Debunking Security–Usability Tradeoff Myths

M. Angela Sasse | University College London
Matthew Smith | University of Bonn
Cormac Herley | Microsoft
Heather Lipford | University of North Carolina at Charlotte
Kami Vaniea | University of Edinburgh

Guest editors M. Angela Sasse and Matthew Smith discuss the origins of the security–usability tradeoff myth with leading academic experts Heather Lipford and Kami Vaniea and industry expert Cormac Herley.

Sasse: Have there been instances in which people have said, "we can't make this usable," or "making it usable would make it less secure, so we're not even going to try"?

Herley: This is the go-to language I hear when people want to phone in an excuse: "We can't do better in usability because security is so important. Password length and composition policies might be unwieldy, but they're necessary to make things secure."

Lipford: Researchers have discovered that changing passwords every three months is ineffective at increasing security and only encourages people to reuse passwords in predictable ways. What's more important than changing passwords is getting people to use unique passwords for important sites. This finding has resulted in changes to guidelines (www.gov.uk/government/uploads/system/uploads/attachment_data/file/458857/Password_guidance_-_simplifying_your_approach.pdf).

Herley: Another example of the security–usability tradeoff myth is the masking of passwords in the browser. Until recently, this was pretty much ubiquitous in every version of every browser—but why? Where's the evidence that it's actually accomplishing any good?

Sasse: Password masking is a classic example of the security–usability myth because, from a user's viewpoint, if you can't see that you've made a mistake—if you're missing that feedback—you don't know what's wrong when you can't log in. Under certain circumstances, unmasked passwords could give some information to an attacker. This risk isn't high, whereas statistics show that legitimate users mistype their passwords at least one in 10 times.

Herley: Right, this has become much more pronounced with the ubiquity of soft keyboards and mobile devices. Mobile devices were the first to give the option of displaying your password when the unnecessary misery it was causing became obvious.

Generally, the way I think of tradeoff in an engineering sense is, I'm trading off X for Y, which means I give up a little of X, and I get a little more of Y. But implicit in the idea of tradeoff is that I'm not really talking about zero X or infinite X, or a zero Y or infinite Y.

If we're talking about a tradeoff between security and usability, it implies there's some operating point where getting a little more security for less usability wouldn't necessarily be a good deal. But when people

 Copublished by the IEEE Computer and Reliability Societies 1540-7993/16/$33.00 © 2016 IEEE

claim there's a tradeoff, they never follow it up with, say, "Here is how much security you're going to get, here is a precise statement of how much password masking helps in terms of security, and here's how much it hurts in terms of usability. We're making an informed decision and getting a good deal for what we're giving up."

It masquerades as a tradeoff, but it's really just a stalking horse for the claim that we always need more security.

Lipford: However, time and again, researchers have shown that decreasing the usability can lead to less security because you're asking people to do things that they, in reality, aren't able to do. Again, this leads to a false tradeoff. We need to figure out what behaviors people can do, are willing to do, have the time and effort to do to get the most security possible. So we can talk about a balance, but to me that doesn't imply, like Herley said, that security will necessarily decrease. In many cases, you're actually improving security by increasing usability.

> **Time and again, researchers have shown that decreasing the usability can lead to less security.**

Herley: If you're serious about building a good system, you wouldn't insist that there's a tradeoff between two things and then not bother to measure either of those things. For instance, for the battery in your phone, there's a tradeoff between weight and how long it will charge. That doesn't lead us to the conclusion that the battery's weight should be zero, or be 10 pounds so that it never runs out. You measure how much you're getting and decide on some operating point.

The usable security community is shining the light on the fact that we're actually not getting much in exchange for the usability burden imposed. The issues we've been discussing—password masking and composition policies—come with a fairly clear demand in terms of what they ask of technology users. I'm asking you to change your password every 70 days. The promise of what you're getting in return for it is very vague.

Smith: One issue here might be something we can't measure that well—how attackers will react to a security feature disappearing. We're currently saying, "well, the risk of someone shoulder-surfing a password in real life is actually fairly slim, so password masking doesn't make sense in a usable security tradeoff situation." However, if we drop the feature, we open a new attack vector. Any kind of measurement we did beforehand isn't valid once we've made the change.

Sasse: Even when, from a usability viewpoint, you provide evidence of an actual burden, there's no evidence being provided to quantify the security risks. So we can't actually make an informed tradeoff.

Herley: It's a case of trading off X versus Y but I need infinite X (security), which isn't what we generally describe as a tradeoff. When one of the variables is zero or infinity, calculations get strange quickly.

