Social Navigation as a Model for Usable Security

Paul DiGioia and Paul Dourish
Donald Bren School of Information and Computer Sciences
University of California, Irvine
Irvine, CA 92697-3425
{ pdigioia, jpd } @ics.uci.edu

ABSTRACT
As interest in usable security spreads, the use of visual approaches in which the functioning of a distributed system is made visually available to end users is an approach that a number of researchers have examined. In this paper, we discuss the use of the social navigation paradigm as a way of organizing visual displays of system action. Drawing on a previous study of security in the KaZaa peer to peer system, we present some examples of the ways in which social navigation can be incorporated in support of usable security.

INTRODUCTION
Security has always been a critical concern for information systems, but the rapid rise of the Internet as a site for everyday activity has made it a particularly pressing concern lately. The Internet is a major means for consumer commerce, for individual banking, and for participation in civic life (e.g. in the form of early experiments with online voting in the 2004 US Presidential Election.) As the daily use of the Internet has increased, so has its attractiveness to attackers.

Bruce Schneier [2000] has observed that "Security measures that aren't understood and agreed to by everyone don't work." Security as a concern for end users, then, has become an increasingly important topic of research interest. A number of perspectives in this work have emerged.

One research approach has focused on the critical examination of the usability of security mechanisms available in current networked systems. Whitten and Tygar's [1999] study of the usability of PGP for secure electronic mail is perhaps the quintessential example of this approach, applying traditional usability analysis techniques to the technologies of security. Analyses of these sorts have uncovered a range of problems with the ways in which security technologies have been "grafted on" to applications and infrastructures, and also demonstrated the considerable knowledge of security technologies that they seem to require on the parts of their users.

A second research approach has been to create new mechanisms designed to replace existing security facilities while providing greater usability (and, by implication, greater security.) For example, the use of "passfaces" rather than passwords is designed to allow authentication mechanisms that are less likely to be forgotten and less susceptible to attack [Brostoff and Sasse, 2000].

A third approach has been to step back from the specific problems of current security mechanisms and to examine security as a facet of interaction more broadly. This is the approach that we wish to examine here. Empirical work [e.g. Weirich and Sasse, 2002; Dourish et al., 2004] has looked at security as a practical concern and examined the ways in which people go about working securely, while design activities [e.g. dePaula et al., 2005] have examined new approaches in which security is understood not simply as a set of features to be included in a software system, but rather as a pervasive aspect of its design. The central concern here is that neither usability nor security can be added on to systems after the primary design work is done; rather, both need to be central aspects of the design effort.

In our own work, we have been especially concerned with the use of visualization-based approaches to security [Dourish and Redmiles, 2002; DePaula et al., 2005]. In particular, we have argued that the central problem of security for most users is to match the settings within which they find themselves to an immediate set of needs and practical concerns. Quite what "secure" means at any given moment is a determination that only an end user can make. Attempts to make systems inherently secure, then, are problematic because they presuppose what "secure" might be, taking that decision out of the users' hands; and attempts to incorporate "transparent" security into a system are equally problematic because they make it impossible for users to determine whether and how a system is secure. Our approach, then, has been to explore the use of dynamically coupled visualizations of system activity that can make aspects of a system's internal operation visible and

examinable. We read "system" broadly here; our concern is not simply with a particular computer, but with the collective functioning of a range of components that together make up the "system" at any given moment. Security is an end-to-end phenomenon, and so too must our visualization strategy.

Beyond this broad concern for visualization as an approach to security, little has been said about particular approaches or design techniques. In this paper, we want to explore one particular paradigm for visual security interfaces, that of social navigation.

SOCIAL NAVIGATION

Social navigation is an approach to interaction design initially presented by Dourish and Chalmers [1994]. It has since been explored by a range of researchers and incorporated into a wide range of systems (see [Hook et al 2003] for an overview.)

The essential observation behind the social navigation approach is that, in the everyday world, we navigate complicated spaces by making use not only of their spatial organization but also our understandings and interpretations of the activities of others. For example, a worn path across a field of grass shows us where others have walked in the past, and so can help guide us towards points of interest. In this way, we encounter a space not only in terms of its own structure, but also as a space that has been occupied by others in the past, whose behaviors might be cues to us that allow us to organize our own activity. This may be past or present activity. Another common example of social navigation, for instance, is the case of walking down a street past a number of cafes; the presence of absence of people in the cafes displays which places are popular and which are not, which might help us make a selection amongst alternatives (or similarly, seeing who the people are who frequent each café might help us select one that is close to our own particular tastes.)

It is worth noting, in both of these real-world examples, that the activity of others is in no way a constraint on our own, but only a cue to decision-making. It may be that we seek solitude, and would rather be in a quiet café, or away from the more occupied spots, and so we might choose, say, to leave the path in order to get away from other people. That is, the social navigation approach is not, in general, concerned with a normative distinction between good and bad, but rather about the ways in which we can understand a space in terms of the activities of others.

