Password Entry Usability and Shoulder Surfing Susceptibility on Different Smartphone Platforms

Florian Schaub, Ruben Deyhle, Michael Weber
Institute of Media Informatics
Ulm University
89069 Ulm, Germany
{florian.schaub | michael.weber}@uni-ulm.de, ruben.deyhle@alumni.uni-ulm.de

ABSTRACT

Virtual keyboards of different smartphone platforms seem quite similar at first glance, but the transformation from a physical to a virtual keyboard on a small-scale display results in user experience variations that cause significant differences in usability as well as shoulder surfing susceptibility, i.e., the risk of a bystander observing what is being typed. In our work, we investigate the impact of both aspects on the security of text-based password entry on mobile devices. In a between subjects study with 80 participants, we analyzed usability and shoulder surfing susceptibility of password entry on different mobile platforms (iOS, Android, Windows Phone, Symbian, MeeGo). Our results show significant differences in the usability of password entry (required password entry time, typing accuracy) and susceptibility to shoulder surfing. Our results provide insights for security-aware design of on-screen keyboards and for password composition strategies tailored to entry on smartphones.

Categories and Subject Descriptors

H.1.2 [**User/Machine Systems**]: Human factors; H.5.2 [**User Interfaces**]: Graphical user interfaces (GUI); K.6.5 [**Security and Protection**]: Authentication

General Terms

Human Factors, Security, Experimentation

Keywords

password, shoulder surfing, smartphone, touchscreen, usability, user experience, virtual keyboard

1. MOTIVATION

Smartphones with mobile Internet access are becoming increasingly common. Most smartphones have replaced physical keyboards with virtual keyboards displayed on the device's touchscreen. Yet, most websites and online services require text-based passwords for authentication. Corresponding password composition rules and recommendations are often optimized for desktop keyboards. They emphasize password length and a mixture of different character types (lowercase, uppercase, numbers, and special characters) to increase password entropy [13]. The shift from physical to virtual keyboards has been shown to influence typing ability, due to the lack of tactile feedback and the size of soft keys [21, 16]. Furthermore, the typing effort of certain characters varies between physical and virtual keyboards. On smartphone keyboards, typing numbers or special characters requires navigation to a second or third keyboard page. Therefore, we formulate the hypothesis that the design and layout of virtual smartphone keyboards affects password entry performance. Smartphone-based password entry also increases the risk of shoulder surfing, i.e., someone observing what is being typed. Reasons are that smartphones are often used in public places and that virtual keyboards employ accessibility features to improve typing accuracy, such as magnification of the typed character or displaying the last typed character as cleartext in the password entry field. Although virtual keyboards of different mobile platforms seem quite similar at first glance, differences exist in their user experience (see Sec. 3) that affect the usability and shoulder surfing susceptibility of password entry.

In order to investigate the specific impact of different virtual keyboards on smartphone-based password entry, we performed a between subjects study with 80 participants, in which we investigated password entry performance and shoulder surfing susceptibility on common mobile platforms (iOS, Android, Windows Phone, Symbian, MeeGo). Our results show that significant differences in terms of password entry exist between these platforms. The results further provide insights on how specific virtual keyboard design elements can improve password entry security by improving password entry usability, while reducing shoulder surfing susceptibility. To our knowledge, this is the first study that explicitly evaluates usability and shoulder surfing aspects of text password entry on virtual keyboards. In addition, our results also provide indicators for improving password composition rules with respect to mobile devices.

We first discuss related work in Section 2. Section 3 assesses characteristics of the virtual keyboards investigated in the study. Our study design and hypotheses are described in Section 4. Our results are presented in Section 5 and discussed in Section 6. Section 7 concludes the paper.

MUM '12, December 3-6 2012, Ulm, Germany

2. RELATED WORK

Password-based authentication and potential alternatives, such as graphical passwords, have been studied extensively. For sake of brevity, we limit our discussion to work directly pertaining to text passwords.

2.1 Usability of Text Passwords

Bonneau & Preibusch [4] find in an analysis of 150 websites that the majority did not enforce password composition restrictions besides requiring a minimum password length, i.e., $\geq$6 characters for 52% of analyzed sites. Florencio & Herley [6] study password habits of website users. They find that most users use lowercase-only passwords. However, security policies of companies typically pose more restrictions on password composition. Inglesant & Sasse [8] find that inflexible password policies, which focus on password strength rather than context of use, decrease password usability. Users also develop coping strategies, such as writing down passwords, that undermine a password's effectiveness. In an analysis of user-created passwords, Komanduri et al. [13] find that requiring 16 character passwords without composition rules resulted in passwords with higher entropy than asking for 8 character passwords with comprehensive composition rules. However, Keith et al. [11] find that users with longer passwords needed more login attempts due to recall errors. These results indicate that password composition plays a major role in password usability.

Mnemonic passwords, for which the password characters are taken from a mnemonic sentence, can be as strong as randomly generated passwords but are easier to remember [26]. Jeyaraman & Topkara [10] propose a mechanism to automatically generate mnemonics from a text-corpus. However, Kuo et al. [15] show that mnemonic passwords can be attacked by assembling a dictionary from phrases available online. Forget et al. [7] propose persuasive text passwords (PTP) to enhance the security of user-generated passwords without explicit composition rules, while retaining memorability [1]. When creating a new password, a user freely chooses a password, which is then enhanced with randomly generated characters by PTP. The user can request different random characters until comfortable with the result. Biddle et al. [2] propose a text password scheme in which digital objects function as mnemonics. The user selects a set of personal files, e.g., images or videos, which are then hashed. The resulting hash is converted into a password string. Thus, only the used files not the specific password need to be remembered, but they must be available on the login device. While different usability and security enhancements for text passwords have been proposed [3], their impact on password entry on virtual keyboards has typically not been considered.

