



Analysis of privacy and security of Privacy Vault Applications
(INSE 6120 - Crypto-Protocol and Network Security)

Submitted to

Prof. Mohammad Mannan, PhD, P.Eng.

Submitted by

Kirtan Patel (40203819)
Mahendra Singh Manral (40192561)
Pranesh Sekar (40195581)
Pranshu Mainrai (40206628)
Quiterie Pitel (40206892)

Table of Contents

ABSTRACT.....	3
INTRODUCTION	3
METHODOLOGIES & TOOLS USED	4
TOOLS USED.....	4
<i>Android Debug Bridge (ADB)</i>	<i>4</i>
<i>Android Mobile/VM with Root Access.....</i>	<i>4</i>
<i>Android Studio</i>	<i>4</i>
<i>Jadx - Dex to Java decompiler.....</i>	<i>4</i>
<i>MobSF.....</i>	<i>5</i>
<i>VS Code.....</i>	<i>5</i>
METHODOLOGIES	5
<i>Scenario Generation</i>	<i>5</i>
<i>Analyzing the APK file</i>	<i>5</i>
<i>Reading and Configuring the Shared Preferences.....</i>	<i>6</i>
APPLICATION ANALYSIS.....	7
SECURITY ANALYSIS	7
<i>Applock</i>	<i>7</i>
<i>Audio Manager.....</i>	<i>8</i>
<i>Keepsafe</i>	<i>8</i>
<i>HD SMTH.....</i>	<i>9</i>
<i>GalleryVault.....</i>	<i>9</i>
<i>Calculator.....</i>	<i>9</i>
<i>1Gallery.....</i>	<i>9</i>
<i>PhotoGuard.....</i>	<i>9</i>
<i>Audio Manager.....</i>	<i>10</i>
<i>PhotoSafe.....</i>	<i>10</i>
<i>Private Vault</i>	<i>10</i>
<i>LOCX.....</i>	<i>10</i>
PRIVACY ANALYSIS	10
<i>Permissions requested by the apps</i>	<i>11</i>
<i>Trackers</i>	<i>11</i>
DISCUSSION & CONCLUSION	12
REFERENCES	13
APPENDIX.....	14
APPENDIX I: SCREENSHOTS.....	14
APPENDIX II: TRACKERS PER APPLICATION	21

Abstract

Privacy vault apps are a type of mobile application that allow users to store sensitive information such as passwords, credit card numbers, and personal identification documents in a secure, encrypted environment. These apps are designed to protect user privacy and prevent unauthorized access to sensitive information. In this paper, we will analyze the privacy and security measures employed by privacy vault apps in android, including encryption algorithms, password protection, and access controls. We will also discuss potential vulnerabilities and risks associated with using privacy vault apps and provide recommendations for best practices to ensure the safe and secure use of these tools.

Introduction

The increasing prevalence of personal devices and online platforms has made it more important than ever for individuals to protect their personal information from third parties. One way to do this is by using privacy vault apps, which are mobile applications that allow users to store sensitive information in a secure, encrypted environment. These apps are designed to prevent unauthorized access to sensitive data, such as passwords, credit card numbers, and personal identification documents.

Here is a summary of the findings:

- Most of the privacy vault apps uses some form of password-based offline encryption.
- These encryption parameters such as password, salt, hash is stored in the device itself.
- Some applications store the password in cleartext allowing anyone to read the password without any difficult procedure.
- Some applications do not encrypt the file. Instead, they change the file extension and hide the original name in a config file.
- Some applications move the file to another folder which is marked as hidden.
- Many applications use older, primitive hash functions such as MD5, SHA1 which was broken in nature and reversible.
- Some applications have other obfuscation techniques, like posing as other utility apps like calculator, audio manager etc.
- Even though the files are encrypted in some apps, password recovery data such as security email, security question along with the required answer are in clear text. Examiners can modify the email in the file to a self-owned email, thus allowing the app to send the password to the examiner directly.
- All the applications rely on code obfuscation to represent themselves as a secure vault app.

Methodologies & Tools Used

Tools Used

Android Debug Bridge (ADB)

Android Debug Bridge (ADB) is a command-line tool that is used to communicate with Android devices. It allows developers to access the device's internal functions and manage the device's state. ADB is included in the Android Software Development Kit (SDK) and is commonly used for debugging, installing custom ROMs, and creating and managing backups of the device. ADB can be used over a USB connection or over a network connection, and it provides a variety of commands for interacting with the device. For example, ADB can be used to install and uninstall apps, push and pull files, and reboot the device.

Android Mobile/VM with Root Access

Root access on an Android device is the ability to gain privileged control over the device's operating system. This means that the user has access to the device's system files and can modify them, as well as any system-level settings and configurations. Root access is typically achieved by using a third-party app or utility to unlock the device's bootloader and gain access to the root file system.

Having root access on an Android device can be useful for a variety of reasons, such as installing custom ROMs, removing pre-installed bloatware, and increasing the device's performance. However, it can also be risky, as modifying system files can cause serious problems if not done carefully. It is important for users to understand the potential risks and consequences of rooting their device before attempting to do so.

