Real Life Challenges in Access-control Management

Lujo Bauer[†] lbauer@cmu.edu
Lorrie Faith Cranor[†] lorrie@cmu.edu
Robert W. Reeder[†][*] roreeder@microsoft.com
Michael K. Reiter[†][‡] reiter@cs.unc.edu
Kami Vaniea[†] kami@cmu.edu

[†]Carnegie Mellon University, Pittsburgh, PA, USA
[‡]University of North Carolina, Chapel Hill, NC, USA
[*]Microsoft, Redmond, WA, USA

ABSTRACT

In this work we ask the question: what are the challenges of managing a physical or file system access-control policy for a large organization? To answer the question, we conducted a series of interviews with thirteen administrators who manage access-control policy for either a file system or a physical space. Based on these interviews we identified three sets of real-world requirements that are either ignored or inadequately addressed by technology: 1) policies are made/implemented by multiple people; 2) policy makers are distinct from policy implementers; and 3) access-control systems don't always have the capability to implement the desired policy. We present our interview results and propose several possible solutions to address the observed issues.

Author Keywords

Access control, policy creation

ACM Classification Keywords

D.4.6 Security and protection, K.4.3 Organizational Impacts, K.6.5 Authentication

INTRODUCTION

Effectively controlling access to resources within an organization is a challenging problem for access-control policy professionals. Making sure the correct person has access to the correct resource at the correct time often requires communication within and between departments. Access-control policy changes are needed when employees are hired, terminated, or change job roles. Temporary changes may be needed when employees are given temporary assignments. Changes to specific access-control policies may be needed when company-wide policies change. The introduction of new computer systems or the retirement of old systems, as well as changes in physical office space are other reasons for access-control policy changes.

Many high-profile access-control policy failures occurred when employees or former employees were able to use their legitimate accounts to carry out crimes because they had been given inappropriate access to a system, or their access had not been revoked in a timely manner. For example, in 2008 the Société Générale Bank in Europe reported a $7.2 billion trading loss. An employee had transferred from a compliance role in which he monitored trading to a trading role in which he made the trades. Using his extra knowledge from working in compliance and access rights that were not removed, he was able to make trades far in excess of company policy. The result was one of the largest reported trading losses in banking history [11].

A study of 23 insider attacks found that "in 78% of the incidents, the insiders were authorized users with active computer accounts at the time of the incident. In 43% of the cases, the insider used his or her own username and password to carry out the incident" [16]. A related study of 49 insider attacks found that 59% of the insiders were former employees and 43% still had authorized system access at the time of the attack [10]. These findings indicate that current systems may be inadequate at supporting policy professionals' needs, even for routine tasks such as revoking access when employees leave an organization.

In this study we sought to understand the challenges policy professionals face in their daily tasks. We focused on understanding to what extent policy management technology was successful or unsuccessful in helping policy professionals meet these challenges. To do this we conducted 11 interviews with 13 policy professionals in 5 organizations.

The data from these interviews lead to three key findings. First, we find that policy professionals take different roles in creating policy—some are high-level policy architects, others are implementers of policy designed by others. Current policy-management technology does not acknowledge this distinction, and hence fails to provide tools specifically suited to each role. Second, we find that policy is often jointly managed by several people rather than a single individual. Although technology sometimes aids these individuals in coordinating their activities, such tools are typically poorly integrated with the mechanisms for creating and manipulating policy. Third, we find that some commonly desired policies cannot be fully enforced with the access-control mechanisms that are used to implement them, leading to cumbersome workarounds.

In the remainder of the paper we discuss the methodology of the study, the results of our interviews, and how they support

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHI 2009, April 3 - 9, 2009, Boston, MA, USA.

Copyright 2009 ACM 978-1-60558-246-7/07/0004...$5.00.

Pseudonym	File or Physical	Organization	System Managed
Ann & Kristen	Physical	University A	Department-wide swipe-card, physical-key and key-pad systems
Henry	Physical	University B	University-wide swipe-card system
Tony	Physical	University B	Department-wide swipe-card system and physical-key management
Kevin	Physical	University B	Department-wide swipe-card, physical-key and key-pad systems
Fred	File	University B	Department-wide Windows and Unix-like file systems
Jerry	Physical and File	University B	Physical-key and electronic systems for a lab
Sue	Physical	University B	Department-wide physical-key system
Seth	File	Organization A	Organization-wide file system
Ralf	File	Organization A	Organization-wide file system
David	Physical	Organization B	Organizational-wide swipe-card and physical-key system
Beth & Sara	Physical	Organization C	Department-wide swipe-card and physical-key systems

Figure 1. List of the interviewees, the type of system they worked with and the role they played in managing the access-control policy for that system. All interviewees are referred to by pseudonym.

each of these key findings. For each key finding we suggest ways in which technology that supports policy professionals could be improved to better match the needs of its users.

METHODOLOGY AND DATA ANALYSIS

The study was designed to elicit an understanding of the challenges access-control policy professionals face and how current technology helps them meet these challenges. We began the study with no hypothesis and through the interview and data analysis processes we incrementally constructed theories concerning access-control management.

Interviewees

Interviewees were recruited using existing contacts. We interviewed thirteen policy professionals from five organizations. In two cases two professionals who shared the same job function were interviewed together. Eight of the interviews were with policy professionals who manage a physical-access-control system and three were with policy professionals who manage access control for a file system.