2.2 Shoulder Surfing Susceptibility

Tari et al. [25] study shoulder surfing of text passwords and the graphical password scheme PassFaces. Twenty participants were asked to reproduce passwords entered by the experimenter on a computer. Surprisingly, non-dictionary passwords were easier to shoulder surf than dictionary-based passwords in their study. Kim et al. [12] follow the same study design to evaluate touch-enhanced PIN entry mechanisms on tabletop displays. In Nicholson's study [20] participants also acted as shoulder surfers observing the experimenter entering PINs and different graphical passwords from three fixed positions. Participants had to solve a memory rotation task before entering the observed password to force recollection from long-term memory. Their results show that PIN entry is more vulnerable to shoulder surfing than graphical password schemes. Zakaria et al. [27] evaluate enhancements to the recall-based draw a secret (DAS) scheme. The experimenter entered three DAS passwords on a PDA observed by participants standing to the left. Dunphy et al. [5] focused in their smartphone-based shoulder surfing study on how many observations were required by participants to reproduce graphical passwords with different entropy. Participants were randomly assigned the roles of victim and observer. The observer could request to see the login attempt up to ten times and could stop when confident to be able to reproduce the password. Dunphy et al. find that high entropy passwords require 7.5 observations on average compared to 4.5 for low entropy passwords.

A common approach for improving shoulder surfing resistance is overwhelming the observer's short-term memory [5]. Tan et al. [24] propose a spy resistant keyboard for public touch displays that separates mapping and selection. The keyboard produces a randomized mapping between characters and a property (e.g., color). The user has to remember the color for the desired character. In the selection phase, the mapping is removed and the user selects by property. While the user only needs to remember the mapping for one character, a shoulder surfer would need to remember the mappings for all characters. In a study where pairs of participants played victim and observer, Tan et al. find that their keyboard is less vulnerable to shoulder surfing compared to a normal virtual keyboard. However, typing takes twice as long. Roth et al. [22] employ a similar approach for PIN entry. PIN pad buttons are randomly separated by color into two groups. For each PIN digit, the user indicates the color group containing the digit in multiple rounds. The user never explicitly enters the PIN, but PIN entry takes substantially longer this way. In a small-scale study (n=8), they find their scheme more resistant to shoulder surfing than normal PIN entry. Zhao & Li [28] also avoid direct entry of password characters by letting users click inside a convex triangle shaped on the keyboard by the next three password characters. Kumar et al. [14] suggest gaze-based password entry with eye tracking. Their assumption is that if the user does not touch the keyboard, the shoulder surfing risk should be reduced. However, a shoulder surfing study has not been performed yet. Sasamoto et al. [23] use visible and hidden channels to convey password challenges that must be combined by the user to give the correct answer. For example, presenting a visual yes/no question and instructing the user via headphones to lie or tell the truth.

All discussed studies compare different password entry mechanisms. However, the effect of different instantiations of the same authentication mechanism on usability and shoulder surfing susceptibility is rarely analyzed. Furthermore, to our knowledge, no studies have directly evaluated password entry on virtual smartphone keyboards so far.

3. VIRTUAL KEYBOARD VARIANTS

We employed multiple virtual keyboards in our study. In order to reduce complexity, we only consider portrait orien-

tation of the virtual keyboard, which is the default orientation on most devices. Keyboard sounds and vibration were deactivated. Because all test subjects were German, the German keyboard layout (QWERTZ) was used on all systems. An additional keyboard layout (US) was enabled, triggering all keyboards to display a layout toggle button. All keyboards consist of four pages: the initial page shows lowercase characters, the shift button activates a second page with uppercase characters. A third and fourth page contain numbers and special characters. For Android and Symbian, we included multiple common keyboards, resulting in eight variants in total. Figure 1 shows the primary page of the analyzed virtual keyboards, we describe their specific characteristics in the following.

iOS. The iOS keyboard always displays uppercase characters on the first page, only a highlighted shift key indicates uppercase input. A key press triggers a popup showing the typed character, as is the case for all other keyboards unless stated otherwise. Holding down the key of some characters allows to select similar characters in a pop-out menu. The "123" button toggles the first special characters page. We used an iPhone 4S with iOS 5.0.1 in our study.

Android-Vanilla. The plain Android keyboard offers numbers as alternate functions for the first row of letters, which can be typed by pressing the key slightly longer. The first page shows additional buttons for period, comma, and keyboard preferences. The button "?123" toggles the first special character page, which also contains dedicated number keys. We used a Google Nexus One with Android 2.3.6.

Android-Sense. The keyboard of the HTC Sense UI has almost white keys. All letter keys have special characters as alternate functions. Dedicated keys for period and comma exist. A small button captioned "12#" toggles the special characters page. The grave accent is not available on the keyboard. We used an HTC Desire with Android 2.2.

Android-Swype. The Swype keyboard supports typing by drawing a path over keys. While this feature is not activated for password entry, the keyboard has been included because of its popularity. All keys on the first page have special characters as alternate functions. The alternate characters change when shift is activated. The first page features additional keys for German umlauts. The first row of letter keys is aligned in columns to the other rows and not offset. We used a Google Nexus One with Android 2.3.6 for Swype.

Windows Phone. The virtual keyboard of Windows Phone displays keys as grey boxes without any decoration, resulting in a clean look. The keyboard has additional keys for period and comma. We used an HTC 7 Trophy with Windows Phone 7.5.

Symbian-QWERTZ. The Symbian keyboard has dedicated keys for German umlauts, period, comma, question mark, and ß on the first page. This results in very narrow and vertically aligned keys. Shift changes the displayed special characters. Numbers on the special characters page are arranged in a numpad layout with three columns. An extra menu button activates a larger map with special characters. We used a Nokia N8 with Symbianˆ3.

(a) iOS (b) Android-Vanilla (c) Android-Sense (d) Android-Swype (e) Windows Phone (f) Symbian-QWERTZ (g) Symbian-T9 (h) MeeGo

Figure 1: Primary pages of the analyzed virtual keyboard variants.

Symbian-T9. Symbianˆ3 also offers a virtual numpad with T9 completion as known from feature phones. This keyboard only has twelve buttons: numbers, *, and shift. T9 is deactivated for password entry, keys have to be tapped repeatedly to input characters. The keyboard does not display a popup of the typed key. Instead of special character keyboard pages, a character map (same as Symbian-QWERTZ) is used, which consists of two pages filling the entire screen. The grave accent is missing on the keyboard. We used a Nokia C7 with Symbianˆ3.

MeeGo. MeeGo is a Linux-based mobile platform originally developed by Nokia and Intel. The keyboard has black keys with plain white captions, aligned horizontally and vertically. It has additional keys for German umlauts, comma and period. Keys are larger compared to Symbian-QWERTZ. Curly brackets are missing on this keyboard. We used a Nokia N9 with MeeGo 1.2.

