Empowering Ordinary Consumers to Securely Configure Their Mobile Devices and Wireless Networks

Cynthia Kuo Vincent Goh Adrian Tang Adrian Perrig Jesse Walker

December 7, 2005
CMU-CyLab-05-005

CyLab
Carnegie Mellon University
Pittsburgh, PA 15213

Empowering Ordinary Consumers to Securely Configure their Mobile Devices and Wireless Networks

Cynthia Kuo — *Carnegie Mellon University* — *cykuo@cmu.edu*

Vincent Goh — *Stanford University* — *vgoh@stanford.edu*

Adrian Tang — *Carnegie Mellon University* — *jangace@cmu.edu*

Adrian Perrig — *Carnegie Mellon University* — *perrig@cmu.edu*

Jesse Walker — *Intel Corporation* — *jesse.walker@intel.com*

First Draft: May 8, 2005
Updated: December 6, 2005

Abstract

Despite the best efforts of application designers, security configuration interfaces are hard to use. The conventional wisdom for designing consumer applications does not work for designing security applications. Using 802.11 networks as a case study, we present a set of principles for the design of configuration interfaces. The key insight is that users have a difficult time translating their goals for wireless network security into specific feature configurations.

We design and implement a configuration interface that guides users through an 802.11 wireless network configuration. We overcome users' configuration difficulties by automating the translation from high-level goals to low-level feature configurations. The design empowers non-expert users to securely configure their networks as well as expert users. We also design and conduct a user study which demonstrates that users perform dramatically better using our prototype, as compared with the two most popular commercial access points. In general, our research addresses problems that are common across mobile system configurations.

1 Introduction

For home consumers, the setup and configuration of new technologies is a daunting experience. Configuration is generally a difficult task, and configuring a secure system is especially difficult. The challenge is compounded by characteristics particular to mobility. For example, mobile system configuration requires an understanding of several advanced concepts, such as wireless networking and encryption. Without this understanding, users have incomplete or incorrect mental models of how the system functions. This makes proper system configuration rather difficult. Furthermore, mobile devices often have smaller screens, more limited user interfaces, and fewer hardware capabilities than their non-mobile counterparts. This means system designers must work harder to make configuration easy for the end user.

Although the situation has been steadily improving, many configuration interfaces continue to intimidate end users. These interfaces are often feature-based: they list the different technical features that end users can configure. Users select the appropriate radio button or drop-down box option, and the product changes its behavior accordingly. This approach is effective – if users know what they are doing. For users who are unfamiliar with the system or the technology, the obstacles are formidable. Users must articulate the goals that they want to accomplish and map these goals to the product features that they need to configure.

Today's configuration interfaces often fail to consider how people interact with technology. Reeves and Nass showed that we apply the same social norms that we use for human beings to our "conversations" with computers [20]. Now consider the typical interaction today between a person and a security product. It is a dysfunctional conversation. The product screams, "I have features A through Z!" The person says, "I would

like to achieve Goals 1, 2, and 3." Unfortunately, user goals and product features often do not easily map to one another. Since this mapping process is challenging, users struggle or give up entirely. For security professionals, we argue these interfaces are psychologically *un*acceptable [21].[1]

In the early stages of mobile computing, security configuration was a lesser problem; systems were configured by early adopters who tended to be expert users. These people had the ability and the willingness to master psychologically unacceptable configuration schemes. More recently, however, the explosion of personal computers and mobile devices in the home has changed the nature of the problem. These home systems are now regularly managed by non-expert users – and the security configuration needs to be completed for each system, in each home. Today, we are beginning to see the consequences of difficult configuration interfaces: very few users enable available security features. This problem will only continue to grow as devices proliferate.

In this paper, we discuss our findings for the configuration of secure 802.11 (or "Wi-Fi") networks. We believe that the lessons we learned in this domain will apply to other mobile systems as well. Because of their large impact, security or privacy problems in mobile systems will be widely publicized. In turn, this will reduce consumer confidence. For example, the vulnerabilities in the 802.11 WEP standard were widely publicized. As a result, many experts believe that adoption of wireless technologies was slowed by concerns over the technology's vulnerabilities. Developing easy-to-use, trustworthy mobile devices is critical to the sustained success of mobile technology.

For 802.11 wireless networks, only 20% to 30% of home users who successfully deploy an 802.11 wireless LAN (WLAN) today enable security [4]. Some security experts interpret this statistic as evidence that home users are too ignorant or too unconcerned about security to enable security measures. However, the problem is more fundamental: the user experience of 802.11 products is seriously flawed. Roughly one out of ten products sold generates a technical support call. Most calls address basic setup issues, such as establishing Internet connectivity. Moreover, representatives of the Wi-Fi Alliance report that up to 30% of all 802.11 equipment purchased for the home is returned [10]. This is an order of magnitude higher than other electronics products, such as VCRs. Furthermore, the vast majority of returned products – an estimated 90% – are *not* defective. These statistics paint a troubling picture: for many home consumers, basic network setup is too difficult – even without considering secure network setup.

For application designers, it is unclear how to design security configuration interfaces that home consumers can use. The design rules that work for most consumer applications often do not work for security applications (as we discuss in Section 2). Furthermore, the effectiveness of security applications is difficult to evaluate. Applications that are not evaluated generally will not be improved; without evaluation, it is difficult to demonstrate the need for corporate resources or define a discrete deliverable.

In this paper, we present our design, implementation, and evaluation of a configuration interface for 802.11 access points. The interface enables home consumers to configure their wireless networks securely. Our system acts as an "expert friend," asking simple, high-level questions to elicit the users' needs and goals. This information is automatically translated into a security policy for users. By avoiding feature-based questions, our system empowers end users – even novices – to make configuration decisions appropriate to their situation. With existing interfaces, more knowledgeable users are better able to configure secure networks than novice users. Our system levels the playing field, enabling non-experts to perform as well as experts.

We conducted a series of preliminary studies, which led us to articulate the following design principles:

[1] Thirty years ago, Saltzer and Schroeder outlined eight design principles for minimizing application security flaws. The eighth principle is psychological acceptability:

> *Psychological acceptability*: It is essential that the human interface be designed for ease of use, so that users routinely and automatically apply the protection mechanisms correctly. Also, to the extent that the user's mental image of his protection goals matches the mechanisms he must use, mistakes will be minimized. If he must translate his image of his protection needs into a radically different specification language, he will make errors. [21]

- Assume no prior technical knowledge or expertise on the part of users;
- Minimize human effort: maximize application work;
- Maintain a positive user experience;
- Anticipate error states; and
- Separate distinct concepts.

Using these principles, we developed a configuration interface and tested how well users were able to secure a wireless network.

In the remainder of this paper, we first discuss the challenges in designing and evaluating good security applications in Section 2. We then define our problem space and our design principles in Sections 3 and 4. The design principles were used to implement our configuration interface, which is described in Section 5. We also tested our implementation against two commercially available access points. The evaluation method and experimental results are both briefly summarized in Sections 6 and 7. This is followed by a discussion of related work in Section 8. Finally, we discuss how this work may be applied to other domains in Section 9 and conclude in Section 10.

