

# **Technical Analysis of the Data Practices and Privacy Risks of 43 Popular Mobile Health and Fitness Applications**

**Research Performed For:**  
**Privacy Rights Clearinghouse**  
<https://privacyrights.org>

**Technical Research, Analysis, and Documentation By:**  
**Craig Michael Lie Njie**

Version History of this Document:

- First Public Release: Version 1.0, Released July 15, 2013
- Most Recent Update: Version 1.1, Released August 12, 2013

Mr. Lie Njie has over 20 years of professional technical experience, and in the last four years has designed, developed, deployed, and performed technical analysis of a variety of mobile applications for both Apple's iOS and Google's Android. He is Founder and CEO of Kismet World Wide Consulting: <http://www.KismetWorldWide.com>

## **Table of Contents:**

1. Executive Summary: Technical Report of PRC’s Research into Mobile Health and Fitness Application Data Practices .....	5
2. Why Should Mobile App Developers Care about Privacy? .....	6
2.1. Building privacy-protective applications is not difficult .....	6
2.2. Protecting privacy up front could save time, money, and effort later .....	6
2.3. Ignoring privacy could result in legal liability .....	6
2.4. One bad review of an app can dramatically reduce sales .....	6
3. Project Goals, Apps Selected, and Some Caveats .....	7
3.1. Technical goals of this research .....	7
3.2. 43 Mobile apps reviewed: 23 free apps, 20 for-cost (paid) apps; iOS and Android platforms only; other limits to our analysis .....	7
3.3. Sample size: We believe it was large enough .....	8
3.4. Our research bias inherently favors privacy .....	8
3.5. This study is a snapshot in time because apps change frequently .....	8
4. Methodology: Overview of Technical Evaluation Environment .....	9
4.1. Devices used: representative of the current iOS and Android devices .....	9
4.2. Data collection spreadsheet .....	9
4.3. Overview of our app analysis setup .....	11
4.4. Breaking SSL with Charles Proxy: man-in-the-middle SSL attacks .....	11
4.5. Watching the packets on the wire: WireShark and tcpdump .....	12
4.6. Copying off the devices: DiskAid (iOS); adb and FTP Server software (Android) .....	12
4.7. Databases found on devices: Using SQLite Database Browser .....	13
5. Analysis: Evaluating an Application’s Technical Privacy Risks .....	13
5.1. Evaluating technical risk to the user’s privacy and whether an app complies with its own privacy policy .....	13
5.1.1. Technical operation vs. posted privacy policies .....	13
5.1.2. Assigning a subjective score .....	13
5.2. Features/functions analyzed to reveal information practices .....	14
5.2.1. Data collection: Data a user provides and other data an app collects .....	14
5.2.2. Data stored on the device .....	14
5.2.3. Data transmission: network communication .....	14
6. Results: Data Flow Analysis: Collection, Storage, Transmission and Usage of Data by Mobile Health and Fitness Apps .....	15
6.1. Non-obvious data flow analysis of mobile apps .....	15
6.1.1. Apps send data in the clear without user knowledge .....	15
6.1.2. Apps connect to many third-party sites without user knowledge .....	15
6.1.3. Unencrypted connections expose embarrassing data to everyone on a network .....	15
6.2. Types of data collection and their associated risks .....	15
6.2.1. Explicit collection .....	15
6.2.2. Implicit collection .....	15
6.2.3. Implicit collection of data by third-party advertisers .....	16

6.2.4. Linking to external data sources such as ad networks, data brokers, etc. . . . .	16
6.2.5. Data collection by third-party login/user identification services . . . . .	17
6.3. Data transmission: types, recipients, and risks. . . . .	17
6.3.1. Data sent to app developers’ websites. . . . .	17
6.3.2. Data sent to third-party sites as part of the app . . . . .	17
6.3.3. Data sent to third-party sites for analytics and advertising . . . . .	17
6.3.4. Data sent in the clear via URLs and GET requests . . . . .	18
6.4. Data storage. . . . .	18
6.4.1. Local storage . . . . .	18
6.4.2. Local removable storage and the risks of data stored on the mobile device’s SD card . . . . .	18
6.4.3. Data stored externally. . . . .	19
6.4.4. Cookies and other tracking and identification technologies . . . . .	19
7. Results: Risk of Apps . . . . .	19
7.1. Overall technology risk assessment: high/medium/low risks . . . . .	19
7.2. Risk levels and risk assessments of all 43 apps: risk levels 1-9. . . . .	20
7.3. Do apps do what their privacy policies say they do? . . . . .	21
7.4. Free vs. Paid: Paid apps still pose major privacy risks because they use HTTP, not HTTPS . . . . .	21
7.5. Apps with advertisements pose a significantly higher privacy risk . . . . .	22
7.5.1. Advertisements are more likely to be found in free apps . . . . .	22
8. Problems Encountered. . . . .	23
8.1. Our spreadsheet questions and/or answers are biased toward the edge cases. . . . .	23
8.2. Temporal: Apps change constantly, making it possible to analyze only a moment in time . . . . .	23
8.3. Context: Some apps that have a specific medical purpose can't really be evaluated in a general context . . . . .	24
8.4. Comparing apps on iOS vs. Android is like comparing apples to rainbows. . . . .	24
8.5. Bias: We instinctively avoided the most seemingly egregious apps. . . . .	24
8.6. A complete analysis of all platforms/screen sizes/OS versions was not possible. . . . .	24
8.7. We were unable to evaluate long-term risk and risk of future app updates . . . . .	25
8.8. We were unable to evaluate apps with currently buggy versions . . . . .	25
8.9. Should we be comparing apps only within the same “type” or “category” or market? . . . . .	25
9. Appendix A: HOW TO: Privacy Aware Mobile App Development . . . . .	26
9.1. Executive Summary: A privacy HOW TO for mobile app developers. . . . .	26
9.2. Intended audience. . . . .	26
9.3. Overview: Primary technical privacy risks to users of mobile apps . . . . .	27
9.3.1. Unencrypted network communication: Always use HTTPS . . . . .	27
9.3.2. Advertisers. . . . .	27
9.3.3. Analytics services. . . . .	27
9.4. Privacy checklist . . . . .	27
9.4.1. Limit the collection, storage, and network transmission of user data . . . . .	27
9.4.2. Notify users of all data collection and explain its purpose. . . . .	27
9.4.3. Encrypt <u>all</u> network communication with SSL or stronger . . . . .	28
9.4.4. Use high-grade encryption for users’ personal information. . . . .	28
9.4.5. NEVER send user information in clear text . . . . .	28

9.4.6. Always use POST, never GET .....	28
9.4.7. Salt and hash all passwords before sending or storing .....	29
9.4.8. Do not expose information in URLs – obfuscate page names and data .....	29
9.4.9. Protect against URL replay attacks by using single-use or expiring URLs .....	29
10. Appendix B: Spreadsheet Used for Data Analysis: CSV Format .....	30

## **1. Executive Summary: Technical Report of PRC's Research into Mobile Health and Fitness Application Data Practices**

This document discusses the goals, research methodology, and findings of the technical evaluation of 43 mobile health and fitness apps available on iOS or Android platforms.

This research was performed between March and June 2013 by Kismet World Wide Consulting, a technology consulting firm (<http://www.KismetWorldWide.com>), for Privacy Rights Clearinghouse (<https://privacyrights.org>), on a project funded by the California Consumer Protection Foundation.

The technical evaluation process involved reviewing mobile application privacy policies, installing and using the apps, analyzing what data apps store locally on devices, and detailed analysis of the network communication between the app and the Internet.

Our goal was to compare how an app actually functions technically with the developer's description of what it does in any published privacy policy, and to identify key privacy risks in the technical practices some apps currently use.

We found that the biggest risks to the information privacy of users of mobile health and fitness apps were technical in nature, usually due to unencrypted connections to third-party advertisers and analytics services, which often disclose personal information. Developers of these apps can mitigate these risks by:

- (1) Always using HTTPS (SSL-encrypted) network connections for all communications between the app and any internet server, to ensure that any data sent over those connections is encrypted and kept private;
- (2) Not using third-party advertiser or analytics services – data disclosed to these third parties was found to be a major privacy risk. If a third-party service is used, the least amount of privacy-sensitive data possible should be sent, and only over HTTPS (SSL-encrypted) connections; and
- (3) Not sending privacy-sensitive keywords and information in URLs or via GET requests, but instead sending them as POST requests over an encrypted HTTPS (SSL) connection. For example, this URL discloses the user's medical condition (AIDS) and location (latitude and longitude):

`http://site.com/Search.asp?condition=AIDS&latitude=35.24234&longitude=24.5253`

In addition to this report, we have authored and published a HOW TO document for mobile app

developers that describes best practices for building privacy protections into apps in order to correctly collect, store, and transmit personal data. That HOW TO is also included in this document as Appendix A.