3.1 Character Typing Effort

The required effort for typing characters varies on virtual keyboards, because of required navigation between keyboard pages. To ensure comparability of password entry across devices, the effort of typing the passwords in our study must be the same on all devices. Therefore, we partitioned available characters[1] into categories of same effort: lowercase (*z*), uppercase (*Z*), numbers (*0*), and four categories for special characters (*1-4*). While the first three categories already exhibit consistent effort across devices, typing effort of special characters had to be analyzed on each keyboard variant. Table 1 shows the required number of taps for each special character on different platforms (a tap-and-swipe gesture corresponds to 1.5 taps). If multiple options exist for typing a character, the minimum effort was chosen. The clustering of special characters into the different categories (tier 1-4) is based on the mean effort across all keyboards. Thus, the special characters in one category require approximately the same input effort. Later on, we define password patterns for our study based on these categories. We analyzed 62,000 passwords leaked by LulzSec in June 2011 to determine the occurrence probability of each character in its category to guide construction of the password patterns.

4. STUDY DESIGN

The virtual keyboard variants discussed in Section 3 exhibit interesting differences. We hypothesize that these subtle differences cause significant differences in usability and shoulder surfing susceptibility. Therefore, we designed a usability experiment and a shoulder surfing experiment. We opted for a between subjects design with the keyboard variant as independent variable, resulting in eight experimental groups. Each participant used a single virtual keyboard to perform one usability experiment and one shoulder surfing experiment. In contrast to a repeated measures test, the between subjects approach provided a number of advantages. First, confronting each participant with eight keyboard variants would have inadvertently caused exhaustion and training effects that could have skewed results even for randomized ordering of keyboard variants. Second, for the shoulder surfing experiment, it was essential to use the same passwords for all participants to achieve reliable results. This would not have been possible in a repeated measures design where participants would perform the shoulder surfing experiment multiple times.

The study was conducted at Ulm University in a dedicated room as a lab study. The room contained a desktop computer to complete questionnaires and the shoulder surfing setup, consisting of a chair and table. For all smartphones cellular services were disabled. Wifi was used to connect to a university server which hosted the study's web application and stored collected data. Participants were recruited from the campus population with posters and flyers and via mailing lists of the university. Participants were promised and received chocolate. In total, 80 people participated in the study, resulting in 8 groups with 10 participants. The 21 female and 59 male participants were roughly equally distributed between groups. The majority of participants were students and academic personnel from the fields of computer science, media computer science, and mathematics. The average age of participants was 24. The majority owned a smartphone (68.75%), 23 participants owned a standard feature phone (28.75%), and 2 participants owned no mobile phone (2.5%).

Each session started with an *entry questionnaire* to be filled out on a desktop computer. First, a short text explained the goal of the study. It was further pointed out that participants did not need to use any personal passwords and that all required passwords would be provided. The questionnaire gathered demographic information (age, gender, occupation) and asked participants to rate their experience with common smartphone platforms, as well as specific technologies (multitouch screens, T9 keyboards, QWERTZ keyboards), on 7-point Likert scales. Participants were further asked to specify brand and model of their primary mobile phone. The questionnaire closed by asking the participant to use their prescription glasses, if required.

The experience information was directly evaluated to assign the participant to a group. Participants were preferably assigned a mobile platform they were already familiar with. This had two advantages over random group assignment: participants already familiar with a specific keyboard would require less training and typing behavior would reflect natural use, thus enhancing ecological validity of measurements. Regardless of experience, all participants received sufficient instruction and ample training time to properly familiarize themselves with the assigned keyboard variants. Participants were encouraged to type a provided sample text on the assigned smartphone, which contained characters of different categories (see Sec. 3.1). Participants were especially encouraged to familiarize themselves with the entry of special characters. Participants could start the first experiment whenever they felt ready.

4.1 Usability Experiment

To assess the usability of virtual keyboards, we employed two metrics reflecting typing performance: *entry time* and *mean error rate* for entering a password. Entry time is an indicator for the difficulty of locating required characters. The error rate is an indicator for typing accuracy, i.e., how often do users have to correct mistyped characters. In addition to quantitive usability results, we were also interested in qualitative assessment of perceived usability. Thus, we derive three hypotheses for usability. Note that we refrain from forming hypotheses that address specific keyboards at this point due to the explorative nature of our study. We will provide specific post-hoc analysis in Section 5.

H_1 Significant differences exist in entry time between virtual keyboards.

H_2 Significant differences exist in mean error rate between virtual keyboards.

H_3 Significant differences exist in the perceived usability between virtual keyboards.

We developed a web application to measure entry time and typing accuracy of password entry. Key strokes entered in a

[1]We excluded curly brackets and the grave accent because they are not available on all virtual keyboards.

Table 1: Typing effort and occurrence probability of special characters

Character Category	Character	Probab. in category	Android-Vanilla	Android-Sense	Android-Swype	iOS	MeeGo	Symbian-QWERTZ	Symbian-T9	Windows Phone 7	ø
Tier 1	SPACE	0.27	1	1	1	1	1	1	1	1	1
Tier 1	.	0.61	1	1	1	2	1	1	1	1	1.125
Tier 1	,	0.12	1	1	1	2	1	1	2	1	1.25
Tier 2	?	0.06	2	1.5	1.5	2	2	1	2	2	1.75
Tier 2	@	0.19	2	1.5	1.5	2	2	2	2	2	1.875
Tier 2	!	0.14	2	1.5	1.5	2	2	2	2	2	1.875
Tier 2	-	0.11	2	1.5	1.5	2	2	2	2	2	1.875
Tier 2	/	0.06	2	1.5	1.5	2	2	2	2	2	1.875
Tier 2	)	0.02	2	1.5	1.5	2	2	2	2	2	1.875
Tier 2	(	0.01	2	1.5	1.5	2	2	2	2	2	1.875
Tier 2	;	0.01	2	1.5	1.5	2	2	2	2	2	1.875
Tier 2	&	0.04	2	1.5	1.5	2	2	3	2	2	2
Tier 2	#	0.02	2	1.5	1.5	3	2	2	3	2	2.125
Tier 2	*	0.21	2	1.5	1.5	3	2	2	2	3	2.125
Tier 2	+	0.04	2	1.5	1.5	3	2	2	2	3	2.125
Tier 2	-	0.09	3	1.5	1.5	3	2	2	2	3	2.25
Tier 3	%	0.13	2	1.5	2.5	3	3	3	2	2	2.375
Tier 3	$	0.13	2	1.5	1.5	3	3	3	2	3	2.375
Tier 3	=	0.13	3	3	1.5	3	2	2	2	3	2.438
Tier 3	€	0.13	3	3	2.5	2	3	3	2	2	2.563
Tier 3	\	0.13	3	3	2.5	3	2	2	3	2	2.563
Tier 4	~	0.13	3	3	2.5	3	2	3	3	3	2.813
Tier 4	[	0.15	3	3	2.5	3	3	3	2	3	2.813
Tier 4	]	0.15	3	3	2.5	3	3	3	2	3	2.813
Tier 4	<	0.12	3	3	2.5	3	3	3	2	3	2.813
Tier 4	>	0.09	3	3	2.5	3	3	3	2	3	2.813
Tier 4	^	0.24	3	3	2.5	3	3	3	3	3	2.938
Tier 4	|	0.12	3	3	2.5	3	3	4	3	3	3.063

