Toward a Method for Privacy Vulnerability Analysis

Carlos Jensen
GVU Center, College of Computing
Georgia Institute of Technology
Atlanta, GA 30332, USA
carlosj@cc.gatech.edu

Author Keywords
Design, Privacy, Heuristics, Goal-oriented Analysis.

ACM Classification Keywords
H5.2 [User Interfaces]: Theory and Methods. D2.1 [Requirements/Specifications]: Methodologies. H5.m [Information Interfaces and Presentation]: Miscellaneous.

INTRODUCTION
Users are concerned about their privacy, and many of the problems they face in managing their privacy are fundamentally HCI problems. Palen and Dorish note that privacy is a fluid process [4]. Individual differences in preferences and thresholds are often significant, and constantly evolving over time. Given a moderately complex system, it is likely impossible to accommodate everyone.

Designing for privacy must therefore focus on facilitating awareness of policies, information collection, information use, and control over exposure. The challenge is identifying the processes users care about. Many non-trivial problems are caused by high-level architectural problems affecting many aspects of a system. These are hard to identify before a system is built, or fix afterwards.

We present a method for analysis focusing on the discovery of privacy issues. A goal-oriented analysis [5] provides structure to the problem, and a set of design heuristics help identify and analyze potential vulnerabilities. We derive our heuristics from the FIPs [2]. We break these down; emphasizing areas we have identified as problematic [3], and augment them with parts of other guidelines [1].

FROM GUIDELINES TO STRUCTURED ANALYSIS
In goal-oriented analysis, a domain is a collection of actors that achieve goals. Actors correspond to the major architectural components and human roles, whereas goals identify the system’s purpose. High-level goals are broken down into sub-goals which together guarantee the top-level goal. Goals are decomposed until they can be trivially understood and each leaf can be assigned to an agent. A vulnerability is an “anti-goal.” To identify vulnerabilities we resort to heuristics.

Privacy Heuristics
1. Notice/awareness
   I. Available, Accessible and Clear
   II. Correct, Complete and Consistent
   III. Presented in context
   IV. Not overburdening
2. Choice/Consent
   I. Meaningful options
   II. Explicit consent
3. Integrity/Security
   I. Communication, Storage and Execution
   II. Transparency of transactions
4. Enforcement/Redress
   I. Access to own records
   II. Ability to revoke consent

Once the vulnerabilities have been identified, we look for common causes to identify solutions. It is also possible to do a cost-benefit analysis to determine which vulnerabilities are worth addressing. It is not always possible to address all vulnerabilities. It may be the case that these are caused by dependencies on other systems such as the OS, though it is important to know that these may affect the user.

STATUS OF RESEARCH
The focus of this work has been refining and experimenting with this framework. We have used this method to analyze web browsers and group calendars. We are currently in the process of implementing some of our designs to verify the validity of the analysis. We are preparing user studies to determine the ease of use of our method.

REFERENCES