Participatory Design for Security-Related User Interfaces

Susanne Weber, Marian Harbach
Usable Security and Privacy Lab
Gottfried Wilhelm Leibniz Universität Hannover, Germany
{weber,harbach}@dcsec.uni-hannover.de

Matthew Smith
Usable Security and Privacy Lab
Rheinische Friedrich-Wilhelms-Universität Bonn, Germany
smith@cs.uni-bonn.de

***Abstract*—In this short paper, we explore the advantages of using Participatory Design (PD) to improve security-related user interfaces. We describe a PD method that we applied to actively involve users in creating new SSL warning messages. Supported by a designer, participants tapped into their experiences with existing warnings and created improved dialogs in workshop sessions. The process resulted in a set of diverse new warnings, showing multiple directions that the design of this warning can take. Applying PD lets participants engage more with the subject matter and thus create nuanced designs. Overall, our exploration suggests that PD can provide a suitable, versatile, and simple set of methods that support the creation of design ideas for security-related user interfaces. Users are empowered to critically appraise and adapt security measures that they come into contact with in their everyday life on their own.**

I. Introduction

In the past decade, it has been commonly accepted that IT security measures for end-users need to be designed in a way that users can understand and apply them without unwarranted effort. However, improvements of user interface design for security measures described in previous research were often achieved by experts making educated guesses, based on experiences with users and the results of user studies. This form of design process has limitations, as it is dependent on both the quality of the collected data and experiences as well as the ingenuity of the designer. In this short paper, we propose to take the generation of design ideas one step further by directly integrating end users into the design process of security measures, instead of only benefiting from their experiences indirectly. Through such a deeper involvement of the target audience, the quality of the gathered insights can be improved and even the smallest aspects of a security system can be more easily addressed, compared to iteratively eliciting opinions through formal studies and usability evaluations. This approach is known as Participatory Design (PD) in other HCI and software engineering disciplines and has been successfully applied in diverse contexts [6], [12], [13], [14].

USEC '15, 8 February 2015, San Diego, CA, USA

http://dx.doi.org/10.14722/usec.2015.23011

To explore the utility of this approach in designing usable security measures, we conducted PD workshops with 15 participants on the design of SSL warning messages. These warnings require users to make an informed decision in situations where they may be at risk of exposing sensitive information. Unfortunately, it has been shown that users struggle to understand the problem at hand [1], [3], [4] and thus too easily dismiss the warning. This is commonly referred to as *click-through*. Previous work has shown that users are mostly overwhelmed and do not understand security issues, as the technical background, consequences, and risks are often not communicated well enough [9], [10], [15], [19].

However, Akhawe and Felt [1] were able to show that the warnings can be effective in some instances and that varying the design of a warning does have an effect on click-through. Browser developers are thus currently in the process of changing the SSL warnings, as the warnings in Chrome and Safari have changed significantly in 2014. Felt et al. [7] have used a large-scale experiment to demonstrate the effectiveness of certain elements of SSL warnings, such as different icons and structures. However, the tested prototypes were derived from experts' opinions and experiences. We posit that this approach, while being user-centered and keeping usability in mind, creates an unnecessarily narrow and exclusive focus on issues that the experts consider relevant and may be dependent on a spark of inspiration. In other areas of HCI, it has been shown that PD methods can be used to support the identification of problems and development of suitable options to overcome them.

Our exploration yielded encouraging results that suggest significant benefits from including end-users in security-related design activities. By working with a designer to create an actual SSL warning and brainstorming on existing design limitations, users were empowered to generate diverse proposals that highlight the importance of several design elements in warning messages: using signal colors, having a choice, getting a recommendation, and seeing all necessary information at once without being overloaded with technical detail. We thus find that this method has the potential to overcome existing limitations, arising from including end-users only in testing and evaluation phases before experts make educated guesses how to improve discovered problems. PD can enable the designers of security systems to include the invaluable experience of end-users into their design process, providing important and diverse insights more directly. To the best of our knowledge, this is the first attempt to create new SSL warnings and to

improve widely-used security-related user interfaces for a non-specific audience using participatory design methods.