text field on the smartphone were logged with timestamps and evaluated to determine the actual entry time between the first and last characters (in ms). Each participant had to enter five passwords. Passwords were individually generated according to fixed patterns with increasing complexity based on the character categories defined in Section 3.1:

z00Z2z3 A relatively easy password that requires switching to special characters pages and back to type numbers and tier 2 and 3 special characters.

Z3z00z4Z This password requires switching to the second special characters page and back and includes two uppercase characters.

z4z302Z0z Nine characters from nearly all categories with adjacent characters from different categories.

0z4Z20zZ2 Variation of the previous pattern starting with a number.

22z1z34Zz Easy special character between two lowercase letters.

Before the first password, it was explained that passwords are case-sensitive and may contain special characters, that both typing speed and accuracy will be measured, and that participants should correct typing errors. The generated password was displayed on screen in the system's default monospace font with a password entry field below. After password entry the participant had to click *send* to continue with the next password. No feedback about typing accuracy was provided to prevent participants form going back and trying to re-enter a password.

After completion of the password entry task, participants were asked to complete the post-study system usability questionnaire (PSSUQ) [18] on a desktop computer to assess perceived usability. The PSSUQ consists of 19 items to be rated on 7-point Likert scales with an additional no answer option and the ability to provide a comment for each item. The items can be combined to four scales that exhibit high reliability [18, 19]: *system usefulness*, *information quality*, *interface quality*, and *overall satisfaction*. For our study, we translated the PSSUQ to German and slightly adapted items to disambiguate terms without changing their meaning. We changed "system" to "keyboard", for example.

4.2 Shoulder Surfing Experiment

The usability experiment was followed by the shoulder surfing experiment. We assessed a keyboard's susceptibility to shoulder surfing with the success rate of the shoulder surfer, i.e., how well the typed password could be reproduced. The success rate of an observer may further depend on the chosen shoulder surfing strategy, i.e., where the observer focused her attention when trying to read the typed password. We further assumed that the perception of shoulder surfing susceptibility varies between virtual keyboards, regardless of their actual susceptibility. Therefore, we derive three hypotheses:

H_4 Significant differences exist in mean shoulder surfing success between virtual keyboards.

H_5 Significant correlations exist between shoulder surfing success and chosen shoulder surfing strategy.

H_6 Significant differences exist in the perception of shoulder surfing susceptibility between virtual keyboards.

We followed the common approach of having participants act as shoulder surfers [5, 12, 20, 25, 27]. The experimenter played the victim and entered previously trained passwords of varying difficulty, while taking care to maintain a constant typing speed for all participants that reflected typing the user's own passwords. A reverse setup with the participant as victim would suffer from varying typing speed, uncontrollable attempts for obstructing shoulder surfing, and training effects for the shoulder surfer. In our setup, the victim was sitting at a table typing passwords with the thumb of his right hand, which rested on the table to reduce vibrations and arm movements. Participants could choose to stand in the middle behind the victim or behind the left or

right shoulder. Those positions were marked on the floor directly behind the victim's chair. The following passwords were used in the study:

sunshine A frequent dictionary password from the LulzSec leak.

uMo.37 x Both special characters are easy to input on all systems.

g<G,9o-l A hard password, designed to confuse the shoulder surfer. Comma, g, 9, and o are easily confused.

Before the experiment commenced, the danger of shoulder surfing in public places was explained. Participants were asked to estimate shoulder surfing susceptibility of the assigned virtual keyboard by rating four items on 7-point Likert scales. Participants were given a paper sheet to take notes. After entering a password the experimenter waited until the participant was finished taking notes before typing the next password. Afterwards, the participant tried to enter the passwords on the device, with maximum three attempts per password. We rejected Nicholson's approach [20] of having participants complete memory rotation tasks before entering passwords to avoid noise due to differences in memory capabilities between participants. We automatically logged key strokes of each password entry attempt by the participant to measure the shoulder surfing success rate.

We quantify the error rate with the Levenshtein distance d_L [17], which is defined as the number of delete, insert, and substitute operations required to transform the recorded password into the correct password. Thus, the Levenshtein distance better accounts for shifted characters (1 substitution instead of 2 wrong characters) [9].

The experiment closed with a short questionnaire asking for a re-assessment of the perceived shoulder surfing susceptibility. In addition, participants were asked to rate which shoulder surfing strategy they followed by rating their focus on the virtual keyboard, the finger movement, and the entry field. Additional comments could be provided in a text field.

5. RESULTS

5.1 Usability Results

In order to filter out incomplete samples, we only used those samples that were sent off correctly, i.e., where the whole password was typed before pressing "send".

