A Case Study of Usable Security: Usability Testing of Android Privacy Enhancing Keyboard

Zhen Ling[1(✉)], Melanie Borgeest[2], Chuta Sano[3], Sirong Lin[3], Mogahid Fadl[4], Wei Yu[5], Xinwen Fu[3], and Wei Zhao[6]

[1] Southeast University, Nanjing, China
zhenling@seu.edu.cn
[2] University at Albany - SUNY, Albany, NY 12222, USA
mborgeest@albany.edu
[3] University of Massachusetts Lowell, Lowell, MA 01854, USA
Chuta_Sano@student.uml.edu, {slin,xinwenfu}@cs.uml.edu
[4] Wartburg College, Waverly, IA 50677, USA
mogahid.fadl@wartburg.edu
[5] Towson University, Towson, MD 21252, USA
wyu@towson.edu
[6] University of Macau, Macau, China
weizhao@umac.mo

Abstract. We invent a novel context aware privacy enhancing keyboard (PEK) for touch-enabled devices to keep users safe from various password inference attacks. When a user inputs normal text like an email or a message, PEK shows a normal QWERTY keyboard. However, every time a user of a touch-enabled device presses a password input box on

the interaction between the hand and the keyboard is exploited. For example, the hand movement and finger position indicates which keys are being touched [7,18,19,21]. In sensor-based attacks [2,6,9,12,13,15,17], the malware senses a device's motion difference via its accelerometer (acceleration) and gyroscope (orientation) when different keys are touched and the device moves slightly.

To fight against these attacks listed above, we invent a novel context aware privacy enhancing keyboard (PEK) for touch-enabled device. The attacks introduced above can work because the keyboard keys are always at the same position. With PEK, every time a user of a touch-enabled device presses a password input box on the screen, we will randomly shuffle the positions of the characters on the keyboard and show this randomized keyboard to the user. That is, the user can derive a randomly shuffled keyboard every time while tapping their passwords on the screen. We maintain PEK's usability through its context aware feature: a randomized keyboard only shows up when a user inputs a password or pin. When a user inputs normal text like an email or a message, PEK shows a normal QWERTY keyboard or a system default keyboard. **We are the first to design a generic randomized keyboard for Android** while the idea of randomizing the key layout was proposed before for other applications with dedicated keypads [14]. PEK can be chosen as the default keyboard for Android so that it can be used for any app.

We released PEK as a free Android app to Google Play in August 2014 after our presentation at Black Hat USA [19]. It has been downloaded 2352 times at the time of writing. We released 7 versions of PEK, correcting bugs and improving the interface. PEK 1.0 is based on an Android code example. PEK 2.x.x is based on OpenWnn [11] although we fixed bugs and adapted it to later versions of Android. The current version of PEK is 3.1.0.0.

Since the number of PEK installations is below our expectation, for the purpose of usable security and privacy, we designed a two-stage usability test to evaluate the user experience of PEK and find out the reason behind the lukewarmness of using PEK. The first usability test was a pilot usability test. A major finding from the pilot test is the complicated installation and configuration processes discourage users from using PEK although installation and configuration instructions are given. We then performed the main usability test including a web survey and a focus group usability test. The web survey used Amazon Mechanical Turk. A major finding from the web survey and focus group study is that more people show interest in using PEK if a randomness toggle button is provided. With the button, users may enable or disable the random keyboard on the fly. Based on the usability test, we implemented a PEK app that allows a user to configure and enable PEK through an app on the launcher screen. We also add a randomness toggle button to the randomized keyboard.

The rest of this paper is organized as follows. We introduce the design and implementation of the third party keyboard of PEK in Sect. 2. The methodology of the usability test is presented in Sect. 3. The results of the usability test are given in Sect. 4. We conclude this paper in Sect. 5.

2 Privacy Enhancing Keyboard

In this section, we present the basic idea of the privacy enhancing keyboard. Given the limited space allowed in the paper, we do not include the technique details of PEK implementation. An extended version of technical report is available on demand.