We purposely chose to consider both physical and file access control in our work because both use increasingly similar computer-based management interfaces. Every administrator who was interviewed worked with at least one access-control system that had digital components and was administered using a computer interface. Additionally, researchers are starting to create computer technologies to solve problems with physical security systems [5, 6], further eroding the line between physical and file access-control systems.

Prior work indicates that, in some organizations, responsibility for administering access-control policy tends to be delegated, with a central authority delegating responsibility to department administrators, who in turn pass on the responsibility to other people in the department [5]. To better understand the needs of professionals at different levels of an organization's hierarchy, we specifically selected participants from multiple levels of the organization.

Organizations

We use pseudonyms to identify the universities, organizations, and policy professionals discussed in this paper.

University A is a public university that has approximately 37,000 faculty, staff and students at its main campus. We interviewed two administrative assistants, Ann and Kristen, who manage physical access control in their department using a swipe-card system, physical keys and key-code pads. Their department contains roughly 150 faculty, staff and graduate students. They also interact with undergraduate students, who have a high turnover rate.

We conducted separate interviews with six policy professionals at University B, a private university with a campus population of approximately 12,000. Henry manages the campus-wide swipe-card system that controls access to most university buildings. Kevin and Tony manage physical access control for their respective departments using the swipe-card system, keys, and key-code pads. Both Kevin and Tony support departments of approximately 1,500 people. Fred manages a file system for the same department as Tony. Sue manages another building at University B that houses 170 people from several departments. Finally, Jerry is the lab manager for a lab located in Sue's space. Jerry's research group includes 70 people, but researchers from other groups occasionally need access to Jerry's lab.

Organization A is a large non-profit membership organization that has research departments. Organization A has approximately 1,200 full- and part-time employees and about 1000 volunteers. The organization is divided into five divisions, which are physically located at different sites around the city where it is based. Each division has its own departments, systems, policies and cultures, which are loosely linked by the main organization. Seth is the Security Director for all the divisions in Organization A and Ralf is the central network administrator.

Organization B is a non-profit organization that spans multiple states. The organization takes security very seriously; for them, a single breach could be detrimental to their business model. David is their central administrator and he controls access using swipe cards and physical keys.

Organization C is a smaller non-profit of around 200 people that researches and evaluates data provided by other organizations. They also take security very seriously because their business model depends on other organizations trusting their security measures. Beth and Sara are the two administrators tasked with overseeing the physical-key and swipe-card systems for the organization.

The individuals we interviewed represent a broad range of policy professionals from a variety of different types of organizations with differing organizational structures and access-control needs. However, this study did not include interviews with policy professionals in for-profit companies or very large organizations. While we expect that most of our findings are likely applicable to for-profit companies and very large organizations, interviews with policy professionals in these organizations would likely reveal additional issues not discussed in this paper.

Semi-structured interview

We used semi-structured interviews as our method of inquiry because they allowed us to focus on several primary questions but still have the flexibility to explore comments made by the interviewees. We designed our questions to focus on typical policy-management tasks but we also asked if the person had ever had to deal with specific incidents such as quickly revoking access rights from a terminated employee. The majority of people we interviewed performed policy-management tasks as only a portion of their job and in some cases fairly infrequently. The questions were designed to not only explore topics of interest, but to specifically bring up common incidents as a way of encouraging interviewees to remember specific events.

Our questions focused on several topics:

- Overview of interviewee's role in the organization.
- Technologies used by the organization and interviewee to control access within the organization.
- Policy changes caused by employee movement in the organization, including new employees, terminated employees, temporary employees, and employees who have moved internally in the organization.
- Written and unwritten procedures for making changes to the implemented access-control policy for a resource.
- Security incidents that have happened or could happen in the organization.
- Procedure for reviewing the implemented access-control policy for errors and checking the access logs for irregularities.

Data Analysis

The interviews were conducted in concurrence with the data analysis to better facilitate theory building. After conducting each semi-structured interview we used the audio recordings or detailed notes collected during the interview to analyze the interview content by building workflow, artifact, sequence and cultural models [7]. During these analyses we identified interesting topics which were recorded and added to the list of questions used in successive interviews. When the interviews were completed we used affinity diagrams to organize the topics we identified in our interviews. Topic groups were then used to construct theories. This approach is similar to other studies presented at CHI and SOUPS [8, 14, 19].

We constructed affinity diagrams [7] using comments, issues, breakdowns and successful solutions identified while constructing the work models. We wrote each piece of information on a sticky note and organized them into groups of similar topics. When reviewing topic clusters on the affinity diagram, we noticed that some topic groups described both problems and solutions but others only described solutions that indicated the presence of unmentioned problems. Using information from the work diagram and the affinity diagram, we identified the problem and solution (if any) that each topic represented.

Using the complete list of problems and solutions discussed in our interviews, we identified common problem themes. We grouped the problems based on similarity and causation in order to better understand the larger issues.

ROLES OF POLICY PROFESSIONALS

Our interviews revealed two types of policy and two roles for policy professionals. *Policy makers* formulate *intended* policy—policy that they believe should be enacted. Intended policy represents a single person's or a group's intentions; multiple intended policies that refer to the same resource could potentially be inconsistent with each other. For example, an employee at Company A may want to give access to her files to her friend at a competing company, but this is inconsistent with the general policy of Company A. *Policy implementers* translate the intended policy into the *implemented policy*—policy that is enforced by the access-control technology deployed by the organization. In doing this, they may need to adapt abstractly defined intended policy to fit the capabilities of the access-control mechanism and recognize or resolve inconsistencies or oversights in the intended policy. The distinction between policy makers and policy implementers is key to understanding how access-control policy is managed in an organization.