One can't just say, "let's measure how much of every particular attack we see and this countermeasure. No one is hitting this countermeasure right now; therefore, if we take it out, it won't do any harm." But data doesn't answer all questions. Drawing an attack tree or writing down the precise circumstances that apply for the assumptions under which a countermeasure is making a difference—sooner or later there's no escape from making plausibility assessments. We can't invest in absolutely every countermeasure for every attack that anyone can think of. We must make tradeoffs and say we're going to invest in this and this, and not in that.

Lipford: There's an important set of users that I want to make sure aren't overlooked in this discussion, which are the security experts—security administrators, systems administrators, software developers, and so on. They use numerous tools for security to configure infrastructures and look for security bugs in their code. Usability plays a large role here, too. If these tools are unusable, experts will misconfigure, leaving vulnerabilities in applications. And yet usability is still overlooked in those tools. There doesn't appear to be a usability–security tradeoff argument here. Usability isn't a large focus yet, but it could have a big impact because the security of our servers, infrastructure, and applications impacts end user security.

Smith: That's a very excellent point. Have you come across any myths in this area?

Lipford: The main myths are that experts are mostly happy with their tools and they don't care much about usability. Many of the tools are command line and aren't heavily GUI based, but those that are come with a lot of power.

At the same time, there's a culture, and some experts, having undergone the large learning curve for these tools, now have this set of tools at their command that

others don't. It means that they're experts in their fields and are valuable. But overall, this perception hurts the community. There's a high bar of entry to becoming, for example, a security administrator and learning some of the ins and outs of the tools. It takes a lot of learning and training and money and time to get those up to speed. Yes, experts are willing to learn unusable tools and get around the lack of usability—and in the end, they understand and can use those tools well. But this learning curve hurts the overall security of an organization or a country.

Smith: One of the things you said is my favorite myth in this area: experts don't make mistakes. Basically every single vulnerability, every single misintegration is a developer or administrator making a mistake. That's something we need to rethink. Every mistake an expert makes affects so many more people than just themselves.

Sasse: One impact of warnings, whether for SSL certificates or other things, is that they've basically trained users to ignore all warnings. Erring on the side of security might seem the safe choice, but if a feature isn't working for users, it will ultimately undermine security.

Smith: Right. There's no doubt about the click-through phenomenon because people to just want to make warnings go away as fast as possible. Application writers throw popups and warnings, and ask users far too often for security decisions that users are not necessarily able to make. Even though there's scary stuff out there, the relentless procession of false positives has effectively taught people that clicking through warnings is almost always the right answer. Therefore, the warnings that we're trying to give people, when they're not false positives, get thrown out with the bathwater. That's a definite example where more security has produced very much the opposite of the intended effect.

Lipford: Warning research has come a long way. Some very good research has come out of Google in particular. The warning's design, how you communicate the information, and the options you provide are very important; these can make a big difference in how much people pay attention and adhere to the warnings.

And then there's a habituation problem—basically bothering users with warnings perceived as false positives, whether they are or not. Reducing the number of warnings that people saw actually increased the likelihood that they adhered to those warnings.

Herley: Which ties back to your earlier point: one of the reasons for so many false positives—misconfigured certificates—has been reduced in the last two or three years. The experience of trying to deploy and administer an SSL site was itself very unusable.

Sasse: The myth in the security community was that SSL certificates were a good thing, when in reality they weren't, because few developers were able to implement them correctly.

Herley: Yeah, I wrote in a paper—must be like six or seven years ago now [**add reference?**]—that 100 percent of SSL certificate errors were false positives and that not a single user had been saved from harm by a certificate error warning. I dared people to give me counterexamples. And in the seven years since, I think I've had one person approach me with what seemed like a credible case in which a certificate error had actually been spotted—not by an end user but an administrator. The fact that such an extreme statement didn't produce a deluge of security experts coming to me saying, "no, you're wrong, and here's the documented instances where it's not 100 percent the case," is somewhat telling.

Vaniea: I've been doing some work with updates, and Windows 10 is a perfect example of this. Many administrators, and I imagine, the people behind Windows 10, believe that giving users options is a good way to destroy your security. So they just blow aside all the user concerns and automatically set up security. Then when users start figuring out ways to get around it—because going around it is the only way to get anything sane happening—the organizations take away that option. I cite Windows 10 here because it's prevented home users—and home users only—from disabling automatic updating on the system.

While I was interviewing users about updates, one told me that he looked through all the updates and unchecked the suspicious-looking ones. I explained this to an administrator that evening as an example of how people are handling the updates, and he just turned to me and said, "the audacity of a user to question the opinion of the expert who put this update in there is shocking."