To mitigate various attacks including residue-based attacks, computer vision-based attacks, and sensor-based attacks, we randomly shuffle the positions of keys of a software keyboard on a touch screen in order to show the user a randomized keyboard each time they input a password. As a result, profiles for particular keys cannot be established via vibration or orientation information through an accelerometer. Finger oily or thermal residue left on the screen does not imply particular keys. Vision based attacks also fail since a touched position by a finger does not refer to a fixed key.

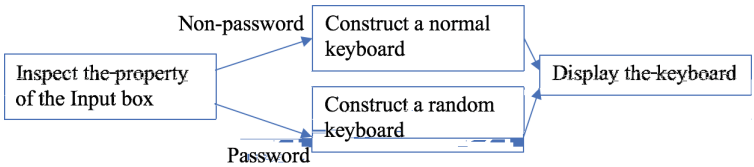


Fig. 1. Workflow of PEK constructing a keyboard

Figure 1 shows the basic idea and the workflow of PEK constructing a keyboard when a user touches an input box. First, we inspect the property of the input box to determine whether or not the input box is a password input box. If the input box is a password input box, we parse the property of the keys from a XML file that stores the layout of the keyboard, and change the label and value of the keys so as to shuffle the positions of the keys. If the input box is not a password input box, a QWERTY keyboard is shown.

We implemented two versions of PEK. One version is a third party keyboard implemented through an Android service that runs in the background. A third party keyboard is installed in the format of an Android app. A user has to find the system input setting menu in her phone in order to enable PEK. However, the location of the input setting menu is different in distinct phones. Before PEK 3.0, we provided a generic introduction to the input setting process and pretty much count on users to find the input setting menu. A note is we are also able to revise the source code of the Android system default keyboard and recompile it with the entire Android project. Apparently such a strategy implementing PEK is not practical for users. The second version of PEK is a 10-digit keypad for the unlock screen. To implement the randomized keypad, we have to revise the Android system source code, override the method "createKeyFromXml()" in the code file "PasswordEntryKeyboard.java" and recompile the entire Android project. Since a user has the option of choosing a conventional keyboard for the

unlock screen and recompilation of the entire Android project is not feasible for broad adoption, our usability study below focuses on the PEK - a third party keyboard and the term PEK refers to the third party keyboard particularly.

3 Usability Testing Methodology

In this section, we present our two-stage usability study of PEK: the pilot study and the main study, which are similar although the main study involves more participants, questions, and other measurements. In a usability study, in general there are not too many participants in the interview and focus group study. However, face-to-face interaction with participants provides us lots of detailed information/insights about users' view to our research questions. A web survey engages more subjects and produces quantitative and statistic results. That's the main difference between qualitative research (e.g., interview, focus group) and quantitative research methods. We used multiple methods to gather users' information from different perspectives.

3.1 Pilot Usability Test

There are two sessions in the pilot usability test that forms and improves the main usability test conducted after the completion of the pilot usability test. The first session is composed of a pre-survey with 10 questions, an interview with 5 open ended questions, and a post survey with 4 questions. Both the pre-survey and post-survey have multiple-choice questions so that the answers are easily interpreted and classified. Two to three days after the first one, the second session is conducted and includes an interview with 10 open ended questions. The interview involves recording the participants' answers and there is a portion of the interview, which was timed to see how long participants took to install and configure PEK. Three major issues are addressed during the pilot test.