A person filling a policy-maker role is both empowered to make decisions concerning portions of the organization's policy and has (some of) the knowledge required to know what changes should be made. Policy decisions include assigning users to groups and giving individual users access to specific resources. Policy makers do not necessarily know how to change or view the implemented policy.

A person filling a policy-implementer role has the ability to make changes to and view the implemented policy. Unlike a policy maker, a policy implementer does not necessarily have insight into what changes need to be made or why, or what policy needs to be put into place. Implementers depend on policy makers to decide what changes should be made and what the policy should look like.

It is possible for a single person to simultaneously fill the roles of policy maker and policy implementer. For example, an end user of a file system may both know what the policy for his files should be and have the ability to change the access-control permissions for those files. In a centralized system, an end user may be forced to request certain changes to the file permissions, which the central system administrator then implements. Our interviews were focused on policy professionals in centralized environments and all our interviewees were either policy makers or implementers.

We found that the largest issue faced by implementers is knowing what changes need to be made to the policy and when to make them. Conversely, policy makers know what the policy should look like but have limited to no ability to view or manipulate it. This issue arises because makers and implementers are typically different individuals, and because coordination can be difficult.

POLICIES ARE MANAGED BY MULTIPLE PEOPLE

Many policy professionals expressed concerns about managing a policy where multiple people are capable of changing the policy with little or no notification. Issues mentioned by policy professionals ranged from concerns about synchronizing policy edits across multiple professionals to the difficulty of managing exceptions to the policy. A common theme was a need to have a way to know at all times what the policy says and whether it is still accurate.

Maintaining an understanding of the implemented policy

For many policy professionals the biggest challenge is maintaining a good understanding of the current implemented policy. Policy implementers need a working understanding of the policies they maintain because they are asked to make decisions based on the policy and it is not always convenient to access the policy itself to answer the questions. While we were interviewing him, Ralf received a phone call on his mobile phone concerning an employee in department A who was filling in for an employee in department B. The employee couldn't log into the computer of the employee she was replacing. Because he has an excellent working knowledge of his organization's network access-control policy, Ralf was able to identify the problem, determine whether a temporary exception was needed and instruct his assistant to fix the problem, all without accessing his computer. Ralf explained that being able to solve problems over the phone was very valuable because he was rarely at his desk and didn't always have access to a computer where he could look things up.

When only one person manages the policy it is easy to maintain a working understanding of the policy. However, 11 of our 13 interviewees worked with at least one *policy implementation peer*, a person with similar responsibilities and abilities as themselves. With multiple policy implementation peers making changes to the implemented policy, it is difficult for any one policy professional to maintain an understanding of the state of the current policy.

The policy implementers we interviewed solved the issue of not knowing what other policy professionals were doing by using a standard set of heuristics for dealing with policy maker requests and by notifying others about changes. David is the primary policy implementer for a physical-access-control system. Whenever an incident occurs in any of the buildings he manages, he is the one who gets called and asked to explain why the incident happened. David likes to know what changes are being made to his system because he may be asked about the implemented policy at any time. When discussing how he coordinated policy changes with his team, he told us that he trusts his fellow policy implementers to know what a normal request looks like and to address the request appropriately. However, he still wants to be notified after any such change.

Ralf has a more complex coordination problem amongst his policy implementation peers. Each of Ralf's coworkers is responsible for a different part of the organization's policy. For example, one of Ralf's coworkers manages the firewall policy. Another coworker manages the file-system policy for one of the departments. Only Ralf has an understanding of how all the different systems and policies interact. When any of his coworkers has a question about another part of the system, the coworker goes to him. Ralf told us that he makes sure that he is aware of all changes occurring on his system. He instructs all his fellow coworkers to report any changes they make to him so that he always knows the state of the system. Having a holistic knowledge of the system lets him make decisions without having to consult anyone else or dig through system files. Ralf commented how his memory was the most complete set of documentation at the organization. His manager wanted him to start documenting the information he was collecting because it isn't written down anywhere, and if Ralf ever had an accident then no one would know what was going on in the system.

Only one policy implementer, Fred, wasn't concerned about maintaining a working knowledge of the implemented policy he worked with. Fred's department has about ten policy-implementation peers and the department is known for having a lot of employee turnover. Each policy maker who uses the file system has the ability to make changes to the implemented policy for their files. With so many individuals making changes, maintaining a working knowledge of the policy is infeasible. Fred doesn't bother trying to understand the current state of the system before making changes and instead simply verifies that the requested changes don't conflict with the high-level intended policy of the university, which is fairly loose.

Implementers also discussed giving two independent groups of policy professionals responsibility for making alterations to a resource's implemented policy. All four implementers who mentioned it felt that it was a bad idea. Kevin and Tony both felt that either they should manage the implemented policy for a room themselves or the policy maker should manage it directly, but they didn't want to be placed in a situation where they might be blamed for a change they did not make. Henry, who manages a physical-access-control swipe-card system for all of University B, has a similar opinion. He makes sure that every door in his system has exactly one group that can change its implemented policy. Tony talked about how he had once requested that Henry let him share management responsibilities for a door with another department. Henry had refused the request and told Tony that either Tony's department could manage the door's policy or the other department could, but not both.