5.1.1 Password entry time (H_1)

Figure 2 shows the *median entry time* of each group for the five different password categories. Using the non-parametric Kruskal-Wallis test, we find a significant difference in entry time between groups ($H(7)$=74.40, p<.01). Thus, we can reject the null hypothesis and accept hypothesis H_1: *Significant differences exist in entry time between different virtual keyboards.* Variances in the different groups are equal according to Levene's test (p=.09). A Bonferroni post-hoc test shows that *entry time* is significantly higher for Symbian-QWERTZ compared to Android-Sense (p=.02), Android-Vanilla, iOS, MeeGo, and Windows Phone (p<.01). Thus,

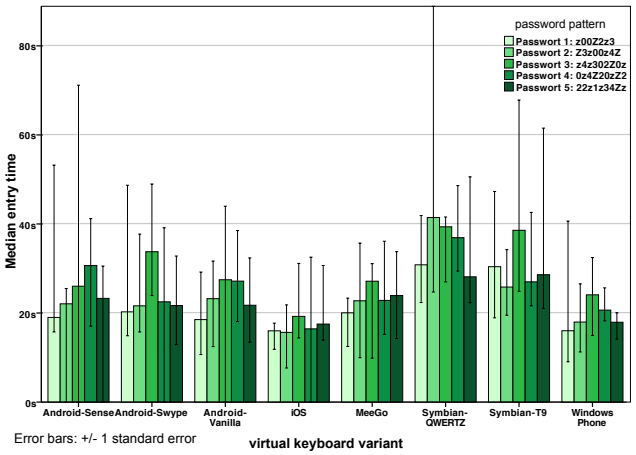


Figure 2: Median password entry time per group.

of all tested virtual keyboards with full layout, Symbian-QWERTZ is the least efficient in terms of password entry time. Also for Symbian-T9, *entry time* is significantly higher compared to iOS (p<.01) and Windows Phone (p=.01). Similarly, *entry time* is significantly higher for Android-Swype compared to iOS (p=.01) and Windows Phone (p=.03). Thus, the virtual QWERTZ keyboards of iOS and Windows Phone facilitate faster password entry than the non-QWERTZ variants T9 and Swype.

5.1.2 Error rate (H_2)

We counted mistyped characters and characters corrected before sending as errors. The data sets for Android-Sense and Symbian-T9 required rectification because our measurement system counted errors on those systems when accessing alternate keys or switching through characters. The data sets were manually corrected and are used in the following. Figure 3 shows the *mean error rate* per password and group. According to the Kruskal-Wallis test, significant differences exist between groups ($H(7)$=26.85, p<.01). Thus, we can accept hypothesis H_2: *Significant differences exist in mean error rate between different virtual keyboards.* Levene's test shows no variance homogeneity (p<.01). Therefore, we use the Games-Howell post-hoc test. The Games-Howell test shows that the *mean error rate* on Windows Phone is significantly lower than on Android-Vanilla (p=.03), Android-Swype (p=.05), and Symbian-QWERTZ (p<.01). The strong difference between Windows Phone and those other variants is also apparent in Figure 3. Table 2 shows the combined mean error rate over all passwords for each keyboard variant. On Symbian-QWERTZ at least one error occurred on average per entered password. Windows Phone, iOS, and Symbian-T9 provided the best typing accuracy, but only the Windows Phone results are statistically significant.

5.1.3 Qualitative Assessment (H_3)

Figure 4 shows the results of the four PSSUQ scales. The variables *SYSUSE*, *INFOQUAL*, and *INTERQUAL* are not normally distributed (Shapiro-Wilk test). The Kruskal-Wallis test shows significant differences for *SYSUSE* ($H(7)$=18.83, p<.01), but they are not strong enough to show in post-hoc analysis. Significant results also exist for *INTERQUAL* (Kruskal-Wallis: $H(7)$=19.63, p=.01). Specifically, the in-

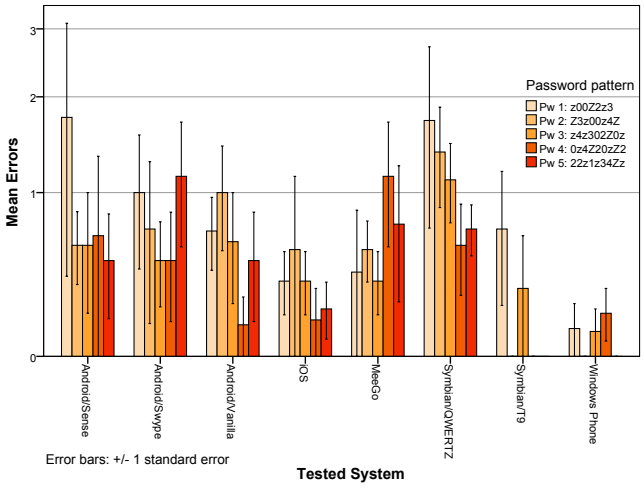


Figure 3: Mean error rate per group.

Table 2: Mean error rate over all passwords

Keyboard variant	mean error	std. dev.
Symbian/QWERTZ	1.07	1.42
Android/Sense	.90	2.09
Android/Swype	.78	1.27
MeeGo	.65	1.06
Android/Vanilla	.61	.89
iOS	.34	.75
Symbian/T9	.30	.80
Windows Phone	.09	.29

terface quality of Android-Sense was rated significantly better than that of Symbian-QWERTZ (Bonferroni: p=.01). *INFOQUAL* exhibits no significant differences, likely, because displayed information is similar across keyboards.

The *OVERALL* variable is normally distributed (Shapiro-Wilk test) and a one-way ANOVA shows significant differences ($F(7,72)$=3.50, p<.01). Android-Sense (p=.02) and Windows Phone (p=.04) are both rated significantly higher than Symbian-QWERTZ. Therefore, we accept hypothesis H_3: *Significant differences exist in the perceived usability of different virtual keyboards.* This confirms the quantitative results of H_1 and H_2. Symbian-QWERTZ provides inferior usability compared to other full keyboard variants. In contrast, Windows Phone provides high usability. iOS exhibits good quantitative usability results, but no significant differences exist in perceived usability.

5.2 Shoulder Surfing Results

5.2.1 Success rate (H_4)

A low Levenshtein distance d_L indicates a guess closer to the original password and higher success in recognizing the password. For each participant, we selected the best of their three attempts to enter the observed password (minimal d_L). Figure 5 shows the mean minimal Levenshtein distance for the three passwords per group. The Levenshtein distance was not normally distributed for all groups (Shapiro-Wilk). The non-parametric Kruskal-Wallis test shows significant differences between groups ($H(7)$=14.98, p=.04). Therefore, we accept hypothesis H_4: *Significant differences exist in mean shoulder surfing success between virtual keyboards.* Due to lack of variance homogeneity (Levene's test, p=.03), we employ the Games-Howell post-hoc test. For Symbian-T9, d_L was significantly higher than for Android-Sense (p=.03) and Android-Swype (p=.03). Thus, Symbian-T9 is less susceptible to shoulder surfing than the two Android variants.

