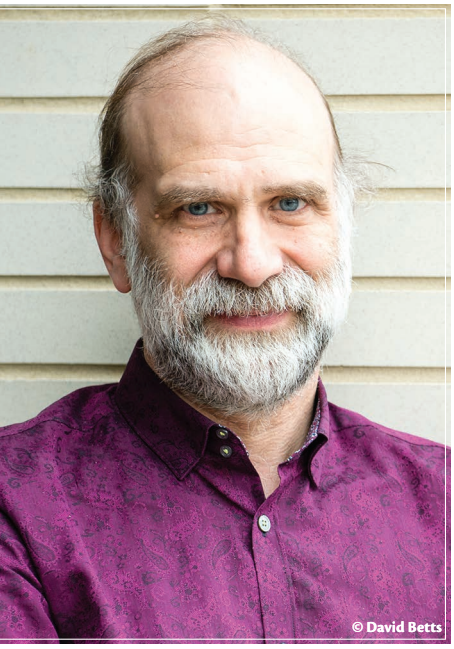

Bruce Schneier
Harvard University

Stop Trying to Fix the User

Every few years, a researcher replicates a security study by littering USB sticks around an organization's grounds and waiting to see how many people pick them up and plug them in, causing the autorun function to install innocuous malware on their computers. These studies are great for making security professionals feel superior. The researchers get to demonstrate their security expertise and use the results as "teachable moments" for others. "If only everyone was more security aware and had more security training," they say, "the Internet would be a much safer place."

Enough of that. The problem isn't the users: it's that we've designed our computer systems' security so badly that we demand the user do all of these counterintuitive things. Why can't users choose easy-to-remember passwords? Why can't they click on links in emails with wild abandon? Why can't they plug a USB stick into a computer without facing a myriad of viruses? Why are we trying to fix the user instead of solving the underlying security problem?

Traditionally, we've thought about security and usability as a tradeoff: a more secure system is less functional and more annoying, and a more capable, flexible, and powerful system is less secure. This "either/or" thinking results in systems that are neither usable nor secure.

Our industry is littered with examples. First: security warnings. Despite researchers' good intentions, these warnings just inure people to them. I've read dozens of studies about how to get people to pay attention to security warnings. We can tweak their wording, highlight them in red, and jiggle them on the screen, but nothing works because users know the warnings are invariably meaningless. They don't see "the certificate has expired; are you sure you want to go to this webpage?" They see, "I'm an annoying message preventing you from reading a webpage. Click here to get rid of me."

Next: passwords. It makes no sense to force users to generate passwords for websites they only log in to once or twice a year. Users realize this: they store those passwords in their browsers, or they never even bother trying to remember them, using the "I forgot my password" link as a way to bypass the system completely—effectively falling back on the security of their email account.

And finally: phishing links. Users are free to click around the Web until they encounter a link to a phishing website. Then everyone wants to know how to train the user not to click on suspicious links. But you can't train users not to click on links when you've spent the past two decades teaching them that links are there to be clicked.

We must stop trying to fix the user to achieve security. We'll never get there, and research toward those goals just obscures the real problems. Usable security does not mean "getting people to do what we want." It means creating security that works, given (or despite) what people do. It means security solutions that deliver on users' security goals without—as the 19th-century Dutch cryptographer Auguste Kerckhoffs aptly put it—"stress of mind, or knowledge of a long series of rules."

I've been saying this for years. Security usablity guru (and one of the guest editors of this issue) M. Angela Sasse has been saying it even longer. People—and developers—are finally starting to listen. Many security updates happen automatically so users don't have to remember to manually update their systems. Opening a Word or Excel document inside Google Docs isolates it from the user's system so they don't have to worry about embedded malware. And programs can run in sandboxes that don't compromise the entire computer. We've come a long way, but we have a lot further to go.

Blame the victim" thinking is older than the Internet, of course. But that doesn't make it right. We owe it to our users to make the Information Age a safe place for everyone—not just those with "security awareness." ■

Bruce Schneier is a security technologist at the Berkman Klein Center for Internet and Society at Harvard University. He's also the CTO of Resilient and Special Advisor to IBM Security. Contact him via www.schneier.com.

Copublished by the IEEE Computer and Reliability Societies