- ***PQ1:*** After the release of PEK, there are some complaints on the Google Play Store page for PEK that specified that the configuration process was difficult. Hence, we want to find out the answers to the following questions. How easily can smart device users install and configure PEK onto their smart devices? Does the installation and configuration process discourage users from using PEK?
- ***PQ2:*** Perhaps the underlying reason why smart device users are not broadly employing PEK is simply because they are not interested in protecting their information and/or they are uneducated about security on their smart devices. Therefore, we ask: are smart device users in general concerned with the security on their phones?
- ***PQ3:*** When PEK is enabled and the user selects a password input box, the keyboard is randomized, therefore, it takes users longer to find the characters compared to when using a regular QWERTY keyboard.Whenr

3.2 Main Usability Test

The main usability test consists of a web survey and a focus group usability test based upon the findings in the pilot usability test. The web survey is hosted on the Qualtrics platform on Amazon Mechanical Turk and does not require any tasks from participants except completing the survey. Each participant is compensated a dollar for following directions and answering the survey honestly and correctly. The focus group usability test involves an interview. The participants are asked to install and configure PEK on their own devices and answer several questions. Four major issues are addressed during the main test.

- ***MQ1:*** What are the most frequent activities performed by smart device users on their personal devices? If the results showed one of the most frequent activities performed by smart device users involved sensitive information, they could be apart of PEK's target audience.
- ***MQ2:*** Do smart device users utilize any default security precautions already provided on their smart devices? This question relates to the one from the pilot usability test and whether or not typical smart device users are concerned with the security measures on their personal devices.
- ***MQ3:*** Do users consider that their smart devices are properly protected from outsider attacks?
- ***MQ4:*** Would smart device users consider implementing more security measures on their devices?

4 Usability Testing Results and Interpretation

This section presents results from the pilot usability test and main usability test performed between May and July 2016.

4.1 Answers for Pilot Usability Test

In the pilot usability test, there are 2 male participants who have Android mobile smart phones. During the interview, participants have to install and configure PEK on their own devices. They are timed for how long it takes them to successfully configure PEK and for the randomized keyboard to show up successfully when they try to input a password and/or pin.

Answers to Question PQ1: Users are able to find PEK on Google Play and install PEK without difficulty. However, when it comes to configuring PEK, some issues arise. Table 1 illustrates the time of installation and configuration during the pilot usability test. The configuration time is obviously longer. Along with the longer times, we note that both participants are not able to configure PEK by themselves; both of them need additional instructions from the researcher to configure the application. The participants **look for a PEK application icon** on their devices but find none. When they try to login to one of their accounts, such as an email, they are confused when the randomized keyboard does not show up when they hit a password field. The participants are frustrated during

Table 1. Installation and configuration time of PEK

Participants	Installation time (seconds)	Configuration time (seconds)
Participant 1	29.01	45.79
Participant 2	15.00	125.00

the configuration process. If the researcher does not aid them during the process, both of the participants most likely would have given up trying to configure PEK.

Answers to Question PQ2: Both participants admit that they would not use PEK on a regular basis on their own personal devices. Neither participant has information on their personal device that they consider sensitive. Nor do either of them have any other security enhancements enabled on their smart devices. The only security precaution Participant 1 admits undertaking is not using applications or services that request important data or sensitive data on their mobile phone; they prefer doing those types of activities in their home on their laptop or on their desktop. However both participants acknowledge that they might not be apart of PEK's target audience since both of them considered themselves educated about mobile security and how to prevent related attacks.

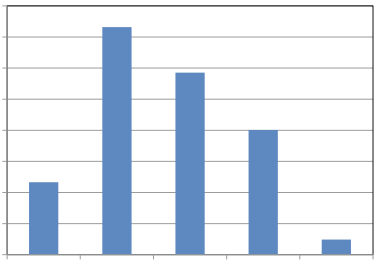
Answers to Question PQ3: Participant 1 during the second session after two to three days, does not consider the tradeoff between his time spent entering, for example, his pin to open his phone, worth protecting whatever personal information that is contained on his mobile phone. Participant 1 predicts that a user could never get better at entering a password and/or pin using PEK since the keyboard is randomized each time and no key is in the same place. Unlike a regular QWERTY keyboard, which a user can memorize and use easily, PEK cannot be learned. It is also challenging to multi-task when using PEK. For instance, if a user is on the move and trying to login to their phone, it is more difficult to login when using PEK than a regular QWERTY keyboard. Another difficulty that Participant 1 encounters is their mobile phone go to asleep when they attempt to enter their password using PEK to unlock their phone and the user has to enter their password all over again; this leaves Participant 1 frustrated at his time lost by using PEK. Unlike Participant 1, Participant 2 reckons his time lost by entering passwords using PEK is worth protecting the information stored on his mobile phone. Participant 2 compares PEK to someone using their hand to cover their screen while inputting their password with their other hand; except that PEK is more practical and dependable than a user's hand covering their screen.