Consequently, I began collecting piles of anecdotal evidence from users who talked to me about how they're in a permanent update cycle—their systems start updating then crash, they spend two hours removing the update, and do it all again tomorrow. They literally have to pull the plug on the computers and the Internet to use it for anything. Users are making bad decisions. We can take away their ability to make decisions, but now we're starting see to the side effects of lack of usability.

Lipford: Updating is a good example of where, for the most part, systems are more secure because it's become

automated. However, Vaniea and others have also demonstrated that users aren't getting very good information about what's in the updates. We've all had experiences where an update will change the user interface of an application or will suddenly not work because of a certain system configuration. Users don't understand that critical security updates are also an aspect of updates, so if this important piece of information isn't communicated, users might decide to delay updates, thinking that that's not a problem. What they want to do is delay the nonsecurity problems associated with updates, but in the end, this has security implications—they're missing potentially important security updates.

> **Users are making bad decisions. We can take away their ability to make decisions, but now we're starting to see the side effects of lack of usability.**

So giving users an option here is important because updates can negatively affect them. But we need to give them additional information to help them make better and more informed decisions.

Vaniea: Some beautiful work on this has come out of Google. Rob Reeder examined the best way to stay secure online, and he has a graphic that shows that security experts rate updating at the top of their list. Updating didn't even make the list of things that users considered to be important. The security experts are reacting, saying "updating is the most important thing; why aren't users doing it; let's take away the decision making." But we're not communicating to users that updating things like Adobe Reader has anything to do with security, so they don't perceive it as a security problem. Instead of fixing the communication, we're just seeing enforcement.

Sasse: There's also the problem of "improving" security that doesn't need improving. We've seen an increase in two-factor authentication [2FA]. Some well-established 2FA products have a one-time password [OTP] that's valid for one minute, and it's just six numbers. Some companies produce more sophisticated 2FA products that can generate longer alphanumeric OTPs. So some security experts decided to change to an eight-digit alphanumeric OTP "because it is more secure than a six-digit code." The consequences for users trying to enter these are disastrous: you can't read off an eight-character complex OTP, rehearse it in short-term memory, and enter it into the system that you need to enter it into in one step. So you need to start looking back and forth between the token and the screen, and that causes users to make mistakes and then run out of time. But a six-digit code is strong enough for an OTP that only lasts a minute.

Herley: That's a good example of reasoning backward from "more security." More security is better, and therefore, we're going to do this. It's unusable, but that's where the tradeoff is, so "too bad, your usability is going to be worse." The set of circumstances where the security benefit between six and eight digits is so small as to not be worth any usability hit whatsoever. If I can see a scenario in which this might increase security, I now have carte blanche to do whatever—no matter what happens to usability. That's collateral damage.

Lipford: So how do we combat this myth? One way is to advocate for understanding the impact on usability, the security impact as well. And Herley, you did this nicely when you pointed out the difference between an online password that can sustain an online attack versus a password that can withstand an offline—say a brute-force attack [**add reference?**]. The required strength for those two is very different, so if a password composition policy requires you to fall somewhere in the middle, you're not gaining any additional security—and you're adding more burden to users.

Organizations need to decide what their goals are. If you need to withstand an offline attack and to have that strong a password, then you're asking users to be burdened to do something more for no real security gains—or the nebulous security gains that you have difficulty explaining. This is where policymakers, developers, designers, and so on, make mistakes because they're not gaining real security, and yet they're making things less usable. One way to combat this is to help people understand what the real threats are and how to prioritize them.

Herley: I think you've hit it on the nail: the problem with talking about the tradeoff between usability and security is that it encourages very imprecise thinking. If you try to tabulate or write down the set of assumptions or the attack scenario under which this will make a difference, it ends up being either nonexistent, or so narrow that it's not worth it.

But it's as you said: all too often, that step never happens. Security experts simply invoke the myth of tradeoff between usability and security and use this as cover to avoid the exercise of saying precisely what security

benefit in precisely what scenarios this usability burden is going to deliver.

Vaniea: It's also an economics problem of who makes decision and who has the controls. One interesting usability question from an administrator perspective came up during the recent Cyber UK, which includes practitioners and academics, and a range of different groups. Someone said "we made this mistake before; we used to include a dial on administrators' user interfaces so they could choose the security level that they felt most appropriate for their organization." And it caused every security person to dial it to the highest number they could possibly put it on, because their job isn't to make the company efficient; their job is to make certain they're not getting blamed for security compromises.

