Enhancing Privacy and Security in Apps for Disabled Groups: A Comprehensive Analysis

Shubhankar Mehrotra (40309967), Raju Ahmed (40293811), Vishal Vishal (40294677), Kathiraesh Lakshmi Narayanan (40303408), Raghu Pavan Annam (40303699), Sarath Kumar Manchineni (40153139), Kunati Bala Krishna Yadav (40292128). Concordia University, Montreal, Quebec, Canada
https://github.com/raghupavan009/INSE-6120

Abstract— Applications catering specifically to the requirements of persons with disabilities are proliferating, and the utilization of these applications is on the rise across all demographics, including those with disabilities. Mobile applications assist individuals with disabilities in enhancing their everyday life and connectedness, while enabling guardians or family members to keep track of the well-being and health-related activities of those they care about. Despite their many benefits, these applications handle sensitive personal information, including medical records, real-time whereabouts, and the disabled peoples highly critical data. There is a dearth of research that specifically addresses disabled apps, in contrast to the abundance of literature on mobile app security and privacy for the public. We illuminate the privacy and security concerns of mobile applications designed for impaired users, using a mix of dynamic and static analysis on 15 prevalent android applications from the Google Play Store. As a result of our study, we have found many security and privacy flaws that have allowed attackers to get access to user data and compromised their privacy. Our investigation reveals that most applications inadequately safeguard user security and privacy in many respects; notably, 10 applications permit important permission analysis, while 8 applications exhibit inappropriate tracker identification, among other deficiencies. We want for our research to enhance awareness about the security and privacy vulnerabilities posed by these applications and to prompt programmers to fortify their protective strategies.

Keywords: Disabled Apps, Android Application, Security and Privacy of Applications

I. Introduction

With more and more people using smartphones and tablets, the disabled are finding themselves thrust into a complicated digital environment where online dangers may have real-world consequences. Although many people with disabilities have smartphones and are enthusiastic users of mobile technology, research has shown that they are at a higher risk of privacy and security breaches than the public. On a limited scale, a few of studies have shown privacy concerns among a single vulnerable population, such as the disabled or the crippled. Research on handicapped populations has mostly focused on how the disabled act in relation to their personal safety and privacy.

The primary goal of this survey is to examine the security and privacy features of Android applications created for marginalized communities, such as those dealing with homelessness, drug misuse recovery, or those with disabilities. The main goal is to find out whether these applications secure data and if they comply with certain regulations. Analysis of privacy policies, technological evaluations, and information flow analyses are some of the methods outlined in the text for reaching this goal. We examine 15 popular Android applications for the disabled in detail in the coming sections. Security flaws, backend problems, third-party trackers, and unsafe data transfer are just a few examples of the privacy and security concerns that we identify and assess in these applications.

Contributions and Notable Findings.

1. To assess privacy and security concerns in disabled applications, we develop a method that combines static and dynamic analysis.
2. Regarding the privacy analysis, it was found that three out of ten applications evaluated specifically mention being HIPAA compliant. Seventy percent of applications encrypt data while it's in transit, and forty percent make sure it's encrypted while it's at rest.
3. Static Code Analysis: In terms of permissions, half of all applications have over ten potentially harmful permissions. Eighty percent of applications employ trackers, and each app typically uses four trackers. Problems with Encryption: 60% of applications still utilize insecure, out-of-date techniques of encryption (sha-1, MD5).
4. Analyzing Dynamic Code: HTTP vs. HTTPS: Only 30% of applications use HTTP to transfer data. Half of the servers support weak cypher suites.
5. Securing Servers: SSL Labs The majority of applications (60%) get an "A" in SSL security. The setting of the cipher suite is vulnerable in 40% of the cases.
6. App Security Flaws: End- to- End Encryption is inconsistently not guaranteed by 6 out of 10 applications. Unsafe random number generators are used by 40% of applications.
7. Risks to Personal Information: Without explicitly stating it, 80% of applications exchange user data with other parties. Out of all the applications out now, only 30% provide a way for users to explicitly authorize to data sharing.
8. Safety Ratings: Out of all the applications that were tested, the average security score is 50 out of 100. On average, health-related apps (such Work it Health and Be My Eyes) obtain 53 out of 100 points, which is a little higher.
9. Openness in Policymaking: Four out of ten applications' privacy policies don't make it clear that they follow international privacy rules.

II. METHODOLOGY

A. Privacy Policy Analysis:

Since all these apps store highly sensitive information, they must adhere to HIPAA (Health Insurance Portability and Accountability Act) privacy standards. Hence we manually go through the privacy policy of each android application and examine key questions covering both privacy and security rules. These include checking if the app has a clear privacy policy, specifies data collection and usage, meets HIPAA access requirements, involves third-party storage, permits entire data deletion, includes secure login and ensures data encryption.

We also used an online tool called Exodus Privacy Checker, which is a popular privacy auditor. It checks the privacy policy for any given app, all the permissions which it requires to function at software and hardware level.

B. Information Flow Analysis:

This method we use to analyze how the data is actually processed in the app from the moment it has been input to the system. We list out the data types and their formats analyzing how they are stored and transmitted. Further we also looked into how the data move within the app, putting emphasis on if they are being shared with any third party API.

C. Technical Assessment and Access permissions:

In this step we will do a more thorough analysis regarding the data transmission in the app, we must make sure it is end-to-end encrypted, making sure vital information like passwords are hashed before storing into the database. Furthermore, we dive into users' access permissions, whether they can control what personal data they are willing to share and have permissions to opt out of services they are not comfortable with.

D. Code Analysis:

In the code analysis part, we combine privacy and security rules to evaluate the app. We set up a testing environment using MobSF (Mobile Security Framework) and SSL Labs and run two main tests: Static Analysis and HTTP Analysis. Based on the MobSF report, we check key points and ask questions about HIPAA compliance, like whether information is encrypted, if there is a unique user ID, and if data transmission is secure.

1. Static Analysis: Static Analysis focuses on the APK file of the app, analyzed through MobSF. This includes examining permissions, APK ID analysis, browsable activities, Manifest analysis etc.,

● Permissions: We check each permission to see access levels, such as read or modify SD card, GPS location and network status.

● APK ID: The APK ID reveals how the APK was built and highlights any obfuscation methods.

● Browsable Activities: Lists all the activities accessible by the browser.

● Manifest: We check the manifest file to see essential app details, including permissions, package name, components and requirements.

● Code Details: This helps in identifying potential issues like logging sensitive information, insecure code, stored secrets or any database vulnerabilities.

● Domain Malware and URLs: Scans for malicious domains and lists URLs used for data transactions.

● Trackers: Identifies any tracking frameworks that monitor user behavior on Android Apps.

2.Dynamic Analysis: Dynamic Analysis focuses on executing and testing the mobile application in a monitored environment for observing its real-time behavior and interactions with the operating system and network. It is specifically helpful in identifying security vulnerabilities that may not be detected in static analysis, such as, Runtime vulnerabilities, Insecure Network communication, Unauthorized file access etc.,

3. HTTP Analysis: In the HTTP Analysis, we examine the web server's configuration to evaluate the security of data transmission. We use Qualys SSL Labs, a tool that tests server security and assigns a letter grade. SSL Labs also checks for open ports as a part of the test.
SSL Labs grades the application's server security by checking:

● Certificate: Verifies the servers identity.