II. Background & Related Work

In the following, we will briefly discuss related work for participatory design in general as well as the design and evaluation of SSL warnings in particular.

A. Participatory Design

To overcome usability issues, participatory design aims to focus the design process of products or processes directly on users' needs. The concept was originally conceived in the 1970s as part of the scandinavian workplace democracy movement [2]. In a PD process, prospective users of a system are being actively involved in an iterative design process and included as equals in design decisions. Designers cooperate with the users and, as a team, incorporate respective opinions, criticisms, and improvements. The premise of this set of methods is that users of a system have the most experience with an existing problem and can contribute this often implicit knowledge through design decisions, as they may not be able to fully articulate the problem otherwise. The designers cooperate at eye level with the participants and empower them to contribute the necessary information [17].

During this process, communicating opinions and issues in a suitable way can become a problem due to differences in background knowledge between users and designers. Therefore, a "shared language" between the designer and the user needs to be established [5]. Furthermore, users should be treated respectfully and encouraged to tell their true opinion. It is essential to create and maintain a trusted relationship with the participants, creating a comfortable working atmosphere that keeps motivation high [8]. Running a PD workshop is thus comparable to running a focus group, but with the added challenge of creating an actual user interface design in the process.

Possible limitations of a PD process can include a lack of true innovation, i.e. that the results tend to be more of "an evolution than a revolution", as users tend to orient themselves on the existing system [18]. Additionally, the selection of participants from the targeted user group can influence the outcome of the process, as some users may be able to contribute more than others.

However, the application of PD has proven valuable in past work. A large number of previous work has used different PD methods to overcome usability and design issues in diverse application areas. These include the medical domain [13], e-government [6], or elderly people [12]. Most design efforts in the Usable Security research space follow a user-centered design approach and PD has been used to gather requirements [14]. However, we are not aware of any work that empowered users to actively participate in the design process itself.

B. SSL Warnings

Previous work has repeatedly shown that current SSL warning messages have drawbacks in terms of usability and comprehensibility. In a large-scale field study with about 25.4 million seen warnings, Akhawe and Felt [1] found that security warnings are ineffective and tend to be ignored. Bravo-Lillo et al. [3] showed that SSL warnings are perceived to be the most confusing error messages in their study. An earlier study by Dhamija et al. [4] already revealed that 68% of participants clicked through the SSL warning in Mozilla Firefox without reading it. Harbach et al. showed that the textual content of a warning is an important factor that significantly influences comprehensibility [9]. Felt et al. [7] subsequently also investigated the impact of changing design elements of Chrome's existing warning on click-through rates and found that there is room for improvement through changes in the user experience.

Improvements for SSL warning messages design have already been suggested based on observed experiences: Sunshine et al. explored the behavior of 400 users in a study and designed two new warnings based on their findings [19]. The evaluation of their picture-based warnings showed that they performed significantly better than existing warnings. Similarly, Bravo-Lillo et al. [3] stated that most participants do not read warnings precisely, thus, they inferred, showing more text might worsen the problem. Regarding the content, Kauer et al. [11] suggested that instead of communicating only technical facts and details, telling personal risks might lead to increased attention and comprehensibility.

In summary, the work briefly outlined above demonstrates that many aspects of SSL warning design can be improved. In this paper, we exemplarily apply PD methods to this subject in order to show the method's potential to generate design ideas and go beyond the ingenuity of a few designers and researchers.

III. Method and Procedure

A participatory design process can take many forms which depend on the problem at hand. For the scope of this work, we decided to focus on realizing workshops, which are used to explore the problems of a user interface with a smaller group of users in a flexible manner [16]. A main concern of applying PD to an end-user security measure is the breadth of the targeted user group, as virtually anybody can come in contact with an SSL warning when browsing the Internet and is thus a potential PD participant. We limited our exploration to a convenience sample of students and alumni, but aimed to include a range of different views by actively seeking participants from different backgrounds and of both genders.

