Escape From the Matrix: Lessons from a Case-Study in Access-Control Requirements[*]

[Poster Abstract]

Kathi Fisler
WPI
kfisler@cs.wpi.edu

Shriram Krishnamurthi
Brown University
sk@cs.brown.edu

1. A QUESTION AND A CASE STUDY

The freedom to share information online has made the ability to restrict that sharing critical. Access-control policies are thus a central and growing part of contemporary Web-based system security. Many policy languages—both industrial and academic—are essentially defined in terms of role-action-resource triples. As authors of non-trivial policies [2], however, we have often preferred to describe rules in other forms as well: for instance, as information flows between end-points without having to specify the intermediate operations by which the information may be transmitted. We thus set out to understand the different ways in which users express their policy expectations.

To this end, we conducted interviews to gather requirements for a Web-based application to manage faculty job applications for a computer science department.[1] In such systems, applicants submit their materials (vita, statements, etc.) via a Web interface. The software emails a letter-submission URL to each reference letter writer. Department faculty, and perhaps graduate students, can view and comment on the applications online. Administrative staff handle various requests from members of the department. Technical (computing support) staff maintain the infrastructure. Even in this setup, there can be significant disagreement about some access decisions (as our interviews confirmed): Should applicants be allowed to check which of their reference letters have arrived (and when)? Should students in the department know who has applied? Should administrative staff be able to read the reference letters?

Twelve faculty gave recorded interviews about their "security" requirements, each lasting 20–30 minutes. As all participants had extensive experience with the problem domain, we did not describe it to them. Even though we asked about *security*, the vast majority of participants focused of their own accord on *access-control.* Once participants had run through the cases that had occurred to them, we asked follow-up questions about roles and resources not yet covered. We let participants speak free-form rather than structuring the discussion.

2. FINDINGS

Analogy and Relationship are Fundamental Idioms.

Participants' reliance on analogy was striking, particularly given the variety of forms in which it arose. They used analogies both to express rules and to justify them, often combining the two. They routinely phrased rules or rationales in terms of relationships between roles and resources. Nine of the twelve participants stated at least some rules using the form "treat X the same as Y" (possibly "with the exception of Z"). Sometimes the analogies were used to express a negative, rather than positive, expectation (that two roles should not be treated similarly). Participants often spoke of relationships between the sets of permissions accorded particular roles. For example, one participant wanted permissions of one role to lie between those of two other roles, but didn't know what set of restrictions might achieve that.

The Org Chart is Dead, Long Live the Org Chart.

Security-requirements processes often start from whatever documentation an organization has available about the system to be secured and the staff who will use it [1, 3]. Organizational charts are potentially useful starting points for specifying access control, as they suggest preliminary roles and relative levels of responsibility within the organization. Hierarchies between roles with respect to levels of privilege emerged naturally during our interviews. However, in two cases, *the privilege hierarchy that emerged from the interviews reversed or contradicted relationships that would have been in the org chart.* This suggests that the org chart, vested with authority, is actually dangerous as a starting point for forming role-hierarchies within policies.

Policy Authors Don't Track Roles in Space and Time.

An individual's access privileges can change over time even as the policy remains fixed: an individual's role might change, or the access guards might depend on the status of a resource (such as whether an application is complete). Role overlaps and changes, in particular, can result in information leaks. Only four participants raised the possibility of overlaps and changes, such as graduate students who become applicants and applicants who become faculty. While the annual cycle of faculty hiring may have masked these issues, participants tended to not reference time at all.

Social Contracts Identify and Protect the Real Assets.

Thinking about access-control policies in terms of concrete resources, as tabular- or rule-based authoring does, can sometimes entirely miss the point. Our interviews revealed that the single most important resource was one that shows up nowhere in tables and rules: the department's *reputation.* Often, this was the resource that people were really

[*] A full version of this paper is online as Brown CS TR 09-05.

[1] The resulting software, which has been used to conduct multiple searches, is available for free from the authors.

trying to protect, even though they were stating concrete rules about other (tangible) resources based on how they thought those decisions would impact this resource. While only one participant explicitly mentioned reputation, others cited correlating concerns such as practices in other departments, departmental loyalty, and the tension between the educational value and (unstated) consequences of letting students have too much access to application materials.

Participants Exhibit Personality Styles.