● Protocols: These are standards that both communication parties must follow.

● Cipher Suites: This refers to the set of encryption and security protocols used. In TLS 1.3, the suite primarily defines encryption and message authentication.

● Handshake Simulation: This simulates the initial connection process, where communication parameters are established between the client and the server.

● HTTP Requests: These are requests sent by the client to trigger actions on a server.

This analysis helps in ensuring that the app's server meets security standards for safe data transmission.

III. FINDINGS (PRIVACY/DYNAMIC ANALYSIS) OF APPS

A. App: Loosid: Sober Recovery Network (v3.20.1)

Loosid is a mobile app designed to support individuals in recovering from their alcohol and substance abuse habits. It provides a community based platform, where users can connect and interact with other people, who are also working towards sobriety by sharing their experiences and staying motivated in the process of recovery.

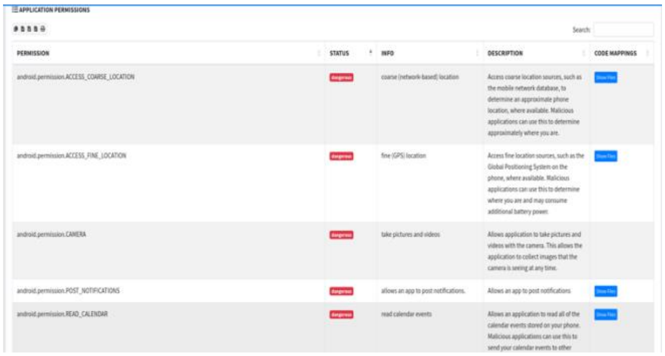
This app aims to create a safe and supportive environment for people looking to maintain long-term sobriety and build a healthy, substance-free lifestyle.

Privacy Analysis: The privacy policy is clearly stated regarding the data collection and usage, but doesn't explicitly mention if it's HIPAA compliant. It is mentioned that the app explicitly avoids collecting data from users under 28, which aligns with general privacy principles. All the data, especially the PHI is encrypted in transit and at rest. However, since data is shared with third parties, they must sign Business Associate Agreements (BAAs) to ensure HIPAA compliance.

The absence of documenting secure login measures like multi-factor authentication also raise potential concerns.

Technical Analysis: The app was given a score of 48/100 and uses 10 trackers, indicating a moderate level of security. According to the report, there are a total of 17 activities of which 5 are exported activities. It has 29 services and 2 of them are exported services. Also, we can see that there are 25 receivers and 10 of them are exported receivers.

According to the report, 13 out of the 48 permissions are classified as dangerous. Some dangerous permissions (shown in Figure 1) to mention are accessing the location information, accessing camera and permissions for reading and writing to the external storage. These permissions can be misused by any harmful applications and hence pose a serious risk to PHI data.



APPLICATION PERMISSIONS

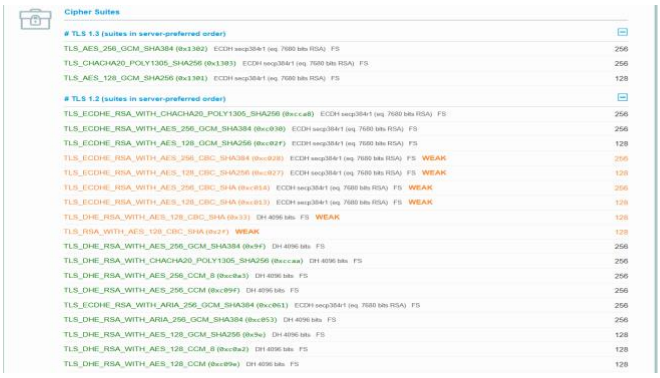
Search:

PERMISSION	STATUS	INFO	DESCRIPTION	CODE MAPPINGS
android.permission.ACCESS_COARSE_LOCATION	dangerous	coarse (network-based) location	Access coarse location sources, such as the mobile network database, to determine an approximate phone location, where available. Malicious applications can use this to determine approximately where you are.	Show Files
android.permission.ACCESS_FINE_LOCATION	dangerous	fine (GPS) location	Access fine location sources, such as the Global Positioning System on the phone, where available. Malicious applications can use this to determine where you are and may consume additional battery power.	Show Files
android.permission.CAMERA	dangerous	take pictures and videos	Allows application to take pictures and videos with the camera. This allows the application to collect images that the camera is seeing at any time.	
android.permission.POST_NOTIFICATIONS	dangerous	allows an app to post notifications.	Allows an app to post notifications	Show Files
android.permission.READ_CALENDAR	dangerous	read calendar events	Allows an application to read all of the calendar events stored on your phone. Malicious applications can use this to send your calendar events to other	Show Files

Figure 1 : Permissions

Moreover, the app is susceptible to three severe vulnerabilities. For instance, it uses CBC mode of encryption with PKCS5/PKCS7 padding, which is highly vulnerable to padding oracle attacks.

We have tested the server configuration on the Loosid app using SSL Labs and it is seen that the app has scored an overall grade A, meaning that the server security is in a very good posture. However, in the cipher suites part, we find that the application accepts 30 weak suits out of a total of 47 cipher suites for communication, which the adversary can take advantage of.



Cipher Suites

```
# TLS 1.3 (suites in server-preferred order)
TLS_AES_256_GCM_SHA384 (0x1302)   ECDH secp384r1 (eq. 7680 bits RSA)   FS                                   256
TLS_CHACHA20_POLY1305_SHA256 (0x1303)   ECDH secp384r1 (eq. 7680 bits RSA)   FS                             256
TLS_AES_128_GCM_SHA256 (0x1301)   ECDH secp384r1 (eq. 7680 bits RSA)   FS                                   128
# TLS 1.2 (suites in server-preferred order)
TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xcca8)   ECDH secp384r1 (eq. 7680 bits RSA)   FS              256
TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (0xc030)   ECDH secp384r1 (eq. 7680 bits RSA)   FS                    256
TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (0xc02f)   ECDH secp384r1 (eq. 7680 bits RSA)   FS                    128
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (0xc028)   ECDH secp384r1 (eq. 7680 bits RSA)   FS   WEAK             256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (0xc027)   ECDH secp384r1 (eq. 7680 bits RSA)   FS   WEAK             128
TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (0xc014)   ECDH secp384r1 (eq. 7680 bits RSA)   FS   WEAK                256
TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (0xc013)   ECDH secp384r1 (eq. 7680 bits RSA)   FS   WEAK                128
TLS_DHE_RSA_WITH_AES_128_CBC_SHA (0x33)   DH 4096 bits   FS   WEAK                                          128
TLS_RSA_WITH_AES_128_CBC_SHA (0x2f)   WEAK                                                                  128
TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 (0x9f)   DH 4096 bits   FS                                              256
TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (0xccaa)   DH 4096 bits   FS                                      256
TLS_DHE_RSA_WITH_AES_256_CCM_8 (0xc0a3)   DH 4096 bits   FS                                                 256
TLS_DHE_RSA_WITH_AES_256_CCM (0xc09f)   DH 4096 bits   FS                                                   256
TLS_ECDHE_RSA_WITH_ARIA_256_GCM_SHA384 (0xc061)   ECDH secp384r1 (eq. 7680 bits RSA)   FS                   256
TLS_DHE_RSA_WITH_ARIA_256_GCM_SHA384 (0xc053)   DH 4096 bits   FS                                           256
TLS_DHE_RSA_WITH_AES_128_GCM_SHA256 (0x9e)   DH 4096 bits   FS                                              128
TLS_DHE_RSA_WITH_AES_128_CCM_8 (0xc0a2)   DH 4096 bits   FS                                                 128
TLS_DHE_RSA_WITH_AES_128_CCM (0xc09e)   DH 4096 bits   FS                                                   128
```