Android Studio

Android Studio is an integrated development environment (IDE) designed specifically for developing Android apps. It is developed by Google and is available for free on the company's website. Android Studio provides a range of tools and features to help developers create, test, and debug their Android apps, including a code editor, a debugger, and an emulator for testing apps on different devices and configurations. Android Studio also includes a range of built-in templates and frameworks to help developers get started quickly, and it is based on the popular IntelliJ IDEA Java IDE. Overall, Android Studio is a powerful and user-friendly tool for developing Android apps.

Jadx - Dex to Java decompiler

JADX is a tool that can be used to convert Java bytecode into human-readable source code. It is commonly used for reverse engineering Android apps, as it can take the compiled code of an Android app and produce the equivalent Java source code. This can be useful for a variety of

reasons, such as analyzing the inner workings of an app, identifying potential vulnerabilities, or learning from the app's design and implementation. JADX is an open-source tool and is available for free on GitHub. It can be run from the command line or used as a plugin for popular IDEs such as IntelliJ IDEA and Eclipse.

MobSF

Mobile Security Framework (MobSF) is an open-source, automated, and all-in-one mobile application (Android/iOS/Windows) pen-testing framework. It supports static, dynamic, and malware analysis. It can be used by both researchers and developers to identify vulnerabilities and security issues in mobile apps. MobSF also has a RESTful API that can be integrated into CI/CD workflows. It is written in Python and can be installed on Linux, macOS, and Windows operating systems. MobSF is available for free on GitHub.

VS Code

Visual Studio Code (VSCode) is a free and open-source code editor developed by Microsoft. It is a lightweight but powerful tool that is well-suited for a wide range of programming tasks, including web development, data science, and cloud development. VSCode has several features that make it easy to use, including an integrated debugger, a code completion tool, and support for a wide range of programming languages and frameworks. It is available for download on the company's website, and it can be installed on Windows, macOS, and Linux operating systems. VSCode is highly customizable, with a variety of extensions and plugins available to add new features and functionality. In the project, we use VS Code for reading the apk file extracted by jadx and the xml files from shared config.

Methodologies

Scenario Generation

Privacy vault apps typically allow users to store files in a secure, encrypted environment. To store files in a privacy vault app, the user will first need to download and install the app on their device. Once the app is installed, the user can create a password-protected account and use the app's features to add files to their secure storage. This typically involves opening the app and navigating to the appropriate menu or screen, where the user can select the files they want to add to the vault. The app will then encrypt the files and store them securely in the app's virtual "vault." The user can access and view the stored files at any time by logging into the app and navigating to the appropriate menu or screen.

Analyzing the APK file

To analyze an APK file, we will first need to download and install JADX on our computer. Once JADX is installed, we can open it and use the following steps to analyze an APK file:

1. Open JADX and click on the "File" menu.
2. Select "Open" and choose the APK file that we want to analyze.
3. JADX will begin to decompile the APK file and display the resulting Java source code in the main window.
4. Use the navigation panel on the left-hand side of the window to browse the source code.
5. Use the search function to find specific classes, methods, or variables and backtrack to obtain a certain required value.

As we browse the source code, we can look for potential vulnerabilities, security issues, or other problems that may be present in the app.

Reading and Configuring the Shared Preferences

To read and write the shared preferences of an Android app using ADB and Android Studio, you will first need to have ADB installed and set up on your computer. You will also need to have the app that you want to read and write the shared preferences for installed on an Android device.

Once you have ADB set up and the app installed on your device, you can use the following steps to read and write the app's shared preferences:

1. Connect your Android device to your computer using a USB cable.
2. Open Android Studio and launch the ADB shell by entering the following command in the terminal window:

```
adb root
adb shell
```

3. Once you are in the ADB shell, use the following command to display a list of the app's shared preferences files:

```
cd /data/data/[package name]/shared_prefs
ls
```

4. Replace "[package name]" with the actual package name of the app that you want to read and write the shared preferences for. This will display a list of the app's shared preferences files, which are typically stored as XML files.
5. To read the contents of a shared preferences file, use the following command:

```
cat [filename].xml
```

Replace "[filename]" with the actual name of the shared preferences file that you want to read. This will display the contents of the file in the terminal window.

6. To write to a shared preferences file, use the following command:

```
echo "key=value" >> [filename].xml
```

Replace "key" and "value" with the key and value that you want to write to the shared preferences file and replace "[filename]" with the actual name of the shared preferences file that you want to write to. This will add the key-value pair to the shared preferences file.

Application analysis

These are the list of applications analyzed in this project. It includes the app unique identifier i.e. package name, latest version number, last updated date, and the number of downloads.

