Annals of Science and Technology - A, Vol 7 (2): 1-20, 2022
Copyright: An Official *Journal* of the Nigerian Young Academy
ISSN: 2 544 6320

sciendo

ARTICLE

A Multilateral Privacy Impact Analysis Method for Android Applications

Kelly E. Orjiude[1] and Chika O. Yinka-Banjo[1]

[1]Department of Computer Science, University of Lagos, Nigeria

Received 12th April, 2022, Accepted 26th June, 2022
DOI: 10.2478/ast-2022-0005

**Corresponding author*
Chika O. Yinka-Banjo E-mail: cyinkabanjo@unilag.edu.ng

Tel: +2348(0)33424289

Abstract

Most people's private lives can be monitored by smartphone applications (apps). Apps have the potential to invade private spaces, access and map social interactions, track users' whereabouts, and track their online activities. Our interest is in the volume of data that a specific app can and seeks to retrieve on a smartphone. Smartphone app privacy friendliness is normally evaluated based on single-source analyses, which often do not offer a thorough assessment of the app's actual privacy threats. In order to analyze Android apps' privacy, this study proposes a multi-source methodology. Our data sets and methodology from app manifestos, privacy policies, vulnerability analysis and user reviews were described. Results from a case study on ten well-known finance applications operating in Nigeria were provided in order to assess our methodology. Our findings showed distinct patterns regarding the possible privacy implications of apps, with some of the apps in the data set infringing fundamental privacy principles. The case study's findings reveal significant differences that can guide users in making relevant app choices.

Keywords: data, permission, privacy, protection, smartphone, vulnerability.

1.1 Background

With over 3.8 billion smartphone users worldwide, it is not surprising that the mobile application industry is booming (Turner, 2021). Smartphone penetration and application usage are continuing to grow at a steady pace, with no signs of slowing down anytime soon. As compared to the App store owned by Apple with over 2.21 million applications, Play Store owned by Google boasts of over 3.47 million applications as of Q1 2021 (Statista, 2021). Smartphones, such as the iPhone and Android, provide users with a wide range of capabilities through a diverse set of software apps. Application developers employ numerous sensors on smartphones, such as the accelerometer, camera, and GPS, to provide users with needed functionalities.

Since smartphones are often switched on and in the hands of users, application developers take advantage of this to acquire unprecedented access to user data. Smartphones, however, come with many useful features and capabilities, as well as many privacy and security concerns. While consumers are constantly using smartphones for their daily tasks, the fact that these gadgets process a lot of personally identifiable information is not often clear to the user, and they often have no control over it. The case with Cambridge Analytica (Isaak et al., 2018) is amongst the popular scandals that ushered privacy to the forefront and demonstrated how difficult it is to secure our privacy in the age of social network. TickTok admitted to sending and processing private and individually identifiable user information to China in its first few years of operation (Ryan et al., 2020).

Although incorporating privacy as well as data security and protection requirements in mobile apps is a difficult task, it is expected that when sensitive and personal user data is at risk, mobile applications follow legal data protection rules as well as conventional privacy and security recommendations to maintain data privacy. Due to either poor design decisions or inadequate implementations, many popular programs that take and process sensitive user data often fail to offer even minimal privacy protection to customers. (Papageorgiou et al., 2018).

1.2 Motivation and Research Questions

Our motivation is driven by several fundamental questions: What privacy sensitive data do the apps aim to obtain from the users? Are the apps' behaviour consistent with what has been indicated in their privacy policies? Are there vulnerabilities that would undermine the information security and privacy protection objectives? To which extent are the apps compliant with legal requirements? To address these questions, a method that combines four metrics: (i) apps' data access potential from their permission requests, (ii) vulnerability analysis (iii) coverage of data protection principles by their corresponding privacy policy texts, and (iv) user review analysis. The collected data is used as a basis for a multi-perspective privacy and security analysis of apps, enabling an understanding of privacy and security related risks of Android apps and informing personal decisions about whether to use an app in the future.

1.3 Problem Statement

With the rapid growth of technology in recent years, the large populace is surrounded by or even dependent on the use of smartphones as they are now an indispensable part of people's daily lives. Smartphone applications provide a wide range of features such as navigation, entertainment, fitness, banking, etc. To provide such context-sensitive services to users, applications need to access users' data including sensitive ones, which in turn, can potentially lead to privacy invasions. While various legislations such as the European Union General Data Protection Regulation (EU GDPR), the Nigeria Data Protection Regulation (NDPR), etc. demand best data privacy practices, applications are still found wanting in the implementation of these best privacy practices.

1.4 Aims & Objectives

The aim of this study is to see how much data Android applications can and will retrieve in smartphones. The following objectives would have been accomplished by the end of this project:

a. Identification of how much personally identifiable information (PII) applications collect from smartphones.
b. Identification of the possible misrepresentations or inconsistencies in application privacy policies in comparison to the actual PII collected by the apps.
c. Identify common security vulnerabilities on Android apps.
d. Educate users on what they could do to protect themselves from app privacy invasion in their smartphones.

2. Theoretical Background

2.1 Literature Review on Related Area

Android and iPhone are the leading smartphones, with Microsoft and Blackberry trailing well behind in terms of sheer market dominance. Independent application developers can promote their apps on Google and Apple's platforms, which customers can download on their devices. As of June 2021, the App Store had over 2.2 million iOS apps and Google's Play Store had over 3.48 million Android apps, allowing many autonomous app developers to market their apps directly to end consumers and resulting in a diverse range of apps (Statista, 2021).

Previous research has identified a correlation between data collecting and advertisement as a strategy for sales. Companies have now shifted their strategy on ads that are targeted to consumers (Leontiadis et al., 2012). Codes from third parties, can be used to generate adverts in applications. Because targeted advertising demands the collection of personal information about users, such advertising revenue models may mandate additional permissions, which brand them invasive when it comes to privacy (Book et al., 2013). (Lin, 2013). Apps could as well include analytics programs from third parties, the goal of which is to collect data on users' interactions with the app. Previous studies examined application developers' security habits and revealed significant sections of apps having flaws in security and/or security code that aren't well-implemented (Egele et al., 2013; Fahl et al., 2012).

Users' views and aspirations concerning privacy and security in smartphones have been studied in the past (Beneson et al., 2012; Felt et al., 2012; Mylonas et al., 2012; Fife et al., 2012). The types of permissions that apps seek often surprise users, the frequency with

which data is collected, as well as the receivers of the data (Lin et al., 2012; Chin et al., 2013). Furthermore, most consumers do not understand privacy notifications, particularly on Android smartphones (Felt et al., 2012). Consumers are anxious about their mobile privacy, yet they are neither notified nor permitted to take protective measures. As a result, the choices made by the developers of these apps have a significant impact.

Users' reactions to privacy policies have been studied previously. While privacy policies appear to provide notice to consumers, the reality is that the ambiguous wording and the time required to read long texts create major usability barriers (McDonald et al., 2008).

The assessment of privacy risk and impact is hampered by a widespread lack of empirical data on which to base privacy risk analysis (Fritsch et al., 2011). Due to a rarity of occurrence and damaged data, risk estimates are difficult. As a result, analysis looks for alternative indicators, such as static program code features or code behaviour (Paintsil et al., 2013; Paintsil et al., 2011). The privacy of smartphone apps was investigated by monitoring certain sensitive permissions such geolocation, storage, phonebooks, and mobile number. 5 of those apps were discovered to implement dangerous functionalities from a sample of 311 of the most popular apps on Google Play, and therefore, should be installed with caution (Enck et al., 2009). Following that, a decompiler that retrieves the source codes of Android app straight from its installation image, with the goal of better understanding smartphone app security was proposed (Enck et al., 2011). They used automated tests and manual inspection to examine 21,000,000 lines of retrieved software code from about 1,100 free apps, revealing the usage or misusage of personal/phone identifiers as well as significant infiltration of marketing and analytics networks. TaintDroid is a technique that examines the activities of 30 prominent Android apps (Enck et al., 2010). According to the findings, two-thirds of the applications handle sensitive data suspiciously, and 15 of the apps send consumers' whereabouts to external marketing services. FAIR is an Android tool for privacy risk evaluation on its apps (Hatamian et al., 2017). It takes advantage of an app habit surveillance techniques to gather data on sensitive resource access. The authors proposed utilizing a fuzzy logic-based technique to calculate a privacy risk score that considers the kind, amount, and regularity of resource called based on some already set rules. Their survey of the 15 highly prominent applications on Google Play by installation throughout different app categories reveals a quantitative evaluation of applications via informing users of observed privacy violating occasions.

Although these are important works and provide insights for privacy researchers, but they do not consider the importance of app meta data analysis such as user reviews, privacy policy, manifest declaration, etc.