Two observations can be made from the pilot usability test.

1. The configuration of PEK is difficult for both participants during the pilot usability test. Neither could complete configuration without aid. To remedy this, it is desirable to have more instructions on the Google Play Store to assist users and an icon for users to open when PEK is installed. Both the

participants look for a PEK icon on their mobile phone's interface when the application finishes downloading, however, PEK does not have an icon.
2. Participant 1 mentions difficulty using PEK when attempting to access their mobile phone quickly and while multitasking. We decide to create a separate

degree to which the web survey takers were concerned with security. The top answer was “Probably yes” at 36.59%, followed by “Maybe” at 29.27%, then “Probably not” at 20.05%. How users rate the degree of protection on their personal mobile devices may differ a lot from how they are actually protected. The high level of certainty the web survey takers display about their smart devices being protected is a little worrisome. Every smart device user should feel doubt when it comes to how well protected their smart devices are. Figure 4 portrays the distribution of answers web survey takers chose when asked this question.



applications as their top three activities they perform on their mobile smart device. Participant A most likely would not have a use for PEK. Participant B may be a more likely candidate for PEK and have more use for it than Participant A. However, neither lists any activities that were prominently chosen by the web survey takers on the web survey.

(b) *What kind of security have you implemented on your mobile phone?* Both Participant A and Participant B have the same exact answer for this question, "Nope." Neither has any default security installed on their mobile phones.

(c) *Are you satisfied with the level of security on your mobile phone?* Again, both Participant A and Participant B have the same answer for this question, a simple "Yes."

(d) *Would you ever consider adding more security features to your mobile phone?* Surprisingly both participants are somewhat open to considering implementing more security features to their mobile phones. Perhaps it is out of pure laziness that they do not have any security installed on their mobile devices, or they are sure to not perform any actions that require sensitive data on their mobile phones.

(e) *At this point during the interview we have both participants install and configure PEK.*

(f) *Would you recommend this application to a friend?* Participant A says yes they would recommend it to a friend who is concerned with security and who might be in public a lot. Participant B says as well that they would recommend PEK to a friend if and when a friend asks them about adding more security to their mobile phone.

(g) *Do either of you have any suggestions about improving the application?* Participant B's first impression of PEK is, "It can be used, but I will not use it." Participant A complains about the keys on PEK, how the larger popup disappears too quickly. For example, when you hit the key "U" on PEK, a popup will emerge from the key "U" and display a larger version of the letter, and that is for any letter when typing on PEK. Participant A recommends getting rid of this feature since he considers it annoying.

4.4 Improvements in PEK 3.x

In the pilot usability test, we learn that both two participants take long time to configure PEK, since they cannot find the PEK icon on their smartphones. To mitigate this problem, we put an icon of the PEK on the Android home screen as shown in Fig. 6. A user can tap the icon and configure the settings of PEK as shown in Fig. 7. Then, the user can click the "Open Android Input Settings" and set PEK as a keyboard available for users.

In addition, the participants suggest creating a new button for turning on/off the randomization of the PEK. Because PEK cannot be learned and it is inconvenient to use PEK in some circumstances. To this end, we implementa random

toggle button on the PEK as shown in Fig. 8. Then, users can decide to save time using a regular keyboard to input the password or protect their password using PEK.

