Serge Egelman explains why AppCensus privacy checks help parents

Dr Egelman is co-founder of **AppCensus**, which builds tools to analyze the privacy behaviors of mobile apps

EVERYONE KNOWS WE'RE BEING TRACKED. WHY SHOULD IT MATTER TO PARENTS?

Online tracking is the bread and butter of the free Internet. In order to monetize online services, companies pay the operators of websites and mobile apps to show specific advertisements to specific users. Companies do this by inferring individual users' preferences based on data automatically collected from them, data such as the services they use, how they use them, where they live and work, and so forth. In short, online and offline activities are tracked, which allows companies to maintain detailed profiles of individual user behavior, which in turn is used to predict users' interests, preferences, and even demographics. In most cases, this data is used to show targeted advertisements, but in some cases that tracking data is sold to data brokers, who use it to augment profiles of the same users that were gathered from other sources. This more nuanced data is then on-sold to whomever is willing to pay for it.

Contrary to popular belief, the reason why you receive oddly prescient ads is not because your devices are secretly recording all of your conversations, but because of this type of inference: your online and offline activities are tracked, and then sophisticated algorithms use that data to make predictions about you. Tracking is made possible by "persistent identifiers." An identifier is any piece of information that allows an individual — or a device — to be uniquely identified.

"Persistent" identifiers are identifiers that tend not to change over time. For example, motor vehicles have persistent identifiers in the form of license plates: a license plate uniquely identifies a vehicle and vehicles tend to have the same license plates over time. Thus, if someone records all of the license plates at a particular place over time, they can determine how many times in that period any individual vehicle was there. Similarly, if license plates are recorded at many different locations and that data is combined into a single dataset, one could use that to reconstruct the movements of individual vehicles in that dataset. In short, combining a persistent identifier with information about where those identifiers were observed allows a data recipient to reconstruct an individual's activities. Using this knowledge, the data recipient can infer information about individuals' routines, preferences, demographics, and even relations and social connections!

This is precisely how mobile tracking occurs. Mobile phones have various identifiers associated with them, including some that cannot be easily changed (e.g., serial number, WiFi MAC address, IMEI, etc.). As mobile phones are very personal devices, a unique identifier for a mobile phone is consequently a unique identifier for that phone's user. Therefore, it can be used to collect data about the user's activities, preferences, and demographics, simply based on data collection that associates it with the apps that were used, when, how, and where.

Why does this matter? By and large, this data is used for advertising purposes: these profiles are used to decide which ads to show which users. However, the data is increasingly used for other purposes that are often completely opaque to consumers, particularly parents. For example, location data collected by apps is frequently resold to other businesses and used for everything from predicting

social relations in the physical world, predicting retail sales trends, law enforcement surveillance, and even for political fundraising and advocacy.

Worse, new uses for this type of data are invented all the time, which means that there's no way of knowing exactly how collected data may be used in the future. Data collected from mobile apps and other services could end up being used for making major life decisions. When this data comes from children, it is obviously even more concerning.

What could happen if we do nothing?

In the extreme case, this data could put someone at risk for their physical safety. For example, if real-time location data were disclosed, or location data sufficient to determine routines, such as one's residence, or place of schooling or employment, it could be used for stalking. But most likely, this data will be used to build profiles of people's inferred interests and demographics, so that others who buy those profiles can target individuals with advertising and other messages (e.g. political messaging).

Because the data is collected automatically and is used to make assumptions about people, it can and will have errors. These errors could lead to erroneous and/or unfair decisions, such as denial of credit or employment; some may have legal repercussions. This is why it's important that there be transparency into what information is being collected and with whom it is being shared. Without knowing that information is being collected, it's not easy to correct any resulting errors, much less request that collection stop.

In many cases, information is collected from apps and services for perfectly legitimate reasons. Nonetheless, by understanding when it's occurring, you can make informed privacy decisions for yourself and on behalf of your children.

A BRIEF OUTLINE OF WHAT THE APPCENSUS SYSTEM IS SET UP TO DO AND WHAT IT CAN FIND.

AppCensus analyzes the behaviors of mobile apps by relying on a combination of "static" and "dynamic" analysis.

- **Static analysis** refers to analyzing programs without running them, in order to detect the presence of specific instructions that *may* be executed when the program is run. For example, static analysis can be used to answer the question, "Does the program include a particular function?"
  Static analysis is quick, because it does not involve interactively running the program. However, it is prone to false positives, as not all program code is reached during execution (i.e. not every function will necessarily be executed every time the program is run); code detected through static analysis may never get executed in practice.

- **Dynamic analysis**, on the other hand, refers to running programs to directly observe the programs' behaviors. For example, dynamic analysis can answer the question, "What functions does the program actually use?"
  Dynamic analysis more realistically models program behavior, as the program is executed in a testing environment that is designed to model real-world usage. However, it is prone to false negatives: program code not executed during the testing period may be executed under different conditions. Dynamic analysis is desirable because it does not yield false positives: conclusions are observations of *actual* program behavior during the testing period.

