Visualizations and Switching Mechanisms for Security Zones

Peter Riedl [1], Phillip Koller [1], Rene Mayrhofer [1], Andreas Möller [2], Marion Koelle [3], Matthias Kranz [3]
[1] University of Applied Sciences Upper Austria, 4232 Hagenberg, Austria
[2] Technische Universität München, Arcisstraße 21, 80333 München, Germany
[3] Universität Passau, Institute for Embedded Systems, Innstraße 43, 94032 Passau, Germany
peter.riedl@fh-hagenberg.at, phillip.koller@students.fh-hagenberg.at, rene.mayrhofer@fh-hagenberg.at, andreas.moeller@tum.de, marion.koelle@uni-passau.de, matthias.kranz@uni-passau.de

ABSTRACT

The ongoing evolution of mobile phones to "pocket computers" generated a demand for more and more applications to be ported to the mobile phone. Because a full security assessment for a whole mobile operating system would be prohibitively costly, currently security critical applications can not be implemented. We address this challenge by introducing security zones to enable applications with high security demands like driving licenses, health insurance cards, or passports on mobile phones. This zone concept creates the need for visualization of the current zone and a way to switch between zones. In this paper we discuss several possible ways of achieving this.

Categories and Subject Descriptors

H.5.2 [**User Interfaces**]: Graphical user interfaces, Interaction Styles.; D.4.6 [**Security and Protection**]: Access Controls, Informationflow Controls.

Keywords

Usability, Security Zones, Mobile Devices

1. INTRODUCTION

The history of mobile phones has shown that there is a demand for convergence of appliances. When they were initially developed, mobile phones fulfilled the task of establishing voice communication without the need for a wired connection. As technology advanced, more and more resources became available on mobile phones and a plethora of functionality was merged onto them. Examples are text messaging, cameras, global positioning system (GPS), wireless network connectivity and many more. Besides hardware capabilities, also software on mobile phones evolved to make them "pocket computers".

We perpetuate this development by bringing security critical applications (SCA) to the device. Examples for SCA could be the contents of a wallet (e.g. money or travel tickets), a key chain, authentication at public terminals [5], personal identification documents like the passport or driving license. Compared to nowadays common mobile phone software like games, email clients or a navigation application, SCA have much stricter requirements on security and privacy.

Despite the lack of non-exploitable operating systems, the desire for SCA (e.g. mobile banking, mobile payment, access to company network, virtual identity documents, etc.) is growing steadily. Another emerging problem is the wish or need of employees to use their own device privately and for business. This brings up security and maintenance problems for system administrators. The commonly used term for this problem is "Bring Your Own Device Problem" (BYOD problem). The approach we take to address these demands is the introduction of security zones as described in the next section.

2. SECURITY ZONES

Enabling SCA on mobile devices and solving the BYOD problem go hand in hand with restrictions on what an installed application is allowed to do. These restrictions could prevent installing applications on the device, disabling parts of the hardware (e.g. bluetooth, camera), permission limitations for applications, or only allowing certain network connections such as a virtual private network (VPN) to the company network. All of them would influence user experience negatively. Our approach to address the trade-off between security and usability is the introduction of security zones. The reasoning behind this is simple: to account for the inherent trade-off between security and usability in one device, we divide one physical device into multiple logical devices for dedicated purposes. One such logical device is called a security zone. Examples for different zones could be: An *Open* zone without any restrictions (current situation), a *Secure* zone for security critical operations in the private

MoMM 2013, Vienna, Austria

domain (e.g. mobile banking), and a *Managed* zone which is fully controlled by an organization system administrators. The compartmentalization of one physical device into those zones provides the benefit of giving users all the freedom they are used to, while still providing enhanced security in the other zones. This brings up the challenge of visualizing the currently active zone and to enable switching between different security zones.

One very important aspect besides the used security technologies is user interaction [7]. We are fully aware that the most solid secure implementation can not be utilized without proper user notification and a way to easily apply the proposed security zone system for the average user. In order to find out how to bridge the gap between security and simple usage, we are currently conducting a user study. Other aspects of the usability security trade off are discussed in [4] and [6].

Color Scheme.

