Eyes-Free Gesture Passwords – A Comparison of various Eyes-Free Input Methods

Chen Chen **Soon Hau Chua** **David Chung Wing Kei** **Simon T. Perrault** **Shengdong Zhao**
NUS HCI Lab, School of Computing, National University of Singapore (NUS)
117418, Singapore
{chchencc, chuasoonhau, cwkdavid, perrault.simon}@gmail.com, zhaosd@comp.nus.edu.sg

Figure 1a. Augmented Number Pad Input Interface

Figure 1b. Wheel Input Interface

Figure 1c. Stroke Input Interface

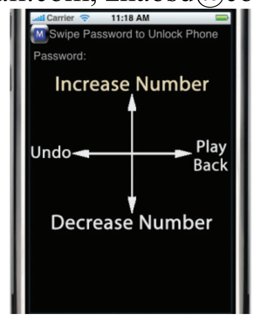

Figure 1d. Scroll Input Interface

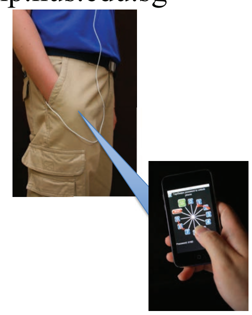

Figure 1e. Using BlindPass in a pocket

ABSTRACT

With modern smartphones, users can access to secured applications such as e-banking, booking flight tickets, and etc. Using these applications in public areas is risky since it is easy for people around the user to peep at the password input. One way to enhance security is to enter the passwords inside one's pocket or purse in an eyes-free manner. In this paper, we designed, prototyped and evaluated four eyes-free password entry methods (Number Pad, Wheel, Stroke, and Scroll) on the iPhone. Our experiment results showed the comparative advantages of the Wheel (Figure 1b) and Stroke (Figure 1c) methods. In addition to that, guidelines and implications for designing eyes-free password entry methods are also discussed.

AUTHOR KEYWORDS

Security; passwords; eyes-free; gesture.

ACM Classification Keywords

H.5.2. [Information interfaces and presentation]: User Interfaces - Prototyping.

INTRODUCTION

The introduction of smartphones with internet capabilities has increased the mobile internet usage in public areas. A study by Nylander, Lundquist and Brännström [6] showed that 23% of the participants used mobile phones to access the Internet outdoors, another 23% in transit, and 16% at indoor public areas. Security and privacy issues could arise when users are entering passwords in public areas. People who are standing close to the user entering the password can easily glance at the user's inputs and obtain the password.

So far, authentication methods like digital signatures and biometrics have been used in smartphones to enhance security. SecurePhone [8] enables biometrically authenticated users to deal mobile contracts (m-contracts) during a mobile phone call in a secure way through digital signature, face and audio recognition. However, biometric recognition often requires special hardware (e.g. fingerprint scanner for fingerprint recognition), limiting its implementation in the current crop of smartphones devices.

In our preliminary survey with 114 users, 59% expressed that they are uncomfortable with entering passwords in public. Meanwhile, 68% of the respondents indicated that entering passwords in an eyes-free manner is desirable because it can prevent people nearby from seeing their passwords in public areas.

With the security risks in mind, we present BlindPass. BlindPass allows users to enter password in an enclosed area, such as one's pocket or purse, while people around the user are unable to see the entered password. We facilitate its ease of use by suggesting various input interfaces and providing audio feedback to users through headphones.

The challenge in designing effective eyes-free user interfaces for mobile devices is in the rethinking of the traditional desktop user interfaces [1] and the absence of visual feedback. Without visual feedback, icons and buttons need to be large enough for high accessibility. From our pilot study, we found that a perceived vertical straight line

Chinese CHI '14, April 26 - 27 2014, Toronto, ON, Canada

http://dx.doi.org/10.1145/2592235.2592248

that a user drew without visual feedback is not a straight line drawn in the mobile device (Figure 2a). Likewise, when users lifted up their finger to end an input session, it left a tail on the touch input (Figure 2b). Hence, our gesture recognition algorithm has to recognize these features and cater for different degrees of recognition and acceptance rates. We also provide different audio feedback for tapping and error inputs so that users know their input to the system [10]. For realistic use of BlindPass in the public, users will need to use headphones to listen to the auditory feedback.

Figure 2a. Straight line drawn by a right thumb performing the swiping motion

Figure 2b. 'Tail' generated by lifting up the right thumb after performing a left-to-right swipe motion.

