PermWatch: A Tool for In-Situ Research on Users' Awareness and Control of Android Permissions

Verena Winterhalter
LMU Munich, Germany
verena.winterhalter@ifi.lmu.de

Anouk Moreno
LMU Munich, Germany
anouk.moreno@campus.lmu.de

Sarah Prange
LMU Munich, Germany
sarah.prange@ifi.lmu.de

Harel Berger
Ariel University, Israel
harelb@ariel.ac.il

Florian Alt
LMU Munich, Germany
University of the Bundeswehr, Germany
florian.alt@ifi.lmu.de

Abstract

We present *PermWatch*, a field-ready research tool for studying users' awareness, perception, and control of Android permissions. Designed for in-the-wild deployment, *PermWatch* supports fine-grained logging of app permission states and user-driven permission changes. The tool enables in-situ experience sampling to enhance automated data logging with user feedback. *PermWatch* was successfully deployed in a previous SOUPS study (N=132) [16], where it revealed low user awareness of current permission states and identified opportune moments for permission control. This poster presents the implementation and data collection methods of *PermWatch*, early insights, and opportunities for future research. We invite feedback from the community on expanding the tool's capabilities and welcome collaboration for future deployments.

1 Introduction and Background

It is normal and expected for smartphone applications to request permissions to access the device functionalities they need in order to function. Android's permission system has evolved to give users fine-grained control, allowing permissions to be granted at runtime and revoked at any time [2, 4]. Yet, users often remain unaware of what permissions are active, how they got there, or how to change them [8, 12, 15, 18]. In addition, permission explanations are often misleading [11], limiting users' ability to make informed decisions. Beyond usability concerns, permission data is also valuable from a security perspective: combinations of permissions can reveal patterns indicative of malicious behavior [10, 13, 17, 19].

Prior research has identified challenges with user awareness and control over permissions [14], though relying on self-reports [15] or one-time data collections [9]. However, real-world insights on how users actually handle permissions in daily life are crucial to understand practical challenges and design effective interventions. To address this gap, we developed *PermWatch*, a research tool that enables continuous logging of permission states and user-driven changes. It complements automated data collection with in-situ experience sampling to gather user feedback at key decision points. This poster introduces *PermWatch*, outlines how it enables new research protocols, and highlights potential directions for future work. Our goal is to support the community in building a deeper understanding of permission behavior in-the-wild, and to inform the design of more effective privacy interfaces.

2 Research Approach

To understand users' awareness of and dealing with privacy permissions in context, it is essential to gather in-the-wild data, ideally over a longer time period. To achieve this, we implemented *PermWatch*, an Android application that we successfully deployed in a previous SOUPS study [16].

2.1 Study Application *PermWatch*

PermWatch is implemented as a native Android application aimed at version 8.0 and above. The app supports two major ways of collecting data (cf. Section 2.2): 1) permission states of all installed apps are monitored regularly by the *Permission Scanner*; 2) an in-app interface for *questionnaires*. The permission scanner runs in the background. In-app questionnaires are built using the *SurveyKit* [5]. *PermWatch* also provides a simple home screen with contact information and response statistics, for participants to keep track of their possible compensation (by completing sufficient questionnaires,

USENIX Symposium on Usable Privacy and Security (SOUPS) 2025.
August 10–12, 2025, Seattle, WA, United States.

cf. Section 2.4). After installation, *PermWatch* guides participants through the setup:

1. *Consent:* Participants need to accept the data collection procedure (cf. Section 2.4) to start the study.
2. *Permission Access:* The app requires access to Android's Package Manager [6] and Usage Stats Manager [7]. The app only then begins to collect data.
3. *Initial Questionnaire:* Participants fill an initial questionnaire covering basic demographics.
4. *Notification Schedule:* Participants can set a custom time frame in which they are happy to receive notifications/questionnaires (e.g., 7 am to 9 pm), following recommendations from related work [20].

After a set time (e.g., 14 days), *PermWatch* presents a final questionnaire and stops collecting data.