The authors explored the matter of trust while installing a new smartphone application in (Kuehnhausen et al., 2013). They evaluated the trustworthiness of mobile apps using trust criteria such as app ratings, reviews, and permissions. In a similar vein, an algorithmic approach was developed to evaluate the integrity of smartphone apps (Habib et al., 2018). Their architecture is based on the app's reputation as well as cutting-edge static analysis technologies. They put their architecture to the test on a sample of Google Play store apps and discovered that it outperformed earlier methods. The privacy-friendly features of smartphones applications were not examined in any of these two studies. The significance of analysis of application privacy policy, as well as the link between dangerous permission demands (in the manifest file) and purpose description (in privacy policy), were not investigated. As a result, the relevance of such issues in were considered in our work and sought to overcome these constraints.

2.2 Privacy and data protection risks

The threats to the security and privacy in smartphone apps are primarily due to their nature as smartphone software, and the peculiarity of mobile app creation and dissemination. A summary of the pertinent dangers and threat factors are provided below:

i. Numerous sensors and a plethora of data

Smartphones have access to a wide range of confidentiality data (like banking, medical, and info) given by consumers via a variety of mobile apps. Furthermore, smartphones contain a variety of sensors (camera, WIFI, GPS, microphone, accelerometer, and so on) that generate personal data and metadata (geolocation, temp, and timestamp) that can have unintended privacy ramifications. Smartphone motion data, like gyroscope & accelerometer signals given by even the most prevalent handsets, has been shown to quickly recognize and authenticate people (Gadaleta et al., 2016). Similarly, it has been proved that the capacity of a smartphone's battery may be used to track it (Olejnik et al., 2015).

ii. Always-on personal device

Smartphones are often seen as a part of the user, and they are more inclined to see them as a trustworthy and private device that they will not trade.

These smartphones are mostly switched on, held by their owners, and linked to the internet. These smartphones are known for storing quite a number of sensitive information for a lengthy period. This causes them to be attractive to advertisers and those that mine and track data, resulting in extensive and continuing user surveillance. Consumers are also accustomed to using voice recognition, which is made possible by tools such as Alexa, Siri, Google Assistant, and Cortana. Consumers are unaware, however, that voice recognition functions are just available to gadgets that are listening throughout – at least to reply to given set of operating terms like "Hello Siri," "Alright Google," or ""Hi Cortana," and thus have direct exposure to all uttered words.

iii. There are various types of identifiers.

Smartphones include a variety of features acting as identifiers (device ID, metadata, and saved data), as well as biometrics, which mobile apps can utilize to monitor consumers (Achara et al., 2016; Kurtz et al., 2016. It was shown that just about any four applications downloaded on a consumer's smartphone are enough to identify them with a 95% chance (Achara et al., 2015). Many of those identifiers are persistent, such as behavioural biometrics, are difficult (if not impossible) to remove.

i. Connected and mobile

Smartphones could be geo-located and physically tracked. This functionality has the potential to put your privacy at compromise. In fact, a user's mobility record can reveal a great deal of potentially personal details (such as medical and banking) (Blumberg et al., 2009). Furthermore, because they are portable, they communicate with a variety of networks, some of which are dangerous, raising new security and anonymity concerns (Zou et al., 2016).

ii. *Surveillance as a possibility*

As previously stated, smartphones might be physically monitored to develop "profiles" via wireless interfaces offered by third parties.

When they connect to the internet, they may be tracked by third parties. Many third parties engage in cross-domain tracking, which combines phone consumers' physical and digital identities (Arp et al., 2017). The above type of surveillance may provide a more comprehensive picture of a consumer's activities, but it also poses new threats and privacy risks. Cross-device surveillance, in which 3rd party service providers attempt to associate a user's smartphone, and cross-app surveillance, in which an application attempts to find and surveil the other applications running on the smartphone, are becoming increasingly frequent practices that create new and serious concerns with respect to privacy. It has been demonstrated, for instance, that a subset of a user's mobile app list can predict a variety of attributes, including status of relationship, language groups, faith, nations of affinity, and so on (Seneviratne et al., 2014).

i. Physical security is limited

Smartphones are typically tiny, therefore difficult to safeguard. They are easily lost or smashed, posing a risk to data confidentiality and availability. Furthermore, several risk sources (friends, spouses, and children) may have easy accessibility to the devices, or associated features.

ii. User interfaces are limited

Tiny User Interfaces are common on smartphones (UI). This has a huge effect on privacy, openness, and security. For example, it was discovered that passwords made on smartphones are weaker (Melicher et al., 2016). Privacy notices on a smartphone are more difficult to comprehend and require extra attention. Consequently, privacy rules ought to be developed using a 'tiered' approach in which the most relevant features are stated first, with further information available if the consumer wishes to learn more.

iii. App developers' limitations

Smartphone applications are usually created by a sole person or a group of developers with scarce funds and poor knowledge of privacy and security. As a result, implementing the best data privacy-related practical solutions and strategies for smartphone application developers is challenging.

iv. Third-party software is used

Many smartphone apps are made up of a number of features produced by companies other than the app's creator. The 3rd party libraries help app creators perform analytics, such as tracking user activity, integrating them with social networking, and generating cash by displaying advertisements. Libraries may, however, gather sensitive private information for their own use in addition to the services they provide. The library's authors may be able to use this data to develop precise digital buyer personas by combining information from multiple mobile apps. For instance, a customer may offer one app access to only obtain their geolocation data while granting access to their contacts to another app. When both apps are using the same 3rd party library, the creator of the library might be able to link the two datasets (Achara et al., 2016). Furthermore, since the libraries sometimes are not open-source, analysis is challenging. As a result, it's possible for an app developer to be oblivious of the data collected by these functionalities (Vallina-Rodriguez et al., 2016).

i. Storage in the cloud

Personal identifiable information is typically stored in the cloud by mobile applications. The cloud must be safe and secure in order to prevent data leaking.

In reality, it has already been established (Leibenger et al., 2016) that the majority of apps only store user data in the cloud. This presents a new potential threat by asking the consumer to trust the cloud service in the absence of an analytical criterion from which to base a confident decision.

ii. Online Social Networks

Many smartphone apps enable consumers to share data with one another for statistical or comparison purposes (in social networks for example). The feature has the potential to reveal personally identifiable information to other users, as well as present new security and privacy risks that must be considered.

3. Research Methodology

3.1 Data Acquisition Methodology

As illustrated in Fig. 1, a four-pillar technique underpins our multilateral data privacy analysis. Data on application privacy impact is gathered and analysed from 4 different origins, designated as A1–A4. Data comes from a variety of sources, including app makers, consumer feedback, and the security of the application. This solution works with static data about every application's access to private information. Each pillar (A1 – A4) is discussed in further depth in the subsections that follow.

The list of applications studied (referred to as the appset throughout the rest of this article), as well as their functions, can be seen in Table 1. All applications studied have operations within Nigeria.

3.1.1 Permission Manifest Analysis (A1)

The Android manifest shows how to acquire intent of access to data from app developers. In this app's metadata, the developers specify the usage of sensitive privileges/permissions, which allow access to records such as phone records, address books, sensors, and geolocation tracks on smartphones.

Every Android application must come with an AndroidManifest.xml file which explains vital information about the application to Android's operating system, build tools and Google Play. Consumers used to confirm all requested rights at installation prior to Android 6.0, and they couldn't cancel them later. As a result, data access in the future was unrestricted. The data subjects were not provided any details about how often, or how much private data was gathered and sent, which is still true to some extent in the post-Marshmallow age. In Android version 6.0 and higher releases, Google introduced a permission manager system that allows users to change or revoke rights in real time. Although this was a step forward in terms of giving consumers additional privacy controls, it was ineffectual. This is mostly because most non-technical users are unfamiliar with permission request definitions (Felt et al., 2012). Furthermore, consumers may place a larger value on app use than on privacy (Solove, 2011). Many apps rely on third-party servers to transfer large amounts of data. App developers use access permissions to give hints about what private data will be gathered from a smartphone, but the privacy policy text that should serve as the foundation for consent from consumers when installing an application is sometimes hard to comprehend. As a result, determining how applications actually use personal data is challenging, making data subject risk assessment and impact assessment problematic.

After the consumer has accepted permissions during installation, an application's rights are generally assigned on an indefinite basis. The consumer will have no idea how frequently certain permissions are utilized to obtain their data. The easiest way to extract the manifest file from an Android APK file is to unzip the APK file. Figure 2 is a sample manifest file describing vital information about the application such as READ_CONTACTS permission which grants the application access to the user's contacts.