Figure 2 - HTTP Analysis

B. App: Link2Care (v1.2.1)

Link2Care is a mobile application designed to connect homeless individuals with caregivers. This app helps in improving the access to support services aiming to reduce substance abuse, psychological distress, homelessness duration and re-arrest rates. It also includes features for connecting users to care managers and crisis resources, by leveraging mobile technology to address both health and safety challenges.

Privacy Analysis:
The privacy policy is publicly available and clearly outlines the types of data collected. However, while the data sharing is limited, the scope of third-party involvement could require clarification regarding their compliance with privacy rules, as sharing data with third parties could pose privacy risks without explicit user consent. The policy mentions that data may be used to respond to inquiries or for future communications (e.g., marketing). However, explicit user consent for such purposes isn't mentioned, which could conflict with stricter HIPAA rules, as it might involve protected health information (PHI). The policy also doesn't explicitly mention if it is compliant with privacy standards such as HIPAA, which is a concern as it handles health data.

Technical Analysis: Upon Static Analysis, the app was given a score of 46/100 and uses 3 trackers, which indicates a moderate level of risk to its users. According to the report, there are a total of 42 activities out of which 3 are exported activities. It has 16 services and 4 of them are exported services. Also, we can see that there are 3 receivers and 1 of them is an exported receiver. According to the report, 14 out of the 26 permissions are classified as dangerous. Some of these permissions include accessing the contact list information and reading and writing permissions to the external storage, which could pose a serious risk to PHI and data safety. Code Analysis reveals that the app uses weak encryption algorithms which can lead to leaking sensitive health data.

Dynamic Analysis: We have loaded Frida scripts into the app to test various issues like SSL pinning, root detection and debugger checks. In the tests, we have successfully bypassed the debugging, indicating that there are no anti-debugging measures setup in the app. It is also seen that the app does not restrict the operation on rooted devices, which is confirmed by Root Bypass. It is also found that SSL pinning frameworks were not implemented in the app, making it potentially vulnerable to man-in-the-middle (MITM) attacks, where an attacker could intercept network traffic.

C. App: Workit Health (v3.5.1)

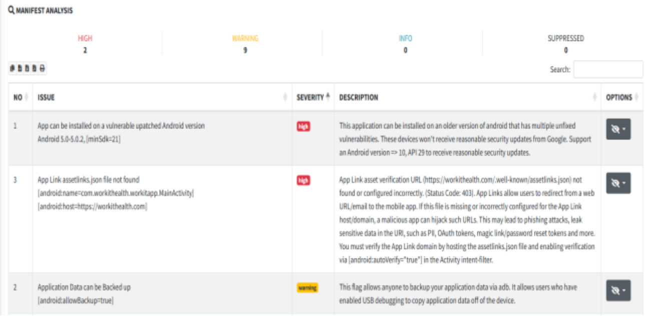
It is a mobile-first telemedicine platform designed to provide accessibility and personalized care for individuals struggling with substance use disorders, mental health disorders and other behavioral needs. This app takes a proven and well-rounded approach to recovery and wellness, offering users with digital tools and access to professional support for people seeking to regain control over their health.

Privacy Analysis: Work it Health has a clear Privacy Policy and is explicitly mentioned that it complies with the HIPAA policy. It mentioned clearly regarding our data collection and its usage for critical health related tasks like processing prescriptions and enhancing the treatments, if required. Our information is shared with thief parties only as needed, for example, with healthcare providers, pharmacies, or insurance companies and always in line with legal and HIPAA requirements. We also have the right to access, review or modify our medical records and revoke consent for certain uses of our information, thereby clearly aligning with HIPAA regulations.

Technical Analysis:

The app was given a score of 51/100 and uses 3 trackers, which means the app has decent security and confidentiality. All the app data is properly encrypted and has secure login measures in place like multi-factor authentication. According to the report, there are a total of 7 activities of which 1 is an exported activity. It has 14 services and 2 of them are exported services. Also, we can see that there are 14 receivers and 4 of them are exported receivers. Additionally, the app has 12 providers, of which 1 is an exported provider. According to the report, 12 out of the 46 permissions are classified as dangerous. Some of these dangerous permissions include accessing geographic location and reading audio files from external storage. These permissions can be misused by any harmful applications and hence pose a serious risk to PHI data.

On analyzing the manifest file (shown in Figure 3) , it is clear that the app can be installed on an older version of android, which can have multiple unfixed vulnerabilities and hence poses a dangerous threat to security.



MANIFEST ANALYSIS

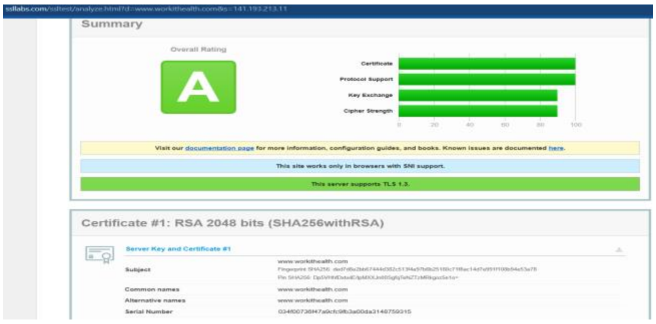
HIGH	WARNING	INFO	SUPPRESSED
2	9	0	0

Search:

NO	ISSUE	SEVERITY	DESCRIPTION	OPTIONS
1	App can be installed on a vulnerable upatched Android version Android 5.0-5.0.2, [minSdk=21]	high	This application can be installed on an older version of android that has multiple unfixed vulnerabilities. These devices won't receive reasonable security updates from Google. Support an Android version => 10, API 29 to receive reasonable security updates.	
3	App Link assetlinks.json file not found [android:name=com.workithealth.workitapp.MainActivity] [android:host=https://workithealth.com]	high	App Link asset verification URL (https://workithealth.com/.well-known/assetlinks.json) not found or configured incorrectly. (Status Code: 403). App Links allow users to redirect from a web URL/email to the mobile app. If this file is missing or incorrectly configured for the App Link host/domain, a malicious app can hijack such URLs. This may lead to phishing attacks, leak sensitive data in the URI, such as PII, OAuth tokens, magic link/password reset tokens and more. You must verify the App Link domain by hosting the assetlinks.json file and enabling verification via [android:autoVerify="true"] in the Activity intent-filter.	
2	Application Data can be Backed up [android:allowBackup=true]	warning	This flag allows anyone to backup your application data via adb. It allows users who have enabled USB debugging to copy application data off of the device.	