Exceptions are hard to manage

Exceptions to normal policy were a problem even for groups who had established an effective method for communicating policy changes. An exception is any change to the implemented policy that violates the "normal" intended policy of the organization. For example, giving an office key to someone who doesn't work in that office when the organization has a "one office, one key" policy is an exception. Normal policy changes such as adding a new user have a well-defined set of tasks associated with them. Adding an exception to the policy means the implementer must manage the exception separately. Implementers who tried to maintain knowledge about the current policy state found exceptions especially irksome.

None of the implementers like exceptions and four of them attempt to ban exceptions from their systems. Ralf dislikes allowing exceptions because they are hard to manage, and worse, it is hard to remember that the exception exists. On his file server, Ralf has a policy that each user gets her own directory that only she can access and each project group gets a common directory that can be accessed only by members of that group. Ralph explains:

> They have that common [disk] drive, and occasionally they get into this situation where they're like, "I don't want anyone else to see that," you know, because anyone in their department can see that.... And you're like, "OK, so, like now I have to make another folder just for you two?" It actually starts to become an administrative nightmare.... I try not to make too many changes and I try and explain that to them upfront and say, "Look if you want I'll do this once but I don't want to be doing this five times."

David, Beth and Sara were concerned about the possible negative effects of allowing exceptions to their policies. Because their organizations take security very seriously, it is important that employees such as security guards be able to spot abnormal access behavior. One way this is done is by using chemically-treated temporary badges that change color over time, allowing anyone to identify temporary visitors who have stayed too long. Similarly, they want employees to be able to identify odd access behavior. Allowing exceptions makes the policy non-standard and makes it harder for employees to determine whether someone has legitimate rights to a space or not. In general, the policy professionals from both organizations attempt to limit or prevent exceptions. In the rare case where an exception is necessary, Beth and Sara grant the exception, but their resistance to exceptions and the small size of their department means that there are only ever a few exceptions in place at any given time. David manages several departments so he completely refuses to add exceptions to the system's implemented policy. Instead, the security guards maintain a list of people whom they can let into certain rooms. A new person, room pair can be added to the list by filling out a form at the guard desk. This workaround allows people to be given access without adding exceptions to the implemented policy. The solution also allows the security guards to identify odd behaviour.

Getting policy-change notifications

Many policies depend on information from multiple sources. A common type of policy, for example, gives all employees access to a resource. The policy maker who formed this policy does not, however, know who all the employees are; this information is managed by the human resources department, which in this way also plays the role of a policy maker (e.g., granting access to newly hired employees and revoking access to departed employees). The implemented policy is a result of appropriately combining input from the two policy makers. Accounting departments, which typically allocate internal charges for network access and other services based on each employee's home department, are also potential sources of information about employee internal movement and employee termination.

Four of the interviewed implementers mentioned the benefits of setting up their access-control system to use records maintained by another department. Three other implementers mentioned how they were currently trying to establish better relations with the human resources or accounting departments in an effort to more quickly get information about changes in employee status.

Henry manages a swipe-card system that controls access to physical and virtual resources at University B. The turnover of people involved with the university is so high that he doesn't want to individually add and remove each person from the system. Instead, his department works closely with the Registrar, which monitors the status of all faculty, students and staff at the university. The swipe-card system is linked in with the Registrar's system so that when new people join the university they are automatically given access to communal university resources. When people leave the university their access rights are automatically removed.

Henry's arrangement with the Registrar also helps Tony who manages access control for one of the departments at Henry's university. Since Henry's department automatically adds and removes swipe-card accounts, Tony doesn't need to worry about routine university events and can focus on department-specific access-control concerns.

Documentation is old or wrong

In several cases policy implementers discussed making decisions based on information stored inside the system that was out of date or wrong. In these cases, policy implementers had to recognize that the documentation was not valid and find alternative ways to get the data they needed.

One such example came from Fred, who receives requests from people who want access to files and folders on a file server. Fred uses the access-control list in the file system to determine who owns the folder or file and treats that person as the only person who is allowed to make decisions about it. Occasionally, he will tell a requester that they have to get the folder's owner to send him the change request only to be informed that the owner is no longer at the university. This is problematic for Fred since he must then find the new person in charge of the folder's policy and update the system.

Another example of information being entered into the system and becoming stale comes from Kevin, who manages physical access for his department at University B. Every time a person is given a key to a space, this fact is noted on an index card titled with the person's name. If the lock is rekeyed, however, this information is not added to the card. In order to determine if someone has access to a specific room, her card must be pulled up and the number of the key she was given must be compared with the current key number for the door, which must also be looked up in a separate record. Information stored on the card about which door the key opens cannot be trusted since it may be old.

Documentation can also be completely missing. Kevin told us that his department also uses a swipe-card system to control access to some resources. However, since not all students, staff or faculty need access to these resources, swipe cards are issued only on an as-needed basis. An administrative assistant gives out the card and notes this fact on a piece of paper, so that later an implementer can activate the card and add it to the system when he has time. Consequently, the database of swipe cards is often incomplete since the implementer doesn't always have all the information available when he enters the card into the database. Without complete information, knowing who has what card is difficult.

