Proceedings of the Eighth International Symposium on
Human Aspects of Information Security & Assurance (HAISA 2014)

Reengineering the User: Privacy Concerns about Personal Data on Smartphones

M. Tsavli[1], P.S. Efraimidis[2] and V. Katos[2]

[1]Dept. Digital Systems, University of Piraeus
[2]Dept. Electrical and Computer Engineering, Democritus University of Thrace
e-mail: matina.tsavli@gmail.com; {pefraimi, vkatos}@ee.duth.gr

Abstract

Smart mobile devices carry an enormous amount of sensitive personal data of their owners. The access to this data by mobile operating systems and mobile applications is regulated by a corresponding security and permissions framework. In this paper we discuss the privacy and security concerns that have risen from the permissions model in the Android operating system, along with two shortcomings that have not yet been addressed up to date. We focus on personal data and propose a smartphone data taxonomy as a tool to highlight these concerns. Additionally, we study the impact of the applications' evolutionary increment of permission requests from both the user's and the developer's point of view and finally, we propose a series of remedies to the privacy and security corrosion.

Keywords

Personal data, privacy concerns, smartphone data taxonomy, user awareness, user's un-training, security issues

1. Motivation

As smartphone usage and capabilities grow rapidly, more complex operating systems and applications are developed in order to meet the user's ever increasing needs. Apart from the traditional mobile phone functionalities such as voice calls and text messaging, smartphones offer a variety of capabilities such as GPS services, email services, video recording, web-browsing and third-party apps (throughout this document we will use the term "app" as an abbreviation for mobile applications). Huge amounts of personal data are generated and stored on the smartphones such as location traces, usage logs, contacts, photos, documents, calls and messages. Each data type serves a series of purposes ranging from the enrichment of its functionalities in order to improve the user experience, to formatting, publishing or just safe-keeping of the data. Even when the smartphone is not used, it produces personal information about the user such as location traces, date-time logs of smartphone activation or shutdown. These potentially sensitive pieces of data are often collected from the operating system or the apps in order to support their functionality requirements. This requires the user giving his consent to these apps to access his personal data as dictated by the permissions model. As there is currently no unified applicable security policy to consolidate the data flows, which data can be accessed, by whom and for what purpose, many privacy and security concerns arise. Additionally, the users are becoming conscious of this matter and a number of tools

have already started to appear on the market. A representative tool is TaintDroid, which examines thoroughly the personal data flows by analysing the operations of the underlying app (Enck et al., 2010). An approach to assess the risks of installing an app based on the category and the permission requests of the app is presented in (Sarma et al., 2012).

Our motivation is to address the necessity for effective and in-depth control of the user's personal data flows. More specifically, due to the diversity of the data sources and the value of personal information, we suggest a data taxonomy based on the actors that have or request access to the user's personal data. Additionally, there seems to be currently a considerable amount of obscurity on the permission requests by the apps as well as the way these apps manipulate user's personal data albeit the fact that legal frameworks exist in many countries that specify how the personal data are supposed to be handled. Finally, there is a tendency of users and developers to unsubscribe from security awareness related actions; the former could be due to fatigue of the consecutive acceptance of more and more permission requests and the latter due to the high complexity of the permissions model. In (Felt et al., 2012), the authors describe an approach of users' attention, comprehension and behaviour as "warning fatigue" for gradually losing their privacy concerns, while in (Balebako et al., 2014) the authors conclude that developers lack awareness of privacy measures and make decisions in ad hoc manner.

This research is organized as follows. In Section 2 we provide a smartphone data taxonomy according to the entities that have or request access to user's private data. Section 3 discusses the privacy and security concerns that have risen from the permissions model in the Android operating system, along with two shortcomings that have not been addressed before. In Section 4 we study the impact of the apps' evolutionary increment of permission requests from both the user's and the developer's view. Finally, in Section 5 we critically reflect upon the ways that personal data have been manipulated and we provide suggestions to address the current state of privacy and security issues.

