

Data Decryption and Analysis of Note-Taking Applications

Seyoung Yoon¹[0009–0008–3254–9256], Myungseo Park², Kyungbae Jang³[0000–0001–5963–7127], and
Hwajeong Seo^{*}[0000–0003–0069–9061]

Convergence Security, Hansung University, Seoul, South Korea,
sebbang99@gmail.com, pms91@hansung.ac.kr, starj1023@gmail.com, hwajeong84@gmail.com

Abstract. As smartphone usage continues to grow, the demand for note-taking applications, including memo and diary apps, is rapidly increasing. These applications often contain sensitive information such as user schedules, thoughts, and activities, making them key targets for analysis in digital forensics. Each year, new note-taking applications are released, most of which include lock features to protect user data. However, these security features can create challenges for authorized investigators attempting to access and analyze application data. This paper aims to support investigators by conducting a static analysis of Android-based note-taking applications. It identifies how and where data is stored and explains methods for extracting and decrypting encrypted data. Based on the analysis, the paper concludes by proposing future research directions in the field of digital forensics.

Keywords: forensics · android application · analysis

1 Introduction

As smartphone usage continues to grow, the demand for note and diary applications (commonly referred to as note-taking applications) for storing information has significantly increased. This study aims to clearly define the role of such applications, explain their features and security mechanisms naturally, and maintain a coherent flow. Currently, numerous applications with record-keeping capabilities are available, and new applications are released every year. These applications go beyond simple information storage, offering a variety of convenience features such as schedule management, checklists, and backup functionality. Notably, most of these applications include app-lock security features to enhance the protection of user data, reflecting efforts to strengthen information security. From a digital forensic perspective, such record-keeping applications are likely to contain critical information relevant to investigations. The purpose of this study is to assist authorized investigators in efficiently collecting data from applications equipped with security features. To achieve this, the study identifies the storage locations of critical data in each application and provides methods to decrypt the data when it is encrypted.

We briefly review recent/remarkable work on digital forensics for smartphones. In [1], S. Wu et al. performed a forensic analysis of WeChat on Android smartphones, focusing on methods to extract, decrypt, and analyze user data, including chat messages and multimedia files. The authors detailed techniques for accessing encrypted databases and recovering Moments data from storage paths, offering practical approaches for investigators to retrieve and interpret WeChat data effectively. K. Rathi et al. [2] conducted a forensic analysis of encrypted instant messaging applications on Android, with a focus on WeChat, Telegram, Viber, and WhatsApp. They investigated data storage locations, encryption methods, and the challenges associated with extracting and decrypting data from these applications. Their findings provided valuable insights for investigators, particularly in understanding security differences and methods to access encrypted data. M. Park et al. [3] examined the forensic analysis of two note-taking applications, ClevNote and Samsung Notes, emphasizing their security features, such as access control and data encryption. They employed reverse engineering techniques to identify methods for extracting and decrypting application data, highlighting challenges in analyzing protected files. This study offered important insights for investigators aiming to use securely stored app data as digital evidence. G. Kim et al. [4] explored the forensic analysis of encrypted instant messaging applications, specifically focusing on Wickr and Private Text Messaging. Their research outlined methods

for decrypting databases, multimedia files, and user-entered passwords using reverse engineering and cryptographic analysis. These findings provided actionable approaches for investigators to retrieve data securely stored within these applications. S. Shin et al. [5] analyzed the security features of 56 note and journal applications on Android and iOS, focusing on how secret values and user-generated content are stored. They categorized the applications into three types based on their security levels: no security, partial security, and full security. Their findings revealed that 95% of the applications stored user data insecurely, highlighting critical vulnerabilities in note and journal applications.

In this work, we analyze 11 note-taking applications that had not been addressed in previous research. Particularly, we identify where critical data is stored, described methods for decrypting data when it is encrypted, and outline approaches for extracting backup data. Notably, our analysis includes many of the latest applications released after prior studies were conducted. This allows us to closely examine how data is stored and assess the security levels of these more recent applications.

In summary, Table 1 shows an overview of the results of this work.

Table 1: Evaluation of Data Encryption and Retrieval in Lock-Providing Applications

Application	Data encryption	Cryptographic algorithm	Data extraction method	Backup data transformation
ColorNote Notepad Notes	✓	AES128/CBC/PKCS5Padding AES256/CBC/PKCS5Padding	Device internal storage*	Encryption
Color Notes, Notebook, Notepad Daybook - Diary, Journal, Note Daylio Journal - Mood Tracker Diary & Journal with lock Diary with Lock: Daily Journal (A) Diary with Lock: Daily Journal (B) Moodie - Mood Diary With Lock My Veggie Diary Notepad: Notes Organizer To Do Simple Diary - journal w/ lock	✗	✗	Device internal storage* Device internal storage* Google drive Google drive, Dropbox, Device internal storage* Google drive Device internal storage* Google drive Email, Google drive Device internal storage* Google drive	Header modification None None Base64 encoding None Header Modification None None None None

*: Accessible without root.

