Poster: AppPrivacy: Analyzing Data Collection and Privacy Leakage from Mobile Apps

Minjie Zhu*, Qingqing Ye*, Xin Yang*, Xiaofeng Meng*‡, Haibo Hu†,
*School of Information, Renmin University of China
zhuminjae@hotmail.com, {yeqq, yangxincps, xfmeng}@ruc.edu.cn
†Department of Electronic and Information Engineering, Hong Kong Polytechnic University
haibo.hu@polyu.edu.hk

***Abstract*—While collecting legitimate usage data, many mobile applications (apps) have reportedly posed privacy threats to their hosted mobile devices and individuals, who are, unfortunately, unaware of data leaks and measures to protect themselves against these leaks. In this poster, we present a system that analyzes the two sides of mobile application ecosystem — data collection and privacy risk. The system consists of three main modules that correspond to mobile apps, users and service providers, respectively. To the best of our knowledge, this is the first work to evaluate privacy risk by analyzing data collection and privacy leakage from mobile apps.**

I. Introduction

With the growing accessibility of mobile devices and the Internet, various apps have been developed to provide all kinds of Internet services, such as shopping, banking and gaming. Apps run on mobile platforms are expected to provide personalized services to users with the collection of individual information, which can lead to potential privacy leakage. Even worse, sensitive personal information could be misused and potentially leaked to third-parties. For instance, it is reported that 73% of Android apps share email address with third parties and 47% of iOS apps leak user location data [1]. Fig. 1 illustrates the current practice of user data collection from mobile apps. When users run apps on mobile devices, their personal data leave the devices through these apps (step ①) and are then collected by various service providers (step ②).

Even though privacy concern of mobile apps has received increasing attention, individuals rarely have clear knowledge of privacy violation because their data are silently collected and transferred away. With General Data Protection Regulation (GDPR) enforced in EU from May 2018, there is a compelling need for users to safeguard their personal data. Therefore, it is necessary to deliver data flow messages to users and guarantee the transparency of data collection.

Goal. The goal of this research is to identify potential privacy leakage by apps and analyzing user data flow to make the whole data collection transparent and recorded. We aim to give insights in data collection and privacy leakage among mobile apps, users and service providers.

Contributions. To achieve this goal, we present a system to analyze the whole data collection and quantify privacy leakages from mobile apps. Our research makes two contributions as follows. First, we design three modules to identify and track data collection among apps, users and service providers. Second, we use four existing methods (see Section II) to identify sensitive data collection and propose a quantification method for privacy risk.

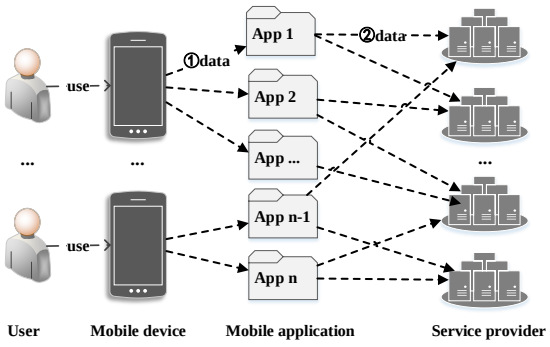

Fig. 1. User data collection from mobile apps

II. Preliminary: Data Leak Analysis

The first thing of privacy leakage analysis is to identify data flow from apps. Prior work on identifying data leaks can be categorized into four types.

Privacy Policy Analysis. Privacy policy is offered to show how a party collects, processes, uses, discloses and stores data. With natural language processing (NLP) techniques, data-related entities can be extracted from privacy policies, and used to identify data sent out of mobile devices [2].

Permission Analysis. Permission request is the way for an app to apply for system resources or user data, which is directly linked with privacy leakage [3]. Mobile operation systems such as Android and iOS adopt permission-based security scheme, which checks the permission requests of an app before installation or during usage.

Static Code Analysis. Static analysis leverages the Android APIs which provide more extensive and accurate data flow information at fine-grained level [4].

Dynamic Analysis. Dynamic analysis is used to monitor network data traffics between apps and service providers in realtime, including dynamic taint tracking, virtual private network, and man-in-the-middle Wi-Fi network [1], [5].

III. Design and Implementation

In this section, we discuss the design and implementation of our system. Fig. 2 shows the overall architecture which consists of three main modules.

