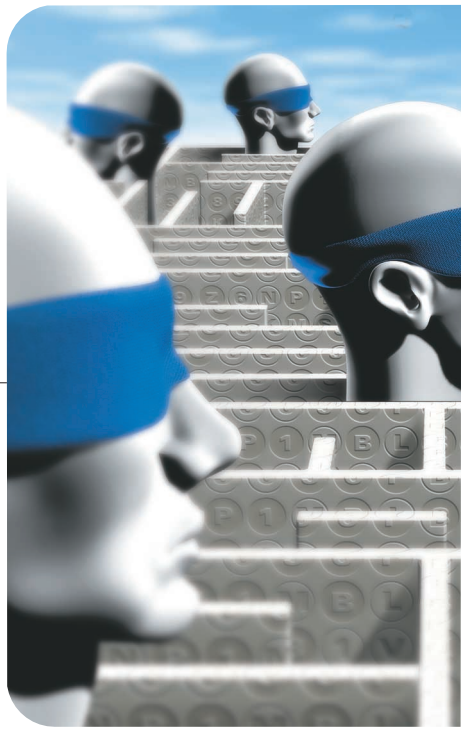
Secure or Usable?

There's an old joke that computers are actually easy machines to secure: just turn them off, lock them in a metal-lined room, and throw away the key. What you end up with is a machine that is very secure—just not very usable.



Lorrie Faith Cranor
Carnegie Mellon University

Simson Garfinkel
Massachusetts Institute of Technology

As the joke's continuing popularity demonstrates, many people believe that there is an inherent trade-off between security and usability. A computer without passwords is usable, but not very secure. On the other hand, a computer that requires you to authenticate every 5 minutes with your password and a fresh drop of blood might indeed be very secure, but nobody would want to use it.

Security and usability

This "received wisdom" on the inherent conflict between usability and security goes against common sense and more than 30 years' experience in building secure systems. Common sense tells us that security and usability must go together: systems that are secure but not usable will not be used, while systems that are usable but not secure will get hacked, compromised, and otherwise rendered useless.

In 1975, Jerome Saltzer and Michael Schroeder[1] identified "psychological acceptability" as one of the eight key principles for building secure systems. In 1983, Don Norman noted that many user errors resulting in data loss are often partly the result of poor interface design. "People will make errors, so make the system insensitive to them," he wrote. Instead of simply requiring confirmation of irreversible actions—confirmations that themselves become automatic—Norman argued that systems should be designed so that their actions are both visible and undoable.[2]

Nevertheless, software that implements security has rightfully earned the reputation of being dramatically more difficult to use than software that lacks explicit security functionality. A simple explanation is that any system that implements some kind of secure functionality in addition to its underlying purpose is inherently more complex than a similar system that does not, because there is more to configure. Another reason is that security mechanisms are designed to make things difficult for attackers: thus, any system that improves usability must ensure that this benefit is only offered to authorized individuals, or else overall security will be compromised.

As design goals, security and usability have many aspects in common. Both require a holistic approach and a vision that system architects, developers, and marketers share. Rarely is security or usability successfully sprinkled on at the end of a product's development: instead, both must be designed and built in from the beginning. But because security and usability are different skills, it's generally harder and more expensive to build systems that are both secure and usable.

In recent years, there has been a growing realization that usability problems are hindering security efforts. Two of the best-known examples are Alma Whitten and Doug Tygar's study on the difficulty of encrypting email with PGP 5.0,[3] and Anne Adams and Angela Sasse's study on users' difficulty in complying with password policies.[4] A small but growing group of researchers have dedicated themselves to working at the interface between security and usability—an interface that has come to be called human-computer interaction and security (HCI-SEC).

HCI-SEC

In spring 2003, an HCI-SEC workshop at the ACM Computer Human Interaction Conference was perhaps the first formal event devoted to discussions of usable security. This was followed by a birds-of-a-feather session at the 2003 Usenix Security conference and a July 2004 Workshop on Usable Privacy and Security Software (WUPSS) held at Rutgers University's DIMACS center. These events show that HCI-SEC is gaining legitimacy as a separate and legitimate field of study. More than 80 researchers and students attended WUPSS, which included two days of formal presentations along with a third day of informal, small-group discussions. Participants came from a variety of disciplines, including security, privacy, software engineering, HCI, and psychology. Speakers identified numerous open problems and suggested a variety of approaches ranging from preliminary proposals to examples of implemented systems.

The password problem

Virtually all computer users come face-to-face with a leading HCI-SEC problem when they try to use an ATM, listen to messages on their cell phone's voicemail system, or access their email: an increasing proliferation of passwords. Individuals are asked to remember dozens of unique passwords, all of which are supposed to be difficult to guess, yet easy to remember. And, to make matters worse, they're often frequently required to change their passwords.

Not surprisingly, user authentication is a problem area that WUPSS attendees discussed at length (and is the subject of three articles in this special issue). As individuals are asked to authenticate themselves to an increasing number of computer systems, traditional text password systems appear to become less secure.