## **2. Why Should Mobile App Developers Care about Privacy?**

### **2.1. Building privacy-protective applications is not difficult**

If privacy-sensitive personal data is not necessary to an app's functionality, don't collect it.

If an app's function requires collecting personal information, be sure to encrypt or obfuscate all such data the app stores or transmits. There are a variety of open source products, closed source products, and third-party consulting and implementation services available that will do this for you.

Always use HTTPS; never use HTTP, for any Internet communication.

### **2.2. Protecting privacy up front could save time, money, and effort later**

It's difficult, if not impossible, to remove data from a database once it has been collected. If your app allows malicious third parties to collect users' personal data, and that information becomes public, it will cost more time and money to remedy the situation than it would have to make the app secure at the outset.

### **2.3. Ignoring privacy could result in legal liability**

Developers who ignore privacy and security may risk incurring legal liability in the event of exposure of users' personal information.

### **2.4. One bad review of an app can dramatically reduce sales**

App sales are linked directly to the most recent user reviews: better reviews mean more sales, bad reviews mean fewer sales. A review that a developer exposed users' personal data to a third party could easily affect sales.

### **3. Project Goals, Apps Selected, and Some Caveats**

#### **3.1. Technical goals of this research**

The primary goals of our technical evaluation were as follows:

- Analyze and describe the technical functions of a representative sampling of mobile health and fitness apps, including what data they collect, and to the extent possible, what they do with that data and how it flows outward in ways users may be unaware of;
- Develop a process and supporting tools for this kind of technical evaluation, then describe our methodology and release our tools for others interested in conducting similar research;
- Provide a technical and non-technical basis for evaluating the privacy risks involved in using mobile apps;
- Compare the technical operation of the applications with the information in their privacy policies and settings options;
- Publish a [HOW TO \(http://www.privacyrights.org/mobile-medical-apps-privacy-developers-howto-report.pdf\)](http://www.privacyrights.org/mobile-medical-apps-privacy-developers-howto-report.pdf) document for mobile app developers that describes how to build privacy-protecting apps that correctly collect, store, and transmit user data, along with other privacy-friendly features to consider building into their apps. The HOW TO is also included in this report as Appendix A.

#### **3.2. 43 Mobile apps reviewed: 23 free apps, 20 for-cost (paid) apps; iOS and Android platforms only; other limits to our analysis**

Because of budget and time constraints on this project, we limited our analysis to the following:

- 43 health and fitness apps: 23 free and 20 paid; half on iOS, half on Android;
- Paid apps chosen from different pricing tiers: 6 apps under \$1.50, 6 apps between \$1.50 and \$2.50, 6 apps between \$2.50 and \$10, and 2 apps over \$10.
- Only mobile apps developed for Apple's iOS and Google's Android, and available through the Apple App Store or Google Play. These represent almost all of the mobile app market share;
- Apps that did not require any external devices or additional expenses;
- Apps developed for consumer use, rather than use by health professionals;

- Apps that were popular or had significant downloads (paid apps were chosen from the Top 200 Paid apps on both stores);
- Apps for users 13 or over and therefore outside the restrictions of the Children’s Online Privacy Protection Act (COPPA);
- No testing of apps when linked to social network sites like Facebook or Google+.

### **3.3. Sample size: We believe it was large enough**

We considered whether analyzing just 43 apps out of the many thousands available might produce inaccurate results. We quickly found during our technical research, however, that the technologies of many apps were similar and that they posed similar privacy risks. We concluded that our sample size and selection methodology (see the “Methodology” section below) was, in fact, large enough to produce adequate and representative observations about the general state of mobile health and fitness apps for use without external devices.

### **3.4. Our research bias inherently favors privacy**

As privacy advocates, we place a high value on protecting the privacy and security of user information that mobile apps collect. One goal of the project is to evaluate the privacy risks built into the development and distribution of mobile apps. Since any definition of risk is subjective and individual, we acknowledge an unavoidable bias in our research. We also acknowledge that different biases exist in the [consumer analysis \(http://www.privacyrights.org/mobile-medical-apps-privacy-consumer-report.pdf\)](http://www.privacyrights.org/mobile-medical-apps-privacy-consumer-report.pdf) that accompanies this project, which was not done by a computer scientist, and the technical analysis, which was.

### **3.5. This study is a snapshot in time because apps change frequently**

We looked at apps available at this moment in time (December 2012-June 2013). We are aware that before we publish these results, many of the apps we analyzed will be updated, or may have been removed from the marketplace.

The reality of mobile apps is this: The only way to know for sure what an app does at any given moment is to test the most current version of that app at that moment.

## **4. Methodology: Overview of Technical Evaluation Environment**

### **4.1. Devices used: representative of the current iOS and Android devices**

We tested the 43 apps primarily on an iPad 3 running iOS version 6.1.3 and a Nexus 7 running Android “Jelly Bean” OS version 4.2.2. We also performed limited testing with an iPhone 4S running iOS version 6.1.3 and a Motorola Droid 2 Global running Android 2.3.

The devices used in our testing are representative of the majority of the iOS and Android devices and operating system versions currently in use.

After initial analysis, we chose to test primarily on iPad 3 because (1) it can run both iPhone and iPad apps (had we used an iPhone we would have been limited to just iPhone-only apps), and (2) it can run the same apps as most other iOS devices in use today, including all current versions of the iPad and iPhone from the 3GS through the iPhone 5, since those devices either came with iOS 6 or can be upgraded to iOS 6.

Most Android devices sold in the last two years are running some version of Android OS version 4.x. We chose to test primarily on the Nexus 7 because like the iPad, it too could run all apps from both the phone and tablet categories.

We chose a combination of apps from both the “phone” category (smaller screens) and the “tablet” category (larger screens). We did not find any significant difference in technological privacy risks between apps written for the tablet and the phone.

### **4.2. Data collection spreadsheet**

From the start, we realized there were a number of dimensions of privacy risk and technical evaluations we needed to assess for each app in order to cover the spectrum of potential risk. We built a spreadsheet for data collection and analysis. A [clean version of this spreadsheet](http://www.privacyrights.org/mobile-medical-apps-privacy-evaluation-spreadsheet-report.pdf) (<http://www.privacyrights.org/mobile-medical-apps-privacy-evaluation-spreadsheet-report.pdf>) for others to use as a guide in doing their own privacy analysis of mobile apps is provided, and a CSV version of the spreadsheet is included in this document as Appendix B.

The spreadsheet contains a total of 150 questions that we answered for each app. The questions were broken down into sub-categories: Notice/Awareness, Choice/Consent, Access/Participation, Integrity/Security, Enforcement/Redress, Permissions Required to Use the App, Costs, Developer Information, Technical Evaluation, and Overall Assessments. We

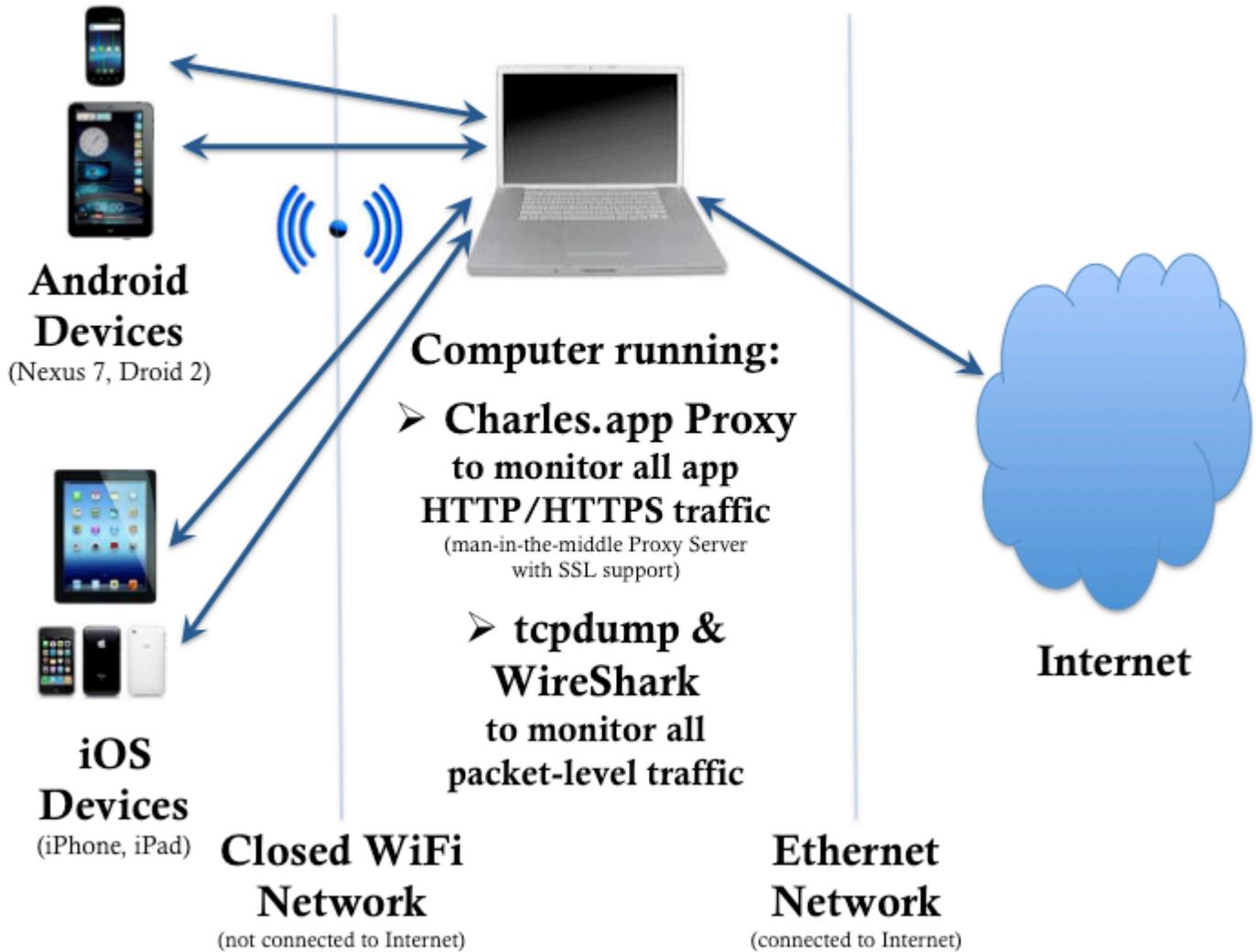
analyzed the results of that data collection to arrive at the aggregate results and observations we detail in these reports.