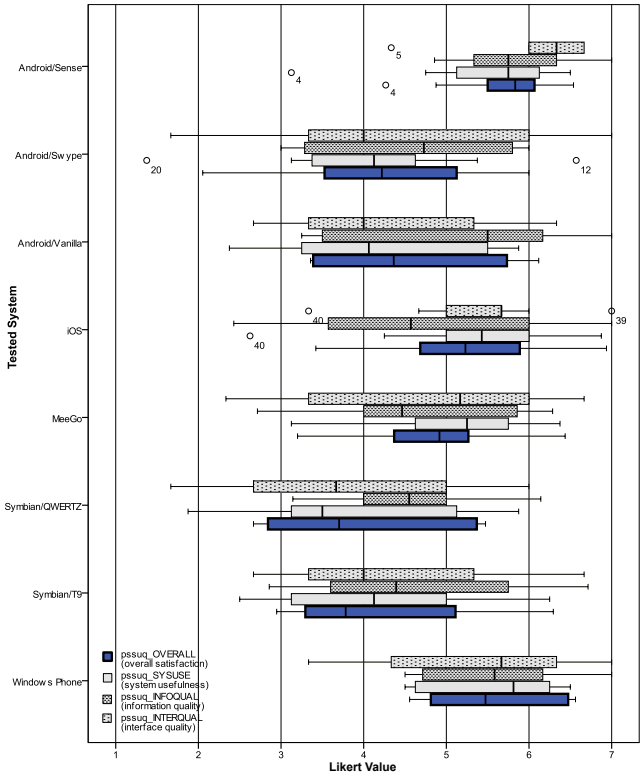


Figure 4: PSSUQ results after usability experiment.

Table 3: Number of correct guesses

Keyboard variant	total %	pw1	pw2	pw3
Android/Sense	23.3%	5	2	0
Android/Swype	20.0%	4	1	1
Android/Vanilla	20.0%	3	1	2
iOS	20.0%	3	1	2
Windows Phone	16.7%	3	1	1
MeeGo	13.3%	4	0	0
Symbian/QWERTZ	13.3%	2	1	1
Symbian/T9	6.6%	2	0	0

While the mean Levenshtein distance indicates the closeness between guesses and correct passwords, the number of completely correct guesses is also of interest. Table 3 ranks systems according to the total percentage of correctly guessed passwords. Not surprisingly, the dictionary password (pw1) was recognized most frequently across systems. The two more complex passwords show similar numbers of correct guesses. The most passwords were correctly guessed on Android and iOS variants (>20%), while Symbian-T9 was the hardest to shoulder surf with only 6.6% correct password guesses.

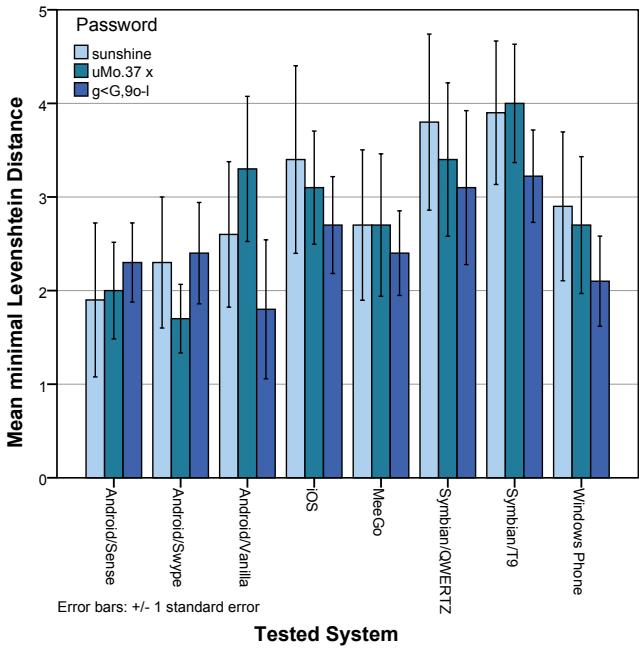


Figure 5: Levenshtein distance of guessed passwords.

5.2.2 Recognition rate per character

The success rate analysis led us to also analyze the *recognition rate* of individual characters in the different passwords across groups to determine which characters and character categories are easier to observe. Figure 6 shows the total *recognition rate* for the individual characters of the three passwords. In general, characters at the beginning of a password are recognized well, followed by a slight decrease in recognition rate over the password length. The last character is again frequently recognized, likely because it is displayed longer, while other characters are hidden when the next character is typed.

Password 1. Fig. 6(a) shows a decreased recognition rate for the second and fifth characters. A potential explanation could be that the victim has to move the right thumb to the far left of the keyboard to type "*s*". Thereby, a large part of the keyboard is obscured for the observer, while the victim can easily type following characters when moving the thumb back to the right.

Password 2. Fig. 6(b) shows a continuous decrease in recognition rate for the first four characters. The two special characters (*tier 1*) exhibit the lowest recognition rate, Especially *space* was rarely recognized, probably because it can be typed quickly and would be hard to observe in the entry field. In contrast, numbers exhibit a high recognition rate. Most likely due to the switch to a different keyboard page or holding down a character key until the number is selected.

Password 3. Figure 6(c) shows that the delay required to type "<" (2.8 taps) results in a relatively high recognition rate. Interestingly, the recognition rate increases even further for "*G*". The *comma* has a similar recognition rate as the *dot*, in the second password. This suggests that special characters from the same category (tier 1) have similar recognition rates. An explanation for the relatively low recognition rate could be the small dimension of either character when displayed in the entry field and the resulting ambiguity, especially between these two characters. The "*o*" can be confused with a "*0*", especially on keyboards where they are close together, e.g., the Android variants. The "*l*" at the end has a very low detection rate, because it was often confused with "*I*" or the *pipe* character—although typing a pipe requires at least 2 additional taps on all devices.