Discussion

Working with multiple policy professionals can cause problems with keeping relevant people apprised of the current policy state and keeping the policy synchronized. Policy implementers feel they need to be notified about changes in the policies they manage. This suggests a need for technologies that provide notifications when policies change and provide methods of documenting why a change was made. They also need a way to incorporate parts of the implemented policy that are maintained by other departments.

Make documenting implemented policy changes part of the natural workflow. The majority of the problems described by policy professionals trying to coordinate edits to the implemented policy centered on their need to know what the current implemented policy looks like. One solution to this problem is to encourage policy implementers to document their changes to the policy. Good documentation would allow other policy professionals to learn about the policy without having to memorize it.

It would be better, however, if documenting the reasons for a change in the implemented policy was an integral part of making a change to the policy. This could be done in two ways. First, policy management systems could require users to document a policy change (and aid them in doing so) before the change was accepted by the system. Second, the implemented policies could be specified in a self-documenting policy-specification language [1, 2, 15]; i.e., the implementation of the policy could preserve many of the abstractions of the intended policy. For example, the implemented policy could explicitly encode the sub-policies, "John is a student," and, "students can access the lab," instead of encoding just, "John can access the lab," as is more common. Some policy-management systems provide such functionality and could be further improved to support implemented policies that are even closer to intended policies.

Provide a way to keep policy implementers apprised of changes to implemented policy. For many of our policy implementers, having good documentation that could be consulted in case of need wasn't enough. They needed to have an excellent understanding (without referring to documentation) of the implemented policy at all times to properly do their job. For these policy implementers, we recommend using a publish-subscribe technology where the system automatically sends out updates when the policy changes and implementers can indicate that they want to receive updates about certain parts of the policy.

Automatically update compound policies. Implemented policies may depend on information that is maintained on separate information systems, e.g., databases spread among several departments may house different pieces of information relevant to the policy. Integrating these different systems so that the implemented access-control policy is automatically kept up to date has many potential benefits. Several groups we interviewed used systems that had this functionality.

POLICY MAKERS ARE DISTINCT FROM POLICY IMPLEMENTERS

Another major issue expressed by both implementers and policy makers is the challenge of knowing when a change needs to be made and determining what that change should be. Policy makers expressed concerns that the implemented policy does not match their intended policy and it is difficult to view the implemented policy in order to make sure that the changes they requested were actually made. Implementers discussed problems with knowing when a change needs to be made, verifying that the person requesting a change has the appropriate authority, and maintaining records demonstrating the request.

Viewing implemented policy

Unlike policy implementers, policy makers typically do not have the ability to view or manipulate implemented policy directly. Instead, they have to find and query an implementer to get an understanding of what the implemented policy looks like. Everyone we interviewed mentioned at least one incident where they had to ask for or were asked for a report about the implemented policy.

Since policy makers do not know what the implemented policy looks like, they have no way of knowing if it is correct or not. Those policy makers who are concerned about the wrong people accessing resources for which they are responsible request portions of the implemented policy from an implementer and review it for errors. According to Sara, both she and Beth review the access-control lists (ACLs) for each door in their department once or twice a year—however, they would like to do so more often. Since they don't have direct access to the ACLs they send a request to the implementer who sends them the ACL for each door. They then go through these lists looking for anyone with inappropriate access. Sara tells us that occasionally they do find people who shouldn't have access. She attributes this to a "slip of the finger" on the implementer's part. After they review all the ACLs they send a list of corrections to the implementer who makes the requested changes.

Kevin and Henry both mentioned that they occasionally get requests from policy makers wanting to know the implemented policy for their resources. Henry says that he gets general requests asking for a list of everyone who can access a specific area. Kevin's system doesn't support the ability to create a list of everyone who can access a space, and so he doesn't get many of those requests. Instead, he gets asked about specific people. Kevin says that a few times a year he will get an email from a policy maker asking if a specific person has access to a specific room because someone has just spotted the person there and is not pleased about it.

Getting notifications about policy changes

A major problem faced by implementers is knowing when the policy needs to be changed. Since an implementer doesn't always know the intended policy, it is difficult for them to detect inconsistencies without the help of a policy maker. Implementers either ignore these problems, trusting that a policy maker will notify them when a change needs to be made, or they proactively attempt to get change information from policy makers.

Ralf, Seth, David, Kevin, Sue and Fred all discussed instances where they were not notified about pertinent personnel changes which should have resulted in changes to implemented policy. Ralf, in particular, was annoyed about not being told when employees leave the organization:

> I try to disable an account as soon as I know that account [holder] is gone.... As soon as you do it then all of a sudden they are complaining because they will try and bring somebody else in and say well they were using that account and I'm like, "No, that doesn't work, they need a new account...." I just don't like them using [an account] under somebody else's name.... Who knows if someone else knew what their password was or that person got back into their account again and is using it along with this person.

David found a more proactive solution. Instead of waiting for a policy maker to complain or request an access-control list for review, he proactively sends out lists to all policy makers on a monthly basis. This method allows David to find potential errors in the access-control policy for his organization before they become issues. It is unclear how effective this method actually is since Beth and Sara, whom we also interviewed, regularly receive these periodic lists but only review them once or twice a year.

David explained how he also tries to encourage policy makers to send him information about policy changes—such as employee termination and internal movement—in advance so that he can schedule the changes and ensure that the employee loses and gains access at the appropriate times. Temporary employees such as students are entered into the system with a start and end date so they are automatically removed once they leave. David says the system works well but every so often a policy maker will forget to tell him about a change in plans and someone will be denied access.