Although Dourish and Chalmers' original concern was to distinguish between spatial, semantic, and social navigation in collaborative systems, they noted its application in traditional interactive applications. In particular, they drew upon two important examples of the existing use of this basic approach. One of these was the Tapestry system [Terry et al., 1993], an early example of collaborative filtering, in which people could vote on the usefulness of email messages and news articles, as a way of helping others deal with large volumes of information. The other, more relevant to our discussion here, was Hill et al's [1992] notion of "edit wear and read wear." The notion of "wear" here is that of "wear and tear," or something that is "worn away," a form of digital erosion. Hill et al. describe an interaction approach in which activity over an artifact leaves traces on the artifact itself, so that, for example, the scroll bar of a document might have markings that indicate which parts of the document have been read most frequently ("read wear") or edited most often ("edit wear.") Clearly, this technique generalizes in a range of ways, but the central concept – that an artifact can display the accumulated pattern of activities that have been performed over it – is a key element of the social navigation approach.

Perhaps the most common and most prominent application of social navigation has been in recommender systems, particularly those associated with electronic commerce sites such as Amazon.com. In these cases, people are matched through a comparison of profiles derived from their purchase histories, and these comparisons are used to recommend related products. However, the sense of social navigation that we want to explore here is one closer to the approach of Hill et al.

SOCIAL NAVIGATION AND SECURITY

It would be a stretch to think of security as a navigation task, but we can take the social navigation approach broadly to suggest that we think of applications as "spaces" in which multiple people may act, and that the history of their actions might be displayed in those spaces [Dourish, 2000]. In other words, as in the Hill et al model, artifacts accumulate and display aspects of the history of actions over them.

When we think of social navigation in these terms, then the opportunity to use this model in a security context becomes clear. Our particular concern is with allowing people to assess how a system matches their needs, and one critical aspect of that is to allow them to see the relationship between a system or information artifact and activities, either their own or others.

There are at least three ways in which this fundamental approach can be used.

First, we can use social navigation to show *the history of a user's action*. That is, as users act within a system, we can use the artifacts of that system to show the history of the users's actions – paths followed, objects used, and so forth. This is the most direct application of Hill et al's ideas to the security domain.

Second, we can use social navigation to show *patterns of conventional use*, and therefore to show deviation from them. This is related to the first approach, but in this second approach, we attempt to form generalizations of user activity and, rather than presenting the cumulative history of activity, we attempt to determine and therefore display

"usual" patterns. The central issue here is the ability to be able to highlight deviations form normal routines, e.g. the use of different servers than usual, communication with unusual other parties, etc. Again, as in the traditional social navigation approach, this is not intended to designate certain activities as inappropriate or disallowed, but rather to provide a context within which user actions can be assessed.

Third, we can use social navigation to show *the activities of others within a system*. That is, for systems in which objects are in some sense shared, then we can use those objects to display a history of others' activity. This is a way of making the activity of users visible even in an application in which those users are not themselves visible. While this is most directly applicable in cases where data objects are explicitly shared, there are other ways to apply it. Elsewhere, we suggested that this fundamental idea – that the patterns of others' activities can be presented to me as a context for my own – can be applied even in cases where "sharing" of information objects is not a fundamental feature of interaction – e.g. the configuration of network settings and other system and application preferences [Dourish, 2000].

As a way to both explore and convey these ideas, we will focus for the rest of this paper on an extended design example. Our example is inspired by Good and Krekelberg's [2003] study of potential security issues in the KaZaa peer to peer filesharing application. This study differs from that of, say, Whitten and Tygar [1999] in that, while it is a study of the interactions between usability and security, it is not focused on specific "security" features or components. Instead, it considers security more holistically within the design. The study comprises, essentially, three components.

The first is a brief empirical examination of the Kazaa network which reveals a significant number of files which, it could reasonably be surmised, were not intended to be shared. These include people's email inbox files, spreadsheets of credit card information and financial data, web browser caches, cookie files, etc. While it is not possible to be certain that these files had been shared unintentionally, it is certainly a plausible speculation. The second component of their study was a cognitive walkthrough of the Kazaa interface, which revealed a number of problems that could potentially lead users to misconfigure the system so that they shared more information than they intended or indeed than they realized. This walkthrough also suggested that it was difficult for people to determine the extent of sharing. The followed up on the cognitive walkthrough by a third study component, an empirical laboratory study of Kazaa use designed to determine how easy it might be for someone to misconfigure Kazaa by accident. Their results confirmed the implications of the cognitive walkthrough; only two out of twelve users were able to determine correctly which files were being shared.