We learned that a spreadsheet this size becomes difficult to use as an organizing tool for such a large number of questions for as many apps as we evaluated. For future projects we suggest instead creating a database with web-based data entry front-ends for the collection, analysis, and reporting of the data.

Where the spreadsheet provided the most value was in having a detailed list of exactly what the reviewer should be looking for, a list that was consistently used for each app. This level of detail and rigor greatly helped in both our technical and non-technical analysis, and we recommend a similar level of rigor for any privacy analysis of mobile apps.

### 4.3. Overview of our app analysis setup

Here's a picture of our app analysis setup:



### 4.4. Breaking SSL with Charles Proxy: man-in-the-middle SSL attacks

To watch the network communication between the apps and the Internet, we put a laptop in the middle. That laptop “proxied” or bridged the data between the apps and the Internet – it acted as a middle man for the network traffic.

In order to see exactly what data an app transmitted over the network, we employed a “man-in-the-middle, SSL-enabled proxy server.” This allows apps to run on the devices without any modifications, and without any notice that their network communication is being monitored and analyzed. Every request is detailed and reported by the proxy server, including all cookies and other technologies used.

There are several proxy servers available that can perform this task. We chose to test with the Charles Proxy Server on the Mac OS/X platform. Charles is an HTTP proxy/HTTP monitor/Reverse Proxy that enables a developer to view all HTTP and SSL/HTTPS traffic between their machine and the Internet. This includes requests, responses and the HTTP headers (which contain the cookies and caching information). Charles Proxy is available here:

<http://www.charlesproxy.com/>

#### **4.5. Watching the packets on the wire: Wireshark and tcpdump**

In addition to the high-level network communication details provided by the Charles Proxy Server software described above, we also used the low-level tools Wireshark and tcpdump to look at the individual network packets that were sent and received. Wireshark provides a UI front-end to tcpdump, and is much easier to use.

Wireshark is available here: <http://www.wireshark.org/>

tcpdump is an open source utility that comes standard on many operating systems. Source code for tcpdump is available here: <http://www.tcpdump.org/>

#### **4.6. Copying off the devices: DiskAid (iOS); adb and FTP Server software (Android)**

In addition to network communications, we also analyzed the data that was stored by the apps on the smartphone or tablet device. To do this, we used:

- DiskAid: This is an easy-to-use app that provides access to all the documents stored by apps on an iOS device. We chose it because of its simplicity and extensive feature set. DiskAid is available here: <http://www.digidna.net/diskaid>
- adb: The Android Debug Bridge (adb): The adb software allowed us to connect to and access data on the Android device. adb comes as part of the Android SDK: <http://developer.android.com/sdk/>
- FTP Server: It turned out to be time consuming and difficult to get access to the Android device over adb, so to simplify that access we installed an FTP server app on the Android devices to allow quick access to the data stored there. This was much faster than just using adb. There are several FTP server apps available on the Google Play marketplace for Android apps. We used the one called “Ftp server” available here: <https://play.google.com/store/apps/details?id=com.theolivetree.ftpserver>

#### **4.7. Databases found on devices: Using SQLite Database Browser**

Using the utilities described above to look at the data apps stored on the devices, we found several SQLite databases as part of that data. Some apps had several databases. To view the contents of the SQLite databases, we used the open source “SQLite Database Browser” available here: <http://sourceforge.net/projects/sqlitebrowser/>

### **5. Analysis: Evaluating an Application’s Technical Privacy Risks**

#### **5.1. Evaluating technical risk to the user’s privacy and whether an app complies with its own privacy policy**

##### **5.1.1. Technical operation vs. posted privacy policies**

We checked for privacy policies on developers’ websites and within the apps themselves. Each privacy policy, if there was one, was compared with the actual information practices we observed as part of the technical analysis of the app. We believe that apps that do not follow their posted privacy policies put the user at greater risk.

##### **5.1.2. Assigning a subjective score**

Several of the 150 questions we investigated for each app required a subjective analysis of risk to the end-user. We used two types of risk assessment: a consumer-level assessment -- described in the companion document “Mobile Health and Fitness Applications and Information Privacy” -- and a technical risk assessment described here.

The technical risk assessment looked at what data the app was collecting, storing and transmitting over the network. We assigned a score of 0 (no risk) to 9 (high risk), based on the information an app exposed to third parties. We then grouped those risk levels into “high,” “medium” and “low” risk buckets.

Risk levels were assigned subjectively. Here are some examples of what kinds of personal information were linked to each risk level:

- Risk level 9—Highest risk: address, financial information, full name, sensitive or embarrassing health (or health-related) information, information that a malicious actor could use to steal or otherwise cause a user to lose money
- Risk level 8—High risk: geo-location

- Risk level 7—Medium-high risk: DOB, ZIP code, any kind of personal medical information
- Risk level 6—Medium risk: risk evaluated to be between level 5 and level 7
- Risk level 5—Medium risk: email, first name, friends, interests, weight, information that is potentially embarrassing or could be used against a person (e.g., in employment)
- Risk level 4—Medium risk: risk evaluated to be between level 5 and level 3
- Risk level 3—Medium-low risk: anonymized (not personally identifiable) behavioral tracking (e.g., app usage), device info, a third party knows the user is using a mobile medical app
- Risk level 2—Low risk: risk evaluated to be between level 3 and level 1
- Risk level 1—Low risk: any kind of anonymized data that does not include medical health-related data or personally identifiable information
- Risk level 0—No risk

## **5.2. Features/functions analyzed to reveal information practices**

### **5.2.1. Data collection: Data a user provides and other data an app collects**

We used the apps as a typical user might, and evaluated the privacy risks of the kinds of data the app collected, both through user input (e.g., the user's name, email address, gender, and other personal information) and from background/hidden collection (e.g., GPS or network-based location information which a user is unaware is being collected).

### **5.2.2. Data stored on the device**

Using the tools described in Section 4, above, we looked at what kinds of data an app stored on a device, whether that data was obfuscated or encrypted, and how easy it was for a malicious person with physical access to the device to acquire the data.

### **5.2.3. Data transmission: network communication**

Using the tools described in Section 4, we monitored all network communications including high-level HTTP/HTTPS transmissions, and low-level packet transmissions. We looked at what information was being transmitted by URLs, cookies, HTTP GET Requests, HTTP POST Requests, and AJAX connections.

## **6. Results: Data Flow Analysis: Collection, Storage, Transmission and Usage of Data by Mobile Health and Fitness Apps**

### **6.1. Non-obvious data flow analysis of mobile apps**

#### **6.1.1. Apps send data in the clear without user knowledge**

It's not always obvious to users what data is being collected. For instance, one well-known company's app lets users learn about particular drugs. What they don't tell you is that the names of drugs researched by users are sent to third-party advertisers, who can link that data to the user's web browsing history.

#### **6.1.2. Apps connect to many third-party sites without user knowledge**

We were shocked at how many of apps connect to third-party sites and services without informing users. For example, apps sent latitude/longitude or ZIP code data with all requests to help target users, but their privacy policies did not mention that personal location information was being collected and shared with third parties.