Verifying requests and keeping records

Since an implementer does not have perfect knowledge of the intended policy, she has to trust policy makers to make the correct decisions about what policy should be applied to the resource. However, when a problem occurs, implementers are concerned they will be blamed since the state of the implemented policy is their responsibility. To address this issue, implementers perform sanity checks on requests, verifying that the request matches the organization's policy and that the person making the request is authorized. Implementers also keep records of the change requests they receive so they can reference the records if there is ever a problem.

One of the first issues an implementer encounters when presented with a policy change request is validating that the requester has the authorization to request the change. Of our interviewees, six know who owns each of the resources they support. The rest of the implementers either consult documentation to determine ownership or find another trusted person to ask. For example, when Fred gets a request to give someone access, he consults the file or folder in question and determines if it is owned by the requester. If the folder's system-indicated owner is no longer at the organization (a reasonably frequent occurrence), Fred sends an email to a trusted administrative assistant and asks who has taken the previous owner's place in the project group.

Implementers are also concerned about accountability. Most implementers we talked to keep records of who requested each change along with some sort of proof. Typically, these records are the emails requesting the change. Our interviewees expressly pointed out to us that they keep these emails specifically for accountability. Other types of records are also kept by implementers. Kevin's department requires that the requester sign a form before new access is given to someone. Fred's department also uses a form that must be filled out for a new user and includes who the requester is. For other types of requests, Fred uses a help request tracking system that allows him to tag requests involving policy changes.

Discussion

There appears to be a natural divide between policy makers and policy implementers. Both policy makers and implementers perform their own specific sets of tasks, but they need to communicate with each other to accomplish their tasks. Access-control systems should seek to reduce this divide or better facilitate communication across it.

Allow policy makers to directly edit the implemented policy. The principle of least privilege—that a person should be given the minimal rights needed to do her job—is a well established axiom in security [18]. We observed that, in practice, policy implementers often do not have sufficient understanding of the intended policy to accurately enforce the principle of least privilege. Providing policy makers with the ability to make changes to the implemented policy themselves would let them leverage their greater knowledge of the intended policy to create a more accurate implemented policy. To enable this sort of policy creation, access-control systems would have to support interfaces tailored to policy makers, exposing and allowing the policy makers to control only the portion of the policy for which they are responsible and only in ways that match their authority. There has been some success in designing experimental systems with such features: Bauer et al. found that when they gave policy makers a more flexible access-control system the participants created less permissive policies that better fit their needs [4].

Provide feedback to policy implementers. Giving policy makers direct access to their portion of the implemented policy makes some implementers uncomfortable, as they worry that policy makers will introduce errors or leave the policy in an inconsistent state. Beyond building in safeguards that ensure that policy makers cannot implement policies that they are not authorized to make, as described above, systems could improve the feedback implementers receive as a result of changes being made to the implemented policy. In the limit, systems could allow implementers to preview and approve the policy changes introduced by potentially technically unskilled policy makers, thus providing the flexibility for policy makers to implement policy and still allowing other policy implementers to ensure the changes are reasonable.

SYSTEM CAN'T ENFORCE DESIRED POLICY

Policy professionals also have to consider how policies will work in combination with the technology that enforces them and what will happen when people do not follow secure practices. The topic of policy enforcement is very broad and mostly outside the scope of this paper. However, we touch here on enforcement issues that arise as a result of the decisions policy professionals make about managing their resources.

Choosing an access-control technology

When implementers discussed the access-control technologies they used, they almost always began by discussing the enforcement abilities of the system. Implementers were very interested in features such as reliability, the ability to fail gracefully, and simplicity. They had less, if anything, to say about the types of policies the system supported or the management interface.

Nearly all the implementers who managed physical access-control had participated in the selection of the system they worked with. Kevin explained to us how he used different combinations of technologies on every door to get the perfect mix of reliability, security and usability for each lab. Technologies such as keys and key pads were used by implementers because of their reliability, stand-alone qualities and the ease with which access could be shared.

Implementers were very interested in the capabilities of the systems but David was the only implementer who seemed interested in the management software associated with the system. He purposely selected his system because it had a management interface that was easy to use and integrated information from his other security technologies. The other implementers only minorly considered the management software in their selection process. University C's requirement document for their new building primarily specified physical requirements of the system and only occasionally mentioned a requirement for the software (such as the ability to schedule exceptions at least a year in advance).

Implementers didn't mention the usability of the management system when selecting technologies, but they were annoyed by poor management systems. David talked about the old system his organization used which could require that a user be added or removed from as many as ten databases when making a change. Tony tried to show us the management system for the swipe-card system he works with and quickly gave up. He told us that his co-worker had received the training and performed all interactions with the system.

Knowing who has an access token

In the previous sections we have assumed that the person exercising an access right is the person who is intended to be doing so. Unfortunately, this is not always the case, as several commonly used access-control technologies do not link the access request to a person (e.g., as is the case with physical keys).

Using key codes may be more convenient for policy makers, since they can give access without interacting with management technology, but it means that no one knows exactly who can access a resource. Ann and Kristen talked about the problems with using key codes for lab doors. Each lab door has a single key code which is given out to all the occupants. Theoretically, the key code is only known to the room occupants, but nothing stops the occupants from sharing the code with others. As a way of ensuring that only the room occupants have access, the key codes are changed once a term and the new code is emailed out to all the room occupants.