We were struck by the different approaches—bordering on "personalities" or "styles"—that participants adopted during the interviews: **social thinkers** were conscious of the values that policies encoded regarding department culture, collegiality, and social contracts with applicants and letter writers; **problem avoiders** saw policy as protecting against undesirable situations; **pragmatists** focused on realities such as making sure policies wouldn't interfere with workflow or on granting access based on similar data available to users from outside the system; **protectionists** framed most comments around what principles of confidentiality or least privilege would demand. In our subject pool, we identified two social thinkers, two problem avoiders, and four each of pragmatists and protectionists.

The pragmatists were most likely to state rules fairly abstractly ("give [staff] a certain level of privilege so they can do whatever that faculty member needs", or "I want to work on paper"). However, most did have clear boundary cases for which they felt strict access controls were essential. The participants who needed the most prompting were pragmatists or protectionists. These two groups were also more likely to state rules in the form "treat X like Y". Social thinkers contributed only one comment about the human processes surrounding the hiring software, whereas problem-avoiders and protectionists each made roughly ten comments in this space. Perhaps not surprisingly, the two problem-avoiders had administrative experience; they almost always stated rules very concretely, rather than through or relative to general principles. The full paper describes several other patterns such as these.

No Dominant Structure or Consistent Format.

Given the technical expertise of the participants, we expected interviews to reveal a systematic process for exploring the policy space. Only one of the twelve interviews had such a structure; that participant articulated scenarios for each role in turn. Each of roles, resources, scenarios, and the existing process was used by a third to a half of the participants to initiate a new thread of conversation. While this does not imply that participants would have had difficulty articulating policy against a single organizing structure, it does raise questions of whether a single organizing principle fits all, and whether using a variety of prompts will better cover the state space. Each participant used multiple forms to state access-control rules, such as "role R is permitted/denied to do action A on resource S" and "whatever access role R needs to do task T".

3. IMPLICATIONS

Significant research remains to be done into cognitive aspects of policy authoring and their impact on policy languages and authoring and analysis tools. Much policy language research focuses on expressive power from a purely logical perspective. Though valuable, such work ignores fundamental questions of how well the languages capture what people are trying to say while making policies easy to understand and maintain. Understanding and designing policy languages that better account for users raises numerous important problems, including the following:

- The simple and rigid forms of modern policy languages fail to capture common idioms such as analogies or rules based on sources of information. These are subtle, as analogies may become invalid over time.

- Policy-authoring tools need to go beyond asking users for roles, actions, and resources (the typical inputs to matrix-based tools [4]). Users' mental models of security may be very limited (e.g., the persistent reliance on least privilege). Furthermore, informing users about other paradigms (e.g., separation-of-duty) may help determine whether these are manifest in the domain.

- Identifying assets to protect is part of every security requirements process, but the true assets are not necessarily the resources under access-control. How to align these with controllable resources is an open question.

- Policy-authoring tools should include analyses such as reporting similarities between roles and flagging contradictions with existing documentation (such as org charts). Prompting authors about these relationships may help identify missing policy cases. In general, we believe there is considerable room for what we dub an *inquisitive environment* that takes the place of today's passive policy-entry interfaces, asking users questions ("Does this role have sub-roles?", "Can this permission change over time?", "Does this datum have parts?", and so on) to tease out the actual requirements, instead of blindly trusting the user's original selections.

Acknowledgments.

This work is partially supported by the NSF and by a Google Research Award. We thank our participants for their time and insights.

4. REFERENCES

[1] A. I. Antón and J. B. Earp. Strategies for developing policies and requirements for secure e-commerce systems. In A. K. Ghosh, editor, *Recent Advances in E-Commerce Security and Privacy*, pages 29–46. Kluwer Academic Publishers, 2001.

[2] K. Fisler, S. Krishnamurthi, L. A. Meyerovich, and M. C. Tschantz. Verification and change-impact analysis of access-control policies. In *International Conference on Software Engineering*, pages 196–205, May 2005.

[3] C. Haley, R. Laney, J. Moffett, and B. Nuseibeh. Security requirements engineering: A framework for representation and analysis. *IEEE Transactions on Software Engineering*, 34(1):133–153, 2008.

[4] R. Reeder, L. Bauer, L. Cranor, M. Reiter, K. Bacon, K. How, and H. Strong. Expandable grids for visualizing and authoring computer security policies. In *ACM SIGCHI Conference on Human Factors in Computing Systems*, 2008.