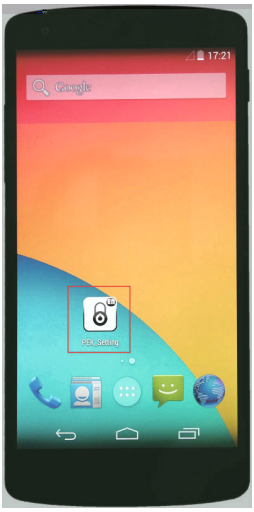


Fig. 6. Home screen app

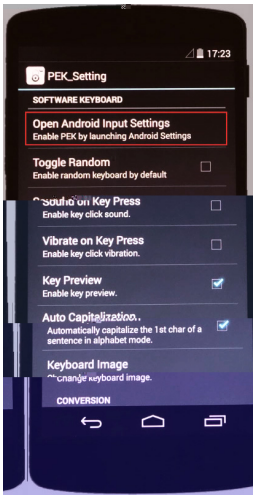


Fig. 7. PEK setting

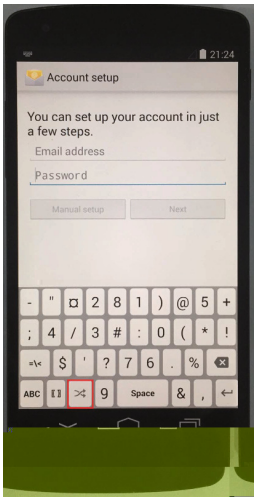


Fig. 8. Toggle button

5 Conclusion

This paper conducts a full-scale usability testing of a generic Android privacy enhancing keyboard (PEK) that can prevent various attacks against touch-enabled devices from inferring user pins or passwords. We perform both the pilot usability test and main usability test in order to identify how to improve PEK for broad adoption. Based on the results of the usability study, we implement two new features in PEK 3.x, a home screen app to easily activate PEK and a toggle button to enable/disable randomness of PEK. The usability test also demonstrates the worrisome phenomena that many users blindly trust their phones for security or are not concerned with the possible breaches. This phenomena demonstrates the human factor that contributes to the vulnerabilities of the cyber space. For future work, we plan to continue to improve PEK and perform another round of usability test in order to find out if the improved PEK attracts more adoption and better rating.

Acknowledgments. This work was supported in part by National Natural Science Foundation of China under grants 61502100, 61532013, 61402104, 61572130, 61602111, 61632008, and 61320106007, by US NSF grants 1461060, 1642124, 1547428, and CNS 1350145, by University System of Maryland Fund, by Jiangsu Provincial Natural Science Foundation of China under grants BK20150637 and BK20140648, by Jiangsu

Provincial Key Technology R&D Program under grants BE2014603, by Jiangsu Provincial Key Laboratory of Network and Information Security under grants BM2003201, by Key Laboratory of Computer Network and Information Integration of Ministry of Education of China under grants 93K-9 and by Collaborative Innovation Center of Novel Software Technology and Industrialization. Any opinions, findings, conclusions, and recommendations in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