Kevin's department also uses key codes for some doors. He has similar problems as Ann and Kristen, except that his department is much larger and he is not always certain who should be given the new key codes. To solve the problem, he instead emails out the new code to the administrative assistants who work with people in the space, and asks them to distribute the new code. Ann talked about a specific incident where a door code had to be changed:

> We just had a professor the other day who sent an email saying, "Some people I don't want in the lab sought access." So [we changed the code] ... then she just gave it out to two people that she said was ok.

Physical keys can also cause problems. Users frequently give keys to others to facilitate achieving a goal, even if this is not consistent with an organization's intended access-control policy [4]. Keys, also, can be easily copied even if stamped "Do not copy." As a result, it is hard to know who has a key to a room even if accurate records are kept. Tony, Kevin, Ann and Kristen all talked about the need to periodically re-key doors just to be certain that only the correct people had keys.

Unexpected events

Dealing with unexpected events is another important part of enforcement that needs to be considered in the implemented policy. Owners do not think of all possible events *a priori* and unexpected events do occur.

Jerry spent the most time talking about unexpected events since he is the policy maker for a lab filled with expensive equipment. His intended policy for whom he will allow to have access to the lab is fairly restrictive, with only a few staff members given access. However, if an incident occurs (e.g., a fire) he trusts a much larger number of people to enter the lab (e.g., in order to shut down expensive equipment). He would like a system that allows him to give out a type of access that could only be used in emergencies and would immediately warn him when it was used.

Kevin has also had issues with unexpected events. This happens often enough that Kevin started adding key-pad locks onto all lab doors. He puts an administrator code onto each door so that if he gets a call and can't come personally, then he can simply relay the code to the caller and change it the next day.

> What was happening is we were having different people coming in at night, emergencies and what not, not everyone has a key, not everyone has a card. So if I get a call at home. A guy calls and says "Hey I'm down here, I can't get into [a lab]," so I have a code in there that I can give them. I've given it to a lot of maintenance people and security people which gets them in there.... doot doot doot you're in.

Discussion

Managing the actual implemented access-control policy in the wild is a challenging task. Policy implementers are limited by the types of technology available to them. Even when they can choose the technology that best suits their needs, they still have trouble configuring it to their specific situation. Current technologies don't necessarily support policy implementers' need to change intended policy into implemented policy.

Prioritize management interfaces and ability to implement desired policies when choosing systems. The policy implementers who worked with physical access-control systems viewed reliability as a major requirement. Physical keys, key pads and some of the swipe-card systems were selected because they could function autonomously if necessary and they had a low failure rate. However, an insufficiently flexible management interface or the inability to enforce desired policies can be as great a detriment to security and convenience as unexpected failure of the system, e.g., a power outage. Hence, we suggest that these features be given greater consideration when systems are being chosen.

Take advantage of new technologies. New access-control technologies make it possible for access-control systems to achieve a previously unprecedented degree of flexibility and security. Smart-cards, RFID badges, or even software on commercially available mobile phones can all be used to enable access-control systems that make it cheap and convenient to extend access to new users, delegate access on demand and in an ad hoc manner, and yet provide a high degree of auditability and assurance that unauthorized access will not be allowed. These technologies can make it unnecessary, for example, to share keys or key codes, and we believe that adopting them would benefit many organizations.

RELATED WORK

To the authors' knowledge there has been no other study of physical or file access-control policy professionals. Other researchers have studied computer system security professionals in general but have not focused on the specific role of access-control policy management. Barrette et al. studied security professionals who worked in a system administrator role. They found that administrators are very collaborative and work together combining their specialized knowledge to solve problems [3]. Much of the information administrators use is both specific to their organization and exists in many places, requiring administrators to combine the information using custom tools [3, 8].

Few studies examine the challenges of managing a physical access-control system. Bauer et al. interviewed members of a university department prior to the creation of new a physical access-control system. They determined that authority to grant access to resources was passed down the departmental structure with the department head delegating to the building administrator, who delegated to various staff members [5].

Designers have considered the problem of creating tools to assist policy professionals. One example is the SPARCLE Policy Workbench, which allows policy professionals to write privacy policy in natural language and parses the policy and converts it into an implemented policy [9]. The Expandable Grid is another example of a tool which allows users to manipulate the implemented policy for File System rules [17]. These tools are promising but neither are based on actual experience of policy professional issues and tasks.

Other studies have focused on users and how they use security enhancing technologies. Gaw et al. studied the use of PGP for securing email communication. They looked at an environment where security was very important and the employees were motivated to secure communication. They found that employees still did not regularly encrypt their email for various social and convenience issues [13]. Dourish et. al. explored end user's use of security technology. They found that end users tend to delegate security concerns to trusted individuals or groups and trust that their resources are secure. In organizations this sometimes caused a mismatch between the security settings and the current needs of a group [12].

Bauer et al. examined the physical access-control policies created by people who manage policy for their own resources [4]. They found that participants' intended policies were better matched using a flexible access-control system, where policy makers could quickly and easily change the policy, than with a more traditional physical-key-based system. They also found that when policy makers made direct edits to the implemented policy, the policy became less permissive.