2 Challenges in Security Configuration

In recent years, application designers have discovered that the design guidelines that work for most consumer applications fail for security applications. Intuitively, the explanation is simple: users' mental models of the world do not match the assumptions underlying the technical implementations. More specifically, Whitten and Tygar outlined five properties of security that makes designing user interfaces problematic [25]. Each property applies to our 802.11 technology case study; in fact, the design challenges associated with each property are potentially magnified for mobile systems. Below, we summarize each of Whitten et al.'s security properties and discuss how each is relevant to security configuration and mobile system design.

The unmotivated user property. First, the unmotivated user property signifies that security is usually a secondary goal for users:

> People do not generally sit down at their computers wanting to manage their security; rather, they want to send email, browse web pages, or download software, and they want security in place to protect them while they do those things. [25]

How this is relevant. For designers, this means that they cannot assume that users are motivated enough to wade through volumes of product documentation or decipher cryptic labels on configuration options (e.g., see Figure 1).

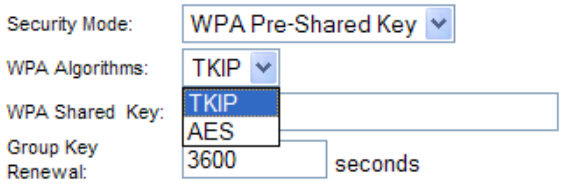


Figure 1: Example of Wireless Security Options
(Image Taken from Linksys WRT54G Access Point Configuration Interface)

The abstraction property. Next, the abstraction property highlights how most users have difficulty conceptualizing security concepts. Computer security management often entails specifying abstract rules, e.g., use

the AES algorithm or "only allow these machines to use my network."
How this is relevant. Abstract concepts make many non-expert users uncomfortable. As we discuss in Section 7, non-expert users are generally able to configure fewer security features than more knowledgeable users. Unfortunately, the most obvious solution – adding more explanatory text – is also inappropriate for display on smaller form factors.

The lack of feedback property. Unfortunately, providing good feedback for security configuration is even more difficult than it is for regular consumer applications. Security systems are complex, and concise summaries may not be adequate.
How this is relevant. Security configuration is often frustrating because users do not know what is happening. For example, users often do not know how to determine whether encryption has been enabled successfully for their 802.11 network. If the network seems to work, does that mean the configuration was successful, or could the configuration still contain errors? This information is difficult to convey in a simple, concise manner, and it is even more difficult on smaller form factors.

The barn door property. The barn door property says that "once a secret has been left accidentally unprotected, even for a short time, there is no way to be sure that it has not already been read by an attacker" [25].
How this is relevant. In a mobile world, important information is stored on devices that have network access. Faulty security configuration may lead to information being compromised. Once that has occurred, nothing can be done in general to repair the breach of secrecy. With mobile devices, it is also extremely difficult to detect a breach, since communication is wireless, and devices may be too limited to keep logs of communication activity.

The weakest link property. Last, the weakest link property reminds us that the security of a system is only as strong as its weakest link.
How this is relevant. Today, many security vulnerabilities stem from faulty configurations or from human action that deliberately seeks to bypass security measures. For any system, designers want to ensure that the user is not the weakest link. Therefore, configuration should be easy for users to complete successfully. In addition, security configuration should not be a separate chore for users to complete; it should be integrated into users' primary tasks. This is especially true for mobile systems; smaller screens and limited input mechanisms make users even less inclined to engage in configuration.

For these reasons, the design rules that work for most consumer applications often fall short for security applications, and this is only magnified for mobile applications. We will describe how we designed a configuration interface in Section 5.

3 Problem Definition

In this paper, we examine the issues behind secure network configuration from several perspectives. In the previous section, we examined the application designer's problem. We introduce the factors that make designing and evaluating a configuration interface difficult – and how these challenges are exacerbated in mobile systems. In this section, we will delve into the user's predicament.

We begin with a model of how networking or security experts might evaluate their own wireless networks. A secure configuration depends on the successful completion of each step in Figure 2. Each step represents a potential point of failure.

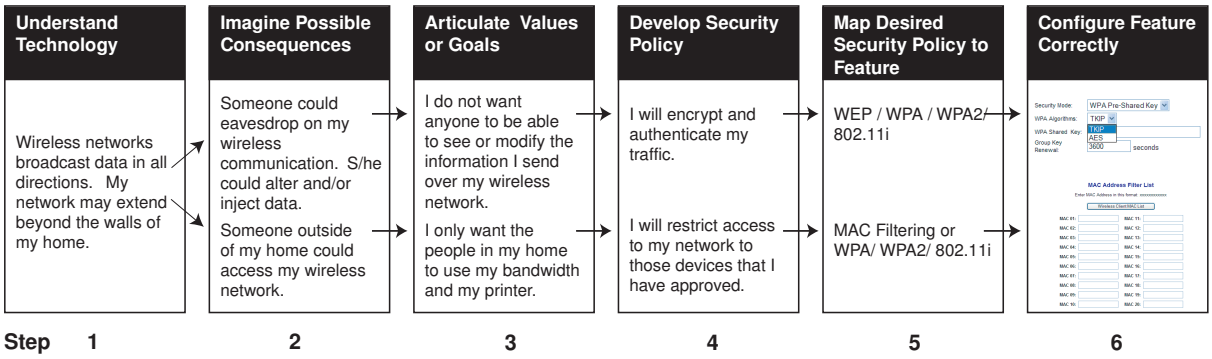


Figure 2: Process for Configuring a Secure Wireless Network (Existing Consumer Systems)

Existing configuration interfaces are often organized around the *features* of a wireless network – not the problems that the user wants to solve. Currently, consumers will reach the configuration step (Step 6 in Figure 2) only if they want to enable a certain feature (Step 5 in Figure 2). Thus, unless consumers *know* that they want encryption (Step 4 in Figure 2), the likelihood of enabling it is small.

Now suppose that average consumers do not have tech-savvy friends or relatives. In this case, consumers only know that they want encryption if they can articulate their goals or values regarding wireless network security (Step 3 in Figure 2). Articulation relies on the consumer's knowledge of security vulnerabilities and their possible consequences (Step 2 in Figure 2). Evaluating the consequences requires a working knowledge of wireless networks and radio signals (Step 1 in Figure 2).

Without a fairly sophisticated level of technical understanding, it is unlikely that today's consumers will be able to effectively reason about their security needs. Users may be unaware that the broadcasting of their data leads to security vulnerabilities; that these vulnerabilities may warrant concern; and that if security is important, steps must be taken to protect their data.

Note how the configuration process illustrated in Figure 2 is extremely delicate. If the user fails to negotiate any of the six steps, the outcome will tend towards an insecure network.