The workshops were conducted with one designer in the role of a moderator and a group of three participants in a neutral working environment. We repeated the workshop five times with different participants. When planning the setup, we were afraid that the group would quickly become too large, such that individual views would overwhelm the discussion and thus limit productivity. Thus, we begun our exploration with three participants in the first workshop to accommodate enough diverse views while also having enough space for detailed discussion in a suitable amount of time.

Each workshop group was welcomed informally and the purpose of the session was explained clearly, as it was important for the whole process that all participants exactly knew the requirements and what the group would aim for. As a first activity, a brainstorming phase was used to break the ice and introduce the topic. The moderator elicited prior experiences

and opinions of SSL warnings by showing the existing SSL warnings ("untrusted certificate") of Chrome, Firefox, Safari, Internet Explorer, and Opera[1]. In this phase, participants were encouraged to share their uncensored opinions and experiences, much like in a focus group. Positive and negative aspects of the warnings were collected on a flip chart for future reference. As showing existing warnings may bias participants towards the existing designs, we slightly modified the setting of this phase in the fifth and last workshop: Instead of showing example warnings, we asked participants to remember and discuss warnings they had previously seen.

Afterwards, a short and very high-level explanation on the underlying problem of the SSL warning was given by the designer. To ensure that each workshop group got the same information, the designer freely read aloud the prepared explanation, while drawing a sketch of the situation (see Figure 1). In this explanation, *bank.de* was used as a fictional online banking service to frame the problem within a sensitive scenario. This phase served to create a shared language between designer and participants.

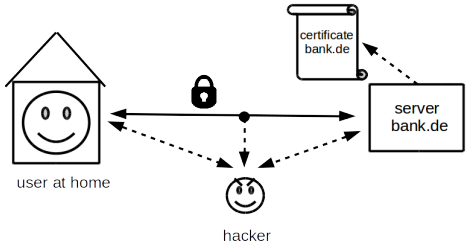

Fig. 1. The explanatory sketch used in the workshops.

Next, the design phase followed. At the beginning, the designer emphasized that the workshop did not aim at facilitating click-through but rather at allowing users to make an informed decision when confronted with an SSL warning. Also, participants were instructed that any ideas were allowed, regardless of being applicable or realistic. Using a mockup software[2], participants were asked to create a new warning. The designer remained largely passive and only helped in case the discussion got stuck or the realization of a design idea was unclear and asked for further details to better understand opinions and issues. During the whole process, the designer created a friendly atmosphere, underlining his role as a team member as opposed to a knowing expert.

In a last phase, participants were asked to critically assess the workshop itself as well as the warnings they had created. To this end, they completed feedback forms and were asked to discuss their results. This meta phase aimed to let participants reflect on the activity.

IV. Results

In the following, we will outline the results we obtained from the different phases of the workshops. We used the first group to pilot the workshop procedure. As only a few small aspects of the explanation were changed, all results will be evaluated jointly.

[1]As shown in the current version of these browsers in October 2014.
[2]https://moqups.com

A. Participants

The workshop sessions lasted between one and a half and two and a half hours. Altogether, fifteen participants took part in the workshops and were divided into five groups of three participants each, each run on a separate day in October 2014. We found this group size to be suitable during the pilot workshop, as their discussions involved all participants, yet, diverse aspects were covered. We maintained this group size for the remaining workshops.

Eight participants were female and ten were students of computer science. They were between 22 and 35 years old. We tried to divide them into groups as homogeneously as possible: Three groups consisted only of CS students, one only of males, one was mixed, and one contained only females. We also ran one group of lawyers with no IT expertise, and one group of mixed genders and varying IT expertise. Participants were snowball-recruited from the authors' acquaintances, which led to some participants knowing each other beforehand. However, we did not find that this influenced the outcome in comparison with other groups.

B. Brainstorming Phase

During the brainstorming phase, many aspects of existing messages were criticized. All but one of the following aspects were discussed in all five workshop sessions (number of sessions in parentheses).