2. Data taxonomy on smartphone devices

Every smartphone consists of a multitude of components and structures that combined provide a series of functionalities offered to the user. These components are hardware resources, network services, informational data and application services and constitute the assets of the smartphone. These assets can be classified in four distinct categories: i) Device, ii) Connectivity, iii) Applications and iv) Data (Theoharidou et al., 2012).

The Device asset encompasses all the hardware components of the device. These are the physical device and its resources (processor, memory, storage, sensors, display, battery, camera etc.).

The Connectivity asset refers to the technologies used in order to provide mobile network connectivity services. These are the i. GSM services (Global System for

Mobile communications), ii. WPAN services (Wireless Personal Area Networks), iii. WLAN (Wireless Local Area Networks) and WMAN (Wireless Metropolitan Area Networks) services, iv. Cellular network services, and v. NFC interface services (Near Field Communication).

The Applications assets refer to all apps that are installed on the smartphone. These apps can be preinstalled by the manufacturer or the carrier or can be third party apps that have been installed by the user.

The Data assets are all the information stored and used in a smartphone. This information can be contacts, financial data, calling history, location information, usage history, pictures etc. and can be categorized into personal, financial, business, health, authentication or connectivity data types.

The data taxonomy according to their source is (Mylonas, 2008):

- *Messaging data*: data derived from the carrier's messaging services (SMS, EMS and MMS) or instant and e-mail messages. This category includes messaging logs as the receiver, the sender, the time and date of delivery, attachments etc.

- *Device data*: all the data of the device and the operating system that are not related to third party apps (contacts, images, IMEI, Wi-Fi MAC address, device serial number etc.).

- *(U)SIM card data*: these data include specific information of the user to be uniquely identified by the telecommunication carrier, such as the IMSI (International Mobile Subscriber Identity), the MSIN (Mobile Subscriber Identification Number) and the ICCID (Integrated Circuit Card Id). The SIM card contains the mechanisms for the operating system work flow, user authentication, data encryption algorithm, and it's file system resides in persistent memory and stores data as names and phone number entries, text messages, and network service settings.

- *Application data*: all the necessary data accessible by apps and necessary for their execution. These can be configuration files, logs or temporal data.

- *Usage history data*: all the log data relating to the usage of the smartphone. These can be the call logs, the browsing history logs, the network connection history logs and the event logs of the operating system.

- *Sensor data*: all the data relating to the sensors of the smartphone. These can be location data, temperature data, direction data, vibration data etc. The most significant sensors that exist in almost every smartphone are the camera, the microphone, the GPS, the compass and the accelerometer sensors.

- *User Input data*: these data are produced from the interaction of the user with the smartphone. For example, in this category we have the keystrokes, the button presses and the user gestures. As gestures we can characterize the drags, swipes, taps, double taps, touch-n-holds and shakes, that is all the interactions a user can make in order to complete a specific task.

These data sources can handle many information types, such as personal, business, authentication, financial, health and connectivity data. According to the information type these data sources handle, some can be more critical than others. For example, the apps can handle all types of data, including sensitive data, such as health information.

Different parties can have access to different data on the smartphones. A list of entities that can or/and have access to user information on smartphones is presented below:

- *Mobile device*: can have access to the device data and the sensor data.

- *Operating System*: can have access to the messaging data, the device data, the application data, the usage history data, the sensors data, the user input data and some of the U(SIM) card data.

- *Applications*: the app's functionalities define which data can be accessed. According to the functionalities, a certain categorization applies. These types of applications can be:

  A. Games → can access sensor data and user input data.
  B. Content and media consumption apps (music, photo & video, sound recordings, books etc.) → can access device data, sensor data and user input data.
  C. Core functionality and utility (phone tools, mapping, navigation etc.) → can access sensor data.
  D. Social networking, communication & lifestyle (VoIP, micro blogging, instant messaging, social media, shopping, news, ad networks etc.) → can access messaging data, device data, some of the (U)SIM card data, usage history data, sensor data and user input data.
  E. Business and productivity apps (mobile banking, translation, office, calendar etc.) → can access usage history data and sensor data.