3.1 Existing Configuration Interfaces

We conducted a series of preliminary studies to gain first-hand experience observing users' difficulties with network setup. First, we mapped out the information architecture of various access point interfaces. All past and most current configuration interfaces for 802.11 access points were almost entirely feature-based. An example of this is pictured in Figure 1.

As more and more users have adopted wireless technology – and called vendors' technical support lines – the configuration interfaces have improved. More recently, some vendors have shifted towards a configuration wizard, such as the one shown in Figure 3.

This is good news for both consumers, who appear to be struggling less with network setup, and vendors, who have reduced the volume of technical support calls. In addition, the current configuration interfaces for wireless networks appear to be quite good, according to conventional design wisdom:

- It takes at most three clicks to reach any page in the interface;
- Context-dependent documentation is available on every page;
- All functionality is available from the main menu; and
- It is possible to recover from errors by restoring the access point to its factory defaults.

Typically, when a home consumer opens an access point package, she will find a paper "quick start" guide that illustrates how to connect the access point correctly. Next, the guide will direct the user to pop in an

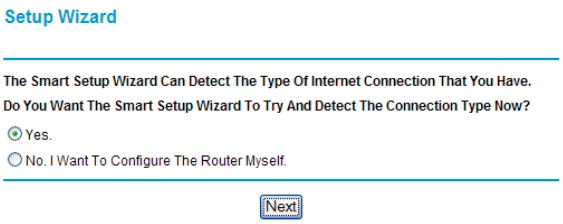


Figure 3: Example of Configuration Wizard for Commercial Access Point
(Image Taken from Netgear WGT624 Access Point Configuration Interface)

installation CD or go to the URL of the configuration interface (e.g., `http://192.168.1.1`).

Despite the extensive directions, however, we observed many users who struggled with network configuration. We used two kinds of user study techniques in this stage: contextual inquiry and usability study. Contextual inquiry is a technique in which researchers select a few representative individuals, visit them in their workplace or home, and observe their behavior. We conducted several contextual inquiries in people's homes, watching users setup and configure secure wireless networks. Each study lasted anywhere from one to four hours. The usability study is probably the best-known HCI technique, where experimenters give participants a set of tasks and observe participants while they try to complete the tasks. We conducted a handful of usability studies, testing whether users were able to complete the tasks we outline in Section 6.2.

For basic configuration, we found many users had difficulty establishing an Internet connection and configuring the Windows networking dialogs. In addition, users failed to secure their networks for a variety of reasons. For example, some users were unaware of the vulnerabilities in unsecured wireless networks. Other users did not know what features needed to be configured, since the paper guides omit any discussion of security configuration.

A major obstacle is that current configuration interfaces are organized by technical functionality. The Linksys and Netgear interfaces expose on the order of 50 distinct features that can be configured. The different features are grouped by similarity in the underlying engineering implementation. This is often unrelated to users' high-level goals. Users often need to visit several different pages in order to achieve one goal.

These preliminary studies led us to develop the model in Figure 2. We found that users stumbled at each step in Figure 2. In general, however, users had more difficulty completing Steps 4 through 6, compared to Steps 1 through 3.

3.2 Issues Addressed

The work we describe in this paper addresses three main issues:

Empowering users to make their own choices. A one-size-fits-all approach to mobile system configuration cannot work in all circumstances. For example, there may be different categories of users who run 802.11 networks in their home. Some households may use their wireless networks to transmit confidential information and desire a high level of security. Other households, such as those full of college students, may have many transient users, so that only the most basic access control measures are practical. Still others may choose to run an open wireless network on principle, allowing anyone within range to use their network. On a practical level, a single default cannot work for everyone. On a philosophical level, we believe technology users should be fully aware of their technology's capabilities and drawbacks. Users should have the right to configure and change that behavior as desired.

Leveling the playing field: making security more accessible to end users. With current products, experts are able to configure features more successfully and more quickly than non-experts. A proliferation of mobile devices is expanding the user base to non-expert users. Configuration interfaces need to accommodate these novice users; they should also be able to setup and configure secure wireless networks.

Maintaining flexibility for application designers and vendors. People often use products in unexpected ways. Keeping changes in software allows vendors to make quick modifications. This is particularly useful for initial product generations, as application designers figure out who is buying their products and what they will be used for. Once usage models have been more clearly delineated, the software can be easily customized for different audiences or uses.

4 Design Principles

Based on the preliminary user study observations we present in Section 3.1, we define the following set of design principles for developing user-friendly security applications.

1. **Assume no prior technical knowledge or expertise on the part of users.** Making security accessible means that we must allow people of *all* expertise levels to perform equally well.
2. **Minimize human effort: maximize application work.** Lighten users' cognitive loads by automating as much of the configuration work as possible. Also, present only as much information as users need, and make that information available when users need it, in the relevant context.
3. **Maintain a positive user experience.** Small details make a big difference. For example, we noticed in our preliminary studies that users strongly preferred setup directions on paper. As a result, we made a point to provide information in the medium which was most appropriate for users. Also, we observed that people have little patience for configuration. At 30–45 minutes, users began to express their displeasure. At 60–70 minutes, users were visibly frustrated. We set a goal of a maximum of 45 minutes for our configuration process.
4. **Anticipate error states.** Users will get lost and make mistakes. A good design needs to anticipate what issues require troubleshooting. It should handle errors gracefully. It should provide useful feedback. Were the configuration settings successfully applied? Do they make sense? Do they do what the user thinks they should do?
5. **Separate distinct concepts.** Conflating different concepts leads to confusion. First, separate users' values and goals from security policies. Novice users are comfortable stating their values, but they are not experts in designing security policies. A better design elicits users' values and derives consistent security policies from the values. Second, separate security policies from their underlying mechanisms. This concept is well known in many disciplines, such as operating system design [11]. Existing configuration applications require users to become experts in security mechanisms before they can realize their preferred policies. Automating the policy–mechanism translation removes a substantial barrier to configuration.

Although these principles may appear obvious, the access point configuration interfaces we described in Section 3 violate several of these principles. We believe that made the configuration experience unnecessarily challenging. In Sections 5 through 7, we show that applying these principles can improve the configuration experience a great deal, particularly for novice users.

5 Design and Implementation

We developed a configuration interface that helps users articulate and implement a security policy using existing tools and technology. This was accomplished using a Linksys WRT54G access point and source code. The source code was downloaded off Linksys' web site, firmware version 3.01.3. It was compiled on Red Hat Linux 2.4.20-8 using gcc 3.2.2.