Figure 3 - Manifest Analysis

We have analyzed the Workit Health app's server configuration using SSL Labs and found that it has scored an overall grade A (as shown in Figure 4), indicating a strong server security. However, in the cipher suite section, 14 out of the 23 cipher suites supported by the server are considered weak. These weak suites could potentially be exploited by attackers to compromise secure communications.



sslabs.com/ssltest/analyze.html?d=www.workithealth.com&s=141.193.213.11

Summary

Overall Rating

A

Visit our documentation page for more information, configuration guides, and books. Known issues are documented here.

This site works only in browsers with SNI support.

This server supports TLS 1.3.

Certificate #1: RSA 2048 bits (SHA256withRSA)

Server Key and Certificate #1

Subject	www.workithealth.com Fingerprint SHA256: [illegible] Pin SHA256: [illegible]
Common names	www.workithealth.com
Alternative names	www.workithealth.com
Serial Number	034f00736f47a9cfc9fb3a00da3148759315

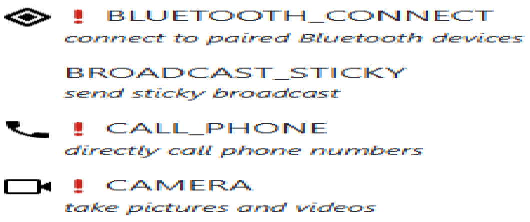
Figure 4 - HTTP Analysis

Dynamic Analysis: On performing dynamic analysis in MobSF, specifically from the TLS/SSL Security tester, it is shown that, the app passes TLS Misconfiguration test and Cleartext Traffic test, but fails to pass the TLS Pinning test and TLS Pinning Bypass test, which means the app is generally secure in terms of transport layer encryption, but is vulnerable to Man-In-The-Middle (MITM) attacks, due to lack of proper TLS pinning.

D. App: Be my Eyes (v3.0.3)

Be my eyes is an application which helps visually challenged or people who have lost their sight complete everyday tasks. How this works is, the person can contact or connect with people through live video calls who can provide them real time assistance in completing their task.

Privacy Analysis: After doing initial Exodus analysis we found it has 3 trackers on crash reporting and general analysis, furthermore there were 17 permission which the app requires from the user to have access to crucial hardware and software features of their device, for example accessing the camera, the microphone, the gps and the phones network access, this things are crucial for the app but also vital towards the security aspect of things.



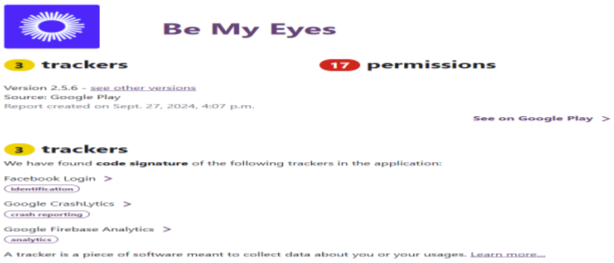
! BLUETOOTH_CONNECT
connect to paired Bluetooth devices

BROADCAST_STICKY
send sticky broadcast

! CALL_PHONE
directly call phone numbers

! CAMERA
take pictures and videos

Figure5- Some Critical hardware permissions



Be My Eyes

3 trackers

17 permissions

Version 2.5.6 - see other versions
Source: Google Play
Report created on Sept. 27, 2024, 4:07 p.m.

See on Google Play >

3 trackers

We have found **code signature** of the following trackers in the application:

Facebook Login >
identification

Google CrashLytics >
crash reporting

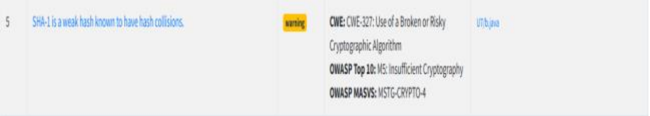
Google Firebase Analytics >
analytics

A tracker is a piece of software meant to collect data about you or your usages. Learn more...

Figure 6- Trackers List

Regarding the volunteers demeanor to the user's there are certain protocol put inside the app which binds them into an agreement of carrying themselves in a respectful behavior with an helpful and cheerful attitude towards their helper since this is crucial for the uses privacy and security, this is done by asking the volunteers to agree to a code of conduct when they register to offer their services. Adding on to that, users data is not shared with any third party app according to their privacy policy, the particular data type that is shared is used for crash logs and to perform diagnostics which in turn is used to enhance the usability experience of the app, users can also request to have their data completely removed from the system as soon as they have declared not needing the services anymore.

Static and Dynamic Analysis: Furthermore moving onto the analysis of end-to-end data encryption, it is mentioned on their services that all video, audio and text data is end-end encrypted, to further assurance we ran diagnosis using MobSF dynamic test analyzer and found contradicting statements where the data is being transmitted over HTTP not HTTPS, there is a possibility that some of the data is sent as clear text traffic, this is a significant security vulnerability. Doing the code analysis using MobSF, there are 8 code cases where it is moderately vulnerable like SHA-1 and MD5 is used for hashing which are weak forms of hashing. Adding on to that there is one critical code analysis where production level code is debug-able, this is unacceptable. (Figure-7 Code Analysis).

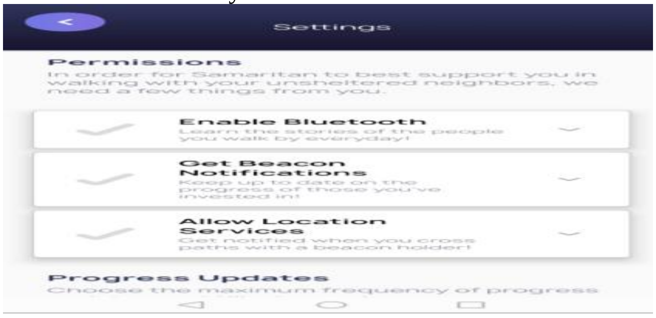


5	SHA-1 is a weak hash known to have hash collisions.	warning	**CWE:** CWE-327: Use of a Broken or Risky Cryptographic Algorithm **OWASP Top 10:** M5: Insufficient Cryptography **OWASP MASVS:** MSTG-CRYPTO-4	UT/b.java

E. App: Samaritan (v6.0.13)

This is an app which helps support individuals who have trouble finding a place to stay, going through financial difficulty or need food, clothing or other necessities. After registering through the app, you will be notified about nearby people who need help, you can recognize them through their digital beacons. Upon reading their story you can donate them money which will be sent securely to their in-app digital wallet.

Privacy Analysis: Looking into the privacy side of things there is not much we can find out where the app data is shared and true nature of how it is served but going through their official website it is mentioned that some information shared with services providers which handles their email, notification and gps services, rest there are protocols put in place to ask your consent when there is a need to share. You can always ask or request the administration for a copy of your data or make sure of its termination once you are done with the app's functionalities. Going through the privacy settings on the app, there are options which allows the users to grant permission in allowing the app to turn off or on the Bluetooth capabilities of the device, the same protocol applies for the beacon notification and GPS services which is an important factor since the system will always know your current location at any moment.