To retrieve the Android manifest.xml file, the Android APK file is first decompiled on a Linux terminal using the command in figure 3

3.1.2 Privacy Policy Analysis (A2)

Since smartphone applications deal with consumers' private information, they must abide by a variety of security and privacy regulations, such as the GDPR (EU General Data Protection Regulation, 2016) and the NDPR (Nigeria Data Protection Regulation, 2019). By law, app developers must provide users with a written privacy policy that describes how they gather and process data. As a result, privacy policies are the major source of information for users interested in knowing how an application uses their private details (Reidenberg et al., 2015). The privacy texts of the appset were investigated using keyword and semantic-based search approaches to establish the extent to which privacy policy texts are relevant to the developer's request in our analysis (in manifest). As a result, the app privacy policies were examined to assess how focused they are on app data collection tactics, such as if the purpose definition of information gathering premised on dangerous permissions is expressed explicitly inside the policy text. As the NDPR is relatively new and not fully enforced, this research was based on the GDPR.

The principles governing the processing of personal data according to the GDPR (EU General Data Protection Regulation, 2016) are:

i. Lawfulness, fairness, and transparency: There has to be a good reason for processing personal data. According to the GDPR, these reasons are:

a. When the user has given consent

b. It has to be done to make good on a contract

c. When it is absolutely necessary to fulfil a legal obligation

d. For protection of vital interests of a natural person

e. When it is a public task done in public interest

f. When you can establish the fact that you have legitimate interest, and it is not overruled by data subject's rights and interests.

ii. Purpose Limitation: This means data is only "collected for specified, explicit, and legitimate purposes".

iii. Data Minimization: Only collect the smallest amount of data you'll need to complete your purposes.

iv. Accuracy: Set up checks and balances to rectify, update, or delete data that is erroneous or incomplete.

v. Storage Limitation: You must justify the retention period for keeping the personal data that you store.

vi. Integrity and Confidentiality: You must protect the data you acquire from internal and external threats by preserving its integrity and confidentiality.

vii. Accountability: As proof of your compliance to the data processing principles, you must have appropriate measures and records in place.

3.1.3 Vulnerability Analysis (A3)

Concerns are surfacing concerning the prospect of a wide range of security vulnerabilities lying within these apps, putting a person's sensitive information at danger of being abused, as firms and developers rush to build mobile apps. Furthermore, as noted in the prior sections, these apps have access to potentially harmful permissions, meaning that they are a security risk. As a result, because these apps access, store, and share sensitive data, it's vital that they're protected against harmful attacks.

MobSF, an open-source tool was used to do static code analysis on the entire app set given in Table 1 (Mobile Security Framework, 2020). While various static code analysis tools are available, MobSF was chosen because of its prominence, adaptability, simplicity of use, and effectiveness in promptly discovering vulnerabilities in Android apps (Ibrar et al., 2017; Zhang et al., 2018)

3.1.4 User Reviews Analysis (A4)

Another source of data regarding an application's properties is user reviews on Play Store. Some of them contain privacy concerns from users. These types of complaints may indicate legitimate privacy concerns. As a result, this data was extracted from the reviews. However, such data is unorganized, and manually analyzing thousands of user reviews to learn about the privacy concerns of apps takes a long time. Sentiment Analysis and Topic Modeling are two topics of Natural Language Processing (NLP) that can aid with this, but Google Play Scraper package, a free Python package was used to retrieve the app reviews by scraping Google Play Store and storing the app's reviews (Google-play-scraper 1.0.2, 2021). Figure 4 shows the commands used to install MobSF dependencies on a Linux terminal while figure 5 shows the commands used to download and install MobSF.

4. Analysis, Experiments and Results

The case study of appset is discussed in this part (A1–A4) from four different perspectives. It goes over the findings as well as the conclusions reached. Our information was gathered throughout the months of October and November of 2021. The research's first two phases are devoted to identifying datasets that can be used in ex-ante transparency scenarios. To begin, data about apps is obtained from the Play store to establish which permissions/privileges are required prior to installation. Second, as mentioned in 3.1.2, the application's privacy policy texts are obtained and reviewed to determine whether they are consistent with the application's actual intent as expressed in the manifest file. The sources of data that are obtainable via ex-post transparency scenarios are the focus of the 3rd and 4th phases. 10 Nigerian finance apps (referred to as the appset) were picked from the top Google Play store search results in the finance category and

installed to examine their permission access requests. The subsections following illustrate the analysis phases A1–A4 from Fig. 1.

4.1 Step A1: Analysis of Permission Manifest

Permissions are used by Android apps to gain access to the device's resources. Depending on the resource type, users' consent may be required. Android defines four categories of permissions: normal, dangerous, signature, and SignatureOrSystem (Google Developers, 2021). Permissions at the normal level allow support for small-impact components and are enabled at installation of the package that requires them. Permissions that are regarded unsafe could allow access to dangerous resources. In this case, the consumer is explicitly requested to provide permissions. During the installation process, signature level rights are provided, but only if the application seeking to utilize the permission/privilege is signed by the exact certificate as that of the application that defines the permission/privilege.

SignatureOrSystem permissions provide both packages with the same creator and packages installed on the system access to specified resources. The AndroidManifest.xml file stores info relating to the application. This information could the app's name, its creator, logo, and/or summary, as well as permissions that provide access data on the smartphone, like call logs, phonebooks, and SMS. By collecting and analyzing app developers' data access intentions from the Android app's manifest, permission request trends, suspicious permission requests, and differences/resemblances in applications' permission requests issued by EU and non-EU authorities were examined.

4.2 Detected Permission Requests

There was a total of 54 different permission requests found. 30 permissions (55.6%) fall into the Normal category, 22 (40.7%) fall into the Dangerous category, 1 (1.9%) falls into the SignatureOrSystem category, and 1 (1.9%) falls into the SignatureOrSystem category. Permission requests can jeopardize a user's privacy and be easily abused to profile them. Our primary emphasis, however, is on permission requests that could be dangerous. For example, the GET TASKS permissions, that are sought by two different applications, can reveal private information regarding the user's app usage. As a result, the vendor of the appset has access to ActivityManager. RecentTaskInfo can be leveraged to look up tasks that the consumer has begun or visited lately. Likewise, SYSTEM ALERT WINDOW is a dangerous permission that could disclose private details by creating overlays which deceive consumers by concealing specific sections of the display whilst making the overlaying region responsive (Alepis et al., 2019). It's also worth noting that combining these permission requests can result in useful user data (Fritsch et al., 2017; Momen et al., 2020).

4.2.1 Step A2: Privacy Policy Analysis

Attention was paid to the appset privacy policies and compliance with 13 basic legal standards, the degree with which the privacy policies of the appset align with the developer's demand in manifest file, and finally, the disparities/resemblances in app privacy policies written by EU and Non-EU organisations with regards to capturing foundational privacy principles (Hatamian, 2020). The findings of the data set's privacy policy study are summarized in Table 3. The result demonstrates how effectively an app adheres to privacy policy criteria. The LCredit app, according to the findings, meets the most requirements (11), next is GTWorld (9), Palm Pay (8), and Kuda Bank (7 principles). Interestingly, several of these apps (six apps) fail to fulfil any privacy policy criteria. This is because they do not have privacy policy texts accessible on their platforms, which is the case with 5 of the apps (Fair Money, Fast Money, Go Cash, iMoney, and UBA Mobile), or because, as is the case with one of them, they contain highly generic content that do not cover the data gathering and sharing methods of applications, but superfluous information (Chipper Cash). Principle fulfilment (✓), non-fulfilment (N), and absence of privacy policy text or very generic/outdated text (X) were differentiated in Table 3 below.

4.2.2 The transparency of Dangerous permission in privacy policy texts

According to privacy-by-design as well as the GDPR, transparency is a key data protection strategy (Cavoukian, 2010). As a result, it's critical to investigate how well apps address this need. A list of key terms (e.g., GPS, geolocation, accessibility, precision, approximation, tracking, motion, contact, SMS, etc.) was created matching each dangerous permission (e.g., ACCESS FINE LOCATION) defined by Android to conduct a manual inspection for this study (Google Developers, 2021). As seen in Table , none of the apps in the appset completely justify the use of potentially dangerous permission queries. Additionally, 30% (3 apps) of the appset provide only a fuzzy explanation as to why they require some dangerous permission. This implies that as many as two-thirds of the applications (7 applications or 70 percent) didn't state the necessity for dangerous permissions or had no privacy policies applied. Permission usage transparency (green) was differentiated from permission usage non-transparency (orange). Apps without privacy policies are marked (red).

4.3 Step A3: Vulnerability Analysis

An investigation is needed to uncover any security vulnerabilities that might be abused by hackers in each of these apps so as to restrict any possible danger that could arise from abusing a vulnerability. Consequently, MobSF was utilized to do static analysis on all the apps in Table 1. Static analysis uses data from the application's APK, that contains the manifest file and generated code, to assess vulnerability (Knorr et al., 2015). The permission evaluation will not be focused on in this example because it has already been adequately addressed in the preceding sections.