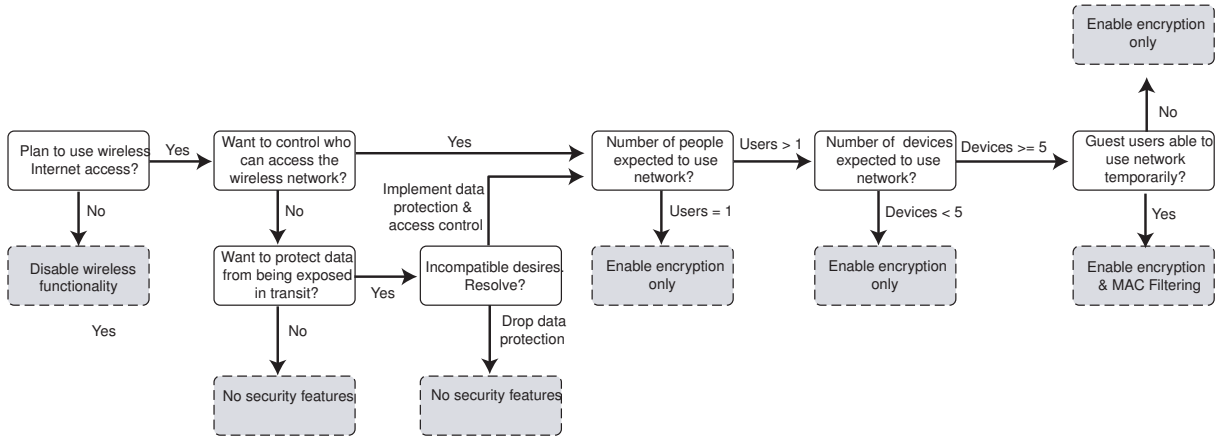


Figure 4: Flowchart of Application Logic
White boxes with solid border: question for user
Gray boxes with dashed border: system recommendation

We modified the source code and compiled a new version of the firmware. The new firmware includes our configuration interface, which co-exists with the original vendor user interface. Users access the configuration interface just as they would access the vendor user interface. Once they connect the access point to a DSL/cable modem and a computer, they open a web browser and direct their browser to `http://192.168.1.1`. This opens the home page of our configuration interface.

A dual-interface design was created so that both our design and the original vendor interface could be used. This was achieved by creating an HTML frame that contained two tabs. The Easy tab switches to our prototype (see Figure 5 for an example), and the Advanced tab switches to the original vendor interface.

Our configuration interface mirrors an online checkout process: the changes are not applied until the entire configuration has been reviewed. The wizard attempts to elicit a user's goals and values by asking general questions, as documented in the flowchart in Figure 4. The questions were crafted so that they would include information about the consequences of making a particular choice. This was done to address the abstraction property of security, as discussed in Section 2.

The system automatically maps the user's preferences to the system's technical features. Any decisions that can be made for the user – and still reflect users' preferences – are automated. This addresses the unmotivated user property (discussed in Section 2), as well as our design principle to minimize human work.

The mapping produces a recommended configuration for the user, which can be changed if desired. The recommendation clearly states the implications of adopting a particular configuration. For example, the recommendation lists how the user can add or remove devices from the network. If the user's preferences produce a set of feature settings that conflict with one another, the wizard asks the user to resolve the conflict. This addresses the lack of feedback and barn door properties (Section 2), as well as the principles of anticipating error states and separating distinct concepts (Section 4).

Each time users access the configuration application, they are taken to the home page. The wizard is always available on the home page. For other situations, we grouped possible actions by goals. The list

includes the common actions that we expected consumers to take, and the items in the list change by context. For example, if no security settings have been enabled, the menu offers the option to turn on access control or encryption. Otherwise, it shows options for giving and taking away network access. Showing different options based on context addresses the design principle for maximizing application work.

We believe that the set of configuration questions shown in Figure 4 balances the needs of our users with the simplicity necessary for a positive user experience. However, this design is not a definitive design for 802.11 configuration. The questions and the application flow may be tailored to specific groups of users. *As the target population changes – as users' needs change and their level of technical understanding changes – the questions may also change.*

In fact, the particular wording of the questions and the choice of questions *should* adapt as the technology changes and as system designers identify different target audiences. The goal is for designers to craft a system where the target audience understands the questions, and the system provides the desired configuration. We believe the best way to accomplish this is by automating the knowledge required in Steps 4 to 6 in Figure 2. In other words, configuration interfaces should automate the translation from human goals to technical features – something that taxes users' abilities.

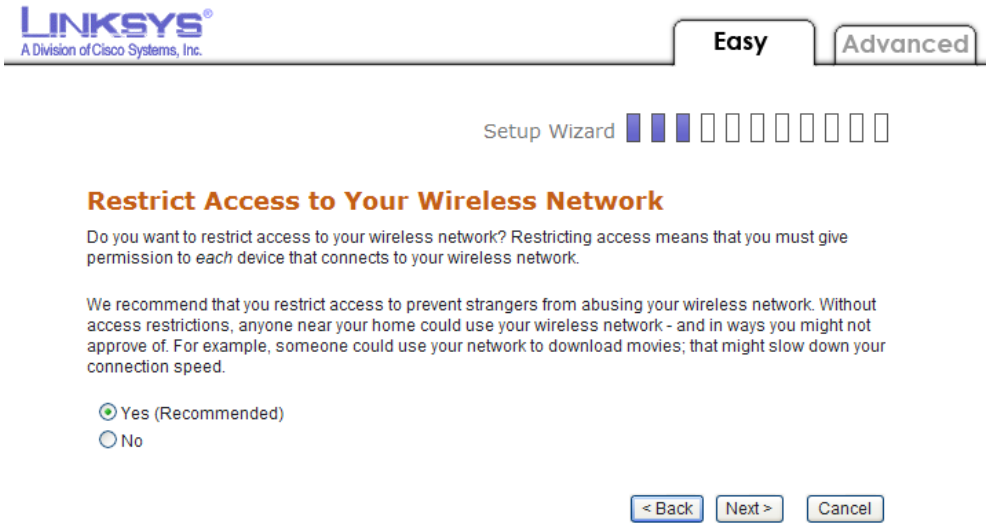


Figure 5: Sample Prototype Screen (Usually the Most Advanced Question Users Will Encounter)

6 Evaluation of Design

In order to test the effectiveness of our design, we developed a methodology for assessing security interfaces. We then tested our configuration interface against the two best-selling commercial access points.

6.1 Target Population

We define the target population for 802.11 products as someone who:

1. Uses wireless Internet access at home, school, or work place on a daily basis (5+ days per week);
2. Has broadband access at home; and
3. Uses a laptop as his or her primary computer.

We included individuals who already had wireless networks at home, as well as individuals who did not.

Eighteen participants were recruited from a broad university population, drawing from both humanities and technology backgrounds. We recruited participants by posting paper flyers on bulletin boards throughout campus and by posting messages on electronic bulletin boards. Interested individuals were directed to a web-based survey form. We selected participants based on their level of computer networking expertise. This was computed using: a self-assessment of their network troubleshooting abilities; whether they had ever managed a wired network; and whether they had ever managed a wireless network. The age of the participants ranged between 18 and 32. Seven participants were female.

Participants were randomly assigned an access point: the Linksys WRT54G, the Netgear WGT624, or our prototype (see Table 1).

