# HappyPermi: Presenting Critical Data Flows in Mobile Application to Raise User Security Awareness

**Mehrdad Bahrini**
Digital Media Lab University of Bremen
Bremen, Germany
mbahrini@uni-bremen.de

**Marcel Meissner**
Digital Media Lab University of Bremen
Bremen, Germany
marcel.meissner@uni-bremen.de

**Rainer Malaka**
Digital Media Lab University of Bremen
Bremen, Germany
malaka@tzi.de

**Nina Wenig**
Digital Media Lab University of Bremen
Bremen, Germany
nwenig@uni-bremen.de

**Karsten Sohr**
Digital Media Lab University of Bremen
Bremen, Germany
sohr@tzi.de

## ABSTRACT

Malicious Android applications can obtain user's private data and silently send it to a server. Android permissions are currently not sufficient enough to ensure the security of users' sensitive information. For a sufficient permission model it is important to account the target of the outgoing data flow.

**KEYWORDS**
Usable Security; Android; Permission; User Awareness; Data Flow

*CHI'19 Extended Abstracts, May 4–9, 2019, Glasgow, Scotland UK*
© 2019 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-5971-9/19/05.
https://doi.org/10.1145/3290607.3312914

On the other hand, permission dialogues often contain relevant information, but most of the users generally do not understand the implications or the visualization fails to guide the user attention to it. It is important to empower users by providing applications that show them who can access their private data and who might send this data to the outside. In order to raise user awareness considering Android permissions, we developed HappyPermi, an application that visualizes which user information is accessible by the granted permissions. Our evaluation ($n = 20$) shows that most users are not aware of the sensitive data that their installed applications have access to. Our results suggest how different users feel about accessing their sensitive data when they are aware of its outgoing destinations.

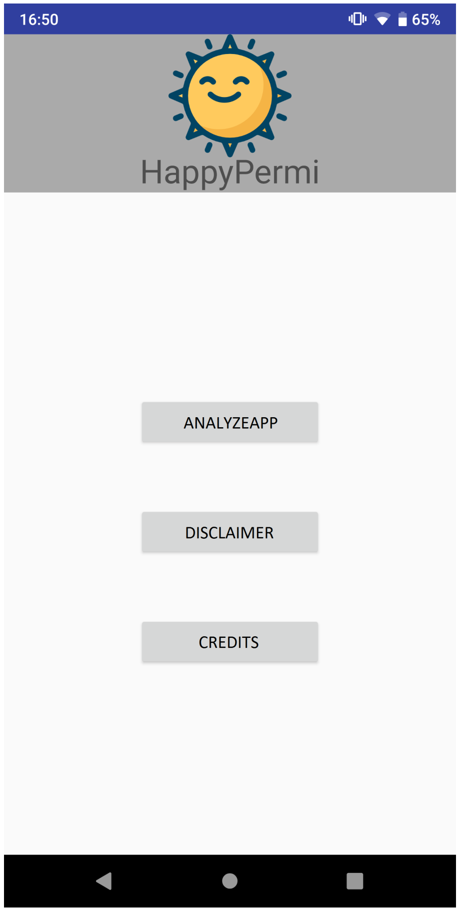
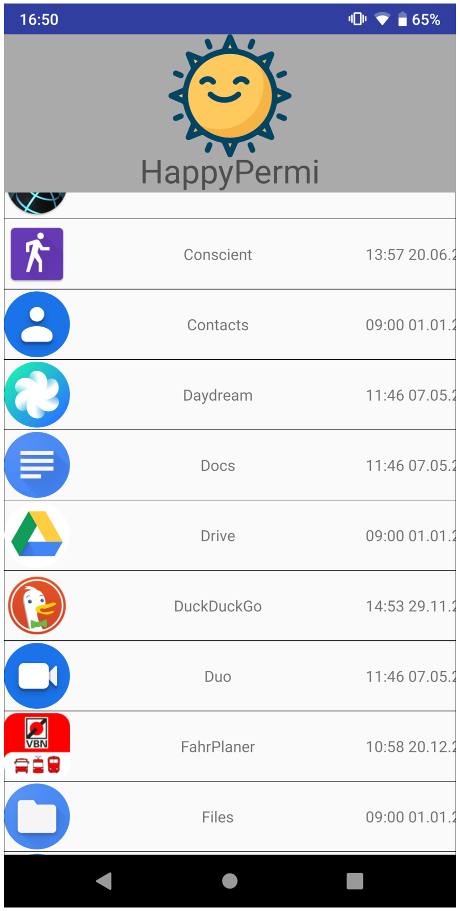