1.1 Contribution

- **Analysis of 11 applications with locking features** The 11 applications selected for analysis meet the following criteria. We prioritize applications that have not been previously analyzed, focusing on those with a high number of downloads or released after 2022. For applications that have been analyzed in the past, we ensure they have undergone multiple updates. This is because applications that did not encrypt data in earlier studies may add encryption features through subsequent updates.
- **Provision of methods for extracting backup data** We observe how note-taking applications store data and propose methods to extract backup data without requiring root privileges.
- **Provision of methods for decrypting data** We explain methods to decrypt encrypted data, providing detailed information about the encryption algorithms used and the parameters required for decryption.

1.2 Organization of the paper

Section 3 presents the analysis and findings for the 11 note-taking applications examined in this study, along with details of the experimental environment. Section 4 provides an overall evaluation of the analysis results for the 11 applications. Finally, Section 5 concludes the study by summarizing the findings and offering suggestions for future research directions.

2 Background

2.1 Note-Taking Application

A note-taking application is a tool designed to help users record and organize information in a digital environment. Users can input and save various types of content, including notes, ideas, and schedules. The application typically provides text input functionality and supports multiple formats such as checklists, calendars, tables, and diagrams, allowing users to organize and view their notes in diverse ways. Since note-taking applications are often used to record personal information, they include security features such as password locks, encryption, and user authentication to protect data. They also offer keyword search functionality, making it easier for users to quickly locate specific information among numerous notes. Recently, many applications provide premium services that allow users to customize themes and fonts, enhancing the appearance of their notes and adding a creative element. These features not only improve the visual appeal but also support professional tasks like creating reports and plans.

To prevent data loss, note-taking applications support backup and account synchronization services. With cloud-based backups, users can manage and access their notes seamlessly across multiple devices, ensuring continuity and efficiency. Beyond being simple recording tools, these applications play a crucial role in managing personal information and improving productivity, making them widely used by students, professionals, and general users alike.

2.2 Android Debugger Bridge (ADB)

ADB is a tool that facilitates communication between Android-based smartphones and a host system. While it is primarily used for application development and debugging, it can also serve as a forensic tool in certain cases. C. Easttom et al. [6] explain that ADB is not limited to being a development tool but can also be effectively utilized for data extraction in forensic investigations.

Commonly used ADB commands include the following. The "adb devices" command displays a list of currently connected Android devices. The "adb pull" command copies specified data from the target device to the host system. This command is frequently used to extract application data and is a key tool utilized in this study. The "adb push" command, in contrast, transfers data from the host system to the device. The "adb logcat" command captures system logs and enables real-time analysis. Logcat can be used without root privileges and is particularly helpful for analyzing malware.

ADB serves as a valuable tool for digital forensic investigations. It is free to use and allows users to inspect and control data within Android devices through its commands.

3 Analysis of 11 Android note-taking Applications

We focus on the latest versions of the applications, updated as of the experiment date, and note that sensitive information (such as master passwords and user data) is typically stored in files that require root privileges to access.

Using a rooted device, we examine various pieces of information and identify their storage paths. Note-taking applications often store user data in the form of backup files, which are accessible without root privileges.

Target Applications Table 2 presents the names of the analyzed applications, the update and version information of the applications used for analysis, and the package names of each application. "ColorNote Notepad Notes" encrypts user data, master passwords, and backup data. The remaining ten applications store data without encryption. Details about the data storage locations and the analysis of each application are provided in their respective sections.

Table 2: Target Application List

Name	Date	Version	Package Name
ColorNote Notepad Notes	Jul. 2024	4.5.3	com.socialnmobile.dictapps.notepad.color.note
Color Notes, Notebook, Notepad	Aug. 2024	2.2.3	tidynotes.notepad.notes.notebook.note.checklist.todolist
Daybook - Diary, Journal, Note	Nov. 2024	6.34.0	com.bigheadtechies.diary
Daylio Journal - Mood Tracker	Nov. 2024	1.59.0	net.daylio
Diary & Journal with lock	Dec. 2024	2.0.8	com.zlq.diary.journal
Diary with Lock: Daliy Journal (A)	Nov. 2024	1.047.49.GP	diary.journal.lock.mood.daily
Diary with Lock: Daliy Journal (B)	Oct. 2024	1.4.0	diary.journal.mood.tracker.diarywithlock
Moodie – Mood Diary With Lock	Nov. 2024	1.46	com.casoft.gbdiary
My Veggie Diary	Sep. 2024	2.1.0	com.waddev.vef
Notepad: Notes Organizer To Do	Jun. 2024	9.1.2	pl.netigen.notepad
Simple Diary – journal w/ lock	Nov. 2024	2.2.0	com.komorebi.diary

Environment Table 3 presents the tools and device environment used for application analysis. Most of the analysis in this study is conducted on a Windows 10-based laptop. To access sensitive app data, a rooted device is used, and since the study focuses exclusively on Android applications, a Samsung Galaxy S10 running the Android OS is utilized. Application data is extracted using the ADB 'pull' command, and among the extracted data, database files are analyzed using DB Browser for SQLite. Static analysis of application APK files is performed using Jadx. For backup data, an unrooted Galaxy S23 device is used to explore methods of acquiring data without rooting. Finally, decryption experiments for encrypted data are conducted by implementing decryption algorithms in Java using Eclipse.

