Poster: Exploring Contextually Bounded Access Control

Andrew Besmer
University of North Carolina at Charlotte
9201 University City Blvd
Charlotte, NC 28223
arbesmer {at} uncc.edu

Jason Watson
University of North Carolina at Charlotte
9201 University City Blvd
Charlotte, NC 28223
jwatso8 {at} uncc.edu

Heather Richter Lipford
University of North Carolina at Charlotte
9201 University City Blvd
Charlotte, NC 28223
heather.lipford {at} uncc.edu

1. INTRODUCTION/BACKGROUND

Users are required on a regular basis to configure access control settings to use devices, surf the web, interact with their profiles, and more. In order to meet this demand users are adopting a number of different strategies including: circumventing security, delegating security decisions to others, and accepting defaults policies [2]. There are users however who are motivated to use controls to protect their privacy and security [1]. By contextually binding access control policy configuration we hope to reduce the burden on users and allow them to make better decisions for maintaining their privacy and security.

Current methods of policy configuration for applications can be grouped into two configuration times: those that are configured at install time and those configured at run time. Configuration of policies at install time have been shown to be problematic because users have little to reason with why an application might need a particular attribute[1]. As an example, consider the addition of a books recommendation application on an Android device. As the user installs the application they are shown a set of permissions that are granted as a result of its use. There is little for the user to reason with besides the name of the application and the perceived benefits of its use.

Configuration of policies at run time also has drawbacks. For example, policy configuration is rarely the primary task of the user [6] or the user may lack an understanding of whats being asked [3]. Additionally, there are issues of desensitization to policy configuration screens or notices similar to what we see in legal notices during software installation.

2. CONTEXTUAL ACCESS CONTROL

Our current work focuses on exploring contextually bound access control techniques now described. Contextually bound access control is the configuration of policies during run time by providing the ability to manipulate those policies within the context of their use. To accomplish this we note that an application is a set of inputs that results in a set of outputs. We also note that information sharing is highly related to the social norms of a given context and privacy problems will arise when information is shared beyond the social expectations of the context [5]. Users policies also change over time and are situational [4].

The inputs of a system are generally in some sort of model but can also be real time data. For example, Facebook applications may request access to certain parts of a users profile or real time feeds. An Android application similarly may request access to a set of contacts or real time microphone data. Many times applications then send output to the view or user interface. For example, it is common practice for applications to consume basic user information like a first name in order to provide a customized user experience. In this case, the decision to share the input of a users first name from a database can be bound to its container in the view. Users are able to determine from looking at the user interface that the result of an output is bound directly to their providing access to an input as well as their current decision about that access.

Applications may also be consuming data to be sent to a different model such as their own database or even a third party. In cases like these the decision about where to bind the indicator will be an important one. Developers of an application will want to ensure that the location they chose matches as closely as possible to the context in which it needs to be used. For example, if a user uses an application on their phone that has chosen to bind a bluetooth decision to a friends selection screen users may not understand its use. As a result they may chose not to allow the access since its out of context. However, the same permission bound to file transmission context would be more appropriate and make sense contextually.

It is not hard to envision a situation where multiple attributes are needed in order to accomplish a task. In cases like these, contextual binding might seem to imply that each label or button on the interface would need a icon. However, since we are binding to a controls container we aggregate settings into a single indicator for a subset of the interface. By binding the configuration indicator to the container instead of the element itself we are able to prevent having indicators for each and every control that uses user data.

Contextually binding access control decisions by aggregating may also benefit end users by providing increased awareness of sharing by reminding users of sharing in a particular context. We would hope that this increased awareness would lead to memorability of attributes they have shared with an application without requiring them to leave their primary task and visit and interface dedicated to the policy alone. Users who are motivated to make security decisions will also be able to make better risk/benefit judgments since they can better understand why they are allowing access to a particular set of attributes.