4.3.1 Detected Vulnerabilities

Tables 5 and 6 highlight the security problems uncovered during the MobSF assessment of the APK files. As shown in Table 5, the framework lists, and scores several of the most known vulnerabilities in the app set using CVSS V2, OWASP Top 10 Mobile Risks, and CWE, where the vulnerabilities observed have been picked and paired against the standards. Whereas every app has a huge number of vulnerabilities, issues with CVSS V2 varying from medium to high was selected, per the MobSF report. CWE10 is a source of comparison for prevalent software / hardware vulnerabilities, and also a benchmark for vulnerability detection whereas CVSS V2 is a public framework for conveying the nature and impact of technology vulnerabilities (Mell et al., 2007). OWASP Top 10 on the other hand, gives a shortlist of ten mobile security concerns, raising awareness about the importance of software application security (Qian et al., 2018). The framework also delivers the average CVSS score and App Security Score for the app.

In Table 6, the vulnerabilities indicate several flaws that could be considered significant with regards to data privacy. For instance, 90 percent of the examined applications have possibly hard-coded private information such as credentials, keys, and so on, which is classified as high severity by CVSS version 2 (7.4 score). The usage of hard-coded private data (CWE-312) poses a huge risk since it allows any malicious user to bypass software administrator-set authentication.

The source code of the appset can be decompiled to obtain this information. As a result, any malicious user with this detail can obtain access to even more sensitive data, such as financial information and location data.

According to additional data analysis in Table 6, 50% of the applications utilize SQLite Database that conduct raw SQL queries. Potentially malicious user input in raw SQL queries in the appset could result to a local SQL Injection. Furthermore, the apps frequently employ an unencrypted SQLite database. As a result, a hacker with direct contact to the smartphone or a malicious application having root privileges to the smartphone have access to important information. Furthermore, the absence of encryption may result in privacy violations and non-compliance with data privacy laws and regulations. Furthermore, two apps (Chipper Cash and LCredit) have an insecure SSL implementation, resulting in insecure communication. The apps may be subject to MITM attacks, as previously stated, undermining the confidentiality and integrity goals of information security (Jain et al., 2012).

4.4 Step A4: User Reviews Analysis

User reviews for apps are another source of information for spotting privacy risks. It helps in mapping the identified dangers to the related instances while considering the individual's privacy preferences. For the appset, gathered user reviews were gathered from the app market. Using the process flow in Figure 3, a data set of user reviews corresponding to the appset from the Google Play app store were collected.

4.4.1 Analysis of Privacy-Related Complaints

Our aim was to identify what consumers were saying about application privacy. It was important to get this data first, then assess the intricacy of the privacy-related remarks (to uncover probable privacy concerns from apps based on user evaluations). A significant amount of user reviews was discovered to relate to privacy and security. Table 7 presents some samples of different categories of user reviews and their corresponding threat (T) to better understand categorized user reviews while Table 8 shows the privacy threats that have been identified for each app (represents the threats that have been identified during the analysis period).

4.5 Synthesis of Analysis

The collected data was fused with a grading algorithm to create an overall app privacy impact analysis. When fusing the results, the different data sets (A1 – A4) were treated as contributing equally to privacy impact. For that reason, a point is added to an app for each violation of the data sets and the total points gives us the cumulative privacy impact score. As defined in Section 3.1.2, the inadequacies in the privacy policies were evaluated. In addition, as a result of our vulnerability analysis, security flaws were uncovered. Table 9 provides our conclusion, which is a ranking of the app set based on the app privacy impact analysis. It was conceded that the total number of privacy impact violation points lack certain significant factors, such as reliance on personal context, subjectivity of perception of risk, real-time communication with apps, personal preferences, and so on, which were out of scope for this study due to the immense complexity of adding valuable weights to the impact score, and therefore can be considered a limitation.

Because the data from four different sources are combined into a total privacy impact score, as shown in Table 9, an overall comparison may be established by sorting the impact scores from highest to lowest, which corresponds to the highest to lowest privacy impact. As a result, an app has the potential to amass N number of impact score (1 point for seeking each dangerous permission, 1 point for failing to specify purposes of each dangerous permission in a privacy policy, 1 point for each security vulnerability discovered through vulnerability analysis, and 1 point for each threat discovered through user review analysis). iMoney, for example, has a privacy impact score of 36 as seen in Table 9 (12 for the total number of requested permissions, 12 for missing justifications, 10 security vulnerabilities, and 2 identified threats from user review analysis).

According to our analyses, an app is more privacy-preserving if it seeks fewer dangerous permissions, has less conflict between the manifest and available clarification in the policy document, has fewer security vulnerabilities, and faces fewer threats from user review analysis. Fig 4 shows the ranking of apps by their privacy score impact. iMoney is the most privacy-invasive app with the privacy impact score of 36, whereas GTWorld is the least privacy-invasive app with the privacy impact score of 20.

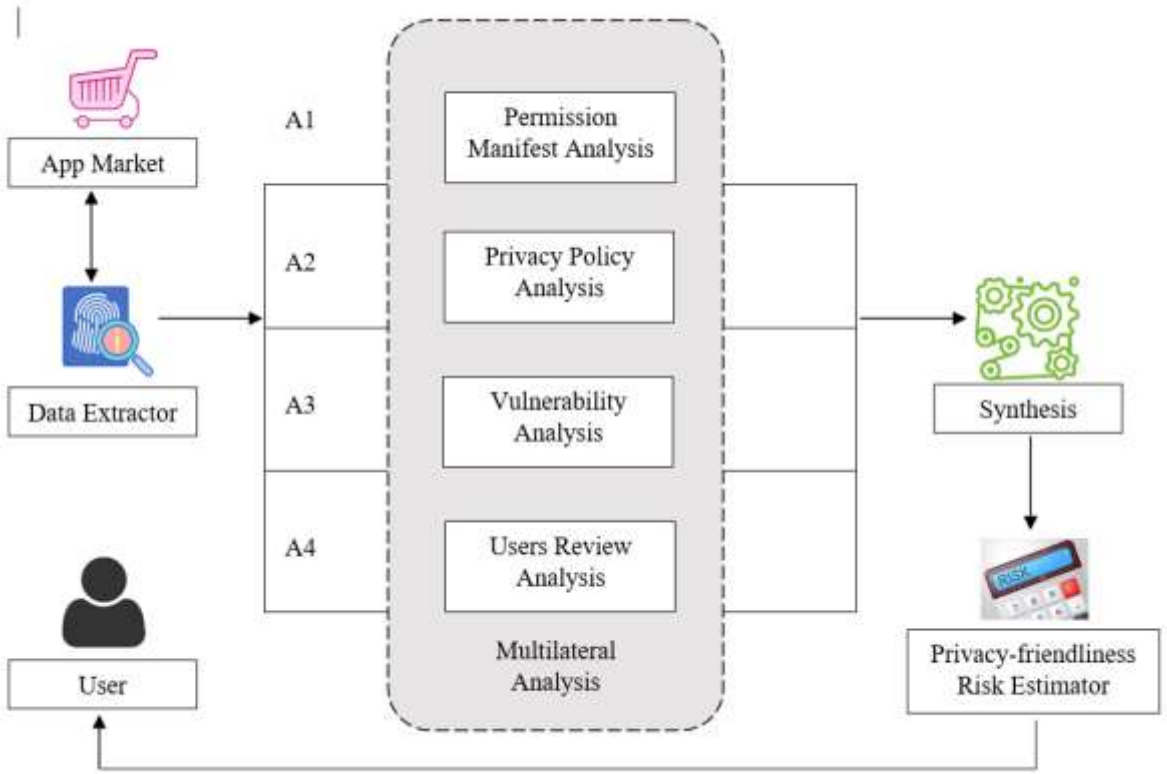

Figure 1: A high-level summary of our methodology to privacy analysis

```
apktool d path/to/app.apk
```

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.myfirstapplication">

    <uses-permission android:name="android.permission.READ_CONTACTS"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <meta-data
            android:name="preloaded_fonts"
            android:resource="@array/preloaded_fonts" />
    </application>

</manifest>
```

Figure 2: Sample Android manifest.xml file

```
apktool d path/to/app.apk
```

Figure 1: A Decompiling APK file from a Linux terminal

```
sudo apt-get install git
```

```
sudo apt-get install python3.8
```

```
sudo apt-get install openjdk-8-jdk
```

```
sudo python3-dev python3-venv python3-pip build-essential libffi-dev libssl-dev libxml2-dev libxsltl-dev libjpeg8-dev xliblg-dev wkhtmltopdf
```

Figure 2: Commands to install MobSF dependencies on Linux terminal