AppCensus performs dynamic analysis on apps to examine whether personal information would be transmitted over the Internet during the course of realistic app usage. During testing, our automated

system installed apps on smartphones and then human users interacted with them. We examined the network traffic generated by the apps in order to detect the transmission of personal information (e.g., persistent identifiers, photos, contacts, etc.). Additionally, we ran the apps with our own modified version of the operating system, which included additional instrumentation to monitor how apps attempted to access sensitive data stored on the device, including usage of the Android permissions system. Finally, we performed static analysis of the apps to identify bundled third-party software development kits (SDKs) and the use of various privacy-related functions and settings.

The test devices were Google Pixel 3a smartphones running a modified version of AOSP 9.0 (The Android Open Source Project, or AOSP, is an open source branch of the Android operating system) located in Australia. We modified the operating system by instrumenting the permission-checking Application Programming Interface (API). This means that through using AppCensus's modified version of the operating system, whenever an unmodified Android app attempts to access a resource protected by Android's permissions system, our instrumentation makes note of this. Thus we can understand which apps accessed protected user information during testing. This allows us to monitor the execution of individual apps without having to modify them.

In testing Android apps, AppCensus instrumentation also monitored the payloads of network traffic, allowing us to examine even Transport Layer Security TLS-encrypted traffic. Unlike other app- and traffic-monitoring tools, AppCensus instrumentation was generally invisible to apps. We monitored apps' transmissions for the presence of information that indicated that they did not detect that they were running on "rooted" or otherwise modified/monitored devices.

We tested each app a minimum of two times: first automatically (with a robot tester) and then with a real human using each app. Our testing procedure was to first install an app by downloading its newest version from the Google Play Store or by "sideloading" it, which is the process of transferring an app from a computer via USB cable to the testing phone running our instrumentation. Once the app was installed, it was launched. The testing device was logged into a Google account associated with that device (as is the norm for Android).

We first used the robot tester to automatically interact with the app (i.e., by performing random "clicks" and "swipes") while collecting data on the app's behaviors while it was being used for a 10 minute time period. Lastly, the logs generated during testing were downloaded from the phone for analysis. Then the entire testing procedure was repeated, but instead of the automated robot tester, a human tested the app for an additional 10 minute time period.

Finally, we examined all of the network traffic generated by the app during the testing periods to examine what personal information it transmitted (both back to its own servers, as well as third party servers), as well as to whom this information was transmitted. The information we present here documents the data types transmitted, the recipients, as well as the permissions used and requested by the app during testing and any third-party code that could be used for data collection.

In short, the AppCensus testing system allows us to see:

- *What data do apps have permission to access?*
- *What data do apps actually access?*
- *To whom is that data sent?*
- *Is that data sent securely?*

Why is it important for parents to know about the AppCensus system?

Currently, mobile apps are under no obligation to provide their users with useful information about their data collection and data sharing practices: even laws like the General Data Protection Regulation (GDPR) in Europe only require that privacy policies list broad categories of recipient, which makes it nearly impossible for consumers to identify those recipients, much less their privacy practices. The goal of AppCensus is to help people make sense of mobile app privacy risks, not via privacy policies, but by presenting them with empirical examples of how those apps actually behave.

The data we present show not just how apps behaved during testing, but also how these apps *might* behave under different circumstances. That is, our primary findings are the types of data that we observed being sent by the apps to various recipients, however, the permissions and embedded third-party code both indicate what an app is *capable* of doing.

Just because we did not observe an app transmit user information to a data recipient during testing does not mean that it might not occur under different conditions. Thus, to attain another measure of the likely recipients of user information from mobile apps, we examined the software development kits (SDKs) present in the apps. SDKs are third-party software components that app developers bundle within their apps to provide certain functionality. While some of this functionality may be in the service of providing primary app features, other SDKs may collect user information for secondary purposes. Because third-party SDKs embedded in an app have access to the same data and system resources as the host app, they can potentially collect a lot of sensitive user data; in some cases, the app developer may not even be aware of the data being collected by a third-party SDK. Therefore, measuring the prevalence of the most popular SDKs among Android apps provides a metric for *potential* data collection from mobile apps.

What is the significance of each of the AppCensus findings?

- *What sort of risk does each app pose?*
- *What level of risk does each app pose?*
- *When is this a threat? When is it not?*
- *Can each 'risk' can be avoided? If so, how can it be avoided?*

Overall, AppCensus data show that tracking is ubiquitous / found everywhere on the Internet and that parents are not given enough information to make informed decisions about their own privacy, much less their children's. Right now, the primary tool that parents are expected to use to make decisions about online privacy is a website privacy policy. But these documents are generally ambiguous, couched in terms of what companies "may" do with personal data, rather than unequivocally specifying what a company *will* and *will not* do with that data.