**Figure 1: HappyPermi app: It analyzes the permissions of the installed applications on a mobile device. (The HappyPermi icon made by Dimitry Miroliubov and is from www.flaticon.com)**

## INTRODUCTION

The use of mobile phones has increased dramatically in recent years[1]. Mobile devices are a tool for calling and writing messages, but they also offer a platform for entertainment and personalized services with high performance data sharing and users participating in a mobile social network. In order to protect users' critical information, Android offers permission systems in which users can grant or deny a third-party application (app) access to their sensitive resources. In earlier versions of Android (5.1 and below) users are notified about requested permissions at the install-time of an application. However, users were not allowed to choose, they had to grant permissions in order to install and use the application. This model is criticized as ignoring the Principle of Least Privilege (PoLP) [4]. In fact with this model, many applications tended to request much more permissions than necessary and this could cause more data leaks. Therefore, in modern versions, Android implements an ask-on-first-use policy which helps users to make better decisions due to the context of a permission request. Despite that multiple problems can arise from this model e.g., a user grants the permissions for a specific task, but the application can abuse the permissions without the knowledge of the user [10]. For example, an application component may send the stored contacts to an unknown server as long as it has the permission to access contact information. Such unwanted behavior is hard to detect for common users. Therefore, in this ongoing work, we explore how an interactive application can help users to understand permissions and their effects more comprehensively. The HappyPermi application (see Figure 1) analyzes the permissions of the installed applications on a mobile device in two steps. At first, HappyPermi reads the intended application manifest (describes essential information about the application) and presents the critical permissions graphically. Secondly, it uses Uniform Resource Locator (URL) representations to show how critical data propagates throughout the application execution from the smartphone to multiple receivers (URLs). Since the second step depends on static analysis of an application, Mobile Security Framework (MobSF) was used for detecting destination URLs. With MobSF, it is not possible to find flows but instead delivers and presents URLs that are stored or called in the application. In order to use this URL data in our evaluation phase, we choose

[1] https://www.statista.com/statistics/263437/global-smartphone-sales-to-end-users-since-2007



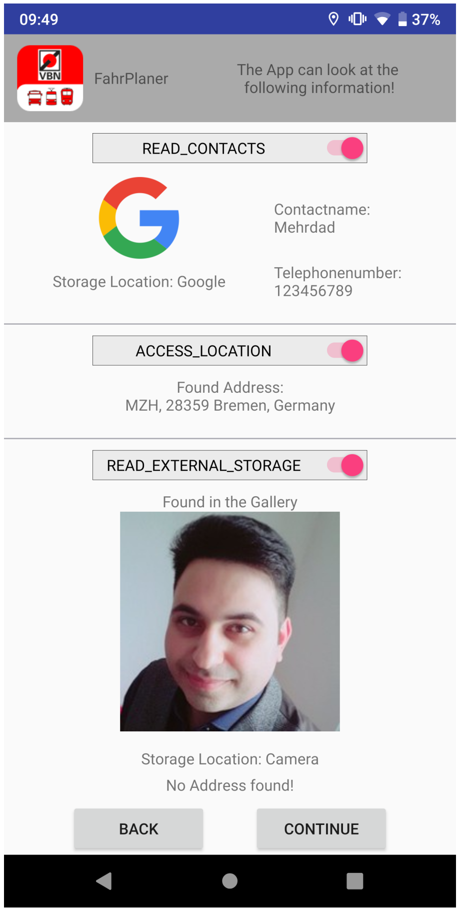
| | |
|---|---|
| Conscient | 13:57 20.06.2 |
| Contacts | 09:00 01.01.2 |
| Daydream | 11:46 07.05.2 |
| Docs | 11:46 07.05.2 |
| Drive | 09:00 01.01.2 |
| DuckDuckGo | 14:53 29.11.2 |
| Duo | 11:46 07.05.2 |
| FahrPlaner | 10:58 20.12.2 |
| Files | 09:00 01.01.2 |

**Figure 2: HappyPermi app: It presents a list of all installed applications with the icon, name and installation date. This screen is a main feature in both versions.**

the FahrPlaner application and analyzed it with MobSF. FahrPlaner is a popular local application, for getting information about the public transportation system for large parts in the north of Germany and has access to contacts, location, camera and storage permissions. We believe that HappyPermi will increase user awareness in terms of the Android permissions and they will pay more attention to the requested permission. The users can make an informed decision about whether to install or use the selected application or not. The main contribution of this work is to improve the balancing of usability and security in available Android permission models. It also provides a user-center approach focusing on the motivation of the users to learn and perform security and privacy related tasks.