Because it is common knowledge that humans have a higher capacity for remembering images than words, graphical password schemes have been proposed as another possible solution to the password problem. But little work has been done to demonstrate that graphical password schemes are actually secure, or that they fare any better than traditional text passwords when users must employ them to access dozens of different systems. One recent study found that user selection of graphical passwords can be highly correlated with the user's race or gender.[5]

Biometrics and hardware tokens represent other approaches to user authentication. All of these systems have fans and detractors, and have been the subject of several research projects. There is no standard approach for evaluating and comparing them. Indeed, assumptions about their relative security or usability are often taken for granted, with little or no empirical research to back them up.

Approaches

A reoccurring theme among HCI-SEC researchers is the question of how to approach usable security for a particular application. They have suggested three types of approaches:

The first is to build systems that "just work," meaning they can perform security- or privacy-related functions without user intervention. While an appealing goal, one potential problem is that when users are unaware of security issues, they might inadvertently act in ways that undercut protections that have been put in place.

The second is to develop security- and privacy-related metaphors that let users intuitively use security or privacy software correctly. It's clear that today's metaphors of keys and locks are incomplete and somewhat inaccurate; unfortunately, there are no better proposals on the table, and it isn't clear if widespread terminology could even be changed at this point.

The third is to teach users what they need to know to effectively use security or privacy tools. But delivering information to users in a format that users will take the time to read and understand is a challenge that no one has yet solved.

Although it's easy to write that the final solution will probably involve a combination of these approaches, it's important to realize that they represent fundamentally different philosophies and implementations.

Feature articles

Dirk Balfanz et al. shows that the "just work" approach can be implemented in their article, "In Search of Usable Security: Five Lessons From the Field." They show that much of the complexity associated with today's security systems can be minimized or eliminated by redesigning the way that we interact with these systems. Instead of sacrificing security, such changes can actually promote it.

For many years, computer-security experts have dispensed a variety of recommendations for choosing and changing passwords, but rarely has this advice been validated by controlled experiment. In "Passwords Memorability and Security: Empirical Results," Jeff Yan and his colleagues did just that: by breaking several first-year students into different groups, dispensing different kinds of advice, and then trying to crack the students' encrypted passwords, the authors were able to determine that while some commonly dispensed advice is good, other recommendations are bogus.

When users forget passwords or otherwise lose them, many modern systems present them with a series of challenge-response questions for automated password recovery or reestablishment. In "Designing Challenge-Question Systems," Mike Just explores these systems' design parameters. Although some systems can be implemented securely, Just points out that many are not.

Biometrics are frequently suggested as an alternative to password-based systems. Although much attention has focused on finger and iris

prints, these approaches require additional hardware. By contrast, any computer with a keyboard can measure a person's typing patterns. In "Typing Patterns: A Key to User Identification," Alen Peacock and his colleagues review both the research and the intellectual property landscape of this exciting authentication technique.

Security, of course, is about much more than system setup and user authentication. In "Aligning Security and Usability," Ka-Ping Yee suggests a range of design principles that can be combined to build systems that are both secure and usable.

These five articles represent recent work in the usable security area, but they're only a small sample. Nearly 30 articles were submitted to this special issue; sadly, we didn't have the space to print more of them.

Last year, the Computer Research Association identified usable security as one of the "grand challenges" for information security researchers. It is our hope that one day, developers of computer-security systems will no longer consider usability issues challenging. Today, that day seems to be far away, but it's a goal that can be realized. □

References

1. J. Saltzer and M. Schroeder, "The Protection of Information in Computer Systems," *Proc. IEEE*, vol. 63, no. 9, 1975, pp. 1278–1308.
2. D. Norman, "Design Rules Based on Analyses of Human Error," *Comm. ACM*, vol. 26, no. 4, 1983, pp. 254–258.
3. A. Whitten and J.D. Tygar, "Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0," *Proc. 8th Usenix Security Symp.*, Usenix Assoc., 1999, pp. 169–184.
4. A. Adams and M.A. Sasse, "Users Are Not the Enemy," *Comm. ACM*, vol. 42, no. 12, 1999, pp. 41–46.
5. D. Davis, F. Monrose, and M.K. Reiter, "On User Choice in Graphical Password Schemes," *Proc. 13th Usenix Security Symp.*, Usenix Assoc., 2004, pp. 151–164.

Lorrie Faith Cranor *is an associate research professor in the School of Computer Science and the Engineering & Public Policy Department at Carnegie Mellon University, where she directs the CMU Usable Privacy and Security Laboratory. Her research interests include online privacy, usable privacy and security, technology policy, and social impacts of computing. She received a DSc in Engineering & Policy from Washington University in St. Louis. Contact her through her Web site at http://lorrie.cranor.org/.*

Simson Garfinkel *is a researcher in the field of computer security and a frequent commentator on information technology. Currently a doctorial candidate at MIT's Computer Science and Artificial Intelligence Laboratory, Garfinkel's research interests include computer security, the usability of secure systems, and information policy. Prior to joining CSAIL, he founded Sandstorm Enterprises, a computer-security firm that develops offensive information warfare tools used by businesses and governments to audit their systems. Contact him at simsong@acm.org.*