Name device	Package name	Version	Date	Downloads
Applock	com.domobile.applockwatcher	5.6.2	22/11/2022	100M+
Audio Manager	com.hideitpro	8.6.2	23/03/2022	50M+
Keepsafe	com.kii.safe	11.3.1	11/11/2022	50M+
HD SMTH	com.colure.app.privacygallery	6.8.0.3	04/06/2022	10M+
GalleryVault	com.galleryvault	6.2	18/09/2022	10M+
Calculator	com.hld.anzenbokusucal	9.1	27/11/2022	10M+
1Gallery	app.galleryx	1.0.6- 16.170522	17/05/2022	100K+
PhotoGuard	com.photovault.photoguard	2.9.7	29/08/2022	100K+
Audio Manager	com.wrinfosoft.audiomanager	1.2.8	11/11/2022	500K+
PhotoSafe	hidephoto.hidevideo.vault.photosafe	2.0.16	05/11/2022	10K+
Private Vault	com.techuz.privatevault	1.6	27/07/2021	10K+
LOCX	com.cyou.privacysecurity	2.3.9	31/07/2020	10K+

Security analysis

Applock

DoMobile AppLock is a mobile app that allows users to password-protect specific apps on their Android device. This can help prevent unauthorized access to sensitive information or prevent others from using certain apps without permission. DoMobile AppLock uses a variety of security measures, including pattern locks and PIN codes, to protect the user's apps. It also includes a feature called "intruder selfie," which takes a photo of anyone who attempts to unlock a protected app using the wrong password. DoMobile AppLock is available for download on the Google Play Store.

DoMobile AppLock follows the regular procedure of setting up the password/pattern and the security question. When we open the app for the first time, it will allow you to setup password / pattern of your choice. And obtains permissions like “draw over other apps” and accessibility permissions. It should be noted that it uses the “draw over other apps” permission to lock apps which draws an overlay when other apps are opened.

After setting up the password, we hide a media file the usual way and we will then go ahead and open its shared preferences file in the adb. In that directory there will be `com.domobile.applockwatcher_preferences.xml`, you can find the password in SHA1 hashed form, if we replace the value, with our own value, we can decrypt the app, using our password. We can also see that things like security email were on plain text, replacing this allows us to get the password directly from the mail.

Audio Manager

Audio manager is a vault app with volume control on its main screen, i.e. On opening it looks like it is as audio manager application. We can access a vault feature by press and hold the audio manager icon for 3 seconds.

While running the application on a rooted device, we were able to view all the system files. After analyzing those files, we found the PIN code that was set by me during initial setup of the app i.e. “1111”, was found in a plaintext. The exact location of the PIN code is in the file `data/data/com.hideitpro/shared_preferences.xml`.

This major security issue is not the only one. By analysing the files, we were able to find the image stored clearly, no encryption was performed to protect a malicious person to view the document.

Keepsafe

The “bast app ever for privacy” (promote as such) is a simple vault application to manage secured data on Android or iOS device. A password is required to access the stored data and some functionality such as a secure cloud vault or burglary alerts notifications are available.

It was indeed possible to find where the information on the security key was stored: `key-fabric-analytics`, `key-new-real-password` and `eefa4cb166` are information stored in hash in the file `com.kii.safe_preferences.xml`, as shown in the screenshots. Also, another file stores information about the login: `file_common_login.xml`, also in `shared_pref` folder. Although it was possible to push those files to perform a swap attack, the only working PIN was the user’s PIN : we were not able to find how this was possible but the developers surely put some efforts to provide a secure app, unlike previous versions where it was possible to retrieve the PIN stored in clear text [1].

One thing we were able to notice in the file `safe_preferences.xml` is that the value `key-new-real-password` is only linked to the value of the PIN but the values of `key-last-saved-pin-hash` and `eefa4cb166` depends on the value of the PIN but also on other characteristics of the system.

HD SMTH

[add security content]

GalleryVault

GalleryVault is a straightforward vault app which allows us to lock photos and videos from the gallery. It uses a numerical 4-digit PIN for entry. The app has a simple and easy-to-use interface, and it provides a range of features to help users keep their files private.

Cracking this application was easy. When we go to the application's shared preferences xml file which was ironically named as `passcode.xml`. It had the password in clear text. Furthermore, they also added a fail-safe, like monitoring the key last used time which leaves a fingerprint when the key is modified. By comparing the file modified time and key modified time, the app gets the idea that the file is modified and logouts of the application for safety.

Calculator

Calculator - photo vault is a unique app which hides the vault behind a calculator UI. For other people it showcases itself as a simple calculator providing all mathematical functions, but the user can enter the passcode and hit "=" to get into the vault. We can change the passcode and handle other stuff from there. It also allows you to have a fake vault when someone forces the user to show the vault.

Upon analyzing the storage of the app, we get to know that it uses extensive advertisement and user tracking for monetization. This application uses SHA1 for hashing the password. For some reasons they even have the names of the keys of the xml file as well. It also uses salt to help with hashing which is also stored in the same file. To crack the application, we create a password in an another device and swap the value in the original device, so we can access the app.

1Gallery

[add security content]

PhotoGuard

This application promises the user to store the selected folders securely, with military-grade AES-256 encryption.