## RELATED WORK

Previous research has shown that many users are either unaware that permission settings even exist or they pay just little attention and never change them. Felt et al. [4] showed that only 17% of the participants in an internet survey and laboratory study ever take a look at the permissions during installation of the applications. Recent evaluations show that users correctly understood neither the permissions that were requested, nor the reasons why they were needed [9], [10]. Also users were mostly concerned about risks that would cost money or delete important information such as the contact lists [3]. They are often unaware of the data collected by their applications and are surprised and feel uncomfortable when they learn about them [5]. However, Android's permission system is intended to inform users about the risks of installing applications and each permission dialogue offers information to make rational decisions. To deal with difficulties in checking permission settings, Baarslag et al. [1] have investigated how helpful it is for the user to negotiate with the permissions. This means that the user grants the application a certain permission and receives a monetary reward, usually in the form of a price discount. Therefore the user information will be valuable to them. In order to help users make low risk application choices, Rajivan and Camp [6] have simulated the Google Play Store and used emoticons, eyes and padlocks as icons to show the users the safety ranking of the applications. They also used privacy priming to assess this influence. Their results showed that priming had little influence on the user selection, compared to the symbols. Although, there are no consistent standards, because symbols are not used for individual permissions. Van Kleek et al. [8] focused on the presentation of data flows, indicating which data goes to which operator. In addition, they showed information about the company and the suspected reason why the data is collected. They introduced a prototype that can automate the mobile traffic purpose inference and help the users to make informed decisions. The application "PermissionWatcher" lists all non-system applications, showing their permissions and indicating whether an application is dangerous or not. An application is dangerous if there is a dangerous permission combination. In addition, each permission is described with a short text. The goal of Struse et al. [7] was to find out if a smartphone user can understand the access rights and their implications with a permissions-based application security



The App can look at the following information!

Contactname: Mehrdad

Storage Location: Google

Telephonenumber: 123456789

Found Address: MZH, 28359 Bremen, Germany

Found in the Gallery

Storage Location: Camera

No Address found!

**Figure 3: HappyPermi app: The granted permissions in visual form for the FahrPlaner application. HappyPermi app shows which private data of the user might be used by the application. This screen is a main feature in both versions.**

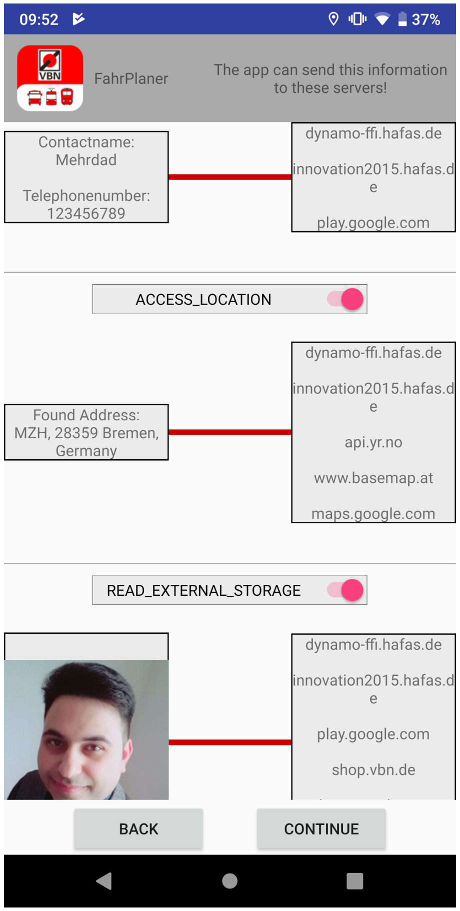
model. In our work, we concentrate on the Android application permission model and develop an application visualizing data flows on users' smartphones. Therefore, we use their private information and guide the users to pay more attention in granting permissions. The features of HappyPermi are a simplification of the permission concept.