Table 3: Analysis Environment

Device and Software	Name	Version
Laptop	Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz, 8GB	Windows 10
Mobile Device (Rooting)	Galaxy s10	Android 9
Mobile Device (Backup)	Galaxy s23	Android 14
Debugging Tool	Android Debugger Bridge	35.0.2
DB Viewer	DB Browser for SQLite	3.12.2
Raw Data Viewer	HxD	2.5.0.0
Decompiler	Jadx	1.5.0
Decryption Implementation	Eclipse	2021-06(4.20.0)

3.1 ColorNote

ColorNote [7] is an Android application released in 2009, with over 100 million downloads, making it one of the most widely used note-taking applications. It offers features for document-style notes and checklists, as well as the ability to save schedules and receive reminders through a calendar. The application provides convenience and security features, such as note search, lock functionality, and backup options. The master password can be set with a minimum of 4 characters and a maximum of 16 characters. It is configured using a combination of numbers, alphabets, Korean characters, and special characters. Information about the storage paths for each type of data is available in Table 4.

Table 4: Data storage paths of ColorNote

Data Name	Path
User Data	databases/colornote.db/"notes"table/"note"column
Master Password	databases/colornote.db/"notes"table /"title"column: "name_master_password" /"note"column: "BACKUP_SECRET_KEY"
Backup Data	Android/data/com.socialnmobile.dictapps.notepad.color.note/files/backup/

User data User data in ColorNote can be obtained from a rooted device. The path where the data is stored is found in Table 4. User data is located in the "colornote.db" file within the "databases" directory and is accessed through the "notes" table. Data with lock settings is set to encryption mode 1 and is stored in an encrypted state in the database. The method for decrypting the encrypted user data is described in Algorithm 1. First, the encrypted data must be retrieved from 'colornote.db'. Next, a fixed password, salt, and iteration count are required, which are confirmed to be hardcoded in the source code through static analysis of the APK file. The fixed password is "ColorNote Password" (excluding quotes, including spaces), the salt value is "ColorNote Fixed Salt" (excluding quotes, including spaces), and the iteration count is 20. With these, a PBE key for AES256 with a key length of 256 is generated using the PBEWITHSHA256AND256BITAES method. Using the generated PBE key along with the initialization vector (IV), which is hardcoded to 0, the user data is decrypted using the AES256/CBC/PKCS5Padding method.

Algorithm 1 Decryption of Encrypted ColorNote User Data

Input: C (encrypted user data in colornote.db)

Output: Decrypted User Data

- 1: $FixedPassword \leftarrow$ "ColorNote Password"
 - 2: $Salt \leftarrow$ "ColorNote Fixed Salt"
 - 3: $Iteration \leftarrow$ 20
 - 4: $Keylength \leftarrow$ 256
 - 5: $PBEKey \leftarrow$ PBEWITHSHA256AND256BITAES-CBC-BC($FixedPassword, Salt, Iteration, Keylength$)
 - 6: $IV \leftarrow$ 0
 - 7: Decrypted User Data \leftarrow AES256/CBC/PKCS5Padding($C, PBEKey, IV$)
 - 8: **return** Decrypted User Data
-

Master Password The master password in ColorNote, like user data, is found in the "colornote.db" file within the "databases" directory, accessed through the "notes" table. First, "name_master_password" is located in the "title" column, and then the "BACKUP_SECRET_KEY" is checked in the corresponding data of the "note" column. The master password is encrypted into a 128-bit ciphertext using the PBEWITHMD5AND128BITAES method, as described in Algorithm 2, and then encoded in Base64 before being stored in the database. The salt and iteration count used during the encryption process of the master password are hardcoded in the source code as "ColorNote Fixed Salt" and 20, respectively. The master password is encrypted to serve as a key for encrypting backup data.

Backup Data The backup data in ColorNote is encrypted and stored on the user's device. As shown in Figure 1, the header of the encrypted backup data contains information to identify the file. The first 8 bytes store the identifier "NOTE" to distinguish the backup data. The second 4 bytes store the integer value 7, hardcoded in the APK source code. It is confirmed that version 1.8.0 of ColorNote stores the value 1, version 2.1.1 stores the value 2, version 3.9.82 stores the value 6, and version 3.9.91 onward stores the value 7, allowing us to distinguish between application versions. The third 4 bytes store data

Algorithm 2 Encryption Master Password of ColorNote

Input: *Password* (Master Password)**Output:** Encrypted Master Password

- 1: $Salt \leftarrow$ "ColorNote Fixed Salt"
 - 2: $Iteration \leftarrow 20$
 - 3: $EncryptedMasterPassword \leftarrow$ PBEWITHMD5AND128BITAES(*Password*, *Salt*, *Iteration*)
 - 4: **return** Encrypted Master Password
-

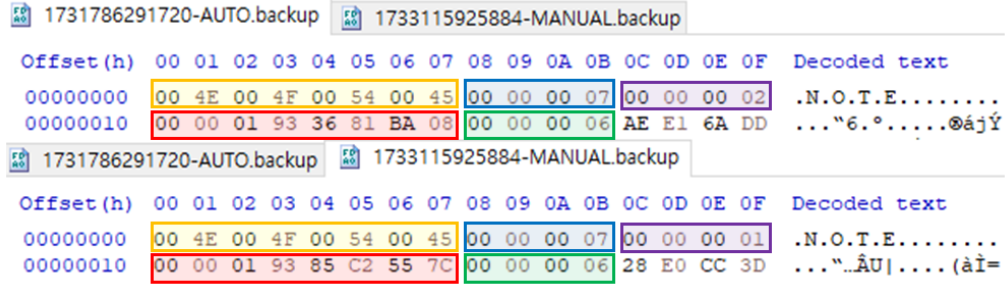


Fig. 1: ColorNote Backup Header

that identifies the type of the backup file. Figure 1 illustrates the header information for automatically saved "AUTO" files and manually saved "MANUAL" files. For the third 4 bytes, "AUTO" contains the value 2, and "MANUAL" contains the value 1. This information is used during the decryption of the backup files. The fourth 8 bytes represent the time the backup file is saved, calculated in milliseconds and stored as a hexadecimal value. Lastly, the fifth 8 bytes indicate the number of notes included in the backup.