References

1. Aviv, A.J., Gibson, K., Mossop, E., Blaze, M., Smith, J.M.: Smudge attacks on smartphone touch screens. In: Proceedings of Workshop on Offensive Technology WOOT (2010)
2. Aviv, A.J., Sapp, B., Blaze, M., Smith, J.M.: Practicality of accelerometer side channels on smartphones. In: Proceedings of the 28th Annual Computer Security Applications Conference (ACSAC) (2012)
3. Backes, M., Chen, T., D1rmuth, M., Lensch, H.P.A., Welk, M.: Tempest in a teapot: compromising reflections revisited. In: Proceedings of the 30th IEEE Symposium on Security and Privacy (S&P) (2009)
4. Backes, M., Duermuth, M., Unruh, D.: Compromising reflections - or - how to read LCD monitors around the corner. In: Proceedings of the 29th IEEE Symposium on Security and Privacy (S&P) (2008)
5. Balzarotti, D., Cova, M., Vigna, G.: Clearshot: eavesdropping on keyboard input from video. In: Proceedings of the 29th IEEE Symposium on Security and Privacy (S&P) (2008)
6. Cai, L., Chen, H.: TouchLogger: inferring keystrokes on touch screen from smartphone motion. In: Proceedings of the 6th USENIX Workshop on Hot Topics in Security (HotSec) (2011)
7. Cai, Z., He, Z., Guan, X., Li, Y.: Collective data-sanitization for preventing sensitive information inference attacks in social networks. IEEE Trans. Dependable Secur. Comput. (2016)
8. Maggi, F., Volpatto, A., Gasparini, S., Boracchi, G., Zanero, S.: A fast eavesdropping attack against touchscreens. In: Proceedings of the 7th International Conference Information Assurance and Security (IAS) (2011)
9. Miluzzoy, E., Varshavskyy, A., Balakrishnany, S., Choudhury, R.R.: Tapprints: your finger taps have fingerprints. In: Proceedings of the 10th International Conference on Mobile Systems, Applications, and Services (MobiSys) (2012)
10. Mowery, K., Meiklejohn, S., Savage, S.: Heat of the moment: characterizing the efficacy of thermal camera-based attacks. In: Proceedings of Workshop on Offensive Technologies (WOOT) (2011)
11. OMRON SOFTWARE Co., Ltd., Openwnn (2012). https://sourceforge.net/u/lluct/me722-cm/ci/890e9a90d9a7fe5f0243b9392eaa787d1381e987/tree/packages/inputmethods/OpenWnn/
12. Owusu, E., Han, J., Das, S., Perrig, A., Zhang, J.: ACCessory: keystroke inference using accelerometers on smartphones. In: Proceedings of the Thirteenth Workshop on Mobile Computing Systems and Applications (HotMobile). ACM, February 2012
13. Ping, D., Sun, X., Mao, B.: Textlogger: inferring longer inputs on touch screen using motion sensors. In: Proceedings of the 7th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WiSec) (2015)

14. Shin, H.-S.: Device and method for inputting password using random keypad. United States Patent No. 7, 698, 563 (2010)
15. Simon, L., Anderson, R.: Pin skimmer: inferring pins through the camera and microphone. In: Proceedings of the 4th ACM Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM) (2013)
16. Sun, J., Jin, X., Chen, Y., Zhang, J., Zhang, R., Zhang, Y.: VISIBLE: video-assisted keystroke inference from tablet backside motion. In: Proceedings of the 23rd ISOC Network and Distributed System Security Symposium (NDSS) (2016)
17. Xu, Z., Bai, K., Zhu, S.: Taplogger: inferring user inputs on smartphone touchscreens using on-board motion sensors. In: Proceedings of The ACM Conference on Wireless Network Security (WiSec) (2012)
18. Yue, Q., Ling, Z., Fu, X., Liu, B., Ren, K., Zhao, W.: Blind recognition of touched keys on mobile devices. In: Proceedings of the 21st ACM Conference on Computer and Communications Security (CCS) (2014)
19. Yue, Q., Ling, Z., Fu, X., Liu, B., Yu, W., Zhao, W.: My google glass sees your passwords! In: Proceedings of the Black Hat USA (2014)
20. Zalewski, M.: Cracking safes with thermal imaging (2005). http://lcamtuf.coredump.cx/tsafe/
21. Zhang, L., Cai, Z., Wang, X.: Fakemask: a novel privacy preserving approach for smartphones. IEEE Trans. Netw. Serv. Manag. **13**(2), 335–348 (2016)
22. Zhang, Y., Xia, P., Luo, J., Ling, Z., Liu, B., Fu, X.: Fingerprint attack against touch-enabled devices. In: Proceedings of the 2nd Workshop on Security and Privacy in Smartphones and Mobile Devices (SPSM) (2012)