Permissions

In order for Samaritan to best support you in walking with your unsheltered neighbors, we need a few things from you.

Enable Bluetooth
Learn the stories of the people you walk by everyday!

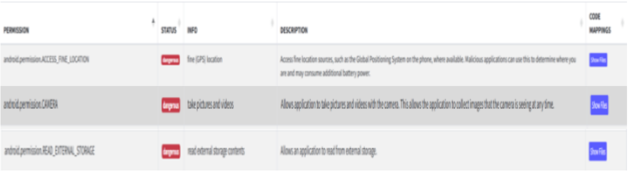
Get Beacon Notifications
Keep up to date on the progress of those you've invested in!

Allow Location Services
Get notified when you cross paths with a beacon holder!

Progress Updates
Choose the maximum frequency of progress

Figure 8- App Settings

Utilizing Mob SF we were able to retrieve a full list of permissions the app requires to operate on a hardware and software level, there are 17 permissions among them 4 of them critical which involves permission in operating your gps and reading data from your phones storage.



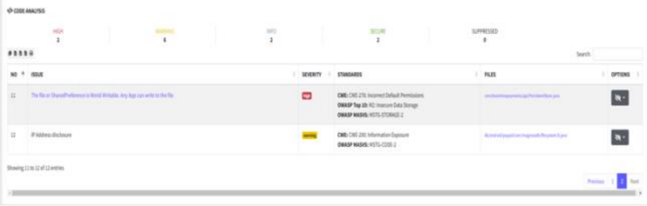
PERMISSION	STATUS	INFO	DESCRIPTION	CODE MAPPINGS
android.permission.ACCESS_FINE_LOCATION	dangerous	fine (GPS) location	Access fine location sources, such as the Global Positioning System on the phone, where available. Malicious applications can use this to determine where you are and may consume additional battery power.	Show Files
android.permission.CAMERA	dangerous	take pictures and videos	Allows application to take pictures and videos with the camera. This allows the application to collect images that the camera is seeing at any time.	Show Files
android.permission.READ_EXTERNAL_STORAGE	dangerous	read external storage contents	Allows an application to read from external storage.	Show Files

Figure 9- Screenshot of the Critical Permissions

Static and Dynamic Analysis: After downloading the app, during the registration we have realized that there are no multi-factor security layers for the process, nor was there any email verification of the user's identity, which is not a deal breaker but since this app process 1000's of users every day a layer of security for the user's identity could be crucial.

For analyzing how the data moves around the app we did a review on how the data is input into the system, its validation, the encryption methods used for its transfer. After installing the app, when the user registers by inputting his basic information, it is validated and transmitted to the server via secure Api endpoints. The transfer of data is done over HTTPS and other encryption methods to make sure the packets are secure when they are traveling from the client side to the server mentioned as one of their safety features, although after doing our analysis using MobSF there are some form of data sent as clear text traffic, we are not sure if those data sent are crucial information, moving on to domain malware check, the app connects to 5 domains in different location, no malicious activity has been detected. Doing further investigation on the app, we analyzed the code and found 2 high risk factors involving insecure default permissions for data which signifies any app can write to that file.

Figure 10- High risk vulnerabilities in the code



CODE ANALYSIS

HIGH	WARNING	INFO	SECURE	SUPPRESSED
2	6	2	2	0

Search:

NO	ISSUE	SEVERITY	STANDARDS	FILES	OPTIONS
11	The file or SharedPreference is World Writable. Any App can write to the file	high	**CWE:** CWE-276: Incorrect Default Permissions **OWASP Top 10:** M2: Insecure Data Storage **OWASP MASVS:** MSTG-STORAGE-2	[illegible]	
12	IP Address disclosure	warning	**CWE:** CWE-200: Information Exposure **OWASP MASVS:** MSTG-CODE-2	[illegible]	

Showing 11 to 12 of 12 entries

Previous 1 2 Next

F. App: I am Sober(v1.1)

This app is used to help people who are recovering from bad habits or addictions. It allows them to track the number of days they have abstained from their said habit and improve on it.

Privacy Analysis: Going through their privacy policy page, it is mentioned what services your data is shared with and for what purposes, services like Amazon, Google Analytics and Post Hog. They are mostly for storage, user usage analytics and subscription management providers. Furthermore, you can also completely terminate your data after you have logged out or are no longer in need of their services.

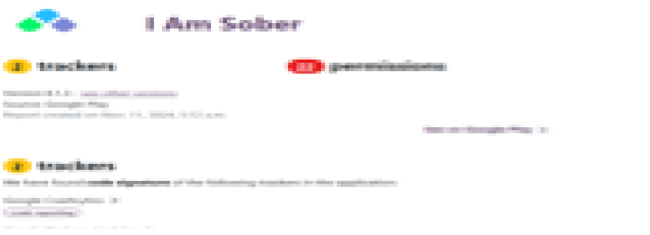


Figure 11- Exodus Analysis

Additionally, we did an analysis on Exodus to see the number of trackers used in the app and the hardware resource permission which the app utilizes. There are 2 trackers, and 22 permissions found on their analysis.

Digging through the in-app permissions, there were 3 critical permissions which the app needed for operation: the camera, the gps and the ability to read data from your shared storage.

Static and Dynamic Analysis: As an initial first step we referred to the google app store safety details to have a brief authentic analysis of the app's security.
The documents confirm the encryption of the data on transit using TLS. HIPAA compliant practices are also used to ensure strict data access protocol methods by maintaining audits of all interactions within the app. We further conducted dynamic analysis for the traffic data, no data is sent as plain text has been confirmed. The app connects to 10 domains across the globe and their corresponding malware check has come out to be negative. Furthermore, doing the signer certificate analysis, sha 256 is used as the hashing algorithm in the app.

G. App: AccessNow(v.2.5.1)

Access Now is an application used for sharing accessibility information about places around the world. It lets users search for places like restaurants, hotels or stores, or to see what is nearby with the accessibility features being the highlight. If any information is not available already on the app, then it can be contributed that would help the worldwide community. It's like a Google Maps like experience but centered around accessibility.

Privacy Analysis: When examining privacy policy on its website, it mentions the use of various methods to collect information that can be linked back to the user. Those include continuous permission for precise location, approximate location and being able to use any sensor available on the smartphone. When the user comments on any content of the app, the developer collects information such as username. First name and last name. The app request's fine location permission and can access the location in the background even when the app is not being used. The app could read and as well as write to external storage. Running the app through exodus privacy reveals that it contains 6 trackers in total and 18 different permissions which include collecting the advertising ID of the user with activity recognition.

Dynamic Analysis: The app can be used without an account and asks for precise location permission on launch. When creating an account on the app, the app uses encryption in transit making sure that the information being entered by the user is safe while it is reaching the developer.

H. App: Aira Explorer (v.2.6.15)

Aira Explore is an app designed to give visually impaired or blind people real-time visual descriptions. By integrating artificial intelligence with human agents, subscribers can get into detailed descriptions of their surroundings, receive navigation support, and be assisted in various tasks during daily routine.