Algorithm 3 Decryption of Encrypted Backup Data of ColorNote (MANUAL)

Input: *C* (encrypted backup data), *MK* (Base64 decoding "BACKUP_SECRET_KEY" in colornote.db)**Output:** Decrypted User Data

- 1: $C \leftarrow$ remove 28-byte header
 - 2: $IV \leftarrow 0$
 - 3: Decrypted Backup Data \leftarrow AES128/CBC/PKCS5Padding(*C*, *MK*, *IV*)
 - 4: **return** Decrypted Backup Data
-

Algorithm 3 describes the process of decrypting manually saved ColorNote backup data. Automatically saved backup data follows the same algorithm, but the password used may vary depending on the case. Since backup data is encrypted and then has header information like that shown in Figure 1 added, this header information must be excluded before decryption. Therefore, as shown in line 1 of Algorithm 3, the first 28 bytes of the header are removed. The backup data follows the AES128/CBC/PKCS5Padding method, and the IV value is hardcoded to 0. For manually saved 'MANUAL' files, the key used for encryption is the encrypted value of the master password. As mentioned earlier, this value is stored as a Base64-encoded string in "BACKUP_SECRET_KEY." For automatically saved 'AUTO' files, the default encryption key is "0000". However, if the user enables the "Use Master Password for Auto Backup" setting, the same key as used for 'MANUAL' files is applied.

Figures 2a and 2b respectively show the hexadecimal values of the encrypted backup data and the decrypted backup data. After removing the 28 bytes of the header and performing decryption, it is confirmed that the data is correctly decrypted.

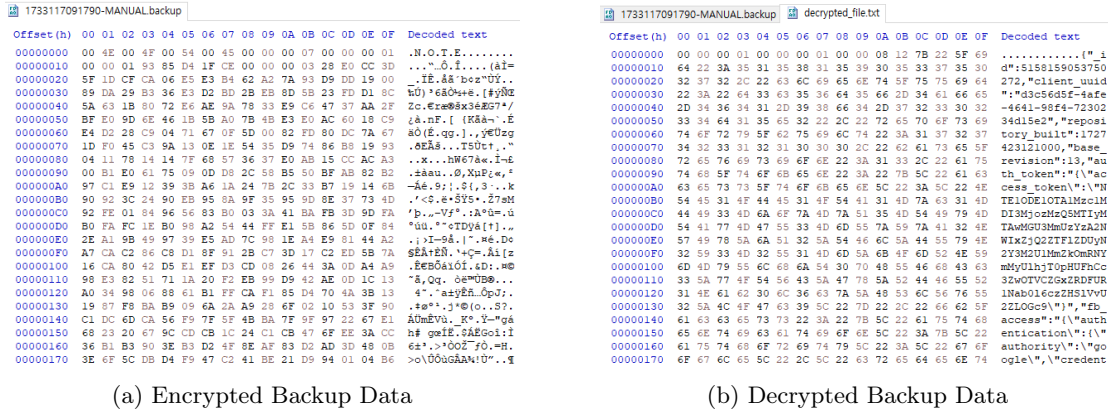


Fig. 2: Decryption Backup Data

3.2 Color Notes, Notebook, Notepad

"Color Notes, Notebook, Notepad" [8] is a note-taking application released in 2023. Basic features such as locking and backups are available for free, while additional features like various background themes, PDF conversion, and ad removal are offered as paid options. The master password can be set as a 4-digit PIN or a gesture lock that connects a minimum of 4 dots to a maximum of 6 dots. Information about the storage paths for each type of data can be found in Table 5.

Table 5: Data storage paths of "Color Notes, Notebook, Notepad"

Data Name	Path
User Data	databases/note.db/"note"table
Master Password	databases/note.db/"td_system_param"table/"param_value"column
Backup Data	Documents/tidyNotes_backup/

Master Password The master password in "Color Notes, Notebook, Notepad" can be found in the "spUtils.xml" file within the "shared_pref" directory or in the "td_system_param" table of the "note.db" file. To retrieve it, check the "param_value" corresponding to the "param_name" entry with the value "KEY_LOCK_PASSWORD" in the table. If 'KEY_LOCK_TYPE' is 0, it is stored as a Gesture Lock, and if it is 1, it is stored as a 4-digit PIN. For Gesture Lock, the corresponding index values for each position are calculated as shown in Figure 3b.