```
git clone https://github.com/MobSF/Mobile-Security-Framework-MobSF.git
```

```
cd Mobile-Security-Framework-MobSF
```

```
./setup.sh
```

Figure 3: Commands to download and install MobSF on a Linux terminal

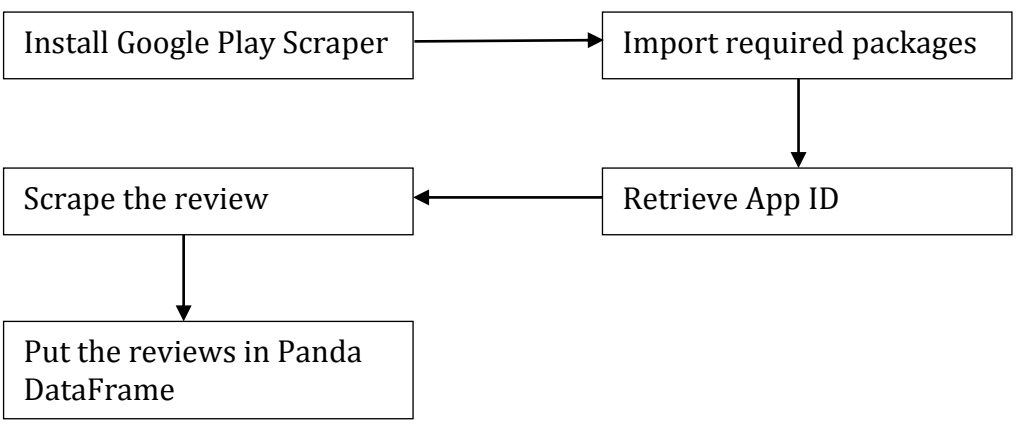

Figure 6: User Review Analysis Process Flow

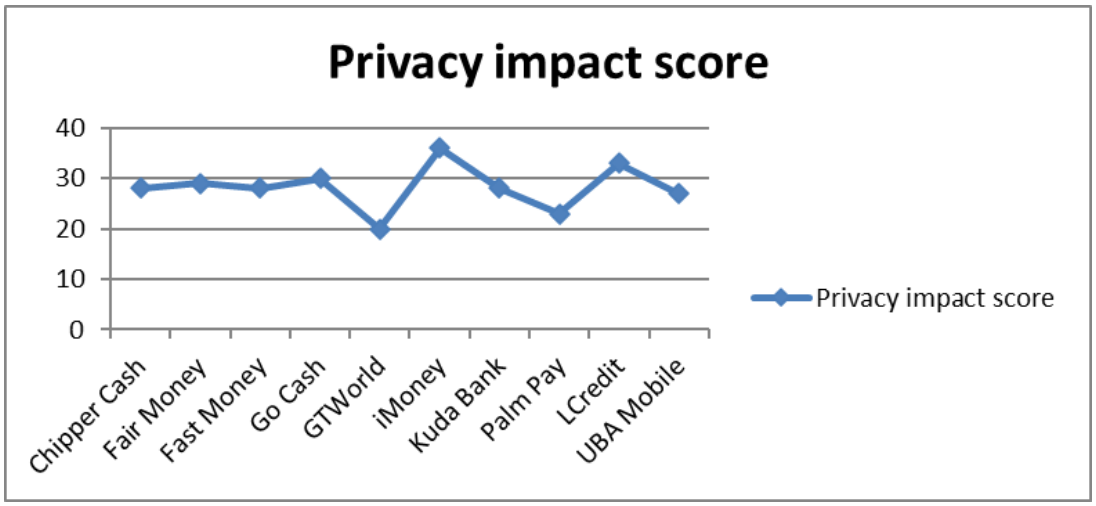

Figure 7: Visual comparison of apps' privacy impact

Table 1: Collected data set: apps, country of origin and functionality

#	Country	Appset	How it functions
1	Nigeria	Chipper Cash	Send and receive money the fast and easy way. Transfer funds across town and across Africa, right from your mobile phone.
2	Nigeria	Fair Money	Instant loans & more from a Microfinance digital bank. Loan amounts range between ₦1,500 to ₦1,000,000 with repayment periods from 61 days to 18 months at monthly interest rates that range from 2.5% to 30% (APRs from 30% to 260%).
3	Nigeria	Fast Money	Fast Money is a quick and online Android mobile loan app, available in Nigeria. Download and follow a simple application process to get an instant loan in a few minutes. Loan up to NGN 300,000, without any mortgage, low interest rate, long time repayment, quick and safe.
4	Nigeria	Go Cash	A quick and online loan app in Nigeria. Loan up to NGN 300,000, without any mortgage, low interest rate, long time repayment, quick and safe.
5	Nigeria	GTWorld	GTWorld is a mobile banking app for GTBank customers for managing bank accounts, bills payment, transfers etc.
6	Nigeria	iMoney	A quick and online loan app in Nigeria. Loan up to NGN 500,000. Loan tenor is between 91 to 180 days at an interest rate of 12 to 36% APR. Service fee is 0%.
7	Nigeria	Kuda Bank	Kuda is a free digital bank with a microfinance banking license from the Central Bank of Nigeria. With a free debit card, zero maintenance fee and free transfers, Kuda gives Nigerians an easy way to manage their money by using mobile banking apps on their smartphones.
8	Nigeria	Palm Pay	Palm Pay is the rewarding way to make payments. The simple and secure app for sending cash quickly between friends and paying for airtime, bills and more in Nigeria. Earn Palm Points and coupons as you spend and use them to discount your transactions.
9	Nigeria	LCredit	Borrow money online with LCredit Loan app. Instant loan with a flexible repayment plan and favourable interest rate.
10	Nigeria	UBA Mobile	UBA Mobile is a mobile banking app for UBA customers for managing bank accounts, bills payment, transfers etc.

Table 2: Steps taken to retrieve user reviews of an application on Google Play Store using Google Play Scraper

S/N	Steps	Instructions
1	From a Linux terminal, download and install Google Play Scraper package	`pip install google-play-scraper`

2	Import required packages	`from google_play_scraper import app`
		`import pandas as pd`
		`import numpy as np`

3	Retrieve the App Id from Google Play Store	`https://play.google.com/store/apps/details?id=com.kudabank.app`

4	Scrape the review	`from google_play_scraper import Sort, reviews_all`
		`ng_reviews = reviews_all(`
		`'com.kudabank.app',`
		`sleep_milliseconds=0, # defaults to 0`
		`lang='en', # defaults to 'en'`
		`country='ng', # defaults to 'us'`
		`sort=Sort.NEWEST, # defaults to Sort.MOST_RELEVANT`
		`)`

5	Put the reviews into Panda DataFrame	`df_kuda = pd.DataFrame(np.array(ng_reviews),columns=['review'])`
		`df_kuda = df_kuda.join(pd.DataFrame(df_kuda.pop('review').tolist()))`
		`df_busu.head()`

Table 3: Details of privacy policy principles fulfilment per app

#	Privacy Policy Principles	Chipper Cash	Fair Money	Fast Money	Go Cash	GTWorld	iMoney	Kuda Bank	LCredit	Palm Pay	UBA Mobile
1	Lawfulness, Fairness & Transparency	N	X	X	X	□	X	□	□	□	X
2	Accuracy	N	X	X	X	□	X	N	N	N	X
3	3rd Party Sharing	N	X	X	X	□	X	□	□	□	X
4	3rd Country Sharing	N	X	X	X	N	X	N	□	N	X
5	Integrity & Confidentiality	N	X	X	X	□	X	N	□	□	X
6	Data Retention	N	X	X	X	□	X	□	□	□	X
7	User's Control	N	X	X	X	□	X	□	□	□	X
8	Privacy Policy Changes	N	X	X	X	N	X	□	□	□	X
9	Privacy Breach Notice	N	X	X	X	N	X	N	N	N	X
10	Minimization	N	X	X	X	N	X	N	□	N	X
11	Purpose Limitation	N	X	X	X	□	X	□	□	□	X
12	Contact Information	N	X	X	X	□	X	□	□	□	X
13	Children Protection	N	X	X	X	□	X	N	□	N	X

Table 4: Details of dangerous permission usage transparency in privacy policy text of our appset

#	**Permission**	**Chipper Cash**	**Fair Money**	**Fast Money**	**Go Cash**	**GTWorld**	**iMoney**	**Kuda Bank**	**LCredit**	**Palm Pay**	**UBA Mobile**
1	GET_TASKS								□	□	
2	ACCESS_COARSE_LOCATION	□			□	□	□	□	□	□	□
3	ACCESS_FINE_LOCATION	□	□			□	□	□	□	□	□
4	WRITE_EXTERNAL_STORAGE	□	□	□	□	□	□	□	□	□	□
5	MANAGE_ACCOUNTS								□		
6	WRITE_SETTINGS								□		
7	READ_CONTACTS	□	□	□	□	□	□	□	□	□	□
8	READ_SMS		□				□		□		
9	READ_PHONE_STATE	□	□	□	□		□		□	□	□
10	READ_CALENDAR		□	□	□				□	□	
11	WRITE_CALENDAR			□	□				□	□	
12	CAMERA	□	□	□	□	□	□	□	□	□	□
13	GET_ACCOUNTS		□	□	□	□					□
14	WRITE_CONTACTS			□	□	□	□				□
15	READ_EXTERNAL_STORAGE		□	□	□	□	□	□		□	□
16	ACCESS_BACKGROUND_LOCATION							□			
17	CALL_PHONE							□			□
18	RECORD_AUDIO	□						□			
19	REQUEST_INSTALL_PACKAGES						□			□	
20	SYSTEM_ALERT_WINDOW	□					□			□	
21	READ_PROFILE	□									
22	READ_CALL_LOG						□				

Table 5: Summary of the selected vulnerabilities within the finance app set

APP	**Issue**	**Standards**		
		CVSS V2	CWE	OWASP
LCredit	Insecure implementation of SSL. Trusting all the certificates or accepting self-signed certificates is a critical security hole. This application is vulnerable to MITM attacks	7.4 (High)	CWE-295: Improper Certification Validation	M3 – Insecure Communication
Fast Money	Files may contain hardcoded sensitive information like usernames, passwords, keys etc.	7.4 (High)	CWE-312: Cleartext Storage of Sensitive Information	M9 – Reverse Engineering
Fair Money	Insecure WebView implementation. Execution of user-controlled code in WebView is a critical security hole.	8.8 (High)	CWE-749: Exposed Dangerous Method or Function	M1 – Improper Platform Usage
Kuda Bank	The App uses an insecure random number generator	7.5 (High)	CWE-330: Use of Insufficiently Random Values	M5 – Insufficient Cryptography
Go Cash	App can read/write to external storage. Any app can read data written to external storage	5.5 (Medium)	CWE-276: Incorrect Default Permissions	M2 – Insecure Data Storage
UBA Mobile Banking	The App uses the encryption mode CBC with PKCS5/PKCS7 padding. This configuration is vulnerable to padding oracle attacks	7.4 (High)	CWE-649: Reliance on Obfuscation or Encryption of Security-Relevant Inputs without Integrity Checking	M5 – Insufficient Cryptography
Chipper Cash	App uses SQLite Database and execute raw SQL query. Untrusted user input in raw SQL queries can cause SQL Injection. Also, sensitive information should be encrypted and written to the database.	5.9 (Medium)	CWE-89: Improper Neutralization of Special Elements used in an SQL Command (“SQL Injection”)	M7 – Client Code Quality
iMoney	Insecure WebView Implementation. WebView ignores SSL Certificate errors and accept any SSL Certificate. This application is vulnerable to MITM attacks	7.4 (High)	CWE-295: Improper Certificate Validation	M3 – Insecure Communication
GTWorld	The App uses the encryption mode CBC with PKCS5/PKCS7 padding. This configuration is vulnerable to padding oracle attacks	7.4 (High)	CWE-649: Reliance on Obfuscation or Encryption of Security-Relevant Inputs without Integrity Checking	M5 – Insufficient Cryptography

Table 6: An overview of the results from mobSF static analysis

Vulnerabilities	**Percentage of apps with the vulnerabilities**
The App logs information. Sensitive information should never be logged	80%