Privacy Analysis: The app can request permissions to access advertising ID, fine location access, record, make phone calls, write to external storage. It uses coarse and fine location information that can be used by malicious applications to obtain a general or precise user location, probably consuming extra battery power.

Dynamic Analysis: It also needs permissions to connect with paired Bluetooth devices and write to external storage, both of which can be misused if not properly secured. In addition to the above vulnerabilities, this application uses CBC encryption mode with PKCS5/PKCS7 padding and is vulnerable to padding oracle attacks.

I. App: Dateability(v53)

Dateability is one of those few dating apps created to make love accessible for disabled people and chronic illnesses by building a platform that is inclusive of physical disabilities, intellectual, and psychiatric disabilities.

Privacy Analysis: There are several security and privacy concerns: it accesses coarse and fine location data, which might be misused by other malicious applications to track the approximate or precise location of the device. Besides that, it needs permissions to connect to paired Bluetooth devices, request authentication tokens, and read image files from external storage. These can, if not managed correctly, lead to potential high-level security breach incidents.

Dynamic Analysis: It is also vulnerable to the Janus vulnerability, which might leak sensitive data such as hardcoded usernames, passwords, or keys. The use of weak hashes prone to hash collisions, such as SHA-1 and MD5, also compromise security. The application also uses an insecure Random Number Generator; that is, it is predictable and may be vulnerable to exploits.

J. App: CoughDrop(v2023.11.01)

CoughDrop app is an Augmentative and Alternative Communication (AAC) app targets users who face verbal communication problems due to conditions such as autism or cerebral palsy.

Privacy Analysis: Using this app, the personal information may be collected, used, and shared as will be evident through a review of the following App Privacy Policy. Notably, the policy emphasizes that personal information, including names, email addresses, and biographic data, is collected on a voluntary basis and is used to notify users of updates, respond to inquiries, and fulfill other specified purposes. The policy also addresses the handling of information from children under 13, adhering to the Children's Online Privacy Protection Act, and specifies that health-related information should not be submitted through the app.

Static and dynamic analysis: Reports show several important security issues.

The overall grade is medium risk with a score of 52 out of 100, which means the application might be vulnerable. Key issues of the app are being installable on older Android versions that are vulnerable, like Android 4.4-4.4.4, which lack critical security updates. It logs sensitive information and copies data to the clipboard, which can be accessed by other applications. Also, using external storage to read and write data is risky, as any application can read data written to the external storage. The application also uses SQLite databases and runs raw SQL queries; if these are not maintained properly, they may get attacked by SQL injection attacks. Moreover, the app requires various dangerous permissions, such as fine location, record audio, and read phone state, which, if used in a specific context, may adversely affect user privacy and security.

Dynamic analysis underlines more issues regarding the security of TLS/SSL implemented in an app. Although it passed cleartext traffic and TLS misconfiguration tests, it failed the TLS Pinning/Certificate Transparency Bypass Test, pointing to potential vulnerabilities regarding secure channels of communication. The app also communicates with various domains, most of which raise flags regarding potential malware. Additionally, the use of the outdated SHA1 hashing algorithm in the app raises concerns about integrity and security related to data. These findings raise the need for CoughDrop to enhance security measures that include, but are not limited to, supporting newer Android versions, enhancing data encryption, and improving secure communication protocols to protect user data and ensure safe usage of the app.

K. App: Fuel Service(v24.10.07)

The fuel Service app, developed by Fuel Service Ltd., is intended to support disabled drivers in approaching and requesting assistance from petrol stations.

Privacy Analysis: The privacy policy described in the app explains how personal information is collected, used, and shared, with an indication that user-provided information, including email addresses and mobile phone numbers, is collected on a voluntary basis. The policy further elaborates on the use of GPS technology for determining location, to provide services relevant to the users' location, while ensuring that such location data is not shared with other users or partners. The policy on data retention, opt-out rights, and security measures to protect user information also applies.

Static and Dynamic analysis: App uses an insecure random number generator and logs sensitive information that, unless properly secured, might be accessed. Besides, the app requests quite a few dangerous permissions, such as fine location and coarse location, which might have adverse effects on the privacy of users if used unwarily. Also, its use of the Flurry tracker for analytics and advertisement brings up some questions regarding user data collection and usage.

Reports show various security-related concerns. The security rating of the app is low risk, standing at 62 out of a possible 100, though there are significant vulnerabilities. The app can be installed on older and vulnerable Android versions, such as Android 5.0-5.0.2, that do not contain critical security updates, thus exposing the user to the Janus Vulnerability.

Dynamic analysis further highlights that while it passes cleartext traffic and TLS misconfiguration tests, it communicates with various domains which have been flagged for hosting malware. In addition, it uses external storage to both read and write data. This brings additional risks since any application can read data written to external storage. These findings mean that fuel Service needs to consider much better security measures like supporting Android versions, more enhanced data encryption, and better secure communication protocols that avoid user data leakage, making it safe to use. On the other hand, the malware permission score of this app is relatively lower, being 5 out of 25, implying fewer permissions that might get exploited by malware.

L. App: ShelterApp(v2.0.11)

ShelterApp is an AI chatbot that links at-risk youth with key services such as shelters, LGBTQ+ advocacy groups, and food banks.

Privacy Analysis: The privacy policy shares how personal information is collected and used, pointing out that user-submitted information of name, email address, and location in real-time is collected to provide relevant services. The policy discusses data retention, opt-out rights, and security measures to protect user information. However, the app's security score is medium risk, amounting to 48/100, for which attention may be warranted due to some potential vulnerabilities.

Static and Dynamic analysis: Analysis reports identify a few security concerns. Specifically, the app can be installed on older, vulnerable Android versions (Android 5.0-5.0.2), which do not contain critical security updates that are required for it to be protected from the Janus Vulnerability. The app also logs sensitive information, and the WebView implementation is insecure, which might be exploited unless properly protected.

The application also requests dangerous permissions from users, which include access to fine location and external storage; these could be misused and may raise issues about user privacy. Tracking within the app-including but not limited to Facebook Analytics and Google Firebase Analytics-can be considered a concern for end-user data collection and usage.

Dynamic analysis further reveals that while the app passes cleartext traffic and TLS misconfiguration tests, it communicates with several domains that raise red flags regarding malware. In addition, there is a risk because the app uses external storage to both read and write data; other applications can access data written to external storage. These findings indicate that ShelterApp needs to consider much stronger security practices, including support for more current Android OS versions, data encryption, and secure communication to better protect user data and ensure safe use of the application.

Since ShelterApp targets Homeless and Teenagers, who are among the most vulnerable groups in society, security and privacy need to be applied more vigorously. The application should invest in regular security audits, expand updates to support the latest Android versions, and espouse open data practices to keep user information safe. This will reduce potential vulnerabilities and be better prepared to give its users more security.

M.App: Avaz AAC (v6.6.7)

Avaz AAC is a user-friendly program created for people with speech and language problems, notably children with autism spectrum disorder (ASD) or cerebral palsy. It employs Augmentative and Alternative Communication (AAC) approaches to help people communicate.