Indeed, although we used a rooted device, it was impossible to find out how and where the key to access the application data was stored. We wanted to see if it was possible to retrieve the stored images and videos without having to attack the application frontally: the files are stored in the `photoguard_encrypted_DoNotDelete/pics` folder. They are encrypted and therefore not visible, as the screenshots show it. This app uses the encryption mode CBC with PKCS5/PKCS7 padding. This configuration is vulnerable to padding oracle attacks.

Audio Manager

[add security content]

PhotoSafe

PhotoSafe application hide and encrypt photos, videos and other files by disguising as a security app. To access the vault, the user should ignore the “Scan now” and press during a few seconds the app logo on the home screen. Then, a PIN between 4 and 16 digits is required to access the hidden folders.

First, we managed to find where the PIN hash is stored: in the file `Kidd.xml`, in the folder `shared_pref`. The string named “LockPin” store the hash of the PIN. It was possible to generate this same file on another device and thus know what the hash was corresponding to the PIN “2222”. Thus, by performing a push of this new file and by refreshing the device we were able to access the data stored via PIN “2222”. Although this flaw requires root access to be used, it represents a real danger.

Private Vault

[add security content]

LOCX

LOCX is a classic vault app to help user secure their data and apps by providing app locker and a vault for photos, videos, messages.

To lock the app, a pattern is required: we used the pattern L to initialize the app on the device. Using this root device, we were able to investigate the configuration file and the file “Shared Preferences.xml” stores the string associated to the L pattern. By generating the string corresponding to the Z pattern and pushing it using the adb command *push*. The swapping attack worked, and we access the app using the generated pattern.

Another security issue is the recovery email stored in plaintext: it can easily be change to make it impossible for the user to access its data or to change it for an email accessible for the attacker.

Privacy analysis

We produced analyses of each application using the MobSF tool. To do this, just download the apk of each application and analyze the results of the report. Although it's very rich, only two parts are of interest to us here for the privacy analysis: the **permissions** requested by the applications and the **trackers**.

Permissions requested by the apps

The table below summarizes for each application the permissions that can be requested, the definition of those permissions can be find in the documentation [5].

Permission	1	2	3	4	5	6	7	8	9	10	11	Total
Access_Fine_Location										X		1
Access_Coarse_Location		X								X		2
Camera	X	X	X			X		X		X	X	7
Get_Accounts			X			X				X	X	4
Get_tasks	X					X					X	3
Manage_External_Storage	X	X	X		X	X	X		X	X		8
Mount_Unmount_Fileystems	X											1
Read_External_Storage	X	X	X	X	X	X	X	X		X	X	10
Read_Phone_State				X						X		2
Read_Profile			X									1
Record_Audio		X						X				2
System_Alert_Window	X					X				X		3
Write_External_Storage	X	X	X	X	X	X	X	X	X	X	X	11
Write_Settings										X		1

1. com.domobile.aplockwatcher, 2. com.hideitpro, 3. com.kii.safe, 4. com.colure.app.privacygallery, 5. com.galleryvault, 6. com.hld.anzenbokusucal, 7. app.galleryx, 8. com.photovault.photoguard, 9. com.wrinfosoft.audiomanager, 10. hidephoto.hidevideo.vault.photosafe, 11. com.techuz.privatevault

All these permissions are detected by the MobSF tool as dangerous, but the most requested ones are necessary for the functionality used by the vaults. For instance, external storage can be used to store encrypt data. However, some users should be alerted when they are asked for many permissions (for instance n°10, photosafe) and more precisely permissions that do not bring any value to a vault app.

In red in the table are examples of problematic permissions: location (GPS or network-based), get_tasks (allowing to retrieve running applications), read_profile (allowing to user's personal data) and write_settings (allowing to modify global system settings).

Trackers

The tackers implemented in each app are available in the table in **appendix II**.

Each app has between 1 and 11 trackers. A tracker is used in an application to connect information to fulfill a purpose, categorized here by MobSF (identification, analytics, profiling, advertising...). At a time when there is a market for data, implementing trackers on user habits

can be a fruitful market especially since the user rarely (if ever) realizes the presence of these modules. The most used trackers come from Google and are used for: analytics (Google Firebase Analytics), advertising (Google AdMob) and crash reporting (Google CrashLytics).

It should be noted that there is not necessarily a correlation between the number of trackers in an application and the amount of data collected on the user and his habits. For example, 4 of the 6 trackers used by the 11th application are linked to the use of a facebook account via the application. If the user does not use this feature, the trackers are useless.

The question of privacy is crucial in today's world and an awareness of tracker security is essential: the average user trusts an application that offers him a way to secure his data, but we must not forget that the counterpart is to give up part of his own data.

Discussion & Conclusion

Privacy vault apps are designed to keep your sensitive information, such as photos and videos secure by encrypting the data and requiring a password to access it. However, like any digital security measure, privacy vault apps are not foolproof and can potentially be hacked or breached by malicious actors.