The goal of AppCensus is to help parents make more informed decisions about their children's privacy by showing them exactly what data various apps collect, and with whom these apps were observed sharing it. One major limitation of our approach is that our visibility is limited to where data gets sent by an app, which means that we cannot be certain of how it is used after it leaves the mobile device, nor if it is subsequently re-shared with additional recipients, beyond the first. Nonetheless, one can make assumptions about the intended use of the shared data based on the services offered by the recipient. For example, personal data that is observed being transmitted to advertising companies is likely to be used for targeted advertising purposes.

The question is: When are these data flows dangerous?

Unfortunately, the answer is that it depends. The permissions that an app has been granted indicate the types of data that it *may* access. Having this information is useful, because it allows you to reason about whether or not it is appropriate for the app to be accessing that type of data, based on your knowledge of what the app is supposed to do. However, it is a lot more nuanced than labeling an app as being good or bad simply based on the types of data that it *may* access. For example, it is obvious why a mapping app might want access to location data, whereas it is less obvious for a flashlight app.

Nonetheless, just because you can't personally think of a reasonable justification for a certain permission request does not mean that one does not exist! For example, it might not be clear why a game requires access to your address book contacts, until you realize that it includes a feature to invite social contacts. It might be unobvious why a coupon app needs access to the device's camera, until you realize that that permission is needed to scan barcodes. As a result, making privacy decisions based simply on what data an app *might* access is an imperfect strategy. In our analysis, we provide information about app permissions, in order to indicate what an app might do, but more importantly we provide information about what data was observed *actually* being transmitted from the device and to whom.

By testing the apps and monitoring their network connections through using our instrumentation, we are able to document the privacy behaviors that users — including children — are likely to encounter during normal use. Nonetheless, these results are also nuanced: the transfer of personal information for advertising purposes generally involves many different recipients who bid for the user's attention. This means that depending on the precise circumstances, the entities receiving personal information from mobile apps may change from one moment to the next. Similarly, during our testing, we simulated user behavior by manipulating app user interfaces, but it is possible, or even likely, that we failed to test certain app features that would have yielded additional behaviors of interest.

When an app is found to have particular 'risks', is it okay still to use it with parental guidance or particular settings to mitigate the 'risks', or should the app be avoided?

That really depends on what the risks are. As with all risky activities, sometimes there are ways of mitigating the risk without having to avoid the activity. There are various device settings, which, when enabled, mitigate many of the privacy concerns that stem from reputable apps. For an otherwise reputable app, the main risk is that a child will be profiled and targeted with ads. This risk can be mitigated by configuring the device to always opt out of behavioral advertising.

For example, many apps collect persistent identifiers for the purpose of profiling individual users in order to target them with ads. However, the policies of both Apple and Google require app developers to use a resettable identifier for this purpose. The reason for this policy is that both platforms allow users to use a system setting to either reset this identifier—akin to clearing cookies from a web browser—or to opt out of profiling/targeted advertising altogether. Reputable ad networks will honor this setting. However, you must actually change the setting as the default on both Android and iOS is for this setting to be disabled: by default,[1] users will be subjected to profiling/behavioral advertising). Instructions on how to do this on Android can be found here, while instructions for iOS are here.

Similarly, these types of system settings can be used to restrict what data apps can access via the permission system. For example, if your primary concern is the collection of location data or access to the phone's camera, location services can be disabled and the camera permission can be revoked.

Finally, both platforms also require that children's apps do not engage in behavioral advertising. Thus, if your child uses apps that are listed in either the "Kids" category within the Apple App Store or the

[1] This is about to change on iOS, once Apple begins enforcing an opt in system for behavioral advertising.

"Design for Family" program within the Google Play Store, these are less likely to collect data for secondary purposes (though because these policies are not always enforced, it is by no means a guarantee).

IN SUMMARY, WHEN CHOOSING APPS FOR CHILDREN:

1. Limit the selection to apps found within the Design for Families section of the Google Play Store (Android) or the Kids section of the Apple App Store (iOS). These programs have more stringent privacy requirements than apps for a general audience.

2. Use the child's device's system settings to opt out of targeted advertising. This way, even if third party data recipients do not know to correctly apply policies for handling children's data, they are nonetheless instructed to not use received data for profiling/targeting purposes.

3. Use system-wide privacy settings to restrict what data certain apps are allowed to access. For example, if you do not believe a certain app should have access to location data, deny the app that permission or disable location services altogether.

****

AppCensus is working in conjunction with the Australian Council on Children and the Media to deepen parents' understanding of the traps embedded within apps and to provide strategies for reducing the harm that children may be exposed to through their app use.

Return to the Apps can track page for links to other supportive ACCM resources.

ACCM acknowledges the support of the
Australian Communications Consumer Action Network (ACCAN)
which funded the research for, and publication of, this document.