Recovery Backup Data Backup data can be obtained from the device without rooting. "Color Notes, Notebook, Notepad" automatically generates a file named "tidyNotes_backup" in the Documents folder of the internal storage during the backup process. The automatically generated file stores the backup data in the form of a zip file. However, as shown in Figure 4, upon examining the hexadecimal values of the backup data using HxD, it is confirmed that the header value is set to "TI," indicating that it does not follow the standard zip file format. By decompiling the APK source code of TidyNotes, it is discovered that the zip file's header is intentionally coded to be modified from "PK" to "TI" during the backup data creation process. Therefore, by changing the first 4 bytes of the extracted backup data's header back to "PK," a valid compressed file can be obtained. Finally, decompressing the file allows the extraction of the "databases" folder and the "note.db" file.

Table: td_system_param

	param_name	param_value
	Filter	Filter
1	KEY_LOCK_TYPE	0
2	KEY_LOCK_PASSWORD	["13", "9", "5", "8"]
3	KEY_IS_ENABLE_LOCK	true

(a) Master Password in note.db

0 1 2
 3 4 5
 6 7 8

(b) Gesture Lock

Fig. 3: Master Password of "Color Notes, Notebook, Notepad"

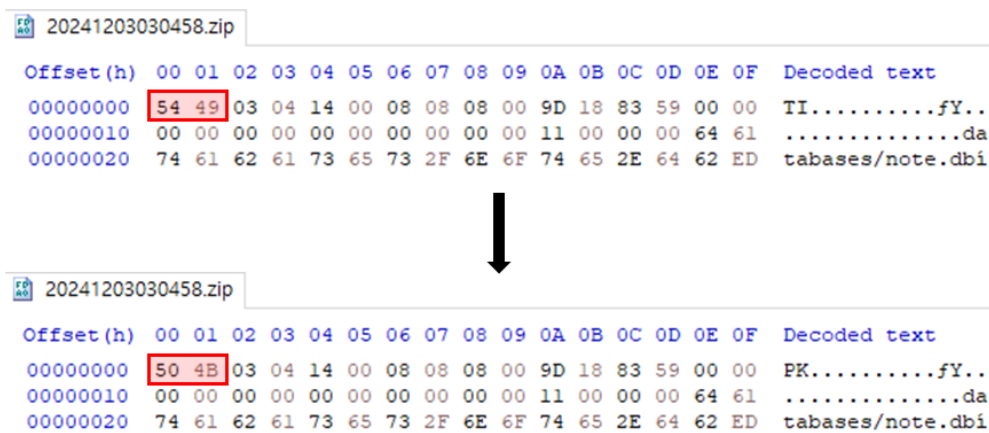


Fig. 4: Recovery Backup Data

3.3 Daybook

Daybook [9] is a diary application released in 2017. Although a prior study by S. Shin et al. [5] was conducted in 2022, this application is included in the analysis to investigate whether multiple updates and changes in the package name over the past two years have affected data encryption. Daybook allows users to lock the application using a 4-digit PIN and requires login through Google, Facebook, email, or an Apple account to use the application. Backups can be saved in CSV format to a user-specified path on the device after re-verifying the login account. If investigators have login credentials for the Daybook application, they can easily extract backup data. Backup data is not stored in an encrypted format, and user data and the master password are also stored without encryption, as is the case in 2022. The storage paths for each type of data are shown in Table 6. A comparison with the study by S. Shin et al. confirms that while the package name has changed, the storage paths remain the same.

3.4 Daylio Journal

Daylio [10] is an application released in 2015 that allows users to record their daily lives briefly. The master password can be set as a 4-digit PIN or biometric information such as fingerprint recognition. In the case of a 4-digit PIN, it is stored in plaintext in the "net.daylio_preference.xml" file located in the "shared_prefs" folder. User data is also stored in plaintext in the "entries.db" file inside the "databases" folder. For backup data, it can be saved using Google Drive, and data can be extracted by logging in with multiple Google accounts, regardless of the Daylio app user. Information about the storage paths for the data can be found in Table 7.

Table 6: Data storage paths of Daybook

Data Name	Path
User Data	databases/cache-daybook-entry.db/"DaybookEntryRoom"table/ "content"column
Master Password	databases/diary-a77f6.firebaseio.com_default.db/"serverCache"table /"Settings/UID/InstallationID": "passcode"
Backup Data	user-specified path

Table 7: Data storage paths of Daylio Journal

Data Name	Path
User Data	databases/entries.db/"table_entries"table/"note"column
Master Password	shared_prefs/net.daylio_preferences.xml/"PIN"item
Backup Data	Google Drive