Access Point	Low Expertise	High Expertise
Linksys WRT54G	3	3
Netgear WGT624	3	3
Prototype	3	3

Table 1: Participant Assignment

6.2 Tasks Tested

We define the ideal secure wireless network as one where the consumer has:

1. Changed the default password;
2. Changed the SSID;
3. Generated or entered an encryption key on the access point;
4. Entered the encryption key on a client; and
5. Enabled MAC filtering.

We felt these five measures could provide a basic level of security for the average home user.[2] They address the security requirements (i.e., secrecy and authenticity) that commercial technology is equipped to handle. These measures by themselves may be insufficient; for example, attackers may guess a key based on a password. However, such issues are outside the scope of our study.

6.3 Evaluation Method

To compare the effectiveness of different 802.11 configuration interfaces, we developed a technique that combines elements from several different methodologies: mental models interviews, contextual inquiries, usability studies, and surveys.

Mental models interviews are used to understand how interviewees conceptualize certain ideas [17]. Generally, the interviewer will start with a neutral statement, such as, “Tell me about X.” The interviewee is allowed to respond with whatever thoughts come to her mind. The interviewer may ask her to talk more about an idea, and if there are other topics that the interviewer wants to cover, he may ask more specific follow-up questions.

Inspired by the mental models technique, we designed our evaluation method around the concept of *gradual revelation*. Participants were given no indication that the study was focused on wireless security; they were told we were studying wireless network setup. The questions we asked and the activities we planned were ordered such that no information about our study focus was revealed before we first evaluated participants’

[2]Note that MAC filtering becomes unnecessary when WPA or WPA2 is enabled, as each frame received is authenticated by a session key instead of a hardware address. Many access points are now equipped with WPA, but the basic principles that motivate our study remain equally effective.

knowledge of it. For example, we did not mention "encryption" (1) unless participants brought up the concept themselves; or (2) until participants had an opportunity to configure the network and failed to bring up the concept.

When participants arrived for the study, we interviewed them briefly to understand how they conceptualize wireless technology. We then asked participants to fill out a questionnaire. The questionnaire gathered participants attitudes towards various aspects of wireless networks, including availability, reliability, ease of use, use of open wireless networks, security, privacy, and health. Many of these topics are unrelated to security so that participants would not suspect the focus of our study.

Next, participants were handed an access point. The access point was packaged in the box, as if it had been recently purchased. Experimenters present participants with an open-ended scenario:

> Okay, let's pretend you just received an 802.11 access point as a gift. You would like to set up and use the wireless connection today. Your laptop is already configured to use wireless – you just need to worry about the access point. Just set up the access point as you would if you were at home.

We refrained from giving participants a list of tasks to complete to avoid giving indications of our study focus. We observed participants while they set up and configured the access point as they deemed appropriate. During this phase, the experimenter treated the study like a contextual inquiry. Contextual inquiries are generally non-directed observations that allow researchers to observe what users actually do. We incorporated this element of qualitative analysis to evaluate what tasks we would expect participants to attempt on their own.

Since participants were not directed to complete any set of tasks, they may not have completed the tasks (Section 6.2) we had in mind. The experimenter first waited until the participant declared that the configuration was complete. Then the experimenter asked a series of follow-up questions to help guide the participant to the security tasks. For example, if the participant neglected to change the default administrative password, the experimenter would ask:

> With your current configuration, did you know that anyone who knows the default password can log in to your access point? That means they could change any of your configuration settings without your permission. They could even lock you out from your own network if they wanted to. Did you know that could happen?

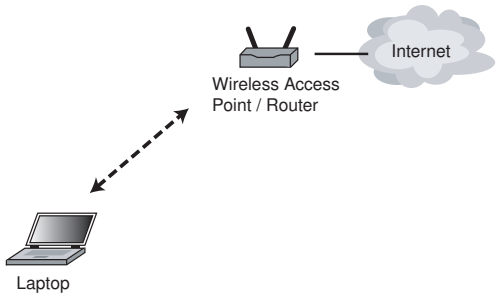
We then asked participants to complete the task. At this point, the study was more similar to a usability study. A usability study allows researchers to gather quantitative data about peoples actions in a limited amount of time. We evaluated participants on their ability to complete the set of five tasks in Section 6.2.

Once the tasks were completed or participants ran out of time, we asked participants to complete the questionnaire again. Surveys allow researchers to gather quantitative data about peoples attitudes quickly. However, because attitude ratings are highly subjective, we only used this data to measure within-subject changes in attitude.

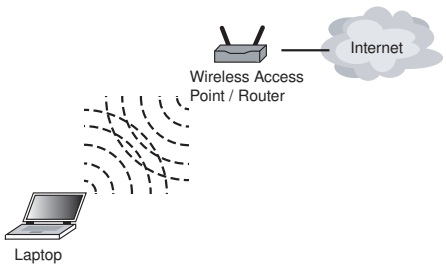
In combining the different evaluation methods together, we believe our technique was able to capitalize on the strengths of each method and minimize its respective shortcomings.

7 Experimental Results

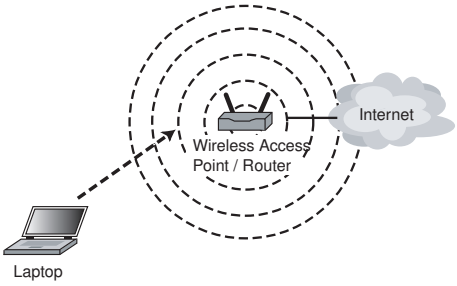
We used the data that we collected to assess how well we expect users will be able to navigate each step in Figure 2. In this section, we highlight the points that are most relevant to the mobile systems community. First, we discuss users' understanding of wireless technology in Section 7.1. This corresponds to Step 1 in Figure 2. Second, we demonstrate in Section 7.2 that on commercial access points, low expertise users have more problems configuring the security of wireless networks than high expertise users. In contrast, users perform comparably using our system, which automates Steps 4 through 6 in Figure 2.



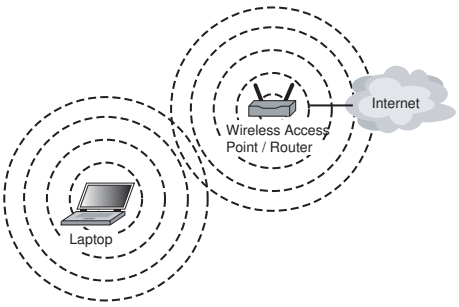
(a) The laptop and the access point figure out precisely where the other is located. The laptop and the access point beam data directly communicate to one another.



(b) The laptop and the access point figure out where the other is generally located. To save power, the laptop and the access point send out data in the general direction of one another.



(c) The wireless access point sends data out in all directions. To save power, the laptop figures out where the access point is located and beams data directly to the access point.



(d) The laptop and the access point do not know where the other is located. Data is sent out in all directions.

Figure 6: Follow-up Exercise to Assess Users' Notions of Wireless Broadcasting.

7.1 Understanding of Wireless Technology

As we discussed earlier, we interviewed participants in our user study briefly to understand how they conceptualize wireless technologies. For example, participants were asked to draw a picture illustrating how data travels from a wireless device to the Internet, and vice versa. As a follow-up question, the experimenter then asked participants to choose the diagram in Figure 6 that most closely matches their ideas.