#### **6.1.3. Unencrypted connections expose embarrassing data to everyone on a network**

Analytics data sent unencrypted over HTTP could allow third-party viewers (e.g., on your local network) to see information about your behavior.

### **6.2. Types of data collection and their associated risks**

#### **6.2.1. Explicit collection**

User data collection happens explicitly when an app asks a user for information. Examples of this are questionnaires and account creation. Users are aware of this type of collection and can control the information they provide.

#### **6.2.2. Implicit collection**

Most users do not know or understand how much data app developers and third-party analytics sites collect implicitly as they use an app. One example of implicit data collection is **behavioral tracking**; that is, what content a user views, in what order and at what speed and frequency. In other words every page a user clicks on, in sequence, with a timestamp associated with every click.

**More than 75% of the free mobile health apps and 45% of the paid apps we researched use some kind of behavioral tracking, often through multiple third-party analytics tools.**

Of the apps that use third-party analytics, **most free mobile health apps and half of all the paid apps use more than one analytics service at the same time; some send usage data to as many as five different third parties.** Furthermore, **only 6% of free apps and 15% of paid apps send the data to third-party analytics services using encrypted SSL connections exclusively;** the rest send some or all of the analytics data in the clear over HTTP.

In our technical research we observed analytics information being sent to many different URLs, including:

google-analytics.com, flurry.com, 2o7.net, mediallytics.com, newrelic.com, fluentmobile.com, bugsense.com, crittercism.com, g2trk.com, wptrak.com, track.celtra.com, localytics.com, crashlytics.com, appsflyer.com, scorecardresearch.com

### **6.2.3. Implicit collection of data by third-party advertisers**

Most consumers do not know or understand how much data third-party advertisers collect implicitly while they are using an app. We found that **almost half of the free mobile health and fitness apps we researched use some kind of third-party advertising, yet only one paid app (5%) in our sample had any kind of advertising.**

Of the apps that use third-party advertising services, **some send usage data to as many as 10 or more different third-party advertisers in the first few minutes an app is in use.** In addition, **NONE (0%) of the apps send the data to third-party advertisers using an encrypted SSL connection;** all of the apps send the data to advertisers in the clear over HTTP.

### **6.2.4. Linking to external data sources such as ad networks, data brokers, etc...**

Data collection can happen without a user's knowledge when their data is linked with an external data source. For example, credit card companies and store loyalty card programs often sell user data. When a user provides some kind of personally identifying information, such as a name or address, the app developer can link that user's information with an external data source. This can happen with third parties such as advertisers; for example, Google's Doubleclick ad network can link a user's behavior across different sites.

### **6.2.5. Data collection by third-party login/user identification services**

Data collection may occur without a user’s knowledge by using a third-party site or service to provide account login and/or user identification and verification services. Examples are apps that let you log in to your Google+, Facebook, and Twitter accounts from the app.

## **6.3. Data transmission: types, recipients, and risks**

### **6.3.1. Data sent to app developers’ websites**

User data is often transmitted to app developers’ websites. We believe the user generally expects this, although for some apps it may not be obvious. Nevertheless, we found that **78% of the free mobile health and fitness apps and 40% of the paid apps we analyzed send data to the developer**. Of those apps, **only 15% encrypt all of their communications with SSL**; the rest send some or all of the data in the clear over HTTP. Only 53% of free apps and 44% of paid apps that send personally identifiable information (PII; e.g., name, email address, address, geo location, etc...) send that PII encrypted.

### **6.3.2. Data sent to third-party sites as part of the app**

User data is often transmitted to third parties that the developer uses to provide core functionality. Generally, users have no idea that when they use an app from one company that they are in fact often using products and services from others.

**52% of the free mobile health and fitness apps and 40% of the paid apps we analyzed send data to third-party sites as part of their core functionality**. Of these, **only 15% encrypt all of their communications with SSL**; the rest send some or all of the data in the clear over HTTP.

### **6.3.3. Data sent to third-party sites for analytics and advertising**

Most users don’t know that many apps—especially free apps with embedded advertising—often send unencrypted personal and usage data in the clear over HTTP for third-party collection of user behavior, usage analytics and targeted advertising. Sometimes the information is in keywords (e.g., one health app sent “sexual dysfunction” as a keyword to an undisclosed third party when we were looking at content about sexual dysfunction). Information is also sent via cookies (which often can track behavior across multiple sites), and sometimes user-supplied information is sent (birthday and gender are common examples). Data may be collected without a user’s direct knowledge (e.g., sending the longitude and latitude along with a “list of AIDS

support groups” request to third-party mapping services). And information embedded in web page URLs is often passed in the clear (see Section 6.3.4 below for examples).

The real risk is that many privacy policies do not cover third-party advertisers. Clicking on an ad takes a user to a new web page, with a separate and often hidden privacy policy.

#### **6.3.4. Data sent in the clear via URLs and GET requests**

Data keys in URL GET/POST info over unencrypted (non-SSL) links show what you're looking at to everyone who can see your web request. For instance, consider these URLs sent in the clear and often to third parties to see how much information is exposed:

```
http://site1.com/AIDS_support_groups/FL/Dade_County/Miami
```

```
http://site2.com/Provider_Search?condition=bleeding&location=penis&cause=STD  
&age=23&location=GA&city=Atlanta&latitude=33.65123123&longitude=84.42234234
```

### **6.4. Data storage**

#### **6.4.1. Local storage**

User data is often stored on the mobile device. Data is stored locally in an app’s “documents directory” (aka, the app’s documents sandbox) and is easily accessible if a device can be unlocked, either because there is no passcode or a malefactor can guess what it is. Also, on iOS, there are many programs available that easily give access to data stored on a device, such as DiskAid (<http://www.digidna.net/diskaid>). Access to the sandbox reveals a user’s data history. So for example, an app that stores a cache of all advertisements viewed and all URLs called could reveal personal medical and drug information, including URL arguments that contain the health condition a user might be searching for.

On Android, the process is a little harder. The device needs to be rooted and requires access to the Android shell, but it’s pretty straightforward to anyone who has done it once.

**83% of the free mobile health and fitness apps we researched store data locally on the device. Of the apps that store data locally, NONE encrypt the stored data.**

#### **6.4.2. Local removable storage and the risks of data stored on the mobile device’s SD card**

iOS devices do not use SD card storage—data the apps store goes to the device’s internal flash disk. That data is at risk if the device is stolen, but the thief must first get around Apple’s protections against access. Options to do this exist, but are technically complex or expensive to procure and are generally not used by common thieves looking to steal a device and resell it.

Android devices, however, often have external storage options, usually in the form of replaceable SD cards, and in most cases will continue to work if the SD card is removed. Data stored on the device itself has strong protection; a malicious person can access it, but that requires time, energy and possibly expensive tools.

Data on the SD card is at risk, though, if someone with physical access to the device removes the card. No protections limit access to an SD card if it is physically removed from a device. For that reason, we recommend users of apps that collect and locally store data either: (1) avoid using devices with SD cards; (2) make sure their apps only store data on the internal storage; (3) remove their SD cards when using the medical apps (so they don't store data on the SD card); (4) or at least check the SD card periodically to see what data is stored and determine the risk of allowing it to remain on the SD card.

#### **6.4.3. Data stored externally**

User data is often stored on the developer's website. It is impossible to know for sure how that data is stored and used. Users often have only the privacy policy posted on a site to rely on for information about the developer's practices concerning that data.

However, our analysis found that **of the sites that post a privacy policy, those privacy policies were less than 50% accurate in describing what was actually happening technically in the app.** The level of accuracy for the descriptions of the data storage policies on the developer's servers may be similar.

In reality, **users will probably never know what happens to their data once it is submitted to a developer site.**

#### **6.4.4. Cookies and other tracking and identification technologies**

In addition to the user data stored on the device, often there are cookies – persistent data commonly used by websites to track their visitors – and other tracking identifiers stored on the local device. We found that **79% of the free mobile health and fitness apps we analyzed use and store cookies and other tracking identifiers locally on the device.**

## **7. Results: Risk of Apps**

### **7.1. Overall technology risk assessment: high/medium/low risks**

- 28% of the apps were low to medium risk (risk levels 1-3: 12 of the 43 apps)

- 32% of the apps were medium (risk levels 4-6: 14 of the 43 apps)
- 40% of the apps were high risk (risk levels 7-9: 17 of the 43 apps)

## **7.2. Risk levels and risk assessments of all 43 apps: risk levels 1-9**

Technology risk levels were assigned according to the privacy risk of using the app, including what data it collected, stored or transmitted. The technology risk levels were assigned as described in the previous Section 5, “Analysis: Evaluating an Application’s Technical Privacy Risks.”