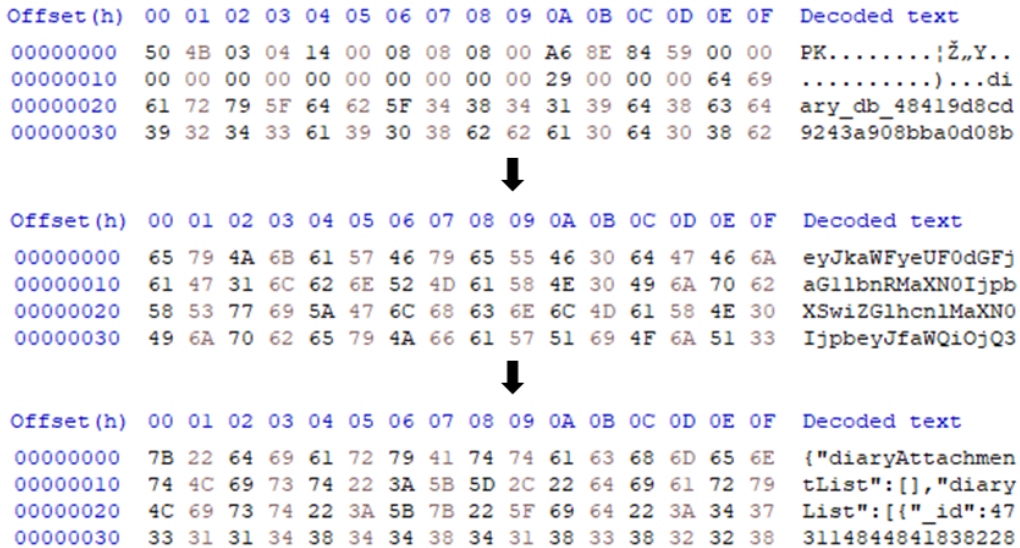


Fig. 5: Recovery Backup Data

3.5 Diary & Journal with lock

"Diary & Journal with lock" [11] is released in 2022 and supports 4-digit PIN lock, gesture lock, and fingerprint unlock. Except for fingerprint unlock, other passwords can be retrieved from the pb file located in the 'datastore' folder within the 'files' directory. Using HxD, we identify the password by analyzing the hexadecimal values of the pb file. User data is stored in plaintext in the 'diary.db' file inside the 'databases' folder. For backups, the application supports Google Drive backup, Dropbox backup, and device storage. Google Drive and Dropbox backups can be extracted by logging in with the desired account. When saving directly to the device, the backup is automatically stored in the 'download' folder on the Galaxy S23. As shown in Figure 5, the header of the backup data extracted directly from the device contains the 'PK' value, and decompressing it yields a new file. This file is then decoded using Base64 to finally retrieve the original backup data. Information about the storage paths for the data is found in Table 8.

Table 8: Data storage paths of Diary & Journal with lock

Data Name	Path
User Data	databases/diary.db/"diary_text"table/"content"column
Master Password	files/datastore/common_settings.preferences_pb
Backup Data	Google Drive, Dropbox, Device storage

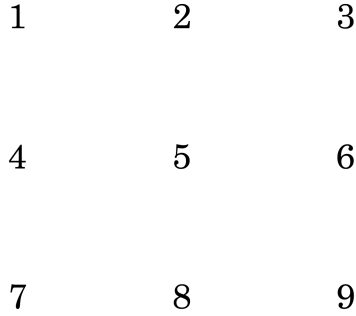


Fig. 6: Gesture Lock (pattern)

3.6 Diary with Lock: Daliy Journal (A)

"Diary with Lock: Daliy Journal (A)" [12] is released in 2022, and basic features are available for free. However, a premium pass must be purchased to access additional fonts and themes. The master password can be set using fingerprint authentication, a 4-digit PIN, or a gesture lock (pattern) that connects a minimum of 4 dots to a maximum of 9 dots. When analyzing data obtained from a rooted device, a file named "App" was found inside the "databases" folder. Although the file format is indicated as "file" checking its header using HxD revealed that it follows the "SQLite format." Therefore, the file's extension can be changed to ".db," or the contents can be viewed directly without modification using DB Browser. User data is stored in plaintext within this App.db file. For the master password, a 4-digit PIN is stored in the "password" column of the "user" table, while the gesture lock can be found in the "shared_prefs" directory, within the "App.xml" file. The string name "passcord_config" contains the "pattern" value, which, as shown in Figure 6, stores the positional values of each dot in sequence. Backup data can be extracted externally by logging in with Google Drive. Information about the storage paths for the data can be found in Table 9.

Table 9: Data storage paths of Diary with Lock: Daliy Journal (A)

Data Name	Path
User Data	databases/App/"diary"table/"title", "content"column
Master Password (PIN)	databases/App/"user"table/"password"column
Master Password (Gesture Lock)	shared_prefs/App.xml/string name="passcord_config": "pattern"
Backup Data	Google Drive

3.7 Diary with Lock: Daliy Journal (B)

"Diary with Lock: Daliy Journal (B)" [13] is released in 2023, is a diary app that provides notification and calendar features. The master password can be set using a 4-digit PIN, and a "security question" can be configured to recover the password if forgotten. User data can be accessed via the "note.db"