One major vulnerability of personal vault apps is that they often rely on the user's device for security. This means that if someone were to gain access to the device, they would potentially be able to access the information stored in the personal vault app. Additionally, if the device is not protected by a strong password or biometric authentication, it may be relatively easy for someone to gain access to the device and subsequently the personal vault app.

Another potential vulnerability of personal vault apps is the risk of server breaches. Most personal vault apps sync the user's data like the security email or other password recovery Q and A with a server. However, if the server were to be hacked, the attacker could potentially gain access to the sensitive information stored on the server.

The applications also must follow proper security policies themselves so they would not be cracked. Like using proper encryption, use of salt when using hashes for the password, properly encrypting the files using the resultant hash of the password + salt. The application must also be audited by security personnel so that there is no vulnerabilities or loopholes which an attacker can bypass.

In conclusion, while personal vault apps can provide an extra layer of security for your sensitive information, they are not foolproof and can potentially be compromised. It is important to use strong passwords and other security measures to protect your device and be cautious about the information you store in personal vault apps.

References

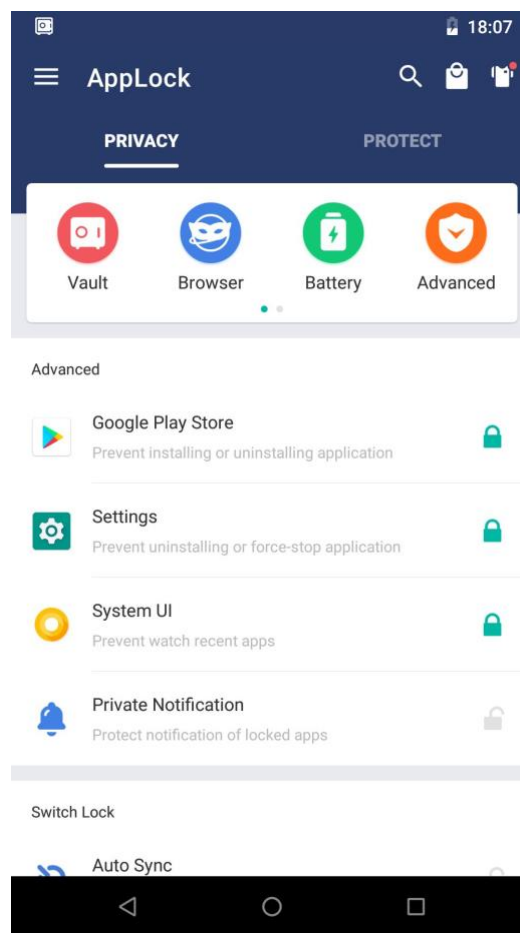
- [1] "Android Debug Bridge (adb)," Alphabet Inc., [Online]. Available: <https://developer.android.com/studio/command-line/adb>.
- [2] "Create and manage virtual devices," Alphabet Inc., [Online]. Available: <https://developer.android.com/studio/run/managing-avds>.
- [3] "Android Studio," Alphabet Inc., [Online]. Available: <https://developer.android.com/studio>.
- [4] "jadx: Dex to Java Decompiler - Github," <https://github.com/skylot>, [Online]. Available: <https://github.com/skylot/jadx>.
- [5] "MobSF Documentation," Ajin Abraham | Magaofei | Matan Dobrushin | Vincent Nadal, [Online]. Available: <https://mobsf.github.io/docs/#/>.
- [6] "Documentation for Visual Studio Code," Microsoft, [Online]. Available: <https://code.visualstudio.com/docs>.
- [7] "DoMobile Applock," DoMobile Lab, [Online]. Available: <https://play.google.com/store/apps/details?id=com.domobile.applockwatcher&hl=en&gl=US>.
- [8] "Hide Photos, Video and App Loc," ANUJ TENANI, [Online]. Available: <https://play.google.com/store/apps/details?id=com.hideitpro&hl=en&gl=US>.
- [9] "Private Photo Vault - Keepsafe," Keepsafe, [Online]. Available: <https://play.google.com/store/apps/details?id=com.kii.safe&hl=en&gl=US>.
- [10] "Hide Something: photos, videos," Colifer Lab, [Online]. Available: <https://play.google.com/store/apps/details?id=com.colure.app.privacygallery&hl=en&gl=US>.
- [11] "hide photo, video," 302 Lock Screen, [Online]. Available: <https://play.google.com/store/apps/details?id=com.galleryvault&hl=en&gl=US>.
- [12] "Calculator - photo vault," FishingNet, [Online]. Available: <https://play.google.com/store/apps/details?id=com.hld.anzenbokusucal&hl=en&gl=US>.
- [13] "1Gallery:Photo Gallery & Vault," todayweather.co, [Online]. Available: <https://play.google.com/store/apps/details?id=app.galleryx&hl=en&gl=US>.
- [14] "PhotoGuard Photo Lock Vault," CUBETIX, [Online]. Available: <https://play.google.com/store/apps/details?id=com.photovault.photoguard&hl=en&gl=US>.
- [15] "Audio Manager:Hide photo,video," MizzOraninlky, [Online]. Available: <https://play.google.com/store/apps/details?id=com.wrinfosoft.audiomanager&hl=en&gl=US>.
- [16] "Hide Photos & Videos-PhotoSafe," DC Mobile Dev Team, [Online]. Available: <https://play.google.com/store/apps/details?id=hidephoto.hidevideo.vault.photosafe&hl=en&gl=US>.