Risk levels of the 20 free (no-cost) apps reviewed:

- Low Risk: Free apps with technology risk level 1: 1 (4% of 43 free apps)
- Low Risk: Free apps with technology risk level 2: 2 (9% of 43 free apps)
- Low Risk: Free apps with technology risk level 3: 4 (17% of 43 free apps)
- Medium Risk: Free apps with technology risk level 4: 2 (9% of 43 free apps)
- Medium Risk: Free apps with technology risk level 5: 5 (22% of 43 free apps)
- Medium Risk: Free apps with technology risk level 6: 2 (9% of 43 free apps)
- High Risk: Free apps with technology risk level 7: 5 (22% of 43 free apps)
- High Risk: Free apps with technology risk level 8: 1 (4% of 43 free apps)
- High Risk: Free apps with technology risk level 9: 1 (4% of 43 free apps)

Risk levels of the 20 paid apps reviewed:

- Low Risk: Paid apps with technology risk level 1: 0 (0% of 20 paid apps)
- Low Risk: Paid apps with technology risk level 2: 2 (10% of 20 paid apps)
- Low Risk: Paid apps with technology risk level 3: 3 (15% of 20 paid apps)
- Medium Risk: Paid apps with technology risk level 4: 1 (5% of 20 paid apps)
- Medium Risk: Paid apps with technology risk level 5: 3 (15% of 20 paid apps)
- Medium Risk: Paid apps with technology risk level 6: 1 (5% of 20 paid apps)
- High Risk: Paid apps with technology risk level 7: 2 (10% of 20 paid apps)
- High Risk: Paid apps with technology risk level 8: 4 (20% of 20 paid apps)
- High Risk: Paid apps with technology risk level 9: 4 (20% of 20 paid apps)

### **7.3. Do apps do what their privacy policies say they do?**

On average, **only about 50 percent of free and paid apps posted a privacy policy and adhered to it.** The rest either didn't have a policy, or their technical behavior included things not covered in their privacy policy.

**Of the sites that post a privacy policy, those privacy policies were less than 50% accurate in describing what was actually happening technically in the app.**

Our analysis also showed that **apps with the most detailed privacy policies posed some of the greatest privacy risks.** Such privacy policies seem to be written primarily to protect the company developing an app, not the users; they cover all their notice requirements in a way that's difficult for an average user to decipher. The only way for a user to know how great a privacy risk an app may be posing is by doing a technical evaluation—something beyond the ability of almost all users.

### **7.4. Free vs. Paid: Paid apps still pose major privacy risks because they use HTTP, not HTTPS**

The single largest technical risk to users of mobile apps is developers' failure to use encrypted (HTTPS) connections for all network transmissions. Most apps send all or most of their data over unencrypted (HTTP) connections. We found that both free and paid apps generally failed to use HTTPS (SSL-encrypted) connections when sending privacy-sensitive data over the network.

## **7.5. Apps with advertisements pose a significantly higher privacy risk**

Advertisements were the second largest source of privacy risk in mobile apps, as described elsewhere in this document. Privacy-sensitive users should avoid using apps that have embedded advertisements.

### **7.5.1. Advertisements are more likely to be found in free apps**

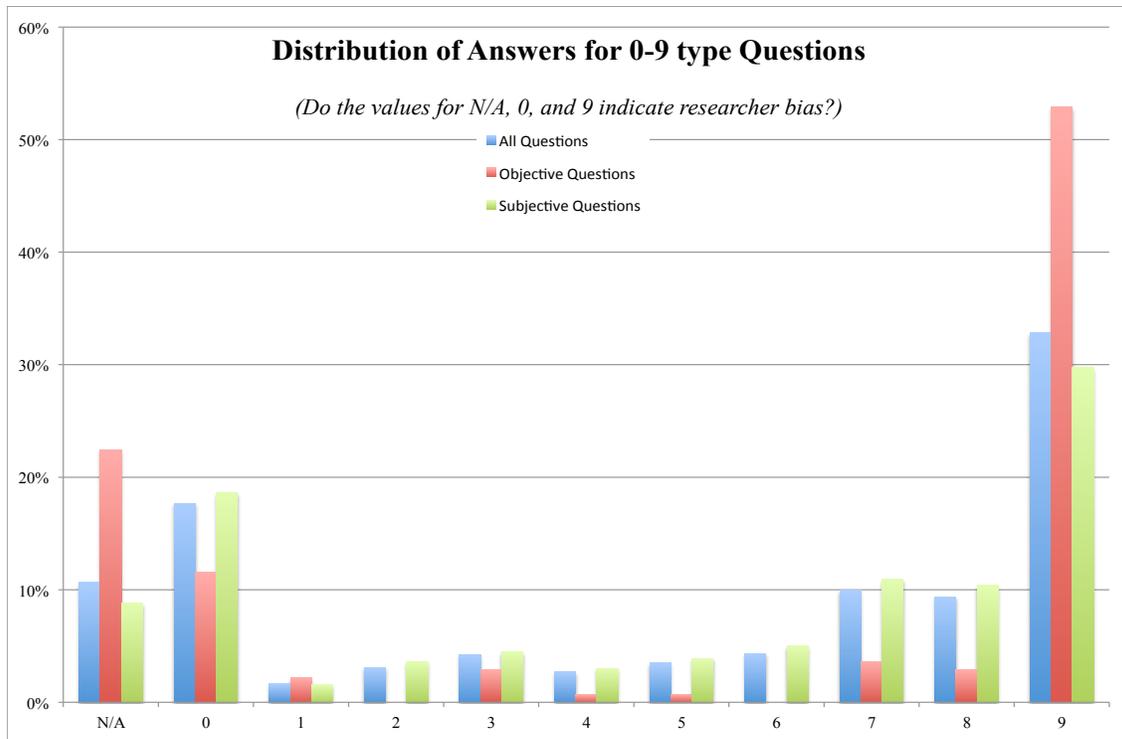
Free apps rely on advertising for revenue. Privacy-conscious mobile app users should consider restricting themselves to paid apps. Our research found that it didn't matter how much an app costs, only that paid apps generally carried a somewhat lower privacy risk because of the absence of ads. The correlation is logical, because ads make more money for apps that provide more detailed information about their users. Paid apps have less need of revenue from advertisers, so very few of them have ads.

This does not mean that paid apps pose any less privacy risk overall, because they are just as likely as free apps not to use HTTPS/SSL encryption to transmit sensitive personal data.

## 8. Problems Encountered

### 8.1. Our spreadsheet questions and/or answers are biased toward the edge cases

Notice in the following distribution of answers for the 0-9 questions how skewed the answers are to N/A, 0, and 9. This possibly indicates one or both of (1) a researcher bias -- a tendency to answer in the extreme, and (2) a bias in how the questions were written to favor answers in the extremes. Although beyond the scope and limited budget for this project, future studies of this kind may want to work on ways to identify and remove these potential biases.



### 8.2. Temporal: Apps change constantly, making it possible to analyze only a moment in time

Mobile apps are always changing. By the time this report is released, many, if not most, of the apps we analyzed will have been updated with new features and functionality, which will likely affect the privacy implications of using the apps. The only way to know for sure what an app is doing on a given device at a moment in time is to analyze it.

### **8.3. Context: Some apps that have a specific medical purpose can't really be evaluated in a general context**

We were unable to fully analyze some apps due to our lack of experience with a particular medical condition or field of medical inquiry. For example: how can we fully evaluate a diabetes app if we don't have diabetes? We tried our best to imagine what use cases might be like for typical users, but we know our experience is limited by our own lives. With medical apps particularly, users should try to figure out what an app is doing before entering personal data, but without a technical analysis, it's impossible to know the full extent of an app's data practices.

### **8.4. Comparing apps on iOS vs. Android is like comparing apples to rainbows**

Some apps have settings in the app, some have them on external websites, some iOS apps use the iOS "settings" app, some apps allow users to indicate their privacy permissions in the device settings—there's no single place to look.

Android and iOS also have different assumptions and development models about how privacy and user permissions work – it is impossible to compare their data privacy and security practices directly. We tried to resolve this by asking higher-level questions about the apps, but future efforts might want to analyze individual platforms independently instead of trying to do a combined research and analysis as we've done in this report.

### **8.5. Bias: We instinctively avoided the most seemingly egregious apps**

We only had time and budget to test a total of 43 apps. As a result, there were many apps we did not even consider testing because we they were obviously badly designed, functionally incomplete, or we suspected they were fronts for viruses or other malware. Although it was outside the scope and budget for this project, future studies may want to look at what percentage of the mobile health and fitness apps available suffer from these flaws.

### **8.6. A complete analysis of all platforms/screen sizes/OS versions was not possible**

It's impossible to get every device and test every possible downloadable app, but we know there are differences between different devices (e.g., iPad and iPhone versions of the same app have potentially different behaviors), although we didn't see these differences in the apps we tested.

We believe that our results are applicable for the vast majority of the iOS and Android devices in use, but future efforts might want to test on more devices and screen resolutions to verify this.