2.2 Data Collection

Automated Data Logging *PermWatch* collects the following data automatically:

- **Assessing Current Permission States:** Upon setup, *PermWatch* collects data on currently installed apps and permission states, as well as data on app usage times.
- **Monitoring Permission Changes Over Time** During the study phase, *PermWatch* regularly scans all permission states of all applications (every two hours). In case a permission state changed compared to the last scan, *PermWatch* detects a *permission update* and triggers an experience sampling questionnaire (see below).

Collecting User Feedback During the study, participants receive different types of questionnaires (via notifications):

- **Experience Sampling:** Upon a permission update, *PermWatch* puts questions along the lines of *Why did you (allow, revoke) (app) access to (permission)?*
- **Daily questionnaires:** *PermWatch* asks about a subset of installed apps and respective permission states *daily*. Questions are along the lines of *Do you think (app) has access to (permission)?*
- **Final Questionnaire:** At the end of the study phase, participants fill a post-survey questionnaire. Participants are then advised to uninstall the app.

All questionnaires are withdrawn (i.e., notifications expire) after a certain time to ensure in-situ answers [20].

2.3 App Distribution & Recruitment

The app can be made available to participants online, e.g., through Firebase App Distribution [1] or an official app store. Recruitment can be achieved through online pools (e.g., Prolific [3]) or other online channels. Participants will then be redirected to a download link and guided through the setup of the app. For compensation, participant IDs and proof of study completion must be collected.

2.4 Ethics and Privacy Considerations

The setup of *PermWatch* includes a thorough *consent procedure*: participants receive detailed information on 1) the study goal and procedure (which can be easily tailored to the research question at hand), 2) a privacy policy in line with GDPR, and 3) detailed information on the data collection. Only upon acceptance, participants can continue with the study and *PermWatch* will start collecting data. Data *collection stops automatically* at a pre-defined point in time (e.g., after participants filled the final questionnaire). Receiving compensation can be bound to prerequisites (e.g., participants need to fill in at least 80% of the questionnaires). The amount of compensation should consider the total estimated effort (e.g., 5 minutes every day over 12 days would add up to an hour). This procedure was approved by the ethics committee of our faculty at LMU Munich for an upcoming study.

3 Early Insights and Future Research

Permission Types vs Permission Handling In our previous study [16], we found differences with regards to permissions that are essential to an app's functionality vs those that enable additional features. For instance, social media apps can be used to *consume* content without granting access to the camera permission (necessary to *generate* content). In future work, we also want to investigate other types of permissions, such as *install time* vs *runtime* permissions, to see whether there are any effects on users' awareness and control.

App Usage vs Permission Handling Another interesting direction is to look deeper into app usage – e.g., apps that are frequently used – in relation to permission states and updates. Alsoubai et al. found different user profiles among the number of installed apps and the number of permissions granted [9]. Longterm insights as acquired by *PermWatch* might be able to reveal more detailed user profiles.

Malicious Permission Pairs As a next step, we plan to take a closer look at the combination of certain permissions, as these can help determine malicous apps [10].

4 Conclusion

PermWatch enables scalable, in-the-wild studies of how users perceive and manage app permissions over time by combining automated logging with contextual user feedback. By capturing real-world behavior, it can uncover key usability and security challenges in mobile permission handling. We hope *PermWatch* will support future work in designing interfaces that inform users about granted permissions, increase awareness, and help them understand potential risks.

Acknowledgments

We would like to thank Gabriel Knoll for his continuous support with implementing the study app.

References

[1] Firebase App Distribution. https://firebase.google.com/docs/app-distribution, 2022. last accessed: 2025-05-20.

[2] Permissions on Android. https://developer.android.com/guide/topics/permissions/overview, 2022. last accessed: 2025-05-20.

[3] Prolific. A higher standard of online research. https://prolific.co/, 2022. last accessed: 2025-05-20.

[4] Request app permissions. https://developer.android.com/training/permissions/requesting, 2022. last accessed: 2025-05-20.

[5] SurveyKit: Create beautiful surveys on Android. https://github.com/quickbirdstudios/SurveyKit, 2022. last accessed: 2025-05-20.