- Text too long, too complicated, too technical, but too little information, situation remains unclear (5)
- Icons do not fit the text and are unclear, but are in general a good idea, visually attracting (5)
- Possibility for user to make a decision (e.g. continue) and the potential consequences should be communicated clearly (5)
- Technical details not helpful, could be hidden (e.g. collapsed and opened on request) (5)
- Interrupting and blocking character, user wants to continue but has to stop and react (5)
- Use of colors is helpful, symbolic colors like green and red (4)

This shows that participants were able to detect many of the commonly agreed upon problems of warning messages in a very short amount of time. They easily identified design challenges and immediately proposed ways to overcome them. The results of this phase are also similar to what a typical focus group session would yield as an output. In our PD workshops, we took this process one step further and let participants come up with their own design. Actually overcoming the problems listed in this phase proved challenging for the participants, but also generated actual examples of how they would like such a warning to be.

C. Creating a Shared Language

The high-level introduction of the underlying problem was well received by participants, after the formulation was slightly changed based on the experience from the first workshop. We observed that participants had a sufficient grasp of the problem in order to complete their task. Unclear aspects were easily resolved between participants and the designer. While we were

Fig. 2. Warning message designed in the third and fifth workshop.

afraid that letting the designer introduce these technical details would promote him to the role of an all-knowing expert, the participants processed the provided information but also accepted that they would be creating the new warning design as a team, without deflecting decisions to the designer.

D. Design Phase

Each group successfully created a warning design. While groups consistently mentioned similar issues (see above), designs varied considerably. Figures 2 and 3 show the warnings created during the workshops. Three of five groups intentionally created very short warnings. Using colors to signal danger or show recommended options was prominent in all designs but barely found in existing warnings. All of them also wanted to include visual elements to avoid displaying a wall of text. Structuring elements like bullets and headlines were also common, containing signal words like "security problem", "error", or "danger". This more concrete and alarming language was a major difference to the existing warnings.

The group that created the design in the top half of Figure 2 focused on suggested actions and also wanted to provide support in case these actions fail. All designs stated a clear recommendation that the user should not continue to the site or should at least be aware of certain risks. At the same time, they also included the option to accept this risk and continue

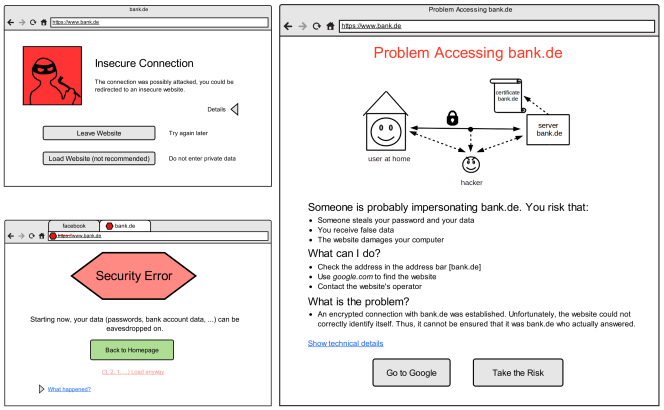

Fig. 3. Warnings created in remaining workshops.

to the desired site. Participants criticized the lack of such an option in existing warnings and called it "censorship", as they did not want to be patronized, especially not by a stranger or a technical device like their computer. One group would only allow to continue after several seconds had passed.

Another interesting aspect is the concrete visualization of a hacker which was used in two of the newly designed warnings, whereas the existing warnings only mention an attacker as "anybody" or use visualizations that confused participants (Firefox). Technical details were also perceived to be important by the workshop participants as they might be "helpful when contacting an administrator", but should be collapsed by default to keep the text short enough to be read completely and to not overwhelm novice users with information which they might not understand. This is already the case in several existing warnings. Another information which the participants perceived to be important were recommendations for the user how to continue after seeing the warning and which consequences this choice may have. These recommendations were given directly in the text or shown indirectly using signal colors like red and green and different button or font sizes. Recommendations were present in some of the existing warnings, but participants found them to be too subtle.