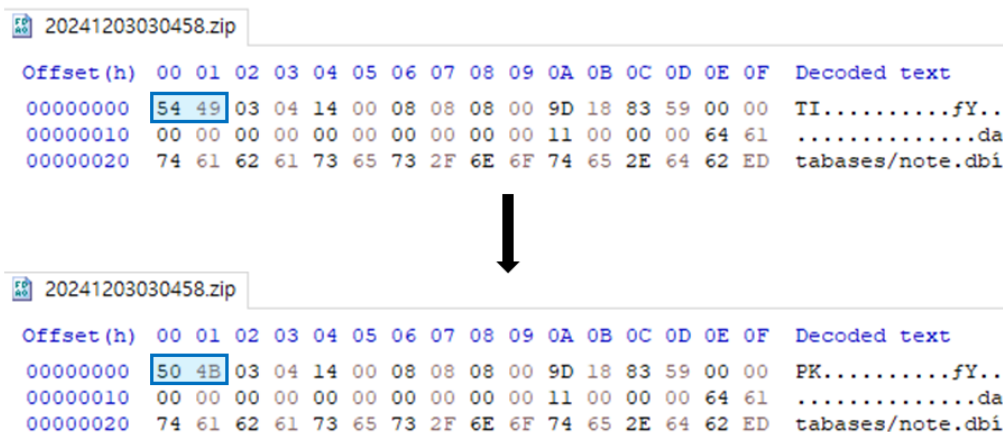


Fig. 7: Recovery Backup Data

database file in the "databases" folder, where plaintext information is stored in the "noteContent" and "noteName" columns of the "note" table. The master password is stored in plaintext under the key "KEY_LOCK_PASSWORD" in the "spUtils.xml" file within the "shared_prefs" directory. Backup data can be obtained without rooting the device. By navigating to the Android data folder and locating "diary.journal.mood.tracker.diarywithlock," the backup files can be accessed inside the "files" directory. These backups are stored as zip files. Information about the storage paths for the data can be found in Table 10.

When examining the header of the zip file obtained from the backup data, it can be observed, as shown in Figure 7, that it starts with "TI" instead of "PK." Therefore, by changing the first 4 bytes of the extracted backup data's header from "TI" to "PK" (50 4B), a valid compressed file can be obtained. Finally, decompressing the file allows the extraction of the "databases" folder along with the "note.db" file.

Table 10: Data storage paths of Diary with Lock: Daliy Journal (B)

Data Name	Path
User Data	databases/note.db/"noteContent", "noteName"column
Master Password	shared_prefs/spUtils.xml/"KEY_LOCK_PASSWORD"
Backup Data	Android/data/diary.journal.mood.tracker.diarywithlock/files/backup/

3.8 Moodie – Mood Diary With Lock

Moodie [14] is released in 2022, supports notification and diary search features. The master password can be set using a 4-digit PIN or fingerprint authentication. User data is stored in plaintext in the "gbdiary.db" file within the "databases" directory, specifically in the "DiaryItem" table. Among the data obtained from a rooted device, it was confirmed using HxD that the master password is stored in plaintext within the "gbdiary_preferences.preference_pb" file located in the "datastore" folder inside the "files" directory. Backup data can be exported or imported using a Google Drive account. It has been confirmed that changing the Google Drive account does not affect the ability to use the backup data. Information about the storage paths for the data can be found in Table 11.

Table 11: Data storage paths of Moodie

Data Name	Path
User Data	databases/gbdiary.db/"DiaryItem"table
Master Password	files/datastore/gbdiary_preferences.preference_pb
Backup Data	Google Drive

3.9 My Veggie Diary

"My Veggie Diary" [15] is released in 2021, is a diary application that supports both English and Korean. The password can be set using a 4-digit PIN. For Android devices, backup data can be saved in the form of a zip file via email or Google Drive. User data and the master password are stored in the paths listed in Table 12. It has been confirmed using HxD that the data contents are stored in plaintext.

Table 12: Data storage paths of My Veggie Diary

Data Name	Path
User Data	/app_flutter/diary_box.hive
Master Password	/app_flutter/settings_box.hive
Backup Data	Email, Google Drive

3.10 Notepad: Notes Organizer To Do

"Notepad: Notes Organizer To Do" [16] is released in 2016, is a note-taking application with over 1 million downloads. The app's locking feature allows users to lock individual notes or the app itself. The master password can be set using a 4-digit PIN, and a password hint can be added in case the password is forgotten. It was confirmed that both user data and the master password are stored in plaintext in the "NotepadDatabase" file within the "databases" directory. Detailed paths are listed in Table 13. Since the "NotepadDatabase" file follows the SQLite format, the database contents can be viewed using DB Browser. Backup data can be accessed without rooting the device and is stored in the form of a zip file. Upon inspection, the backup data was found to be stored without encryption. Additionally, backups can be made using Google Drive, but there were no restrictions on account login.

Table 13: Data storage paths of Notes Organizer To Do

Data Name	Path
User Data	databases/NotepadDatabase/"Item"table/"text"column
Master Password	databases/NotepadDatabase/"Preferences"table/"code"column
Backup Data	Android/data/pl.netigen.notepad/files/Documents/Notepad/

3.11 Simple Diary – journal w/ lock

Simple Diary [17] is released in 2021, provides diary and note-taking features. The master password can be set using a 4-digit PIN, and backup data can be managed using Google Drive. It was confirmed

that backup data can be freely managed regardless of the Google account owner. User data can be accessed through the "app.db" file in the "databases" directory. The master password can be found in the "PREF_NAME.xml" file within the "shared_prefs" directory. Both user data and the master password were confirmed to be stored in plaintext. Detailed paths are listed in Table 14.