RELATED WORK

Eyes-free interaction

There are a number of existing works on eyes-free interaction technique. However, due to space limitation, we only reviewed those that directly influenced our design. In 2007, earPod [10] was developed to provide eyes-free menu selection with touch input and reactive audio feedback for the iPod. This input technique is similar to one of our eyes-free input methods (the Wheel). Despite that, we do not have the plastic cut out in earPod to provide haptic guidelines, since it prohibits the normal use of the rectangular touch screen.

Samsung developed *Gesture Lock* for their touch screen smartphones. It allows users to unlock the device and launch application by writing an English character at the lock screen. However, by just using a single English character is too simple and vulnerable to break in. We aim to support a more secure method of password entry that can be a pin number of any length.

At Google I/O 2009, Raman [7] demonstrated an adaptive interface for smartphones that provide audio and tactile feedback. This allows users to enter numbers in an eyes-free manner. The interface works by designating the first tap on the touch screen as number 5. Further strokes in other directions allow users to access other numbers in orientation that mimics the number pad format. In this work, the number pads of the Augmented Number Pad interface we designed are spatially fixed across the screen, similar to the virtualization of hard keys in a conventional mobile phone to provide closer comparison between the conventional number keypad input technique versus the eyes free input techniques.

Mobile Authentication Security

In 2012, De Luca et al. [2] proved that using an implicit authentication layer could improve mobile security. They tracked the touch screen data (e.g. speed and time) of the mobile owner and used them to identify the rightful user.

De Luca et al. later proposed "BoD Shapes" [3], which they identified as the best concept to enhance security and speed of Back-of-Device Authentication. Back-of-Device Authentication is also a practical way to prohibit unwanted observations.

Zezschwitz et al. conducted [9] a field study to compare the performance, usability and likeability between Graphical Passwords and personal identification numbers (PINs). They found that even though PINs won much in terms of speed and accuracy, users tended to prefer Graphical Passwords.

DESIGN OF EYES-FREE PASSWORD INTERFACES

For BlindPass, we designed and prototyped four different eyes-free password entry methods, all of which are described below.

Augmented Number Pad Entry

One easy way to allow eyes-free password entry is to augment the conventional input methods such as number pad (Figure 1a) with auditory feedback, tactile feedback, or both. To enter a password, users can enter the password by tapping on the respective number buttons or by lifting up the finger when the system plays desired number through the headphones after a continuous swipe. Due to the lack of visual feedback, additional editing functions include undo, reset, and playback were added. The undo feature removes the last entered number, while the reset function clears all the entire password string entered. The small button at the top left corner of the screen provides an audio/tactile queue to the user that he/she has reached the top left corner of the touch screen.

Wheel Input Method

The Wheel input method interface (Figure 1b) mimics the spatial location of numbers in the clock face and an input technique similar a circular number lock (or menu selection as in earPod [10]). Users can tap the numbers according to their spatial difference, or swipe along the screen to enter a password. This swiping action could be in a circular motion, as if turning a physical circular number lock, or a straight line. The editing functions include reset and playback. From the preliminary survey, 28.2% of the participants like this input method.

Stroke Input Method

The Stroke input method allows users to enter the password using gesture, similar to a marking menu [5]. The system maps each number to a specific gesture (Figure 1c). There are no fixed positions where the user must start the gesture. This method is more suited for advanced users where they could remember the gesture rather than the actual number combination. Besides, this method has a higher entropy compared to the other methods. Let us consider the asterisk symbol gesture as an example: a different way of forming the asterisk will result in a different number combination. If the user makes the vertical (|) swipe first, followed by the horizontal (–) and the two diagonal swipes, the number formed is 6357. If user were to perform the diagonal swipes before the horizontal and horizontal swipes, the number formed is 5736. From our preliminary survey, 25.9% of the respondents like this input method.

Scroll Input Method

The Scroll input method was inspired by the elevator system. Users input the password by swiping up or down from any position on the screen until the desired number is reached before lifting up their finger (Figure 1d). An upward swipe motion increases the number, analogous to an up-going elevator. A fast upward swipe motion increases the speed of the number increment, and vice-versa for downward swipe motion. In order to select the extreme numbers (nine and zero) quickly, the increment and decrement will stop at number nine and zero respectively. Unlike the Google eyes-free interface where every new tap on the screen is the number 5, our system continues from the number that the user stopped previously. The Scroll input method helps users to decide the next swipe motion in order to enter the next number quickly.