5.2.3 Shoulder surfing strategies (H_5)

To analyze if focusing on particular parts of the phone correlate with higher success rate, we calculated a participant's *mean minimal Levenshtein distance* over all three passwords as the success metric, which resulted in a reliable scale (Cronbach's α=.70). Because values are not normally distributed (Shapiro-Wilk), we employ the non-parametric Spearman rank correlation. We found a significant negative correlation (ρ=.303, p=.01) with medium effect between *mean Levenshtein distance* and focus on the *entry field*. Thus, participants that focused on the entry field had a higher probability of achieving a high success rate (low d_L). Hypothesis H_5 can be accepted, but with caution because of the small effect size. The Spearman rank test also showed significant negative correlations between *entry field* and *keyboard* (ρ=.561, p<.01) and *entry field* and *finger movement* (ρ=.335, p=.01), as well as a positive correlation between *keyboard* and *finger movement* (ρ=.309, p=.01). Thus, participants focused either on keyboard and finger movement or on the entry field. This seems sensible considering the spatial distance between keyboard in the lower screen half and entry field in the upper half.

5.2.4 Perceived shoulder surfing susceptibility (H_6)

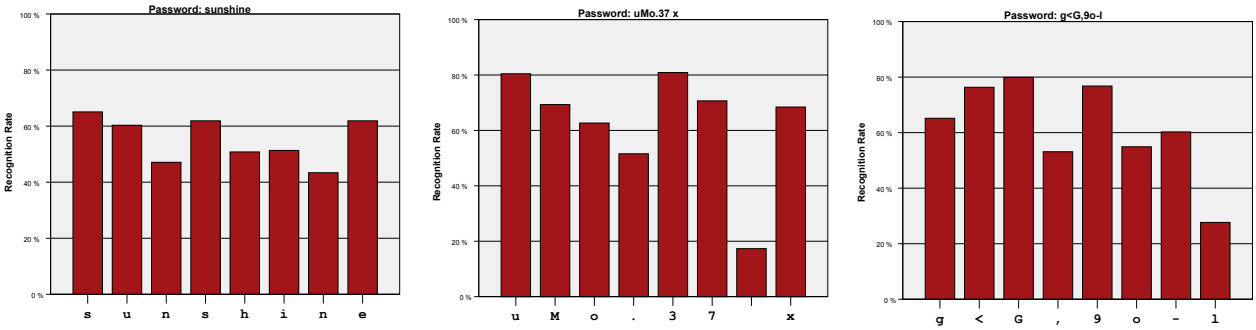
Participants were asked to rate the perceived shoulder surfing susceptibility of their assigned keyboard variant before and after the shoulder surfing experiment. However, no significant differences between groups could be found in either case. Therefore, the null hypothesis for H_6 cannot be rejected.

5.3 Limitations

The study was performed with a German keyboard layout. Specific results might look different for keyboard layouts of other languages. However, assuming a similar layout with minimal changes, most results should be transferable and reproducible.

In the usability experiment, participants had to type displayed passwords. This does not reflect the natural situation of entering a memorized password. However, we wanted to eliminate differences in memory capability, because memorability of passwords was not the focus of this study.

In the shoulder surfing experiment, some participants pointed out that they might achieve higher recognition rates if they would explicitly train shoulder surfing. However, our focus were differences in shoulder surfing susceptibility between different keyboard variants and all participants had the same level of shoulder surfing experience. Training participants in



(a) first password (b) second password (c) third password

Figure 6: Successful recognition rate per character.

specific shoulder surfing strategies could provide an interesting follow-up study, especially to better understand correlations between shoulder surfing success and strategy.

6. DISCUSSION

The Windows Phone and iOS virtual keyboards faired best in the usability experiment with short password entry times and high typing accuracy. Those two variants, as well as MeeGo, provide a clean keyboard interface without alternate functions. Symbian-QWERTZ faired considerably worse. An issue pointed out by multiple participants was the small button size and the cramped keyboard layout, which likely explains the high rate of typing errors. The Symbian-T9 variant, on the other hand, had a relatively low error rate. Yet, typing passwords with T9 takes considerably longer because multiple taps are required for many characters, which are directly accessible on the other keyboards. The word completion typically available for T9 typing is also deactivated in password entry fields. The Android variants and MeeGo rank in the mid-field of the usability results, without significant differences between them.

Interestingly, the keyboard variants with low usability turned out to be more resistant against shoulder surfing. Especially Symbian-T9 fairs very well. A likely explanation is that switching through characters makes it difficult for an observe to determine at which character the user stopped. A reason for the relatively high shoulder surfing resistance of Symbian-QWERTZ could be the small button size, which is also a likely cause for the above mentioned usability issues. The three Android variants were most susceptible to shoulder surfing. In the post-shoulder surfing questionnaire, participants named the magnification of pressed buttons as a main issue for Android. This is surprising considering that the other keyboards also magnify or highlight pressed keys. Windows Phone and iOS show that there is no general apparent trade-off between shoulder surfing risk and usability. Both keyboard variants were leading in the usability experiment and also performed on an acceptable level in the shoulder surfing experiment. In our shoulder surfing experiment, we could not reproduce Tari et al.'s result that non-dictionary passwords are easier to observe than dictionary ones [25]; our results suggest the opposite.

Based on our results, we can derive certain insights that can serve as pointers for further investigation, support improvement of virtual keyboard design, and help users with password composition for mobile use.

6.1 Keyboard Interface Insights

- Alternate functions on keys of the keyboard's primary screen may not improve usability and likely increase shoulder surfing susceptibility.
- Magnification of typed characters increases shoulder surfing susceptibility. Magnification should be disabled for password fields accordingly.
- Reduced keyboards (e.g., T9) can be accurate and improve security, but reduce typing efficiency.

6.2 Mobile Password Composition Insights

- Special characters that can be easily typed (e.g., *dot*, *comma*, and *space*) increase shoulder surfing resistance.
- Ambiguous characters increase shoulder surfing resistance, e.g., characters that can be entered with the same button (e.g., *space* and *dot* on iOS) or look similar (e.g., "*0*" and "*o*").
- Character recognition rates decrease over the password length. Taken together with the results by Komanduri et al. [13] that longer passwords are more usable than complex ones, longer passwords with simpler characters are preferable for smartphone entry.

7. CONCLUSIONS

Virtual keyboards on smartphones seem quite similar at first glance. Therefore, one would intuitively assume that they offer similar usability and resistance against shoulder surfing. But by transferring the layout of a physical keyboard onto a small-scale touchscreen the effects of small user interface and experience differences can be elevated to significant differences in terms of usability and shoulder surfing susceptibility. A primary observation is that special characters—still commonly required by password composition rules—differ in typing effort on virtual keyboards with full QWERTZ/QWERTY layout. We categorized special characters according to their typing effort (measured in finger taps) to generate passwords with different complexity.