The problems that Good and Krekelberg point to are particularly problems about the visibility and consequences of action. Accordingly, we have found it a fruitful example to which we can apply ideas about the use of social navigation. We have focused in particular on the second and third strategies – representing conventional patterns, and disclosing the activities of others.

PATTERNS OF CONVENTIONAL USE

Essential to the second idea of social navigation is the ability to become aware of previous actions by a group of other users. Here, we describe our designs which show patterns of conventional use, in the hopes that the information presented allows the user to make informed decisions. For any decision required of a user by a system, it is likely that a number of users were previously forced to make the same decision. It would therefore be helpful for others to be aware of the choices made by previous users, to aid in the present decision-making process.

Using a Folder Metaphor to Support Social Navigation

The initial setup phase of the Kazaa peer-to-peer application requires a great deal of decisions to be made by the user. Specifically, the user must determine which files on their (formerly private) hard drive should be made available for sharing to others on the Kazaa network. There is an inherent tradeoff between security of personal data, and sharing of personal intellectual property or art – making it difficult for many to decide whether certain files should be shared. Further complicating the decision-making process is the clumsiness of the Kazaa user interface. Good and Krekelberg suggest that users are generally unable to determine which files on their system were currently being shared, due in large part to the awkward treatment of folders by the Kazaa interface.

Additionally, we argue, users flounder because there is no method to compare one's sharing level against that of other users. Lacking this social information, each new user is forced to make this decision anew, on his own. In light of Hill et al, it would be desirable to indicate whether or not a folder is commonly shared by others. Users should be able to make an informed decision, based on the information left behind by those who have already made the decision. This would allow the user to get a sense of whether his choice of folders is within the bounds of what other users of the system have shared.

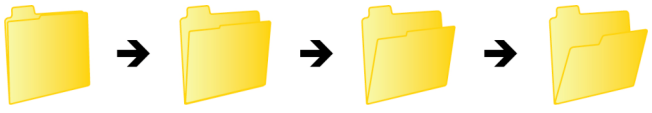

Figure 1. The visual representations of folders. Progression is shown from "least shared" to "most commonly shared."

Design

Our design allows folders to exhibit how frequently they are shared by other users. In this model, folders retain their

use in the file-structure as containers of files and (optionally) other folders. For the purposes of the Kazaa application, the folders have the additional responsibility of displaying this social navigation information, which is reflected in the very icon of that folder. A commonly-shared folder – *My Music,* for instance – itself reveals the fact that it is commonly-shared. Specifically, the frequency with which other Kazaa users have shared a certain folder is analogous to how open the folder icon appears. The degree of how 'open' others have been to the idea of sharing certain folder is reflected in how 'open' the representative folder icon appears. The more "closed" a folder appears, the less commonly it is shared [Figure 1]. This concept is then integrated into the standard "Folder Selection" dialog, within which the user selects which folders to share globally [Figure 2].

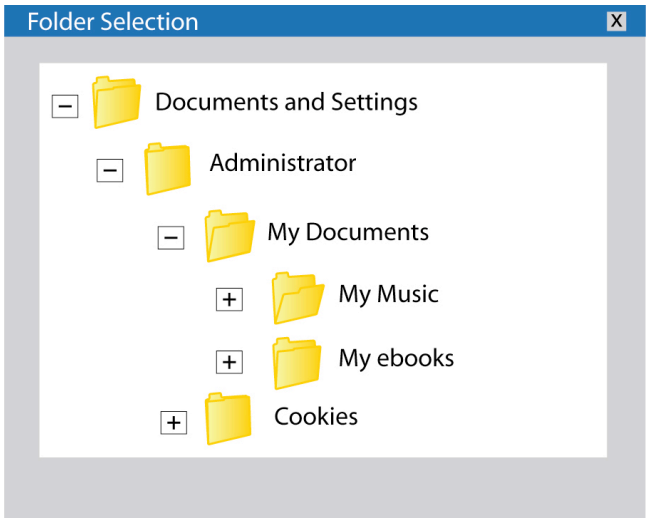

Figure 2. The Folder Selection dialog. Representative icons reflect how generous others have been in sharing each folder. In this example, the *My Music* folder is usually shared by other users, while the *Cookies* folder is not.

It is clear that this scheme gathers much more longitudinal information on use when dealing with folders created by the operating system; user-created folders present an interesting situation. This is, of course, because there is be less historical information available on folders titled "baseball_stats_2001" than on "My Documents." If a user discovers that it is rare to share one's "My Documents" folder, it is probably safe to assume that he should follow suit. Given that everyone *has* a My Documents folder, coupled with the fact that the visualization shows that it is not a popular folder to share, implies that a majority of users specifically choose *not* to share this folder. However, in the case of a more customized folder, such as "baseball_stats_2001," the low degree of sharing may indicate that information of this nature is *rare*, rather than *risqué*. This may have the opposite effect; it may, for instance, *encourage* the sharing of this folder. It should be reiterated here that the information displayed using social navigation ideas are not intended to necessarily *force* the user to make a specific decision. Rather, it is our intention that social navigation information simply *be available* to the user; it should be *interpreted* however the user sees fit.