  Browsing apps combine C and E type functionalities. These hybrid functionalities can apply because even if most popular operating systems have a pre-installed web-browsing app, it is possible to install third party web-browsing apps. These functionalities allow access to browsing history data, GPS sensor data and application execution data. All apps have access to the application data related to their usage, such as logs and configuration files, but cannot access the application data of other apps.

- *Mobile telecommunication carrier*: service providers collect incoming and outgoing calls and text messages, location data and data concerning the Internet usage (the frequency the email is checked, the frequency and the duration of the internet access). They can have access to the messaging data, the (U)SIM card data, the usage history data and the sensor data.

Entities \ Data Sources	Messaging Data	Device Data	(U)SIM Card Data	Application Data	Usage History Data	Sensor Data	User Input Data
Mobile Device		✓				✓	
Operating System	✓	✓	~	✓	✓	✓	✓
Application Type: *A. Games*				*		✓	✓
B. Content & media consumption		✓		*		✓	✓
C. Core functionality & utility				*		✓	
D. Social networking & communication	✓	✓	~	*	✓	✓	✓
E. Business & productivity				*	✓	✓	
Mobile T/C carrier	✓		✓		✓	✓	

Table 1: Smartphone data taxonomy based on the entities that can gain access (✓ depicts access to the specified data, ~ depicts partial access, * depicts access only to the data related to their usage)

3. Contemporary privacy issues

Popular operating systems for smart devices offer a large number of permissions to handle the access of apps to the vast set of personal data items. For example the Android operating system version 4.4 supports over 140 different app level permissions to control the access of apps to the resources of the smart device. However, the permissions handling framework of modern operating systems for smart devices is far from adequate. There are noteworthy shortcomings both in the way the current features are implemented and most importantly, due to the fact that several fundamental services for handling personal data are not supported at all. We will focus on the Android operating system, but similar issues exist in other operating systems like iOS and Windows Mobile. We start with a brief description of representative known issues of the permissions model of Android and then discuss in detail two shortcomings, which, to our knowledge, have not been discussed before.

In Android, when an app is installed the user is prompted to approve the permissions that the app requests. Unfortunately, the user has no option to "negotiate". For example, if a "compass" app requests access to read the specific sensors but also the identity and the contacts of the user, the user cannot grant access only to the permissions that are related to the functionality of the app.

Certain permissions would be much more effective if they could support a more fine-grained access control. In (Jeon et al., 2012) the authors evaluate a fine-grained approach for app permissions. For example, an app that needs to connect to a specific Internet address should be granted this permission and not full access to the Internet.

In (Wei et al., 2012) the authors study the permission evolution and usage in the Android ecosystem since its inception in 2008. A key finding is that the permission model of Android is becoming more complex and hard for users and even developers to understand. A further observation is that permissions are not becoming more fine-grained and that the whole platform is not becoming more secure from the user's point of view.

A technical detail with important consequences on the effectiveness of the security model is that certain permissions are grouped in a way that makes the fair agreement between users and apps hard, if not impossible. Harmless data items are grouped together with critical PII (Personal Identifying Information) fields. For example, an application that needs to know if the phone is currently in a call, must be granted the "read phone state" permission. However, the same permission provides access to critical PII information, like the IMEI of the device, the subscriber ID, the serial number of the SIM, etc. The situation with this particular permission is even worse. For backwards compatibility reasons, any app that supports older versions of Android must request this permission because in early versions of Android this was granted by default to the apps. The app developers have no other option if they intend to support the early versions of Android. Thus, several applications are requesting the particular permission without actually needing any of the granted personal data items. From the users' point of view, it is not possible to distinguish if and how apps will make use of this permission.

The permissions have become so complex that in (Vidas et al., 2011) the authors address the complexity of the permissions framework and propose a utility to support developers in aligning their permission requests with the needs of their apps.

There are more known issues such as the above.