- [17] "Digital Private Vault," Techuz InfoWeb Pvt Ltd, [Online]. Available: <https://play.google.com/store/apps/details?id=com.techuz.privatevault&hl=en&gl=US>.
- [18] "LOCX," thememaker.studio, [Online]. Available: <https://m.apkpure.com/locx-applock-lock-apps-photo/com.cyou.privacysecurity/download>.

Appendix

Appendix I: Screenshots

Applock







Screenshot: App Home Screen

```





<map>
  <boolean name="key_vault_upgrade" value="false" />
  <string name="com.applovin.sdk.impl.isFirstRun">false</string>
  <boolean name="vault_detect" value="true" />
  <string name="password_hint"></string>
  <long name="lock_record_time" value="1668816219721" />
  <string name="pk_applock_uuid">c22f3fc5a7544e16b029bf6f719c319a</string>
  <boolean name="is_image_lock_pattern" value="true" />
  <boolean name="key_accept_privacy_policy" value="true" />
  <long name="pk_password_error_time" value="0" />
  <boolean name="setup_email_tips" value="true" />
  <boolean name="vibrate_pattern_lock" value="false" />
  <boolean name="first_launch" value="true" />
  <string name="image_lock_pattern">k5U2miJ/pdGjbruD4c9s/IbL7iZoU/yx1xxQWyY/Ccc</string>
  <long name="lock_open_count" value="11" />
  <long name="key_install_time" value="1668815773192" />
  <long name="key_record_time1" value="1668815773685" />
  <long name="auto_keep_time" value="1668821336950" />
  <int name="vault_data_level" value="2" />
  <int name="pk_open_main_count" value="11" />
</map>

```

Screenshot : Shared Preferences file <com.domobile.applockwatcher.xml>

 medias	File folder		15/11/2022 15:17
 important_folder,dont_move	File	0 KB	15/11/2022 15:17
 please_backup_all_files_when_in_need	File	0 KB	17/11/2022 20:18
 please_backup_all_files_when_in_need	Text Document	0 KB	15/11/2022 15:17

Screenshot: .do0mo7bi1e1 folder created to store the data

 0db8ef3bd8584e6a8dccc5c9026d0a7a	File	33 KB	15/11/2022 15:00
 54ae02eef203441e81f024b4de61911f	File	2,458 KB	15/11/2022 15:17
 268f3a8c7256478588a56e16d60dada6	File	2,006 KB	15/11/2022 15:17
 e1b7bb6db6a04569befa1b52dfcfbb3b	File	8 KB	15/11/2022 15:00

Screenshot: content stored in media (two (2) videos and two (2) photos)

Audio Manager

Keepsafe

```
com.kii.safe_preferences.xml
1 <?xml version='1.0' encoding='utf-8' standalone='yes' ?>
2 <map>
3   <boolean name="hub-tutorial-finished" value="true" />
4   <string name="couchbase_owner_id">YJgrlyajSN2kzTkb6h0tiw</string>
5   <set name="PREF_AVAILABLE_VAULTS" />
6   <string name="COUCHBASE_SESSION_ID">18c1673eb5d30b9bc49ebf72d05cb9f04c68c8b3</string>
7   <long name="session-last-activity-time" value="1670530810270" />
8   <int name="pin-entry-number-of-tries" value="0" />
9   <string name="key-fabric-analytics">X7K8IEpzK7nLLSi+Fn7XLjc=&#10; </string>
10  <long name="last_update_time" value="1670530809784" />
11  <boolean name="dont-ask-for-storage-permission" value="false" />
12  <string name="key-new-real-password">7C9367530148F12C6ABF4B4F69D5BA40E31E3CA72680ACAFE0994E1108BAFC45</string>
13  <string name="eefa4cb166">odeq+Q1IjjIehg0KimYonErjIY=&#10; </string>
14  <boolean name="ACCOUNT_PIN_MIGRATION" value="true" />
15  <boolean name="USE_REWRITE" value="true" />
16  <int name="pref_lock_type" value="0" />
17  <long name="USERLOG_MIGRATION" value="1670440564644" />
18  <string name="key-last-saved-pin-hash">$2b$04$nVsJGdC0/5hmZJGaSMKQSeXJXEKz61zSdIkukHvfouWPjF0nJppL2</string>
19 </map>
20
```