We feel that the simple, non-obtrusive nature of this design is in some ways beneficial. It requires very little alteration of the existing interface, and its information storage requirement is modest. Additionally, it utilizes the already pervasive paradigm of *folders* to represent groups of files – users would not need to substantially change their methods of thinking about file structures in order to benefit from this information. For the same reasons, however, we feel that the impact of this idea is limited. We also propose a second, more novel mechanism for displaying this historical information.

Using the Pile Metaphor to Support Social Navigation

In a study of alternative desktop interfaces, Mander et al [1992] proposed a pile metaphor for the informal grouping of files. 'Piling' places less cognitive load on the user, and is much more natural to people for ad-hoc organization than 'filing.' Typically, people form physical stacks of media (we use the term 'media,' since piles are not limited to paper – they may contain CD's, video tapes, hardcover books, etc.) more often than people take the time to file them correctly. Studies of office work and organization have also drawn attention to the relationship between working activities and physical arrangements, and the ability of physical arrangements to convey important information about work state [Malone, 1983; Kirsch, 1995]. We feel that we can couple this pile metaphor with our ideas of social navigation, producing a dynamic information space within which a user has a new form of control over the shared media, and makes sharing decisions based on social navigation cues.

Summary & Rationale

To more directly address the peer-to-peer usability guidelines established by Good and Krekelberg – which deals with the initial selection of files to be made publicly-accessible, and which should remain private – we propose a pile-driven preview visualization. Using piles as a visual representation of digital files, the user is able to get a quick overview of the files selected for sharing. Any discrepancies between his *intended* sharing level and his *actual* sharing level quickly come to light.

The pile-driven preview function would compliment the folder selection dialogs of the Kazaa application. The folders currently selected for sharing are displayed as they are in the current Kazaa application, with the aforementioned 'open folders' design modification. This paper proposes an additional view in tandem to the folder selection dialog, to allow the user to get a sense of the *files* that have been flagged for sharing as each folder is selected.

This additional preview would show the currently-flagged files grouped into piles. Additionally, the images representing the files appear visually different, depending on the file's type (e.g., a photograph, a document, or a song). Even if the grouping mechanism used to separate the files into distinct piles is entirely random, the user will nonetheless become aware of the following important properties:

- The sheer number of files going to be shared. A plethora of large piles may indicate that a user is sharing too many files; a small number of short piles may be a cue that a user has his sharing under control.
- The true repercussions of selecting a folder to share. In the existing Kazaa interface, users merely observe a check-mark adjacent to a folder icon – hardly enough information to get a true sense of how many files have been selected, and definitely confusing when nested folders are involved. In our design, the user notices files literally 'piling up.' When the user detects that files from sub-folders are also added to the visualization, he is instantly aware of the implications of his action, and may think better of his selection.
- The *types* of files being selected for sharing. Users are able to distinguish between the different types of media being added to the publicly-available shared space. Since our design renders the visual representations of files differently based on their file type, the user would be able to visually (and therefore quickly) distinguish between, say, an Excel spreadsheet and a music file. Mander proposed a similar file-type differentiation scheme, although this was based solely on color. By generalizing files with visual, metaphorical representations (rather than mere colors) using file types, our design provides additional information for use in the determination of the proper files to share. Operating systems such as Microsoft Windows and Mac OS X currently achieve similar results, albeit in a direct line-of-sight manner; icons representing images are visibly different than text file icons. However, these examples require the user to view the icons straight-on. For our design, we instead utilize an informal yet fairly accurate representation of different file types based on their *edges*. Mander et al. [1992] has shown that by merely "looking at [a] pile's outside form, [subjects] were able to infer quite a lot about its contents." Given Mander et al's findings – based on piles in their physical form – this edge visualization does not sacrifice a user's understanding of the correlation between a certain file and its representation.

Applying further structure and additional visual cues to this model would result in an even greater understanding of the Kazaa interface. In addition to allowing users to group files into piles using the familiar drag-and-drop method, our design model also consists of a function to allow "piling by" content, file type, or other criteria – in much the same way that Mander's proposed metaphor allows. In viewing the various piles created by the system, users have an easier time identifying inappropriate files that *would* have been shared under the existing Kazaa interface.