We would like to emphasize two additional shortcomings of the security models of mobile operating systems as follows:

1. Smart mobile devices carry an enormous amount of (sensitive) personal data of their owners. The fair access and treatment of personal data is defined in the Data Protection Directive 96/46/EC (European Parliament, 1995). In the current permissions framework of Android, the app simply

requests permissions without specifying the purpose of accessing the personal data and terms of using these data. A fundamental right of any citizen when he is asked to disclose a personal data item is to be informed about the exact usage of this data item, as well as the exact terms and conditions of this usage (where and how long will the data be stored, etc.)

2. A key issue for the protection of personal data on smart devices is whether the requesting app has the right to send the accessed data items outside of the smart device. This permission is related to the terms and conditions discussed above (bullet 1), but, in our view, deserves to be discussed separately. An app requesting a personal data item on a smart device differs from web or client desktop applications in that the app is running on a platform owned by the end user. More precisely, when an app requests a personal data item it should be clearly stated if this information is used only within computations on the device or if this personal information can be transferred outside the device. For example, if an app requests information about the age and the gender of the user simply to adjust the user interface to the corresponding age class, then there is no reason for the app to send this information anywhere outside the smart device, and the privacy of the user is not seriously threatened. If, however, the app plans to send these personal data items to some database servers of the app provider or somewhere else, then there is disclosure of personal data and this should be clearly stated in the request.

4. Un-training the user and the developer

The progressive corrosion of privacy caused by the increasing permission requests on each and every app update, may have adverse effects on the user's attitude toward security. (West, 2008) enumerated a series of psychological attributes involved in security depriving actions and the lack of user motivation is one of the main traits a user may exhibit. As the user may be predisposed towards not performing security enhancing actions - such as software updates in this case - it is reasonable to expect that the privacy degenerative app updates will further fuel such lack of motivation or even provide the user with an alibi not to perform updates.

Not performing software updates is particularly problematic for smartphones that are part of a corporate network. According to the Bring Your Own Device (BYOD) trend where an employee prefers to use his personal devices (laptops or smartphones) in order to carry out work related tasks, the security risks of an organization may rise significantly. With BYOD the traditional network (firewalled) perimeter does not exist, as user devices can "freely" enter and leave the corporate network, bypassing the perimeter controls. This situation has triggered a significant amount of research on risk analysis, security policy requirements and security controls for managing BYOD insecurity. In addition, when the device is a smartphone and also an employee's property, the risk of introducing malware in the corporate environment is high, as the administrator will not have adequate control over the device.

Therefore the software app provider's appetite to unreservedly collect and consume user personal data may be done at the expense of the security of the organization the user works as an employee. Furthermore, the increased complexity of the permissions may also cause similar effects to the app developer. A developer may not fully understand the different permissions, or may not appreciate the need for requesting the minimum set of permissions - he may in fact prefer to "play it safe" by requesting more permissions in order to avoid time consuming troubleshooting and debugging in case of software failures due to restrictive policies.

5. Discussion and outlook

It is highly worrisome that today's smartphone operating systems do not provide an adequate level of security for the user's personal data and it is possible that they are consciously developed with vulnerabilities. As depicted in Table 1, sensor data are the most critical, because they can be read by any entity that has access on the smartphone device. Moreover, the entity that seems to be the most threatening is the operating system, which has access to all personal data and yet has security vulnerabilities.

As discussed earlier, several interesting ideas such as fine grained permissions and a better grouping of the permissions have been proposed in the recent literature. If applied, such ideas can significantly improve the situation and mitigate privacy concerns of users of smart devices. In this work we make the following additional suggestions:

- First, the permissions framework should be extended to comply with the Data Protection Directive (European Parliament, 1995) by covering also the purpose and the usage terms for each personal data item. For example, when an app requests the "Read Phone State" permission it should at least clearly state how it will use each of the affected data items, for how long, that it will be securely stored during this period and that it will be securely deleted afterward.