According to [1] in the Western culture colors are associated with different meanings. The connotations are as follows:

- Red: Danger, hot, fire.
- Yellow: Caution, slow, test.
- Green: Go, okay, clear, vegetation, safety.
- Blue: Cold, water, calm, sky.
- Warm colors: Action, response required, proximity.
- Cool colors: Status, background information, distance.

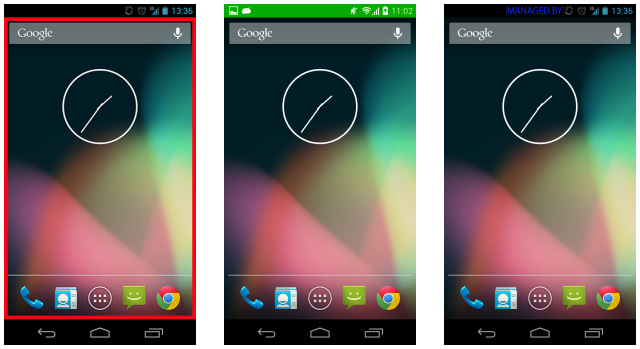
Based on this classification, we chose the color red for the *Open* zone. This should represent the "danger" or "insecurity" of working in an unrestricted and hence not secured zone. In our system the color green stands for the *Secure* zone. For the third proposed zone – *Managed* – there is no easy direct association with colors. For this zone we chose the color blue because we wanted to use one of the primary colors and avoid any confusion of our system with the commonly known principle of traffic lights (red, green, yellow). In addition, blue seems to be a good candidate for the *Managed* zone because blue is a cool color that can be associated with "background" or "distance" which could be interpreted as remotely managed.

3. VISUALIZATIONS AND SWITCHING

All software visualizations of security zones have a common weakness – full screen applications. Whenever the whole screen is occupied by an application (e.g. watching video/photos or remote desktop connection) any software based visualization is either obscured by the full screen application or the available screen space for the application has to be reduced. Our solution to this weakness is the hardware prototype discussed in Section 3.7.

3.1 Colored Border Visualization (CBV)

The colored border visualization uses the color scheme described in Section 2 to visualize the currently active zone. This is done by drawing a colored border around the home screen and any non-full-screen application (see Figure 1a).



(a) (b) (c)

Figure 1: Colored border visualization at the example of the *Open* zone (a), Colored notification bar visualization at the example of the *Secure* zone (b), Colored text visualization at the example of the *Managed* zone (c).

This way of visualization is very simple and does not interfere with any standard theme provided by Android. However, background images using similar colors as in our color scheme could lead to confusions whether the colored border is part of the background image or the CBV.

3.2 Colored Notification Bar Visualization (CNV)

The colored notification bar visualization (see Figure 1b) uses the Android notification bar[1] in conjunction with the color scheme presented in Section 2 to indicate the current zone.

The advantage of this visualization compared to CBV is that no space for the home screen or an application is "wasted". The problem with CNV is that custom themes that change the color of the notification bar would have to be disabled.

3.3 Colored Text Visualization (CTV)

The colored text visualization – just like CNV – makes use of the Android notification bar. Instead of coloring the whole notification bar, the name of the currently active zone is displayed in its respective color according to our color scheme (see Figure 1c).

Instead of solely relying on color, the CTV uses text for security zone visualization. Compared to CBV and CNV, this approach has benefits regarding color blind people, because the information about the currently active zone is not just color coded. A clear disadvantage of CTV is that it uses more or less space on the notification bar depending on the length of the zone name. Another potential issue is, that customized themes could make coloration of the text according to our color scheme impractical. In that case, the color of the text could not be used as an additional cue.

3.4 Swipe Switching Mechanism (SSM)

Nowadays swipe gestures are used quite frequently on mobile phones. This input method resembles turning a page of a book and therefore it implies the notion of changing what is currently presented to the user. Commonly, swiping is used for tasks like scrolling on a screen or browsing through pho-

[1] `http://developer.android.com/guide/topics/ui/notifiers/notifications.html`

tos. We propose the use of swiping to cycle through security zones with the swipe switching mechanism (see figure 2).

Figure 2: The Swipe switching mechanism allows to switch between security zones in a circular manner. This is done with a horizontal (left or right) three finger swipe.