Mander's model hints at the ability of sub-piles to be moved in and out of the visualization area. In our model, piles deemed inappropriate for sharing are moved out of the visualization area to prevent sharing with other Kazaa users. Should an inappropriate file be identified in the existing Kazaa interface as tagged for sharing (which is itself a difficult task – the user can only see the folders chosen, not the files), the user must de-select the folder within which the folder resides. There is no method to selectively un-share a single file, without removing the entire folder from the shared space. In our design model, this process is greatly simplified – the user may simply drag a file out of the shared space. If, instead, the user determines that an entire pile is inappropriate for public sharing, the user may also select the pile itself and drag it out of the shared space.

Design

Visualizations of these ideas can be accomplished in many different ways. One design may pull from familiar, physical metaphors. Files that have been selected for sharing in the folder selection dialog [Figure 2] appear in the shared space, which may be represented visually by a desk. The question, "what do you bring to the table?" applies figuratively to this design. Piles may be created, either by the user, or automatically by the system. Piles (or sub-piles, or individual files) can be selectively dragged out of the shared space, and into a safe haven, which might be represented visually by a filing cabinet. The filing cabinet is used as a metaphor for the safekeeping of piles, since files are generally considered to be "put away" or "in their place" when stored in a filing cabinet. This design example seeks to make use of the physical distinction of storing items out of sight for safekeeping, and laying out items on a table, for review by others.

Selectively un-sharing a single file or group of files is unacceptably difficult in the existing Kazaa interface, according to Good and Krekelberg. Its adherence to strict folder-ing schemes renders the task of selectively un-sharing a subset of files next to impossible. Folder schemes cannot be dynamically changed – it usually requires a large amount of time and cognitive load to significantly change one's filing method. Since our pile metaphor design would *compliment* the underlying file-structure, it would simplify the selection of multiple files which are *related* but do not necessarily reside within the same folder. File-folder

structures would remain in-tact at the file system level, but the task of grouping and selecting piles from various folders would be greatly expedited.

Good et al claim that "users should be made clearly aware of what files are being offered for others to download." In this respect, we feel that our pile metaphor succeeds. However, we argue that this is not enough. Even after becoming aware of which files are currently *being* shared, users should get a better sense of which files they *should* be sharing. Using social navigation techniques, we believe this is possible.

Expanding the Design to Support Social Navigation

We have shown that our design model forms piles using different criteria, so that piles *with these attributes* or *containing these types of files* are visually distinguished. We have also mentioned tabletop and file-cabinet metaphors for the visual representation of the shared space and private space, respectively. We now expand these design ideas to incorporate information regarding how frequently *other users* are willing to share piles with similar attributes. We couple this information within the visualizations of the piles.

As with the "open folder" metaphor, the intention here is to communicate to the user the frequency with which other users chose to share similar piles. Again, we rely on the fact that this decision – whether or not one should share a particular pile (a group of files with a specific set of properties) – has been made by previous users. Integrating this information with the visual space would supply cues which may be used to determine one's *own* sharing decisions.

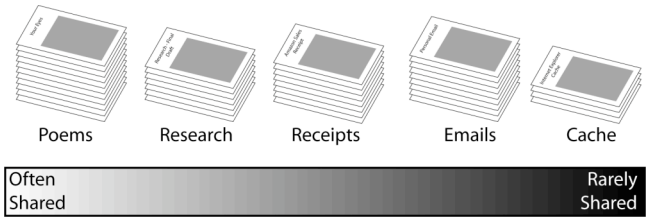

Figure 3. Piles plotted linearly, by percentage of users who share related piles.

Our design utilizes spatial relationships between the piles to encapsulate the notions of commonly-shared piles, rarely-shared piles, and various states between these two extremes. We incorporate the social navigation information in the *arrangement* of the piles. The design plots the "riskiness" associated with sharing a group of files linearly – resembling the x-axis of a Cartesian plane. [Figure 3]. The greater the "risk index" associated with a pile, the further the pile appears from the Cartesian "origin." The "risk index" for any given group of files is defined informally as the number of users who have chosen *not* to share this group of files (with some specific properties), divided by the total number of Kazaa users. Optionally, the "risk index" may be calculated using the total number of users who *have* a similar group of files, rather than *all* users. This latter calculation accounts for the fact that not all users make their file selections using a standard set of files. In doing so, this alternative calculation gives us the percentage of users who chose not to share a certain pile, *but had the option of choosing* such a pile.

This notion of "usually shared" and "rarely shared" groups of piles can be expressed visually in a number of ways. For example, in an example grounded in a strongly metaphorical ineraction style, a desk is used to symbolize the shared space; items placed in a nearby filing cabinet are removed from the shared space. We represent this new social navigation information in the layout of the piles on the table. The closer a pile is located to the file cabinet, the more likely others have been to keeping it filed away in *their* filing cabinets [Figure 4].