[6] Android Developers - Documentation. PackageManager. https://developer.android.com/reference/android/content/pm/PackageManager, 2023. last accessed: 2025-05-20.

[7] Android Developers - Documentation. UsageStatsManager. https://developer.android.com/reference/android/app/usage/UsageStatsManager, 2023. last accessed: 2025-05-20.

[8] Nourah Alshomrani, Steven Furnell, and Ying He. Assessing user understanding, perception and behaviour with privacy and permission settings. In *HCI for Cybersecurity, Privacy and Trust: 5th International Conference, HCI-CPT 2023, Held as Part of the 25th HCI International Conference, HCII 2023, Copenhagen, Denmark, July 23–28, 2023, Proceedings*, page 557–575, Berlin, Heidelberg, 2023. Springer-Verlag.

[9] Ashwaq Alsoubai, Reza Ghaiumy Anaraky, Yao Li, Xinru Page, Bart Knijnenburg, and Pamela J. Wisniewski. Permission vs. App Limiters: Profiling Smartphone Users to Understand Differing Strategies for Mobile Privacy Management. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, New York, NY, USA, 2022. Association for Computing Machinery.

[10] Anshul Arora, Sateesh K. Peddoju, and Mauro Conti. *PermPair* : Android Malware Detection Using Permission Pairs. *IEEE Transactions on Information Forensics and Security*, 15:1968–1982, 2019.

[11] Michalis Diamantaris, Elias P. Papadopoulos, Evangelos P. Markatos, Sotiris Ioannidis, and Jason Polakis. REAPER: Real-Time App Analysis for Augmenting the Android Permission System. In *Proceedings of the Ninth ACM Conference on Data and Application Security and Privacy*, CODASPY '19, page 37–48, New York, NY, USA, 2019. Association for Computing Machinery.

[12] Bharathi Donku, Shahriar Rahman Khan, Tariqul Islam, and Raiful Hasan. Discrepancies in mobile app permissions: Exploring transparency and user awareness in the android ecosystem. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, CHI EA '25, New York, NY, USA, 2025. Association for Computing Machinery.

[13] Adeel Ehsan, Cagatay Catal, and Alok Mishra. Detecting Malware by Analyzing App Permissions on Android Platform: A Systematic Literature Review. *Sensors*, 22(20):7928, October 2022.

[14] Adrienne Porter Felt, Elizabeth Ha, Serge Egelman, Ariel Haney, Erika Chin, and David Wagner. Android permissions: user attention, comprehension, and behavior. In *Proceedings of the Eighth Symposium on Usable Privacy and Security*, pages 1–14, Washington, D.C., July 2012. ACM.

[15] Alisa Frik, Juliann Kim, Joshua Rafael Sanchez, and Joanne Ma. Users' Expectations About and Use of Smartphone Privacy and Security Settings. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22, New York, NY, USA, 2022. Association for Computing Machinery.

[16] Sarah Prange, Pascal Knierim, Gabriel Knoll, Felix Dietz, Alexander De Luca, and Florian Alt. {"I} do (not) need that {Feature!"}–understanding {Users'} awareness and control of privacy permissions on android smartphones. In *Twentieth Symposium on Usable Privacy and Security (SOUPS 2024)*, pages 453–472, 2024.

[17] Yash Sharma and Anshul Arora. A comprehensive review on permissions-based Android malware detection. *International Journal of Information Security*, 23(3):1877–1912, June 2024.

[18] Bingyu Shen. Can Systems Explain Permissions Better? Understanding Users' Misperceptions under Smartphone Runtime Permission Model. 2021.

[19] Gulshan Shrivastava, Prabhat Kumar, Deepak Gupta, and Joel J. P. C. Rodrigues. Privacy issues of android application permissions: A literature review. *Transactions on Emerging Telecommunications Technologies*, 31(12):e3773, December 2020.

[20] Niels van Berkel, Denzil Ferreira, and Vassilis Kostakos. The Experience Sampling Method on Mobile Devices. *ACM Comput. Surv.*, 50(6), dec 2017.