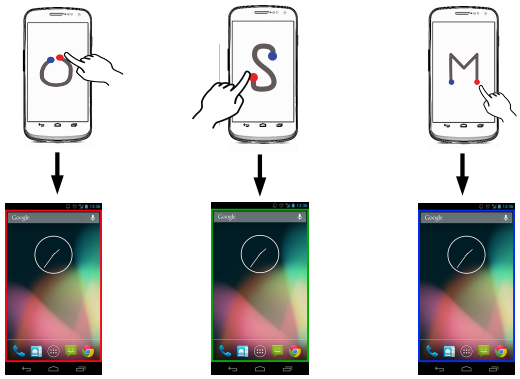


Figure 3: The Gesture switching mechanism allows to switch between security zones by drawing the first letter of the zone's name (*O* for *Open*, *S* for *Secure*, *M* for *Managed*).

The principle behind SSM is to enable the user to switch between security zones with the simple swipe of three fingers. The reason for demanding three fingers instead of a single finger or a two finger swipe is that to reduce the risk of interfering with other applications. For this switching mechanism one can imagine the security zones arranged in a circle. In order to switch between zones, the user first has to go to the home screen and then perform a three finger swipe either to the left or to the right. The direction of the swipe correlates with the direction the imaginary circle is rotated to. A typical switching scenario using SSW could look as follows:

1. Bring up home screen in the *Open* zone.
2. Perform three finger swipe to the right.
3. Now the *Secure* zone is active.
4. Perform security critical task.
5. Bring up home screen in the *Secure* zone.
6. Perform three finger swipe to the right.
7. Now the *Managed* zone is active.

One advantage of this switching mechanism is its simplicity. There is little cognitive load associated with the mechanism regardless of how many security zones are available. Another advantage is that swiping already is a common interaction technique, so learning this mechanism should be fairly quick. A potential disadvantage could be unintentional zone changes. It is imaginable that a swipe gesture is executed accidentally and an unwanted switch is triggered.

3.5 Gesture Switching Mechanism (GSM)

The gesture switching mechanism is different to the SSM in many ways. Opposed to SSM, the GSM does not cycle through security zones, it selects one zone explicitly. This is done using gestures (see Figure 3). Some aspects of using gestures for interaction are discussed in [3].

The gestures associated with each security zone are the first letters of the zone name from the one stroke alphabet called Graffiti [2]. This alphabet was chosen because it fulfills the requirement of being able to be drawn in a single stroke and the letters resemble the commonly known roman letters to a large degree. This reduces the effort to learn and memorize the respective gestures. Opposed to SSM, for using GSM the user needs to learn a dedicated gesture for every available zone. This impedes the scalability of GSM.

3.6 Lock Screen Switching Mechanism (LSM)

The lock screen switching mechanism makes use of the Android lock screen to provide a way of switching between security zones. Figure 4 depicts the principle of the LSM.

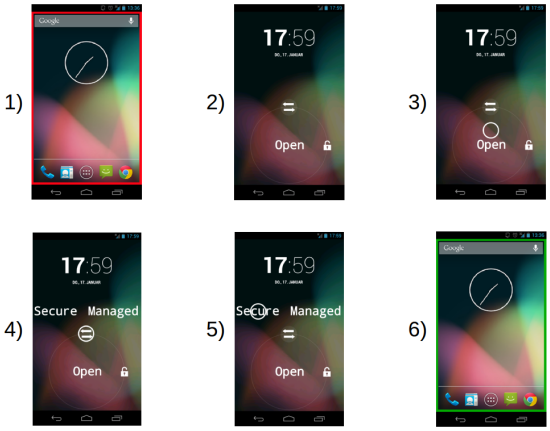


Figure 4: The Lock screen switching mechanism allows to switch between security zones via the device lock screen. This is shown at the example of switching from the *Open* zone to the *Secure* zone.

1. *Open* zone is active.
2. Open lock screen.
3. Drag from the center of the circle to the switch symbol.
4. All available zones appear.

5. Drag further to the desired zone.

6. *Secure* zone is active.