Our between subjects study with 80 participants evaluated the usability and shoulder surfing susceptibility of eight virtual keyboard variants covering the mobile platforms iOS,

Android, Windows Phone, Symbian, and MeeGo. Our results show significant differences in the usability between different keyboard variants for password entry, as evidenced by significant differences in quantitative (entry time, error rate) and qualitative (perceived usability) evaluation.

Our study also showed significant differences in shoulder surfing success for different virtual keyboards. Interestingly, keyboard variants with low usability proved more resistant against shoulder surfing. We further studied the recognition rates of individual characters for different passwords, leading to initial insights on the influence of password composition on shoulder surfing resistance. We also identified focusing on the password entry field as the most successful shoulder surfing strategy. Thus, attention for improving shoulder surfing resistance on smartphones should also be focused on the entry field rather than the virtual keyboard alone. Virtual keyboards provide a large design space with novel features compared to physical keyboards, such as enabling context-based changes of key labels (e.g., pressing shift can change characters to uppercase), pop-out menus to provide quicker access to special characters, and magnification of typed characters. While this design space offers opportunities for usability enhancement, the employed features must be balanced with security considerations and the common usage context of the device. While further studies are required to better understand the effects of specific user interface features on shoulder surfing susceptibility, our work shows that seemingly small design variations can result in significant differences in terms of security.

8. REFERENCES

[1] R. Biddle. Memorability of Persuasive Passwords. In *CHI '08 extended abstracts*. ACM, 2008.

[2] R. Biddle, M. Mannan, P. C. van Oorschot, and T. Whalen. User Study, Analysis, and Usable Security of Passwords Based on Digital Objects. *IEEE Trans. Info. Forensics and Security*, 6(3):970–979, 2011.

[3] J. Bonneau, C. Herley, P. C. V. Oorschot, and F. Stajano. The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes. In *Symp. on Security and Privacy*. IEEE, 2012.

[4] J. Bonneau and S. Preibusch. The password thicket: technical and market failures in human authentication on the web. In *WEIS'10*, 2010.

[5] P. Dunphy, A. P. Heiner, and N. Asokan. A closer look at recognition-based graphical passwords on mobile devices. In *SOUPS '10*. ACM, 2010.

[6] D. Florencio and C. Herley. A large-scale study of web password habits. In *WWW'07*. ACM, 2007.

[7] A. Forget, S. Chiasson, P. C. van Oorschot, and R. Biddle. Improving text passwords through persuasion. In *SOUPS'08*. ACM, 2008.

[8] P. G. Inglesant and M. A. Sasse. The true cost of unusable password policies. In *CHI '10*. ACM, 2010.

[9] I. Jermyn, A. Mayer, F. Monrose, M. K. Reiter, and A. D. Rubin. The design and analysis of graphical passwords. In *USENIX Security Symp.*, 1999.

[10] S. Jeyaraman and U. Topkara. Have the cake and eat it too - Infusing usability into text-password based authentication systems. In *21st Annual Computer Security Applications Conf. (ACSAC'05)*. IEEE, 2005.

[11] M. Keith, B. Shao, and P. Steinbart. The usability of passphrases for authentication: An empirical field study. *Int. J. Hum.-Comp. Studies*, 65(1), 2007.

[12] D. Kim, P. Dunphy, P. Briggs, J. Hook, J. Nicholson, J. Nicholson, and P. Olivier. Multi-touch authentication on tabletops. In *CHI '10*. ACM, 2010.

[13] S. Komanduri, R. Shay, P. G. Kelley, M. L. Mazurek, L. Bauer, N. Christin, L. F. Cranor, and S. Egelman. Of passwords and people: measuring the effect of password-composition policies. In *CHI '11*. ACM, 2011.

[14] M. Kumar, T. Garfinkel, D. Boneh, and T. Winograd. Reducing Shoulder-surfing by Using Gaze-based Password Entry. In *SOUPS'07*. ACM, 2007.

[15] C. Kuo, S. Romanosky, and L. F. Cranor. Human selection of mnemonic phrase-based passwords. In *SOUPS '06*. ACM, 2006.

[16] S. C. Lee and S. Zhai. The Performance of Touch Screen Soft Buttons. In *CHI '09*. ACM Press, 2009.

[17] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8):707–710, 1966.

[18] J. R. Lewis. IBM Computer Usability Satisfaction Questionnaires: Psychometric Evaluation and Instructions for Use. *Intl. J. Hum.-Comp. Int.*, 7(1), 1995.

[19] J. R. Lewis. Psychometric evaluation of the PSSUQ using data from five years of usability studies. *Intl. J. Hum.-Comp. Int.*, 14(3), 2002.

[20] J. Nicholson. *Design of a Multi-Touch Shoulder Surfing Resilient Graphical Password*. Dissertation, Newcastle University, 2009.

[21] Y. S. Park, S. H. Han, J. Park, and Y. Cho. Touch Key Design for Target Selection on a Mobile Phone. In *MobileHCI '08*. ACM, 2008.

[22] V. Roth, K. Richter, and R. Freidinger. A PIN-entry method resilient against shoulder surfing. In *CCS'04*. ACM, 2004.

[23] H. Sasamoto, N. Christin, and E. Hayashi. Undercover: authentication usable in front of prying eyes. In *CHI '08*. ACM, 2008.

[24] D. S. Tan, P. Keyani, and M. Czerwinski. Spy-resistant keyboard: more secure password entry on public touch screen displays. In *OZCHI '05*, 2005.

[25] F. Tari, A. A. Ozok, and S. H. Holden. A comparison of perceived and real shoulder-surfing risks between alphanumeric and graphical passwords. In *SOUPS'06*. ACM, 2006.

[26] J. Yan, A. Blackwell, R. Anderson, and A. Grant. Password memorability and security: Empirical results. *Security & Privacy, IEEE*, 2(5):25–31, 2004.

[27] N. H. Zakaria, D. Griffiths, S. Brostoff, and J. Yan. Shoulder surfing defence for recall-based graphical passwords. In *SOUPS'11*. ACM, 2011.

[28] H. Zhao and X. Li. S3PAS: A Scalable Shoulder-Surfing Resistant Textual-Graphical Password Authentication Scheme. In *AINAW'07 Workshops*. IEEE, 2007.