PILOT USER STUDY

We recruited a group of ten participants with no prior knowledge about these entry methods to participate in the pilot study conducted in a computer laboratory. The purpose of the pilot study was to identify and investigate the problems encountered while inputting the eyes-free password. Participants' feedbacks were then gathered to improve the input techniques.

Results

Across all interfaces, we observed a difference between what the users drew and what they thought they drew (as shown in Figure 2a and Figure 2b). Such irregularities caused the system to detect a different gesture, resulting in an incorrect number input. As a result, we modified the detection algorithm to recognize these tails.

For people who have big palms and thick thumbs, the region that is closest to the palm is difficult to reach. We made two improvements to address this problem. First, we enlarged the buttons for the Wheel input method to cater for greater accessibility to all number keypads. Second, we designed a new input method (Scroll method) that allowed convenient movement of the thumb (upward and downward swipes) to control the input of the password.

EXPERIMENT

We conducted an experiment to investigate the performance of each input interfaces that we designed and also to demonstrate that eyes-free password entry methods are actually deployable and accepted for use in the public. We used the data from Augmented Number Pad to benchmark against other methods. After the experiment, participants filled up a questionnaire to provide subjective feedback.

Setup and Procedure

We conducted the experiment over a week where participants learn the various entry methods and use the methods in the computer laboratory. Participants used headphones to listen to the auditory feedback. In order to minimize learning effect and bias to a particular input method, the attempt sequence of the experiment was counterbalanced with the Latin Square method, so the order of techniques was counterbalanced between users [1]. During the experiment we asked the users to conceal the phone under a table so they could not see it.

Participants

31 participants took part in the experiment, all aged between 25 to 30 years old, and were either third or fourth year undergraduate students in the university. Participants were compensated with credit point in their module coursework for their participation in the experiment. Before the experiment, all participants were given ample time to study and familiarize themselves with the Apple iPhone's iOS user interface.

Task

First, participants changed the default password to their preferred four digits password. Then, they entered this password ten times for all four interfaces. When all four interfaces were completed, participants changed to an eight digits password and repeated the tasks of entering ten times each for all the four interfaces.

In summary, the experiment was designed as the following: 4 interfaces (Augmented Number Pad, Wheel, Stroke and Scroll) × 2 sets of passwords (four and eight character password) × 10 entry repetitions = 80 trials per participants.

Two participants volunteered to do the experiment in the public. Together with the experimenter, the experiment was conducted at the university's canteen and internal shuttle bus trip in the afternoon.

Execution time and the number of correction gestures per trial were measured in the experiment. A correction gesture happens whenever a user performs an incorrect input and thus has to perform a new gesture to correct the error.

Results

Figure 3 below shows the means for the completion time and number of correction gestures for each of the interfaces.

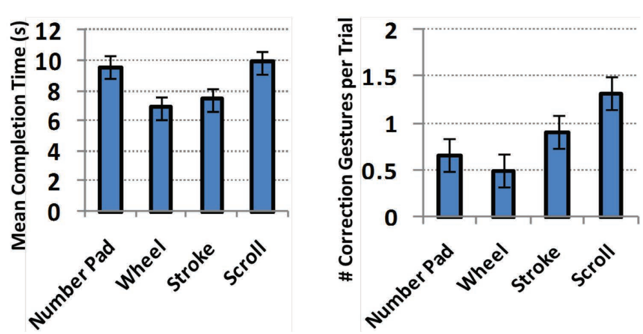

Figure 3. Means for the completion time and number of correction gestures for all four interfaces

In terms of the four-character password, there is a significant main effect on method ($F_{3,90}$=2.86, p=.04) for mean completion time. Pair-wise comparison (LSD) shows that Wheel (6.87s) has significantly faster completion time than the Augmented Number Pad (9.56s) (p<.05). The Stroke (7.40s) has slightly longer completion time than the Wheel but faster than Augmented Number Pad, while no significant difference was observed with the Scroll method.

There is also a significant main effect on method ($F_{3,90}$=6.49, p=.001) for the mean number of correction gestures per trial. Pair-wise comparison (LSD) shows that both wheel (.50 correction gestures used per trial) and tap (0.66 correction gestures per trial) have significantly less correction gestures than Scroll (1.31) and Stroke (.9) (p<.05). However, between Wheel and Tap, or between Scroll and Stroke, there is no significant difference.