## THE HAPPYPERMI APPLICATION

We designed and developed our application based on two research questions: "How does an interactive application help users to understand the access of granting permissions to their private information?" and "To what extent are the users aware where their data is being sent?" To cover these questions, we designed HappyPermi in two versions (HappyPermi and HappyPermi+Flow). In both versions, the user can start the analyzing process by clicking on the "AnalyzeApp" button to inspect all installed applications on the device. Afterwards, HappyPermi presents a list of all installed applications. For each application, the user can see the icon, name and installation date (see Figure 2). By clicking on an application, the user will be redirected to the next view. There, the user can see the granted permissions in visual form (see Figure 3). For example, on clicking "Read Contacts", the user sees the location of a contact, the contact name, and the phone number. For "Read External Storage", HappyPermi presents a private image of the user and its associated storage location. In our evaluation, we choose the FahrPlaner application. As a means of transport application, FahrPlaner has location permission in order to find the best route to the users destination. It also has contacts permission to include an address from our contact list as a starting point or as a destination. Camera and storage are other permissions of this application. Users can take picture and personalize icons with own photos. In order to perform our evaluation, in the first version (HappyPermi), the user is directed to the FahrPlaner permissions settings in the Android operating system. This allows the user to see which permission needs to be customized. The aim of this version is to find out if the user has obtained knowledge about what information is accessible through granting permissions. In the HappyPermi+Flow version, we also present URLs from analyzing MobSF on a PC. Figure 4 shows that only a selection of the found URLs was taken and added to the flow view. We inform the user that the granted permission can cause sending critical information to unknown servers. Like the previous version, by clicking "Continue", the user is directed to the permissions settings in the Android operating system.

## EVALUATION

Through a laboratory study, we evaluated the two versions of HappyPermi in a between-subjects-design with 20 participants (10 in each group). Ten participants had a computer science degree, while one had completed high school and 9 had an advance degree. Each participant used only about 10 minutes one version of the application and did not know about the other version. We asked them to install our applications on their mobile phone and also install FahrPlaner from the Google Play Store,



09:52 37%

FahrPlaner

The app can send this information to these servers!

Contactname: Mehrdad

Telephonenumber: 123456789

dynamo-ffi.hafas.de

innovation2015.hafas.de

play.google.com

ACCESS_LOCATION

Found Address: MZH, 28359 Bremen, Germany

dynamo-ffi.hafas.de

innovation2015.hafas.de

api.yr.no

www.basemap.at

maps.google.com

READ_EXTERNAL_STORAGE

dynamo-ffi.hafas.de

innovation2015.hafas.de

play.google.com

shop.vbn.de

**Figure 4: HappyPermi+Flow: For each permission, the left box indicates private user data that is accessible through granted permissions and the right box shows its destination in the form of URLs. This feature is available in the HappyPermi+Flow version.**

if it was not already installed. We gathered users' feedback through a comprehensive questionnaire consisting of two parts. Part one was the System Usability Scale (SUS), for measuring the usability [2]. In order to measure emotions (Pleasure, Arousal and Dominance), we used the Self-Assessment Manikin (SAM) test. During the evaluation, we used the Thinking Aloud method, to determine the motivation for decisions. In group one (HappyPermi), we had 10 male participants, with an average age of 24.1 ($SD = 2.27$). The participants in group two (HappyPermi+Flow) consist of 1 female and 9 males, with an average age of 22.6 years ($SD = 2.08$). The task for the participant was to analyze the FahrPlaner and, if necessary, the permissions should be adjusted. The participants were not instructed how they should use HappyPermi. They were allowed to use the application freely.

## RESULTS

The average SUS score of group two (HappyPermi+Flow) is 80.75 ($SD = 20.58$), showing a high usability of the application. Group one (HappyPermi) reaches an average value of 72.5 ($SD = 14.62$), a good usability above average. Although, the independent t-Tests for the SUS did not reveal any significant differences (see also Figure 5). The results of the SAM questionnaire are also similar for both groups (see Table 1) and the conducted t-Tests do not show significant differences. Since in both versions users were directed to the Android permissions settings, they could change the FahrPlaner permissions. A large amount of the participants (69%) turned off the contact permission. The participants pointed out that they did not know that the FahrPlaner application had access to their contacts and might send them to a server. Also, it was not clear to them why this application has access to the camera. P07 mentioned that "I think my contact details are not necessary for FahrPlaner and therefore should not be required. The uncertainty of what is finally done with the data and who gets everything is also frustrating. This should be shown more transparent." From this statement, we conclude that by empowering awareness of the users, they pay more attention to what they have done before without knowledge. Many participants believed that they have gained a better understanding of permissions through HappyPermi and they also want to continue to use it as an additional source of information or as a permission tool. P05 stated that "It is interesting, because you have a simple overview, which application uses which rights." Also P12 said that "It was a completely different effect, if one sees concrete telephone numbers or personal photos which can be passed to all sorts of applications without hesitation." Although, this application does not cover all user concerns about permissions, participants pointed out that they have been able to get acquainted with it in a short time and understood its purpose.