### **8.7. We were unable to evaluate long-term risk and risk of future app updates**

Many apps' functionality required long-term usage for full benefits to be evaluated (e.g., weight loss or exercise apps). This project lacked sufficient time and budget for long-term analysis of any app; everything was evaluated (relatively) quickly with little or no follow-up or long-term usage.

That said, we believe we were able to analyze and report on the majority of the apps' functionality and privacy risks in our testing. Our only major long-term concern is what happens when apps are updated and their functionality changes. It is up to users to review privacy policy updates and look at what an app is doing after each upgrade to decide if a newer version of an app suddenly puts them at significantly higher risk.

### **8.8. We were unable to evaluate apps with currently buggy versions**

Some apps crashed on download or start-up so we couldn't evaluate them. It's possible this was a problem with the specific version of the app or the operating system or device we were using, or that we happened to download the app when a buggy version was in the app store, or that the app never worked. (One result of this is that our data is skewed to apps that actually worked.)

### **8.9. Should we be comparing apps only within the same “type” or “category” or market?**

For more valid results there should be a category-by-category comparison done with apps in the same category/usage space compared to each other. For example, does it make sense to compare an app that provides reference information from a published book with an app that samples blood glucose levels and stores the data in a third-party database? It might be more valid to compare information-only apps against similar apps, to see if apps in that category tracked users in any way and what they did with the data they collected; likewise, to compare diabetes glucose apps only with their competitors.

## **9. Appendix A: HOW TO: Privacy Aware Mobile App Development**

### **9.1. Executive Summary: A privacy HOW TO for mobile app developers**

This document identifies and describes several rules and problem-solving approaches to building mobile apps that focus on protecting users' information privacy.

We analyzed how a representative range of mobile health and fitness apps operate technically. After reviewing privacy policies, we installed and used the apps. We looked at what data applications stored locally on the devices, and what they sent over the network. We noted how closely an app's actual functions mapped to the description of the developer's information practices in any published privacy policy, and we identified technical practices that pose privacy risks.

Since most health and fitness applications require continuous input of user data to fulfill their purpose, there is an inherent risk of harm if that data is compromised. Developers can modulate the risk by how they collect, store and transmit data, but nevertheless, the more user data an app collects, and the more sensitive that data is, the greater the privacy risk to the user of that app.

The good news is that there are best practices for building a high degree of privacy protection into mobile applications. This document highlights what we believe are the most useful of such solutions.

The bad news is that **none of the apps we evaluated** followed all the best practices we describe in this paper. We do not believe this widespread lack of technical measures to insure privacy protection is due to malice or ill-intent, but rather because **mobile app developers generally do not know how to build privacy into apps** that collect, store or transmit privacy-sensitive user data.

This document is a summary of best technical practices for mobile app developers to insure the highest levels of privacy protection.

Although this HOW TO is intended for developers of mobile health and fitness apps in particular, the advice it contains applies to mobile apps in general.

### **9.2. Intended audience**

While our focus is on health and fitness apps, mobile application developers in general can benefit from the privacy-protecting suggestions in this HOW TO. Although it is written for a broad audience with a variety of backgrounds, it assumes a base level of technical knowledge

about how mobile apps are built, how they collect and store data, how they communicate data over the Internet, and how third-party tools and services like advertising networks and analytics services are integrated into them.

This document will also be useful for product managers as a checklist to review the code of their mobile apps.

### **9.3. Overview: Primary technical privacy risks to users of mobile apps**

#### **9.3.1. Unencrypted network communication: Always use HTTPS**

Many apps send personal and sensitive user information over unencrypted (HTTP) connections. Developers should insure that all network communication is encrypted. The easiest way to do this is to always use HTTPS; never use HTTP to transmit data from the app to a server on the Internet.

#### **9.3.2. Advertisers**

Apps frequently share personal information with third-party advertiser networks, either in keywords sent to advertisers to target their ads, or embedded in URLs sent to advertisers.

Ideally, developers should avoid using ads in their apps. If the developer's revenue model requires ads, however, network communications between the app and advertisers should not include any personal user information and should always use HTTPS.

#### **9.3.3. Analytics services**

Many apps share users' personal information with third-party data analytics services. Developers should avoid using third-party analytics if possible; if not, they should withhold personal information from analytics companies and fully anonymize all data shared.

### **9.4. Privacy checklist**

#### **9.4.1. Limit the collection, storage, and network transmission of user data**

Don't collect, store, or transmit over a network any data that is not required by an application's core functionality.

#### **9.4.2. Notify users of all data collection and explain its purpose**

Inform users clearly what data is being collected, when and for what purpose. The best way to do this is at the time of collection through a contextual pop-up notice.

If real-time notification is not possible, be sure to describe data collection practices in detail in the app's privacy policy. If the privacy policy is overly lawyerly, call out information about data collection and uses, and user access and controls, in a consumer-readable FAQ.

#### **9.4.3. Encrypt all network communication with SSL or stronger**

*\*EVERY\** single network connection should be encrypted. The simplest option is to use SSL connections for low-risk data. This includes all connections to the developer's servers and to third-party sites like advertisers and analytics.

#### **9.4.4. Use high-grade encryption for users' personal information**

Apps that use, collect and store, personal information should protect it with stronger encryption than just SSL. Even when data is encrypted, always use SSL connections to transmit it.

There are several encryption SDKs and tools available for use, many for free, some as part of the iOS and Android SDKs—too many options to list here. You'll need to find the best tool for your particular needs and budget.

#### **9.4.5. NEVER send user information in clear text**

All network communications should be sent encrypted over SSL. This includes connections to developers' servers and to third-party sites like advertisers and analytics.

#### **9.4.6. Always use POST, never GET**

Never send data as part of a GET argument, for instance:

```
http://site.com/upload?user=MyName&email=me@myemail.com&password=1234
```

This data will still be seen in the clear as it travels over an unencrypted HTTP network connection, the information is usually stored unencrypted in the traffic logs on the remote web server(s), and all local caches stored on the device will hold that URL data in the clear putting users at risk if their device is compromised.

Instead, send user data via another mechanism, either via POST over HTTPS (SSL-encrypted network connections), or something you've built using custom encryption and/or data obfuscation. In all cases, make sure that all user data sent over the network is encrypted from end to end.

#### **9.4.7. Salt and hash all passwords before sending or storing**

Never store or send a password as cleartext; always salt and hash the password when the user enters it and only store and send the resulting hash value. Should you ever need a password entered (e.g., to recreate the hash) ask for it and then store the new hash.

Never send a password in cleartext as part of a GET argument (even over SSL) or as a POST argument over HTTP.

#### **9.4.8. Do not expose information in URLs – obfuscate page names and data**

A number of apps we looked at expose private data in the URL itself, which is sent in the clear over the net and stored in local web caches on devices. For example:

```
http://site.com/aids/recently_diagnosed/support_in_new_york.html?diagnosed=Jan2013
```

Notice that this URL exposes the following personal and sensitive information:

- the user likely has AIDS,
- the user is probably in New York, and
- the user was probably recently diagnosed with AIDS in January of 2013

Compare with this URL:

```
http://mysite.com/get_page.cgi?page_id=141245?session_token=321
```

It is more difficult to gain any insight into the user from the second URL, which obfuscates both the page content and the user's personal data.

If it's necessary to send something like the date of diagnosis, do not send it on the GET request as in the first URL. Instead send it as a POST argument over an encrypted HTTPS (SSL) connection.

#### **9.4.9. Protect against URL replay attacks by using single-use or expiring URLs**

No amount of obfuscation of the URL can protect against a URL replay attack where an attacker simply opens the URL in their own browser to see the content.

To protect against this, make sure the URL is not reusable – i.e., it cannot be used by anyone but the user at the time the user requests it.

There are several technical solutions to the problem, but the simplest is to expire the URL so that it doesn't work after the first time a user loads it. Another solution is to use a cookie to store a one-time-hash of the session, and deny loading of the page without both the URL and the cookie.

Each technical solution to protecting against a URL replay attack has its pros and cons. Developers should choose the one that meets the specific requirements of both the app and their server setup, while maximizing the privacy and security protection of all user data.