An aspect which some groups mentioned to be influential is the way to address the user. They preferred to use simple and clear words which explain the situation in a high-level way and avoided technical jargon. While the old warnings address the user in a polite and reserved manner, the newly created warnings are formulated more informally and tend to talk directly to the user and, for example, explicitly provide advice: "You should do ... now".

Several participants additionally suggested design changes that went beyond the possibilities of a simple mockup: One participant proposed that one could formulate several versions of the same warning, as the length of the text might be an influential factor: "Each user might have a different motivation to read text in general, such that different lengths might be read in a different amount of time by some users". Furthermore, the problem of habituation was addressed, and another participant proposed that "a message should look different on each occurrence – probably by using different texts and lengths" or "varying content, like icons".

The results also exhibit differences based on group com-

position: Male computer science students created a warning that shared many of the others' qualities, but had noticeably less color (right-hand side of Fig. 3). They also included a diagram which was intended to explain what went wrong. On the contrary, the female CS students created a very colorful warning with only two sentences of text (bottom half of Fig. 2). The mixed-gender group of lawyers (top half of Fig. 2) focused on a clear message and options to deal with this problem, thus including more text.

E. Meta Phase

After each workshop, participants were asked to complete feedback forms about the workshop. Almost all of the participants stated that they were very satisfied with the newly designed message. Only three people were unsure about that, and no one was discontented. Furthermore, they liked their warning much more than the existing ones: All participants rated that they perceived the new warning to be better than the existing warnings on a 5-point numeric scale.

Participants were also asked to state which improvements were most important to them in the new warning. They listed the picture of a hacker, naming the risks, the shortness of the message, increased comprehensibility, advice on what could be wrong and how to proceed, and colorful buttons with recommendations. Also, some participants listed further ideas which they did not mention during the workshop. One idea was to play a sound like a scream or some "bad music" when the message appears. Another participant proposed to fit the warning message somehow to the user's technical background knowledge.

V. Discussion

The work presented in this paper aims to show the potential of including users of everyday security measures into the process of improving their efficacy through design changes. The results outlined above were obtained from an initial exploration of this method, but already contain promising results. As the artifacts created in our workshops sessions are only drafts which need to be refined before they can be evaluated, we will focus on the overall impression and common aspects of the created designs.

Participants were easily able to pinpoint important aspects of the design and generated diverse proposals of how to overcome existing issues. At the same time, they gained an improved understanding for warning messages and how complicated it can be to communicate the necessary information effectively. Almost having finished the design phase, the groups seemed to look at the design process with a more abstract and critical look. One participant stated: "Look, guys, we are just doing what we criticized before!" They discovered that they understood the designers of the old messages, as "It is not easy to summarize [such a complex situation] precisely". Thus, it may be worthwhile to explore PD as a method to educate people by letting them empathize with certain problems that security measures face.

Introducing participants to rather specific and unknown topics like SSL warnings did not seem to be a problem. While it remains unclear how well this would work with different, even more challenging topics as well as different participants, our observations from these workshops are encouraging. Using this approach with a small group and a well-prepared explanation containing only the necessary details while being supported by a knowledgeable designer has potential to also work on other complex topics, such as end-to-end encryption. Choosing a suitable level of abstraction is however a challenge that needs to be addressed when preparing the workshops. In our workshops, using metaphoric examples, such as describing a certificate as some kind of identification card, appeared to be a good approach to make an unknown and technical topic more clear and understandable for participants.

The attempt to reduce bias by not discussing examples of existing warnings during the brainstorming phase in the last workshop lead to a message which did not significantly differ from the other groups' messages (bottom half of Fig. 2). The participants of this workshop relied on memories of SSL warnings they had previously seen. Yet, they also considered other, non-SSL warnings from other areas. This can be useful, as a wider scope may encourage additional ideas. We thus suggest to combine remembering own experiences of arbitrary warnings and similar dialogs with a discussion of screenshots of existing user interfaces afterwards.