Sasse: So how could we start tackling this? How do we start to improve understanding of what good security looks like?

Vaniea: A lot of it revolves around measurement. I can measure dialing up to 11, or I can tell you I used a really horrible password policy. It's harder to enumerate how much time it's wasting the company, it's harder to enumerate the opportunity costs that are lost due to security, and it's really hard to measure the number of people sidestepping and therefore decreasing security because they're actively trying to make sure you don't notice.

Sasse: We found it fairly straightforward to measure the negative impact of unusable security on productivity—it's just that nobody had ever done it. When we inform businesspeople the amount of time employees are spending on security, and the kinds of workarounds they're creating just to be able to deliver a business, they are completely aghast—the cost was hidden. In my experience, the security experts just don't consider the impact, or how to measure.

Vaniea: You're explaining to the security experts how to do these types of measurements. You said it well in one of your prior presentations: It's easy to measure how long the password requirements are. I'm trained in how to do that as a computer scientist. I'm not trained in how to do these other, less obvious measurements, and if you want to talk about things like usability for system administrators, how do we make some of the measurements you're talking about easier to do or more accessible, or more obvious. I'm on a "system administrator usability" kick, but I'm focused on usability for all the humans in this process. And at the moment, it's not obvious to system administrators how to do some of these things.

Herley: Usable security and measuring what we can measure have helped to demonstrate the burden of what we ask users to do. It's a two-pronged approach: One is demonstrating the burden. Often, when you point out the burden, security people are very good at saying, well, too bad; it's security, you need to do it.

The second prong of attack is to point out that much of the security benefit has been assumed or asserted rather than shown. For instance, many people assert that forcing people to change their passwords every 70 days is an absolutely necessary policy, but do we have the data to support it? No. Can we prove it from first principles? No. Measuring the usability burden of this and, at the same time, pointing out the set of circumstances under which this actually improves outcomes when you write down all the assumptions or what an attacker has to do—it becomes very narrow.

It's a combination of measuring the usability burden and forcing people to be more precise on exactly under what circumstances going from a six-digit, one-time code to an eight-character alphanumeric will make an improvement.

Vaniea: He's dead on.

Lipford: We need to convince the experts, because they're the ones making the decisions on application design and organization policy. Very few of them get training in usable security, so they aren't aware of the methods or any of the lessons that we've learned.

There are various ways to get the results of our research to these folks. Our community is still fairly young, and it's going to take time for us to have the bulk of research to be able to translate it into practice. But we can make a more concerted effort to do that as well as put in a larger effort to spread usable security education. Not just through folks like us who are in the community, but integrating the important usability lessons—the ones that really demonstrate the value of understanding people and their behavior—into more standard security education. That's one problem I'm very interested in pursuing.

Vaniea: Recently GCHQ [Government Communications Headquarters; the agency that sets information security standards in the UK] released official guidance on passwords-based on research. I'm currently analyzing an email list run by patch administrators who discovered this document. It resulted in about 60 emails of debate on where this document had come from and who this group was that had released it—because most of the experts on the mailing list are Americans—whether this group could be trusted, whether or not this group knew what they were talking about, and if the document

made any sense. They weren't able to get the answers to their key questions out of that document. Many of them seem to have rejected it because it didn't align with their beliefs, and there was nothing there that caused them to change their opinion, because it hadn't actually targeted their opinion problem. Rather, it focused on what they needed to do.

Herley: At the same time, I'm very encouraged that GCHQ did the somewhat brave thing of recognizing limitations in its recent document "Password Guidance: Simplifying Your Approach" [www.cesg.gov.uk/guidance/password-guidance-simplifying-your-approach]. We all want more security, but we've been dialing it up to 11.

I'm definitely encouraged that organizations like GCHQ are now willing to revisit things that have been taken as accepted wisdom for so long, and yet don't have a lot grounding to them. The National Institute of Standards and Technology (NIST) is currently relooking at its 800-63 document [http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-2.pdf], which is a similar set of guidance for the US. Microsoft put out something recently in its consumer-facing messaging about how it dialed back on a few of the password things that were largely unsupported by evidence and were causing unnecessary misery among users [www.microsoft.com/en-us/research/wp-content/uploads/2016/06/Microsoft_Password_Guidance-1.pdf]. In other words, it's progressing slowly in a number of organizations, and not just ones with exclusive responsibility around usability. Ones that are looked to for guidance in security are starting to get more realistic.