No participant selected Figure 6(a), a scenario illustrating the access point and client communicating directly with one another across an "invisible wire." Two participants (11%) selected Figure 6(b), which shows both sides using directional broadcast. We expected more people to select this diagram; it is commonly seen on access point packaging as a stylistic simplification. Interestingly, six participants (33%) selected Figure 6(c). Figure 6(c) shows the access point broadcasting in all directions, while the client sends a directed "beam" of data back to the access point. Last, 10 participants (56%) selected Figure 6(d), which shows both the laptop and client broadcasting data in all directions. Happily, all users selected a diagram that visualizes some element of broadcasting, and over half of the participants recognized that both the access point and the client broadcast in all directions.

Unfortunately, the half who selected the wrong figure holds beliefs that may lead them to underestimate the risks of mobile technologies. What if these users are not concerned about eavesdropping because they mistakenly believe the attacker must be physically located between their wireless device and the access point? We did not establish a link between conceptualization and risk perception in this study, but we believe it may warrant future work.

7.2 Configuration Interface Design

Our studies reveal that the design of a configuration interface has a substantial impact on users' behavior. In this section, we present three fundamental observations. First, in contrast to commercial systems, low expertise users will attempt to configure the same security settings as high expertise users using our goal-oriented design. Second, our design enables users to configure the same level of security, regardless of expertise level. Finally, low expertise users react more positively to our prototype, in contrast to the commercial systems.

In our user study, the experimenter first asked study participants to configure the access point without providing any directions or tasks. There are two interesting points illustrated in Figure 7. First, on the commercial access points (Linksys and Netgear), high expertise users attempted to complete more of the five tasks (listed in Section 6.2) than low expertise users. While disappointing, this is hardly surprising; very few people would expect novices to configure unfamiliar features. However, the extent to which low expertise users did nothing may be surprising: using the Netgear access point, low expertise users did not attempt any of the tasks – not even changing the default password! With the Linksys access point, low expertise users attempted one task each. Two tried to change the default password; the other tried to change the SSID.

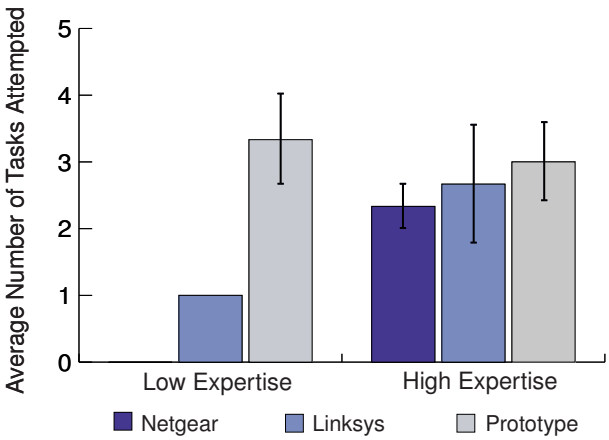


Figure 7: Inclination to Configure Security Features: Average Number of Security Tasks Attempted without Experimenter Prompting

The second lesson in Figure 7 is that given the opportunity, low expertise users would try to configure the same level of security as high expertise users. In contrast to the commercial access points, all users on our prototype, both low and high expertise, attempted to change the default password, enable MAC filtering, and enable encryption. By eliciting users' goals, our prototype interface indicates that users have similar needs to one another, regardless of technical expertise. In feature-based interfaces, however, technical experience and knowledge may serve as a barrier for less savvy users.

Once we began prompting users to complete the tasks, we found that the barrier of technical expertise remained for the commercial access points. This is illustrated in Figure 8. We consider the results in Figure 7 to be more representative of what would happen in the real world, however. During the study, we provided participants with resources that they would have on their own, such as product manuals and access to the

Internet. However, a significant difference between the lab and home environments is that participants did not have access to a technically-savvy friend. At home, users would not be told to complete tasks as they were in our study. These results are shown in Figure 8. It is more likely that users would struggle with the configuration on their own (shown in Figure 7) and/or ask a technically-savvy friend to configure the network for them.

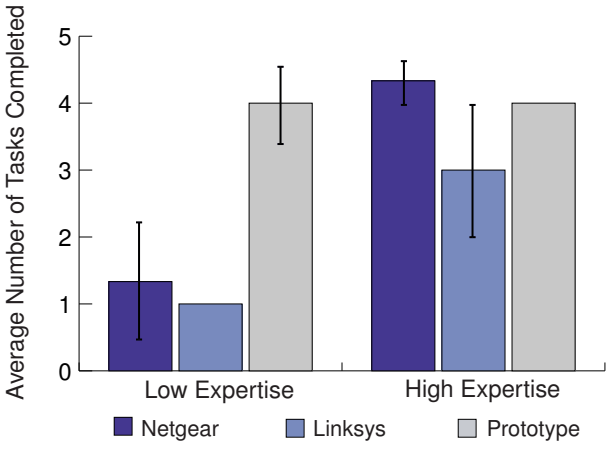


Figure 8: Ability to Configure Security Features: Average Number of Security Tasks Completed

Finally, we evaluated the general user experience of the prototype, compared to the commercial access points. We captured this in the questionnaire with a series of questions assessing how positively the user feels about wireless network setup.

Recall that the questionnaire was administered once before the participants handled the access point and once afterwards. We used participants' change in attitude (measured on a 7-point Likert scale) as a rough indicator of their experience, relative to their prior expectations. A positive change reflects a positive user experience, and vice versa.

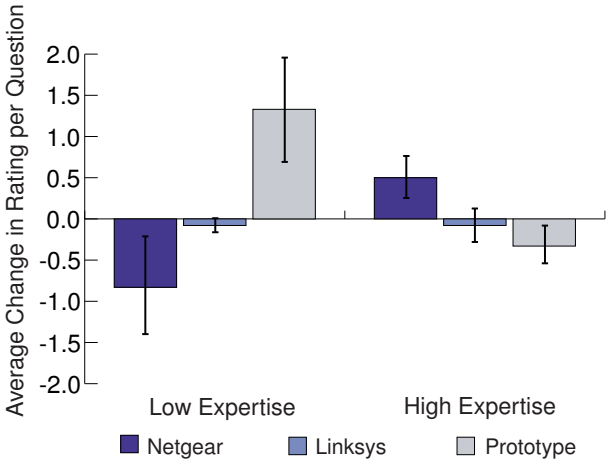


Figure 9: User Experience: Average Change in Ease of Use Rating Per Question

Figure 9 suggests that low expertise users were pleasantly surprised by the prototype. In contrast, low expertise users showed negative shifts in attitude for the commercial access points. We expect this reflects the frustration participants often expressed during the user study. It is also interesting to note that high expertise users may have been less happy with the prototype than with the commercial access points. We speculate that

this is a result of prior expectations: many of the high expertise users managed wireless networks at home, and our prototype did not match their expectations of how a configuration interface should behave.