Some ideas, such as using a countdown as a temporal constraint, visualizing the attacker as a malicious criminal, or using colors to highlight recommended options and giving direct advice are already known from other applications, but were not present in existing SSL warnings at the time of conducting the workshops. This suggests that designers of warnings might have a restricted view on their product, as they tend to use similar elements when they improve old warnings or create new ones. Inexperienced end-users might look onto the design process from a different, less biased, and more creative way.

Another interesting aspect concerns the selection of participants. We intentionally varied group composition to generate diverse results. Especially factors like gender, age, IT expertise and educational background appear to influence results significantly. We believe that this is an advantage of the PD method, as many workshops with diverse participants can be used to generate a large set of design ideas. Analyzing these will yield considerable input that can inspire the creation of improved user interfaces. Making users actually design a warning empowers them to think through additional aspects of a design that may get lost in more traditional focus group settings.

We also believe that the workshop format we chose was suitable to achieve the intended goal. The sessions were divided into parts of appropriate length and alternated active and passive phases for the participants. The brainstorming phase in the beginning was perceived as a suitable introduction to the topic by participants, regardless of showing old example warnings or not. This active phase lasted for twenty to thirty minutes, which was sufficient to let participants start talking and get to know each other and their opinions as well as generate some initial input for their design. The following phase in which the designer introduced technical backgrounds to create a shared language was a passive phase for the participants, in which they could relax, just had to listen, and could reflect on the introduction. This took about ten to fifteen minutes, which was sufficient too, as the participants

quickly started talking afterwards and had a lot of ideas for the following design phase. This third phase was again an active phase, in which the participants could embrace their creativity, as the designer tried not to intervene. Finally, the meta phase allowed participants to reflect on their activities and how their design compared to existing warnings. This phase can be especially important when planning to revisit the design with the same group in the future.

VI. Limitations

First and foremost, the resulting messages do not represent applicable warnings, as we only aimed to generate design ideas. Thus, we also did not evaluate the created designs for efficacy. So while we believe to provide some evidence of the participatory design process allowing insights into improvements for security-related user interfaces, it remains unknown whether the proposed changes would actually affect the click-through rate. We were, however, able to show that PD methods can generate useful design ideas and insights into users' experiences that go beyond more traditional methods like focus groups.

We also did not even begin to fully exhaust the PD arsenal. Other PD methods or a longer-term involvement of users may generate better and deeper insights. PD is often used as an iterative process which aims at enhancing products over an extended period of time and multiple steps. We aimed to explore a simple option to leverage the power of PD first, before evaluating additional methods.

Finally, only a small group of participants was recruited. They all had an academic background and only a few of them were IT novices. Thus, text comprehension and technical expertise was likely above average. Extending the group of participants to include people with a non-university background and of more diverse age could generate additional ideas and insights, but also cause problems with explaining the technical background.

VII. Conclusion

In this short paper, we presented a first exploration of the applicability of Participatory Design methods for improving security-relevant user interfaces in cooperation with users. As a concrete example, we conducted five workshops in which participants designed improved SSL warning messages. We were able to show that users can be empowered to not only critically appraise a security measure they regularly use but also to create an alternative design. This provides a rich set of improvements beyond what a single designer might be able to come up with using a fairly simple method. Participatory design can thus serve to generate ideas in order to improve security-related interfaces. Working with only 15 participants in five sessions, we were able to provide suggestions for improvements to give users more control, more concrete information, as well as signal colors in SSL warnings. Especially security measures commonly used on the Internet can benefit from this method, as almost any user has experiences that PD helps to tap into.

In future work, we aim to further explore the use of participatory design for security-related user interfaces. As PD is often applied as an iterative process, revisiting the design after some time has passed with the same participants could be worthwhile, especially after previously proposed prototypes have been subject to some form of experimental validation. It could also be interesting to bring several groups together and let them create a joint design. We also suggest that participating in PD workshops has the potential to serve as an educational method, as users can achieve greater understanding for the reasoning behind security measures. The participants in our sessions articulated that they saw how warning message design is hard and may therefore become more empathetic and ready to heed such warnings. Another interesting aspect for future work is how results differ for people with more diverse backgrounds, ages, or interests.