The program has an easy-to-use interface for creating phrases with images, text, and voice synthesis.

Privacy Analysis: The app has a medium risk rating owing to permissions that provide access to the camera, microphone, and external storage. These rights, while useful for recording photographs and sounds, may be misused if not adequately protected. Although the software handles potentially sensitive information, such as user-created message templates, no clear mention of HIPAA compliance or other privacy requirements was discovered. Furthermore, the app's privacy policy does not specify how third-party trackers (four discovered) handle user information. Permissions such as reading and writing the external storage grant access to files that, if misused, may reveal personal information.

Dynamic Analysis: Technical flaws include the usage of obsolete cryptographic algorithms like SHA-1 and MD5, which are susceptible to collisions and jeopardize data integrity. Exported activities, services, and broadcast receivers make components available to other apps, enhancing the possibility of illegal interactions. Furthermore, sensitive data may be saved to external storage, making it available to other programs on the device. A lack of effective SSL/TLS pinning increases vulnerability to man-in-the-middle (MITM) attacks during data transfer.

N. App : WheelTrans(v4.0.2.4)

The Toronto transportation Commission (TTC) created Wheel Trans, a transportation app that focuses on accessibility. The app allows people with disabilities to order, track, and manage paratransit services. It streamlines mobility by offering real-time travel scheduling, alerts, and service updates designed specifically for individuals with restricted physical mobility.

Privacy Analysis: The app has a risk rating which requires rights for fine location (GPS) and physical activity recognition, both of which are necessary for delivering transportation services but represent hazards if data is shared or kept incorrectly. The software lacks formal privacy certifications such as GDPR and HIPAA, raising concerns about how it protects user data. The software also employs four trackers, including Mix panel and Firebase Analytics, which may record user activity without user authorization. Furthermore, permissions for receiving notifications and accessing device states provide entry points for potential abuse by third-party libraries.

Dynamic Analysis: Wheel Trans employs HTTPS for secure communication, although its dependence on outdated cryptographic protocols such as SHA-1 compromises security. SQL injection vulnerabilities in SQLite database queries may allow attackers to modify stored data.

Permissions for reading and writing to external storage increase the danger of sensitive information disclosure. While SSL certificate pinning protects against MITM attacks, the app's weak random number generator undermines token-based authentication measures.

O. App : Wheelmap(v5.3)

Wheel map is a collaborative mapping program that helps people with mobility limitations identify and rank wheelchair-accessible venues. It allows users to input information on accessible facilities such as ramps, elevators, and bathrooms, resulting in a community-generated accessibility map.

Privacy Analysis: The program captures location data, which needs constant access to exact GPS data, even while operating in the background. It also gathers user interaction data, such as accessibility ratings and comments, which are saved and might be shared with third-party services. The Exodus research found six trackers in the app, raising concerns about third-party data sharing without explicit user agreement.

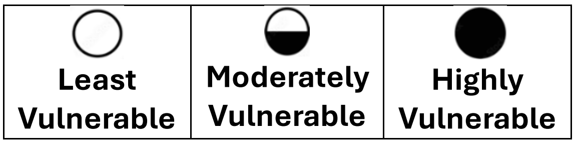
Furthermore, the software requires rights to read and write to external storage, which might be exploited if malicious programs obtain access to shared files. While the app permits anonymous use, registering an account involves the provision of an email address and a password, and the service's policy does not specify the security safeguards for this information.

Dynamic Analysis: The application uses HTTPS to securely send user data. However, trackers built inside the app may damage the whole privacy system. Permissions for accessing external storage and using device sensors pose concerns if used by malicious programs. Sensitive data is encrypted; however, the app lacks robust security measures against vulnerabilities such as SQL injection and poor hash algorithms (e.g., SHA-1 and MD5). The app's reliance on external storage and advertising IDs for tracking user behavior makes it more vulnerable to privacy breaches.

IV. RESULTS

Insufficient security measures are found in some apps, as per the evaluations. Common issues include the use of insecure or outdated cryptographic techniques, insufficient encryption or TLS pinning, and too broad permissions that might provide third-party APIs access to sensitive user data. Examples of companies that were relatively vulnerable include Loosid and Workit Health, which had security gaps in their data processing practices and were not very compliant with privacy standards. The risks of utilizing external storage and tracker activities are highlighted by applications like CoughDrop and Wheelmap, which highlights the necessity for more stringent privacy and security measures.

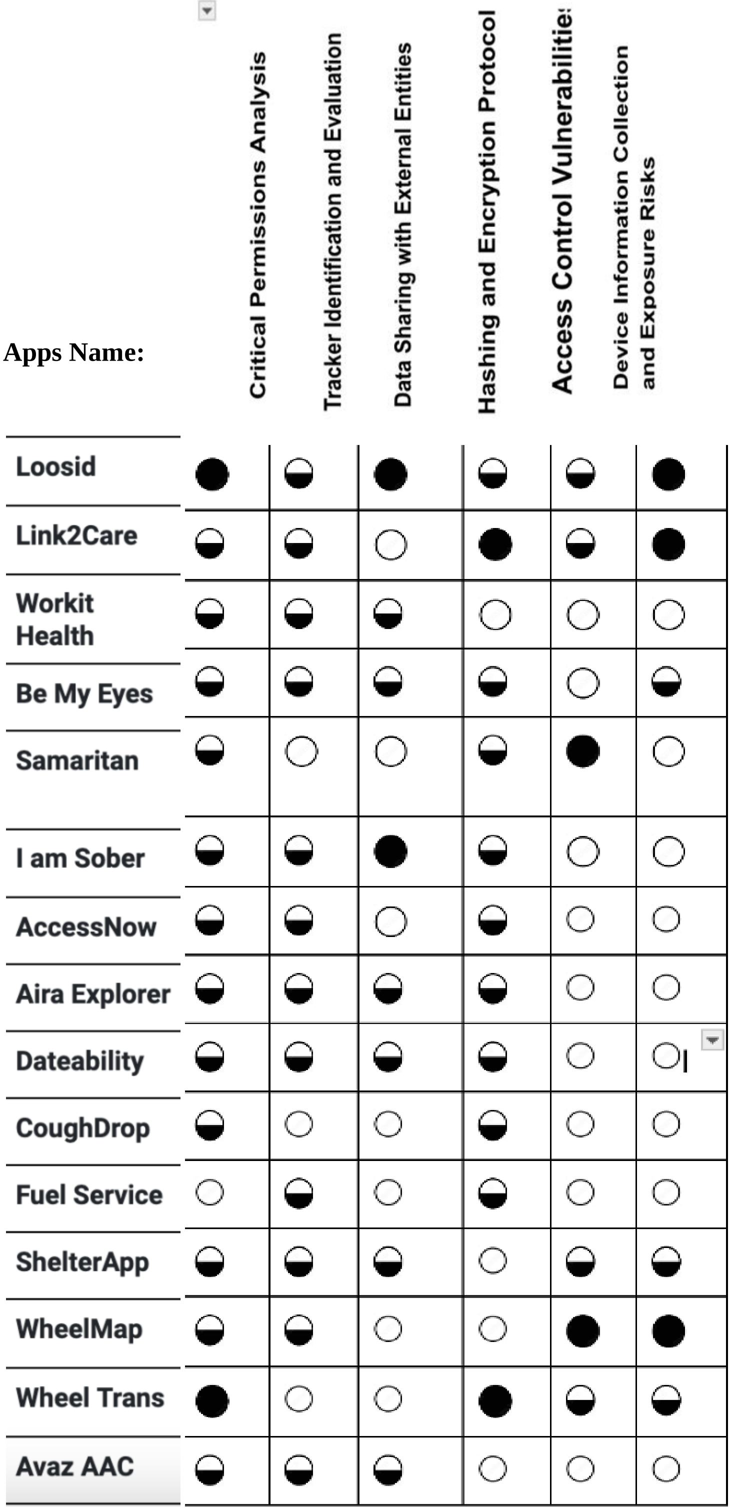
- Privacy and Compliance: When it comes to data collecting and sharing with other parties, many apps don't have users' explicit authorization. The confidentiality of sensitive information is especially important for marginalized communities, making this a major concern.
- Permissions and Data Sharing: Almost every program asks for too many permissions. Despite its prevalence, external data exchange is not always transparent. Users may be concerned about their privacy since apps like FuelService and ShelterApp share user data with analytics companies without
- being completely transparent about the extent of this sharing.
- Encryption: Inadequate mechanisms for encryption. Apps like Be My Eyes show how data exchanged via unsecured networks, like HTTP, significantly increases hazards.
- Server and Network Security: There is an increased danger of exposure to malware since certain applications connect with blacklisted or possibly harmful sites, as shown by domain research.
- Tracker and Analytics Usage: There are an abundance of trackers included in apps for analytics and monitoring: The use of several trackers in Wheelmap and AccessNow raises issues over the degree to which user activity is monitored. Although there are trackers that serve a useful function (like crash reporting), there are other trackers that gather superfluous data and may violate privacy rules.
- Usability vs. Security Trade-Off: Live help (Be My Eyes) and location monitoring (FuelService) are just two examples of how many applications put the needs of vulnerable populations first. But security is usually a victim of this: Authorization for real-time support might put users at danger of spying. Automatic data sharing for analytics and other features that rely on user permission and privacy concerns are at odds with these goals.