Due to the high costs of technical support calls and product returns, access point vendors have large economic incentives to improve their configuration interfaces. Vendors have made numerous attempts to remedy the situation in recent years. Thus, it is even more surprising that our goal-oriented design so clearly enhanced users' inclination and ability to configure security features. These results demonstrate that vendors should be able to improve their products dramatically without incurring major costs. This would reduce user frustration and increase technology adoption.

8 Related Work

The most closely related work is Network-in-a-Box (NiaB) by Balfanz et al. [2]. They address the problem of setting up a secure wireless network that is easy to use for users. They assume a custom-built access point with the additional functionality of providing a *location-limited channel* to enable the user to secure communication with the correct access point; in their paper they use an infrared channel. NiaB assumes that the access point can auto-configure itself. In this paper, we are addressing a different problem: how to empower users for setting up the security of their access point themselves. In environments that feature a common security policy, automatic configuration that does not require any user interaction is certainly ideal; however, applications that require user choice for the security policy will need to leverage approaches that we present in this paper. Furthermore, we believe the ideas we express in this paper generalize to other domains, particularly in the mobile systems space. The lessons we learn will be useful for new technologies.

The design of our system draws on several concepts which are used in the field and documented in the literature. Alan Cooper's *The Inmates are Running the Asylum* drives home the benefit of goal-directed design [5]. Cooper first dissects the differences between the users' goals and tasks, and then he argues that products should be designed to accommodate users' goals (not tasks). Security may be a secondary goal for most users, but we believe that this makes goal-based design even more effective.

In addition, Friedman et al. have explored how to design systems that take human values into account [7, 8, 9]. It could be argued that our system is an implementation of value sensitive design.

More generally, the need for security procedures that support the way humans work has been recognized for decades. In 1975, Saltzer and Schroeder list *psychological acceptability* as one of eight design principles for computer protection mechanisms [21]. In the 1980s, Karat [15] and Mosteller and Ballas [18] conduct some of the first studies that considered the impact of user interfaces on security.

Technical solutions include secure key establishment that is intuitive for humans to use [3, 12, 16, 19, 24]. However, secure key setup is only a subset of the challenges we encountered in secure configuration of wireless access points. In addition, man-in-the-middle attacks can be ruled out by configuration with a physical cable.

On the social science side, there are numerous studies on how people interact with existing security systems. Whitten and Tygar study the usability of PGP 5.0, a public key encryption program which was designed to be easy-to-use [25]. Friedman et al. elicit users' understanding of web security through a semi-structured interview [6].

On the design side, Adams and Sasse [1] point out that security system designs have largely ignored usability issues. Many users face conflicting demands and receive no support or training on these systems. In addition, Holmström [13], Jendricke and Markotten [14], and Yee [26] focus on user-centered designs for building secure applications and established design guidelines.

9 Discussion

Many system designers may wonder why we even give users a choice in their security configuration. It benefits the engineers and designers to make the product more "flexible" and "general," but does it benefit the users? Would it not be simpler to enforce a secure default setting? Many of the choices that users can make in today's software are choices for which the users cannot make informed decisions. A pre-configured, easy-to-use, easy-to-secure access point would certainly be desirable to many consumers. However, there are several reasons why it is important for users to have a choice. On a practical level, there may be different types of users. Some households have a small number of users and devices, so a high level of security may be easily implemented. Others may have large numbers of transient users, so only the most basic access control measures are practical. Still others may choose to run an open access point, allowing anyone within range to use their network. A single default can never work for everyone.

On a more fundamental level, choice is also viewed as a desirable feature. In the language of value-sensitive design, users should be autonomous. Users should "construct their own goals and values, and [be] able to decide, plan, and act in ways they believe will help them achieve their goals and promote their values" [9]. If users are autonomous, they take responsibility for the decisions they make and the actions they take. According to Friedman et al., autonomy is "fundamental to human flourishing and self-development" [9]. Without autonomy, individuals are not morally responsible for their actions. Without user interfaces to support the choices they make, users cannot be autonomous.

As a community, the challenge is to design a system that enables users to successfully configure options with which they may be unfamiliar. Our configuration interface is purely software-based, which means that system designers can iterate through software designs quickly, since no hardware changes are required. It does, however, mean that software development teams need to research their target users in order to formulate the right questions. Determining the right questions to ask target users is time-consuming, and the questions may change as the audience shifts.

Using goal-oriented questions for configuration will generalize to other mobile applications. In fact, this method is especially applicable to mobile systems: the characteristics of mobile systems magnify the challenge of designing easy-to-use configuration interfaces. Mobility is difficult to understand, with invisible relationships between devices that communicate wirelessly. Mobile systems are complicated, often requiring that new devices should be able to enter and exit a given network. Mobile devices need to be portable, which means they often have smaller screens, more limited user interfaces, and fewer hardware capabilities than their non-mobile counterparts. Together, all these factors make mobile system designers work harder to make configuration easier for the end user.

As mobile devices and applications become more prevalent, the configuration problem will only continue to grow. Currently, our system has only been designed for PC screens, but it could easily be extended for smaller devices, such as PDAs. Simple questions are more easily viewed on smaller screens than lists of features. In addition, even devices with limited hardware capabilities can be used for configuration, as long as they have a web browser.

We envision these types of questions can be used for anything from configuring location-based applications to Bluetooth security. For example, take a location-based application where users can choose to reveal their location to family members, friends, or other acquaintances. Since the technology is new to most people, users may not fully understand the privacy implications of revealing their location over time. Goal-oriented questions may be useful for helping users determine what kind of privacy settings would be most suitable for their needs: to whom information would be given; what information would be exposed; the granularity of the information that would be available; and so on. Users may not initially realize what options are available to them. A well-crafted configuration interface will make them aware of the implications of the technology, as well as match the configuration with their comfort level.

Like 802.11, Bluetooth has not been adopted as quickly as hoped and has suffered from various security vulnerabilities [22, 23]. “Bluejacking,” “bluebugging,” and “bluesnarfing” have all raised concerns over Bluetooth security [23]. Currently, the workarounds are still very primitive, e.g., turning discovery mode off by default and enabling it only when needed. This situation could be improved with usable configuration interfaces. For example, a device may only be able to pair with devices that the user has pre-approved, or it can only pair in situations that the user has approved, e.g., a phone can only pair with a headset when it receives a call.

The lessons we have learned in our study with 802.11 can aid mobile systems designers improve the user experience of new technologies. For mobile technologies to succeed, they must be easy-to-use and trustworthy.

Ease of use and trustworthiness imply that users need to understand what the technology is doing – at least to the level where they can form correct expectations of how the technology should behave. Unpredictability breeds intimidation in users’ relationships with technology. Without a basic level of understanding, users will be unhappy and bewildered when something does not behave as they anticipate. Inevitably, this will happen if they form the wrong mental models of the technology. Users who understand the implications and limitations of a technology will ultimately be satisfied because the technology meets – or exceeds – their expectations.