3. CURRENT WORK

We chose the Android platform to begin exploring contextually bound access control to user interface. We created two applications for the Android platform outfitted with contex-

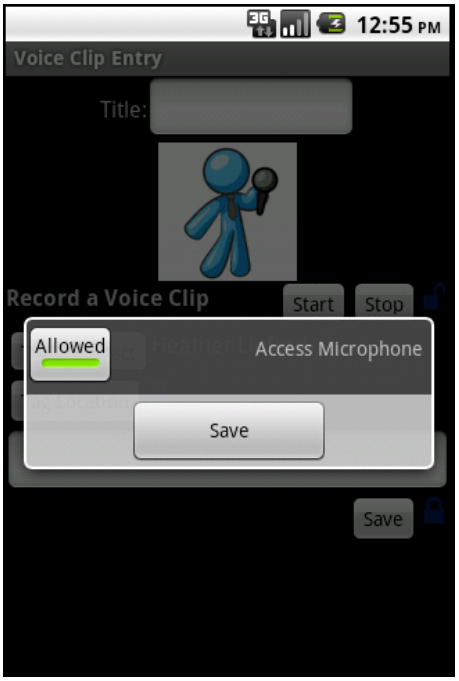

Figure 1: An example of contextual binding with the LifeJournal application.

tual access control. The first application Divide and Conquer was an Android example that we added permissions to as well as created a profile feature to track your high score as well as your friends. This application requested several permissions including access to the vibrator, contacts, Internet and phone identity. We bound each permission closest to their context of use. For example, the vibrator was bound to the game play area with a small icon in the bottom right hand corner and the Internet permission to the save profile area on the profile screen.

Our second application, called LifeJournal, allows users to take audio or picture notes and tag them with contacts, locations, notes, and more. Again we bound the permissions as close to the context of use as possible. Figure 1 shows an example of contextually binding access to the microphone to the creation of an audio note.

4. PRELIMINARY EVALUATION

Both applications were installed on our Android device and loaded with a set of fictitious contacts and pictures. We created a series of tasks asking the participants to interact with each application. As they interacted configuring access control was necessary in order to complete the task.

The study was designed to test several aspects of contextually binding access control vs standard application installs including:

- Memorability - Does providing access control in a contextual manner allow users to better remember the permissions they have configured for a program?
- Effort - How much more or less difficult/complicated is it to configure access control in this way versus the standard installation screen provided by the platform.
- Comfort - How comfortable were users in making decisions? Did they find the method of access control annoying?
- Comprehension - How well did users understand what was being asked for of them?
- Distraction - Was configuring access control in this way distracting them? All the tasks were designed to have users encounter access control which may interrupt their primary task.

We are currently analyzing the data gathered from this user study. Preliminary results seem to suggest that participants did not benefit from increased memorability. Results also seem indicate that participants did not find configuring access control contextually to be complicated and left them feeling in more control of the data on their phone. Most did not seem to find it distracting however there were several participants who felt the access control was annoying because it required interaction not associated with their primary task.

We are currently planning another study on a social network with two similar applications to see how participants in a different domain respond. We are currently designing a game application with a high score feature that consumes participants profile information and a photo journal which uses participants photographs to create a collage of an event. In order to interact with both applications it will be necessary to configure contextually bound access control.

5. ACKNOWLEDGMENTS

The authors would like to thank Ashley Anderson for her work in helping create Android prototypes with contextually bound access control used in this study.

6. REFERENCES

[1] A. Besmer, H. R. Lipford, M. Shehab, and G. Cheek. Social applications: exploring a more secure framework. In *Proceedings of the 5th Symposium on Usable Privacy and Security*, pages 1–10, Mountain View, California, 2009. ACM.

[2] P. Dourish, R. E. Grinter, and M. Joseph. Security in the wild: user strategies for managing security as an everyday, practical problem. *Personal Ubiquitous Computing*, 8:391–401, Sept. 2004.

[3] J. Goecks, W. K. Edwards, and E. D. Mynatt. Challenges in supporting end-user privacy and security management with social navigation. In *Proceedings of the 5th Symposium on Usable Privacy and Security*, pages 1–12, Mountain View, California, 2009. ACM.

[4] M. L. Mazurek, P. F. Klemperer, R. Shay, H. Takabi, L. Bauer, and L. F. Cranor. Exploring reactive access control. In *CHI 2011: Conference on Human Factors in Computing Systems*, May 2011.

[5] H. Nissenbaum. Privacy as contextual integrity. *Washington Law Review*, 79(1), 2004.

[6] A. Whitten and J. D. Tygar. Why johnny cant encrypt: A usability evaluation of PGP 5.0. In *Proceedings of the 8th USENIX Security Symposium*, pages 169–184, 1999.