CONCLUSION

We interviewed thirteen policy professionals from five organizations in an effort to understand the challenges involved in policy management. We found that policy management had three sets of real-world requirements that were either ignored or not adequately addressed by technology: 1) policies are made/implemented by multiple people, 2) policy makers are distinct from policy implementers, 3) current access-control systems can't always implement the desired policy. Based on our observations, we suggest a number of improvements that could be made to access-control system.

Access-control systems should support easy communication between policy professionals. By encouraging policy implementers to document the policy changes they make, it may be possible to provide vital information for those who will manage the implemented policy in the future.

System designers also need to be aware of the existence of two policy professional roles: policy implementer and policy maker. Each of these roles is associated with a different set of skills, abilities, and tasks. Policy implementers have the ability to make direct changes to the implemented policy. Policy makers have the ability and knowledge to know what changes should be made. Designers of policy-management systems should understand the tasks and limitations of both roles and design to support the differences.

Finally, the capabilities of the enforcement technology itself are important. New technologies make it possible to enforce security policies that older technologies, like keys and key-code locks, cannot. Access-control systems also differ in their policy-management interfaces, some of which are far more flexible and expressive than others. In addition to more typical concerns like the ability of a system to withstand a power outage, these capabilities need to be given careful consideration when selecting an access-control system.

ACKNOWLEDGMENTS

This work was supported in part by National Science Foundation Cyber Trust Grants CNS-0433540 and CNS-0627513, and U.S. Army Research Office contract no. DAAD19-02-1-0389 ("Perpetually Available and Secure Information Systems") to Carnegie Mellon University's CyLab.

REFERENCES

1. M. Abadi. On SDSI's linked local name spaces. *Journal of Computer Security*, 6(1–2):3–21, Oct. 1998.
2. A. W. Appel and E. W. Felten. Proof-carrying authentication. In *Proceedings of the 6th ACM Conference on Computer and Communications Security*, Singapore, Nov. 1999.
3. R. Barrett, E. Kandogan, P. P. Maglio, E. Haber, L. A. Takayama, and M. Prabaker. Field studies of computer system administrators analysis of system management tools and practices. In *CSCW*, 2004.
4. L. Bauer, L. Cranor, R. W. Reeder, M. K. Reiter, and K. Vaniea. A user study of policy creation in a flexible access-control system. In *CHI*, 2008.
5. L. Bauer, S. Garriss, and M. K. Reiter. Distributed proving in access-control systems. In *Proceedings of the 2005 IEEE Symposium on Security & Privacy*, pages 81–95, 2005.
6. A. Beaufour and P. Bonnet. Personal servers as digital keys. In *Proc. 2nd IEEE International Conference of Pervasive Computing and Communications*, Mar. 2004.
7. H. Beyer and K. Holtzblatt. *Contextual Design: Defining customer-centered systems*. Morgan Kaufmann Publishers, 1998.
8. D. Botta, R. Werlinger, A. Gagné, K. Beznosov, L. Iverson, S. Fels, and B. Fisher. Towards understanding IT security professionals and their tools. In *SOUPS*, pages 100–111, 2007.
9. C. A. Brodie, C.-M. Karat, and J. Karat. An empirical study of natural language parsing of privacy policy rules using the sparcle policy workbench. In *SOUPS*, pages 8–19, 2006.
10. D. Cappelli, A. Desai, A. Moore, T. Shimeall, E. Weaver, and B. Willke. Management and Education of the Risk of Insider Threat (MERIT): Mitigating the Risk of Sabotage to Employers' Information, Systems, or Networks. Technical Report CMU/SEI-2006-TN-041, CERT, Software Engineering Institute at Carnegie Mellon University and Cylab, 2007.
11. B. Cleary. Employee role changes and socgen: Good lessons from a bad example, April 2008. http://www.scmagazineus.com/Employee-Role-Changes-and-SocGen-Good-lessons-from-a-bad-example/article/108541/.
12. P. Dourish, E. Grinter, J. D. de la Flor, and M. Joseph. Security in the wild: user strategies for managing security as an everyday, practical problem. *Personal Ubiquitous Comput.*, 8(6):391–401, 2004.
13. S. Gaw, E. W. Felten, and P. Fernandez-Kelly. Secrecy, flagging, and paranoia: adoption criteria in encrypted email. In *CHI*, pages 591–600, 2006.
14. E. M. Huang and K. N. Truong. Breaking the disposable technology paradigm: opportunities for sustainable interaction design for mobile phones. In *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 323–332, New York, NY, USA, 2008. ACM.
15. N. Li and J. C. Mitchell. Understanding SPKI/SDSI using first-order logic. *International Journal of Information Security*, 2004.
16. M. R. Randazzo, M. Keeney, E. Kowalski, D. Cappelli, and A. Moore. Insider thread study: Illicit cyber activity in the banking and finance sector. Technical report, Carnegie Mellon University Software Engineering Institute, 2005.
17. R. W. Reeder, L. Bauer, L. F. Cranor, M. K. Reiter, K. Bacon, K. How, and H. Strong. Expandable grids for visualizing and authoring computer security policies. In *CHI*, pages 1473–1482, 2008.
18. J. Saltzer and M. Schroeder. The protection of information in computer systems. *IEEE, Proceedings*, 63:1278–1308, 1975.
19. A. Woodruff, S. Augustin, and B. Foucault. Sabbath day home automation: "it's like mixing technology and religion". In *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 527–536, New York, NY, USA, 2007. ACM.