Table 14: Data storage paths of Simple Diary

Data Name	Path
User Data	shared_prefs/PREF_NAME.xml/"KEY_SETTING_PASS_CODE"
Master Password	databases/app_db/"DiaryEntity"table/"diaryContent"column
Backup Data	Google Drive

4 Evaluation

This study analyzes 11 Android note-taking applications that provide lock security features. The overall results are presented in Table 1. As shown in Table 1, only one application, "ColorNote Notepad Notes (ColorNote)," encrypts stored data. Among the remaining 10 applications, "Color Notes, Notebook, Notepad" stores the master password using an XOR operation, which has minimal impact on overall security. Backup data is stored in device internal storage accessible without root privileges in 6 applications, while another 6 store the data in cloud systems such as Google Drive. Notably, "Diary & Journal with lock" supports both storage methods. Only "ColorNote" encrypts backup data, while "Color Notes, Notebook, Notepad" and "Diary with Lock: Daily Journal (B)" modify zip file headers for storage, and "Diary & Journal with lock" encodes data in Base64, all of which have negligible impact on security. In conclusion, while many note-taking applications provide lock functionality, their overall security is not robust. Investigators may have little difficulty extracting data from these applications, but this also highlights the ease with which user data is accessed, raising significant concerns about data security.

5 Conclusion

In digital forensic investigations, data stored in smartphone applications is used as evidence to track a user's thoughts or actions. Note-taking applications, in particular, are likely to store sensitive personal information such as the schedules of the individual under investigation. Focusing on such note-taking applications, we analyze 11 Android applications. Although all 11 applications offer security features, we divide our analysis into two categories: cases where data is not encrypted and cases where data is encrypted during storage. Additionally, since one of our goals is to assist investigators in extracting data from applications, we describe methods to extract data using "backup data." We provide details about the locking features of each application, the data storage paths, and, for applications with encrypted data, methods to decrypt the data. Even in relatively recently released applications, we find that the level of data security and methods for extracting backup data are not significantly different from those of older applications. Most note-taking applications are found to store data without encryption or rely on insecure methods such as XOR operations or simply modifying file headers, rather than using proper encryption algorithms, highlighting a significant security vulnerability. We hope that our efforts help investigators analyze data from various note-taking applications and emphasize the importance of improving data security in future applications.

References

1. S. Wu, Y. Zhang, X. Wang, X. Xiong, and L. Du, "Forensic analysis of wechat on android smartphones," *Digital investigation*, vol. 21, pp. 3–10, 2017. [1](#)
2. K. Rathi, U. Karabiyik, T. Aderibigbe, and H. Chi, "Forensic analysis of encrypted instant messaging applications on android," in *2018 6th international symposium on digital forensic and security (ISDFS)*, pp. 1–6, IEEE, 2018. [1](#)
3. M. Park, S. Kim, and J. Kim, "Research on note-taking apps with security features.," *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.*, vol. 11, no. 4, pp. 63–76, 2020. [1](#)
4. G. Kim, S. Kim, M. Park, Y. Park, I. Lee, and J. Kim, "Forensic analysis of instant messaging apps: Decrypting wickr and private text messaging data," *Forensic Science International: Digital Investigation*, vol. 37, p. 301138, 2021. [1](#)
5. S. Shin, G. Kim, S. Kim, and J. Kim, "Forensic analysis of note and journal applications," *Forensic Science International: Digital Investigation*, vol. 40, p. 301355, 2022. [2](#), [8](#)
6. C. Easttom and W. Sanders, "On the efficacy of using android debugging bridge for android device forensics," in *2019 IEEE 10th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, pp. 0730–0735, IEEE, 2019. [3](#)
7. ColorNote Notepad Notes. https://play.google.com/store/search?q=colornote&c=apps&hl=en_US&gl=US. last accessed: 2024-12-05. [4](#)
8. Color Notes, Notebook, Notepad. https://play.google.com/store/apps/details?id=tidynotes.notebook.note.checklist.todolist&hl=en_US&gl=US. last accessed: 2024-12-05. [7](#)
9. Daybook - Diary, Journal, Note. https://play.google.com/store/apps/details?id=com.bigheadtechies.diary&hl=en_US&gl=US. last accessed: 2024-12-05. [8](#)
10. Daylio Journal - Mood Tracker. https://play.google.com/store/apps/details?id=net.daylio&hl=en_US&gl=US. last accessed: 2024-12-05. [8](#)
11. Diary & Journal with lock. https://play.google.com/store/apps/details?id=com.zlq.diary.journal&hl=en_US&gl=US. last accessed: 2024-12-05. [9](#)
12. Diary with Lock: Daliy Journal (A). https://play.google.com/store/apps/details?id=diary.journal.lock.mood.daily&hl=en_US&gl=US. last accessed: 2024-12-05. [10](#)
13. Diary with Lock: Daliy Journal (B). https://play.google.com/store/apps/details?id=diary.journal.mood.tracker.diarywithlock&hl=en_US&gl=US. last accessed: 2024-12-05. [10](#)
14. Moodie - Mood Diary With Lock. https://play.google.com/store/apps/details?id=com.casoft.gbdiary&hl=en_US&gl=US. last accessed: 2024-12-05. [11](#)
15. My Veggie Diary. https://play.google.com/store/apps/details?id=com.wadev.vef&hl=en_US&gl=US. last accessed: 2024-12-05. [12](#)
16. Notepad: Notes Organizer To Do. https://play.google.com/store/apps/details?id=pl.netigen.notepad&hl=en_US&gl=US. last accessed: 2024-12-05. [12](#)
17. Simple Diary – journal w/ lock. https://play.google.com/store/apps/details?id=com.komorebi.diary&hl=en_US&gl=US. last accessed: 2024-12-05. [12](#)