## CONCLUSION

In this paper we presented HappyPermi, an application which helps users to obtain a better understanding of granted permissions on Android devices. Our results from our first evaluation imply that,

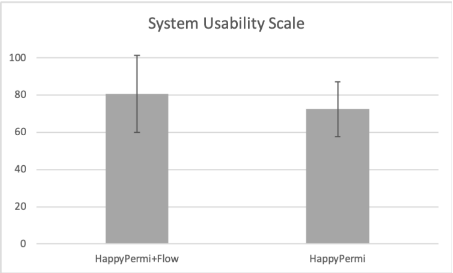


**Figure 5: The SUS for both groups: The HappyPermi and the HappyPermi+Flow application**

**Table 1: The mean and standard deviation for the SAM questionnaire for both groups**

| *Emotions* | *HP* | *HP+Flow* |
|---|---|---|
| Pleasure $M$ | 5.3 | 3.4 |
| $SD$ | 1.2 | 2.0 |
| Arousal $M$ | 3.7 | 4.7 |
| $SD$ | 1.9 | 2.7 |
| Dominance $M$ | 5.9 | 6.3 |
| $SD$ | 2.2 | 2.1 |

by empowering user awareness considering Android permissions, users pay more attention to the permissions. For the future we plan to extend HappyPermi analysis methods in order to present users more accurate data. For example, user should know why an application needs critical permissions. Furthermore, we want to explore novel visualizations of granted permissions while applications that are running.

## ACKNOWLEDGEMENTS

This work was supported by Klaus Tschira Foundation (www.klaus-tschira-stiftung.de).

## REFERENCES

[1] Tim Baarslag, Alper T. Alan, Richard C. Gomer, Ilaria Liccardi, Helia Marreiros, Enrico H. Gerding, and m.c. schraefel. 2016. Negotiation As an Interaction Mechanism for Deciding App Permissions. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '16)*. ACM, New York, NY, USA, 2012–2019. https://doi.org/10.1145/2851581.2892340

[2] John Brooke. 2013. SUS: a retrospective. *Journal of usability studies* 8, 2 (2013), 29–40.

[3] Erika Chin, Adrienne Porter Felt, Vyas Sekar, and David Wagner. 2012. Measuring User Confidence in Smartphone Security and Privacy. In *Proceedings of the Eighth Symposium on Usable Privacy and Security (SOUPS '12)*. ACM, New York, NY, USA, Article 1, 16 pages. https://doi.org/10.1145/2335356.2335358

[4] Adrienne Porter Felt, Elizabeth Ha, Serge Egelman, Ariel Haney, Erika Chin, and David Wagner. 2012. Android Permissions: User Attention, Comprehension, and Behavior. In *Proceedings of the Eighth Symposium on Usable Privacy and Security (SOUPS '12)*. ACM, New York, NY, USA, Article 3, 14 pages. https://doi.org/10.1145/2335356.2335360

[5] Jialiu Lin, Shahriyar Amini, Jason I. Hong, Norman Sadeh, Janne Lindqvist, and Joy Zhang. 2012. Expectation and Purpose: Understanding Users' Mental Models of Mobile App Privacy Through Crowdsourcing. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing (UbiComp '12)*. ACM, New York, NY, USA, 501–510. https://doi.org/10.1145/2370216.2370290

[6] Prashanth Rajivan and Jean Camp. 2016. Influence of Privacy Attitude and Privacy Cue Framing on Android App Choices. In *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*. USENIX Association, Denver, CO. https://www.usenix.org/conference/soups2016/workshop-program/wpi/presentation/rajivan

[7] Eric Struse, Julian Seifert, Sebastian Üllenbeck, Enrico Rukzio, and Christopher Wolf. 2012. PermissionWatcher: Creating User Awareness of Application Permissions in Mobile Systems. In *Ambient Intelligence*. Springer Berlin Heidelberg, 65–80.

[8] Max Van Kleek, Ilaria Liccardi, Reuben Binns, Jun Zhao, Daniel J. Weitzner, and Nigel Shadbolt. 2017. Better the Devil You Know: Exposing the Data Sharing Practices of Smartphone Apps. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, New York, NY, USA, 5208–5220. https://doi.org/10.1145/3025453.3025556

[9] Primal Wijesekera, Arjun Baokar, Lynn Tsai, Joel Reardon, Serge Egelman, David A. Wagner, and Konstantin Beznosov. 2017. The Feasibility of Dynamically Granted Permissions: Aligning Mobile Privacy with User Preferences. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*. 1077–1093. https://doi.org/10.1109/SP.2017.51

[10] Primal Wijesekera, Joel Reardon, Irwin Reyes, Lynn Tsai, Jung-Wei Chen, Nathan Good, David Wagner, Konstantin Beznosov, and Serge Egelman. 2018. Contextualizing Privacy Decisions for Better Prediction (and Protection). In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM, New York, NY, USA, Article 268, 13 pages. https://doi.org/10.1145/3173574.3173842