## 10. Appendix B: Spreadsheet Used for Data Analysis: CSV Format

Here is the spreadsheet we used when analyzing mobile health and fitness apps, in comma-separated values (CSV) format. You can copy and paste this into any spreadsheet program, or copy into a text file and then open that text file in your spreadsheet program. A [clean version of this spreadsheet in XLS form \(http://www.privacyrights.org/mobile-medical-apps-privacy-evaluation-spreadsheet-report.xls\)](http://www.privacyrights.org/mobile-medical-apps-privacy-evaluation-spreadsheet-report.xls) and a [clean version of this spreadsheet in PDF form \(http://www.privacyrights.org/mobile-medical-apps-privacy-evaluation-spreadsheet-report.pdf\)](http://www.privacyrights.org/mobile-medical-apps-privacy-evaluation-spreadsheet-report.pdf) are also available.

```
Main Cat,Question,Data Type,Tech Q?,Subjective /Objective,Notes,,App 1,App 2,App 3,
Platform,"Name of device platform (iOS, Android, other)",text,N,O,,,,,
Notice/Awareness,Quality of app info on Website?,0-9,N,S,0 = no website,,,,,
Notice/Awareness,Quality of Privacy Policy posted on Website?,0-9,N,S,0 = no privacy policy on
website,,,,,
Notice/Awareness,Quality of Privacy Practices in Policy,0-9,N,S,,,,,
Notice/Awareness,Readability of Privacy Policy on Website,0-9,N,S,0 = no privacy policy on website,,,,,
Notice/Awareness,Quality of Privacy Policy Change Notification?,0-9,N,S,,,,,
Notice/Awareness,Quality of Notice of Collection?,0-9,N,S,,,,,
Notice/Awareness,Quality of Privacy Policy posted in App?,0-9,N,S,0 = no privacy policy on website,,,,,
Notice/Awareness,App Links to Website Privacy Policy,Y/N,N,O,,,,,
Notice/Awareness,Readability of Privacy Policy in App?,0-9,N,S,0 = no privacy policy in app,,,,,
Notice/Awareness,Quality of protection of Children`s privacy?,0-9,N,S,0 = not mentioned,,,,,
Notice/Awareness,Specifically discusses Children`s privacy?,Y/N,N,O,0 = not mentioned,,,,,
Notice/Awareness,Allows users under 13?,Y/N,N,S,0 = not mentioned,,,,,
Notice/Awareness,Privacy Policy (FIPPs): Quality of Notice/Awareness?,0-9,N,S,0 = not part of privacy
policy / no privacy policy,,,,,
Notice/Awareness,Privacy Policy (FIPPs): Quality of Choice/Consent?,0-9,N,S,0 = not part of privacy
policy / no privacy policy,,,,,
Notice/Awareness,Privacy Policy (FIPPs): Quality of Access/Participation?,0-9,N,S,0 = not part of
privacy policy / no privacy policy,,,,,
Notice/Awareness,Privacy Policy (FIPPs): Quality of Integrity/Security?,0-9,N,S,0 = not part of
privacy policy / no privacy policy,,,,,
Notice/Awareness,Privacy Policy (FIPPs): Quality of Enforcement/Redress?,0-9,N,S,0 = not part of
privacy policy / no privacy policy,,,,,
Notice/Awareness,Privacy Policy (FIPPs): Quality of Use/Disclosure Info?,0-9,N,S,0 = not part of
privacy policy / no privacy policy,,,,,
Notice/Awareness,Shares collected data with Advertiser(s)?,Y/N,N,O,,,,,
Notice/Awareness,Shares collected data with other apps on device?,Y/N,N,O,,,,,
Notice/Awareness,Shares collected data with unidentified 3rd Parties?,Y/N,N,O,,,,,
Notice/Awareness,Shares collected data in aggregate to marketers?,Y/N,N,O,,,,,
Notice/Awareness,Contact Info: Email address listed?,Y/N,N,O,Is there a support email address
listed?,,,,,,
Notice/Awareness,Contact Info: Phone # listed?,Y/N,N,O,Is there a support phone number listed?,,,,,,
Notice/Awareness,Contact Info: web-based form (outside of app)?,Y/N,N,O,Can you contact support with a
web-based form (not in-app)?,,,,,
Notice/Awareness,Contact Info: in-app contact option?,Y/N,N,O,Can you contact support from within the
app itself?,,,,,,
Notice/Awareness,Uses anonymized-/non-PII data for analytics?,Y/N,N,O,,,,,
Notice/Awareness,Shares anonymized-/non-PII data to 3rd party analytics agencies?,Y/N,N,O,,,,,
Notice/Awareness,Notifies user that Privacy Policy does not apply to 3rd party links?,Y/N,N,O,,,,,
Notice/Awareness,Notifies user that personal info made public is not protected?,Y/N,N,O,,,,,
Notice/Awareness,Explains reasons for app permissions (e.g. access to location)?,0-9,N,S,Describes what
permissions are required and why (i.e. doesn't rely on just the standard notifications),,,,,
Choice/Consent,Amount of Secondary use of info?,0-9,N,S,,,,,
Choice/Consent,Risk of Secondary use of info?,0-9,N,S,,,,,
Choice/Consent,Encouragement to share info with 3rd parties/secondary uses?,0-9,N,S,,,,,
Choice/Consent,Percent of data collection opt-in (after app install)?,0-9,N,O,,,,,
```

*Technical Analysis of the Data Practices and Privacy Risks of 43 Popular Mobile Health and Fitness Applications*

Choice/Consent,Can opt-in to data sharing with 3rd parties?,Y/N,N,O,"Is the default not to share with third parties? If so, can you opt-in to do so?,,,,,  
Choice/Consent,Can opt-out to data sharing with 3rd parties?,Y/N,N,O,Can you stop the sharing of data with 3rd parties?,,,,,  
Choice/Consent,Can use app without providing non-PII info?,Y/N,N,O,,,,,  
Choice/Consent,Can use app without providing PII info?,Y/N,N,O,,,,,  
Choice/Consent,Can use app without providing medically-sensitive info?,Y/N,N,O,,,,,  
Choice/Consent,Can stop the sharing of data with 3rd parties after opt-in?,Y/N,N,O,Can you stop the data sharing after you have opted in?,,,,,  
Choice/Consent,Can opt-out of data developer/vendor shares with 3rd parties?,Y/N,N,O,Can you stop the vendor/developer from transmitting data they have collected about you with 3rd parties?,,,,,  
Access/Participation,Ease of user access to collected data?,0-9,N,S,0 = No access,,,,,  
Access/Participation,Ease of user ability to change collected data?,0-9,N,S,"0 would me no ability, 1 = really hard (like send email, argue, etc...), 9 = super easy with some kind of obvious and easy-to-use interface in-app",,,,,  
Access/Participation,Percent of personal data user can change?,0-9,N,O,"9 = All, 0 = None",,,,,  
Access/Participation,Ease of user ability to delete collected data?,0-9,N,S,0 = No Access,,,,,  
Access/Participation,Percent of personal data user can delete?,0-9,N,O,"9 = All, 0 = None",,,,,  
Access/Participation,Ease of user ability to share collected data?,0-9,N,S,"e.g. to social networks, user contats, health care professionals, etc",,,,,  
Access/Participation,Percent of personal data user can share?,0-9,N,O ,"9 = All, 0 = None",,,,,  
Access/Participation,Ease of user ability to see their permissions?,0-9,N,S,"e.g. to see what the user has chosen to opt-in to, or opt-out from.",,,,,  
Access/Participation,Percent of user permissions that user can change?,0-9,N,O ,"9 = All, 0 = None",,,,,  
Access/Participation,Strength of active promotion of data sharing with 3rd parties?,0-9,N,S,"0 = does not promote, 9 = heavily promotes",,,,,  
Access/Participation,Does app facilitate sharing health info with health professionals?,Y/N,N,O,,,,,  
Access/Participation,"Requires login to 3rd party system (FaceBook, Twitter, G+, etc)?" ,Y/N,N,O,"Requires the user log into a 3rd party system like FaceBook login, Google+, Twitter login, etc..." ,,,,,,  
Integrity/Security,Network communication?,Y/N,N,O,Does the app communicate over the net?,,,,,  
Enforcement/Redress,Quality of Enforcement/Redress options?,0-9,N,S,"0 = none, 9 = best possible options" ,,,,,,  
Permissions,iOS: Requires Location Permissions,Y/N,Y,O,See Settings>Privacy on device; Leave blank for non-iOS devices ,,,,,,  
Permissions,iOS: Requires Contacts Permissions,Y/N,Y,O,See Settings>Privacy on device; Leave blank for non-iOS devices ,,,,,,  
Permissions,iOS: Requires Calendar Permissions,Y/N,Y,O,See Settings>Privacy on device; Leave blank for non-iOS devices ,,,,,,  
Permissions,iOS: Requires Reminders Permissions,Y/N,Y,O,See Settings>Privacy on device; Leave blank for non-iOS devices ,,,,,,  
Permissions,iOS: Requires Photos Permissions,Y/N,Y,O,See Settings>Privacy on device; Leave blank for non-iOS devices ,,,,,,  
Permissions,iOS: Requires Bluetooth Permissions,Y/N,Y,O,See Settings>Privacy on device; Leave blank for non-iOS devices ,,,,,,  
Permissions,iOS: Requires Twitter Permissions,Y/N,Y,O,See Settings>Privacy on device; Leave blank for non-iOS devices ,,,,,,  
Permissions,iOS: Requires Facebook Permissions,Y/N,Y,O,See Settings>Privacy on device; Leave blank for non-iOS devices ,,,,,,  
Permissions,Android: Risk of required permissions?,0-9,N,S,TBD: Should we have an individual data point for each of the 130 permissions? ,,,,,,  
Permissions,Android: Your applications information: Run at Startup,Y/N,Y,O ,,,,,,  
Permissions,Android: Your messages: Recieveve MMS / SMS messages,Y/N,Y,O ,,,,,,  
Permissions,Android: Network: View network connections,Y/N,Y,O ,,,,,,  
Permissions,Android: Network: Full network access,Y/N,Y,O ,,,,,,  
Permissions,Android: Network communication,Y/N,Y,O,allows app to view info about network connections ,,,,,,  
Permissions,Android: Affects Battery -- Control Flashlight,Y/N,Y,O ,,,,,,  
Permissions,Android: Affects Battery -- Control Vibration,Y/N,Y,O ,,,,,,  
Permissions,Android: Affects Battery -- prevent tablet from sleeping,Y/N,Y,O ,,,,,,  
Permissions,Android: Camera,Y/N,Y,O,"Allows app to take pix and videos w/ camera, at any time, w/o yr confirmation" ,,,,,,  
Permissions,Android: Storage: allows app to write to USB storage,Y/N,Y,O ,,,,,,  
Permissions,Android: System Tools: Test access to protected storage,Y/N,Y,O,allows app to test a permission for USB storage that will be available on future devices. ,,,,,,  
Permissions,Android: Location--network based (less precise),Y/N,Y,O ,,,,,,  
Permissions,Android: Location: GPS-based more precise,Y/N,Y,O ,,,,,,  
Permissions,Android: Contacts,Y/N,Y,O ,,,,,,  
Permissions,Android: Phone calls: read phone status and device ID,Y/N,Y,O,"allows app to access phone features of device, determine phone # and device IDs, whether a call is active and # connected to" ,,,,,,  
Permissions,Android: Modify or delete contents of USB storage,Y/N,Y,O ,,,,,,  
Permissions,Android: System tools: mock location sources for testing,Y/N,Y,O ,,,,,,  
Permissions,Android: Bluetooth,Y/N,Y,O ,,,,,,  
Permissions,Android: Your accounts,Y/N,Y,O ,,,,,,  
Permissions,Android: Your applications information: Retrieve running apps,Y/N,Y,O ,,,,,,  
Permissions,"Android: Social information: modify/delete contacts, read/modify call log" ,Y/N,Y,O ,,,,,,

