored in it has been processed with specific unknown algorithms. While on non-rooted Android, all images, messages, contacts, which were previously locked, can still be obtained. Likewise, with data such as XML files, databases, and vaults successfully collected as rooted Android except for AppLock application. In AppLock, the image saved in the vault can be recovered as the original image while in LOCKit, it is still encrypted.

ISSN: 2005-4238 IJAST

5. References.

5.1. Journal Article

[1] Talal M et al "Comprehensive review and analysis of anti-malware apps for smartphones". Springer US vol. 72, 285-337, (2019)
[2] Attia Qamar, Ahmad Karim and Victor Chang, "Mobile malware attacks: Review, taxonomy & future directions". Future Generation Compututer System. Vol 97, 887–909, (2019)
[3] Sundar Krishnan, Bing Zhou and Min Kyung An "Smartphone Forensic Challenges", International Journal of Computer Science and Security (IJCSS), Vol 15, 183-200 (2019)
[4] Abdulhamid S M, Waziri V O, Idris I, Gbolahan A and Alhassan J K "A forensic evidence recovery from mobile device applications". Int. J. Digital Enterprice Technology, vol. 1, Nos 1/2, (2018)
[5] Xiaolu Zhang, Ibrahim Baggili, and Frank Breitinger, "Breaking into the vault: Privacy, security and forensic analysis of Android vault applications". Computer & Security. Vol. 70, 516–531, (2017)
[6] Riaz H "Recovering Data from Password Protected Data Security Applications in Android Based Smartphones" Arab Journal of Forensic Sciences and Forensic Medicine. Vol. 1, 312–322, (2016)
[7] Iosif I Androulidakis, :Mobile Phone Security and Forensics", second edition (2016)
[8] GSMA Association "Connected Society The State of Mobile Internet Connectivity 2019," comScore Mobilens, 2019
[9] Agency G and M. S. Committee "New Research Directions for the National Geospatial - Intelligence Agency," vol. 6, no. 4, (2010) pp. 99–106
[10] Szyjewski G and Fabisiak L "A study on existing and actually used capabilities of mobile phones technologies". Procedia Comput. Sci. vol. 126, (2018) pp. 1627–1636
[11] Chmielarz W "Study of Smartphones Usage from the Customer's Point of View," Procedia Comput. Sci., vol. 65, (2015) pp. 1085–1094
[12] Peterson G and Shenoi S "Advances in Digital Forensics X: 10th IFIP WG 11.9 International Conference Vienna, Austria,". IFIP Adv. Inf. Commun. Technol, vol. 433, (2014) January 8-10
[13] Alghafli K A, Jones A, and Martin T A, "Guidelines for the digital forensic processing of smartphones," Proc. 9th Aust. Digit. Forensics Conf., , pp. 1–8, (2011) January 2011.
[14] Distefano, G. Me, and F. Pace, "Android anti-forensics through a local paradigm," Digit. Invest., vol. 7, no. SUPPL., pp. S83–S94, 2010.
[15] Michaila Duncan, "Detection and recovery of anti-forensic (vault) applications on android devices", CDFSL Proceedings, 2018

ISSN: 2005-4238 IJAST