This approach to switching between security zones does not require the user to remember anything – like SSM. The initial effort to learn how to use LSM is minimal. Scalability is no issue with this mechanism because assuming many (e.g. more than ten) zones are available on the mobile phone, these could be presented in a scrollable list on the lock screen rather than next to each other. The fact that Android multi user functionality[2] is controllable via the lock screen indicates that security features on the lock screen could gain popularity. The LSM could be seen as an enhancement to this functionality.

3.7 Hardware

All software based solutions for visualizing the currently active zone and switching between zones are prone to spoofing. Assuming the *Open* zone is infected with malware, it is conceivable that a malicious application masquerades as a trustworthy one by imitating the visualization of the *Secure* zone, while the device actually is in the *Open* zone. Also the touch events needed for switching could be manipulated by applications in the *Open* zone. One possible solution for this problem is using hardware. Specialized hardware for visualization and switching is much harder to interfere with than any software solution. Under the assumption that access to the hardware can be reliably denied for unauthorized software, it is even impossible to alter the visualization or the switching behavior without physical access to the device.

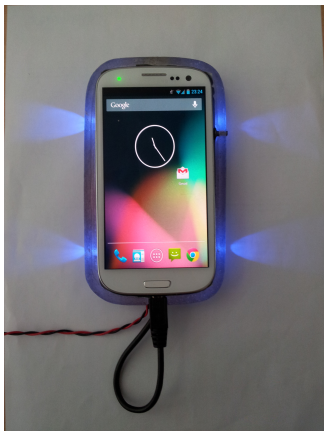


Figure 5: The custom cast resin housing including LEDs for zone visualization at the example of the *Managed* zone.

The hardware prototype for visualization and switching consists of a transparent resin housing for the mobile phone. This housing comprises four multi color light emitting diodes (LEDs) and a hardware sliding switch. The four LEDs always emit the same color to create the impression the whole housing is glowing in the desired color. The housing again adheres to the color scheme specified in Section 2.

The sliding switch for switching between zones is mounted on the top end of the phone. This decision was made to avoid accidental switching and to increase awareness by requiring an explicit change of posture to change zones.

[2] `http://developer.android.com/about/versions/android-4.2.html`

4. CONCLUSIONS

In this paper we discussed one possible way to bring SCA onto mobile phones while still providing good usability. Our approach is to introduce security zones for security critical tasks. We presented several mechanisms for visualizing and switching between zones. The goal of the associated user study is to evaluate if there is a mechanism that is preferred by a majority of users or if we have to provide different mechanisms as options for the user. The results will be published as future work.

5. ACKNOWLEDGMENTS

This work has been carried out within the scope of *u'smile*, the Josef Ressel Center for User-Friendly Secure Mobile Environments. We gratefully acknowledge funding and support by the Christian Doppler Gesellschaft, A1 Telekom Austria AG, Drei-Banken-EDV GmbH, LG Nexera Business Solutions AG, and NXP Semiconductors Austria GmbH.

6. REFERENCES

[1] D. Benyon. *Designing Interactive Systems*. Addison-Wesley, 2010.

[2] C. H. Blickenstorfer. Graffiti: wow. *Pen Computing Magazine*, 1:30–31, 1995.

[3] A. Bragdon, E. Nelson, Y. Li, and K. Hinckley. Experimental analysis of touch-screen gesture designs in mobile environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, New York, 2011. ACM.

[4] R. Dhamija. The battle against phishing: Dynamic security skins. In *In SOUPS '05*. ACM Press, 2005.

[5] L. Roalter, M. Kranz, S. Diewald, and A. Möller. The Smartphone as Mobile Authorization Proxy. In A. Quesada-Arencibia, J. C. Rodriguez, R. M.-D. jr., and R. Moreno-Diaz, editors, *14th International Conference on Computer Aided Systems Theory (EUROCAST 2013)*, pages 306–307, Feb. 2013.

[6] J. Stoll, C. S. Tashman, W. K. Edwards, and K. Spafford. Sesame: informing user security decisions with system visualization. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, New York, 2008. ACM.

[7] A. Toninelli, R. Montanari, O. Lassila, and D. Khushraj. What's on users' minds? toward a usable smart phone security model. *Pervasive Computing, IEEE*, 8(2):32–39, 2009.