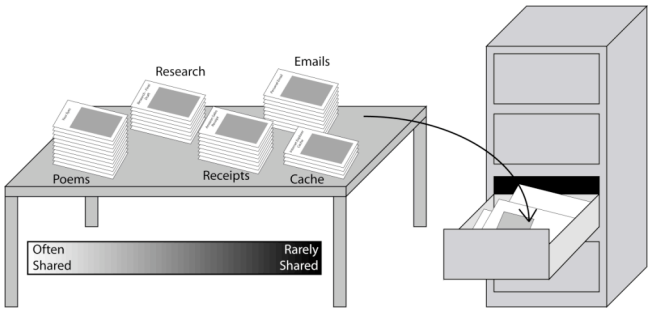

Figure 4. Selection of piles to be shared. These piles have been neatly grouped by keyword. Social Navigation information is expressed laterally. Users remove piles (or individual files) by dragging them into the file cabinet.

Although the Hill et al proposition of digital wear is not central to this visualization, the more general notion of leaving a trace *is* indeed a part of the design. Users still produce an artifact of their decision – albeit in an indirect, statistical manner. In the following section, we present a design which more directly applies the notion of digital wear in order to address the third idea of social navigation.

ACTIVITIES OF OTHER P2P USERS

The third idea of social navigation deals with the awareness of previous activity by other users. Whereas previously the notion of "other users" was taken as a collection of users acting as a group, here the term refers to other entities acting independently of one another. Collectively, users can be said to "typically" act a certain way, while no doubt some users will – as individuals – act completely opposite. We here describe a design which encapsulates this notion of *individual actions* as performed on a user's shared files.

Using Piles to Show the Activities of Other Users

The aforementioned pile mechanism is aimed primarily at solving the folder-selection problem which arises only

during the initial setup of the peer-to-peer application. In addressing the third idea of social navigation – awareness of the actions of others – we present a design for the display of currently-shared files in the Kazaa application, which also draws from the pile metaphor. This design allows the addition and removal of files using the efficient piles method. We also introduce social navigation techniques to make this pile-management design more informative to the user.

Rationale

Firstly, grouping of the currently-shared files would enable the user to gain an overall idea of the files currently being shared. This improved visualization would enable the user to easily determine whether a file should or should not be shared. For instance: if, after "piling by content," a "financial" pile is produced, the user can spot this, and may decide to remove the entire pile from the shared space. In this single action, *all* of the files are tagged as "unshared" in the Kazaa interface, and removed from the shared space – *regardless* of their location within the underlying file system. This solves the problem of manually un-sharing each file – an issue which pervades Good and Krekelberg's study. We use this improved visualization – the pile metaphor for the displaying of currently-shared files – as a foundation for our next design.

We also draw on the concepts presented in the "edit wear" model. Digital wear mechanisms have been proposed not as attention-demanding signals, but rather as subtle indications of previous use. This approach to the display of social navigation information presumes that the user would be interested in this data, and might choose to make decisions based on this supplemental information. However, the case would rarely (if ever) arise where the information supplied by social navigation techniques would enable 'the system' to make the decision on behalf of the user. One would not likely follow the strict rule 'eat only at the more popular restaurant' each and every time. Social navigation (and, specifically, digital wear) information is intended to *aid* in the user's decision-making process. We use this idea of digital erosion – as an integrated, inherently human feature of items – in applying the pile model.

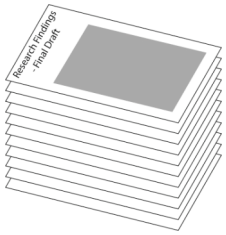

Figure 5. A sample pile. Since this pile has been created by the system, it appears tidy.

Design

Mander et al suggest differentiating the "neatness" of piles: files which have been "piled" by the system appear in neat stacks; piles arranged informally by the user appear disheveled. We choose to modify this distinction in neatness to encapsulate the idea of digital wear. Groups of files currently shared within the Kazaa application appear as miniature representative pile icons, as in [Figure 5]. Users may informally group these files themselves, by dragging and dropping files onto one another. Additionally, the user may rely on the system to aid in the construction of piles (piling by keyword or file type, for instance). Upon the initial formation of a pile (using either method), it appears as a neatly-stacked tower of files. A pile containing fictitious research documents is formed in [Figure 6, a]. This is meant to resemble its physical counterpart; new piles are usually tapped on an edge to "square up" the pile.

When another Kazaa user views or downloads a file within a pile, the structure of the pile changes slightly to reflect this access. First, the file's icon moves to the top of the stack. Additionally, the newly-accessed file is rendered slightly out of line with the rest of the stack. The file "Research Grant Proposal" is searched and downloaded – and its movement within the pile is displayed – in [Figure 6, b]. Each time a file (or an entire pile) is perused, its corresponding pile exhibits these changes in appearance. Slowly, the pile degrades from its original, rigid form as a neatly-formed tower. The end result – after multiple downloads of different files by different users – is an untidy pile, ordered by frequency of access [Figure 6, c]. Again, this is meant to resemble the activity of piles in their physical form; after a great deal of use, a pile will eventually degenerate into a more disheveled state – and its order is largely determined by the manner in which documents are picked out and placed back on the stack.