The App uses an insecure Random Number Generator	90%
MD5 is a weak hash known to have hash collisions	60%
App creates temp file. Sensitive information should never be written into a temp file	60%
Insecure WebView Implementation. Execution of user-controlled code in WebView is a critical Security Hole	70%
SHA-1 is a weak hash known to have hash collisions	70%
Files may contain hardcoded sensitive information like usernames, passwords, keys etc.	90%
App can read/write to External Storage. Any App can read data written to External Storage	80%
App uses SQLite Database and execute raw SQL query. Untrusted user input in raw SQL queries can cause SQL Injection. Also, sensitive information should be encrypted and written to the database	50%
This App may request root (Super User) privileges	30%
The App uses the encryption mode CBC with PKCS5/PKCS7 padding. This configuration is vulnerable to padding oracle attacks	40%
Insecure Implementation of SSL. Trusting all the certificates or accepting self-signed certificates is a critical Security Hole. This application is vulnerable to MITM attacks	20%
Insecure WebView Implementation. WebView ignores SSL Certificate errors and accept any SSL Certificate. This application is vulnerable to MITM attacks	20%
The App uses ECB mode in Cryptographic encryption algorithm. ECB mode is known to be weak as it results in the same ciphertext for identical blocks of plaintext	20%
Remote WebView debugging is enabled	10%

Table 7: An example of classified user reviews

#	Sample user review	T
1	This is not a friendly App, do not loan from them, 2days default, they will be threatening your life and that of your contact list. Even if you have a good record from the beginning with them. My experience with them is bad. Very bad.. I understand why Google play took them down before.	T1
2	This app is a very stupid app, it still preys on your private information online, imagine attempting to pay back a loan countless times and having it rejected untill it got 4 days overdue... Then getting a message from a close friend of mine saying he saw a message on his phone calling me a fraudster, a dupe. For what reason? That is the upmost level of embarrassment i cannot tolerate, please this app should be reviewed more and scrape off	T2
3	Please don't download this app,you will regret it, This is the worst app I've ever seen....my payment was due before I started having issues with my bank so I went to the bank to rectify the issue i was told it's going to take 8-9 days before it will be solved....they started calling and threatening me, before I knew it they sent my pictures to my contacts I was so embarrassed...my experience was hell.....please don't download this app	T3
4	This app, company is one of the worst app ever. I think its going to be wise enough if the government can look for the owner of this company or if they can be sued. Once you borrowed from them, they will start messaging those on your contact a day before the due date of your payment. First, their staff is something we can't talk about because they all lack manner. Therefore, friend i will advice you if truly you don't want to tarnish your image.	T4
5	Afterall the good experience I had at first, the 2nd time I took a loan from the app, due to some unavoidable issues which I spoke with some agents who called, you guys had to still call few of my relatives especially my mum who had an issue and was hypertensive; thank God am almost done paying the loan, am uninstalling the app immediately when am done! No human sympathy at all, not everyone is dubious, mind you	T5
6	I have given my Sister this my device but I find it difficult to do factory reset just because of Palm play security plugin as my device administrator. Please erase on my data from your data base so that my Sister can use the phone.	T6
7	App is great until you use their OKCards which installs a security plugin that takes control of the phone and sends a pop-up whenever you open any app, try to make calls or do anything on the phone. Ed: Recently got a mail requesting to join and request for the OK card, but on the app, nothing happens and received a prompt that am not qualified! Why?!	T7
8	The worst app ever,I was unable to pay on my due date because I had issues with my bank,the following day I gave money to someone to help me make the payment, before I know it I saw message on my phone and a friend phone calling me names,and I immediately told the person that was supposed to make payment on my behalf to holdon,since they already sent out bad messages to tarnish my name,then what's the point in paying the money? Sending messages doesn't work anymore,your company will keep loosing	T8
9	Go cash is the worst and most impatient loan app. Barely a week of late payment and the useless customer care person has already sent messages to all my WhatsApp contacts despite the fact that I asked for some more time due to circumstances beyond me at the moment. You had better correct this error else I won't pay your 10k or so. Haba Google better destroy this app because they keep trying to defame people's character. Gocash is a SCAM guys. Please runaway from this animalistic loan app.	T9
10	Very useless and heartless loan app, just a day of default, all ur contact both home and abroad would be called, and ur picture would be displayed as a criminal, Gosh!! This is the worst loan app ever, God forbid! If u value ur peace of mind , kindly stay away from this heartless loan app called IMONEY.. a word is enough for The Wise!!!	T10
11	This is a useless app,they send messages to your contacts on your due date,run from them is you want to mention your integrity,bunch of thrives. With their outrageous interest they will send messages to your contacts on the very due date	T11

Table 8: Our apps set with their respective identified privacy threats (shown by)

#	App name	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11
1	Chipper Cash	X	X	X	X	X	X	X	X	X	X	X
2	Fair Money	X	X	X	X	□	X	X	X	X	X	X
3	Fast Money	X	X	□	□	X	X	X	X	X	X	X
4	Go Cash	X	X	X	X	X	X	X	□	□	X	X
5	GTWorld	X	X	X	X	X	X	X	X	X	X	X
6	iMoney	X	X	X	X	X	X	X	X	X	□	□
7	Kuda Bank	X	X	X	X	X	X	X	X	X	X	X
8	Palm Pay	X	X	X	X	X	□	□	X	X	X	X
9	LCredit	□	□	X	X	X	X	X	X	X	X	X
10	UBA Mobile	X	X	X	X	X	X	X	X	X	X	X

Table 9: Summary of the synthesized results from a multilateral privacy analysis, ordered by privacy impact score

App	Privacy impact score	Dangerous Permission Groups Requested	Missing Justification in Privacy Policy	Vulnerability Analysis result	Identified Threats from User Reviews
Chipper Cash	28	9	9	10	0
Fair Money	29	9	9	10	1
Fast Money	28	9	9	8	2
Go Cash	30	10	10	8	2
GTWorld	20 (lowest)	8	8	4	0
iMoney	36 (highest)	12	12	10	2
Kuda Bank	28	9	8	11	0
Palm Pay	23	12	9	0	2
LCredit	33	12	8	11	2
UBA Mobile	27	10	10	7	0

5. Concluding Remarks

Our analysis showed of 10 popular Nigerian finance apps with over 1 million downloads on the Android app store showed that apps are far away from adhering to Data Privacy best standards. Our multilateral analysis allows the evaluation and comparison of privacy implication of an app from four different angles: a) a comparison of app resource requirements, b) a review of those requirements based on corresponding privacy policies, c) vulnerability assessment of the app and d) an assessment of user's privacy concerns. Our findings discovered gaps between the privacy policies and the privilege requests, and in addition, discovered suspicious app behaviour of some of the apps in the appset. Based on this preliminary evidence, it can be concluded that this method has potential in providing transparency about app's actual intentions to consume personal data to both end users and regulators as Table 8 and Fig. 4 both show that there are clear differences between app's access request to data and app vendors' declaration about their data access intentions in the privacy policy. Our results can therefore be used as a base for personal decision-making about continued or future app use. Our future work will look into Permission Usage Analysis, which will collect and analyse access logs from installed apps that are idle and look for permission-access activities.