*Technical Analysis of the Data Practices and Privacy Risks of 43 Popular Mobile Health and Fitness Applications*

Permissions,Android: Sync settings,Y/N,Y,O,,,,,  
Permissions,Android: Your Accts-lets app get list of accts known,Y/N,Y,O,,,,,  
Permissions,Android: Microphone,Y/N,Y,O,,,,,  
Permissions,Android: Modify system settings,Y/N,Y,O,,,,,  
Permissions,Android: Network: Connect/Disconnect from Wi-Fi,Y/N,Y,O,,,,,  
Permissions,Android: Network: Control NFC (Near Field Communication),Y/N,Y,O,,,,,  
Permissions,Android: Status Bar: Expand/Collapse status bar,Y/N,Y,O,,,,,  
Permissions,Android: System Tools: Modify System Settings,Y/N,Y,O,,,,,  
Permissions,Android: System Tools: Send sticky broadcast,Y/N,Y,O,,,,,  
Permissions,Android: Network: GooglePlay billing service,Y/N,Y,O,,,,,  
Cost,Initial cost,#,N,O,"Total cost in US Dollars, 0 = FREE",,,,,,  
Cost,Recurring costs (yearly),#,N,O,"Total cost in US Dollars",,,,,,  
Cost,Upsell Costs (expected),#,N,O,"Total costs in US Dollars of any in-app upsell costs expected to be incurred during the usage of the app.",,,,,,  
Developer Info,App Name,text,N,O,"Name of the app",,,,,,  
Developer Info,Developer Name,text,N,O,"Name of the company/person(s) developing the app",,,,,,  
Developer Info,App Version,text,N,O,"Version number of the app when reviewed",,,,,,  
Developer Info,Legal Jurisdiction,text,Y,O,"Name of the country the developer resides in (legal jurisdiction)",,,,,,  
Overall,Overall level of privacy invasiveness?,0-9,Y,S,"High-level, how invasive is this app?",,,,,,  
Overall,Subjective Risk Assessment?,0-9,Y,S,"How risky is it to use this app?",,,,,,  
Overall,Creepiness Factor,0-9,Y,S,"How creepy are the developers, how loud are the warning bells against using this app?",,,,,,  
Overall,Requires hardware?,Y/N,Y,O,"Does the app require an external piece of hardware?",,,,,,  
Overall,Legitimacy of company developing app?,0-9,Y,S,"How legit does the developer feel? Are they a real company?",,,,,,  
Overall,"Overall: Paid: value-for-money? Free: value-for-privacy-tradeoff? (Cost vs. Functionality not including privacy consideration): 0 = None, 5 = got what I paid for, 9 = got way more than I paid for",0-9,Y,S,"How legit does the developer feel? Are they a real company?",,,,,,  
Tech,# Domains: Total #: Analytics,#,Y,O,,,,,  
Tech,# Domains: Total #: Advertisers,#,Y,O,,,,,  
Tech,# Domains: Total #: Developer,#,Y,O,,,,,  
Tech,Total MB of app install on device,#,Y,O,,,,,  
Tech,How close does Privacy Policy fully & accurately describe practices?,0-9,Y,S,,,,,  
Tech,PII Risk of Analytics,0-9,Y,S,,,,,  
Tech,PII Risk of data sent to Advertisers?,0-9,Y,S,,,,,  
Tech,PII Risk of data sent to Developer,0-9,Y,S,,,,,  
Tech,Quality of Security of Communication?,0-9,Y,S,,,,,  
Tech,Ads Served By? (Advertiser names...),text,Y,O,"no adds: leave blank",,,,,,  
Tech,Description of PII stored in Documents Sandbox (leave blank if none).,text,Y,O,,,,,  
Tech,Domains: 3rd Party (core functionality),text,Y,O,,,,,  
Tech,Domains: Advertisers,text,Y,O,,,,,  
Tech,Domains: Analytics,text,Y,O,,,,,  
Tech,Domains: Developer,text,Y,O,,,,,  
Tech,iOS vs. Android: Different Behaviour? / Notes?,text,Y,O,,,,,  
Tech,Signed in differences: notes,text,Y,O,,,,,  
Tech,App sends different data/different domains/etc after signed in?,Y/N,Y,O,,,,,  
Tech,Connects to 3rd party sites for primary functionality?,Y/N,Y,O,,,,,  
Tech,Encrypted: \*ALL\* connections to developer encrypted?,Y/N,Y,O,,,,,  
Tech,Encrypted: \*ALL\* data sent advertisers encrypted?,Y/N,Y,O,,,,,  
Tech,Encrypted: \*ALL\* data sent analytics encrypted?,Y/N,Y,O,,,,,  
Tech,Encrypted: ALL user data sent developer encrypted? pt14256,Y/N,Y,O,,,,,  
Tech,Encrypted: Sends all PII encrypted?,Y/N,Y,O,"Leave blank if no login or N/A",,,,,,  
Tech,Encrypted: Sends login info encrypted?,Y/N,Y,O,"Leave blank if no login or N/A",,,,,,  
Tech,Heartbeat and/or Communicates in background when sleeping?,Y/N,Y,O,,,,,  
Tech,iOS: UDID sent? ,Y/N,Y,O,,,,,  
Tech,Library/[...]/Cache.db || AppSupport || etc has user info cached?,Y/N,Y,O,,,,,  
Tech,Sandbox on Device: any user data found in Documents Sandbox?,Y/N,Y,O,,,,,  
Tech,Sandbox on Device: user data found was encrypted?,Y/N,Y,O,,,,,  
Tech,Sandbox on Device: user data stored in Documents Sandbox (!cache)?,Y/N,Y,O,,,,,  
Tech,Sends data to someone not covered in privacy policy?,Y/N,Y,O,,,,,  
Tech,Sends data to the developer site(s) you'd expect? ,Y/N,Y,O,,,,,  
Tech,Sends user-generated or PII data to advertisers?,Y/N,Y,O,,,,,  
Tech,Sends user-generated or PII data to analytics?,Y/N,Y,O,,,,,  
Tech,Unique IDs always sent over encrypted connection?,Y/N,Y,O,,,,,  
Tech,Uses Cookies?,Y/N,Y,O,,,,,  
Tech,Overall Technical Score: How Close to the HOW TOs are their practices?,0-9,Y,S,,,,,  
Tech,Overall Technical Complexity Score,0-9,Y,S,,,,,  
Tech,Overall Technical Risk Score,0-9,Y,S,,,,,