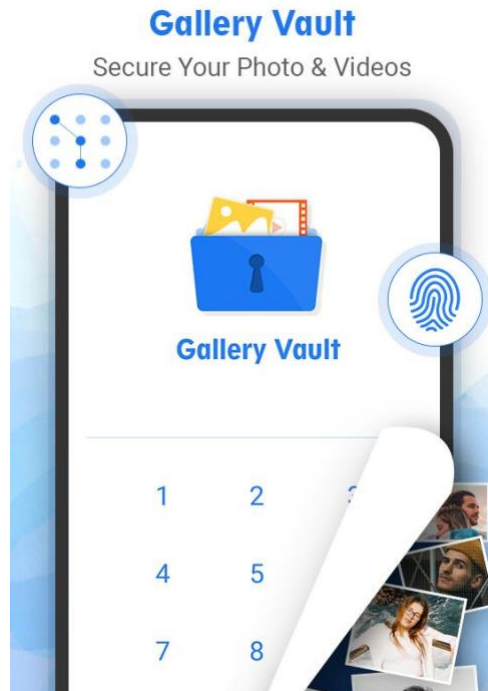
Screenshot 1 : File storing information for logging into the app (hashed information) with the PIN = '1111'

```
consent-prefs.xml device_id.xml.xml file_common_login.xml FirebaseAppHeartBeat.xml SafeDKToggles.xml Usage-Metrics.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
  <string name="key_common_login_file">U/0f3RM6Q6H0upy/8Hi0GwuAE1UeJILJhakdDf7Sws63R1A/APFzhaZ3nVVlnPNRWkL6FhZ9JVNKALKd8leT
  <string name="key_email">FULw7a4Zm21AN0KEY+7RjvC5urcotZRW9qs29nAqQRId1oIk9+Rq&#10; </string>
</map>
```

Screenshot 2 : key_common_login_file.xml storing information about email and PIN

HD SMTH

GalleryVault

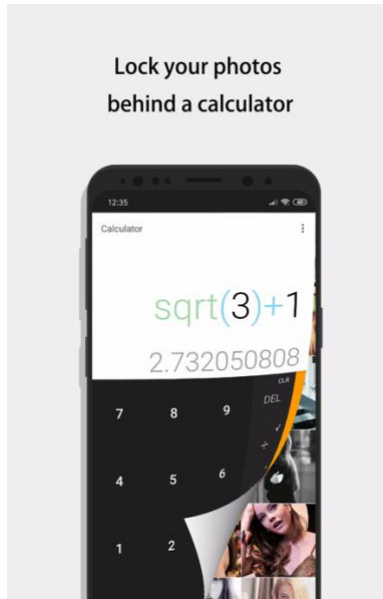


Screenshot: GalleryVault Initial Screen

```
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
  <int name="key_hidden_folder_position" value="0" />
  <boolean name="key_setting_pass_first" value="true" />
  <string name="key_use_last_time">1668823824769</string>
  <string name="key_pass_protection">6666</string>
  <int name="key_type_list" value="0" />
  <boolean name="key_turn_on_pass" value="true" />
  <boolean name="check" value="false" />
  <int name="key_hidden_folder_select" value="0" />
</map>
```

Screenshot: Shared Preferences file <passcode.xml>

Calculator



Screenshot: Calculator: Photo Vault Home Screen

```
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
  <int name="BB7F6051EAFB5D51937DDA821CE7090CAC2337751140B6CBEB44BA5C2B6646BE" value="0" />
  <string name="A4CDA7B11C9207943C99DFE00A362E1503D904E2032364D4D66F22966780A026">
209CD24498A71693A05C65B908E7A76365F323ADC191D7301F83E77594589DD02033D77653FA17A007058B0EAD94A402DBD2E3D926607
</string>
  <string name="57DFA9AEC99CD87013E3862B9DE5B7D">E9759F8CF59EEC2EB08EA324E8CC52AF</string>
  <boolean name="70E68A9A53D3B196256E9BE3051AE5EF8C58A353B24B3170678F8324016E8246" value="true" />
  <boolean name="47B358BD6433B8649576732C3DB54D61" value="false" />
  <boolean name="45C31358EC0450BB34EE65BAC992AA6383B76D839BAFC8CC350C429354691DA7" value="true" />
  <int name="7D1366C9E88E00008EC899E2955C71EE" value="0" />
  <long name="DC2D8610F8D860C92692A6F579787EDE0DF775082E7A68920B8CFA83A4CFE31C" value="1669054700424" />
  <boolean name="60161F74D11DF061F9676DAFE30A2053" value="true" />
  <boolean name="BDF2B6FA524143BC8ECBCD58388054A7" value="false" />
  <boolean name="5763EC2570F3DA44D0313071FEF8725E" value="false" />
  <boolean name="B8AA07D9F2EE25C13802B05445220D22" value="true" />
  <long name="9E5463BB85EC24CE42BA6CE69CDF97361BE43A485AB94C2F367133A74D8F37DD" value="1669054696434" />
  <boolean name="60E8BBAD084AE457138B43AB48A70B0C7F7C5A5DF2535BB452C5F2035C6B43DF" value="true" />
  <long name="DC2D8610F8D860C92692A6F579787EDE998603E3FA72493DF934898979E3B3B3" value="1" />
  <long name="3AE5BC8FEC18A3B957C5DE0A0525AB4" value="1669054510614" />
  <int name="78C6A34CA4FB55D3400911181FB5858D87F9E4E6EC1AD0D0C6D538887EE649EC" value="5" />
  <boolean name="3A87645AEC0A73E4580374F6C7BD3D8264EFA51120E91B73B349F50E0B40FEA" value="true" />
  <int name="175B39143454A26903A3E8195FEAE735" value="4" />
  <boolean name="617A1BEF183B542C4623E189EFBE6B6C0A4A6641D2774D57997992D186F03E10" value="false" />
</map>
```