Conflict of Interest

There is no conflict of interest whatsoever.

Author Contribution

Conception: [KEO]

Design: [KEO, CYB]

Execution: [KEO, CYB]

Interpretation: [KEO, CYB]

Writing the paper: [KEO, CYB]

References

Achara, J.P., Roca, V., Castelluccia, C., and Francillon, A. (2016). MobileAppScrutinator: A Simple yet Efficient Dynamic Analysis Approach for Detecting Privacy Leaks across Mobile OSs. https://doi.org/10.48550/arXiv.1605.08357

Achara, J. P., Acs, G., and Castelluccia, C. (2015). On the Unicity of Smartphone Applications, In Proceedings of the 14th ACM Workshop on Privacy in the Electronic Society (WPES '15). Association for Computing Machinery, New York, NY, USA, 27–36. https://doi.org/10.1145/2808138.2808146

Alepis, E., Patsakis, C. (2019). Unravelling Security Issues of Runtime Permissions in Android, Journal of Hardware and Systems Security (3); 45–63. https://doi.org/10.1007/s41635-018-0053-2

Arp, D., Quiring, E., Wressneger, C., and Rieck, K. (2017). Privacy Threats through Ultrasonic Side Channels on Mobile Devices, IEEE European Symposium on Security and Privacy (EuroS&P) ;35-47. https://doi.org/10.1109/EuroSP.2017.33

Chin, E., Felt, A.P., Sekar, V., and Wagner, D.A. (2012). Measuring user confidence in smartphone security and privacy. In Proceedings of the Eighth Symposium on Usable Privacy and Security (SOUPS '12). Association for Computing Machinery, New York, NY, USA (Article 1); 1–16. https://doi.org/10.1145/2335356.2335358

Benenson, Z., Kroll-Peters, O., and Krupp, M. (2012). Attitudes to IT Security when Using a Smartphone, Federated Conference on Computer Science and Information Systems (FedCSIS); 1179–1183.

Blumberg, A.J. and Eckersley, P. (2009). On locational privacy, and how to avoid losing it forever, Electronic Frontier Foundation. [cited 2021 June 22]. Available from: https://www.eff.org/files/eff-locational-privacy.pdf.

Book, T., Pridgen, A., and Wallach, D. S. (2013) Longitudinal analysis of Android ad library permissions. In Mobile Security Technologies (MoST), San Francisco, CA. https://doi.org/10.48550/arXiv.1303.0857

Cavoukian, A. (2010). Privacy by design: the definitive workshop. A foreword by Ann Cavoukian, Ph.D, vol 3 (2); 247–251. Proceedings of the Eighth Symposium on Usable Privacy and Security (SOUPS '12). Association for Computing Machinery, New York, NY, USA (Article 1); 1–16. https://doi.org/10.1007/s12394-010-0062-y

Blumberg, A.J. and Eckersley, P. (2009). On locational privacy, and how to avoid losing it forever, Electronic Frontier Foundation. [cited 2021 June 22]. Available from: https://www.eff.org/files/eff-locational-privacy.pdf.

Egele, M., Brumley, D., Fratantonio, Y., and Kruegel, C. (2013). An empirical study of cryptographic misuse in android applications. In Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security (CCS '13), Association for Computing Machinery, New York, NY, USA; 73–84. https://doi.org/10.1145/2508859.2516693

Enck, W., Gilbert, P., Chun, B. G., Cox, L. P., Jung, J., McDaniel, P., and Sheth, A. N. (2019). TaintDroid: An information-flow tracking system for realtime privacy monitoring on smartphones, In Proceedings of the 9th USENIX Symposium on Operating Systems Design and Implementation; 393-407. https://doi.org/10.1145/2494522

Enck, W., Octeau, D., McDaniel, P., and Chaudhuri, S. (2011). A Study of Android Application Security. Proceedings of the 20th USENIX Security Symposium, San Francisco, CA; 10-12.

Enck, W., Ongtang, M., Mcdaniel, P. (2009). On lightweight mobile phone application certification, In Proceedings of the 16th ACM conference on Computer and communications security (CCS '09), Association for Computing Machinery, New York, NY, USA; 235–245. https://doi.org/10.1145/1653662.1653691

EU General Data Protection Regulation; 2016 [cited 2021 Aug 8]. Available from: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A02016R0679-20160504.

Fahl, S., Harbach, M., Muders, T., Baumgärtner, L., Freisleben, B., and Smith, M. (2012). Why eve and mallory love android: an analysis of android SSL (in)security, In Proceedings of the 2012 ACM conference on Computer and communications security (CCS '12), Association for Computing Machinery, New York, NY, USA; 50–61. https://doi.org/10.1145/2382196.2382205

Felt, A. P., Egelman, S., and Wagner, D. (2012). I've got 99 problems, but vibration ain't one: a survey of smartphone users' concerns, In Proceedings of the second ACM workshop on Security and privacy in smartphones and mobile devices (SPSM '12), Association for Computing Machinery, New York, NY, USA; 33–44. https://doi.org/10.1145/2381934.2381943

Felt, A. P., Ha, E., Egelman, S., Haney, A., Chin, E., and Wagner, D. (2012). Android permissions: user attention, comprehension, and behavior. In Proceedings of the Eighth Symposium on Usable Privacy and Security (SOUPS '12). Association for Computing Machinery, New York, NY, USA, Article 3;1–14. https://doi.org/10.1145/2335356.2335360

Fife, E., and Orjuela, J. (2012). The Privacy Calculus: Mobile Apps and User Perceptions of Privacy and Security, International Journal of Engineering Business Management. 5(6); 7. https://doi.org/10.5772%2F51645

Fritsch, L. and Momen, N. (2017). Derived Partial Identities Generated from App Permissions, In: Fritsch, L., Roßnagel, H. and Hühnlein, D. (Hrsg.), Open Identity Summit 2017, Gesellschaft für Informatik, Bonn; 117-130.

Fritsch, L., and Abie, H. (2008). Towards a Research Road Map for the Management of Privacy Risks in Information Systems, In: Alkassar, A. & Siekmann, J. (Hrsg.), SICHERHEIT 2008 – Sicherheit, Schutz und Zuverlässigkeit. Beiträge der 4. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI). Bonn: Gesellschaft für Informatik e. V; 1-15.

Gadaleta, M., and Rossi, M. (2018). IDNet: Smartphone-based Gait Recognition with Convolutional Neural Networks; 25-37. https://doi.org/10.48550/arXiv.1606.03238

Google Developers (2021). Permissions on Android; [cited 2021 Oct 9]. Available from: https://developer.android.com/guide/topics/permissions/overview/.

Google-play-scraper 1.0.2; 2021 [cited 2021 Nov 8]. Available from: https://pypi.org/project/google-play-scraper/

Habib, S.M., Alexopoulos, N., Islam, M.M., Heider, J., Marsh, S., and Mühlhäuser, M. (2018). Trust4App: Automating Trustworthiness Assessment of Mobile Applications, 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE); 124-135. https://doi.org/10.1109/TrustCom%2FBigDataSE.2018.00029

Hatamian, M. (2020). Engineering Privacy in Smartphone Apps: A Technical Guideline Catalog for App Developers, in IEEE Access, vol. 8; 35429-35445. https://doi.org/10.1109/ACCESS.2020.2974911

Hatamian, M., Serna, J., Rannenberg, K., and Igler, B. (2017). FAIR: Fuzzy Alarming Index Rule for Privacy Analysis in Smartphone Apps, In J. Lopez, S. Fischer-Hübner, & C. Lambrinoudakis (Eds.), Trust, Privacy and Security in Digital Business: 14th International Conference, TrustBus 2017, Lyon, France, Vol. 10442; pp. 3-18. https://doi.org/10.1007/978-3-319-64483-7_1