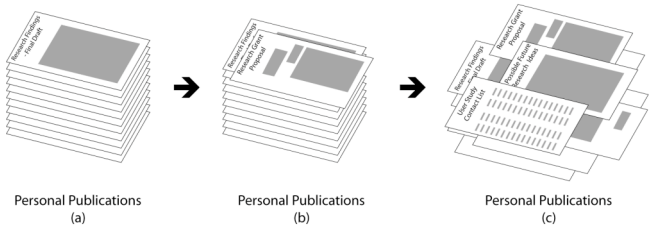

Figure 6. The visual encapsulation of social navigation information within piles. The piles progress from "neatly piled" to "disheveled" as the pile is accessed by more users.

This design retains the tampering information inherent in physical piles. Users are easily able to differentiate between their most frequently accessed piles, and their more underutilized piles. Additionally, users may instantly determine the most popular files within each pile. One may wish to investigate, to determine why a particular file might be so popular. In our illustrated example, it may be useful to know that Kazaa users are more interested in the phone numbers of one's user study participants than the actual

research paper [Figure 6, c]. Users can then use these cues in their decision to continue sharing the files, or to remove the files from the shared space. Conversely, users may (and often do) want some of their files to be popular with other Kazaa users. In the physical world, it may be beneficial for a worker to know that his colleague has, for instance, 'finally looked through those files.' In the digital application of this model, knowing that other Kazaa users are interested in your poems may be inspiring. The goal of our design work is to capture this information – whether an indication of popularity or wrongdoing.

Stealthily perusing piles in the physical world is an action we can all perform, but covering one's tracks is not a trivial task. One may lift the corners of the stack and peek inside, replace a pile after photocopying its contents, or simply re-square the pile to conceal the fact that it had been tampered with. It is particularly curious (and, we argue, counter-intuitive) that the Kazaa interface *encourages* the erasure of any knowledge of activities performed on any user's files. Pilferers have an easy time covering their tracks. In fact, *no* tracks are left – there isn't even a log file. The only possible way to monitor activity would be to continually check the Upload/Download window. However, this would be akin to constantly eyeing the piles on one's desk. You simply can't. (One could set up video surveillance, but video is again an *archive* which can be rewound. Logs would be the equivalent of video surveillance.) Users should *not* be expected to watch the Upload/Download window pane constantly for this knowledge, just as physical piles should not expect their owner to constantly watch them. Instead, people use the activity information inherent in the structural organization of their piles to cue them whether "someone has been picking at my porridge." In life, it is more difficult to hide one's tracks than to leave them. In p2p systems, it is all but impossible to view the activity of others. This completely opposes the physical world, and we feel that this obscurity has no grounds. It is possible that this design decision was made to cut down on legal issues involved in the trading of copyrighted information; with no record of wrongdoing, the investigation of illegal file-trading is more difficult. However, this is beyond the scope of this paper.

DISCUSSION

As we noted, our work so far has been limited to design sketches, for lo-fi prototyping; further prototypes are under development based on these ideas, as a basis for initial user testing and validation. However, although we have been concerned over the past several pages with design concerns, the topic of this paper is not the design specifics themselves, but rather the approach that they exemplify.

The social navigation approach is characterized here by two properties. First, we think of the application as a space populated by users; and second, we apply the principle that artifacts carry the evidence of activities over them.

This pair of principles seems to apply particularly well in some set of security applications. Thinking of applications as spaces that might be populated by people immediately turns our attention towards the ways in which our own activities and artifacts might be visible to others, and to the ways in which others might come to be aware of information that we are generating, storing, etc. In other words, if one of the central problems of security in information systems is that invisibility of potential attackers (or even the inability to distinguish between friends and foes), then a model of an application as an information space in which others might be seen and encountered seems to be a fruitful one; it places others, be they attackers or colleagues, in the forefront of the user experience. Similarly, the idea that the primary way in which a user might become aware of the presence of others or of their recent activity is through the evidence of activities carried by the artifacts over which they have acted (and over which the user himself may act) is also useful, in at least two ways. First, it places the necessary information within the existing "interaction frame;" that is, it does not create some extra window, log file, or panel to check in order to become aware of other's actions, but places it directly within the view through which users interact to accomplish their work. This means that one might become aware of relevant information directly; there is no need to take a special action in order to come across it. Second, it provides a route whereby security-relevant information can be easily incorporated into existing applications and interfaces, since it augmented rather than replacing object- or artifact-centered interaction designs.