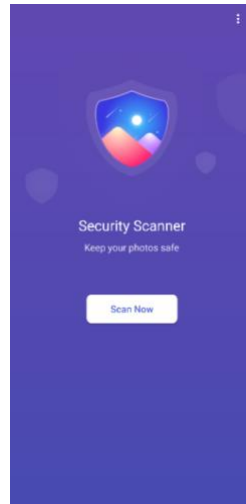
Screenshot: Shared Preferences file <share_privacy_safe.xml>

1Gallery

PhotoGuard

Audio Manager

PhotoSafe



Screenshot 3 : Interface looking like a Security App

```
Kidd.xml
<long name="is_cloud_supported_time" value="0" />
<boolean name="has_ever_start_fresh_user_promotion" value="true" />
<string name="signature">e8611186-c989-4d47-86a1-77e2e097b2cf</string>
<boolean name="is_unlocked" value="true" />
<long name="refresh_account_info_timestamp" value="1670444054648" />
<long name="navigation_finish_time" value="1670442982929" />
<string name="AuthenticationEmail">inse6120f2022@gmail.com</string>
<int name="ChannelId" value="0" />
<string name="gallery_vault_folder">.photosafe_DoNotDelete_1670442978</string>
<long name="fresh_install_time" value="1670442631946" />
<boolean name="db_changed" value="false" />
<boolean name="is_first_show_choose_outside_page" value="false" />
<boolean name="gpph_block_start_background_enabled" value="false" />
<string name="promotion_source">Global</string>
<boolean name="has_add_recent_pictures_and_videos_shown" value="true" />
<int name="FreshInstallVersionCode" value="2016" />
<boolean name="is_license_promotion_shown_in_navigation" value="true" />
<long name="request_time_of_discovery_tools_apps" value="1670442983935" />
<string name="last_verification_sent_data">{"type":3,"time":1670443078954,"ema
<int name="long_press_prompt_show_times" value="11" />
<string name="last_android_id">5b7b0a3c940e39ed</string>
<boolean name="enable_cloud_sync_tip_never_show" value="false" />
<boolean name="prompt_enter_app_enabled" value="false" />
<string name="LockPin">011C945F30CE2CBAFC452F39840F025693339C42B59C67BF196A475
<int name="launch_times" value="4" />
<boolean name="has_show_add_file_tip" value="true" />
<boolean name="setting_changed" value="true" />
<boolean name="sdcard_writeable_of_os_ga" value="true" />
```

Screenshot 4 : Kidd.xml in shared_pref folders

```
<string name="LockPin">011C945F30CE2CBAFC452F39840F025693339C42B59C67BF196A475;
```

Screenshot 5 : Example of PIN hashed (for PIN='1111')

```
"LockPin">FEA7F657F56A2A448DA7D4B535EE5E279CAF3D9A934B535800B1CBA8F96A5D72F
```

Screenshot 6 : Example of PIN hashed (for PIN='2222')

Private Vault

LOCX

Appendix II: Trackers per application

Tracker	1	2	3	4	5	6	7	8	9	10	11	Total
AdColony	X											1
Ajust			X									1
Amplitude			X									1
AppLovin	X	X	X		X					X		5
Bugsnap			X									1
ChartBoost					X							1
Facebook Ads	X	X					X			X		4
Facebook Analytics											X	1
Facebook Login										X	X	2
Facebook Places										X	X	2
Facebook Share										X	X	2
Fyber			X									1
Google AdMob		X	X	X	X	X	X	X	X	X		9
Google Analytics										X		1
Google CrashLytics	X	X		X		X	X	X		X	X	8
Google Firebase Anlaytics	X	X	X	X		X	X	X	X	X	X	10
IAB Open Measurement	X	X	X		X							4
Inmobi	X	X	X		X							4
Integral Ad Science		X										1
ironSource			X		X							2
Moat		X										1
Pangle	X											1
Pubnative			X		X							2
Smaato	X				X							2
Twitter MoPub		X										1
Unity3dAds									X			1
Vungle	X											1
Total	10	10	11	3	1	10	4	3	3	9	6	

1. com.domobile.applockwatcher, 2. com.hideitpro, 3. com.kii.safe, 4. com.colure.app.privacygallery, 5. com.galleryvault, 6. com.hld.anzenbokusucal, 7. app.galleryx, 8. com.photovault.photoguard, 9. com.wrinfosoft.audiomanager, 10. hidephoto.hidevideo.vault.photosafe, 11. com.techuz.privatevault