○ Least Vulnerable	◒ Moderately Vulnerable	● Highly Vulnerable

The figure shows 3 different levels of Vulneability starting from least then moderately and highly for 15 applicattions.

We have 6 security factors that shows different levels in the table:



Apps Name:	Critical Permissions Analysis	Tracker Identification and Evaluation	Data Sharing with External Entities	Hashing and Encryption Protocol	Access Control Vulnerabilitie:	Device Information Collection and Exposure Risks
Loosid	●	◒	●	◒	◒	●
Link2Care	◒	◒	○	●	◒	●
Workit Health	◒	◒	◒	○	○	○
Be My Eyes	◒	◒	◒	◒	○	◒
Samaritan	◒	○	○	◒	●	○
I am Sober	◒	◒	●	◒	○	○
AccessNow	◒	◒	○	◒	○	○
Aira Explorer	◒	◒	◒	◒	○	○
Dateability	◒	◒	◒	◒	○	○
CoughDrop	◒	○	○	◒	○	○
Fuel Service	○	◒	○	◒	○	○
ShelterApp	◒	◒	◒	○	◒	◒
WheelMap	◒	◒	○	○	●	●
Wheel Trans	●	○	○	●	◒	◒
Avaz AAC	◒	◒	◒	○	○	○

V. Conclusion

A total of fifteen Android applications designed to aid individuals with disabilities were examined in detail in our paper. We analyzed these applications using both dynamic and static analysis in our approach.

We expose each of these applications' many warning signs and the ways in which they pose a security concern. Moreover, we have seen patterns in the domain flows and app permissions that reveal the dominance of certain firms, third-party libraries, or permissions in the market. Because we can go even farther into the investigation to uncover other vulnerabilities and faults, we believe it should continue. This will make the world a safer place for the elderly, who can rest easy knowing that their brand-new cellphones are secure.

The findings emphasize the need of finding a middle ground that guarantees both usability and security. Programmers are obligated to:

- Stay current with cryptography standards and use robust encryption methods.
- Decrease dependency on trackers and unnecessary permissions.
- Verify adherence to international privacy regulations such as GDPR and HIPAA.
- Apply static and dynamic analysis techniques to their applications on a regular basis to find and fix security flaws.

To properly safeguard sensitive user data, applications should address these gaps.
This is especially important for vulnerable groups, since they are often the ones most threatened by security and privacy breaches.

To make these apps more private and secure:

Privacy is of the utmost importance, and apps must follow regulations such as GDPR and HIPAA. They should also have transparent rules that explain how data is used and shared.

Cut Down on Permissions: Let users have more say over their data by reducing needless permissions and using consent-based processes.
Implement State-of-the-Art Cryptography: Secure data in transit by using TLS Pinning and replacing weak protocols with stronger ones (e.g., SHA-256).
Periodic Reviews of Security: Find and fix vulnerabilities by doing static and dynamic analyzes; make sure it works with the latest OS versions.
Keep Security and Usability in Check: Especially for at-risk communities, design features that are easy to use without sacrificing security.

References

[1] Reports.exodus-privacy.eu.org. (n.d.). Œμxodus. [online] Available at: https://reports.exodus-privacy.eu.org/en/.

[2] https://www.ssllabs.com/ssltest/index.html

[3] MobSF Documentation - https://mobsf.github.io/docs/#/

[4] Wanrong Zhao., Hossain Shahriar., Victor Clincy., Zakirul Alam Bhuiyan., et al.: Security and Privacy Analysis of Mhealth Application: A Case Study.

[5] Loosid: Sober Recovery Network. (2024). Privacy Policy. [online] Available at: https://loosidapp.com/privacypolicy/

[6] Link2Care. (2024). Privacy Policy. [online] Available at: http://www.dayton.com.hk/doc/smartwatch_privacy_policy.html

[7] Workit Health. (2024). Privacy Policy. [online] Available at: https://www.workithealth.com/privacy-policy/

[8] Bemyeyes.com. (2024). Privacy Policy. [online] Available at: https://www.bemyeyes.com/privacy.

[9] I Am Sober. (2024). Privacy Policy. [online] Available at: https://iamsober.com/en/site/privacy-policy.

[10] MyCough Drop Policy: https://www.mycoughdrop.com/privacy

[11] FuelService Policy: https://fuelservice.org/en/privacy.html

[12] Shelter App Policy: https://shelter.app/privacy_policy

[13] Aira Explorer app's privacy policy available at https://aira.io/privacy-policy

[14] Access Now Policy: https://accessnow.com/privacy-policy/

[15] Dateability Policy: https://info.dateabilityapp.com/privacy-policy/

[16] Wheel map Policy: https://news.wheelmap.org/en/privacy-consent-wheelmap/

[17] Wheel Trans Policy: https://www.ttc.ca/transparency-and-accountability/policies/Privacy/ttc-privacy-code

[18] Avaz AAC policy: https://avazapp.com/privacy-policy/