CONCLUSION

In discussions of the problems of usable security, one persistent consideration has been that usable security technologies may be ones that more successfully incorporate the user into the determination of security rather than taking decisions away from the users. One reason for this is that the precise requirements for security, and even a determination of what counts as secure or insecure, are things that only end users may be capable of determining at any given moment [Palen and Dourish, 2003]. This has led a number of researchers to consider how systems can more accurately and completely inform users of the potential consequences of their actions, in order to allow them to make informed decisions about privacy and security.

We have suggested that social navigation may be a useful approach here. Social navigation is an approach to interactive system design that originated in considerations of the relationship between individual and collaborative work, and the ways in which, in the everyday world, we interpret spaces as being inhabited by others from whose activities we might learn. Social navigation systems attempt to make the action of others available to users as a basis for thinking about their own action. Thinking about networked systems as populated spaces leads immediately to a range of considerations of how it is that others actions may be incorporated into the interactive experience as a basis for informed decision making. One particularly interesting

issue here is the extent to which we might be able to use social navigation approaches to visualize aspects of the behavior both of users and of others; by presenting them within the same frame, we can both contextualize a user's action with respect to others and conventional behaviors, and also help users develop a sense of the ways in which they might be seen by others through their own actions. Both of these are critical issues for usable security systems.

In our own current work, we are exploring this approach as a part of a broader investigation of the use of visualization technologies as foundational elements of usable security.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under awards 0133749, 0205724 and 0326105, and by a grant from Intel Corp. We would like to thank Sharon Ding, Gai Inoue, Charlotte Lee, Jack Muramatsu, Jennifer Rode and Norman Su for discussions around an earlier presentation of these ideas.

REFERENCES

1. Brostoff, S. and Sasse, A. (2000.) Are Passfaces More Usable than Passwords? A Field Trial Investigation. Proc. HCI'2000.
2. DePaula, R., Ding, X., Dourish, P., Nies, K., Pillet, B., Redmiles, D., Ren, J., Rode, J. and Silva Filho, R. (2005). In the Eye of the Beholder: A Visualization-based Approach to Information System Security. Technical Report (under review for publication).
3. Dhamija, R. and Perrig, A. (2000) Déjà vu: A user study using images for authentication. In Proc. 9th USENIX Security Symposium (Denver, CO, USA, Aug. 2000).
4. Dourish, P. (2000). Towards an Infrastructure for Pervasive Recommendations. Position paper for the CHI 2000 Workshop on Social Navigation: A Design Approach.
5. Dourish, P. and Chalmers, M. (1994). Running Out of Space: Models of Information Navigation. Short paper presented at HCI'94 (Glasgow, UK).
6. Dourish, P. and Redmiles, D. (2002). An Approach to Usable Security Based on Event Monitoring and Visualization. Proceedings of the New Security Paradigms Workshop 2002 (Virginia Beach, VA).
7. Good, N. and Krekelberg, A. (2003). Usability and Privacy: A Study of Kazaa File Sharing. Proceedings of the ACM Conference on Human Factors in Computing Systems CHI 2003 (Fort Lauderdale, FL), 137-145. New York: ACM.
8. Hill, W., Hollan, J., Wroblewski, D., and McCandless, J. (1992). Edit Wear and Read Wear. Proc. ACM Conf. Human Factors in Computing Systems CHI'92 (Monterey, CA), 3-9. New York: ACM.
9. Hook, K., Benyon, D., and Munro, A. (2003). Designing Information Systems: The Social Navigation Approach. Springer.
10. Kirsh, D. (1995). The Intelligent use of Space. Artificial Intelligence, 73(1-2), 31-68.
11. Malone, T. (1983). How Do People Organize Their Desks? Implications for the Design of Office Information Systems. ACM Trans. Office Information Systems, 1(1), 99-112.
12. Mander, R., Salomon, G., and Wong, Y. (1992). A 'Pile' Metaphor for Supporting Casual Organization of Information. Proc. ACM Conf. Human Factors in Computing Systems CHI'92 (Monterey, CA), 627-634. New York: ACM.
13. Palen, L. and Dourish, P. (2003). Unpacking "Privacy" for a Networked World. Proceedings of the ACM Conference on Human Factors in Computing Systems CHI 2003 (Fort Lauderdale, FL), 129-136. New York: ACM.
14. Schneier, B. (2000). Secrets and Lies: Digital Security in a Networked World. Wiley.
15. Terry, D. (1993). A Tour Through Tapestry. Proc. ACM Conf. Organizational Computing Systems (Milpetas, CA), 21-30. New York: ACM Press.
16. Weirich, D. and Sasse, M.A. (2001). Pretty Good Persuasion: A first step towards effective password security for the Real World. Proceedings of the New Security Paradigms Workshop 2001 (Sept. 10-13, Cloudcroft, NM), 137-143. ACM Press.
17. Whitten, A. and Tygar, D. (1999.) Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0. Proc. 8th Usenix Security Symposium.