- Second, for each data item requested by an app (and, in general, any applications running on the user-side), it should be clearly stated if this data item will be transferred outside the smart device (the data item as it is or results obtained from this data item) or if it will only be used inside the smart device. If the item will be sent outside the smart device, then its transfer, storage and usage should comply with the requirements specified in the Data Protection Directive.

Both suggestions are technically feasible and would strongly enhance the control of users over their personal data without damaging the legitimate app providers. Moreover, such measures would improve the users' ability to distinguish legitimate apps from malware or grayware (apps that carry out questionable actions without sufficient user notification or approval (Sarma et al., 2012)). Even though there are some tools, such as TaintDroid mentioned earlier, that a skilled user can use to try to

figure out the risks of using an app, there is no established way for apps and other entities of the mobile computing ecosystem to commit to responsible and transparent practices on mobile users' privacy. Our proposal is an inherent way of supporting this functionality into the operating system, along with the obligation of the apps to state why and how the users' personal data will be manipulated.

Another suggestion for limiting the rich list of permissions set by the app vendor is to leverage market attitudes and consumer behaviour that will demonstrate susceptibility on unjustified permission requests. More specifically, a reputation system based on the evolution of permissions during version updates that is made public could allow a user to make an informed decision on the privacy respecting attitude of an app vendor. As with most community networks, their value follows Metcalfe's Law and in order for this recommendation to have impact, it requires commitment and subscription from a significant number of end users.

Our future intention is to conduct a user study to measure the user's awareness and concepts around privacy concerns in order to validate the above mentioned suggestions.

6. References

Balebako, R., Marsh, A., Lin, J., Hong, J. and Cranor, L. F. (2014), "The Privacy and Security Behaviors of Smartphone App Developers", *Workshop on Usable Security* (USEC 2014), San Diego, CA.

Enck, W., Gilbert, P., Chun, B., Cox, L.P., Jung, J., McDaniel, P. and, Sheth, A.N. (2010), "TaintDroid - An Information Flow Tracking System for Real-Time Privacy Monitoring on Smartphones", *Proceedings of the 9th USENIX conference on Operating systems design and implementation*, p.1-6, Vancouver, BC, Canada.

European Parliament (1995), "Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data", *Official Journal of the EC*, *23*, 6.

Felt, A. P., Ha, E., Egelman, S., Haney, A., Chin, E. and Wagner, D. (2012), "Android permissions: User attention, comprehension, and behavior", In *Proceedings of the Eighth Symposium on Usable Privacy and Security,* ACM, p. 3.

Jeon, J., Micinski, K.K., Vaughan, J.A., Fogel, A., Reddy, N., Foster, J.S. and Millstein, T. (2012), "Dr. Android and Mr. Hide: fine-grained permissions in android applications", *In Proceedings of the second ACM workshop on Security and privacy in smartphones and mobile devices (SPSM '12),* ACM, New York, NY, USA, 3-14.

Mylonas, A. (2008), "Smartphone spying tools", *MSc Thesis, Royal Holloway*, University of London.

Sarma, B. P., Li, N., Gates, C., Potharaju, R., Nita-Rotaru, C., and& Molloy, I. (2012), "Android permissions: a perspective combining risks and benefits", In *Proceedings of the 17th ACM symposium on Access Control Models and Technologies*, ACM, New York, USA, pp. 13-22.

Theoharidou, M., Mylonas, A. and Gritzalis, D. (2012), "A risk assessment method for smartphones", *In Proc. of the 27th IFIP Information Security and Privacy Conference*, Springer (AICT 376), p.443-456.

Vidas, T., Christin, N. and Cranor, L. (2011), "Curbing android permission creep", *In Proceedings of the Web*, Vol. 2.

Wei, X., Gomez, L., Neamtiu, I., and& Faloutsos, M. (2012), "Permission evolution in the android ecosystem", In *Proceedings of the 28th Annual Computer Security Applications Conference,* ACM, New York, NY, USA, pp. 31-40.

West, R. (2008), "The Psychology of Security", *Communications of the ACM*, Vol. 51, No. 4, pp. 34-41.