References

[1] D. Akhawe and A. P. Felt, "Alice in Warningland: A Large-Scale Field Study of Browser Security Warning Effectiveness," in *Proc. Usenix Security*, 2013, pp. 257–272.

[2] G. Bjerknes and T. Bratteteig, "User Participation and Democracy: A Discussion of Scandinavian Research on System Development," *Scandinavian Journal of Information Systems*, vol. 7, no. 1, p. 1, 1995.

[3] C. Bravo-Lillo, L. Cranor, J. Downs, and S. Komanduri, "Bridging the Gap in Computer Security Warnings: A Mental Model Approach," *IEEE Security & Privacy*, vol. 9, no. 2, pp. 18–26, 2011.

[4] R. Dhamija, J. D. Tygar, and M. Hearst, "Why Phishing Works," in *Proc. CHI*, 2006.

[5] P. Ehn and M. Kyng, "Cardboard Computers: Mocking-it-up or Hands-on the Future," in *Design at Work*. L. Erlbaum Associates, 1992, pp. 169–196.

[6] A. Ekelin and S. Eriksn, "Citizen-Driven Design: Leveraging Participatory Design of E-Government 2.0 Through Local and Global Collaborations," in *Case Studies in e-Government 2.0*. Springer, 2015, pp. 67–85.

[7] A. P. Felt, R. W. Reeder, H. Almuhimedi, and S. Consolvo, "Experimenting at Scale With Google Chrome's SSL Warning," in *Proc. CHI*, 2014.

[8] J. Gärtner and E. Hanappi-Egger, "Bringing Participatory Design to Practical Application," *Interactions*, vol. 6, no. 2, pp. 13–22, 1999.

[9] M. Harbach, S. Fahl, P. Yakovleva, and M. Smith, "Sorry, I Don't Get It: An Analysis of Warning Message Texts," in *Proc USEC*, 2013.

[10] C. Hochleitner, C. Graf, D. Unger, and M. Tscheligi, "Making Devices Trustworthy: Security and Trust Feedback in the Internet of Things," in *Proc. IWSSI/SPMU*, 2012.

[11] M. Kauer, T. Pfeiffer, M. Volkamer, H. Theuerling, and R. Bruder, "It is not About the Design - It is About the Content! Making Warnings More Efficient by Communicating Risks Appropriately," in *Proc. GI Sicherheit*, 2012.

[12] S. Lindsay, D. Jackson, G. Schofield, and P. Olivier, "Engaging Older People Using Participatory Design," in *Proc. CHI*, 2012.

[13] P. K. Løventoft, L. B. Nørregaard, and E. Frøkjær, "Designing Daybuilder: An Experimental App to Support People With Depression," in *Proc. Participatory Design Conference*, 2012.

[14] N. R. Mathiasen and S. Bødker, "Experiencing Security in Interaction Design," in *Proc. CHI*, 2011. [Online]. Available: http://doi.acm.org/10.1145/1978942.1979283

[15] F. Raja, K. Hawkey, P. Jaferian, and K. Beznosov, "It's Too Complicated, So I Turned It Off! Expectations, Perceptions, and Misconceptions of Personal Firewalls ," in *Proc. SafeConfig*, 2010.

[16] D. Schuler and A. Namioka, *Participatory Design: Principles and Practices*. L. Erlbaum Associates Inc., 1993.

[17] C. Spinuzzi, "The Methodology of Participatory Design," *Technical Communication*, vol. 52, no. 2, pp. 163–174, 2005.

[18] T. Sumner and M. Stolze, "Evolution, Not Revolution: Participatory Design in the Toolbelt Era," in *Computers and Design in Context*. MIT Press, 1997, pp. 1–26.

[19] J. Sunshine, S. Egelman, H. Almuhimedi, N. Atri, and L. F. Cranor, "Crying Wolf: An Empirical Study of SSL Warning Effectiveness," in *Proc. USENIX Security*, 2009.