Participants preferred the Augmented Number Pad for the eight-character password. Experiment result shows that completion time for all four input methods took approximately 1.5 times longer than the four-character password (Augmented Number Pad at 9.98s, Wheel at 11.24s, Stroke at 11.68s while Scroll at 15.48s).

Qualitative Feedback

In the post-experiment questionnaire, we conducted a semi structured interview where we asked participants to comment on each technique. Most participants commented that the Scroll technique was easy to enter given the easy movement of the thumb. However, they felt that the gesture recognition algorithm needs to be improved. Participants also recommended adding both reset and undoing functions to all interfaces to edit the password easily.

For the two participants who did the experiment in more realistic situations (at the canteen and on the bus), they commented that it was still easy to perform the gestures when hiding the iPhone in their bag. Surrounding environmental noise did not affect the sound quality from the headphones and the experimenter reported that he could not hear the audio from the participant's headphones.

DISCUSSION AND DESIGN IMPLICATIONS

The Wheel input method is faster than the Augmented Number Pad because users are able to access them quickly without having drastic movements on the screen compared to the number pad. When number keypads are arranged in a circular manner, users can easily tap on them without making difficult movements to access hard-to-reach corners of the screen. Similarly, the Stroke input method is faster because it does not rely on any spatial separation of number keypads. Users are able to perform their gesture anywhere in the screen that is convenient.

In our original BlindPass design, we included tactile feedback [4]. One tactile feedback represents odd numbers entered, while two tactile feedbacks represent an even number. With tactile feedback, users have another alternate feedback mechanism if they do not use headphones for audio feedback. However, we need to fine-tune the prototypes before an experiment could be carried out with tactile feedback. Research on ways to provide seamless tactile feedback on smartphones remains an open topic for future studies.

CONCLUSION AND FUTURE WORK

In this paper, we presented BlindPass, a new password entry method that prevents people nearby from peeping password entries in public. Our experiment showed that eyes-free passwords are easy to use and relatively easy to learn. Although the Scroll technique is comparable with the Augmented Number Pad technique, it appears that the Wheel and Stroke input methods both have a faster completion time.

Since BlindPass allows users to enter passwords discreetly and securely in public areas, it could be deployed in several different scenarios. On top of locking/unlocking mobile devices, as we have demonstrated in this paper, BlindPass can also be integrated with mobile browsers and applications to use eyes-free passwords for secure login. Finally, BlindPass can be extended to other applications such as military or disabled users where it is important to perform password entry or security authorizations in a secure manner.

REFERENCES

1. Campbell, D. T., Stanley; J. C. Experimental and quasi-experimental designs for research. Boston : Houghton Mifflin, 1980.
2. De Luca, A., Hang, A., Brudy, F., Lindner, C., and Hussmann, H. Touch me once and i know it's you!: Implicit authentication based on touch screen patterns. In *Proc CHI'12*. ACM, 987–996.
3. De Luca, A., Von Zezschwitz, E., Nguyen, N. D. H., Maurer, M.-E., Rubegni, E., Scipioni, M. P. and Langheinrich, M. Back-of-device authentication on smartphones. In *Proc CHI'13*. ACM, 2389-2398.
4. iPhone-haptics http://code.google.com/p/iphone-haptics/
5. Kurtenbach, G. and Buxton, W. User learning and performance with marking menus. In *Proc. CHI'94.* ACM*,* 258-264.
6. Nylander, S., Lundquist, Terés, Brännström, A. At Home and with Computer Access – Why and Where People Use Cell Phones to Access the Internet. In *Proc. CHI'09*. ACM, 1639-1642.
7. Raman, T.V. and Chen, C.L. Eyes-Free Interfaces. *IEEE Computers*, Volume 42, Issue 10, 100-101.
8. SecurePhone http://www.secure-phone.info/
9. Von Zezschwitz, E., Dunphy, P. and De Luca, A. Patterns in the wild: a field study of the usability of pattern and pin-based authentication on mobile devices. In *Proc MobileHCI '13*. ACM, 261-270.
10. Zhao, S., Dragicevic, P., Chignell, M. H., Balakrishnan, R. and Baudisch, P. earPod: Eyes-free Menu Selection with Touch Input and Reactive Audio Feedback. In *Proc CHI'07*. ACM, 1395-1404.