‡ Corresponding author is Xiaofeng Meng

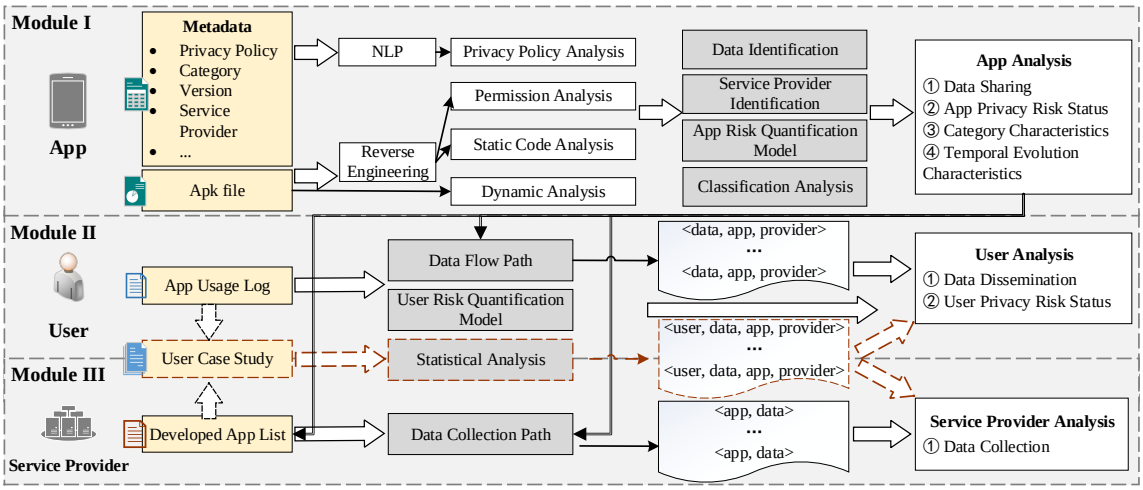

Fig. 2. The architecture of AppPrivacy system

A. Module I: App Analysis

This module focuses on analyzing data sharing status of various apps. With app metadata and apk file (for Android apps) as input, this module aims to identify collected sensitive data and service providers. Then a value of privacy risk caused by each app is calculated using a risk quantification model. Besides, this module also analyzes other relevant characteristics of data collection, such as data gathering specification under a certain app category and evolution of data collection behavior of different versions.

- **Data Identification.** According to the four methods described in Section II, data-related entities, permissions, API calls and data traffics form all possibilities of data collection.
- **Service Provider Identification.** Registered API and service provider can be extracted from *AndroidManifest.xml* in a form of $\langle API, provider \rangle$ map.
- **App Risk Quantification Model.** Based on EBIOS method [6], we quantify the privacy risk of an app from three aspects of identified data, i.e. the sensitive level of data, usage features under a certain app category and the number of service providers that an app is related with.

B. Module II: User Analysis

According to app usage log of a user, this module puts emphasis on user data propagation. Then a value of user privacy risk is calculated by the user risk quantification model.

- **Data Flow Path.** By app analysis results in Module I which contains shared data and service providers of each app, this module can get the data flow path of a user, i.e. $\langle data, app, provider \rangle$.
- **User Risk Quantification Model.** The privacy risk of a user is calculated from two aspects of identified data, i.e. the sensitive level and the number of service providers.

C. Module III: Service Provider Analysis

This module aims to determine the amount of collected sensitive data by a service provider via all apps it develops, operates or supplies functional APIs for.

- **Data Collection Path.** Based on app analysis results in Module I, we analyze the data collection path of each service provider, i.e. $\langle app, data \rangle$.
- **User Case Study.** This work uses sampled anonymous device data with installed apps to perform statistical analysis of each user, i.e. $\langle user, data, app, provider \rangle$.

IV. Conclusion and Future Work

This poster presents a novel system to analyze the whole data collection and privacy leakage among users, apps and service providers. We have gathered about 300 thousand apps and a large sample pool of 30 million anonymous devices which supply app usage logs to build the prototype system. Through permission analysis, we have identified 39 types of sensitive user data, such as location and contacts. Preliminary evaluations have shown that apps under different categories have different data collection tendencies, and that top Internet service providers have collected the most data.

We are currently in the middle stages of implementing this prototype, with many open questions to explore. We have finished permission analysis and will explore more data identification methods. Besides, optimization of privacy risk quantification model is also one of our future work.

References

[1] J. Zang, K. Dummit, J. Graves, P. Lisker, and L. Sweeney, "Who Knows What About Me? A Survey of Behind the Scenes Personal Data Sharing to Third Parties by Mobile Apps," *Technology Science*, 2015.

[2] R. Slavin, X. Wang, M. B. Hosseini, J. Hester, R. Krishnan, J. Bhatia, T. D. Breaux, and J. Niu, "Toward a Framework for Detecting Privacy Policy Violations in Android Application Code," in *ICSE*. ACM, 2016, pp. 25–36.

[3] W. Enck, M. Ongtang, and P. McDaniel, "On Lightweight Mobile Phone Application Certification," in *CCS*. ACM, 2009, pp. 235–245.

[4] K. W. Y. Au, Y. F. Zhou, Z. Huang, and D. Lie, "Pscout: Analyzing the android permission specification," in *CCS*. ACM, 2012, pp. 217–228.

[5] W. Enck, P. Gilbert, B.-G. Chun, L. P. Cox, J. Jung, P. McDaniel, A. N. Sheth, S. Han, V. Tendulkar, B.-G. Chun, L. P. Cox, J. Jung, P. McDaniel, and A. N. Sheth, "TaintDroid: an information-flow tracking system for realtime privacy monitoring on smartphones," *TOCS*, vol. 32, no. 2, p. 5, 2010.

[6] "Methodology for privacy risk management," *Commission Nationale de Linformatique et des Liberts*, 2012.