10 Conclusion

Home consumers are now responsible for configuring the security settings of their devices. While configuration interfaces have improved since the days of inscrutable VCR recording menus, they still terrorize many end users. Configuration interfaces are often feature-based, listing options available for different technical features.

People, on the other hand, are goal-based. Users may not have a deep understanding of the technology – and they probably never want to. This lack of understanding makes it hard for users to properly assess their security and privacy risks. It also makes it hard for users to configure features while trying to accomplish their goals. Very few consumers truly understand mobile or cryptographic technology, and as a result, very few consumers are willing to configure security in mobile devices.

We cannot point accusing fingers at the application designers, however. These configuration interfaces have incorporated all the conventional wisdom for developing easy-to-use applications. The problem is more fundamental: designing a security application is a different beast than designing a regular consumer application.

We studied end users attempting to configure secure wireless networks, and this led us to articulate five design principles for developing user-friendly security applications:

1. Assume no prior technical knowledge or expertise on the part of users.
2. Minimize human effort: maximize application work.
3. Maintain a positive user experience.
4. Anticipate error states.
5. Separate distinct concepts.

We incorporated these principles into an 802.11 configuration interface that we designed and implemented. The interface empowers end users to configure the level of security appropriate to their needs. It accomplishes this by eliciting users’ high-level goals and values. This information is then translated into a recommendation for a security setting. The recommendation includes the implications of the settings, and the user confirms that the implications match the initial goals.

Our user studies confirm that non-expert users can configure a secure wireless network as well as expert users, if the configuration interface is designed in an accessible manner. Our system demonstrates that assisting users with the translation from high-level goal to low-level feature is a simple but powerful method for building easy-to-use security configuration applications.

Our work generalizes to other security configuration problems in mobile systems, and we hope that other researchers will explore this aspect of mobile computing. Making mobile systems easy-to-use and secure is critical to the adoption of mobile technologies, since it depends on the satisfaction of the people who use them.

References

[1] Anne Adams and Martina Angela Sasse. Users are not the enemy: Why users compromise computer security mechanisms and how to take remedial measures. *Communications of the ACM*, 42(12):40–46, December 1999.

[2] Dirk Balfanz, Glenn Durfee, Rebecca Grinter, D.K. Smetters, and Paul Stewart. Network-in-a-box: How to set up a secure wireless network in under a minute. In *Proceedings of USENIX Security Symposium*, pages 207–222, August 2004.

[3] Dirk Balfanz, D.K. Smetters, Paul Stewart, and H. Chi Wong. Talking to strangers: Authentication in ad-hoc wireless networks. In *Proceedings of Symposium on Network and Distributed Systems Security (NDSS)*, February 2002.

[4] David Cohen. Consumer front-end to WPA. Wi-Fi Alliance, June 2004.

[5] Alan Cooper. *The Inmates Are Running The Asylum: Why High-Tech Products Drive Us Crazy and How to Restore the Sanity*. Sams Publishing, 1999.

[6] Batya Friedman, David Hurley, David C. Howe, Edward Felten, and Helen Nissenbaum. Users' conceptions of web security: a comparative study. In *CHI '02 extended abstracts on human factors in computer systems*, pages 746–747, April 2002.

[7] Batya Friedman, Peter Kahn, and Alan Borning. Value sensitive design and information systems. In Ping Zhang and Dennis Galletta, editors, *Human-computer interaction in management information systems: Foundations*, volume 4. 2006.

[8] Batya Friedman, Peyina Lin, and Jessica K. Miller. Informed consent by design. In Lorrie Faith Cranor and Simson Garfinkel, editors, *Security and Usability*, chapter 24, pages 495 – 521. O'Reilly Media, Inc., 2005.

[9] Batya Friedman and Helen Nissenbaum. Software agents and user autonomy. In *First International Conference on Autonomous Agents*, pages 466–469, 1997.

[10] Alex Gefrides. Personal communication, Wi-Fi Alliance, December 2004.

[11] Robert Grimm and Brian Bershad. Separating access control policy, enforcement, and functionality in extensible systems. *ACM Transactions on Computer Systems*, 19(1):36 – 70, February 2001.

[12] Peter Gutmann. Plug-and-play PKI: A PKI your mother can use. In *Proceedings of the 11th USENIX Security Symposium*, pages 45–58. USENIX, August 2003.

[13] U. Holmström. User-centered design of secure software. Technical report, Telecommunication Software and Multimedia Laboratory, Helsinki University of Technology, 1999.

[14] U. Jendricke and D. Gerd tom Markotten. Usability meets security — the Identity-Manager as your personal security assistant for the Internet. In *Proceedings of Annual Computer Security Applications Conference (ACSAC'00)*, pages 344–353, December 2000.

[15] Clare-Marie Karat. Iterative usability testing of a security application. Proceedings of the Human Factors Society 33rd Annual Meeting, 1989.

[16] Jonathan M. McCune, Adrian Perrig, and Michael K. Reiter. Seeing-Is-Believing: Using camera phones for human-verifiable authentication. In *Proceedings of IEEE Symposium on Security and Privacy*, May 2005.

[17] Granger Morgan, Baruch Fischhoff, Ann Bostrom, and Cynthia Atman. *Risk Communication: A Mental Models Approach*. Cambridge University Press, 2002.

[18] William S. Mosteller and James Ballas. Usability analysis of messages from a security system. Proceedings of the Human Factors Society 33rd Annual Meeting, 1989.

[19] Adrian Perrig and Dawn Song. Hash visualization: A new technique to improve real-world security. In *Proceedings of International Workshop on Cryptographic Techniques and E-Commerce (CrypTEC)*, July 1999.

[20] Byron Reeves and Clifford Nass. *The Media Equation: How People Treat Computers, Televisions and New Media Like Real People and Places*. Cambridge University Press, 1996.

[21] Jerome H. Saltzer and Michael D. Schroeder. The protection of information in computer systems. *Proceedings of the IEEE*, 63(9):1278–1308, September 1975.

[22] Yaniv Shaked and Avishai Wool. Cracking the bluetooth pin. In *MobiSys '05: Proceedings of the 3rd international conference on Mobile systems, applications, and services*, pages 39–50, New York, NY, USA, 2005. ACM Press.

[23] Bluetooth SIG. Wireless security. `http://www.bluetooth.com/help/security.asp`.

[24] Frank Stajano and Ross Anderson. The resurrecting duckling: Security issues for ad-hoc wireless networks. In *Security Protocols, 7th International Workshop*, 1999.

[25] Alma Whitten and J. D. Tygar. Why Johnny can't encrypt: A usability evaluation of PGP 5.0. In *Proceedings of the 8th USENIX Security Symposium*, Washington, D.C., USA, August 1999. USENIX.

[26] Ka-Ping Yee. User interaction design for secure systems. Technical Report UCB/CSD-02-1184, Computer Science Division, University of California, Berkeley, May 2002.