> **Often, when you point out the usability burden, security people are very good at saying, well, too bad; it's security, you need to do it.**

Lipford: This field has many open questions, so there's a lot of room for researchers to gather some of that data and demonstrate the impact of different design, usability, and policy choices to policymakers or organizations. We still have a big need for more research in this area, and that in itself will be very valuable to eventually combatting this myth and to making technologies and systems that are both usable and secure.

Sasse: Until last year, many papers focused on how to make users pay more attention to warnings that are utterly useless. That is such a fundamental misunderstanding of what usability is about. One article in this special issue actually points out that both security specialists and developers have fundamental misconceptions about usability [D. Caputo et al., "Barriers to Usable Security? Three Organizational Case Studies," **pp. XX**]. They see it as only about "making something look pretty."

Herley: I find this a problem in the technology field in general, not only in security. User experience, as a field, is still growing and maturing, and putting that message out. So I don't think we're doing it alone. In our field, we're obviously targeting security experts, but we need to look at how others are targeting usability toward everyone in computing, because I find that opinion prevalent, not just among security specialists but people who have no or very little training in usability and human–computer interaction.

The other thing that security experts often have fundamental misunderstandings about is security itself. We simply don't have the data that shows that many of the things we assert make a difference are as important as we say. Couple this with a poor understanding of usability and the usability burdens, and it's a toxic brew. And as Sasse says, there's been a lot of emphasis on getting users to pay more attention to warnings under the unquestioned assumption that (a) they should be spending more time on these warnings and (b) they should always default to the most conservative option, which is to disconnect when there's a certificate error. And often, those assumptions are just simply incorrect. When 99.999 percent of certificate errors are false positives and users have a task to complete, spending more time examining the certificate or terminating the task—that's not the correct answer. The correct answer is to spend as little time on it as possible. Failing to recognize our own fallibility is certainly holding us back.

Vaniea: There's also the broad problem of trying to identify what is it you should be doing. It's a question I get asked all the time. "What do I do to be secure?" We don't say this, but we don't actually know the answer. It's hard to then step back and look at all of security and ask, what is it that must happen in this scenario, or what is the best outcome for this user? We don't see enough researchers stepping back to ask what a person actually needs to do and how we can make that the most usable thing.

Herley: I think saying "I actually don't know the top three things that will guarantee to keep you safe online" is better than doing a denial-of-service attack on the user's

cognitive burden—giving an endless list that stretches to the horizon, happy in the knowledge that no user is every going to do it. But at least I can't be blamed when things go wrong.

For example, if you're trying to trade off two different things, is it more important that I change my password every 90 days or that I never write it down? The honest answer is we don't have the tools to even answer questions like that. Admitting that ignorance, if only at first to ourselves, is a first step in trying to figure out what it would take to answer that question.

Sasse: Finally, who could do something in this space, and what would it take to correct the misconceptions that currently drive this?

Lipford: First and foremost, we need to translate the research we've already seen into actual guidelines and policies for best practices and having conversations with those who can disseminate this info—for instance, NIST in the US and similar institutions in other countries. People trust these organizations and assume the issues have gone through some level of discussion and debate. Because let's face it: a very busy security administrator or software designer isn't going to read any individual paper on behavior outcomes, but they will read and follow guidelines in their field for decisions.

Vaniea: It's important to educate or at least put these ideas in the language of people who make these decisions, from system administrators to software developers. We also need to find ways to codesign education materials within those spaces, with those groups, so that the information is being cast in light of their own interests.

Herley: When people talk about the tradeoff between usability and security, the thing that I find most surprising is that they seem to treat it with the same reverence as one of Newton's laws or Maxwell's equations when it really just has the status of a slogan. The keys to getting rid of the myth and getting a sensible tradeoff in the use of user time are forcing precision, doing more measurements in usability, and making clear the magnitude of the burden as well as forcing clarity on precisely when and under what circumstances this burden will make a difference. Because if we allow a simple slogan to determine our design choices, it's won't be surprising when we end up without an optimal allocation of resources, such as user time.

Sasse: Great, thank you very much all of you for your time and for your contributions to this. ■

M. Angela Sasse is a computer science professor at University College London. Contact her at a.sasse@ucl.ac.uk.

Cormac Herley is a principal researcher at Microsoft. Contact him at cormac@microsoft.com.

Heather Lipford is an associate professor at the University of North Carolina at Charlotte. Contact her at heather.lipford@uncc.edu.

Matthew Smith is a computer science professor at University of Bonn. Contact him at smith@cs.uni-bonn.de.

Kami Vaniea is a lecturer in cybersecurity at the University of Edinburgh. Contact her at kvaniea@inf.ed.ac.uk.

cn *Selected CS articles and columns are also available for free at http://ComputingNow.computer.org.*