Ibrar F., Saleem H., Castle S., Malik M. Z. (2017). A Study of Static Analysis Tools to Detect Vulnerabilities of Branchless Banking Applications in Developing Countries, In Proceedings of the Ninth International Conference on Information and Communication Technologies and Development (ICTD '17), Association for Computing Machinery, New York, NY, USA, Article 30; 1–5.

Isaak, J. and Hanna, M. J. (2018). User Data Privacy: Facebook, Cambridge Analytica, and Privacy Protection, in Computer, vol. 51 (8); 56-59. https://doi.org/10.1109/MC.2018.3191268

Jain, A.K. and Shanbhag, D. (2012). Addressing Security and Privacy Risks in Mobile Applications. IT Professional, 14; 28-33. https://doi.org/10.1109/MITP.2012.72

Knorr K., Aspinall D., and Wolters M. (2015). On the privacy, security and safety of blood pressure and diabetes apps. In: IFIP International Information Security and Privacy Conference. Springer; 571–584. https://doi.org/10.1007/978-3-319-18467-8_38

Kuehnhausen, M., and Frost, V.S. (2013). Trusting smartphone Apps? To install or not to install, that is the question. 2013 IEEE International Multi-Disciplinary Conference on Cognitive Methods in Situation Awareness and Decision Support (CogSIMA); 30-37. https://doi.org/10.1109/CogSIMA.2013.6523820

Kurtz, A., Gascon, H., Becker, T., Rieck, K. and Freiling, F. (2015). Fingerprinting Mobile Devices Using Personalized Configurations, Proceedings on Privacy Enhancing Technologies, Vol.2016 (Issue 1); 4-19. http://dx.doi.org/10.1515/popets-2015-0027

Leibenger, D., Möllers, F., Petrlic, A., Petrlic, R. and Sorge, C. (2016). Privacy Challenges in the Quantified Self Movement – An EU Perspective, Proceedings on Privacy Enhancing Technologies, Vol.2016 (Issue 4); 315-334. http://dx.doi.org/10.1515/popets-2016-0042

Leontiadis, I., Efstratiou, C., Picone, M., and Mascolo, C. (2012). Don't kill my ads! balancing privacy in an ad-supported mobile application market, In Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications (HotMobile '12), Association for Computing Machinery, New York, NY, USA, Article 2; 1–6. http://dx.doi.org/10.1145/2162081.2162084

Lin, J., Amini, S., Hong, J. I., Sadeh, N., Lindqvist, J., and Zhang, J. (2012). Expectation and purpose: understanding users' mental models of mobile app privacy through crowdsourcing. In Proceedings of the 2012 ACM Conference on Ubiquitous Computing (UbiComp '12), Association for Computing Machinery, New York, NY, USA; 501–510. http://dx.doi.org/10.1145/2370216.2370290

Lin, J. (2013). Understanding and capturing people's mobile app privacy preferences, Ph.D. Dissertation, Carnegie Mellon University, PA, USA; No. CMU-CS-13-127.

McDonald, A. M., and Cranor, L. F. (2008). The Cost of Reading Privacy Policies, I/S: A Journal of Law and Policy for the Information Society, 4(3); 540–565.

Melicher, W., Kurilova, D., Segreti, S. M., Kalvani, P., Shay, R., Ur, B., Bauer, L., Christin, N., Cranor, L. F., and Mazurek, M. L. (2016). Usability and Security of Text Passwords on Mobile Devices, In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16), Association for Computing Machinery, New York, NY, USA; 527–539. https://doi.org/10.1145/2858036.2858384

Mell, P., Scarfone, K., and Romanosky, S. (2007). A Complete Guide to the Common Vulnerability Scoring System Version 2.0. FIRST-Forum of Incident Response and Security Teams; 1-23.

Mobile Security Framework; 2020 [cited 2021 Oct 17]. Available from: https://github.com/MobSF/Mobile-Security-Framework-MobSF.

Momen, N. and Fritsch, L. (2020). App-generated digital identities extracted through Android permission-based data access - a survey of app privacy, In: Reinhardt, D., Langweg, H., Witt, B. C. and Fischer, M. (Hrsg.), SICHERHEIT 2020. Bonn: Gesellschaft für Informatik e.V; 15-28. https://doi.org/10.18420/sicherheit2020_01

Mylonas, A., Kastania, A., Gritzalis, D. (2012). Delegate the smartphone user? Security awareness in smartphone platforms. Comput. Secur. 34; 47–66. https://doi.org/10.1016/j.cose.2012.11.004

Nigeria Data Protection Regulation; 2019 [cited 2021 Aug 8]. Available from: https://ndpr.nitda.gov.ng/Content/Doc/NigeriaDataProtectionRegulation.pdf.

Olejnik, L., Acar, G., Castelluccia, C., and Díaz, C. (2015). The Leaking Battery: A Privacy Analysis of the HTML5 Battery Status API, Lecture Notes in Computer Science, vol. 9481; 254–263. https://doi.org/10.1007/978-3-319-29883-2_18

Paintsil, E., and Fritsch, L. (2011). A Taxonomy of Privacy and Security Risks Contributing Factors. 6th International Summer School Conference on Privacy and Identity Management for Life, Aug 2010, Helsingborg, Sweden; 52-63. http://dx.doi.org/10.1007/978-3-642-20769-3_5

Paintsil, E., and Fritsch, L. (2013). Executable Model-Based Risk Analysis Method for Identity Management Systems : Using Hierarchical Colored Petri Nets Executable Model-Based Risk Assessment Method for Identity Management Systems, Trust, Privacy, and Security in Digital Business : 10th International Conference, TrustBus 2013, Prague, Czech Republic; 48–61. https://doi.org/10.1007/978-3-642-40343-9_5

Papageorgiou, A., Strigkos, M., Politou, E.A., Alepis, E., Solanas, A., and Patsakis, C. (2018). Security and Privacy Analysis of Mobile Health Applications: The Alarming State of Practice, vol. 6; 9390-9403. https://doi.org/10.1109/access.2018.2799522

Qian, K., Parizi, R.M., and Lo, D.C. (2018). OWASP Risk Analysis Driven Security Requirements Specification for Secure Android Mobile Software Development, In 2018 IEEE Conference on Dependable and Secure Computing (DSC); 1-2. https://doi.org/10.1109/DESEC.2018.8625114

Reidenberg, J.R., Breaux, T., Carnor, L.F. and French, B. (2015). Disagreeable privacy policies: Mismatches between meaning and users' understanding. Berkeley Technology Law Journal 30(1); 39–68.

Ryan, F., Fritz, A., Impiombato, D., and Australian Strategic Policy Institute, International Cyber Policy Centre, issuing body. (2020). TikTok & Wechat : curating and controlling global information flows Australian Strategic Policy Institute, Barton, Australian Capital Territory [cited 2021 Jun 17]. Available from: http://www.jstor.org/stable/resrep26120.7.

Seneviratne, S., Seneviratne, A., Mohapatra, P., and Mahanti, A. (2014). Predicting user traits from a snapshot of apps installed on a smartphone. SIGMOBILE Mob. Comput. Commun. Rev. 18 (2); 1–8. http://dx.doi.org/10.1145/2636242.2636244

Solove, D.J. (2011). Nothing to Hide: The False Tradeoff between Privacy and Security. Yale University Press.

Statista (2021). Number of apps available in leading app stores as of 1st quarter 2021; [cited 2021 Jun 17]. Available from: https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores.

Turner, B. (2021). Mobile App Download and Usage Statistics; [cited 2021 Jun 17]. Available from: https://www.bankmycell.com/blog/how-many-phones-are-in-the-world.

Vallina-Rodriguez, N., Sundaresan, S., Razaghpanah, A., Nithyanand, R., Allman, M., Kreibich, C., and Gill, P. (2016). Tracking the Trackers: Towards Understanding the Mobile Advertising and Tracking Ecosystem. https://doi.org/10.48550/arXiv.1609.07190

Zhang Y., Yang Y., and Wang X. (2018). A Novel Android Malware Detection Approach Based on Convolutional Neural Network, In Proceedings of the 2nd International Conference on Cryptography, Security and Privacy (ICCSP 2018). Association for Computing Machinery, New York, NY, USA; 144–149. https://doi.org/10.1145/3199478.3199492

Zou, Y., Zhu, J., Wang, X., and Hanzo, L. (2016). A Survey on Wireless Security: Technical Challenges, Recent Advances, and Future Trends, Proceedings of the IEEE, 104; 1727-1765. https://doi.org/10.1109/JPROC.2016.2558521